

# Applications of Natural Language Processing to Archaeological Decipherment: A Survey of Proto-Elamite

by

**Logan Orion Born**

M.Sc., Simon Fraser University, 2018

B.Sc., University of Calgary, 2016

Thesis Submitted in Partial Fulfillment of the  
Requirements for the Degree of  
Doctor of Philosophy

in the  
School of Computing Science  
Faculty of Applied Sciences

© **Logan Orion Born 2023**  
**SIMON FRASER UNIVERSITY**  
**Fall 2023**

Copyright in this work is held by the author. Please ensure that any reproduction or re-use is done in accordance with the relevant national copyright legislation.

# Declaration of Committee

**Name:** Logan Orion Born

**Degree:** Doctor of Philosophy

**Thesis title:** Applications of Natural Language Processing to  
Archaeological Decipherment: A Survey of  
Proto-Elamite

**Committee:** **Chair:** Nick Sumner  
Associate Professor, Computing Science

**Anoop Sarkar**  
Supervisor  
Professor, Computing Science

**Angel Chang**  
Committee Member  
Assistant Professor, Computing Science

**Fred Popowich**  
Examiner  
Professor, Computing Science

**Taylor Berg-Kirkpatrick**  
External Examiner  
Associate Professor, Computer Science and Engineering  
University of California, San Diego

# Abstract

In this thesis, we describe the first-ever large-scale computational analysis of the partially-deciphered proto-Elamite (PE) script. This script was used to write economic accounts which follow a very regular “spreadsheet” structure incorporating many numerals. This sets PE apart from prose corpora which have been considered in prior decipherment work, in ways that both enable and require exploration of new models and methodologies.

In close collaboration with domain experts, we provide a thorough survey of this corpus and answer longstanding questions about its content. We describe novel approaches to multi-modal representation learning, which combine visual information from a VAE-inspired encoder with contextual features from a neural language model. We apply these models to evaluate hypotheses about the script’s underlying character inventory, which remains very uncertain. By analyzing the representations learned by these models, we also deepen our understanding of the relationships between a set of visually complex signs known as complex graphemes, and discover a strict grammar which appears to govern their construction.

We apply a novel variant of the bootstrapping classification algorithm to disambiguate numeric notations with uncertain magnitudes. This enables the first-ever statistical analysis of the corpus’s numeric content, and of the relationships between the numeric and linguistic parts of these documents. Given that numeral notations comprise more than half of the attested corpus, this represents a significant advance in our understanding of the script.

By applying sequence models to study the internal structure of these documents, we independently replicate claims about a structure called the “header”, and adduce new evidence about the size of headers and their distribution across the corpus.

In addition to these main results, we also describe a number of small, focused investigations into word order, the presence of affixal morphology, and other minor features of the texts.

**Keywords:** natural language processing; machine learning; multi-modal representation learning; archaeological decipherment; proto-elamite

# Acknowledgements

I would like to extend my sincerest thanks to Kate Kelley and Willis Monroe, without whose domain expertise this work could never have come to fruition. Thanks also to Barbara Winter for inviting me to volunteer at the Simon Fraser University Museum of Archaeology and Ethnology, where I first worked with Kate and where we conceived of what would become the present work.

I am also grateful to Jacob Dahl, whose monumental efforts in standardising the proto-Elamite sign list and preparing the transliterated corpus served to make this work possible, and whose insightful study of the corpus laid much of the groundwork which I have had the fortune to build upon. Thanks also to the rest of the CDLI contributors who have helped to grow and maintain their invaluable collections.

Thank you to all of the past and present members of the Natural Language Lab at Simon Fraser University, and particularly to Nishant Kambhatla for our many fruitful collaborations on other projects. Thanks also to Carolyn Chen for assisting with the earliest stages of the present work in her time as a volunteer at the lab.

I am grateful to my senior supervisor, Anoop Sarkar, for having been a good friend and the best supervisor I could have hoped for. I am especially thankful to Anoop, to his family, to Willis and Hayley Monroe, and to the Williamsons for looking after me and getting me home safely after my lung chose to explode.

My last and greatest thanks go out to my brilliant and beautiful fiancée, Sara, whose endless encouragement, insightful feedback, and tireless support have brought me through this degree program in one piece.

# Table of Contents

Declaration of Committee	ii
Abstract	iii
Acknowledgements	iv
Table of Contents	v
List of Tables	x
List of Figures	xii
<b>I Background</b>	<b>1</b>
<b>1 Introduction</b>	<b>2</b>
1.1 Notes on Methodology . . . . .	3
1.2 Outline . . . . .	4
<b>2 Proto-Elamite</b>	<b>5</b>
2.1 Scheil . . . . .	5
2.2 Meriggi . . . . .	6
2.3 Friberg . . . . .	7
2.4 Dahl . . . . .	8
2.5 Desset . . . . .	10
<b>3 Data</b>	<b>11</b>
3.1 Overview . . . . .	11
3.2 Data Cleaning . . . . .	13
<b>II Preliminary Results</b>	<b>17</b>
<b>4 Preliminary Results</b>	<b>18</b>
4.1 Descriptive Statistics . . . . .	18

4.2	Sign Frequency and $n$ -Gram Counts . . . . .	21
4.3	Hierarchical Sign Clustering . . . . .	24
4.3.1	Clustering Evaluation . . . . .	25
4.4	LDA Topic Model . . . . .	27
4.4.1	Topic 1 . . . . .	28
4.4.2	Topic 3 . . . . .	29
4.4.3	Topics 4 and 7 . . . . .	30
4.4.4	Topic 5 . . . . .	30
4.4.5	Topic 6 . . . . .	31
4.4.6	Topic 10 . . . . .	31
4.4.7	Remaining Topics (2, 8, and 9) . . . . .	32
4.4.8	LDA Summary . . . . .	32
4.5	Related Work . . . . .	32
4.6	Conclusions . . . . .	33
<b>III Main Results</b>		<b>35</b>
<b>5</b>	<b>Sign &amp; Word Order</b>	<b>36</b>
5.1	$n$ -Gram Entropy . . . . .	37
5.2	$\chi^2$ Tests for Word Order . . . . .	39
<b>6</b>	<b>Complex Graphemes</b>	<b>48</b>
6.1	Methodology . . . . .	48
6.2	Experimental Results . . . . .	52
6.2.1	Additive Composition . . . . .	52
6.2.2	Pairing Consistency . . . . .	53
6.2.3	Analogy . . . . .	55
6.2.4	Complex Grapheme Containment Hierarchy . . . . .	57
6.3	Analysis . . . . .	58
6.4	Related Work . . . . .	61
6.5	Conclusion . . . . .	62
<b>7</b>	<b>Headers</b>	<b>64</b>
7.1	Methodology . . . . .	65
7.1.1	Hidden Markov Model . . . . .	65
7.1.2	Transformer . . . . .	66
7.1.3	Training . . . . .	67
7.2	Experimental Results . . . . .	67
7.2.1	Hidden Markov Model . . . . .	67

7.2.2	Transformer . . . . .	69
7.3	Analysis . . . . .	69
7.3.1	Inter-Annotator Agreement . . . . .	70
7.3.2	Multi-Sign Headers . . . . .	71
7.3.3	Cramér’s V . . . . .	72
7.3.4	Compositionality in Header Signs . . . . .	74
7.4	Related Work . . . . .	76
7.5	Conclusion . . . . .	76
<b>8</b>	<b>Numerals</b>	<b>78</b>
8.1	Background . . . . .	79
8.2	Automated Conversion . . . . .	80
8.3	Disambiguation . . . . .	81
8.3.1	Tablet Summaries . . . . .	81
8.3.2	Bootstrapping . . . . .	82
8.4	Results . . . . .	84
8.4.1	Automated Conversion . . . . .	84
8.4.2	Subset-Sum Analysis . . . . .	86
8.4.3	Bootstrapping . . . . .	86
8.5	Analysis . . . . .	88
8.6	Related Work . . . . .	89
8.7	Interim Conclusions . . . . .	90
8.8	Parallel Coordinates Visualizations . . . . .	90
8.9	Implications to Polysemy and Word Order . . . . .	93
<b>9</b>	<b>Signlist Revisions</b>	<b>95</b>
9.1	Introduction . . . . .	95
9.2	Methodology . . . . .	96
9.2.1	Motivation . . . . .	96
9.2.2	Model Architectures . . . . .	97
9.2.3	Training Details . . . . .	99
9.3	Data . . . . .	100
9.4	Experiments . . . . .	102
9.5	Results . . . . .	103
9.5.1	Modern Scripts . . . . .	103
9.5.2	Cypro-Greek . . . . .	104
9.5.3	Proto-Elamite . . . . .	106
9.6	Related Work . . . . .	108
9.7	Conclusion . . . . .	109

<b>IV</b>	<b>Additional Results</b>	<b>110</b>
<b>10</b>	<b>Fast Cognate Alignment on Imbalanced Data</b>	<b>111</b>
10.1	Background . . . . .	111
10.2	Character-Level Alignment . . . . .	113
10.2.1	Motivation . . . . .	113
10.2.2	Methodology . . . . .	113
10.2.3	Experimental Results . . . . .	115
10.3	Towards Word-Level Alignment . . . . .	116
10.3.1	Methodology . . . . .	116
10.4	Results . . . . .	117
10.5	Conclusion . . . . .	118
<b>11</b>	<b>Additional Results</b>	<b>120</b>
11.1	Linear Elamite Sound Values . . . . .	120
11.2	Kober’s Triplets . . . . .	122
<b>V</b>	<b>Summary</b>	<b>125</b>
<b>12</b>	<b>Conclusion</b>	<b>126</b>
	<b>Bibliography</b>	<b>130</b>
<b>Appendix A</b>	<b>Additional Figures</b>	<b>141</b>
A.1	Dendrograms . . . . .	141
A.2	LDA Topic Stability . . . . .	145
A.3	Precedence Relations . . . . .	151
A.4	Sign Position Heatmaps . . . . .	152
A.5	Complex Grapheme Embedding Spaces . . . . .	205
A.6	Complex Grapheme Containment Hierarchy . . . . .	208
A.7	Sign Clustering . . . . .	209
A.8	Parallel Coordinates . . . . .	209
A.9	Headers . . . . .	226
A.10	Kober Triplets . . . . .	226
<b>Appendix B</b>	<b>Reproducibility Details</b>	<b>229</b>
B.1	Signlist Revision . . . . .	229
B.2	Headers . . . . .	230
<b>Appendix C</b>	<b>Constructing a Validation Set for Numeral Disambiguation</b>	<b>232</b>
C.1	Capacity Measures . . . . .	232



C.2 Sexagesimal Measures . . . . .	232
C.3 Decimal Measures . . . . .	233
C.4 Bisexagesimal Measures . . . . .	234

# List of Tables

Table 6.1	List of models considered in this work. . . . .	51
Table 6.2	Number of compositional graphemes for different similarity cutoffs $k$ . Bold numbers represent cases where the number of compositional graphemes is significantly larger than expected by chance. . . . .	52
Table 6.3	Sample of signs which appear to be compositional in the image LM but not the image recognition model. . . . .	53
Table 6.4	Comparison of pairing consistency for the inner and outer parts of compound signs in 64-dimensional models. Bolded rows represent pairs where the difference between columns is significant. . . . .	55
Table 6.5	Number of analogies which hold between complex graphemes with signs in common, for different similarity cutoffs $k$ . Bold numbers represent values which are significantly larger than expected by chance. . . . .	56
Table 6.6	Sample of analogies which hold in the <code>lm.image+text.64</code> model. . . . .	59
Table 7.1	Contingency table comparing the initial state of a tablet’s Viterbi se- quence against the presence of a header annotation in the tablet meta- data. . . . .	68
Table 7.2	Contingency table comparing predictions from a logistic regression over $\tilde{\mathbf{z}}_{0,n}$ against (left) the presence of a header in the tablet metadata, and (right) the initial state of the Viterbi sequence. . . . .	69
Table 7.3	Agreement (Cohen’s $\kappa$ ) between human and model annotations. . . . .	70
Table 7.4	Mean compositionality of complex graphemes found in expert-annotated headers (Expert), in headers identified using HMM state 7 (HMM), in headers predicted by logistic regression over Transformer self-attention (LR), and in non-initial positions. Bolded values differ significantly from the rightmost column. . . . .	75
Table 8.1	Distribution of readings produced by our automated conversion. A ma- jority of numerals in the corpus can be read using <i>any one</i> of the four number systems. . . . .	82
Table 8.2	Each numeral is associated with a set of features from this list, which we use to train our bootstrap classifiers. . . . .	83

Table 8.3	Distribution of target classes in our numeral disambiguation test set. This set contains every instance of the B class which we were able to manually disambiguate with the help of domain experts; the other classes are kept small to maintain as balanced a distribution as possible.	84
Table 8.4	Number of tablets which unambiguously use more than one number system.	85
Table 8.5	Numeral disambiguation results. In the 4-way setting, we seek to identify exactly which number system is in use for each numeral. In the 2-way setting, we only seek to distinguish C notations from everything else.	87
Table 9.1	Size and character inventories of scripts used for training. PE is undeciphered, and the size of its character inventory remains unknown.	101
Table 9.2	V-measure ( $V$ ) and parameter counts for Cypro-Greek. Best results for each model from Figure 9.8 and Corazza et al. 2022a.	106
Table 9.3	Pairs/triplets of character images which have distinct labels in the working signlist, but which our models merge into single clusters.	106
Table 10.1	Top-1 and top-5 precision of character-level mappings derived from $\alpha$ .	116

# List of Figures

Figure 3.1	Line art (top) and ATF transliteration (bottom) for MDP 26, 102 (P008790). The right half of the image shows the obverse of the text, and the left half the reverse. The offset parts show where the text continues onto the bottom and side of the tablet. Hatching indicates damage. . . . .	14
Figure 3.2	Graph representation for P008815 as parsed from the original ATF (some annotations omitted for space). . . . .	15
Figure 3.3	Sample output from the command-line interface for our converted corpus. Texts are printed in spreadsheet format alongside helpful annotations, such as the value of each numeral in modern Hindu-Arabic notation (see Chapter 8). Comments record working hypotheses about the corpus, and can be used to link related entities together. . . . .	16
Figure 4.1	Log-log plot of sign probabilities (frequency over total sign count) versus rank in the transliterated proto-Elamite corpus. The dotted line shows the idealized shape of the curve when $\alpha = 1.21$ . Orange triangles are complex graphemes, and blue crosses are all other sign types. . . . .	20
Figure 4.2	The 10 most frequent proto-Elamite unigrams, bigrams, and trigrams (top to bottom). In parentheses are given the frequencies of the two unigrams comprising each bigram, and the two bigrams comprising each trigram: note that some frequent $n$ grams are comprised of relatively infrequent $n - 1$ -grams. . . . .	22
Figure 4.3	The 10 most frequent proto-Elamite bigrams and trigrams (top to bottom), limited to signs in Dahl’s (2019) candidate syllabary. In parentheses are given the frequencies of the two unigrams comprising each bigram, and the two bigrams comprising each trigram. . . . .	23
Figure 4.4	Detail of the (a) neighbor-based, (b) HMM, and (c) Brown clusterings showing signs possibly used in anthroponyms. M057, M066, M096, M218, and M371 are considered a stable cluster due to their proximity in all three clusterings. . . . .	26
Figure 4.5	M003~b clusters identically with M263~a in all three techniques. . . . .	26

Figure 4.6	Intertopic distance (measured as Jensen-Shannon divergence) visualized with LDAVis (Sievert and Shirley 2014) using two principal components (PC1 and PC2). Larger circles represent more common topics. . . . .	27
Figure 5.1	Percent change in entropy of the character $n$ -gram distributions for cuneiform and cuneiform-adjacent scripts after shuffling, averaged over 10 shuffles. Larger values suggest stricter ordering principles in the original character distribution, and may reflect a more consistent ordering of characters into words and of words into phrases. . . . .	39
Figure 5.2	Excerpt from Figure A.4 illustrating a partial order over signs occurring together 20 times or more. . . . .	40
Figure 5.3	Heatmap showing average positions of select signs in entries of length 6. For each sign, the brightness of the $n$ th-leftmost cell indicates the frequency with which that sign occurs in the $n$ th position across all 6-sign entries. . . . .	46
Figure 6.1	Architecture for image-aware, multimodal language modeling. . . . .	50
Figure 6.2	Containment hierarchy for a subset of the signs which can occur in CGs. Directed edges point from outer signs to the inner signs they can contain. Note that (excluding self-loops) the graph is acyclic and all edges point from higher nodes to lower ones. Thicker edges represent complex graphemes which are more strongly compositional. Nodes are colored according to modularity class (Blondel et al. 2008) such that nodes are most strongly connected to like-colored nodes. Full hierarchy, showing all signs which occur in complex graphemes, is available in Appendix A.6. . . . .	57
Figure 6.3	Detail from t-SNE decompositions of the GloVe embeddings (left), the image LM (centre) and the image recognition model (right). . . . .	60
Figure 7.1	Illustration of state sequences learned by the HMM. The observed sequence of sign names is shown on the $x$ -axis (truncated to at most 10 signs); the numbers in the cells report the states in the Viterbi sequence. Each color represents a distinct state. (i) HMM state 3 does not suggest the presence of a header, though one is present in the expert annotations; (ii) HMM state 7 suggests the presence of a header, which is present in the expert annotations. . . . .	68

Figure 7.2	Heatmap of $\tilde{\mathbf{z}}_{0,n}$ (mean attention over signs before the first numeral, truncated to length 6) for two tablets, one with a human-labeled header (left) and one without (right). Darker cells indicate stronger attention. . . . .	69
Figure 7.3	Heatmap showing the strength of the association (measured with Cramér’s V) between the first five signs of a tablet and that tablet’s main topic according to an LDA model. . . . .	73
Figure 8.1	Relative values of digits in the main proto-Elamite number systems. $X \stackrel{n}{\leftarrow} Y$ means that one $X$ has the same value as $n$ $Y$ s. Adapted from Englund (2004) . . . . .	79
Figure 8.2	Confusion matrices from classifiers trained using the vanilla bootstrap algorithm (left) and our proposed variant (right). . . . .	87
Figure 8.3	2-way disambiguation results. Confusion matrices from vanilla bootstrapping (left) and our proposed variant (right). . . . .	88
Figure 8.4	Parallel coordinate plots showing all possible readings of every numeral associated with the object sign M001   . . . . .	91
Figure 8.5	Parallel coordinate plot showing all possible readings of every numeral associated with the signs M296 $\diamond$ , M297 $\square$ , and M297~b $\square$ . . . . .	92
Figure 9.1	VAE architecture. This model reconstructs its input from a dense vector encoding. . . . .	97
Figure 9.2	VAE+Neighbor architecture. This model adds the auxiliary task of reconstructing a character image given the encodings of the adjacent characters. . . . .	98
Figure 9.3	VAE+LSTM architecture. This model adds the auxiliary task of drawing the next token given a sequence of encodings for the preceding tokens. . . . .	98
Figure 9.4	VAE+Transformer architecture. This model adds the auxiliary task of reconstructing characters which have been masked by random Gaussian noise. . . . .	99
Figure 9.5	Samples of image sequences from our PE (top), En (middle) and Jp (bottom) datasets. . . . .	101
Figure 9.6	V-Measure on handwritten English. The true character inventory comprises 52 upper- and lowercase letters plus 10 digits. . . . .	103
Figure 9.7	V-Measure on a synthetic mixture of Japanese fonts. The target character inventory comprises 142 hiragana and katakana (46 gojūon, 20 dakuon, and 5 handakuon each). . . . .	104

Figure 9.8	V-measure versus number of clusters for DBSCAN clusterings on Cypro-Greek. We evaluate over the interval $0.1 \leq \varepsilon < 8$ in steps of 0.1. The dotted line represents the true number of signs in the script. . . . .	105
Figure 9.9	Completeness (left) and homogeneity (right) versus number of clusters for DBSCAN clusterings on Cypro-Greek. We evaluate over the interval $0.1 \leq \varepsilon < 8$ in steps of 0.1. The dotted line represents the true number of signs in the script. . . . .	105
Figure 10.1	Top- $k$ precision on Ugaritic-Old Hebrew cognate detection for various thresholds $k$ and values of the smoothing parameter $r$ . . . . .	118

Part I

Background



# Chapter 1

## Introduction

Proto-Elamite is an undeciphered script from the late 4th and early 3rd millennium BCE, recorded on clay accounting tablets first unearthed by Scheil (1900). Despite being known for over a century, and subject to analysis by a host of experts over that time (Scheil 1905; Brice 1963; Meriggi 1971; Amiet 1972; Friberg 1978; Desset 2012; Dahl 2019 *inter conplures alios*) this script remains undeciphered, and is among the largest-known undeciphered corpora remaining from the ancient world.

Past work on this corpus, even that undertaken in recent years (e.g. Kelley 2018), has focused on manual analysis, and attempts at automation have been few and rudimentary (Kelley, pers. comm.). At the same time, much work in computational decipherment has focused on simple alphabetic ciphers or on replicating successful manual decipherments: these tasks are artificially simple compared to *de novo* decipherment of ancient material, and do not trivially generalize to this more challenging setting. Moreover, the proto-Elamite script is speculated to be *semasiographic* to a greater extent than it is *glottographic*; in other words, it may not significantly represent the sound or structure of any particular linguistic utterance, so much as it encodes *symbols* with conventional meanings but no truly “linguistic” content (Damerow 1999; following the terminology of Gorman and Sproat 2023, this would make proto-Elamite a *symbol* system which never developed into a true *writing* system).

This thesis seeks to bridge the gap between computational and archaeological approaches to decipherment. We identify areas where tools from statistics and machine learning could be fruitfully applied to proto-Elamite, and develop new models and resources to modernize and advance the study of this and other ancient scripts.

The severalfold goals of this work are:

- (i) to independently replicate results from historical analyses, some of which have never been replicated due to the small number of researchers in this field, the age and obscurity of the original publications, and the gruelling effort required to manually locate relevant information in the corpus;

- (ii) to develop tools for the exploration and visualization of archaeological corpora, to aid domain experts in hypothesis generation and to facilitate less-technical users in understanding and interpreting data and results;
- (iii) to develop models which automate time-consuming tasks and alleviate the manual effort required to test hypotheses, to help domain experts more rapidly narrow in on promising avenues of analysis; and finally
- (iv) to apply techniques from natural language processing to this script, to identify novel features which can be exploited by decipherers, epigraphers, and other domain experts to advance their work.

This thesis also surveys the challenges which attend work on undeciphered scripts, and demonstrates ways to overcome these challenges using methodologies built around close collaboration with domain experts and multiple replications of results using different models, data, and data modalities.

## 1.1 Notes on Methodology

True decipherment operates in a context where the ground truth is not known. Thus it can be challenging to determine whether a given result is sound, and erroneous claims can easily propagate since they are hard to definitively disprove. Moreover, even well-intentioned analysts are liable to see whichever results they want or expect to see, and may remain unintentionally blind to competing explanations.

We argue that any responsible work on undeciphered data must take explicit steps to anticipate and minimize these issues. To this end, we have sought throughout this work to replicate our own results multiple times across different models, different subsets of the data, and different data modalities. We believe that this approach has helped to denoise and debias our interpretations: results which only hold for a single setting are more likely to be spurious (whether from noisy data or misguided analysis) than those which hold across multiple dissimilar settings.

This work has also been, from its inception, an interdisciplinary collaboration between domain experts and computer scientists. Our principal collaborators have been Kathryn Kelley and Willis Monroe, both trained historians and Assyriologists with experience in cuneiform corpora. Kathryn's research has included graphotactic analysis of proto-Elamite and proto-cuneiform, while Willis has experience in the digital humanities with a focus on the history of religion, including ancient Mesopotamian religion. Beyond contributing their personal expertise, they have also served as intermediaries to Jacob Dahl, the foremost expert on proto-Elamite in the world today. The collective expertise of these collaborators has allowed us to situate computational results in the appropriate historical and cultural

context, and to interpret these results with the appropriate nuance. Together, their contributions have been invaluable to guide and ground every step of this work.

In the other direction, we have sought to make our results accessible to these and other experts by focusing on *interpretable* models and analyses. This has at times meant foregoing the state-of-the-art in favor of more interpretable, albeit less statistically powerful, approaches. This choice is based on the belief that a fully-automated linguistic decipherment of the corpus may be impossible. The apparent paucity of linguistic content in proto-Elamite means that there is little information that could be exploited to determine the underlying language (if the texts are indeed linguistic enough for “the underlying language” to be meaningfully defined); however, the use of logograms or ideograms, the importance of numeric information, and a collection of rich extra-textual features all mean that the texts can likely be well-understood even if the putative underlying language remains unknown. In light of these facts, the features that contribute to a model’s behaviour tend for our purposes to be oftentimes more useful than the model outputs themselves.

## 1.2 Outline

This thesis is divided into five parts. In Part I, we introduce our data and review relevant prior work. Part II describes our initial exploratory analyses, which served to establish familiarity with the corpus and the unique challenges it poses. Part III presents the main bulk of our results, which together offer a survey of the proto-Elamite corpus through a more technical lens than prior work, and give a range of novel insights regarding its possible content. Part IV summarizes additional work, including some ongoing experiments and less technical contributions. Finally, Part V summarizes our contributions and possible directions for future work.

## Chapter 2

# Proto-Elamite

This chapter surveys major contributions from prior manual investigations, in order to establish what is already known about the proto-Elamite corpus and how current intuitions have been developed.

### 2.1 Scheil

The first proto-Elamite tablets were published in Scheil 1900 following their excavation from Susa, in what would today be Iran. This included only two tablets, and it was not until Scheil 1905 that a substantial volume of proto-Elamite text became widely available. Further tablets followed in later editions of the *Mémoires de la Délégation en Perse* (MDP; Scheil 1923, 1935; de Mecquenem 1949), in de Mecquenem 1956, and most recently in Dahl 2019.

As the earliest substantive treatment of these texts, Scheil 1905 offers only “quelques observations générales” [some general observations]<sup>1</sup> about their history and possible content. Already at this time it was clear that “tous les textes de nos tablettes, sans aucune exception, sont des documents de comptabilité” [all of these texts, without exception, are accounting documents], by virtue of the clear ‘spreadsheet’ structure in which they are written. The script itself Scheil takes to be “rigoreusement idéographique” [strictly ideographic] though this is based on his impression of the texts more than any concrete justification.

“Il serait possible, *a priori*, que cette écriture eut une origine propre et un développement autonome” [It would be *a priori* possible for this script to be an autonomous development], observes Scheil, in which case the visual similarities he notes with other ancient scripts would be nothing but “des analogies fortuites, qui ne permettraient pas de conclure à une origine commune ni même à des emprunts considérables” [fortuitous analogies, which would not permit conclusions about a common origin nor of significant borrowings]. Despite this,

<sup>1</sup>All translations in this thesis are the work of the author. We quote the original, non-English publications to ensure that the meaning is preserved in case of translation errors.

Scheil is “porté à croire que ce qu[’il] apelle l’écriture proto-élamite est de même origine que le cunéiforme babylonien” [led to believe that the script he calls proto-Elamite is of the same origin as Babylonian cuneiform], albeit separated from the latter by a long period of independent development; Scheil notes examples where other scripts diverged significantly from a common origin, even where their respective scribes “travaillent, pour ainsi dire, côté à côté” [work, so to say, side-by-side].

Scheil posits that proto-Elamite did not manage to “se dégager [...] de l’idéographie” [divest itself from ideography], and that “conventional” writing (presumably meaning glottographic, capable of faithfully recording language and not merely useful as an accounting technology) developed slowly in Elam by virtue of its position on the “périphérie du monde civilisé” [periphery of the civilised world]. Thus even at this early stage, Scheil acknowledges that proto-Elamite may not be writing in the strict sense, but rather a complex but ultimately non-linguistic symbol system.

Scheil writes at a time when linear Elamite and proto-Elamite were not yet recognized as distinct scripts. There is roughly a 700 year gap between the latest known proto-Elamite texts and the earliest linear Elamite (Dahl 2009), with no evidence for the continuous transmission of this technology between the two periods. Thus, although the visual resemblance between the two scripts is undeniable, and the later may certainly be modeled on the earlier, there is insufficient evidence to treat them as “the same” script as did early scholars (though *contra* this position, see Section 2.5, below). Furthermore, although Scheil assigns this script the name “proto-élamite”, this name reflects a geographic rather than linguistic affinity: there is presently no strong evidence that the script actually represents any form of the Elamite language.

## 2.2 Meriggi

Significant advances in understanding the texts occur throughout the 1970s, among which Meriggi 1971 is particularly notable. This three-volume series undertakes an extensive assessment of proto-Elamite, and includes an attempt to identify and assign labels to every character in the underlying script. (Such “signlists” had previously been published in individual MDP volumes, but had not been combined and standardised across the corpus as a whole.) Today, proto-Elamite signs are labeled with “M-numbers” acknowledging their origin in Meriggi’s signlist, though certain of Meriggi’s signs have been renamed or merged with others as later scholars have come to better understand the texts.

Although parts of the work are of questionable utility (Meriggi, like Scheil, conflates proto-Elamite with the later linear Elamite, acknowledging however that the latter script “è notevolmente diversa da quella delle tavolette” [is notably different from that of the tablets]), in general the work remains a useful summary of the corpus and a survey of insights which remain relevant to the decipherment process. For example, although Scheil

“ha anche il merito di aver notato il carattere fonetico di alcuni segni” [has the merit of having noted the phonetic character of some signs], it is Meriggi who suggests a complete inventory of possible “syllabic” signs and undertakes the first analysis of suspected names in proto-Elamite. To this day it remains unclear whether these signs are truly syllabic, or whether they are truly used to write names. However, it *is* clear that they represent a functionally unique subset of the signary, and have therefore been of particular interest to subsequent decipherment attempts, including our own.

Meriggi’s overview also includes important notes about the expected content of the texts, including what kind of content should *not* be expected (“le [forme verbali] in questi brevissimi « testi » potrebbero anche mancare del tutto” [verbal inflections may thus be entirely absent from these extremely short ‘texts’]).

Meriggi’s use of Sumerian sign names to label some proto-Elamite signs (such as TUR for the sign now called M370) highlights the visual similarities between these two scripts. Some of the signs thus named have since been found to exhibit functional parallels to their perceived proto-cuneiform equivalents (Friberg 1978; Kelley 2018) and may thus reflect early borrowings or evidence of a common origin.

## 2.3 Friberg

Friberg (1978) explores the history and origins of Babylonian mathematics, particularly its relation, if any, to earlier Sumerian commercial accounts.

A mathematician rather than an Assyriologist, Friberg demonstrates that some proto-cuneiform accounts include a summary line counting the total number of items recorded in the preceding text. The summation yields the correct value only when the numbers are read in base-10, where a “cup” sign tallies the ones and a “disk” tallies the tens.

At first the values of these signs appear certain, as the same relationship is found across multiple tablets. However, Friberg notes a separate set of texts where the summaries only equal the expected value when read in base-6, so the disk tallies sixes instead of tens. The reading of 6 disks to one cup parallels the later Sumerian units of 6 *bariga* to one *ban*, used for measuring grain; and indeed, the texts in question contain a sign resembling early forms of Sumerian SZE, later used to denote barley rations.

Observing that some numerals in “SZE” texts contain 8 disks, Friberg concludes that the system cannot be purely base-6, as in that case no more than 5 disks should ever occur. Since no texts contain 10 or more disks, the next largest digit (the so-called “big disk”) likely counts 10 disks; big disks occur in groups of 1 or 2, so the next digit (“big cup”) equals 3 big disks. Proceeding in this way, Friberg ascertains likely values for the complete “SZE-system” in proto-cuneiform, including digits representing small fractions.

When the same technique is applied to proto-Elamite (this time simplified and formalized as solving a series of linear equations), readings for the proto-Elamite numerals are obtained.

These readings produce valid summations across multiple tablets, and in fact exhibit the same intervals between digits as the proto-cuneiform SZE-system (the shapes of the signs are also identical). Moreover, in the same way that the proto-cuneiform version of this number system occurred alongside the sign SZE, the proto-Elamite version occurs alongside a sign GUR (called M288 in modern transliterations). GUR being a grain measure in later Sumerian has led to the assumption that this number system also counts grain (likely in the form of rations) in proto-Elamite. M288/GUR is one of several proto-Elamite signs which are now partially understood by means of their apparent cuneiform or proto-cuneiform parallels.

Friberg’s analysis reveals the presence of multiple distinct number systems in proto-Elamite, which like the proto-cuneiform systems sometimes use the same signs with different relative values, and which all use mixed radices. In the same way that the SZE system (now called the *capacity* system in proto-Elamite) had an apparent association with grain rations, we now know that other number systems associate with their own categories of object (the decimal system being used for animals and laborers, sexagesimal for discrete objects and perhaps some humans, and bisexagesimal for discretized grain products such as rations; Englund 2004: 107).

## 2.4 Dahl

The most recent body of work on this corpus has been undertaken by Jacob Dahl and his students. Dahl 2019 includes the most recent and thorough survey of the corpus, and we encourage the reader to consult that volume for a deep look at this script through the Assyriological lens.

In earlier work, Dahl (2005a) discusses parallels between proto-Elamite and proto-cuneiform signs which, based on their meanings in later cuneiform, likely represent livestock. The work demonstrates that alternations between pairs of signs (either in successive entries, or in parallel contexts in different texts) reflect semantic features which are useful for decipherment, such as different ages or genders of the same animal. These alternations yield analogical relationships which can allow the meanings of several signs to be inferred from the meaning of just one sign. Like Friberg, Dahl also exploits information from the numerals. For example, one tablet records amounts of both M346 and M348; though the *text* of the tablet summary only lists M346, its *value* equals the total of both the preceding M346 and M348 counts. Thus it appears that M348 could, in this context, be a subtype of M346, which is recorded separately in the body of the text but lumped in with the more generic sign in the summary.





Dahl 2005b describes a set of proto-Elamite signs called “complex graphemes”, which appear to be formed from combinations of signs written side-by-side, one inside the other, or one in between mirrored copies of the other. According to Dahl’s analysis, certain kinds

of graphical composition are associated with certain functions, for example “A+B+A” signs (where one sign is written between mirrored copies of another) he takes as likely to represent households in charge of accounts. Dahl also discusses the possible meaning and function of “complex capacity signs” formed by inscribing a numeral inside what is likely a depiction of a vessel or measuring device.

Dahl et al. 2018 discusses the administrative role of the proto-Elamite texts, with a focus on terminology relating to workers and overseers. A doctoral dissertation by one of Dahl’s students (Kelley 2018) expands on this work with a focus on ration texts. “Whereas signs depicting humans or human body parts in other early writing systems have often been used as anchors for decipherments”, Dahl et al. (2018: 15) note that “Proto-Elamite is devoid of any signs that depict humans or human body parts, except for a few early loans from the related proto-cuneiform writing system”, which makes it challenging to identify terms representing human laborers. In place of overtly iconic glyphs, their analysis must therefore rely on other sources of evidence: loans from proto-cuneiform; correlations between numerals (both in terms of value, and the number system used) and different kinds of counted objects; and structural properties of the texts which may reflect properties of the individuals recorded therein.

Of particular note is their conclusion that depictions of tools may sometimes stand in metonymically for the laborers which use those tools; thus a text which appears to count yokes may in fact count amounts of grain rationed to laborers whose work *employs* a yoke. Care should thus be taken when suggesting readings for signs which appear ‘obviously’ iconic.

So-called “roster” texts appear to record groups of workers or other individuals; many such texts record certain signs in a consistent order (M317s before M054s before M003s...; Hawkins 2015). Sheep and goat texts record “more important” animals first (Dahl 2005a); if a similar practice applies in roster texts, the signs at the beginning of the ordering may thus reflect higher-status individuals (perhaps the foremen of the work groups in question).

One of the most important recent contributions, particularly for our purposes, has been the ongoing effort by Jacob Dahl to standardize the proto-Elamite signlist. These efforts have resulted in a more-or-less consistent labeling scheme for all of the signs in the known corpus. The working sign names include two kinds of subscripts (or tilde-annotations) to identify signs which may be related to one another in some way. A numbered annotation denotes that one sign is subtly distinct from another in appearance, but likely equivalent in meaning: see for example M310~1  (also written M310<sub>1</sub>) and M310 . By contrast, lettered annotations denote signs which share a similar level of visual similarity, but which have not been established to share the same meaning: consider M311  and M311~b  (equivalently, M311<sub>b</sub>). Importantly, the meanings of these signs may be *related* to one another, but have not been established as *identical*. A key step in advancing the understanding



of this script is to find concrete evidence that these and other variants are or are not truly equivalent in meaning.

## 2.5 Desset

Recent work by François Desset is notable for pursuing a more traditionally linguistic approach to decipherment, which places particular emphasis on so-called “anthroponyms” which have long been speculated to represent syllabically-written names. Desset 2012 includes multiple tables of these putative names, organized into groups with matching substrings, as well as a list of the main signs used in anthroponymic sequences. Of interest to future studies is “l’hypothèse selon laquelle les anthroponymes notés a Suse, Tépé Yahya et Tépé Ozbaki pouvaient être construits à partir de langues différentes” [the hypothesis that anthroponyms identified at Susa, Tepe Yahya, and Tepe Ozbaki may come from different languages], which would accord with the use of some (few) unique signs in tablets from these locations.

Most recently, Desset et al. 2022 claims a complete decipherment of linear Elamite, and suggests it may be possible to “proceed in a regressive way [...] trying to apply these “readings” to their graphic counterparts in the earlier PE writing [...] The same signs may have been used with similar or identical phonemic values to record the names of the persons involved in the transactions and administrative work documented in the late 4th millennium BCE PE tablets”. This claim is advanced with no accompanying evidence that it actually yields readable names (in any language) when applied to proto-Elamite, and initial investigations suggest that some of the resulting “names” begin with unlikely sound sequences such as *m-la* or *m-t* (Kelley et al. 2022b). Neither do the authors address how these sound values could have been preserved across the ca. 700 year gap between the latest proto-Elamite texts and the earliest linear Elamite. Thus, while this work offers a new avenue for hypothesizing about the sound values of some proto-Elamite signs, its utility presently remains uncertain, and even in the best-case scenario this line of inquiry leaves the (presumably large) logographic portion of the script untouched.

# Chapter 3

## Data

### 3.1 Overview




The Cuneiform Digital Library Initiative<sup>1</sup> (CDLI) hosts digital images and transliterated copies of all published proto-Elamite tablets, as well as some unpublished tablets and seals in personal and museum collections. This corpus is substantially complete, though it continues to grow slowly as new texts are unearthed or published. The corpus contains approximately 1581 texts, though the exact number varies depending on whether one includes damaged artifacts which are unreadable, artifacts with numerals but no text, and artifacts which may actually bear the related proto-cuneiform script.







We refer to individual texts either by citing their original publications, or using the “P-numbers” which uniquely identify them in the CDLI database. For example, the tablet in Figure 3.1 is the 102<sup>nd</sup> text published in MDP 26 (Scheil 1935), so it may be referred to as MDP 26, 102. The same tablet is recorded in the CDLI database as P008790.<sup>2</sup> Occasionally, a broken tablet will have originally been published as multiple fragments, which have since been joined together to restore a more complete text. In these cases, each fragment will have a unique publication, but the CDLI database will only contain the joined text: for example, P008105 combines fragments which were originally published as MDP 6, 316, MDP 6, 324, MDP 6, 332, MDP 26S, 336, and MDP 26S, 335. For the sake of brevity, and to clarify that we always consider the most complete available text, we will primarily refer to texts by their P-numbers.


The proto-Elamite script is read from right to left, top to bottom, like the related proto-cuneiform script. In later cuneiform, the direction of writing changed, but the orientation of the signs did not, so that later cuneiform signs appear to be rotated 90 degrees with respect to their archaic forms. This led to a practice whereby proto-cuneiform texts are often published at a 90 degree angle, so that the signs share the same orientation as their later

<sup>1</sup><https://cdli.mpiwg-berlin.mpg.de/>

<sup>2</sup><https://cdli.mpiwg-berlin.mpg.de/artifacts/8790>

counterparts. This practice has sometimes carried over to publications of proto-Elamite, even though this script never underwent such a rotation. In this work, we choose to depict signs and tablets in their *original* orientations, as they would have been written and read by the original scribes. For example, the sign written as  in other publications and depicted as such in the CDLI tablet images will be shown as  in this work. This makes more iconic signs easier to interpret, such as M417~g  which appears to depict an equine head.

Tablets are transliterated using the ASCII Transliteration Format<sup>3</sup> (ATF) shown in Figure 3.1. Every sign which is believed to represent regular text is assigned a unique name beginning with the letter M (for Piero Meriggi, from whose work in Meriggi 1971 the current sign names are derived). Signs believed to represent digits are given names starting in N. Some sign names include a ~ followed by a string of letters or numbers, as in M157~a . We call the part preceding the ~ the *base name* of the sign, and the part after the ~ the *variant*. Signs which share the same base name also share visual similarities to a sufficient degree that experts speculate they may be alternative ways of writing the same underlying character. The text after the tilde determines how certain experts are about this equivalence: numbered variants are likely the same sign (M024~1  and M024 ) , while letter variants represent cases where experts remain deliberately agnostic (M157~a  and M157 ). There are also a few cases, such as M045~b  , where a sign is labeled as a variant, but there is no other sign with the same base name for it to be a variant *of*. Presumably, M045 did exist in a prior version of the corpus, but was later reinterpreted as being a different sign and was relabeled in the transliterations. This highlights how the proto-Elamite corpus continues to evolve together with our understanding of the script. To guarantee consistency in our results, our experiments use static, offline copies of the CDLI corpus downloaded in June 2018 (Chapters 4–7) and then updated in October 2022 (Chapter 8 onward).

To produce a transliteration, a text is read from right to left, while the ASCII names of the signs are written down left to right. Digits are always parenthetised and prepended by the number of times they are repeated (e.g.  is transliterated as 3(N01) and not N01 N01 N01). This is done even when the digit is only written once (e.g. 1(N01)). The resulting transliteration is divided into numbered lines, where each line represents a logical division of the document as understood by experts. Most lines comprise a single “entry”, with a comma delimiter to separate the text of that entry (apparently describing an object being counted) from the following numeral. Square brackets are used to denote that a span of text is damaged, while a # or ? following a sign marks that particular sign as damaged. If a sign is entirely unreadable, it will be transliterated as X (if it is part of the text) or N (if it is part of a numeral). Breaks of indeterminate length are transliterated with an ellipsis.

<sup>3</sup><http://oracc.museum.upenn.edu/doc/help/editinginatf/cdliatf/index.html>

Comments and other annotations may be added on their own lines, prepended with # or @.

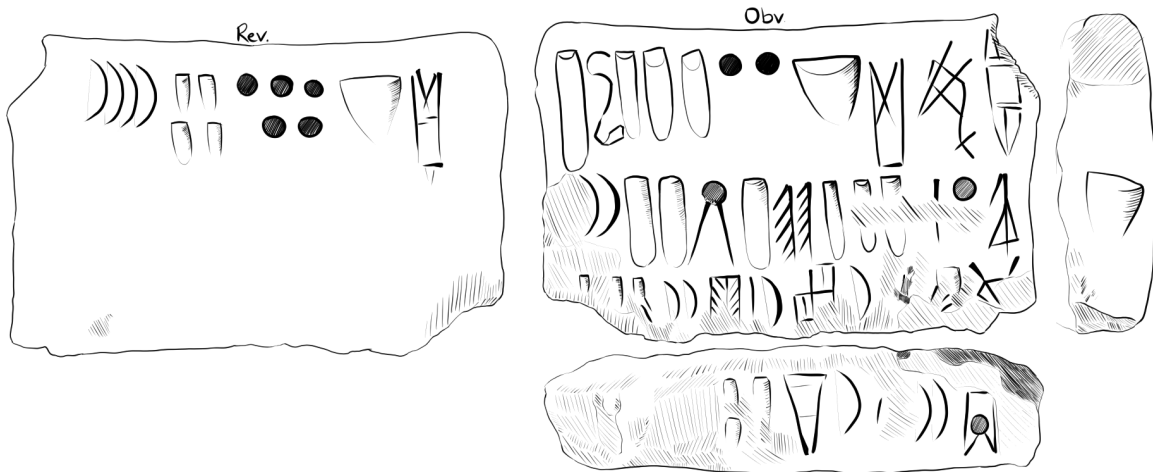
## 3.2 Data Cleaning

To facilitate our work, we convert the raw ATF transliterations into a more structured format. We first parse ATF documents into a kind of abstract syntax tree, with nodes representing spans of text and labeled edges recording containment and adjacency information. Figure 3.2 shows an example obtained by parsing the text of P008815. Each node has attributes which describe the type of text it represents (single sign, digit, numeral, entry, etc) as well as any annotations which were attached to that text in the ATF. (For reasons of space, most of these attributes are omitted from the figure.) We include nodes representing explicit divisions of the text (such as *segments*, which is our term for a contiguous span of text that is clearly delimited by digits or by the beginning or end of the document) as well as *implicit* divisions (such as headers, which experts believe to be distinct from the following entry but which are not visually separated from that entry in any way). This means that the resulting structure is a directed acyclic graph, but not a tree, as some nodes have multiple parents (for example, a sign can be part of the first segment and also part of the header). This graph structure enables us to easily traverse the text of a tablet at various levels of detail (sign-by-sign, entry-by-entry, etc) and to move from one level of detail to another by following the parent/child relationships. Each node is assigned a unique ID (UID) which encodes the type of information that node represents and the location of that information in the original ATF: for example, P008001:6:num is the numeral on line 6 (zero-indexed) of the ATF for text P008001, and P008001:6:sgn:3 is sign 3 (also zero-indexed) within that numeral (in this case, this is the digit 7(N01)).

We parse the entire ATF corpus and serialize the resulting objects as a list of UID strings and object-attribute-value (OAV) triples. Some texts do not strictly follow the ATF specifications; in these cases we correct the transliteration after confirming our proposed corrections with domain experts. We populate a SQL database with the serialized data. The database also includes comments and other attributes describing every named sign, extracted from unpublished notes<sup>4</sup> graciously shared by J. Dahl (Unpublished).

This conversion offers a number of benefits over the original ATF files. It is occasionally desirable to treat the beginning of a text as a single contiguous span, and to ignore the artificial separation between the header and the first entry. Our conversion includes separate objects describing the header, the following text, and the concatenation of the two, and any

<sup>4</sup>We are grateful to Carolyn Chen for helping to sanitize the data extracted from these notes during her time as a volunteer at the SFU Natural Language Lab.



```

&P008790 = MDP 26, 102
#atf: lang qpc
@tablet
@obverse
1. M305 ,
# header
2. M056~f M288 , 1(N34) 2(N14) 3(N01)
3. M051~a , 1(N01)
4. M124 , 1(N14)
5. M001# , 3(N01)
6. M041~j , 1(N01)
7. M367 , 2(N01) 2(N39B)
8. M320~m M387~c# x , 1(N39B)
9. M205~c# , 1(N39B)
10. x , 2(N39B)
11. M387~c#? , 2(N01)#
@reverse
@column 1
1. M289~d , 2(N39B)
2. x , 1(N39B)
3. M131~e# M388#
@column 2
1. M288 , 1(N34) 5(N14) 4(N01) 4(N39B)
@top
1. 1(N34)?

```

Figure 3.1: Line art (top) and ATF transliteration (bottom) for MDP 26, 102 (P008790). The right half of the image shows the obverse of the text, and the left half the reverse. The offset parts show where the text continues onto the bottom and side of the tablet. Hatching indicates damage.

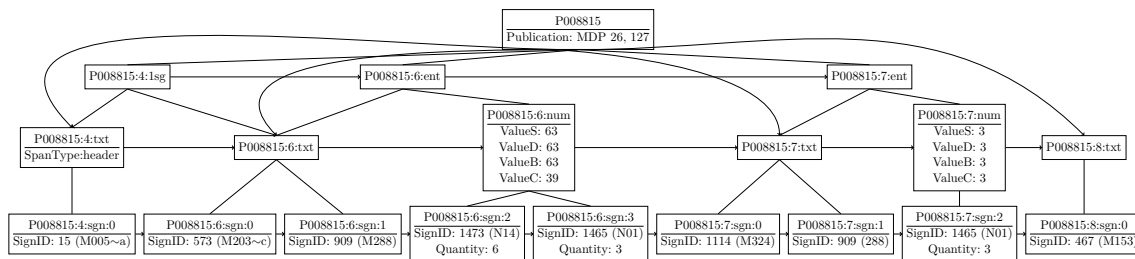


Figure 3.2: Graph representation for P008815 as parsed from the original ATF (some annotations omitted for space).

one of these can be retrieved on-the-fly using its UID. This removes the need to explicitly detect and merge headers with the following text at analysis time.

Some of our experiments involve relabeling signs with alternative names (Chapter 9). In the ATF, this can only be achieved by overwriting the original name with the revised name; this leads to a proliferation of alternative copies of the corpus which are inconvenient to keep up-to-date with one another. Following our conversion, however, it becomes trivial to add any number of alternative names to a token simply by adding new object-attribute-value records to the database. The underlying tokens can still be retrieved using their original UIDs, and the original sign names are not overwritten.

Using structured representations also helps to simplify analyses which deal with complex graphemes (Chapter 6), which are characters made up of multiple parts written within one another or otherwise ligatured together. Depending on the application, these can either be considered as atomic tokens, or as structured objects which combine several tokens. Our conversion assigns distinct identifiers to a complex grapheme and to its component parts, which simplifies the process of switching between these two views.

Finally, the database where the converted objects are stored provides a convenient and centralized location to aggregate the results from our analyses, both in the form of comments attached to particular UIDs and as additional OAV triples. This makes it easy to check what our past analyses have suggested about a given token or sign and to compare results from different techniques. We add comments, each of which is linked to a set of UIDs, to make an explicit record of the relationships which we identify between signs, texts, and other objects of interest. We also provide a command-line interface, which we call CLEE (Command-Line Environment for proto-Elamite; Figure 3.3), to automate common queries and simplify the process of adding these comments.

# CLEE

CommandLine Environment for proto-Elamite  
Type help or ? to list commands.

```
CLEE > desc P008001
```

P008001

P008001 is the UID for MDP 06, 201

	TEXT	NUMERAL	VALUE(S)	LINE
H	M157			:4
	M319 M032	1(N14) 7(N01)	= 17 xN01 (SDB)	:6
	M321-J	1(N01)	= 1 xN01 (SDBC)	:7
	M321	2(N01)	= 2 xN01 (SDBC)	:8
	M005	4(N01)	= 4 xN01 (SDBC)	:9
	M376	1(N34) 3(N01) 1(N08)	= 63.50 xN01 (S)	:10
	M310 M376	1(N14) 9(N01)	= 19 xN01 (SDB)	:11
	M149-A	9(N01)	= 9 xN01 (SDB)	:12
	M381 M149~A	2(N01)	= 2 xN01 (SDBC)	:13
	M218 M039~B			:14

```
CLEE > desc M388
```

VARIANTS

M388 (sign id 1364) has variants M388-C

COMMENTS

- can M124 also function as owner? Also M388 etc.?  
- see also M124
- M141 perhaps graphical (late) variant of M388?  
- see also M141
- M141-a perhaps graphical (late) variant of M388?  
- see also M141-A
- Desset 2012 (Premières écritures iraniennes): "Dahl 2009, p. 25, fig. 1 estime que la paire M305 M388 (parfois accompagnée d'un autre signe) représente dans certaines séquences non pas le début d'un anthroponyme." mais la désignation d'un organisme/bureau précédant la notation de l'anthroponyme lui-même.  
- see also M305

ATTESTATIONS

M388 is attested 729 times in 375 texts:  
P008490 (x25), P008105 (x17), P008185 (x17), P008318 (x15), P272825

Figure 3.3: Sample output from the command-line interface for our converted corpus. Texts are printed in spreadsheet format alongside helpful annotations, such as the value of each numeral in modern Hindu-Arabic notation (see Chapter 8). Comments record working hypotheses about the corpus, and can be used to link related entities together.

## Part II

# Preliminary Results



## Chapter 4

# Preliminary Results

This work began with a series of exploratory analyses originally reported in Born et al. 2019. That publication sought to introduce the proto-Elamite dataset to the computer science community, provide up-to-date descriptive statistics (such as sign counts, which had not been updated since Dahl 2002), and verify that traditional NLP approaches were effective on this data.

The latter point warrants particular attention. Recall that experts have questioned the degree of linguistic information encoded by the proto-Elamite script, and have argued that it may be primarily semasiographic (i.e. not a true *writing* system, but only a symbol system). If these intuitions are correct, then we should not take for granted that this data will be amenable to analytic methods developed for modern writing systems. Before proceeding with this work, we therefore wish to establish that NLP models are capable of replicating results from prior manual analysis. This will help to establish whether it is necessary to develop entirely novel models for this corpus, or whether the data is sufficiently language-like to submit to existing methods. This chapter will consequently focus on simple data exploration and on *applications* of existing techniques to a new corpus; we will introduce more novel computational results starting in Part III of this work.





### 4.1 Descriptive Statistics

We have previously published descriptive statistics, such as sign and token counts, in Born et al. 2019, with additional information available in the accompanying Python notebook. This section updates those numbers based on the more recent copy of the corpus which was used to initialize CLEE. We emphasize, however, that the proto-Elamite corpus continues to evolve as we gain new insight into the content of these texts (see throughout the following chapters for examples where we propose corrections or alterations to the current transliterations). For this reason, even these updated counts should be treated as approximations, which give insight into the general distribution of categories but which are only truly accurate for the corpus as it exists at the time of writing.

**Sign Types** The corpus used to initialize CLEE employs 1693 distinct sign names, which can be further broken down into 293 basic signs, 44 digits,<sup>1</sup> 48 numbered tilde-variants, 1061 lettered tilde-variants, and 247 complex graphemes (signs which are visually compositional, comprising two or three simpler signs either ligatured or written one inside the other). The complex graphemes can be further distinguished according to their number of component parts and the pattern in which these parts are repeated: schematically, we observe 9 A+A type graphemes (the same sign ligatured with or written inside of itself), 211 A+B, 17 A+B+A, 3 A+B+B, and 7 of type A+B+C.

The corpus comprises 35496 tokens, of which 12081 are digits (where we count a cluster of repeated digits like 2(N01) as a single token; thus this is specifically the number of *transliterated* digit tokens) and 1159 are complex graphemes (if we treat complex graphemes as non-atomic units, and count each of their components as a unique token, the overall total instead rises to 36740).

**Hapaxes** There are 736 hapaxes (i.e. hapax legomena, signs which are only attested a single time), of which 117 are complex graphemes. This accounts for 43% of the overall signary, and 47% of the complex graphemes—a roughly similar ratio in both categories. These signs are of key interest, as their rarity makes them challenging for human analysts to interpret. This rarity also makes them challenging for computers to model, and one contribution of our work (see esp. Chapters 6 and 9) is to develop multi-modal language models which are able to learn usable embeddings for these rare signs without the need to replace so many (if any) with a generic “unknown” token.

Many of these rare signs are labeled as possible variants of a more common sign (e.g. M217~f  and M217 ) , or bear some visual resemblance to a more common sign (e.g. M486  and M048~e  ). We speculate that a portion of these are in fact semantically-equivalent graphical variants or mislabeled instances of a more common sign type, a hypothesis which will reappear throughout later chapters (particularly Chapter 9).

**Zipf’s Law** Word frequencies in natural language corpora have been observed to approximately follow a power law, such that the frequency  $n$  of the  $r$ th-most-common word is roughly

$$n \propto \frac{1}{r^\alpha}$$

<sup>1</sup>This is fully 5 signs short of the number of digits reported in Born et al. 2019. The difference owes to the removal of some very short texts which are likely instances of the proto-cuneiform script rather than proto-Elamite. These were included in the original corpus because the two scripts can be hard to distinguish in some settings, particularly on very small clay tokens which bear only a single sign or numeral. Although such texts were included in the corpus used for our earliest publications, we do not believe their presence had a significant impact on any of our results, as we focused on non-numeric signs which do not generally occur on the documents in question.

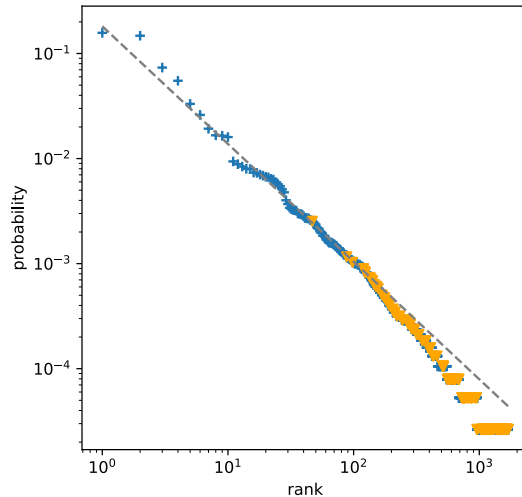












Figure 4.1: Log-log plot of sign probabilities (frequency over total sign count) versus rank in the transliterated proto-Elamite corpus. The dotted line shows the idealized shape of the curve when  $\alpha = 1.21$ . Orange triangles are complex graphemes, and blue crosses are all other sign types.



for some scale term  $\alpha$  close to 1 (Zipf 1935; Moreno-Sánchez et al. 2016). This relation has been termed *Zipf’s Law*. So-called “Zipfian distributions” have been claimed to arise for a wide range of languages and language modalities, prompting some authors to use the presence of this distribution as a test for linguistic content. However, Farmer et al. (2004) note that similar frequency distributions can also arise from a variety of *non*-linguistic processes, implying that a Zipfian frequency distribution is not, by itself, sufficient evidence that a symbolic system encodes language. In the other direction, the *absence* of a Zipfian frequency distribution may suggest that a symbolic system is *not* linguistic, though Moreno-Sánchez et al. (2016) show empirically that some natural language corpora exhibit notable deviations from an ideal Zipf relation, so only extreme deviations should be taken as evidence against linguistic content.

The current working transliterations for proto-Elamite *do* appear to follow a Zipf distribution, with an exponent of approximately  $\alpha = 1.21$  (Figure 4.1). This exponent is within the range expected for natural language texts based on the work in Moreno-Sánchez et al. 2016. Thus, despite the prevailing view that these texts may lack linguistic content, in terms of raw character frequency they are not immediately distinguishable from true language (though the same may not be true at the level of character *n*-grams, see Section 4.2). This suggests that it may be possible to analyze this corpus using tools from natural language processing, even if there is no natural language to be found within.

## 4.2 Sign Frequency and $n$ -Gram Counts

$n$ -gram frequency is another useful datapoint for understanding the overall content of the corpus and for building a more nuanced understanding of sign use (Dahl 2002; Kelley 2018). Figure 4.2 shows the most common proto-Elamite unigrams, bigrams, and trigrams. These counts exclude  $n$ -grams containing numeric signs or broken or unreadable signs (transcribed as X or [ . . . ]);  $n$ -grams which span a boundary between entries are also excluded. Note the sharp drop-off in frequency from the most frequent signs to the rest of the signary; similar results were presented in Dahl 2002.

The most common unigrams include “object” signs and signs belonging to Meriggi’s syllabary. The most common object signs are M288  (a grain container), M388  (“person/man”), M124  (a person/worker category paralleling M388), M054  (a yoke, usually indicating a person/worker category or animal), M297  (possibly “bread” or “beer”), and M346  (“ewe”). The common syllabary signs are M218 , M371  (which may double as an object sign/worker category), M387  (identical to the digit with value “100”), and M066 .

The  $n$ -gram counts reveal the scale at which complex sequences of information are repeated across tablets. At the token level, there are over 1600 trigrams excluding numeric signs; among these, we find only 11 trigram types which are repeated at least 5 times total, and two of these end in the “grain container” sign M288  and are therefore best parsed as spanning a “word” boundary. 52 additional trigram types are repeated three or four times in total, leaving the great majority (98%) of trigram types to appear only once or twice.<sup>2</sup> The most frequent trigram, M377~e M347 M371  (found 17 times per Figure 4.2), appears in no more than about 1.5% of the texts. Even among bigrams, the most common can only occur in up to 3.2% of texts.

Impressionistically, this looks like a lesser degree of repetition than would be expected for linguistic data, where to take an English example, common bigrams like “of the” are frequent both within and across texts. External comparisons are needed to establish whether this impression is correct, but such comparisons are not straightforward. Third millennium Sumerian or Akkadian accounting tablets are reasonable corpora to compare against, but these are generally available only in transliteration (using sign *readings*) while proto-Elamite is transcribed (using sign *names*). This distinction makes  $n$ -gram counts from the two corpora incomparable without further work to transform the data. The training data from the shared task in Zampieri et al. 2019 offers a rare exception where cuneiform text is repre-

<sup>2</sup>These numbers treat tilde-variants as distinct signs, following the working hypothesis among proto-Elamite specialists. Collapsing variants together does not appreciably change these results, however, as it only increases most trigram counts by 1 or 2 instances. A similar result holds for bigram counts.

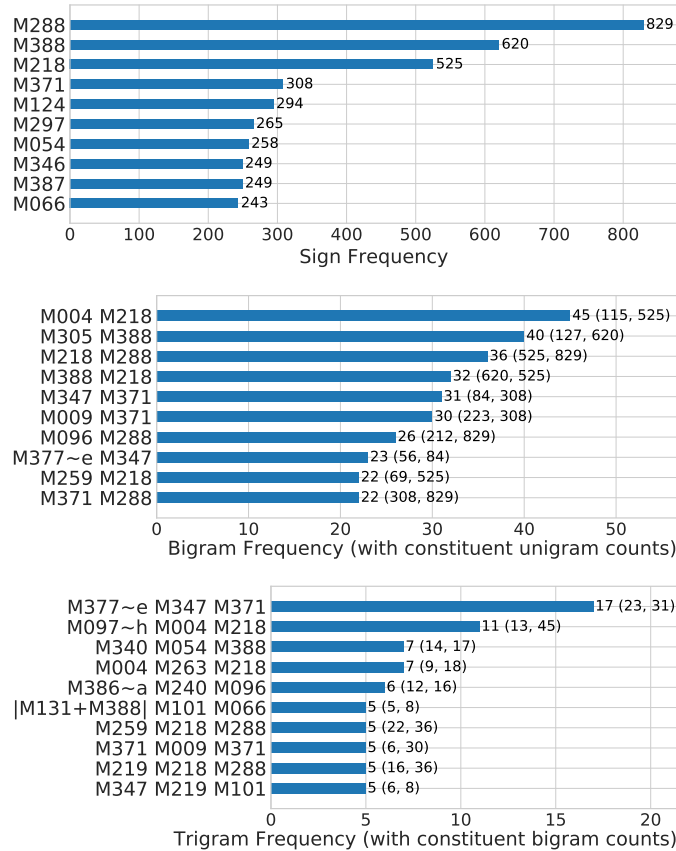




Figure 4.2: The 10 most frequent proto-Elamite unigrams, bigrams, and trigrams (top to bottom). In parentheses are given the frequencies of the two unigrams comprising each bigram, and the two bigrams comprising each trigram: note that some frequent  $n$  grams are comprised of relatively infrequent  $n - 1$ -grams.

sented using Unicode cuneiform signs, however this data comes from literary genres which are more prose-like than administrative accounts, and are thus also likely incomparable for this discussion.

Despite these difficulties, a qualitative assessment of Ur III Sumerian administrative texts suggests that they are highly repetitious, with information of wide importance to the administration (e.g. basic nouns, phrases describing administrative functions, month names, ruler names, etc.) occurring frequently. If one expects such information to be similarly frequent in the proto-Elamite administrative record, it apparently cannot be routinely encoded using trigrams nor possibly even bigrams; only unigrams appear to repeat with sufficient frequency to capture the expected distribution of very common administrative terms. This suggests that the script may employ relatively few multi-sign words, or that multi-sign words were only used to represent relatively uncommon entities.

One type of entity which may have been represented using multi-sign strings are the so-called anthroponyms, or strings speculated to represent personal names. Dahl (2019:

85) lists frequently-attested signs (10 instances or more) with “proposed syllabic values” obtained through traditional graphotactic analysis; Figure 4.3 presents the frequency of the most common bigrams and trigrams limited to this subset of signs. This list fails to include what is thought to be the most commonly attested personal name, M377~e M347 M371  mentioned above, since the middle sign is not included in Dahl’s published list of candidate syllables. Nonetheless the strings in this figure are representative of the set of suspected personal names (Desset 2012), since object signs which are understood to encode separate units of information have been weeded out. Overall we see that a small handful of 3-sign personal names are repeated at least 4 times across the corpus, but the majority appear 3 times or less. 2-sign names appear to be more frequent, although some of the bigrams in the figure simply represent substrings from the trigrams.<sup>3</sup> The ten most common namelike bigrams all appear 13 or more times across the corpus, and the most frequent alone appears 45 times (M004 M218 , including as part of a common trigram in Figure 4.3, accounting for 11 of its uses).

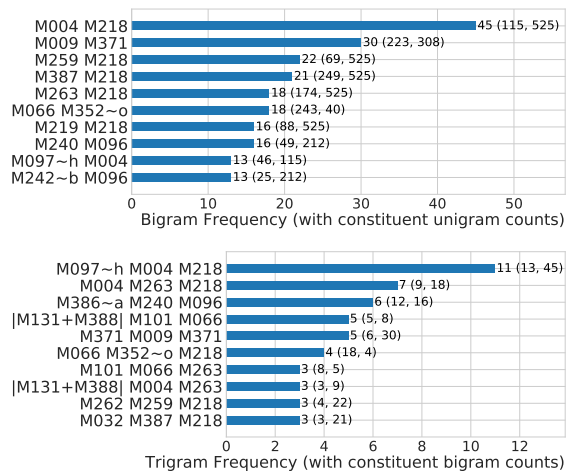














Figure 4.3: The 10 most frequent proto-Elamite bigrams and trigrams (top to bottom), limited to signs in Dahl’s (2019) candidate syllabary. In parentheses are given the frequencies of the two unigrams comprising each bigram, and the two bigrams comprising each trigram.







Repeated  $n$ -grams, anthroponymic or otherwise, become increasingly rare for  $n > 3$ . No 4-gram or 5-gram appears more than 3 times; no 6-gram appears more than twice; and no 7-gram appears more than once. This low level of repetition indicates that common frequency-based linguistic decipherment methods may be ineffective on this corpus, as the distribution of observed collocation frequencies is quite flat. We can overcome this limitation to some extent by performing a “fuzzy” matching to find strings which differ by only a

<sup>3</sup>Moreover, according to Desset’s (2016) traditional analysis of 515 candidate anthroponymic sequences, “250 (48.5 %) were made of 3 signs, 118 (22.9 %) of 4 signs, 83 (16.1 %) of 2 signs, 38 (7.3 %) of 5 signs, 15 (2.9 %) of 6 signs, 8 (1.5 %) of 7 signs and 3 (0.5 %) of 8 signs.”

few characters: for example, the only two 6-grams which occur multiple times in the corpus differ from one another by only a single sign:

M305	M388	<b>M240</b>	M097~h	M004	M218
					
M305	M388	<b>M146</b>	M097~h	M004	M218
					



A further variant appears once in the corpus:

M305	M388	<b>M347</b>	M097~h	M004	M218
					

Traditional graphotactic analysis parses the first of these strings as an institution, household, or person class (M305) and a person class (M388), followed by some further designation(s) of the individual (M240 M097~h M004 M218).

The identical contexts seen above suggest that the signs M240, M146 and M347 may play parallel roles in the proto-Elamite script. These signs may be yet another classifier preceding a stable personal name M097~h M004 M218, or they may reflect a naming pattern in which the first element can alternate. In Section 11.2 we will show that a technique for detecting affixal morphology, adapted from Alice Kober’s (1945) work on Linear B, identifies these as candidate prefixes in keeping with the second hypothesis.

### 4.3 Hierarchical Sign Clustering

Manual decipherment of PE has proceeded in part by identifying that certain signs occur in largely the same contexts as other signs. This has produced groupings of signs into “owners”, “objects”, and other functionally related sets (Dahl 2009). For example, M388  and M124  are known to be parallel “overseer” signs which appear in alternation with one another (Dahl et al. 2018: 25). In a similar vein, Knight et al. 2011 used a hierarchical clustering over characters to discover equivalencies between certain letter shapes, which ultimately led to their decipherment of the Copiale manuscript.



We have investigated three techniques for clustering signs hierarchically based on the way they occur and co-occur within texts in proto-Elamite. In our neighbor-based clustering, we prepare a co-occurrence vector for each sign by counting how many times every other sign occurs to its immediate left or right; we cluster the signs, using these co-occurrence vectors as input features, using the `scipy.cluster.hierarchy.linkage` method in Python. Our HMM clustering groups signs based on the emission probabilities of a 10 state hidden Markov model (HMM) trained on the transliterated corpus: we use the same `scipy` method






for clustering, with the rows of the trained emission matrix as input features. Finally we compute a generalized Brown clustering as described in Derczynski and Chester 2016.

By using three different clustering techniques, we can search for clusters which recur across all three methods to maximize the likelihood of finding those that are meaningful. This reduces the impact of noise in the data, which we expect to be necessary given the small size of the proto-Elamite corpus and the difficulty of distinguishing spurious groupings from those which may reflect as-yet-unobserved similarities between signs.






### 4.3.1 Clustering Evaluation

We identify commonalities between our three clusterings using the following heuristic. Given a set of signs  $S$ , we find for each clustering the height of the smallest subtree containing every sign in  $S$ . If all of these subtrees are short (which we take to mean not larger than  $2|S|$ ) then we call  $S$  a stable cluster.

In many cases, the stable clusters comprise variants of the same sign. This is the case for M157  and M157~a , which cluster together across all techniques and are already believed to function similarly to each other, if not identically.

One very large stable cluster comprises the signs M057 , M066 , M096 , M218 , and M371 . This cluster is shown as it appears in each clustering in Figure 4.4. These signs belong to Meriggi’s proposed syllabary (Meriggi 1971, esp. pp. 173–4) and are hypothesized to represent names syllabically (or as a mixture of logographs and syllables; Desset 2016: 83). Desset (2016: 83) likewise identified “approximately 200 different signs” from possible anthroponyms, “among which M4, M9, M66, M96, M218 and M371 must be noticed for their high frequency.” Desset’s list differs from our cluster by only two signs, replacing M057 with M004 and M009. M004 and M009 group with other members of the putative syllabary in each clustering, but their position is more variable across the three techniques. For M009 at least, this may indicate multivalent use: besides its inclusion in hypothesised names (e.g. Meriggi 1971: 173; Dahl 2019: 85), it appears in various different administrative contexts that don’t appear to include names (e.g P008206) and as an account postscript (see below here and Section 4.4.3).

All three methods group the five signs in our cluster close to other suspected syllabic signs; however, since each technique groups them with a *different* subset of the syllabary, only these five form a stable group across all three methods. This may be due simply to their frequency, or they could in fact form a distinct subgroup within the proposed syllabary. We will see in Section 5.2 that there is evidence for the latter.

While this discussion has focused on the stable clusters for which we can provide some interpretation, others represent groups of signs with no previously recognised relationship, such as M003~b  and M263~a  (Figure 4.5). M003(~b/c)    are “stick” signs understood in some PE contexts to denote worker categories (Dahl et al. 2018); they are



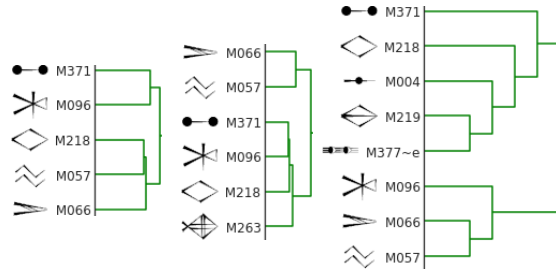


Figure 4.4: Detail of the (a) neighbor-based, (b) HMM, and (c) Brown clusterings showing signs possibly used in anthroponyms. M057, M066, M096, M218, and M371 are considered a stable cluster due to their proximity in all three clusterings.

graphically comparable to proto-cuneiform  $PAP\sim a-c$  ( $\llcorner$ ) and  $PA$  ( $\llcorner\llcorner$ ), the latter of which can, in later Sumerian, indicate *ugula*, a work group foreman/administrator.

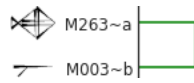


Figure 4.5: M003~b clusters identically with M263~a in all three techniques.

M263~a  $\llcorner\llcorner$  is one of a series of depictions of “vessels”, this particular variant appearing in 27 texts; notably the base sign M263  $\llcorner$  appears as a possible element in personal names (Dahl 2019: 85). Interestingly, M003~b and M263~a only appear together in a single text (P008727), one of a closely-related group of short texts that each end in the administrative postscript M009 M003~b  $\llcorner\llcorner$  or M009 M003~c  $\llcorner\llcorner$ . It can also be noted that M263~1  $\llcorner\llcorner$  occurs in another text belonging to this small group.

We leave for future work to interpret this and the many other stable clusters resulting from our work, as such analysis requires deeper Assyriological exploration than is warranted in the present context. These additional groupings are detailed in the data exploration toolkit released alongside Born et al. 2019. The complete dendrograms for each clustering are also available there, or in Appendix A.1 of this thesis.

Although we have not performed a full study of the clusterings produced when sign variants are collapsed together, preliminary comparisons suggests this may also be worth pursuing. For instance, a new cluster of small livestock signs arises in the neighbor-based clustering, comprising M367  $\llcorner$  (“billy-goat”), M346  $\llcorner$  (“sheep” or “ewe”), M006  $\llcorner$  (“ram”), and M309  $\llcorner$  (possible animal byproduct). Existing clusters, such as the stable cluster of syllabic signs, appear to remain intact, but a complete comparison of the techniques in this setting is warranted.

## 4.4 LDA Topic Model

Latent Dirichlet Allocation (LDA; Blei et al. 2003) is a topic modeling algorithm which attempts to group related words into topics and determine which topics are discussed in a given set of documents. Notably, LDA infers topical relationships solely based on rates of term co-occurrence, meaning it can run on undeciphered texts to yield information on which terms may be related. Note, however, that topics may be semantically broad, and one must be careful not to infer too much about a sign’s meaning simply from its appearance in a given topic. LDA differs from the other clustering techniques we have considered in that it also provides a means for grouping tablets based on the topics they discuss, which may reveal genres or other meaningful divisions of the corpus.

We induced a 10-topic LDA model over the proto-Elamite corpus using the `gensim` Python library. We chose a small number of topics to make the task of interpreting the model more manageable; fewer topics make for fewer sets of representative signs to analyze. Furthermore, with 10 topics the model learns topics which are mostly non-overlapping (Figure 4.6), meaning there are few redundant topics to sort through. We note, however, that model perplexity drops sharply above 80 topics, and topic coherence peaks around 110 topics; future work may therefore do well to investigate larger models.

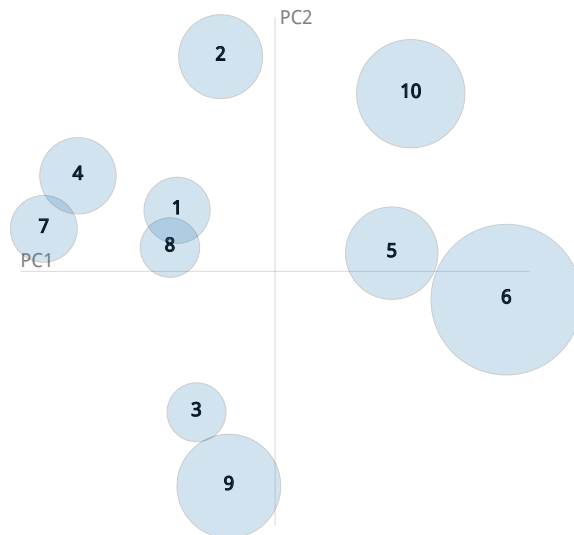


Figure 4.6: Intertopic distance (measured as Jensen-Shannon divergence) visualized with LDAVis (Sievert and Shirley 2014) using two principal components (PC1 and PC2). Larger circles represent more common topics.

In the following sections, we will attempt to interpret the topics from this model to establish whether and to what extent they relate to known patterns of sign use and speculative genre divisions in proto-Elamite, or perhaps even reveal new such patterns. However, as LDA is a probabilistic algorithm, repeated executions generally return distinct sets of topics even on the same data; thus our interpretations should ideally be informed by some




notion of topic stability, so that we do not place too much weight on a particular sign grouping that only occurs in this particular run of the algorithm.

Mäntylä et al. (2018) provide a technique for finding *stable* topics by looking for similarities across multiple executions of the LDA algorithm. These authors suggest learning  $k$  models with  $t$  topics each, to get a total of  $t \times k$  topic-term vectors. Running  $k$ -medoids on these topic-term vectors produces  $t$  topic clusters. The stability of a cluster is defined as the rank-biased overlap (RBO) between all vectors in that cluster:

$$RBO(T1, T2, p, d) = \frac{X_d}{d} \cdot p^d + \frac{1-p}{p} \sum_{i=1}^d \frac{X_i}{i} \cdot p^i$$


where T1 and T2 are ranked lists (a pair of topic-term vectors from the cluster in question),  $d$  is the evaluation depth (only the  $d$  most predictive words for each topic are considered), and  $X_d$  is the length of the intersection of the first  $d$  entries in T1 with the first  $d$  entries in T2.  $p$  is a “rank bias” which determines the effect of reordering between terms in different topics.  $p = 1$  ignores the relative order of terms: so long as the  $d$  most predictive terms are the same across T1 and T2, these will be treated as “the same topic”. As  $p$  decreases, more weight is assigned to terms which are more highly predictive. The authors suggest  $p = 0.9$  which we follow. The higher the mean pairwise RBO between vectors in a cluster, the more stable the topic represented by that cluster, i.e. the more likely a similar topic is to occur in any given run of LDA. We apply this method to our data to measure topic stability across 30 random restarts, and we make note of cases where a topic’s stability influences our interpretations in the coming sections.<sup>4</sup>

#### 4.4.1 Topic 1

The most representative signs for this topic are M376  and M056~f . M376 has been speculated to represent either a human worker category or cattle; M056~f is a depiction of a plow, comparable to the proto-cuneiform sign for plow, APIN . As a candidate for representing cattle, M376 is a sign of particular interest, since a sign-set for bovines has not been securely identified in proto-Elamite, despite the clear cultural importance of cattle suggested by proto-Elamite cylinder seal depictions (Dahl 2016).

We note that M376 can be viewed as a top-down depiction of a plow, whereas M056~f is a view from the side. In light of this suggestive shape, and the apparent association between these two signs implied by the topic model, it is tempting to propose a novel reading for M376 as another kind of plow sign, possibly denoting a cattle-drawn plow in keeping with the prior proposed meanings for this sign. In Chapter 8 we will discuss an interesting text which contains multiple instances of this sign, and which very clearly shows that it is



<sup>4</sup>The raw RBO values and term frequencies from this stability analysis are available in Appendix A.2.

associated with significantly larger amounts of grain than another proposed livestock sign which probably represents a goat (M367~i ). This would seem to be consistent with the reading of M376 as a plow animal which would presumably require larger feed amounts, though this raises the new question of why a single text would record feed amounts for both goats and cattle, where in other documents it is more typical to find goats counted together with sheep.

We also note that M376 and M056~f never occur in the same text, and in our stability analysis there is no stable cluster that contains both of these signs, though there are stable clusters which group both M376 and M056~f with other signs that are predictive of this topic. These two signs therefore appear to have roughly complementary distributions, occurring in similar contexts but never at the same time. If we accept the very tentative proposal that this sign may be another representation for a plow, this complementarity could be explained as different types of plow being used for different products or areas (and thus recorded in different texts), or as some scribes using an idiosyncratic sign for plow in place of the sign with a proto-cuneiform parallel.

At present, these proposals cannot be treated as anything more than speculation, but even as speculation this discussion demonstrates how patterns identified by models can help to spur hypothesis generation and prompt new interpretations of this ancient data.

#### 4.4.2 Topic 3

The visually-similar signs M297~b  and M297  are both highly representative of this topic. This is interesting as the relationship between these two signs has been uncertain (Meriggi 1971: 74). M297~b was hypothesised to indicate a “keg” by Friberg (1978). It is an “object” sign that almost always appears in the ultimate or penultimate position of sign strings; it sometimes appears in the summary line of accounts followed by numerical notations that quantify amounts of grain or liquids. Friberg suspected such texts referred to ale distributions; ale is thought to have been a staple of the proto-Elamite diet at Susa. Meriggi suggested M297 may indicate “bread”, but he also included it in his syllabary; it is the 6th most common sign in proto-Elamite, appearing in 145 texts, and M297~b is the 31st most common, appearing in 66 texts. Yet topic 3 is the dominant topic in only 85 texts, suggesting that this topic covers only a subset of the accounts that refer to M297 or M297~b. Also of note is the fact that M297~b occurs in topic 3 at a significantly higher rate than M297, despite being rarer in general—a much higher percentage of the overall uses of M297~b appear in this topic (around 75%) than do the overall uses of M297 (less than 15%).

In our stability analysis, we find that M297 and M297~b never occur together in any stable cluster, although both signs appear individually across several stable clusters. This may suggest that these signs exhibit some degree of polysemy, with multiple senses that are recognized as belonging to different topics. This view is consistent with the fact that M297

has been interpreted as both object and syllable in prior work. Though these signs evidently share some commonalities which caused them to group together in this topic (and which are evidenced by their similar appearances), the fact that their most stable groupings are with other signs points to them being fundamentally distinct from one another, or at least having senses which are clearly so.


#### 4.4.3 Topics 4 and 7

The texts included in topics 4 and 7 successfully reproduce aspects of Dahl 2005a with reference to the genres of proto-Elamite livestock husbandry and slaughter texts. Dahl was able to decipher the logographic or ideographic meaning (if not the phonetic realization) of signs for female, male, young, and mature sheep and goats and some of their products, beginning with the key observation that proto-cuneiform UDU  $\oplus$  (“mixed sheep and goats”) is graphically comparable to M346  $\oplus$ . The most representative signs in topic 4 are M346  $\oplus$  (“ewe”) and M367  $\wedge$  (“billy-goat”). These signs or their variants also appear together in stable clusters from our stability analysis.

While almost every instance of M346 is representative of topic 4, it is assigned to topic 5 in the atypical text P272825 (see 4.4.4). Several other typical livestock uses of M346 belong to topic 7. Topic 7 corresponds to the cluster with the highest RBO in our stability analysis, i.e. this is the *most* stable topic across 30 repeated runs. The most predictive features for this topic are the sign M009  $\parallel$ , which is also predictive of topic 4 (and appeared in Section 4.3.1), and the bigram M106 M009  $\triangle \parallel$ . At least one variant of M106 is taken to represent dry cheese or another animal product (M106~a  $\triangle$ ) (Dahl 2005a: 113), which points to this being a possible topic of animal byproducts as distinct from topic 4 which may refer more specifically to the animals themselves. Although the model identifies this topic in many texts which currently have no known association with livestock or animal products, it is perhaps noteworthy that several of these (e.g. P009141 and P008407) do bear seal impressions *depicting* livestock.

#### 4.4.4 Topic 5

The reason that the LDA model groups these 144 texts is not immediately apparent to the traditional PE specialist. An odd feature of the topic is that M388  $\cup$  (“person/man”) is considered the most representative sign, but the most representative *text* is a simple tally of equids that never uses M388, and in fact uses few non-digit signs overall. This may be due simply to noise in the model: M388 may be a kind of “stopword” which crops up in unrelated topics due to its high frequency (LDA models are typically trained on data that has been stripped of such stopwords, but this is impossible for proto-Elamite where we do not know which signs correspond to categories that ought to be removed). That said, an intriguing feature is that a significantly larger proportion of the texts in this topic bear

a seal impression than do texts in the other topics. The LDA model is not aware of the presence or absence of seals at training time, and their increased presence suggests that it is at least possible the model has identified similarities in tablet content that are not so easily observed through traditional analysis. The atypical “elite redistributive account” (Kelley 2018: 163) P272825, which is also sealed, is associated with this topic. This text has around 116 entries using complex sign-strings, fifteen of which include M388. Other signs which are highly predictive of this topic include M157 , the most ubiquitous “header” sign, and M305, an “owner” sign which may parallel M388 (Section 5.2) and which also appears in headers. These signs all occur, together with other signs that are predictive of topic 5, in one of the clusters identified by our stability analysis, indicating that however obscure this topic’s meaning may be, the signal that it captures is robust to changes in the model seed.

Given the relative prevalence of both header signs and seal impressions among the texts in this topic, the most concrete interpretation that we can give is that there may be some as-yet unobserved connection between these two phenomena. We will explore this connection when we undertake a focused study of headers in Chapter 7, where we will demonstrate that in fact there *does* appear to be some previously-unknown relationship between headers and seals, which may possibly derive from geographical factors.






#### 4.4.5 Topic 6

The ten most representative signs for topic 6 include the five of Meriggi’s possible syllabic signs that grouped most stably in our sign clustering evaluation (see 4.3.1). These signs also appear together in a stable cluster of topics in our LDA stability analysis. Nine of the ten are also included in Meriggi’s syllabary, excluding only M388, the second most-strongly predictive sign for this topic. M388 has been key to the identification of possible personal names, since it tends to appear just before long sign strings and, through a series of arguments drawing on cuneiform parallels, may function as a *Personenkeil* (a marker for human names; Damerow and Englund 1989; Kelley 2018: 222 ff.; for discussion of how this role relates to the role of other “owner” signs see also Section 5.2). The texts of topic 6 are of diverse size and structure, but do tend to include many candidate names according to traditional analyses.

#### 4.4.6 Topic 10

This topic also confirms existing understandings of a proto-Elamite administrative genre, namely that of “labor administration” (Damerow and Englund 1989; Nissen et al. 1993), and appears to instantiate the second-most stable topic cluster in our stability analysis. The most representative signs are the characteristic “worker category signs” described in the very long ration texts discussed by Dahl et al. (2018: 24–23), and indeed all of those texts appear in this topic, in addition to a variety of other identifiable labor texts of somewhat overlapping content.

#### 4.4.7 Remaining Topics (2, 8, and 9)

Initial assessments also suggest promising avenues of analysis for topics 2, 8, and 9, all of which appear to instantiate stable clusters in our stability analysis. Topic 2 is heavily skewed towards M288  (“grain container”), the most common proto-Elamite sign;<sup>5</sup> its third most representative sign (M391 , possibly meaning “field”) may suggest an agricultural management context for some texts in this topic. Topic 8 is strongly represented by M195+M057 . This is an undeciphered complex grapheme, frequently occurring as a text’s second sign after the “header” M157 (with which we argue it may form a single logical unit: see Chapter 7, and note that the most predictive feature for this topic is actually the *bigram* M157 M195+M057 rather than the sign M195+M057 on its own). In topic 9, the two most representative signs are M387  and M036  (possibly associated with rationing). Since the LDA model is not aware of the numeric notations between entries, it is interesting that the bisexagesimal numeric systems B# and B appear prominently in texts of this topic, whether or not M036 (associated with those systems) appears: see particularly P009048 (the text most strongly associated with this topic) and P008619.

#### 4.4.8 LDA Summary

The preceding sections confirm that there are abundant parallels between the associations revealed by an LDA model and existing understandings of the corpus according to traditional proto-Elamite specialists. Our interpretations of the topics, though brief, serve to highlight this fact, and in turn to demonstrate that models developed for the study of modern glottographic writing can in fact produce usable results on this corpus, despite concerns that proto-Elamite may not be a true linguistic writing system. With this established, we are able to proceed to more sophisticated analyses in the coming chapters, with the new-found confidence that this data is sufficiently language-like to permit ourselves the use of traditional NLP models and techniques. The coming chapters will also feature deeper discussions of some of the patterns identified above, and will demonstrate how lines of inquiry first suggested by this model have led to significant and novel discoveries in our later work.

### 4.5 Related Work

Meriggi (1971: 173–174) conducted manual graphotactic analysis of PE (and later linear Elamite) texts, for example by noting the positions in which certain signs could appear in sign-strings. Dahl (2002) was the first to use basic computer-assisted data sorting to present

<sup>5</sup>A remarkable 37.3% of the topic’s probability mass is allocated to this sign, compared to just 2.5% for the second most predictive sign (M157, the “household” header sign). No other topic is so skewed: only topic 4 comes close, with 20.3% of its mass assigned to M346 (“sheep”).

information on sign frequencies, and Englund (2004: 129–138) concluded his discussion of “the state of decipherment” by suggesting that the newly transliterated corpus would benefit from more intensive study of sign ordering phenomena (for which see our Chapter 5). Apart from the use of Rapidminer<sup>6</sup> to perform simple data sorting in Kelley 2018, no publications other than our own have described a concerted effort to apply computational approaches to this dataset.

Computational approaches to decipherment (see i.a. Knight and Yamada 1999; Knight et al. 2006; Berg-Kirkpatrick and Klein 2013), which resemble the setup typically followed by human archaeological decipherment experts (Robinson 2009), have been useful in several real world tasks. Snyder et al. (2010) propose an automatic decipherment technique that further improves existing methods by incorporating cognate identification and lexicon induction. When applied to Ugaritic, the model is able to correctly map 29 of 30 letters to their Hebrew counterparts. Reddy and Knight (2011) look for linguistic signals in the Voynich manuscript, and show that the letter sequences are generally more predictable than in natural languages, opposite to the trend which we identify for proto-Elamite where the character sequences are *not* highly repetitious. Hierarchical clustering has previously been used by Knight et al. (2011) to aid in the decipherment of the Copiale cipher, where it was able to identify meaningful groups such as word boundary markers as well as signs which correspond to the same plaintext symbol.

Homburg and Chiarcos (2016) report preliminary results on automatic word segmentation for Akkadian cuneiform using rule-based, dictionary based, and data-driven statistical techniques. Pagé-Perron et al. (2017) furnish an analysis of Sumerian text including morphology, parts-of-speech (POS) tagging, syntactic parsing, and machine translation using a parallel corpus. Although Sumerian and Akkadian are both geographically and chronologically close to proto-Elamite, these corpora are very large (e.g. ~1.5 million lines for Sumerian), and are presented in word level transliterations rather than sign-by-sign transcriptions. This makes most of these techniques inapplicable to proto-Elamite. Our study is more similar in spirit to Reddy and Knight (2011), as the Voynich manuscript and proto-Elamite are both undeciphered and resource-poor, making the task of analysis especially difficult.

## 4.6 Conclusions

We have shown that methods from computational linguistics and natural language processing can offer valuable insights into the proto-Elamite script, and can substantially improve the toolkit available to the proto-Elamite specialist. Hierarchical sign clusterings replicate previous work by rediscovering groups of signs with related function, and reveal similarities

<sup>6</sup><https://www.rapidminer.com/>



between yet-undeciphered signs to give direction to the ongoing decipherment. Analysis of  $n$ -gram frequencies highlights the level of repetition in sign strings across the corpus as a point of further research interest, and helps to quantify expert intuitions that the corpus is less repetitive than expected based on other ancient administrative corpora. LDA topic modelling has replicated previous work in identifying known text genres, but has also suggested new relationships between tablets which can be explored using more traditional analysis or more focused computational efforts.

The methods we have used are by no means exhaustive, but they serve to demonstrate the feasibility of computational analysis on this dataset, and also to legitimize computer-assisted decipherment in the eyes of domain experts by showing that this process can replicate past manual findings. Particularly in a field populated by such a small handful of researchers, the faster data processing and ease of visualization offered by computational methods may significantly aid progress towards understanding this ancient writing system. It is our hope that computational techniques will at last provide the necessary impetus for this script to be fully understood after more than a century of only partial decipherments.

## Part III

# Main Results

## Chapter 5

# Sign & Word Order

As noted in Chapter 2, proto-Elamite appears to have a number of signs in common with another partially deciphered script called proto-cuneiform. Like proto-Elamite, the proto-cuneiform corpus is primarily administrative, though unlike proto-Elamite a small number of possible lexical texts have been identified (we assisted with data collection for an analysis of one such tablet in Born and Kelley 2021). However, the structure of accounts differs greatly between the two scripts. Where proto-Elamite text is organized into clear lines, proto-cuneiform rather organizes signs into boxes (called “cases”) with no clear ordering to the signs within each case (an arbitrary ordering is imposed when these texts are transliterated). Thus it is difficult to conceive of these texts exhibiting anything we would recognize as word ordering (though there may have been a conventional reading order which was more readily apparent to the original scribes). If Elamite society adopted the technology of writing from proto-cuneiform, it is possible that this script also lacks a consistent notion of word order.<sup>1</sup>

Establishing the presence or absence of word ordering is important for determining whether this script could possibly represent a true writing system. It is difficult to imagine a system like proto-cuneiform encoding language to any meaningful extent: even languages with relatively free word order exhibit *some* level of syntactic structure, and linguistic processes such as scrambling (Becker et al. 1991), which appear to freely reorder the parts of a sentence, only apply to certain phrasal categories, and not to every individual word. It is notable that later phases of cuneiform writing, which *are* demonstrably linguistic, do away with the free ordering of signs and impose a linear order much like modern writing systems. In sum, the less evidence we find for consistent character ordering principles in proto-Elamite, the greater credence may be lent to claims that this system is non-linguistic.

<sup>1</sup>Throughout this chapter, we make reference to “word order” as an umbrella term for consistent ordering relations between signs. This is purely to make the argumentation more concise: though the relations we identify *may* correspond to true syntactic word ordering rules if the script is linguistic, it is equally possible that individual signs do not correspond to complete words, or that the observed ordering encodes a different kind of relationship altogether.

Even if the system is non-linguistic, an awareness of underlying ordering principles may be useful for determining the possible function of signs with unknown meanings. Experts already assume that the final sign of an entry is generally the object being counted by the adjacent numeral. A preceding sign is possibly a qualifier modifying the object, but in the case where there are several preceding signs it is unclear whether these represent a single multi-sign qualifier or several single-sign qualifiers; it is similarly unclear whether the signs comprising a multi-sign qualifier would be ordered within that “word”, or whether distinct qualifiers would be ordered relative to one another. If some signs consistently follow a particular order relative to one another, it may be possible to group them according to their preferred position and look for commonalities within each group. Some early analyses relied on an assumption that such ordering principles were present between signs and were more-or-less strict (Englund 2004: fn. 9 discussing Meriggi 1971), despite earlier observations that such trends were in fact not strict (Brice 1963: 28). At the same time, consistent principles have been demonstrated to govern the ordering of *entries* across entire tablets (Dahl et al. 2018; Hawkins 2015), and it is conceivable that similar principles also apply at some level *within* entries. Lastly, Brice (1963: 25) remarks on the difficulty of labeling tablet headers, “as there is no indication on the tablets of where the heading ends and the first item begins.” Given that the header always precedes the first entry, any search for ordering principles should reveal clear precedence relations between header signs and other sign types, and thus provides a way to more concretely define this class of signs.

## 5.1 *n*-Gram Entropy

If the signs within an entry do not follow any consistent ordering principles, then we should expect a smoother *n*-gram distribution than if signs are consistently ordered. In other words, word ordering principles should be expected to reduce the entropy of a corpus’ *n*-gram distribution by removing some uncertainty about which sign comes next in any given context. This implies that it may be possible to detect word order by randomly shuffling signs within entries. If this shuffling increases the entropy of the *n*-gram distribution by a significant amount, it suggests that there *was* some ordering which has now been destroyed. However, if the *n*-gram distribution exhibits similar entropy before and after shuffling, this suggests that the ordering of signs was already random enough for the shuffle to have no effect.

We measure the Shannon entropy of the character *n*-gram distribution  $P(w_i, \dots, w_{i+n})$  in the proto-Elamite and proto-cuneiform corpora before and after shuffling the non-numeric parts of each entry or case.<sup>2</sup> We do not include *n*-grams from the numeric portions of texts in either script. The proto-cuneiform corpus can be divided into two groups, reflecting

<sup>2</sup>We observe similar trends when applying the same technique to the conditional entropy distribution  $P(w_{i+n}|w_i, \dots, w_{i+n-1})$ .

administrative tablets (like those from proto-Elamite) and possible lexical tablets speculated to represent simple word lists. We evaluate these subsets separately from one another.


For comparison, we measure the same quantities before and after shuffling each line in the Late Babylonian, Old Babylonian, Neo-Assyrian, Standard Babylonian, and Sumerian data from the 3<sup>rd</sup> VarDial language identification shared task (Zampieri et al. 2019). These dialects are all written in a linear order using cuneiform-derived scripts, and together they span multiple centuries and language families to provide a broad baseline for comparison. We choose the VarDial data specifically because it has been detokenized and encoded using Unicode cuneiform characters. Other datasets for these languages typically convert the original character strings into (possibly inflected) word-forms, in ways that obscure the original characters (for example, by replacing the 4-character string  AB<sub>2</sub>.NUN.ME.DU with the atomic word token `abrig2`). This is a more abstract level of representation than we find in the undeciphered proto-Elamite corpus, where we only have access to raw sign names. By reverting to Unicode cuneiform character sequences, the VarDial data is unique in representing these later languages at a level of abstraction that is actually comparable to the available proto-Elamite data.

Figure 5.1 shows the average change in entropy after shuffling each script, averaged over 10 trials and broken down by  $n$ -gram length. As expected, all of the later cuneiform writing systems exhibit increased entropy as a result of shuffling, reflecting the known presence of syntactic word order relations in the original corpora. The change becomes less pronounced as the value of  $n$  increases, since the original  $n$ -gram distributions become increasingly uniform for large  $n$ , lessening the effects of shuffling.

The proto-cuneiform administrative corpus exhibits the smallest average changes in entropy, reflecting the fact that there is no clear ordering to the signs in the original texts. The proto-cuneiform lexical corpus exhibits somewhat larger changes, which is consistent with the view that these texts may represent an incipient use of the script to represent language. Moreover, during transliteration, lexical texts are intentionally linearized in such a way as to produce known Sumerian words when possible, which likely further contributes to the appearance of some consistent character ordering.

Proto-Elamite occupies a middle ground, exhibiting significantly greater increases than the proto-cuneiform administrative corpus for  $n \leq 3$ , and possibly falling in-between the proto-cuneiform lexical and administrative corpora for larger  $n$ . Proto-Elamite also appears to exhibit larger increases in entropy than the proto-cuneiform lexical texts for small  $n$ , though there is enough variance in both traditions that we cannot claim significance. Proto-Elamite exhibits smaller increases than later cuneiform traditions.

These results are consistent with the view that the proto-Elamite accounts have somewhat more in common with real writing than the proto-cuneiform accounts, perhaps roughly on par with the proto-cuneiform lexical tradition. This may point towards proto-Elamite having only incipient ordering conventions, which are followed inconsistently or only by cer-

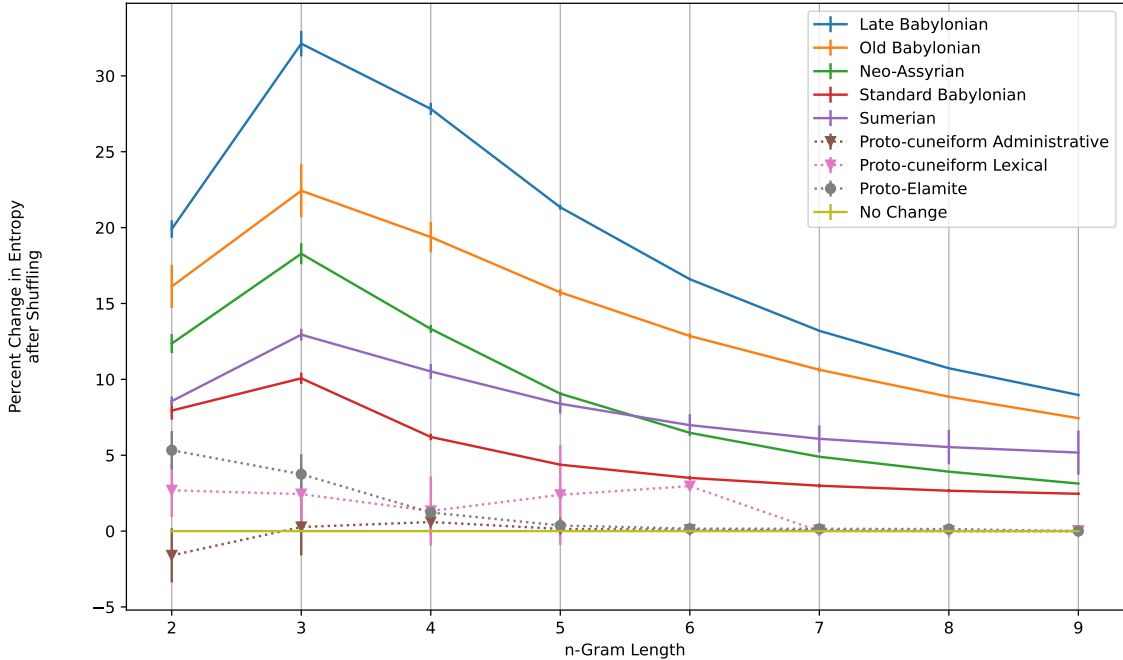


Figure 5.1: Percent change in entropy of the character  $n$ -gram distributions for cuneiform and cuneiform-adjacent scripts after shuffling, averaged over 10 shuffles. Larger values suggest stricter ordering principles in the original character distribution, and may reflect a more consistent ordering of characters into words and of words into phrases.

tain scribes; or proto-Elamite may not exhibit any syntactic word order, but some scribes may use conventional orderings which may differ from scribe to scribe; or perhaps order is only meaningful for some of the signs in an entry, while others can be freely reordered (if, for example, the counted object must be the final sign, but qualifiers may be listed before it in any order). As we lack clear word boundaries in the proto-Elamite data, it is also unclear whether this ordering should be taken obtain between words, or between characters comprising a word, or between concepts which do not map cleanly onto linguistic categories (i.e. ideograms).

Overall, while raw entropy scores are too blunt an instrument to give definite answers to questions of “word” order in these texts, these results are nonetheless useful for positioning the proto-Elamite script more concretely relative to other systems whose relations to language are more clearly understood.

## 5.2 $\chi^2$ Tests for Word Order

If the signs within an entry are unordered, then conditioned on the fact that two signs  $A$  and  $B$  occur in the same entry, we should be equally likely to observe  $A$  before  $B$  as to observe  $B$  before  $A$ . As a consequence, a simple way to check for consistent ordering rules

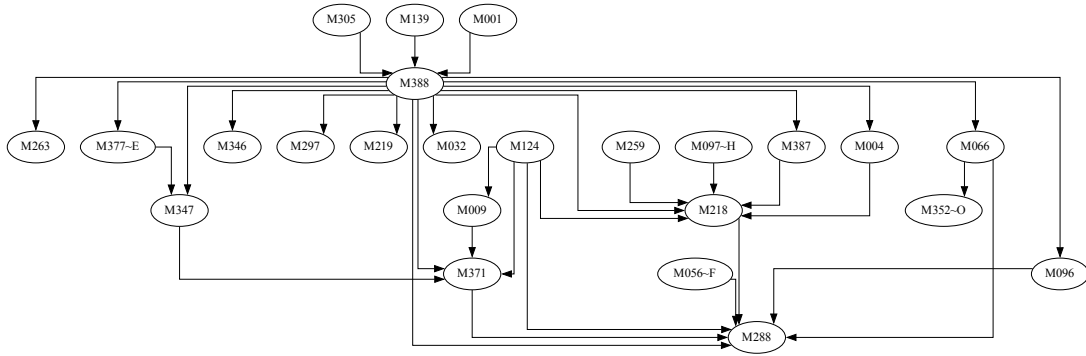









Figure 5.2: Excerpt from Figure A.4 illustrating a partial order over signs occurring together 20 times or more.




is to use a  $\chi^2$  test with two outcomes ( $A > B$  or  $B > A$ ), each of which is expected in 50% of cases. If one case turns out to be significantly more likely than the other, there is at least a possibility that some relationship obtains between the pair which causes them to be written in the observed order more often.

We identify all pairs of characters where a  $\chi^2$  test suggests that one ordering is more likely than the other with  $p < 0.05$ . We only test pairs of signs which occur in the same entry at least 10 times, following conventional recommendations that  $\chi^2$  testing only be applied when each of the expected values is at least 5. From this we derive a partial order, which we represent as a graph with a directed edge from  $A$  to  $B$  whenever  $A$  is significantly more likely to precede  $B$  than *vice versa*. The full graph is available in Appendix A. Figure 5.2 shows a more clearly readable excerpt featuring only those sign pairs which occur 20 times or more.

**Counted Objects as Infima** It is immediately clear that this graph captures many existing intuitions about the internal structure of proto-Elamite accounts. Of the seven apparent infima in Figure 5.2, three are true infima in the full graph (M297 , M288 , and M346 ) and another is a variant of a true infimum (M263 ). These have all been recognized as “object” signs, and are believed to represent containers (M263, M297, M288) or sheep (M346). Object signs are understood to occur at the very end of an entry, and to specify the object that is being counted by the numeral which immediately follows. The position of these signs as infima demonstrates that our proposed approach has accurately captured this aspect of their usage and the corresponding positional preference.

Considering the complete graph, a majority of the other infima are also already labeled as likely object signs in the working sign list. Of those which are not (M262 , M314 , and M317 ), two (M262 and M314) only appear to be infima because all of the

signs which could follow them were pruned for being too rare. With a larger dataset, these signs would almost certainly be positioned nearer to the middle of the partial order. The one remaining infimum (M317) is not labeled as a counted object by Dahl (Unpublished), though it is acknowledged as one by Englund (2004). This sign is reasonably common, occurring 60 times with variants appearing an additional 13 times. Many of these appearances are in entry-final positions that appear suggestive of a counted object, though the sign also appears adjacent to some so-called “syllabic” signs (P008310), near the beginning of texts alongside common header signs (P008291), and as both the inner and outer component of some complex graphemes (P008365, P008499). This variety of uses may explain why M317 is not explicitly labeled as an object sign in the working sign list: particularly its presence next to “syllabic” signs warrants considering whether it may itself be syllabic. This sign has been identified as a likely worker category (Dahl et al. 2018; Kelley 2018; Dahl 2019), though it differs from some of the other putative signs for humans in that it has no obvious proto-cuneiform parallel (Dahl et al. 2018). Dahl et al. 2018 also identify a hierarchical ordering of entries within a tablet (distinct from the ordering of signs within entries, considered in this section) whereby entries containing M317 regularly appear before entries recording other categories of human; they suggest that this may reflect a practice of placing the “most important” categories earlier in a text (Dahl 2005a), which would point towards M317 being a particularly high-status human (consistent with similar claims in Englund 2011: 46-47). Among these varied uses, it is nonetheless possible to find texts which clearly position M317 as a counted object. Perhaps the clearest example comes from the text P008759, where M317 and a variant appear four times. The final occurrence is on the reverse of the tablet, in what appears to be a summary line<sup>3,4</sup> recording the sum total of the values on the text’s obverse. The use of a sign in a tablet summary is seen as strong evidence that the sign in question represents a counted object, and on this basis we argue that the grouping of M317 with the other object signs among the ordering’s infima is yet another case where the inferred order captures and formalizes a known pattern of sign use.

**Headers as Suprema** The set of suprema in the full graph does not map onto any category as cleanly as do the infima, though several (M305 , |M195+M057| , and possibly M377  and complex graphemes containing it) are understood to be “header signs” which provide some global context with which to interpret the following text. Dahl 2005c and Kelley 2018 suggest that headers may represent the household or institution to whom



<sup>3</sup>The text is damaged so that some of the numerals are unreadable, but the visible numbers add up to 112 in the sexagesimal system, and the summary reads 119 in the same system. Most of the visible entries record values of just 1 or 2, and there appear to be approximately 5 obscured entries, so the reading of the final entry as a summary appears plausible.






<sup>4</sup>This text is another apparently unremarked example of an “implicit object” in proto-Elamite, a category which we discuss further in Chapter 8.



the ensuing account belongs, or may specify the type of account which follows. The position of these signs as suprema is natural and expected given what is known of their prevalence near the beginning of texts.



**Owner Signs and Determinatives** Having established with the preceding examples that our inferred ordering accurately captures existing intuitions about this script, we turn next to a case where a deep look at the placement of certain signs in the graph may prompt a re-evaluation of certain signs.

Five different suprema have strong precedence relations with the sign M388 , which is also notable for being the sign with the highest out-degree (i.e. M388 has a strong precedence relationship with more signs than any other). M388 is one of the better-understood signs in the proto-Elamite corpus (see throughout Dahl 2019; Kelley 2018), and is believed to parallel proto-cuneiform KUR<sub>a</sub> in representing a laborer (possibly a male in particular, with the sign being an iconic depiction of the male genitalia, making this one of very few depictions of humans or human body parts in the script). This sign can occur as a counted object, in which case it apparently stands for a number of individuals being tallied. It also occurs near the beginning of many long entries, where it is understood to introduce the owner of the counted item which follows. In this usage it has been compared (Kelley 2018) to a *Personenkeil*, a type of so-called “determinative” known from later cuneiform scripts that indicates that the following signs represent a name. In later cuneiform scripts we find multiple such *Personenkeile*, for specifying masculine, feminine, and divine names, and likewise proto-Elamite exhibits other signs whose usage appears to parallel that of M388, most notably M124 . Signs which typically follow M388 are called “syllabic” signs on the hypothesis that they may represent syllabically-written names, though phonetic values have not been established for any of these signs at present.















The placement of M388 near the top of the inferred sign order reflects this usage at the beginning of possible names, and its high out-degree reflects the variety of so-called “syllabic” signs which consistently follow it. However, M388 is not a supremum, as it is very frequently preceded by one of M305 , M139 , M111~a , M001 , or |M195+M057| , typically with no intervening signs. Prior work has discussed some of these as possible owner signs (Dahl 2019; Kelley 2018), which qualify either the following individual or the counted object as belonging to a particular household or institution. As a concrete example, Kelley 2018: 233 breaks down the string M305~j M388 M032 M066~a from P272825 as

household (M305~j) → Personenkeil (M388) → personal name (M032 M066~a)

This presentation implies (perhaps unintentionally) a structure to the underlying text which we can represent using bracketed notation as [M305~j [M388 [M032 M066~a]]]. Here M305 is taken to be a “head” which qualifies the entire following namelike string, which is itself analyzed as a *Personenkeil* modifying a true name.

An alternative analysis could instead read this string as  $[[M305\sim j \ M388] \ [M032 \ M066\sim a]]$ , where M305 no longer “scopes over” M388, but is rather part of a multi-sign expression modifying the name. We argue that this is the more natural reading when multiple owner signs occur in sequence. P008012 offers support for this view, as this text attests the same trigram M004 M218+M101 M371  after M388 in one entry, and after M139 in another. The trigram in question is comprised of signs that are understood to belong to the putative syllabary, and is therefore a candidate for a personal name. M388 and M139 clearly occupy parallel positions in these two entries, and it stands to reason that they belong to the same functional category by extension. (In essence, we are arguing that these signs are substitutable for one another, and therefore belong to the same underlying category, in the same way that syntactic substitutability gives evidence that two phrases are the same kind of constituent in a natural language.) Another example where these signs appear to substitute for one another can be found in P008310, where they alternate before the trigram M032 M387 M218 .

We can find similar examples where other suprema from our inferred order appear before candidate names where an M388 may otherwise be expected:

- in P008004 we find M305 before the string of possible syllables M066~a M219 M101 M101 M066~a     ;
- in P008979 we find M001 before the syllable sequence M263 M218 M263   ;
- in P008724 we find |M195+M057| separated by one sign from the syllable string M387 M263~1 M110 M066~a    ; and finally
- in P009165 we find M111~a before a broken string beginning with the reduplicated syllable M066 M066  .






Thus all of these signs appear to be associated with the same category of name-like strings as M388, and can stand in the same position as M388 at the beginning of such strings. The simplest explanation is therefore that these signs all occupy the same structural position.

Moreover, although our partial ordering suggests that these signs typically precede M388 when both are present, there are exceptions where M388 comes first, such as M388 M111~a in P009018 or M388 M001 in P008709. If M388 functioned like a true determinative, it should occupy a lower structural position than owner signs which are not true determinatives, and these alternative orderings should never be attested. Thus these exceptions to the usual ordering give further evidence that M388 and the other signs in this section are really interoperable members of the same underlying category, which are ordered by convention rather than any strict structural constraint.


As additional evidence, we appeal to the state sequences learned by a Hidden Markov Model trained on the proto-Elamite corpus. In Chapter 4, we trained a Hidden Markov

Model as part of our initial explorations in sign clustering. That analysis did not make explicit use of the model’s Viterbi state sequences, but for interest’s sake we nevertheless computed these sequences in case they could reveal patterns of interest. Indeed, when considering these sequences, we observe that the model very consistently enters the same state when producing both M388 and M124 (the other candidate *Personenkeil* known from prior work). When producing bigrams involving M388 and the other signs discussed in this section, a majority of cases see the model staying in this same state when emitting *both* signs. On average the model also prefers this state when emitting these signs in the absence of M388, as in the bulleted examples above.<sup>5</sup> This suggests that any differences in function between these signs are minor enough that they can be effectively modeled as belonging to the same category.

In sum, we argue that M388 should not be treated as a true determinative, unlike the *Personenkeile* to which it is sometimes compared. Rather it is just another member of the set of “owner” signs, which alternates with other members of this class across parallel contexts and which can be (but by apparent convention, rarely is) freely reordered relative to other members of this category when they are present. The proposed readings of “man” or “person” are relatively generic, and this genericity is likely reflected in its overwhelming frequency (it is the second most common sign in the corpus). This genericity may also account for its typical placement *after* other owner signs, as seen in the partial order inferred above, if for example scribes chose to order these qualifiers from most to least specific.

**Syllabic Signs** Among the signs which have not been discussed so far, three stand out for having a very high in-degree in Figure 5.2, namely M218 , M096 , and M371 . All three belong to the putative syllabary (Dahl 2019), and their high in-degrees come in large part from other syllabic signs. These signs have very low out-degree, being followed only by the object signs M288  or M297  on a consistent basis.

This points towards the presence of some internal structure in the “names” which follow M388 and the other owner signs discussed above, whereby these three signs commonly end up at or near the *end* of a name. If the “syllabic” signs truly do encode phonetic values, then these three may represent common suffixes, or variant forms of the same suffix (for example, with different vowels or onsets depending on the preceding context). Alternatively,

<sup>5</sup>With the notable exception of [M195+M057], which already appears exceptional as it does not immediately adjoin syllable sequences but is typically separated from them by another sign, often M147 . [M195+M057] was already discussed above as a header sign, and we will argue in Chapter 7 that it is likely part of a multi-sign header in most texts where it occurs. However, since multi-sign headers are rarely annotated as such in the current version of the corpus, many attestations of this sign end up looking like part of the following entry, causing it to be erroneously mixed in with these other owner signs with which it does not actually form a natural class.

if the syllabary is more heraldic in nature, these may be the most or least specific identifiers, which occupy a place of (dis)prominence near the end of the syllabic string.

We emphasize, however, that the partial order inferred in this section only represents the *most common* ordering of signs relative to one another: these three signs do not *always* occur at the end of strings, but they do so most often. For a more robust view of how these signs are distributed, Figure 5.3 shows a heatmap of sign positions averaged across all entries that are 6 signs in length.<sup>6</sup> Each row represents one sign, and the brightness of the  $n$ th cell (reading left to right) indicates how frequent that sign is in the  $n$ th position. Brighter cells in the leftmost columns imply that a sign is more common at the beginning of the entry, and brighter cells to the right indicate higher frequency in later positions. A uniform color across the whole row implies that the sign is equally common in all positions. For context, the figure includes additional signs of interest and signs discussed in previous sections. Rows are ordered according to their median position, so that signs at the top of the plot are more common at the beginning of entries, and those at the bottom are more common at the end.

We see that M218, M096, and M371 exhibit very similar distributions in this figure, with all three showing a spike in occurrence frequency in penultimate position (which we may tentatively call “name-final” position, at the end of a syllable string and immediately preceding a counted object in the ultimate position). Recall that these signs also formed a stable cluster in our initial exploratory analysis (Chapter 4): this “affixal” behaviour is presumably the feature which leads them to cluster together. Other signs that exhibit a preference for the penultimate position include M066  $\nabla$  and M004  $\downarrow$ , also belonging to the putative syllabary. These signs are not uncommon in earlier positions, however, and their preference for name-final positions should be taken as a trend and not a rule.

Although there exist signs, like M219  $\diamond$ , which are *most* common as the antepenult, these signs tend to be rarer than signs occurring as penult, and their frequency distributions appear to be smoother. If this smoothness is a legitimate signal, and not a consequence of sampling bias arising from these signs’ rarity, it may suggest that signs in the “root” or “stem” of a syllabic sequence are not subject to strict ordering requirements, and that only “suffixal” signs like M218 consistently select for a particular location. This would be consistent with the observation that most nodes in the precedence graph in Figure A.4 have relatively low degree: the median in- and out-degrees are both just 1, and the median total degree is 2, meaning that most signs are only consistently ordered relative to two

<sup>6</sup>We limit the figure to entries of a fixed length to ensure we are comparing like with like; similar trends hold across entries of all lengths. We include a complete breakdown by entry length and sign name for every sign in Appendix A.4. We highlight 6-sign entries in particular because these are long enough for different patterns of sign positioning to be clearly distinguishable. For example, in 3-sign entries, there will be no apparent difference between signs which select for the middle of an entry, those which select for the second position, or those which select for penultimate position, but in longer entries all of these behaviours would be clearly distinct.

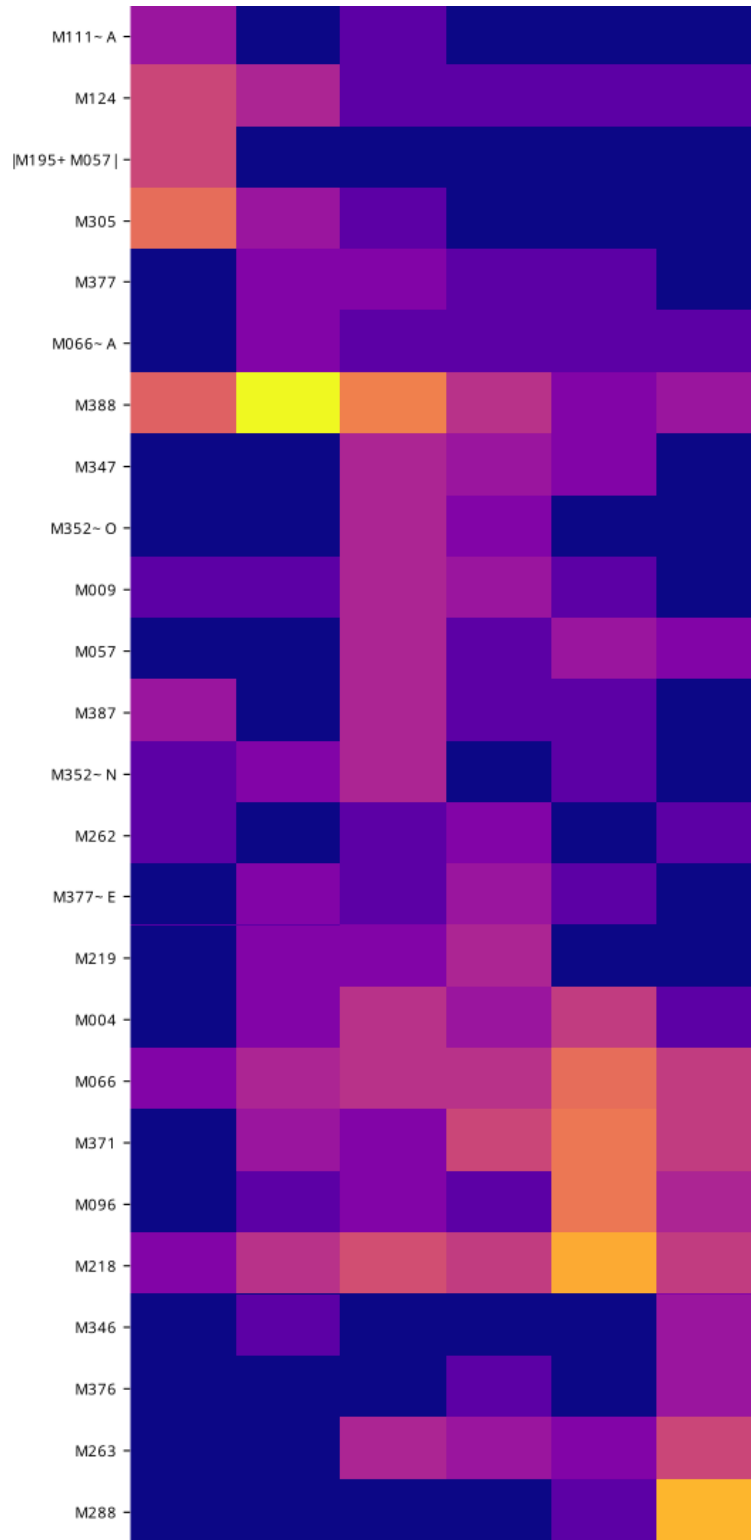


Figure 5.3: Heatmap showing average positions of select signs in entries of length 6. For each sign, the brightness of the  $n$ th-leftmost cell indicates the frequency with which that sign occurs in the  $n$ th position across all 6-sign entries.





other signs. Thus, while we can argue that many signs are likely to follow M388 or precede M288, we can only rarely order those signs *relative to one another*. On the basis of these results, we advance the suggestion that the anthroponyms or putative syllable sequences in proto-Elamite may be broken down into at least two parts, a “stem” for which we observe relatively weak constraints on the relative ordering of signs, and a following “suffix” which prototypically consists of a sign from the set M218, M096, M371, M066, or M004 which empirically prefer penultimate position. We will return to this claim in Section 11.2 where we consider a technique for identifying affix-like behaviour. Although we couch this argument in the familiar terms of stem-plus-affix morphology, we remain agnostic as to whether these strings are truly inflected word forms as opposed to any other kind of structured object.

## Chapter 6

# Complex Graphemes

This chapter sets out to understand a category of signs called complex graphemes, and reproduces results which we have originally published in Born et al. 2021. We propose an architecture for image-aware language modeling, which permits sharing of information between visually similar signs in much the same way that sub-word units share information between words in a traditional language model. We contrast the sign embeddings learned by this and other models to argue that certain complex graphemes are not merely orthographically compositional, but also exhibit signs of semantic compositionality, a fact which has noteworthy implications to the possible readings of the signs in question. We also demonstrate for the first time the existence of a grammar governing the construction of these signs.

### 6.1 Methodology

Recall from our introductory survey of the script that complex graphemes (Dahl 2005c) consist of two or more signs inscribed within one another, ligatured, or otherwise juxtaposed in such a way as to look like a single unit rather than a sequence of disjoint tokens. Generally, in a complex of type “A+B” (e.g. M131+M388 ) , the B sign is written within the A sign, and in a complex of type “A+B+A” (e.g. M153+M320+M153 ) it is written between (possibly mirrored) copies of A. For this reason, we refer to the first component of a two- or three-sign complex as the *outer* part, and the second as the *inner* part. However, there exist more marginal varieties of complex grapheme for which we acknowledge that this terminology is not quite appropriate (e.g. M296+M296  or M059+M038~a ). Most of the signs which occur as part of a complex grapheme can also occur as standalone signs. Exceptions to this are rare, such as M600 which only ever occurs in the hapax M362+M600.

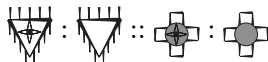
Although these signs are *orthographically* compositional, it is not known whether they are also semantically compositional. Similar constructions exist in proto-cuneiform, including containers with signs inscribed to indicate specific products (Wagensonner 2015). Some proto-cuneiform compounds survive into later cuneiform, where they sometimes obtain id-

idiomatic meanings, e.g. cuneiform GU<sub>7</sub> “eat”, a combination of “head” and “bowl”. Chinese characters likewise exhibit varying degrees of both visual and semantic compositionality (Sproat 2006).

If it can be shown that any of the proto-Elamite signs are semantically compositional, this will open new avenues for decipherment by showing that these signs can be understood from the meanings of their components. If the meaning of the complex can be determined, it would become possible to work backwards to establish the meanings of the parts, and *vice versa*. Moreover, if these complexes have compositional meanings, then identifying the function of a sign in one complex immediately reveals its function in any other complexes where that sign occurs. This kind of analogical reasoning cannot be applied if the complexes are idiomatic, as in that case the meaning of the inner or outer part may not be consistent across different idioms.

Past work (Mikolov et al. 2013b; Salehi et al. 2015; Cordeiro et al. 2016) suggests that word and phrase embedding models learn embeddings which capture semantic compositionality in noun compounds and multiword expressions in modern languages. Concretely, these models assign to such compounds a representation which is similar to the sum of the representations for the words within the compound, or to some other function of the components depending on the precise embedding technique in question. We hypothesize that, if complex graphemes are semantically compositional in proto-Elamite, it should be similarly possible to identify additive or other arithmetic relations between their component parts at a higher rate than expected by chance. The embeddings learned for these signs may also exhibit other signs of internal structure, such as an ability to model proportional analogy between graphemes with shared components:

$$|M136+M365| : M136 \quad :: \quad |M327+M365| : M327$$



If this analogy holds in the embedding space (which is to say that the “3CosAdd” formula (Mikolov et al. 2013b)  $M136+M365 - M136 + M327 \approx M327+M365$  holds between the signs’ embeddings) this would give further evidence that the graphemes involved exhibit some degree of semantic compositionality.

Unfortunately, most proto-Elamite signs are rare, which impedes models’ ability to learn meaningful information about their distributions. Yet many signs with distinct names have striking visual resemblances, and it is usually not known whether these have different meanings. Visual information may therefore improve representation learning by allowing a model to share distributional information across graphically similar signs, and therefore to learn more robust representations for signs which are *individually* rare but belong to a more common graphical archetype. To this end, we propose an initial approach to multimodal language modeling for proto-Elamite in Figure 6.1. This architecture uses two separate embedding components. On the left of Figure 6.1, in red, is a standard embedding layer



which performs a lookup from one-hot inputs to small, learnable, dense representations. On the right, in blue, a separate lookup function retrieves an *image* of the corresponding sign. A CNN extracts a feature vector from this image, which is max-pooled, flattened, and passed through a dense layer to produce a low-dimensional embedding. Both embeddings are concatenated and passed to a BiLSTM (Hochreiter and Schmidhuber 1997; Schuster and Paliwal 1997) which attempts to predict the label of the next (resp. previous) sign in the text at each time step. All timesteps share the same weights for the CNN and embedding layers. By omitting the blue image-embedding component we can obtain a normal (text-only) BiLSTM language model. By omitting the red text-based component, we can obtain an image-only model which *never directly sees the labels assigned to the signs* in its input.

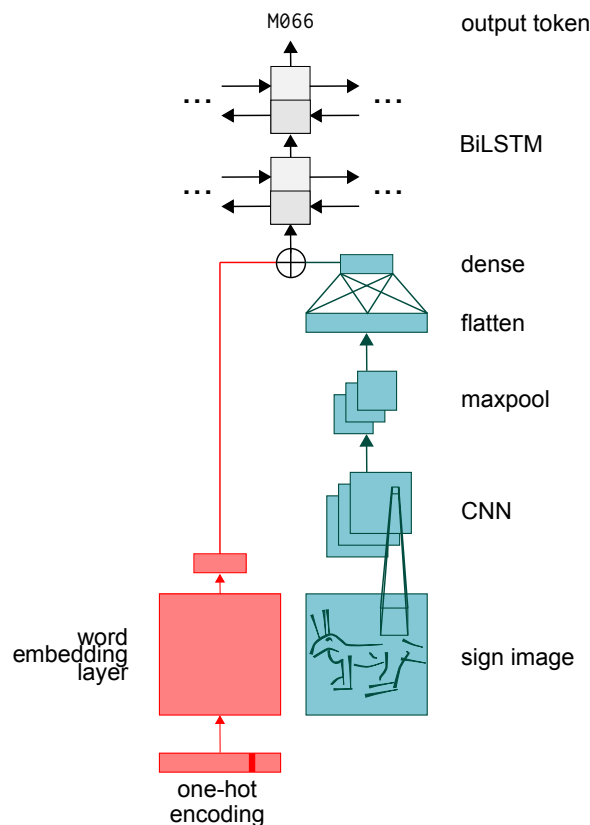


Figure 6.1: Architecture for image-aware, multimodal language modeling.



This model will learn representations which incorporate aspects of visual *and* contextual information. To more clearly establish how each of these modalities impact the geometry of the learned embedding space, we train a separate image *recognition* model to predict a sign’s label given only its image. This model uses the blue image embedding component from Figure 6.1 to produce a dense representation of an input image, with an additional fully-connected layer on top to predict the label for that input sign from this embedding. This model only sees signs in isolation, meaning it will not learn from distributional information.

Model	Input Type	Embedding Sizes <sup>1</sup>	Other Parameters	Description
<b>Text-Only Models</b>				
glove	seq. of sign names	16, 32, 64, 128, 256	window size: 15	Pennington et al. 2014
fasttext.cbow	seq. of sign names	16, 32, 64, 128, 256	window size: 15	Bojanowski et al. 2017
fasttext.skip	seq. of sign names	16, 32, 64, 128, 256	window size: 15	Bojanowski et al. 2017
word2vec.cbow	seq. of sign names	16, 32, 64, 128, 256	window size: 15	Mikolov et al. 2013a
word2vec.skip	seq. of sign names	16, 32, 64, 128, 256	window size: 15	Mikolov et al. 2013a
lm.text	seq. of sign names	64	hidden dimension: 64	Figure 6.1, blue omitted.
<b>Image-plus-Context Models</b>				
lm.image+text	seq. of sign names and images	64	hidden dimension: 64 image size: 64×64	Figure 6.1.
lm.image	seq. of sign images	64	hidden dimension: 64 image size: 64×64	Figure 6.1, red omitted.
<b>Image-Only Models</b>				
image recognition	individual sign image	64	image size: 64×64	Figure 6.1, blue only.

Table 6.1: List of models considered in this work.

This gives us a contextless baseline to compare against: if a result holds for the multimodal LM in Figure 6.1 but not for this image recognition model, this suggests that the result depends on contextual signals, and not simply on visual resemblances between signs.

We also train continuous bag-of-words (CBoW) and skipgram models with FastText<sup>2</sup> (Bojanowski et al. 2017) and word2vec (Mikolov et al. 2013a), as well as GloVe embeddings (Pennington et al. 2014). Table 6.1 summarizes all of these models and important hyperparameters. We train each of these models on our proto-Elamite corpus, treating each entry of a tablet as a distinct sequence, and setting aside 500 entries as a validation set for the language models.

Prior to training, we replace all signs occurring 3 or fewer times<sup>3</sup> with UNK. We replace rare signs wherever they occur, including inside of CGs. The tokens X and . . . represent broken or unreadable signs, so we also replace these with UNK. When training language models, we do not backpropagate losses from samples where the target word is UNK, since it so often represents broken material. To make the data less sparse, we remove annotations marking sign variants, so that for example M157  and M157~a  are considered the same sign. (This is a hackish approach to distance our analysis from potential label bias arising from the working sign names, which must unfortunately be used as target classes for the language modeling tasks even when the model uses sign *images* as input. In Chapter 9

<sup>1</sup>For a more robust comparison, we train these simpler models with multiple embedding sizes (including some larger than those used for the more powerful language models), and report results from whichever dimensionality performs best on each task.

<sup>2</sup>Sign names are largely arbitrary, so we disable sub-words in FastText by setting the maximum sub-word length to 0.

<sup>3</sup>To determine frequency, we count how often a sign occurs both independently and as part of a complex grapheme.

we will consider an improved approach that lets our models fully divest from the working sign names.)

## 6.2 Experimental Results

### 6.2.1 Additive Composition

We predict that if a complex grapheme is semantically compositional, its embedding will approximately equal the sum of the embeddings of the signs it comprises.

Given a sign  $s$ , let  $e_s$  denote the embedding of  $s$ . If  $s$  is a complex grapheme let  $\sigma(s)$  denote the list of signs which make up  $s$ . For every complex grapheme  $s$  in the signary, we check whether  $\sum_{t \in \sigma(s)} e_t \approx e_s$ . If  $\sum_{t \in \sigma(s)} e_t$  is within the  $k$  nearest neighbors of  $e_s$  for some threshold  $k$ , we say that  $s$  appears to have a compositional representation. For different thresholds  $k$ , we measure how many complex graphemes have compositional representations and report these values in Table 6.2. We also compute how many complex graphemes  $s$  fall within the  $k$  nearest neighbors of the sum of  $|\sigma(s)|$  randomly-sampled sign embeddings; bold cells in the table represent cases where the observed number of compositional graphemes is significantly higher than expected based on this random baseline.

Model	$k$				
	1	3	5	10	15
glove.256	0	0	1	3	<b>13</b>
word2vec.cbow.16	0	0	1	9	<b>12</b>
word2vec.skip.32	0	0	5	<b>13</b>	<b>16</b>
fasttext.cbow.128	0	2	3	5	9
fasttext.skip.128	0	3	<b>10</b>	<b>15</b>	<b>20</b>
lm.text.64	0	0	0	0	1
lm.image+text.64	1	<b>14</b>	<b>21</b>	<b>40</b>	<b>51</b>
lm.image.64	<b>11</b>	<b>16</b>	<b>27</b>	<b>48</b>	<b>61</b>
image recognition.64	<b>3</b>	<b>7</b>	<b>15</b>	<b>28</b>	<b>38</b>

Table 6.2: Number of compositional graphemes for different similarity cutoffs  $k$ . Bold numbers represent cases where the number of compositional graphemes is significantly larger than expected by chance.

In text-only models, when  $k$  is small the number of complex graphemes with compositional representations is never higher than expected by chance. However, for image-aware models, and for text-only models with large enough  $k$ , the number of complex graphemes which are close to the sum of their components *is* significant. We note that even for  $k = 15$ , the signs identified as compositional by the `lm.image.64` model average  $>0.97$  cosine similarity to the sum of their parts, suggesting this is not too generous a threshold.

Most notably, the number of compositional graphemes in `lm.image.64` is always larger than the number in any of the other models, *including* the image recognition model.<sup>4</sup> This has the important implication that compositionality in the learned embeddings *is not solely a consequence of visual compositionality*. If these models were purely learning patterns of *visual* composition, the contextual information available to the LMs would not be useful for this task, and the image-based LM would not be expected to discover any more compositional graphemes than the contextless image recognition model. Moreover, we would not expect to find a significant amount of compositionality in any of the text-only models for any  $k$ , as we intentionally avoid the use of subword units which would reveal the internal structure of a sign to these models.

Thus there is at least tentative reason to believe that proto-Elamite includes some complex graphemes with meanings that are semantically compositional, where this compositionality is reflected in models’ learned representations through additive relations similar to those found for compositional phrases in modern languages (Mikolov et al. 2013b; Salehi et al. 2015; Cordeiro et al. 2016). Table 6.3 provides examples of signs which appear to be compositional in the image LM but not the image recognition model. These are signs for which contextual information appears to play the deciding role in making them receive compositional embeddings, and therefore these are the signs which we can most confidently point to as examples of possible semantic compositionality in the proto-Elamite signary. Implications for these and other signs will be discussed in Section 6.3.

M153	+	M106	≈	M153+M106
M175	+	M286	≈	M175+M286
M327	+	M348	≈	M327+M348
M362	+	M244	≈	M362+M244
M157	+	M288	≈	M157+M288
M175	+	M153	≈	M175+M153
M218	+	M388	≈	M218+M388

Table 6.3: Sample of signs which appear to be compositional in the image LM but not the image recognition model.

### 6.2.2 Pairing Consistency

Fournier et al. (2020) introduces a metric called the pairing consistency score (PCS), which is intended to measure whether the offsets between pairs of embeddings are more parallel than expected by chance. We suggest that this metric may provide initial intuitions about whether any given sign contributes the same or similar meanings to all of the complex graphemes where it occurs. If the sign  $s$  always contributes the same meaning whenever it occurs in a complex, then the offset between the embeddings for the signs  $(t, t+s)$  should be

<sup>4</sup>The image recognition model has fewer parameters than the LMs, but it attains >99% accuracy on its original task, suggesting that it is not underparameterized.

expected to be roughly parallel to the offset between the pair  $(u, u + s)$  for most choices of  $t$  and  $u$ , assuming that embeddings are organized into semantically-coherent neighborhoods as has been generally observed for representations learned through language modeling. By contrast, if complexes containing  $s$  have idiomatic meanings (so the contribution of  $s$  is not consistent), we expect that the offsets between such pairs will be parallel with much lower probability. Thus PCS can serve as a proxy for compositionality, and allows us to investigate the impact of an individual sign on the representations of complex graphemes in which it occurs.

To apply PCS to our data, we will construct two relations for each sign  $s$ . Given a complex grapheme  $c$  containing  $s$ , let  $\delta(c, s)$  denote the unique element of  $c$  which is not  $s$ .<sup>5</sup> For example,  $\delta(\text{M153} + \text{M320} + \text{M153}, \text{M320}) = \text{M153}$ . Further, let  $I(s)$  be the set of all complex graphemes with  $s$  as the inner element and  $O(s)$  be the set of all complex graphemes with  $s$  as the outer element. Then define

$$R_{s,in} = \{(\delta(c, s), c) \mid c \in I(s)\}$$

$$R_{s,out} = \{(\delta(c, s), c) \mid c \in O(s)\}$$

Informally, the relation  $R_{s,in}$  contains every complex grapheme where  $s$  is the inner part, paired up with whatever sign makes up the outer part.  $R_{s,out}$  contains every complex grapheme where  $s$  is the outer part, paired up with whatever sign comprises the inner part.

Table 6.4 reports the average PCS of the relations  $R_{s,in}$  and  $R_{s,out}$  for each model, averaged across all signs  $s$ .<sup>6</sup> On average, we find that  $R_{s,in}$  has higher PCS than  $R_{s,out}$ ; a Mann-Whitney U test (Mann and Whitney 1947) suggests that the difference is statistically significant with  $p < 0.05$  for the image-aware LMs, the image recognition model, and FastText.

This implies that, relative to outer signs, inner signs have a more consistent and predictable impact on the representations of compounds in which they occur. In other words, supposing we know the positions of signs  $A$ ,  $A+B$ , and  $C$  in the embedding space, we can roughly predict the position of  $C+B$  by starting from the position of  $C$  and adding the same offset that maps  $A$  to  $A+B$ . In the other direction, if we know the positions of  $A$ ,  $B+A$ , and  $C$ , it is on average harder to predict the location of  $B+C$  through the same process. The fact that this holds for some text-only models as well as for the image-aware LMs implies that it is due to distributional properties of signs and not simply their appearance.

Fournier et al. (2020) note that different categories of relations in English have different average PCS. They find that relations involving inflectional morphology (for example,

<sup>5</sup>We limit the analysis to complex graphemes of type  $A+B$  or  $A+B+A$ , which account for the majority of cases in the corpus.

<sup>6</sup>We compute PCS using the original code published by Fournier et al. (2020), but we adjust their permutation-finding function to avoid infinite loops when a relation contains few items.

Model	Mean PCS	
	$R_{s,in}$	$R_{s,out}$
glove.64	0.542	0.544
word2vec.cbow.64	0.525	0.492
word2vec.skip.64	0.521	0.495
fasttext.cbow.64	<b>0.562</b>	<b>0.484</b>
fasttext.skip.64	<b>0.539</b>	<b>0.500</b>
lm.text.64	0.465	0.529
lm.image+text.64	<b>0.719</b>	<b>0.482</b>
lm.image.64	<b>0.760</b>	<b>0.536</b>
image recognition.64	<b>0.929</b>	<b>0.493</b>

Table 6.4: Comparison of pairing consistency for the inner and outer parts of compound signs in 64-dimensional models. Bolded rows represent pairs where the difference between columns is significant.

between a verb and its gerund) have high PCS, relations involving derivational morphology (as between *heat* and *reheat*) have lower PCS, and other semantic relations (as between *hot* and *cold*) have the lowest PCS of the relations they examine.

We expect that absolute PCS values will not be comparable between proto-Elamite and English, owing to the very different nature of the two writing systems. However, it may be possible to draw broad comparisons between different categories. As the category with the highest PCS, inner signs appear to pattern with inflectional morphology, while outer signs pattern more closely with regular lexical items. This does not imply that inner signs actually encode inflectional morphology: most proto-Elamite signs likely correspond to objects or logograms, and most types of morphological marking were absent in the earliest phases of Near Eastern writing (Nissen et al. 1993). Rather, we interpret these results as suggesting that inner signs offer a minor refinement to the meaning of an outer sign without fundamentally changing its value, parallel to the way that inflecting a verb refines its role in a sentence but does not change its basic meaning.

### 6.2.3 Analogy

Our PCS results measure sign behaviour in aggregate, but do not provide specific examples of relations between signs. We augment these results by searching for concrete analogies which hold in the embedding models.

Given two complex graphemes  $s$  and  $t$  with embeddings  $e_s$  and  $e_t$ , let  $s - t$  denote the signs that are in  $s$  but not  $t$ , and let  $s \cap t$  denote the signs both complex graphemes have in common. Consider the vector

$$A(s, t) = e_s - \sum_{u \in s-t} e_u + \sum_{v \in t-s} e_v$$

This vector represents the analogical formula  $s : (s - t) :: t : (t - s)$ . If  $A(s, t) \approx e_t$  in a particular embedding model, then this analogy appears to hold true according to that model.

We compute how often  $A(s, t)$  is within the  $k$  nearest neighbors of  $e_t$  for different thresholds  $k$  when  $s \cap t \neq \emptyset$ . We also compute how often  $A(s, t)$  is close to  $e_t$  when  $s$  and  $t$  are complex graphemes sampled from the sign list uniformly at random. We predict that complex graphemes which have signs in common will also have some meaning in common, and consequently that the former value will be significantly larger than the latter random baseline.

Table 6.5 shows the results of this evaluation. As in the compositionality task, more analogies hold between complex graphemes with shared components in image-aware models than in text-only models, and the largest number by far occur in the image LM. Once again, in `lm.image.64` the target vector averages  $>0.97$  similarity to the computed vector even when  $k = 15$ . Bold numbers in the table represent cases where analogies are significantly more likely to hold between complex graphemes with shared components than between random pairs of complex graphemes. We see that the number of analogies is larger than expected by chance even in some text-only models, suggesting that there is a meaningful relationship between some complex graphemes which have elements in common. The fact that the image LM outperforms the image recognition model further implies that these analogies reflect legitimate distributional properties and are not purely due to visual resemblance.

Model	$k$				
	1	3	5	10	15
glove.256	0	8	11	25	48
word2vec.cbow.256	0	17	36	<b>65</b>	<b>90</b>
word2vec.skip.128	0	8	29	<b>97</b>	<b>140</b>
fasttext.cbow.128	0	9	22	<b>64</b>	<b>98</b>
fasttext.skip.256	0	11	30	91	<b>145</b>
lm.text.64	0	2	7	16	21
lm.image+text.64	<b>27</b>	<b>82</b>	<b>134</b>	<b>233</b>	<b>320</b>
lm.image.64	<b>69</b>	<b>172</b>	<b>258</b>	<b>393</b>	<b>521</b>
image recognition.64	<b>29</b>	<b>67</b>	<b>92</b>	<b>133</b>	<b>174</b>

Table 6.5: Number of analogies which hold between complex graphemes with signs in common, for different similarity cutoffs  $k$ . Bold numbers represent values which are significantly larger than expected by chance.

Taken altogether, the results throughout this section suggest that many complex graphemes have compositional meanings which can be understood by comparison to the meanings of their component parts and the other complex graphemes with which they share components.

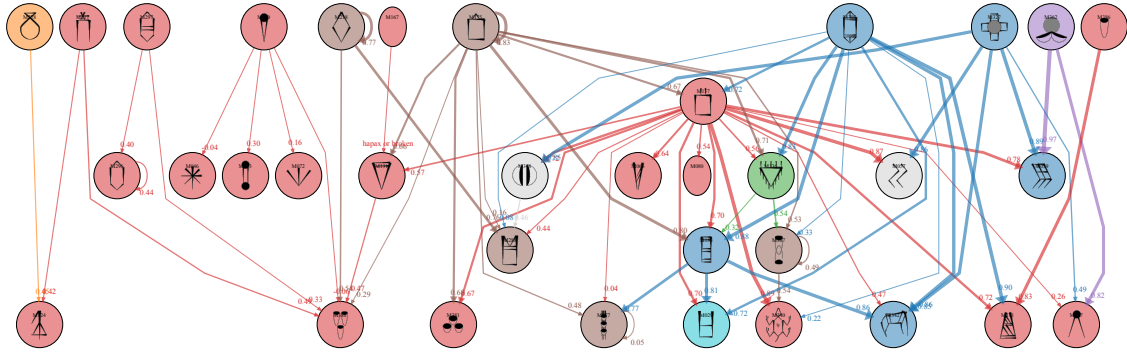


Figure 6.2: Containment hierarchy for a subset of the signs which can occur in CGs. Directed edges point from outer signs to the inner signs they can contain. Note that (excluding self-loops) the graph is acyclic and all edges point from higher nodes to lower ones. Thicker edges represent complex graphemes which are more strongly compositional. Nodes are colored according to modularity class (Blondel et al. 2008) such that nodes are most strongly connected to like-colored nodes. Full hierarchy, showing all signs which occur in complex graphemes, is available in Appendix A.6.

#### 6.2.4 Complex Grapheme Containment Hierarchy

Some signs which occur as the inner part of one complex grapheme may also occur as the outer part of another, as with M348  $\left(\left(\right)\right)$  in M327+M348  $\left(\left(\right)\right)$  and M348+M004  $\left(\left(\right)\right)$ . In principle, one may therefore expect to find pairs of signs A and B where A can contain B in some complex graphemes, while B contains A in others. In truth, *no* such pairs actually exist. In fact, if we draw a directed arc from every outer sign to each inner sign it can occur with, we observe that the resulting graph of containment relations is acyclic (excluding self-loops)—in other words, it appears as though complex graphemes are constructed according to a kind of hierarchy, whereby any given sign is only permitted to enclose itself or another sign which is lower on the hierarchy. This is visualized in Figure 6.2, excerpted from the full hierarchy available in Appendix A.6.


We quantify the compositionality of a complex grapheme as the cosine similarity between its embedding and the sum of the embeddings of its components according to the `lm.image+text.64` model, and we adjust the thickness of each edge in the graph to reflect this quantity. This reveals an apparent relation between a sign’s compositionality and its position in the graph. The signs on the left half of Figure 6.2 have low compositionality on average (seen as thinner edges in the figure) while the nodes to the right have higher compositionality (seen here as thicker edges). This suggests that there may exist different subtypes of complex grapheme, of which some are more idiomatic than others, and that these types have sufficiently little overlap to appear as separate modules when graphed.

This “grammar” governing complex grapheme construction has not been noted in previous scholarship. The ordering of signs within this hierarchy deserves attention in future



work, as it may reflect different levels of administrative units in proto-Elamite society, degrees of specificity in qualifying commodities, or other information which can be exploited to understand the meaning of these complex signs and the content of texts where they occur.

### 6.3 Analysis







Little is known about the role of complex graphemes in proto-Elamite, although these signs make up a significant portion of the corpus. Some occur in “headers” appearing at the beginning of a text. In headers, outer signs (such as M157 ) are hypothesized to indicate the type of household or institution to which the entire account relates. The outer sign may be further specified by an inner sign, but many (including M157) can also appear without another sign inscribed within. Inner signs are hypothesized to specify a particular kind of item being recorded, a person, profession, or administrative department related to an account, and more.







Our results appear to be consistent with these hypotheses. The PCS results point to inner signs playing a specializing role; this is corroborated by visual inspection of the embedding space, which reveals that complex graphemes cluster according to their outer sign rather than their inner sign (cf. Figure 6.3 below).

According to Table 6.2, our text-only models detect additive composition in at most one of every 10 complex graphemes; the image LM detects it in one of every 4 complex graphemes. The image LM suggests that a meaningful analogical relation obtains between slightly less than one-third of all pairs of complex graphemes with signs in common. These values depend on the threshold  $k$ , but even in the worst case they suggest that a non-trivial number of complex graphemes exhibit some degree of compositional behaviour. For graphemes where our models do not detect any kind of compositionality, it is not clear whether this should be taken to mean that the signs in question are truly non-compositional, or that our model has simply failed to capture compositionality which is really present. However, the fact that compositional and non-compositional graphemes appear to be separated from one another in certain parts of the containment hierarchy in Figure 6.2 lends some credence to the view that these signs may be legitimately non-compositional.

Based on the knowledge of other early writing and proto-writing systems, we can make some inferences about the complex graphemes which are compositional. They are not likely to represent either combinations of ideograms with an emergent lexical value (like the Sumerian cuneiform sign for *naη* “drink” combining the signs for human head and water) or ideograms with phonetic complements (signs indicating the proper reading of the complex), as both cases should be expected to produce non-compositional meanings. Our results may also counter-indicate a “heraldic” usage whereby abstract “charges” are combined to create an emblem identifying a group or individual, since we show that the components of com-

plex graphemes can often be understood in relation to their use elsewhere in texts, and since complex grapheme elements on their own often seem to reference products (including foodstuffs and livestock) and their distribution. Future work may train embedding models on proto-cuneiform, a structurally-similar writing system containing compound signs with occasionally known meanings that could act as useful points of comparison.

The two components of a complex grapheme can occur independently, within the same text or even side-by-side. A dramatic example comes from M218+M288 , the components of which appear 37 times as the bigram M218 M288. M288  (“grain container”) is the most frequent sign in proto-Elamite, appearing in diverse contexts but often before numerical measures of capacity. M218  is among the signs speculated to function “syllabically” to write personal names, though it may also have other uses. It is not clear yet whether M218+M288  and M218 M288   operate identically, particularly since M218+M288 is not strongly additively compositional in any of our embedding models. The possible polyvalence of M218 and broad distribution of M288 may impact models’ ability to detect compositionality in M218+M288. Despite this difficulty, the image LM identifies analogies between M218+M288, M175+M288, and M305+M288 (the analogy vector has >0.99 cosine similarity to the target in both cases) implying that we should at least consider M218, M175, and M305 as parallel categories each with relation to grain capacities.

Some signs rarely occur outside of complex graphemes, such as the productive inner sign M342 , about which practically nothing is known. Our data show that it has moderately high PCS (0.69 in `lm.image.64`) and that analogies hold between all but one of the complex graphemes which contain M342 (M157+M342 [no image available], M304+M342 , M305+M342 , M325+M342 [no image available], M327+M342 , and M351+M342 , excluding M153+M342 ). These analogies hold strongly for the image LM but not the image recognition model, meaning they reflect primarily distributional properties. Many of these signs are also additively compositional. We believe that these signs may be suitable starting points for future analysis, as our results imply that they are probably not idiomatic and are likely to have related meanings to one another.

M157+M342	: M157	:: M304+M342	: M304
M157+M377+M377	: M157	:: M175+M377+M377	: M175
M370+M046+M370	: M046	:: M370+M072+M370	: M072
M175+M377+M377	: M175	:: M201+M377+M377	: M201
M351+1(N14)	: 1(N14)	:: M351+M380	: M380
M036+1(N39C)	: 1(N39C)	:: M036+M035	: M035
M136+M365	: M136	:: M327+M365	: M327
M157+M057	: M157	:: M327+M057	: M327

Table 6.6: Sample of analogies which hold in the `lm.image+text.64` model.

Table 6.6 gives additional examples of analogies which hold in `lm.image+text.64`. We see that inner and outer signs both participate in analogical relations, as do both A+B-type and A+B+A-type complexes. Some analogies hold between a complex grapheme with

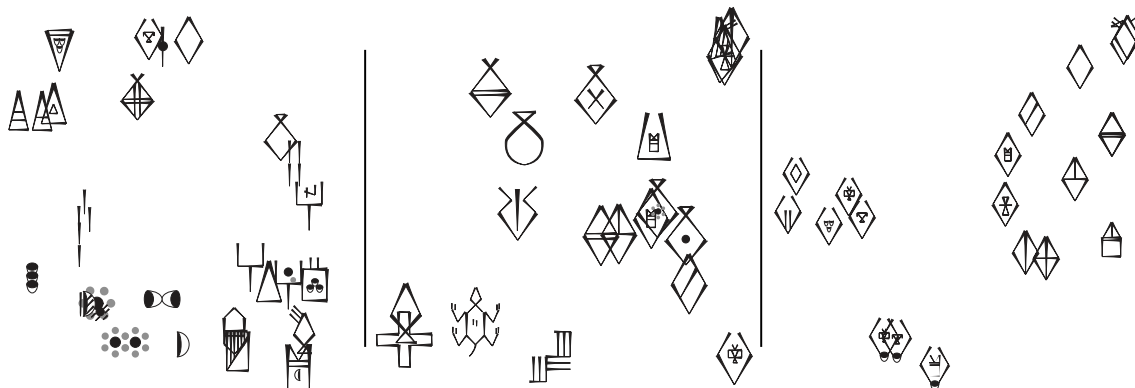







Figure 6.3: Detail from t-SNE decompositions of the GloVe embeddings (left), the image LM (centre) and the image recognition model (right).

a numeric inner sign and one with a non-numeric inner sign, as between M036+1(N39C) and M036+M035 . Such cases may have implications to the meaning of the signs involved; if 1(N39C) and M035 truly have parallel functions in these two complexes, this may imply a kind of quantifying role for M035, or alternatively that 1(N39C) is used for its pronunciation or possible syllabic value rather than as a true numeral. The existence of other M036 compounds containing numerals (e.g. M036+1(N30D) and M036+1(N14) ) would seem to favor the former interpretation.

**Sign Names** The image-only LM found stronger signals for compositionality and analogical relations than the image+text LM, suggesting that sign names may have acted as distractors for those tasks. This has significant implications for the ongoing process of revising the proto-Elamite sign list. Our work relies on the sign names assigned through an exhaustive manual transliteration process; since it is easy to automate mergers between signs should an equivalence ever be proven, this process assumes that most signs are unique from one another until shown otherwise. However, we believe this choice may obscure signals in the transliterated data by making many signs very rare. Moreover, some signs which appear graphically compositional are not currently labeled as complex graphemes, usually when the inner part is never attested as a standalone sign, which makes it hard to include these signs in our analysis despite their likely relevance. For these reasons, future work may benefit from relabeling signs based on a combination of context and sign shape, and from pursuing models which operate on visual data alone. We will pursue these lines of inquiry in Chapter 9.

We caution that some apparently minor visual differences (consider M263 and M262 ) may record distinctions which are meaningful, but very fine-grained, such as (hypothetically) “jug of red beer” versus “jug of dark beer”. Such similarly functioning signs might obtain similar embeddings, but retaining a distinction in the published transliterations still

improves our understanding of the texts. Thus any revisions to the sign list will need to be informed by evidence from multiple sources, and a level of detail which is appropriate for one analysis may not be suitable for another. This observation was a contributing factor in the design of CLEE (Chapter 3), which replaces the monolithic ATF transliterations with richer and more structured representations for individual tokens, among which we can select the desired level of representation as needed.

To conclude this analysis, Figure 6.3 shows details from the embedding spaces learned by GloVe, the image LM, and the image recognition model.<sup>7</sup> GloVe produces small clusters of visually similar signs even though it does not have access to sign images: note the proximity of M353 , M354 , and 2(N30C) , as well as the variants of M036 . These clusters occur in sufficient number that we have confidence the model is detecting meaningful similarities in the usage of visually similar signs. The image recognition model produces much clearer groupings of visually related signs, as would be expected. The image LM replicates some clusters from the image recognition model: a cluster of lozenge-shaped signs is visible in both the image LM figure and the image recognition figure. However, contextual information causes the image LM to relocate other lozenge-shaped signs like M218  to a different part of the embedding space, implying a functional difference between it and the signs in the figure. Overall, these observations confirm that our multimodal architecture is finding a balance between contextual and visual features as intended.

## 6.4 Related Work

Sun et al. (2019) introduce “character-enhanced” embeddings of Chinese words. Their architecture roughly parallels our own, but requires a deeper CNN due to the visual complexity of Chinese characters. We train with a full context language modeling objective whereas they use a sampling scheme similar to word2vec. They use character-level information to improve word embeddings, where we exclusively learn character embeddings. Our application of this architecture to decipherment is novel.

Liu et al. (2017) explicitly learn compositional embeddings for Chinese characters. They use supervised data to help identify when two visually-distinct signs use the same radical (as in 水 and 池). In our data, it is not known which signs are truly related to one another, thus we refrain from giving the model explicit information about compositionality.

Yin et al. (2019) segment and transcribe undeciphered scripts based on visual similarities between glyphs. Although their transcription error rate is high, they still achieve partial decipherments with no human intervention.

<sup>8</sup>Full figures are available in Appendix A.5.

Dencker et al. (2020) perform OCR-style sign detection on images of Sumerian cuneiform tablets, recognizing signs which may be written very differently across the corpus. Their task benefits from the existence of supervised Sumerian training data.

Luo et al. (2019) perform automated decipherment of Ugaritic. Their technique finds alignments between orthographic representations of phonetic information, and thus is not easily applicable to a script which may have a strong ideographic or logographic component. Their approach also requires multilingual data, and cannot extract information from a script with no known surviving relatives.

Our work exploits the embedding space learned by a neural language model, but the actual task of language modeling is otherwise irrelevant to our results. By contrast, Kambhatla et al. (2018) actually sample text from a neural language model to help estimate the quality of a proposed decipherment. Future work could similarly sample from a language model as a means of counteracting the small size of the proto-Elamite corpus; this should be done with caution, however, given the difficulty of evaluating whether the sampled text is fluent.

Salehi et al. (2015) and Cordeiro et al. (2016) demonstrate that English word embeddings tend to be additively compositional and can capture human intuitions about semantic compositionality. Hartung et al. (2017) investigate other methods for decomposing word embeddings.

Sproat (2006) discusses a variety of writing systems and the degrees to which they employ phonetic versus semantic information. The discussion is largely taxonomic and addresses subtle nuances between scripts which are already well-understood. In this way it demonstrates the wide range of variation observed between scripts, and by extension the range of possibilities which should be considered when analyzing an undeciphered script such as proto-Elamite.

## 6.5 Conclusion

Interpreting what a word embedding model has learned typically involves a comparison to native speaker intuitions. In contrast, this chapter has shown how carefully controlling the amount of visual and contextual information available to a model can lead to new insights in a setting where native speaker intuitions are unavailable. Abstracting away from human annotations, we introduced a novel architecture for multimodal or image-based language modeling, which shares information between visually similar signs to better model contextual patterns. This provides a new toolkit for decipherment of an unknown language, distinct from translation-based approaches.

As one of the world’s earliest experiments in writing, employing 774 signs and variants by current estimates, reasonable concerns have existed over proto-Elamite’s level of standardisation and the impact this may have on decipherment (Dahl 2019: 71, 82). The corpus


is small and filled with lacunae, and prior work has done little to understand how NLP techniques function on early writing systems which may reflect linguistic content differently from modern writing systems. Despite these challenges, this chapter has shown that embedding models can indeed identify patterns in proto-Elamite that appear to capture both new and existing intuitions about this script.

We have presented evidence that a subset of complex graphemes are semantically compositional rather than idiomatic, and we have discovered the existence of a simple grammar or hierarchy which appears to govern the construction of complex graphemes. Our results shed new light on this class of signs and suggest new avenues by which aspects of this script can come to be better understood.

## Chapter 7

# Headers

Specialists have hypothesized that proto-Elamite texts frequently begin with a “header”, that is, a sign (or string of signs) which “qualifies all transactions recorded in a text” by specifying an institution or owner in charge of the associated account (Damerow and Englund 1989: 14–16). This understanding of headers depends in part on the claim that they correspond to visually demarcated “colophons” in proto-cuneiform accounts (Englund 2004: 144; Damerow and Englund 1989: 15); however, these are also largely undeciphered and so it is not certain that they consistently convey ownership information.

Some (but not all) of the signs that occur at the beginning of texts have been tentatively labeled as headers by domain specialists. This labeling is recorded using comments in the CDLI transliteration of the texts; no explicit list of header signs has been published. The clearest example of this category is the ubiquitous sign M157 , which occurs at the start of fully one-fifth of all proto-Elamite accounts. Most header signs, including M157, may also appear elsewhere in texts, where they have an uncertain function.

In light of modern scholarship’s very partial understanding of the proto-Elamite corpus, there does not seem to be proof beyond reasonable doubt that headers record ownership, much less that *all* headers do so. Moreover, headers have thus far been identified through manual analysis which has not been fully documented in any publication, and some of the experts who originally identified this category are no longer alive. Thus the criteria for identifying headers are opaque and the question of their existence is a matter of qualitative judgement in some texts.

In this chapter, we combine computer-aided analysis with domain expertise to undertake the first focused study of headers in proto-Elamite. We first use statistical and neural sequence models to show that headers *are* a genuine structural phenomenon in proto-Elamite. We demonstrate that it is possible to independently replicate manual annotations from past work with high accuracy using features from our models, and our models also identify and allow us to correct a number of annotation mistakes. Based on our results, we argue against the conventional understanding that most headers span a single sign, suggesting rather that two- and possibly even three-sign headers are a much more prevalent phenomenon than cur-

rent transliterations would suggest. In conjunction with this, we show that signs in the first and second positions of a text predict distinct information, suggesting that signs in these positions have disparate functions.

This chapter reproduces material originally published in Born et al. 2022.

## 7.1 Methodology

The transliterated proto-Elamite corpus includes rich annotations, such as notes about which signs (if any) are understood to comprise a text’s header. We propose to train two unsupervised sequence models on the proto-Elamite corpus and assess whether these models suggest any internal structure at the beginning of texts, and whether and to what extent the structures so identified correspond with the existing header annotations. We aim to arrive at a richer understanding of the meaning and purpose of headers by (i) examining which features are useful for predicting whether a text has a header; (ii) finding correlations between these features and other document properties; and (iii) identifying why unsupervised models may disagree with (or fail to recover) the human labeling if such disagreements occur. We hope to provide new and quantifiable evidence that headers are a real structural phenomenon in proto-Elamite, and to be able to more concretely justify why any given text may have or not have a header. Our goal is not to indiscriminately replicate the human labeling using automated tools: rather, we seek to *assess* and *understand* the received wisdom in the human labeling through comparisons to interpretable models.

### 7.1.1 Hidden Markov Model

Hidden Markov models (HMMs; Cave and Neuwirth 1980) have become a standard tool for unsupervised analysis of undeciphered text corpora in previous literature. We fit a 15-state HMM to our corpus; this number of states was chosen to slightly exceed the number of different sign categories which can be informally speculated to occur in proto-Elamite (most saliently, headers, counted objects, syllables, owners, other kinds of qualifier, numerals, and subscripts, with some overlap between these). We train ten models from random initializations, using complete tablets including numerals as input sequences; we keep the model which assigns the highest likelihood to the corpus. For each tablet, we compute the optimal state sequence according to this model using Viterbi decoding (Rabiner 1989). We hypothesize that, if headers exist, their existence will be reflected in the HMM by a state which only occurs at the very beginning of texts, and only in *select* texts. If such a state does not exist, it may mean that headers are not a salient structural feature of the corpus; if such a state exists, but is not associated with texts where human annotators believe there to be a header, it may imply that current understandings of headers somehow fail to reflect the true distribution of this structure.



### 7.1.2 Transformer

We also train an autoregressive Transformer (Vaswani et al. 2017) language model from a random initialization using the vanilla `fairseq` recipe.<sup>1</sup> Neural architectures such as the Transformer offer significantly greater inferential power than statistical models like the HMM, though the large amounts of data required for training can make them unsuitable for extremely low-resource archaeological data. For the present work, we are purely interested in using our models as analytic devices (i.e. feature extractors), and we neither require nor expect them to generalize. For this reason, we proceed with training a Transformer language model as a more powerful alternative to the HMM, with full knowledge that it will overfit to our low-resource corpus.

Under the hypothesis that headers convey information which is relevant to the interpretation of a tablet as a whole, we predict that the language model will attend to the beginning of a tablet on all or most time steps if that tablet has a header. In texts without a header, the beginning of the document will contain no such special information, and thus should not be expected to receive stronger attention than any other part of the text. Thus, if headers are a legitimate structural phenomenon, we should observe two classes of text which are differentiated by the average amount of attention paid to their initial signs.

Formally, let  $z_{i,j}$  denote the self-attention score for token  $t_i$  at time step  $j$ , and for a sequence of length  $L$  let  $n_i = L - i - 1$  denote the number of tokens following  $t_i$ . Then  $\tilde{z}_i = \frac{1}{n_i} \sum_{j>i} \frac{z_{i,j}}{\max_k z_{k,j}}$  is the average self-attention paid to  $t_i$  by the rest of the document. This is essentially the mean of the self-attention scores for  $t_i$  across all following time steps (which would be  $\frac{1}{n_i} \sum_{j>i} z_{i,j}$ ), except that we have normalized the scores at each time step so that the largest is always 1 (this controls for text length, as the true mean tends to zero as text length increases). For a given text and indices  $m$  and  $n$ , let  $\tilde{\mathbf{z}}_{m,n} = [\tilde{z}_m, \tilde{z}_{m+1}, \dots, \tilde{z}_{n-1}]$  denote the average attention paid to tokens  $t_m$  through  $t_{n-1}$ .

The first numeral of a text gives an upper bound on the length of that text’s header, if it has one. Hereinafter, let  $n$  stand for the number of signs which precede the first numeral of a given text (each tablet thus has its own value of  $n$ ). We hypothesize that each text’s  $\tilde{\mathbf{z}}_{0,n}$  will capture information about whether that text has a header, and therefore (if headers are a real structural phenomenon) that a logistic regression over  $\tilde{\mathbf{z}}_{0,n}$  should be able to accurately predict which texts human experts have annotated as having a header. Later sections of the text should be less predictive; thus, as a baseline, a logistic regression over  $\tilde{\mathbf{z}}_{10,20}$  (or, equivalently, any other arbitrary span of signs believed to lie outside the putative header) should *not* be able to predict the expert annotations.

<sup>1</sup>[github.com/facebookresearch/fairseq](https://github.com/facebookresearch/fairseq)

### 7.1.3 Training

We train the HMM and Transformer LM on sequences of sign names, where each sequence spans a single document. We omit all annotations, such as those marking damaged signs: this reduces the vocabulary size and makes the distribution for most signs less sparse. We set aside 200 complete tablets for the Transformer to use as a validation set for the language modeling task.

As we are interested in tablet headers, we only evaluate our models on texts where the beginning is substantially intact. If a text’s transliteration contains the comment “beginning broken”, if there is a prime ‘ in the first line number of the transliteration, or if the first sign is X or [...], we omit that tablet from our analysis. After this pruning we are left with 795 documents.

We construct the mean attention vectors  $\tilde{\mathbf{z}}_{0,n}$  and  $\tilde{\mathbf{z}}_{10,20}$  for each text in the pruned corpus (where  $n$  differs for each text, according to how many signs that text has before its first numeral). We zero-pad the  $\tilde{\mathbf{z}}_{0,n}$  vectors to the length of the longest, and train two logistic regressions to predict whether human experts annotated a text as having a header: the first is trained on the set of (padded) text-initial vectors and the second on the set of text-internal vectors. In both settings, the most accurate model is selected using 10-fold cross validation.

## 7.2 Experimental Results

### 7.2.1 Hidden Markov Model


Encouragingly, the Viterbi sequences from our HMM exhibit a heavily skewed state distribution at the beginning of tablets. Specifically, 55% of all texts begin in state 7, and a significant majority of these cases (76%, or 42% of texts overall) *only* exhibit state 7 on the very first sign. A mere 7 tablets (0.8% of the total) exhibit this state on or after the 4th sign. Thus state 7 is strongly localized to the beginning of tablets.

The fact that the model learns such a strongly localized state suggests that some documents *do* have a discernible internal structure and that the beginning of these texts is measurably distinct from what follows. This is fully consistent with the hypothesis that headers exist as a structural phenomenon within proto-Elamite.

The contingency table in Table 7.1 allows us to assess whether HMM state 7 captures the same information as the headers identified by human annotators. We observe that state 7 recovers the human labels with high precision (0.93) but low recall (0.67), for an overall accuracy of 0.70. This could imply that the HMM has failed to recover some crucial feature that human annotators used to identify headers; that human annotators have proposed headers in some contexts where no header truly exists; or that HMM state 7 captures some finer-grained category than is encompassed by the specialist’s monolithic header annotation.

Initial HMM State	Expert Annotation		$\Sigma$
	Header	No Header	
State 7	410	30	440
Other	205	150	355
$\Sigma$	615	180	795

Table 7.1: Contingency table comparing the initial state of a tablet’s Viterbi sequence against the presence of a header annotation in the tablet metadata.

Examining the state sequences from some sample texts (Figure 7.1) helps in comparing these possibilities. In sequence (i), human annotators identified M388  as a header, whereas the HMM places this sign in state 3 rather than the putative “header” state 7. M388 is a very common sign in the body of tablets, and we have already discussed its unique distribution in Section 5.2. The HMM clearly recognizes this uniqueness, and learns a state which is almost exclusively used for M388 and the other “owner” signs discussed in Section 5.2. Most instances of M388 are followed by the so-called “syllabic” signs, which the model also appears to identify using a consistent state (14, as seen in both sequences of Figure 7.1). The M388 in sequence (i) is followed by syllabic signs and looks like other typical examples of this sign, making it unclear why a header was identified here by human annotators (especially given that other tablet-initial M388s are not labeled as headers in the expert annotations). This tablet also contains some unreadable signs (denoted by X), which appear to confound the HMM in most texts where they occur. The model typically predicts state 0 whenever it observes an X, and continues to predict state 0 for every subsequent sign, even when that sign is common and receives a more interpretable state in other contexts. We see this behaviour in sequence (i), where the model remains in state 0 even when seeing the intact numeral sign 2(N48). Thus, although the presence of a header in this text may in fact be questionable, the fact that the model falls into this failure state calls into question the validity of the Viterbi sequence, and suggests that an HMM may lack the power to completely and accurately model this corpus.

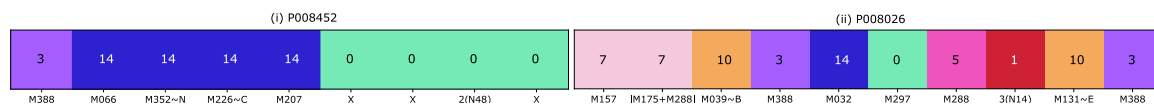


Figure 7.1: Illustration of state sequences learned by the HMM. The observed sequence of sign names is shown on the  $x$ -axis (truncated to at most 10 signs); the numbers in the cells report the states in the Viterbi sequence. Each color represents a distinct state. (i) HMM state 3 does not suggest the presence of a header, though one is present in the expert annotations; (ii) HMM state 7 suggests the presence of a header, which is present in the expert annotations.

### 7.2.2 Transformer

The logistic regression trained on  $\tilde{\mathbf{z}}_{0,n}$  is able to predict whether human annotators identified a header in a text with 92% accuracy. By contrast, the model trained on  $\tilde{\mathbf{z}}_{10,20}$  only achieves 77% accuracy, which is the same score achieved by simply predicting the majority class.

Thus the Transformer’s behaviour at the beginning of a text, as expressed through self-attention scores, *is* predictive of expert opinions about the presence of a header in that text, but these behaviours do not persist into later parts of the document. As we had hypothesized, the model attends much more strongly to the beginning of texts where experts believe there to be a header: Figure 7.2 illustrates this using heatmaps of  $\tilde{\mathbf{z}}_{0,6}$  from two texts, one of which is annotated as having a header and the other of which is not.



Figure 7.2: Heatmap of  $\tilde{\mathbf{z}}_{0,n}$  (mean attention over signs before the first numeral, truncated to length 6) for two tablets, one with a human-labeled header (left) and one without (right). Darker cells indicate stronger attention.

Table 7.2 compares the predictions from the regression over  $\tilde{\mathbf{z}}_{0,n}$  against the expert annotations and the initial HMM states. The regression achieves significantly better recall (0.97) than the initial state of the HMM, which suggests that the HMM may have failed to identify a header in many texts where one does in fact exist.

	Expert Annotation			Initial HMM State		
	Header	No Header	$\Sigma$	State 7	Other	$\Sigma$
<b>LR Predicts Header</b>	596	44	640	421	219	640
<b>LR Predicts No Header</b>	19	136	155	19	136	155
$\Sigma$	615	180	795	440	355	795

Table 7.2: Contingency table comparing predictions from a logistic regression over  $\tilde{\mathbf{z}}_{0,n}$  against (left) the presence of a header in the tablet metadata, and (right) the initial state of the Viterbi sequence.

### 7.3 Analysis

Our results are fully consistent with the prevailing assumption that the beginnings of certain proto-Elamite tablets exhibit some degree of internal structure. This is suggested by the existence of an HMM state which is strongly localized to the beginning of tablets, but which does not occur at the beginning of every text as a generic “start” state. Further evidence is seen in the behaviour of the Transformer, where in certain texts the model pays more attention than usual to early tokens. On their own, these features merely confirm that some internal structure is present, but do not tell us what that structure may represent.

In this section we interpret our models’ predictions in order to understand what factors may have motivated the original human annotations, and what features may be exploited to understand headers’ meanings.

### 7.3.1 Inter-Annotator Agreement

Table 7.3 reports inter-annotator agreement between our three approaches to labeling headers (expert annotations [Expert], initial HMM state [HMM], and logistic regression over Transformer self-attention scores [LR]). We report Cohen’s  $\kappa$  (Cohen 1960), where 1 (resp. -1) implies perfect agreement (resp. disagreement) and 0 implies no more agreement than expected if labels were assigned at random. The purpose of this comparison is not to evaluate the models’ accuracy (since it is not known that the expert labels reflect the ground truth) but rather to assess whether all three techniques recover similar information.





	<b>Expert</b>	<b>HMM</b>	<b>LR</b>
<b>Expert</b>	1.0	0.372	0.766
<b>HMM</b>	0.372	1.0	0.362
<b>LR</b>	0.766	0.362	1.0

Table 7.3: Agreement (Cohen’s  $\kappa$ ) between human and model annotations.

All techniques agree more than expected by chance. The most common disagreement comes from the HMM, which in 205 cases does not assign state 7 to a sign labeled as a header by human annotators. In 188 of these cases, the regression over Transformer attention *does* recover the human annotation, suggesting that these disagreements simply reflect the limited power of the HMM and its aforementioned susceptibility to noise from damaged contexts. Supporting this interpretation, most of these texts offer comparatively little context on which the HMM can base its decision: the majority contain unreadable signs, rare or hapax signs, or are very short. In fact, it is possible to predict whether the HMM will agree with the human annotation with better than chance accuracy simply by knowing whether the second sign of a tablet is intact, which suggests that the HMM is severely hampered by the fragmentary nature of the corpus.

Tablets that are damaged also impact the Transformer’s ability to recognize headers. There are 19 texts where the logistic regression fails to predict the presence of a header in the expert annotations, 17 of which are also disputed by the HMM. In all of these texts, either the document contains only a single readable sign, or some early signs are damaged to the point of being unreadable.

Much more interesting are the cases where the regression proposes a novel header. This occurs in 44 texts, 13 of which also begin in state 7 according to the HMM.<sup>2</sup> Encouragingly, we find among this collection 25 texts<sup>3</sup> where the manner of transliteration indicates that experts have recognised a header but did not mark this according to the usual convention.<sup>4</sup> If we correct the annotation of these texts, we find that the regression’s accuracy rises to 95% and  $\kappa$  to 0.849.

This leaves 18 cases where the regression proposes headers which are truly novel. Several of these texts are substantially intact and contain signs which are generally common and well-understood. M393~g (hapax, no image available) in P008621 appears to the specialist a plausible header, since some other variants of M393  are so labeled, although other variants in second position are not marked as headers either by experts or the models (P009486; P009209). However, M362  (P009075), typically understood as a “counted object” sign (perhaps a nanny-goat; Dahl 2005a) and the related M362+M005  (P008294) challenge the conventional expectation that headers are distinct from counted objects. M489  is unique to P009526: Damerow and Englund (1989) keep open the possibility that M489 could be a header, but express some skepticism given that it also marks the summary line on the reverse (signs in the summary are usually expected to be counted objects). Specialists have not thoroughly fleshed out the distinction between “counted object” and “institution” signs, but believe that headers typically comprise the latter. The predictions from our models suggest that it may be worth considering whether “counted object” signs can also occur in some headers by examining these particular texts in greater detail.

### 7.3.2 Multi-Sign Headers

Two-sign headers are a very marginal category in the expert annotations, occurring only five times.<sup>5</sup> By contrast, in 119 texts the Viterbi sequence stays in state 7 until the second sign of a text, and in 33 texts it stays in state 7 until the third sign. The prevalence of long headers is one of the most significant points of divergence between the human labels and HMM states.

<sup>2</sup>One of these texts, P008329, must be omitted as it has a damaged first sign. This was not removed during our data cleaning as the damage was not transliterated following the usual convention.



<sup>3</sup>Listed in Appendix A.9

<sup>4</sup>The usual convention for annotating headers is to transliterate the header signs on their own numbered line, followed on a new line by a comment that reads # header. In these texts, the first signs were given their own numbered line in the transliteration, setting them apart from the rest of the first entry, but there was no comment explicitly labeling this as a header as opposed to some other kind of logical division.

<sup>5</sup>Listed in Appendix A.9

To predict the presence of a header with the Transformer, we perform a regression over all of  $\tilde{\mathbf{z}}_{0,n}$  and therefore do not identify an explicit boundary where the header ends. However, based on the magnitudes of the coefficients in this regression, it appears that the outcome depends mainly on the attention paid to the second through fourth signs of a tablet, with mean attention to the second sign being most predictive overall. This suggests that the Transformer, like the HMM, has identified relevant structural information beyond the first sign of a tablet.

In fact,  $\tilde{z}_1$  (the mean attention paid to the second sign of a tablet) is, by itself, sufficient to predict the presence of a header with the same accuracy as the entire  $\tilde{\mathbf{z}}_{0,n}$ . Mean attention to the first sign ( $\tilde{z}_0$ ) gives the same accuracy as predicting the majority class (77%), suggesting that the first sign may be less relevant than the second to the rest of the document, despite having been the near-exclusive focus of past examinations of PE headers. We return to this discussion in Section 7.3.3, where we further explore the possible roles of the first two signs of a tablet.

In the expert annotations, most two-sign headers involve compounds of M327 , generally followed by another sign which can also occur as a header on its own. This pattern recurs in the multi-sign headers identified by the HMM, and is expanded to cover more combinations of M327 compounds with a following sign. Notably, the HMM also introduces a new kind of multi-sign header not found in the human-labeled data, comprising M157  plus a following sign. An example of this is found in the right-hand sequence in Figure 7.1, where the HMM replicates the manually-identified header but expands it to also include the second sign of the text.

### 7.3.3 Cramér’s V

Cramér’s V (Cramér and Goldstine 1946) measures relationships between pairs of categorical variables; it ranges from 0 to 1, where 0 signifies that the variables are unassociated and 1 denotes that they are perfectly associated. This section uses Cramér’s V (with the bias correction due to Bergsma 2013) to look for correlations which may have implications for the interpretation of headers. Our interpretation of V values follows the guidelines given by Cohen (1988).

We begin by assessing whether and to what extent header information determines the content of a tablet. Let  $\mathbf{H}_n$  be a categorical variable denoting the name of the  $n$ th sign in a tablet, and let `topic` denote the topic with which a tablet is most strongly associated according to the topic model introduced in Section 4.4. Figure 7.3 depicts Cramér’s V between all of  $\mathbf{H}_n$  and `topic` for  $1 \leq n \leq 5$ .

`topic` has a strong to moderate association with all of the  $\mathbf{H}_n$  features; its strongest relation is  $V = 0.39$  with  $\mathbf{H}_1$ , implying that the first sign of a tablet strongly predicts the genre of the following text. V drops monotonically for later signs, implying that genre-defining information is primarily localized to the beginning of a text.

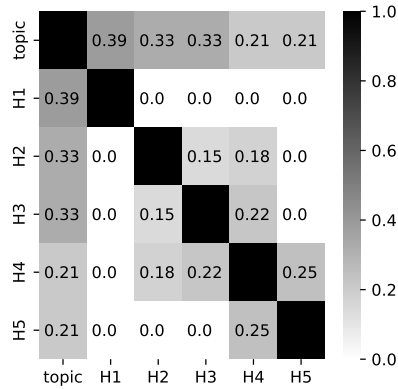


Figure 7.3: Heatmap showing the strength of the association (measured with Cramér’s  $V$ ) between the first five signs of a tablet and that tablet’s main topic according to an LDA model.

Most of the  $H_n$  features exhibit moderate to weak associations with one another, however there is no association at all ( $V = 0$ ) between  $H_1$  and  $H_n$  for  $n > 1$ . This would suggest that the first sign of a tablet is somehow disjoint from the rest of the text, and though this sign may predict the overall topic of the following text, it does *not* predict exactly which signs will immediately follow. This has implications for the interpretation of multi-sign headers, as it suggests that they may not comprise a unified whole (such as a two-sign-long word) so much as a concatenation of distinct signs with complementary roles.

To assess further, we introduce a variable `long_header` which is True for a text just in case the HMM proposes the existence of a multi-sign header in that text.  $H_1$  has no association ( $V = 0$ ) with `long_header`, meaning the first sign of a text does not predict whether the HMM will identify the presence of a multi-sign header. By contrast,  $H_2$  and  $H_3$  have a very strong association to `long_header` ( $V = 0.54$  and  $0.61$ ), and  $H_4$  is only slightly weaker ( $V = 0.43$ ).  $H_5$  also has no association. The lack of association with  $H_1$  further suggests that multi-sign headers are not variants or refinements of whatever sign occurs in the first position, and are rather concatenations of disjoint pieces of information.

Some texts bear one or more *seal* impressions, pressed or rolled against the clay in the manner of a signet ring; proto-Elamite seals depict objects and animals, and their presence is an extra-textual feature reflecting some aspect of ancient administrative practices. A *subscript* is a string of signs which occurs at the very end of some tablets, after the final numeral. Subscripts are unique in proto-Elamite in that they are not directly followed by a numeral, unlike all other spans of text.

We introduce a categorical variable representing whether a text has a seal (resp. subscript), and another representing *which* seal (resp. subscript) is present (each known seal is assigned a number in the transliterations, so that it can be identified when two texts share



the same impression).  $H_1$  does not determine whether a text is sealed ( $V = 0$ ); however, it does predict *which* particular seal was used ( $V = 0.39$ ). Intriguingly,  $H_2$  shows the opposite pattern, and weakly determines whether a text is sealed ( $V = 0.14$ ) but *not* which seal was used ( $V = 0$ ). A similar pattern holds for subscripts, where  $H_2$  predicts the presence of a subscript ( $V = 0.27$ ) and  $H_1$  does not ( $V = 0$ ), though in this case both  $H_1$  and  $H_2$  predict the text of the subscript ( $V = 0.30, 0.33$  resp.).

$H_1$  is strongly predictive of a text’s provenience ( $V = 0.56$ ), which could support theories that headers relate to activities undertaken at particular locales. Given that  $H_1$  also correlates with seal impressions, it is possible that the first sign of a tablet may convey information about where the tablet was sealed (and thus, likely, where it was written).

In sum, we have seen that the first sign of a tablet predicts extra-textual information such as provenience and choice of seal impression, but fails to predict textual information such as the signs that occur near to itself or the presence of a subscript. By contrast, the second sign of a tablet predicts textual content such as adjacent signs and the presence and content of a subscript, as well as some extra-textual content such as the presence of a seal. The first sign thus appears to look “outward” at the administrative context surrounding a text, whereas the second looks “inward” at the text itself.

In Section 4.4, we struggled to interpret one of the topics from our LDA model, which seemed to exhibit a large number of tablets with a header, seal impression, or both, and we speculated that the LDA model may have picked up on some latent connection between these features which was unknown to experts. The results in this section show that those speculations were accurate, and that there are in fact measurable associations between these features which had previously gone unnoticed.

### 7.3.4 Compositionality in Header Signs

Complex graphemes are common near the beginning of tablets, and many of the human-annotated headers are complex graphemes themselves or participate in the construction of complex graphemes in other contexts.

We have shown in Chapter 6 and Born et al. 2021 that, in various embedding models, certain complex graphemes tend to receive additively compositional embeddings which are close to the sum of the embeddings of the signs used in their construction. Similar patterns have been observed (Mikolov et al. 2013b; Salehi et al. 2015; Cordeiro et al. 2016) in modern languages where phrasal representations are often close to the sum of their parts, but only when the phrase is semantically compositional. Embeddings for idiomatic phrases are less likely to receive compositional embeddings.

We hypothesize that there may be some relation between a complex grapheme’s degree of compositionality and its tendency to occur in headers (for instance, signs representing institutions could be essentially heraldic, like Japanese *mon*, with no breakdown into simpler semantic units). To test this, for every complex grapheme  $A+B$ , we measure the cosine

similarity between the embedding for A+B and the sum of the embeddings for A and B using the embeddings from our best performing model in Chapter 6. Table 7.4 shows the average similarity for complex graphemes occurring in headers (as identified by any of our three approaches), versus complex graphemes in non-initial position.<sup>6</sup> We perform the averaging both over tokens (so that a complex grapheme occurring at the beginning of multiple tablets is included in the average multiple times) and over types (so that each type of complex grapheme is included in the average at most once).

	Tablet-Initial CGs			Non-Initial CGs
	Expert	HMM	LR	
Avg. cos over tokens	<b>0.682</b>	<b>0.693</b>	<b>0.683</b>	0.565
Avg. cos over types	0.608	0.648	0.616	0.593

Table 7.4: Mean compositionality of complex graphemes found in expert-annotated headers (Expert), in headers identified using HMM state 7 (HMM), in headers predicted by logistic regression over Transformer self-attention (LR), and in non-initial positions. Bolded values differ significantly from the rightmost column.

Complex graphemes occurring in headers (according to any of the three possible labelings) are on average more compositional than those occurring in the body of a text. The difference is not significant when averaged over types, but is highly significant when averaged over tokens ( $p \ll 0.01$ , Mann-Whitney U). This likely reflects the fact that (i) the more frequent a CG is in tablet-initial position, the more compositional it is,<sup>7</sup> and (ii) there are many fewer types than tokens, so those samples are too small to show significance.

Mean compositionality is lower for signs drawn from the expert annotations than for the other approaches, but the difference is not significant. This difference is mainly a consequence of the broken and fragmentary tablets where our models have failed to identify a header that is present in the human annotations. Many of these broken tablets begin with a complex grapheme, and many of these graphemes are non-compositional, possibly because they occur in short and fragmented contexts and therefore receive poor quality representations.

The apparent overlap between headers and more compositional complex graphemes on the one hand, and non-headers and less compositional graphemes on the other, increases our confidence that complex graphemes can be partitioned into measurably distinct groups, and should not necessarily be conceived of or analyzed as a monolithic category.

<sup>6</sup>Since the Transformer does not identify an explicit boundary to the header, we only count a complex grapheme as being part of the header when it is the first sign of the tablet. If long headers really exist, it is possible that some complex graphemes which are not the first sign of the tablet should still be counted as part of a header.

<sup>7</sup>There is no significant correlation between sign frequency and compositionality in general; this trend only (weakly) applies to tablet-initial signs.

## 7.4 Related Work

HMMs have a storied pedigree in the field of decipherment, being first used (under codename PTAH) by members of the NSA to analyze the Voynich manuscript (D’Imperio 1979). As this work was originally classified, most HMM-based approaches to decipherment instead trace back to Knight et al. (2006) who demonstrate the effectiveness of HMMs on a range of unsupervised decipherment tasks, and whose framework is adopted or used as a baseline in a significant volume of later work (Ravi and Knight 2009; Snyder et al. 2010; Knight et al. 2011; Reddy and Knight 2011; Berg-Kirkpatrick and Klein 2013; Kim and Snyder 2013 *inter alios*). We are not aware of any work which has employed Transformers as feature extractors for a comparable unsupervised analysis of undeciphered text, though applications to supervised decipherment include Aldarrab and May 2021 and Kambhatla et al. 2023.

## 7.5 Conclusion

This chapter offers the most exhaustive assessment of proto-Elamite headers to date in an attempt to inform the ongoing decipherment of this ancient script.

We have demonstrated that two distinct unsupervised sequence modeling techniques exhibit unique behaviours at the beginning of some proto-Elamite texts. These behaviours are consistent with, and offer independent evidence in support of, the prevailing hypothesis that these documents begin with a header.

The features recovered by these models predict with up to 95% accuracy whether experts understand a text to contain a header. This inspires confidence that the expert labels have been applied according to a consistent logic and following salient structural features of the texts. Our error analysis has also allowed us to identify and emend 25 mistakes in the expert annotations, expanding the total number of headers in the corpus by nearly 4% and reducing the amount of noise in a low-resource dataset where small errors may have an outsize effect.

We have demonstrated that there are measurable differences between the contextual embeddings learned for complex graphemes labeled as headers versus those in other contexts, reaffirming that these signs are somehow functionally distinct from other complex signs and from the rest of the script.

Using self-attention scores from a Transformer language model, we have demonstrated that the *second* sign of a text predicts the presence of a header more accurately than the first sign; we have also shown that state sequences from an HMM suggest that many more multi-sign headers exist than were previously assumed. On the basis of these results we have argued against the conventional understanding that header information is localized to a single sign, and suggest that headers may commonly span two or even three signs in some texts.

Finally, we have identified correlations between sign usage at the beginning of a text and other features such as genre, seal impressions, and the presence of a subscript. These correlations suggest that the first sign of a text captures more extra-textual information than later signs, and that if multi-sign headers exist, their two (or more) constituent signs likely convey distinct kinds of information.

## Chapter 8

# Numerals

As alluded to in Part I of this work, proto-Elamite employs multiple distinct number systems, which use partially-overlapping sets of digits that occasionally assign distinct values to identical-looking sign shapes (Figure 8.1). Many numerals can be read according to two or more of these systems, and may represent units of measure or have different absolute values depending on the system used.

Based on prior work (Dahl 2005a; Kelley 2018), there appear to be some regular relationships between the kind of object recorded in an entry and the number system used to count it. This is not entirely dissimilar to the way measure words are used in East Asian languages, where for example Japanese qualifies counts of small animals with 匹, flat objects with 枚, people with 人, etc, and the pronunciation of the associated numeral can vary with the counter that is used (e.g. six is broadly /roku/ in 六枚 “six pages” but /mu/ in 六つ “six things”). In proto-Elamite, knowing which system is in use for a given numeral increases the possibility of understanding what category of object is recorded in the adjoining text, and thus opens new avenues for decipherment.

In this chapter, we consider the task of disambiguating which systems are used in ambiguous proto-Elamite numeral notations in order that the values of these numerals may be determined. We describe an automated conversion from PE notation to modern Hindu-Arabic notation, which allows us to give the first large-scale survey of PE numerals since Friberg 1978 (whose manual analysis occurred at a time when fewer texts were known). We then propose two disambiguation techniques, one based on the subset-sum problem and another which uses a bootstrap classifier (Yarowsky 1995). We describe the construction of a test set for evaluating PE numeral disambiguation models, and propose a novel approach to cautious rule selection which significantly improves the performance of a bootstrap classifier on our data. Our analysis shows how these techniques lead to a deeper understanding of many signs and texts, and reveals transliterations in need of correction to produce a cleaner dataset for future work. This chapter reproduces and expands on results which were originally published in Born et al. 2023a.

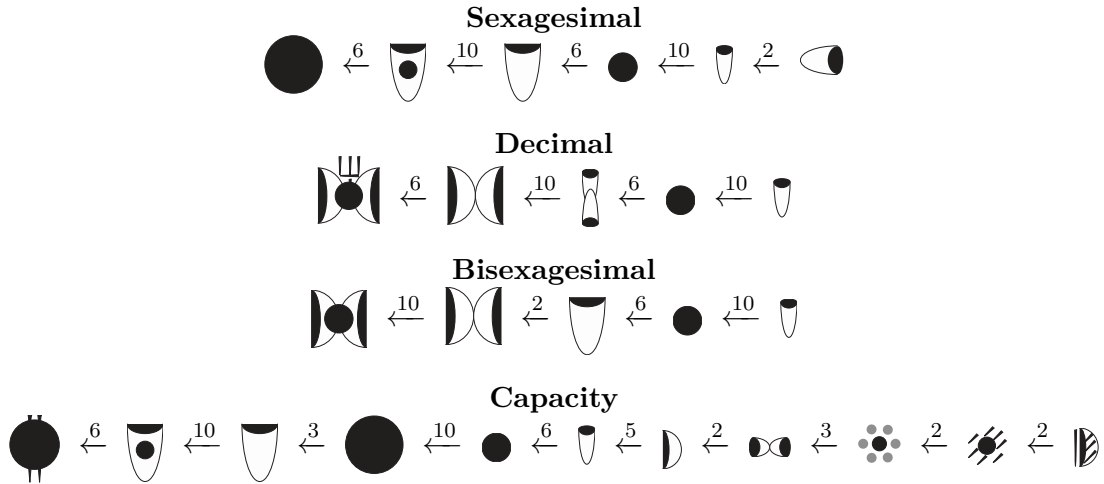


Figure 8.1: Relative values of digits in the main proto-Elamite number systems.  $X \stackrel{n}{\leftarrow} Y$  means that one  $X$  has the same value as  $n$   $Y$ s. Adapted from Englund (2004)

## 8.1 Background

Most proto-Elamite numerals are written using one of four<sup>1</sup> number systems, which are called *decimal* (D), *sexagesimal* (S), *bisexagesimal* (B), and *capacity* (C). In spite of their names, all of these systems use mixed radices. Figure 8.1 shows the relative values of the digits in each of these systems, as derived through prior manual analyses (Friberg 1978; Damerow and Englund 1989; Dahl 2019). Unlike Hindu-Arabic notation, where the value of a digit depends on its position, the values of proto-Elamite digits are fixed, and larger values are denoted by repeating a digit multiple times.


Note that some digits can be used with distinct values in multiple systems (e.g. N14  $\bullet$  equals  $10 \times N01$   $\cup$  in S, but only  $6 \times N01$   $\cup$  in C): this means that it can sometimes be impossible to determine the absolute value of a numeral unless context makes clear which system it employs.

Since  $N01$   $\cup$  occurs as part of every number system, we use this sign as a standard unit and report values as the equivalent multiple of  $N01$  whenever we convert to Arabic notation. Note, however, that the S, D, and B systems are understood to represent unitless cardinal numbers, while C apparently records unitful measures of volume. When it is necessary to emphasize that these systems are not commensurable, we add a superscript to denote the system in use: e.g.  $12 \times N01^C$  is a measure of volume which is not equivalent to  $12 \times N01^D$ , despite having identical magnitude.

<sup>1</sup>Additionally, there are marginal systems (labeled B#, C#, and C'') which appear to be derived from one of the four main systems by the addition of hatch marks or boxes drawn around the digits. These systems are rare, and the extra hatching or boxing makes them trivial to identify, so we ignore them for the remainder of this chapter.

We summarize the above points with an illustrative example. The following notation (read from right-to-left)



would be transliterated as 1(N45) 2(N14) 7(N01). This numeral must use either the S or C system, as the large circle N45  only occurs in these systems (Figure 8.1). Using the readings from the S system, this notation encodes a value of

$$\begin{aligned} 3627 \times \text{N01}^S &= 1 \times \text{N45} && +2 \times \text{N14} && +7 \times \text{N01} \\ &= 1 \times 3600 \times \text{N01}^S && +2 \times 10 \times \text{N01}^S && +7 \times 1 \times \text{N01}^S \end{aligned}$$

Using the C system, it instead encodes

$$\begin{aligned} 79 \times \text{N01}^C &= 1 \times \text{N45} && +2 \times \text{N14} && +7 \times \text{N01} \\ &= 1 \times 60 \times \text{N01}^C && +2 \times 6 \times \text{N01}^C && +7 \times 1 \times \text{N01}^C \end{aligned}$$

Of these, we know that the S reading must be the correct one, since 7(N01) should never occur in the C system (every 6 N01 would be bundled into an N14, so the actual notation for  $79 \times \text{N01}^C$  would be 1(N45) 3(N14) 1(N01)).

A numeral has an *ambiguous system* when it admits a valid reading according to more than one number system, and we say that it is *X-ambiguous* (for some  $X \in 2^{\{S,D,B,C\}}$ ) if it can be read using any of the systems in  $X$ . For example, 1(N14) 1(N01) is SDBC-ambiguous, 1(N34) is SBC-ambiguous, and 3(N34) is SC-ambiguous.

A numeral has an *ambiguous value* if it equals a different multiple of N01 when read in different systems. For example, 1(N14) 1(N01) may equal either  $7 \times \text{N01}^C$  or  $11 \times \text{N01}^D$ . A numeral can have an ambiguous system without having an ambiguous value: for instance,  $2(\text{N01}) = 2 \times \text{N01}^S = 2 \times \text{N01}^D = 2 \times \text{N01}^B = 2 \times \text{N01}^C$ .

## 8.2 Automated Conversion

We extract all of the numeral notations from the transliterated corpus by using a regular expression to find every contiguous sequence of N-signs; we discard sequences which are damaged, which we identify as being immediately adjacent to a transliterated X or . . . . We also discard sequences containing the N sign (which denotes a broken character which must be a digit based on the context where it occurs, but it is not known *which* digit) and those where the number of digits is unknown (where the transliteration records n(N01) instead of a concrete count like 5(N01)). Algorithm 1 uses the relative values from Figure 8.1 to automatically convert each of these numerals to a dictionary of possible readings in modern Arabic notation.

---

**Algorithm 1** PE-to-Arabic Conversion

---

**Input:**  $digits = [(n_1, sign_1), \dots, (n_k, sign_k)]$   $\triangleright$  List of (# times digit occurs, digit name).

**Returns:** A map from number systems to possible readings for this digit list.

---

**for**  $sys \in \{S, D, B, C\}$  **do**

$value_{sys} \leftarrow 0$

**for**  $(n, sign) \in digits$  **do**

**for**  $sys \in \{S, D, B, C\}$  **do**

**if**  $sign \notin signs\_used\_by(sys)$  **then**

$value_{sys} \leftarrow \perp$

$\triangleright \perp$  means there is no valid reading in this system.

**continue**

**if**  $n > max\_count(sign, sys)$  **then**

$\triangleright max\_count$  returns the max num. of times this digit can occur before it would carry over to a higher value digit.

$value_{sys} \leftarrow \perp$

$v \leftarrow$  value of  $sign$  in  $sys$

$value_{sys} \leftarrow value_{sys} + n \times v$

$\triangleright \perp$  plus anything equals  $\perp$

**return**  $\{sys \mapsto value_{sys} \mid sys \in \{S, D, B, C\}\}$

---

Of the 8011 intact numerals which we have extracted, there are 7954 for which this conversion returns at least one reading. Of these, only 1919 unambiguously belong to a particular number system: the remainder are ambiguous between two, three, or even all four systems (Table 8.1). The following sections outline two proposals for disambiguating the ambiguous cases. Section 8.4.1 discusses the 57 cases for which there is no valid reading in any system.

## 8.3 Disambiguation

### 8.3.1 Tablet Summaries

Our first approach to disambiguating these notations relies on the fact that some PE documents end in a summary line, which records the total sum of the preceding entries. Although the entries themselves may be ambiguous, the sums naturally record larger amounts and are therefore more likely to use high-magnitude digits that unambiguously belong to a particular system. When a tablet records values from multiple number systems, they appear to be summarized separately; thus if we can identify unambiguous summaries, we can infer that all of the entries which they sum must belong to the same system.



Possible Readings	Number of Numerals
none	57
B	19
C	1727
D	66
S	107
B or D	21
B or S	5
C or S	143
B, C, or S	185
B, D, or S	292
B, C, D, or S	5389

Table 8.1: Distribution of readings produced by our automated conversion. A majority of numerals in the corpus can be read using *any one* of the four number systems.

To achieve this, we filter the corpus to find texts with one or two entries on the reverse, as current understandings of the corpus suggest that these are likely to be summaries.<sup>2</sup> For each of these texts, we solve an instance of the subset-sum problem to identify whether any combination of readings from the obverse adds up to the same value as any reading of the reverse.

If an accurate summation is found, and any of the component terms has an unambiguous number system, we use this as evidence to disambiguate the entire text to that system. We manually evaluate this approach by confirming with domain experts whether the resulting disambiguations are correct.

The subset-sum problem is NP-hard, so for long texts it is necessary to increase the efficiency of the search by merging readings which share the same magnitude. For example, if a numeral can be read as  $10 \times \text{N01}^S$ ,  $10 \times \text{N01}^D$ ,  $10 \times \text{N01}^B$ , or  $6 \times \text{N01}^C$ , we collapse these into two readings  $10 \times \text{N01}^{SDB}$  and  $6 \times \text{N01}^C$ . We attempt to solve the subset-sum problem using these merged readings; every solution to the merged subset-sum problem corresponds to a family of solutions to the original problem, which can be recovered by un-merging the combined readings.

### 8.3.2 Bootstrapping

Some of the PE numeral notations are inherently unambiguous, either because they use a digit which only occurs in a single system, or because they contain more instances of a digit

<sup>2</sup>Some transliterations include an annotation which explicitly labels a particular entry as a summary. However, not all summaries are labeled in this way, so we rely on automatic detection of summaries to expand the number of texts available for this analysis.

than would be allowed by some systems. We propose to use these cases as seed rules to train a bootstrap classifier (Yarowsky 1995) for disambiguation.

We choose bootstrapping because it requires only a small number of seed labels, and as seen in Table 8.1, some systems have few unambiguous attestations. Moreover, bootstrapping yields interpretable results which can be understood by examining the label distribution associated with each input feature. This helps to legitimize model outputs to domain experts, and to situate model predictions relative to prior manual analyses.

Feature	Description of Value(s)
TABLET	The tablet where this numeral occurs.
FIRST_SIGN	The first sign of the tablet where this numeral occurs (where we may expect to find a header).
SAME_ENTRY	Bag of signs which occur in the entry preceding this numeral.
SAME_TABLET	Bag of signs which occur anywhere on the same tablet as this numeral.
OBJECT	The sign immediately preceding this numeral (where we may expect to find a counted object).
IMPLICIT_OBJECT	The last sign in the first entry of the text where this numeral occurs (where we may expect to find an implicit object).

Table 8.2: Each numeral is associated with a set of features from this list, which we use to train our bootstrap classifiers.

Table 8.2 lists the features used by our classifier. Each numeral is associated with a set of these features which together describe the context where it occurs. A numeral’s initial label distribution is uniform over every system for which our automated conversion returns a valid reading, and zero elsewhere. We train this bootstrap classifier using the DL-2-ML algorithm (Haffari and Sarkar 2007; Abney 2002), which models  $\pi_x(j)$  (the likelihood that sample  $x$  belongs to class  $j$ ) as

$$\pi_x(j) \propto \prod_{f \in F_x} \theta_{fj}$$

where  $F_x$  is the set of features associated with sample  $x$ , and  $\theta_{fj}$  is a learnable parameter which measures the association between feature  $f$  and class  $j$ . We apply the “cautious” approach from Collins and Singer 1999, which limits the number of rules that can be added to the decision list at each iteration of training. Specifically, candidate rules are sorted according to the number of labeled examples that support them, and only the  $n$  with the largest support are added to the decision list.<sup>3</sup>  $n$  starts at 5 and increases by 5 each iteration.

The cautious algorithm was motivated by the observation that “the highest frequency rules [are] much ‘safer’ [than low-frequency rules], as they tend to be very accurate” (Collins

<sup>3</sup>Whitney and Sarkar 2012 note that many details are omitted from the description of the cautious algorithm in Collins and Singer 1999. We follow Whitney and Sarkar in assuming that confidence thresholding is performed using unsmoothed label counts. However, we differ from their approach by selecting the top  $n$  rules overall (not the top  $n$  for each label) as this yields stronger results on our data.

and Singer 1999). This observation does not seem to hold for our data, where many of the most frequent features are barely predictive of any class, and impressionistically, do not appear to be any more accurate than those with lower frequency. We therefore propose a novel approach to cautious rule selection whereby  $\theta_f$  is only updated if the update increases  $\max_j \theta_{fj}$ , i.e. if it increases the confidence of the label distribution associated with feature  $f$ . In this setting we visit the features in a random order each iteration, to prevent a degenerate outcome whereby endless incremental updates are made only to the first rules visited.

<b>System</b>	<b>Number of Test Items</b>
B	3
C	18
D	14
S	13

Table 8.3: Distribution of target classes in our numeral disambiguation test set. This set contains every instance of the B class which we were able to manually disambiguate with the help of domain experts; the other classes are kept small to maintain as balanced a distribution as possible.

Prior to training, we upsample seeds with rare labels to obtain an equal number for each class. We evaluate our classifiers on a test set which we construct by manually disambiguating some of the ambiguous notations in the corpus. We endeavoured to keep this set as balanced as possible, but some systems (particularly B) can only be confidently identified in a few texts. This means the test set cannot contain every numeral for which we know the target label, as doing so would yield too great an imbalance between classes. Appendix C describes what evidence was used to disambiguate each numeral in the set, and Table 8.3 summarizes the overall class distribution. All of the labels in the test set have been verified by domain experts.

## 8.4 Results

### 8.4.1 Automated Conversion

#### Invalid Notations

There are 57 intact numerals for which our automated conversion does not return a valid reading according to any number system. The vast majority of these violate the bundling principles established in prior work and shown in Figure 8.1. For example, the notation 11(N01) in P008043 should not be attested in any system, as every system carries over to a higher digit after at most ten N01s.

Some of these illegal notations may reflect errors on the part of the original scribes. For example, in P008844, the sum of the D values on the obverse equals 9(N23) 3(N14)

Systems Used	Number of Tablets
C and S	12
C and D	15
C and B	4
S and D	1

Table 8.4: Number of tablets which unambiguously use more than one number system.

3(N01), or 573; the scribe has actually written 9(N23) 7(N14) 3(N01), which violates the usual principle that 6 N14s carry over to one N23. This suggests that the scribe may have conceived of the D system as a truly decimal notation (in which case the least significant digits would be written as 7 tens and 3 ones, exactly as we find on the tablet), forgetting that N23 uses a different radix. Several of the other aberrant notations lend credence to this view, such as P008788 which apparently records 88 M367s (likely goats, usually counted with D) as 8(N14) 8(N01). These cases suggest a lack of standardisation across scribes or across documents, which is consistent with the longstanding view that the writing system never achieved a significant degree of standardisation (Dahl 2019).

### Mixed Systems

Our automated conversion reveals numerous accounts (Table 8.4) which unambiguously use two different number systems (no tablets unambiguously use more than two systems). In all but one of these, the C system occurs alongside one of the “integer” systems S, D, or B. This suggests a general pattern of accounts which record capacities of goods received/disbursed from/to individual people, animals, households, or other entities counted in whole numbers. The text P009383 is unique in that it unambiguously uses two of the “integer” systems S and D. On close inspection, however, the original tablet is heavily abraded where the putative S notation occurs, and the sign which forces this notation to be read as S (N08) is almost entirely unreadable. Given the otherwise total absence of texts which mix integer systems, we posit that this text may contain a transliteration error and that the broken sign is not in fact an N08.

Several texts which mix the S and C systems also have other features in common. P008796, P008798, and P008805 are exemplary of this group, which are all two-entry texts where the first entry is an S-denominated amount of M056~f  $\frac{1}{2}$  (possibly a plow), and the second is a C-denominated amount of M288. In all of these texts, there are exactly  $2.5 \times N01^S$  per  $N01^C$ : this ratio was previously identified as likely to represent amounts of seed grain by Damerow and Englund (1989); Englund (2004). P008791 appears to belong to this same class of texts, and given that the first entry records  $128 \times N01^S$  M056~f we should expect  $51.2 \times N01^C$  in the final entry (or 8(N14) 3(N01) 1(N39B)). We actually find  $52 \times N01^C$ , or 8(N14) 4(N01). Close inspection of the tablet, however, reveals that

there has been a transliteration error, and that the final sign, although mostly broken, is recognizably an N39B. The text yields the expected ratio when this mistake is corrected. Errors such as these are much easier to identify when dealing in converted Arabic numerals, which modern readers can understand and manipulate more quickly and intuitively than the original proto-Elamite notations.

Out of 244 signs which precede at least two unambiguous notations, only 11 occur next to notations from distinct systems. These 11 (M001, M056~f, M059, M096, M124, M218, M305, M327, M371, M387, M388) include signs which have been speculated (Dahl 2019) to represent human laborers or overseers (M388, M124), signs with possible syllabic values (M001, M096, M218, M387, M371), and headers or account owners (M059, M305, M327). In other words, these are signs which we expect to qualify or describe an object being counted, and not to be counted themselves. It therefore appears that, while counted objects are consistently recorded using one particular number system, these qualifying signs can potentially be used to qualify objects from several different systems. This suggests a novel approach to distinguish qualifiers from object signs by looking at the variety of number systems a sign can occur beside.

#### 8.4.2 Subset-Sum Analysis

Our subset-sum analysis identifies 24 texts which, upon manual inspection, can be fully or partially disambiguated based on their summary line(s). We highlight one which is of particular interest. In P008014, all entries must use the C system in order to equal the same value as the unambiguous C summary on the reverse. The text of the summary contains only the “grain container” sign M288, implying that the entire tablet should record amounts of M288. However, on the obverse of the tablet, M288 only occurs as the final sign of the very first entry. This suggests that the scribe has only explicitly marked the counted object in the first entry, and left it implicit in the following entries (this is in keeping with known practices from other ancient Mesopotamian accounting corpora; Nissen et al. 1993: 37–38, Englund 2001). This means that there exist long-distance dependencies between the entries in some texts, which need to be accounted for if these texts are to be fully understood.

#### 8.4.3 Bootstrapping

Figure 8.2 and Table 8.5 compare results from our two approaches to bootstrapping. The baseline, vanilla bootstrapping algorithm achieves a mediocre 0.60 F1. Recall of the B and S classes is perfect, but only half of the instances of the C class are correctly labeled. This classifier does not seem to have uncovered any clear signals to identify the D system: 3 instances of this class remain unlabeled (their associated features simply predicted uniform distributions over all systems), and those which are labeled are distributed uniformly across all of the possible classes.

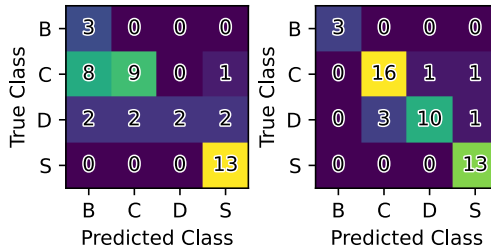


Figure 8.2: Confusion matrices from classifiers trained using the vanilla bootstrap algorithm (left) and our proposed variant (right).

Model	4-way			2-way		
	prec.	rec.	F1	prec.	rec.	F1
Baseline	0.64	0.56	0.60	0.74	0.65	0.69
Ours	<b>0.88</b>	<b>0.88</b>	<b>0.88</b>	<b>0.90</b>	<b>0.90</b>	<b>0.90</b>

Table 8.5: Numeral disambiguation results. In the 4-way setting, we seek to identify exactly which number system is in use for each numeral. In the 2-way setting, we only seek to distinguish C notations from everything else.

By contrast, our proposed approach to cautious rule selection yields 0.88 F1, with significantly better recall of both the C and D classes than the baseline. This suggests that frequency-based caution may be ill-suited for bootstrapping on some datasets, and that, in settings where bootstrap classifiers remain viable, it may be worthwhile to explore alternative approaches to cautious rule selection.

Note from Figure 8.1 that ambiguities between the S, D, and B systems primarily come from the digits N01  $\cup$  and N14  $\bullet$ , which have the same relative values across all three systems. S and B further overlap in the sign N34  $\cup$ , which also maintains the same value across both systems. Thus many numerals which are technically ambiguous between these *systems* nonetheless have unambiguous *values*.<sup>4</sup> In settings where the absolute value of a numeral is all that matters, it is therefore usually sufficient to distinguish these systems from C without distinguishing them from one another. In this two-way setting, our model achieves 0.90 F1, versus 0.69 F1 from the vanilla baseline (Figure 8.3).

We emphasize that our test set only contains numerals which domain experts were able to disambiguate based on manual inspection. Easy cases may therefore be over-represented, and we expect our results to be an upper bound on the classifier’s accuracy across the whole corpus. Despite this, the results are strong enough to suggest that a majority of the ambiguous numerals in the corpus can be disambiguated with some certainty.

<sup>4</sup>Here we assume that the *absolute* value of N01 is the same across all three systems, and not just its value relative to the other digits. Current understandings suggest that this is the case, but the undeciphered nature of the script means this is technically not certain.

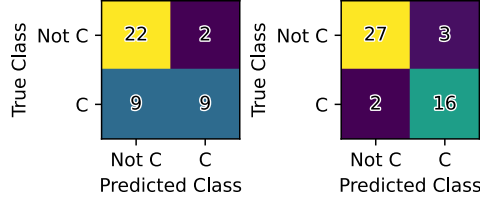


















Figure 8.3: 2-way disambiguation results. Confusion matrices from vanilla bootstrapping (left) and our proposed variant (right).



## 8.5 Analysis



In this section, we investigate how the features from our bootstrap classifier relate to known or hypothesized properties of the script.



A number of signs have been suggested to indicate types of livestock (sheep, goats, etc.); these include M346 , M362 , M367 , and M417  (Dahl 2005a; Kelley 2018). In our analysis, all of these signs also predict the decimal system, which suggests that they were typically used to denote flocks.

M376  has been suggested as either a “high-status human” (Dahl 2005a: 95) or livestock (Kelley 2018: 165); it is strongly predicative of the sexagesimal system in our analysis. Other objects associated with this system include M056~f , (a plow, which also patterned with M376 in Section 4.4); M219 , M371  and M296  (very speculatively, syllables used to write personal names); M059 , M145 , and M365  (“owners”, possibly persons or institutions to whom these accounts belonged); and M269~c  (a vessel?). This may cast some doubt on the livestock reading for M376, in favor of the high-status human reading, which is a more natural fit among the owner signs and possible personal names in this collection.

We note one text, P008212, which may exhibit a consistent ratio related to M376. P008212 alternates between entries ending in M288  and entries ending in M376  or M367~i . The magnitude of the numeral in an M288 entry is always exactly 4 times as large as the magnitude of an adjacent M376 entry, or 2 times as large as an adjacent M367~i entry. This pattern is very unlikely to be due to chance, as it holds across 44 total entries. Whatever the meaning of M376, on the basis of this text we can assert that it is associated with amounts of M288 that are exactly twice as large as those associated with M367~i. To our knowledge this ratio has not yet been noted in previous publications.

After disambiguating every numeral in the corpus to the most likely system according to our bootstrap classifier, we measure the average magnitude of counts associated with each feature. We observe that certain features accompany significantly higher or lower counts than others. Entries ending in M288  have the largest capacity magnitudes on average, while those ending in M263  are among the smallest. Both signs have been speculated to

represent containers; from our results one might further speculate that M263 is a container of smaller dimension, or one that was never dealt with in bulk quantities. Entries ending in M297~b  also accompany unusually large capacity measures, though the visually-similar M297  does not stand out as unusually large or small. We will see additional evidence in Section 8.8 that these signs exhibit distinct patterns of use despite sharing a visual resemblance.

The numerical systems of proto-Elamite have been proposed to have functional uses relating to cultural practices in 3rd millennium south-western Iran. For instance, the capacity system (C) is suggested to be used for counting rations disbursed to households or workers (Kelley 2018: 153–155). Among the recipients of these rations are M388  and M124 , parallel “worker categories” which may represent the heads of work teams. We find that entries beginning in M388 accompany significantly larger capacity measures on average than those beginning in M124, which may point towards M388 individuals heading teams of larger sizes or comprising workers of higher status.

## 8.6 Related Work

Naik et al. (2019) demonstrate that word embedding models fail to capture magnitude and numeration (i.e. the equivalence between *3* and *three*), and suggest the need for dedicated representations of numerals in NLP models. Sundararaman et al. (2020) follow up with DICE embeddings designed to explicitly capture both magnitude and numeration, and demonstrate improved results on numeracy tests introduced by Wallace et al. (2019). Spithourakis and Riedel (2018) describe a GMM-based approach to numeral embedding for language models, which also incorporates explicit representations of magnitude. These models assume that the magnitudes in question are known and must simply be encoded; they do not consider the task of *determining* magnitude from ambiguous notations.

While introducing a benchmark to test LM numeracy, Shi et al. (2022) note that numeral representations can vary across scripts; however, they assume a setting where the conversion to Arabic notation is straightforward, and do not discuss ambiguities which may result from this conversion.

One approach to handling numeric values in word problems is to replace them with variables  $v_1, v_2, \dots$ , generate the solution as an equation in terms of these variables, and substitute the original values back to obtain a concrete solution. Wu et al. (2021) note that the choice of equation can sometimes depend on whether the original quantities were absolute values or percentages, and therefore this replacement can introduce ambiguities which make some problems unsolvable. They introduce a magnitude-aware encoding for digit sequences, and describe a numerical properties prediction mechanism which estimates whether a numeral is an integer, fraction, percentage, etc. This mirrors our attempt to predict an underlying number system.



Berg-Kirkpatrick and Spokoyny (2020) investigate the task of predicting a numeric value given surrounding text as context. They find that models which implicitly separate a value’s mantissa from its exponent achieve better results than those which predict the value directly, and that context from large pretrained text encoders is useful even when the pretraining task was not focused on promoting numeracy. As we are dealing with an undeciphered corpus, our models are unfortunately unable to exploit large pretrained model embeddings for context.

Friberg (1978) is responsible for early analyses of proto-Elamite and proto-cuneiform, which helped to establish the relative values of the digits in these corpora and the existence of distinct numeration systems. Nissen et al. (1993) perform what is possibly the earliest computer-assisted analysis of bookkeeping practices in proto-cuneiform, while Damerow and Englund (1989) and Englund (2011) discuss accounting practices in proto-cuneiform and proto-Elamite and the relationships between the two.

## 8.7 Interim Conclusions

We have described an automated conversion from ancient proto-Elamite numeral notations to modern Hindu-Arabic notation, and have described ambiguities in the original script which make this conversion challenging. We have presented two approaches for disambiguating these ambiguous notations: one exploits a common structural property of proto-Elamite accounts to look for unambiguous summations, and the other exploits the few unambiguous notations to train a bootstrap classifier. We have created a test set for the disambiguation task by manually disambiguating a subset of the corpus, and have described a novel variant of cautious rule selection which significantly improves bootstrapping performance on this test set. As a result of this work, we are able to assign confident values to a majority of the numeral notations in proto-Elamite, to identify and correct a number of transliteration errors in the proto-Elamite corpus, and to shed new light on existing hypotheses about the meanings of some signs in this partially-deciphered script. As the proto-Elamite script was fundamentally an accounting technology, we believe that this work represents a crucial step towards deepening our understanding of this ancient corpus.

## 8.8 Parallel Coordinates Visualizations

Parallel coordinate plots (Inselberg 1985) offer a way to simultaneously visualize all possible readings of an ambiguous numeral. In these plots, each vertical axis represents one of the number systems, and each colored line represents one numeral. The line for a numeral touches each axis at a point corresponding to its value in the corresponding system; if a numeral cannot be read in a given system, it will not touch that axis at all. A numeral with an unambiguous system will therefore be represented by a single point on one axis; numerals

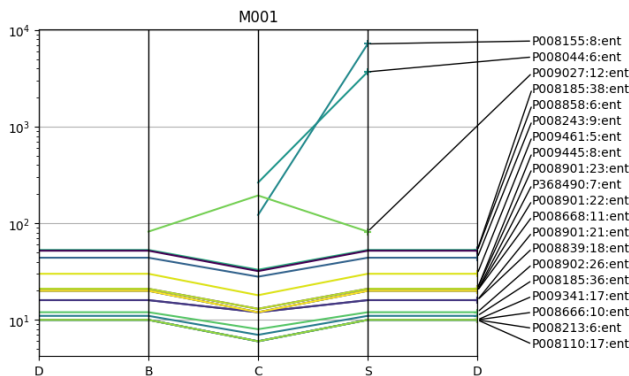


Figure 8.4: Parallel coordinate plots showing all possible readings of every numeral associated with the object sign M001  $\uparrow$ .

with ambiguous systems will appear as line segments spanning multiple axes. Lines which are parallel to the x-axis represent numerals with unambiguous values.

From these plots, we are able to ascertain qualitative differences between the distributions of some objects even without knowing which system(s) they are counted in. For example, when we plot the ambiguous numerals which occur next to M001  $\uparrow$  (Figure 8.4), we observe that three of them (P008155:8, P008044:6, and P009027:12) appear to follow a unique distribution compared to the others. These notations *must* encode larger values than usual, regardless of how they are disambiguated.

This visualization also gives a quick and intuitive way to identify whether a sign is associated with any one number system in particular, and to identify outlying notations which do not belong to this habitual system. For example, in the plots for M296  $\uparrow$ , M297  $\uparrow$ , and M297~b  $\uparrow$  (Figure 8.5), we find that a significant proportion of the counts associated with these signs unambiguously use the C system, being represented by single points on the central C axis. However, in all three we also observe notations which *cannot* be read with the C system, as their lines do not touch the C axis. This suggests that all three signs, which share a visual resemblance, may also share two distinct modes of use, beside capacity and non-capacity measures respectively. Recall from Section 4.4 that the relationship between M297 and M297~b remains unclear, and our interpretation of the LDA topic model suggested that these signs may have polysemous uses: this polysemy appears to be on clear display in the two modes attested by these figures.

Parallel coordinate visualizations therefore offer a straightforward way to identify the broad trends in how a particular object is counted, and to identify nonstandard uses that warrant close attention from domain experts. These plots provide a holistic view of the numeral distributions, as opposed to the point sample predicted by the bootstrap model, which makes them a useful point of comparison as we continue to qualitatively evaluate

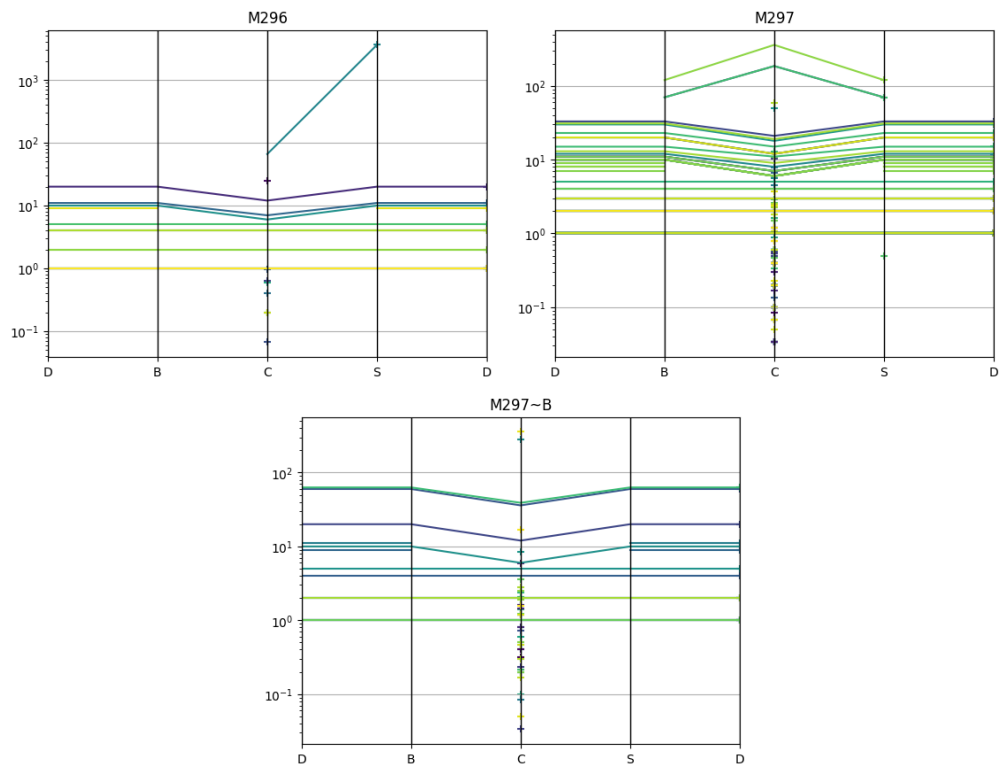


Figure 8.5: Parallel coordinate plot showing all possible readings of every numeral associated with the signs M296  $\diamond$ , M297  $\boxminus$ , and M297~b  $\boxplus$ .

the model predictions in collaboration with domain experts. The interested reader can find similar plots for other common signs in Appendix A.8.

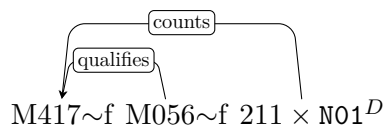
## 8.9 Implications to Polysemy and Word Order

Consider the text P008808, which begins with the entry

$$M417\sim f \ M056\sim f \ 3(N23) \ 3(N14) \ 1(N01) = 211 \times N01^D$$

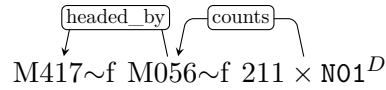
$M056\sim f$  ~~is~~ is known to be counted sexagesimally in the genre of seed texts mentioned in Section 8.4.1, where it refers to a plow, and this sign can also occur next to unambiguous capacity measures. The above text is unique in that it is the only place where  $M056\sim f$  unambiguously occurs beside the *decimal* system.

$M417\sim f$  (and other variants of M417) appears to depict an equid from the neck up, implying that this sign represents a live animal. The use of decimal notations to count animals and low-status humans has been well-established (Dahl 2019; Englund 2004), and in other texts (P008893) the variant  $M417\sim h$  is unambiguously counted with the decimal system. Given the otherwise total absence of  $M056\sim f$  next to unambiguous decimal notations, the most parsimonious reading of this text seems to be that  $M417\sim f$  is the decimally-counted object, and  $M056\sim f$  is a qualifier which clarifies that the animal was used for plowing. We represent this reading schematically using the following dependency tree:



Note, however, that this reading violates two conventions that have been established in prior work. First, experts have annotated the  $M417\sim f$  in this text as being a header, and both of our models from Chapter 7 agree with this labeling. Headers are generally taken to represent a distinct class from counted objects, and it would be unprecedented for a single token to be simultaneously both a counted object *and* a header. Second, counted objects are generally taken to occur at the very end of entries, as the final sign before a numeral. Signs which qualify the object, such as those identifying a possible owner, are taken to occur *before* the counted object. Reading  $M417\sim f$  as the counted object would violate both of these understandings.

Therefore, the more conservative reading is that  $M056\sim f$  is the counted object, in which case we must accept that this sign can be counted decimally in addition to its use with the other number systems. The following dependency tree clarifies how the underlying structure of the text would differ according to this reading:



In this case, M056~f would presumably stand for the animal used to pull a plow, and not the plow itself, in the same way that tools stand in for human laborers in other contexts (Dahl 2019: 79). By extension, this would mean that M056~f must be polysemous, as its use with the sexagesimal system in other texts implies that it was there used to refer to higher-status animals or laborers, or to inanimate plows, or to areas of land to be plowed (implied by the use of fractional counts in some texts, which presumably did not denote fractional people or animals). This is one of the most concrete examples of polysemy in the script, which cannot be easily explained as a possible example of an implied object (as implied objects are usually explicitly written in the first entry, and only implied in later parts of a text).

Both possible readings of this text are interesting, and have implications either to established notions of “word” order or to the existence of distinct “senses” for some signs. This case study highlights how even apparently simple contributions, like extracting the set of unambiguous numerals associated with a given sign, can lead to deeper understandings of signs which would otherwise require exhaustive manual effort to fully document.

## Chapter 9

# Signlist Revisions

A crucial step in deciphering a text is to identify what set of characters were used to write it. This requires grouping character tokens according to visual and contextual features, which can be challenging for human analysts when the number of tokens or underlying types is large. Prior work has shown that this process can be automated by clustering dense representations of character images, in a task which we call “script clustering”. In this chapter, we present novel architectures which exploit varying degrees of contextual and visual information to learn representations for use in script clustering. We evaluate on a range of modern and ancient scripts, and find that our models produce representations which are more effective for script recovery than the current state-of-the-art, despite using just 2% as many parameters. Our analysis fruitfully applies these models to assess hypotheses about the character inventory of the partially-deciphered proto-Elamite script. This chapter reproduces results which were originally published in Born et al. 2023b

### 9.1 Introduction

One of the first tasks in decipherment is to solve an instance of the token-to-type problem by recognizing which tokens represent the same underlying character, and thereby to construct a list of every character used in the script (cf. Gelb and Whiting 1975). Accurate character inventories are important for decipherment, as they indicate patterns of frequency and adjacency which can reveal information about the underlying message. However, it can be challenging for human annotators to determine which characters are truly distinct: tokens with different appearances can represent the same underlying character (such as English  $\text{p}$  and  $\text{P}$ ), while visually-similar tokens can represent distinct characters (such as  $\text{p}$  and  $\text{r}$  or  $\beta$  and  $\text{B}$ ).

This work introduces novel, VAE-based techniques for learning the character inventory of an unknown script by clustering images of character tokens. We show, through a range of experiments on deciphered and undeciphered scripts from modern and ancient corpora, how the complexity and number of characters in a script impact our models’ ability to learn

the underlying character inventory. On the ancient Cypro-Greek syllabary, our models outperform the recent Sign2Vec architecture (Corazza et al. 2022a) despite using just  $\sim 2\%$  as many parameters. We also apply these models to proto-Elamite, and find that they independently replicate expert intuitions about the underlying character inventory and suggest new relationships between signs which have not yet been noted in prior work.

## 9.2 Methodology

Corazza et al. 2022a and Corazza et al. 2022b outline a two-step procedure for learning the character inventory of an unknown script by clustering images of character tokens. They first train an unsupervised neural encoder to learn vector representations for images of the characters in question. After training, they cluster these representations: the resulting clusters serve as an estimate for the script’s underlying character inventory. The authors demonstrate good performance on the ancient Cypro-Greek script, and fruitfully apply this technique to the study of a related, undeciphered script called Cypro-Minoan.

Our work follows the same overall approach, and investigates how changes to the encoder architecture, data quality, and training process can affect the final clustering.

### 9.2.1 Motivation

Sign2Vec (Corazza et al. 2022a) uses the ResNet18 encoder (He et al. 2016), which is an 18-layer convolutional stack with residual connections. ResNet was originally developed for object detection and segmentation on the ImageNet (Deng et al. 2009) and COCO (Lin et al. 2014) datasets, which include photorealistic depictions of complex scenes. In this setting, very deep networks are necessary to capture the full range of visual information present in the input images (He et al. 2016). By contrast, images of written characters tend to be visually simple: they often read clearly in greyscale or black-and-white, and can generally be broken down into simple lines, curves, or wedges against a uniform background. In light of this, we hypothesize that the ResNet encoder may be significantly over-parameterized for the task of script clustering. This may lead to longer training times than necessary, more expensive compute costs, and reduced accessibility to experts outside of computer science who may lack access to hardware for training.

Additionally, one of the tasks used to train the Sign2Vec encoder is a masked prediction task, where information about a character must be recovered given the representations of the characters to its immediate left and right. This provides the model with a very narrow context window, which is sufficient for the experiments in the original work (Corazza et al. 2022a), but which we hypothesize may hamper the model’s performance in settings where wider context is available.

### 9.2.2 Model Architectures

In light of these limitations, we propose to compare four architectures which reduce the size of the encoder relative to Sign2Vec and incorporate varying degrees of context.

**VAE** All of our models are built around a variational autoencoder (VAE; Kingma and Welling 2014) with a convolutional encoder and a deconvolutional decoder (Figure 9.1). This architecture uses three stacked convolutional layers to learn vector representations  $\mu, \sigma \in \mathbb{R}^d$  from an input image  $\mathbf{x} \in \mathbb{R}^{n \times n}$ . These are used to sample a “code”  $\mathbf{z} \sim \mathcal{N}(\mu, \sigma)$ . A stack of transposed convolutional layers decodes  $\mathbf{z}$  to an image  $\tilde{\mathbf{x}} \in \mathbb{R}^{n \times n}$ . This model is trained to minimize the reconstruction error of  $\tilde{\mathbf{x}}$  with respect to the input  $\mathbf{x}$ :

$$\mathcal{L} = \text{BCE}(\tilde{x}, x) \tag{9.1}$$

where BCE is binary cross-entropy.

We use an autoencoder as a way to avoid potential label bias in this work. The working sign names implicitly encode information about the relationships that experts see between tokens: this is reflected by shared “subwords” in the working sign names, which can reflect broad visual similarities (as between M218  $\diamond$  and M219  $\diamond$ ) or a part-whole relationship (as between M218  $\diamond$  and M218+M101  $\diamond$ ). Using an autoencoder allows for fully open input and output vocabularies, which minimizes the likelihood of these inferred relationships biasing the model to reproduce the same divisions as experts.

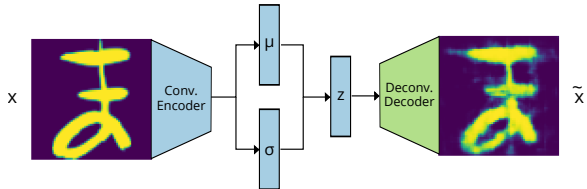


Figure 9.1: VAE architecture. This model reconstructs its input from a dense vector encoding.

**VAE+Neighbors** Our second model adds an auxiliary masked prediction task (Figure 9.2). Let  $\mathbf{z}_{i-1}$  and  $\mathbf{z}_{i+1}$  be the encodings of the images to the direct left and right of a token  $\mathbf{x}_i$ . We learn a projection  $M \in \mathbb{R}^{2d \times d}$  and decode  $M(\mathbf{z}_{i-1} \oplus \mathbf{z}_{i+1})$  to produce an image  $\tilde{\mathbf{x}}'_i$ . We add a new loss term to Equation (9.1) to minimize the reconstruction error of  $\tilde{\mathbf{x}}'_i$  with respect to  $\mathbf{x}_i$ :

$$\mathcal{L}_{\text{Neighbor}} = \text{BCE}(\tilde{\mathbf{x}}'_i, \mathbf{x}_i)$$

This is similar to the auxiliary task in Sign2Vec (Corazza et al. 2022a), with the difference that our model *draws* the masked sign, whereas Sign2Vec was trained to predict a property



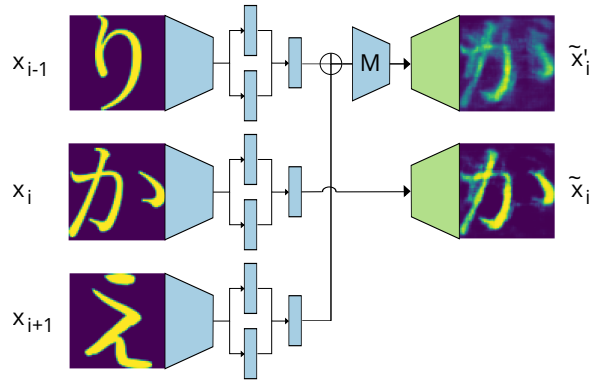


Figure 9.2: VAE+Neighbor architecture. This model adds the auxiliary task of reconstructing a character image given the encodings of the adjacent characters.

called its “pseudolabel” (see Section 9.2.3). By drawing the output instead of predicting a sign name from a fixed list, this model manages to fully divest itself from pre-existing sign names, as our results in Chapter 6 indicated may be necessary.

**VAE+LSTM** To include wider context, we propose a third architecture incorporating an autoregressive LSTM (Hochreiter and Schmidhuber 1997) language model. The input to this model is a sequence of character images  $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ . We encode each image using convolutional encoders with tied parameters to produce a sequence of codes  $\{\mathbf{z}_1, \dots, \mathbf{z}_n\}$ , and decode these using tied decoders to produce a sequence of images  $\{\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_n\}$ . Up to this point, the model is equivalent to a batched version of the basic VAE model. To incorporate context, we pass  $\{\mathbf{z}_1, \dots, \mathbf{z}_n\}$  to a unidirectional LSTM, and use our VAE decoder to decode the LSTM outputs to a second image sequence  $\{\tilde{\mathbf{x}}'_1, \dots, \tilde{\mathbf{x}}'_n\}$ .

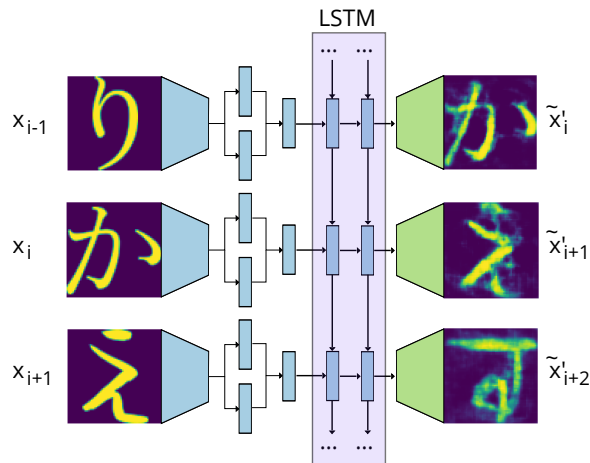


Figure 9.3: VAE+LSTM architecture. This model adds the auxiliary task of drawing the next token given a sequence of encodings for the preceding tokens.

We add the following loss term to Equation (9.1) to minimize the reconstruction error of this image sequence:

$$\mathcal{L}_{\text{LSTM}} = \sum_{i=1}^{n-1} \text{BCE}(\tilde{\mathbf{x}}'_i, \mathbf{x}_{i+1})$$

This can be viewed as an autoregressive character-level language modeling objective, where we wish to draw the image of the next character  $\mathbf{x}_{i+1}$  given all of the preceding characters  $\mathbf{x}_1, \dots, \mathbf{x}_i$ .

**VAE+Transformer** Our final model replaces the LSTM component from the previous model with a Transformer encoder stack (Vaswani et al. 2017); we obtain the output image sequence  $\{\tilde{\mathbf{x}}'_1, \dots, \tilde{\mathbf{x}}'_n\}$  by decoding the top layer of this Transformer. We train this model on a masked language modeling task: we mask input tokens at random by replacing their images with standard Gaussian noise, and train the model to recover the unmasked image sequence by adding the following term to Equation (9.1):

$$\mathcal{L}_{\text{Transformer}} = \sum_{i=1}^n \text{BCE}(\tilde{\mathbf{x}}'_i, \mathbf{x}_i)$$

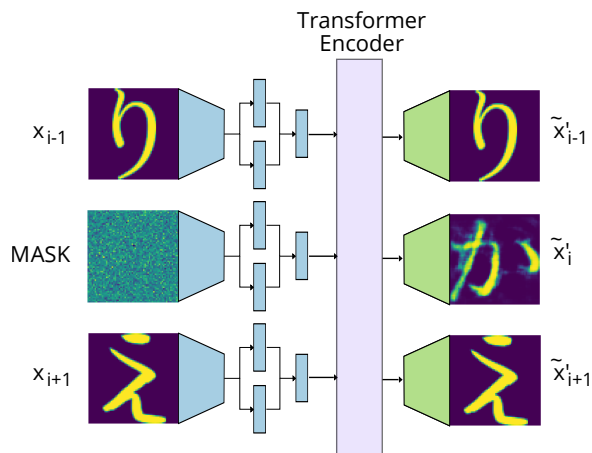


Figure 9.4: VAE+Transformer architecture. This model adds the auxiliary task of reconstructing characters which have been masked by random Gaussian noise.

### 9.2.3 Training Details

At training time, we use a denoising technique (Vincent et al. 2008, 2010) to regularize our models: we apply a random transformation (rotation of up to 45 degrees, shear of up to 25 degrees, and a random scale factor between 0.4 and 1) to each input image, while keeping the target of the reconstruction loss unchanged.

All of our models are trained using stochastic gradient descent (SGD) to minimize Eq. (9.1), plus the appropriate model-specific auxiliary loss ( $\mathcal{L}_{\text{Neighbor}}$ ,  $\mathcal{L}_{\text{LSTM}}$ , or  $\mathcal{L}_{\text{Transformer}}$ ), plus a *pseudolabel* loss term  $\mathcal{L}_{\Psi}$  which we describe below. We jointly minimize the sum of all of the relevant loss terms in a single pass, with no pretraining and no warmup steps.

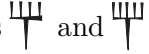
**Pseudolabels** We follow Corazza et al. 2022a in using a soft, unsupervised technique to organize our models’ encodings into loose clusters. This technique, inspired by the neural clustering algorithm DeepCluster-v2 (Caron et al. 2018), begins by clustering the encodings using K-Means with an *arbitrary* number of clusters  $k$ . Let  $C$  be a matrix with columns corresponding to the K-Means centroids (normalized to unit length). Let  $\mathbf{z}_i$  be an arbitrary encoding, let  $C_j$  be the centroid which is closest to  $\mathbf{z}_i$ , and let  $\mathbf{y}_i$  be a one-hot vector with a one in the  $j$ th position. The pseudolabel loss is then given by:

$$\mathcal{L}_{\Psi} = \sum_{i=1}^n \text{CCE}\left(\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} C, \mathbf{y}_i\right) \quad (9.2)$$

where CCE is categorical cross-entropy. For each  $\mathbf{z}_i$ , this constructs a probability distribution  $\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} C$  over  $k$  categories, where the mass in each category is proportional to the similarity between  $\mathbf{z}_i$  and the corresponding centroid. Minimizing this loss concentrates the mass of the distribution into a single term; in other words, this performs a soft clustering by pulling each  $\mathbf{z}_i$  towards the nearest centroid in the embedding space.

We follow Corazza et al. 2022a in using a pseudolabel loss with 100 centroids, which we recompute using K-Means at the beginning of each training epoch.

### 9.3 Data

We ultimately aim to apply these models to the study of proto-Elamite, where for each transliterated sign name there exists an accompanying digital image, produced by J. Dahl, depicting that sign’s “archetypal” form. (These are the same images which have been used throughout this work to provide inline examples of sign shapes.) These images smooth over many of the irregularities of the original shapes drawn on clay, while still preserving slight visual differences between tokens which may actually represent the same underlying character (such as ). They therefore represent an intermediate level of detail that is cleaner than segmented images of the original texts, yet still faithful to the original hand. We convert the entire transliterated proto-Elamite corpus into a set of image sequences by replacing each transliterated sign name with the corresponding sign image (Figure 9.5).

We evaluate our models on their ability to recover three scripts whose character inventories are already known (English, Japanese, and Cypro-Greek). We construct an English dataset by extracting the first 33k alphanumeric tokens (approximating the number of tokens that are attested in proto-Elamite) from the WikiText-2 corpus (Merity et al. 2016).

Language	# Characters	#Tokens	# Images
EN	62	33k	3410
JP	938	33k	1607
CG	55	3k	3005
PE	—	35k	1319

Table 9.1: Size and character inventories of scripts used for training. PE is undeciphered, and the size of its character inventory remains unknown.

We convert this text into image sequences by replacing each character token with a handwritten image of that character. We use images from de Campos et al. (2009), who provide 55 handwritten instances of all 62 upper- and lowercase English letters and digits: one of these 55 images is chosen at random each time a character occurs. The resulting sequences (Figure 9.5) imitate the level of detail in our proto-Elamite data, in that each letter is attested by multiple distinct images, and the same image can be used for multiple tokens.

We construct a Japanese dataset according to the same procedure, using the first 33k tokens from the Japanese Tatoeba corpus (Artetxe and Schwenk 2018; Tiedemann 2012). As we do not have handwritten character images for Japanese, we instead extract the glyphs from two Japanese fonts (Yuji Boku and Zen Old Mincho). The Japanese writing system uses three scripts: *kanji* which are highly logographic, and two syllabaries called *hiragana* and *katakana*. Similarly, proto-Elamite has been speculated to contain a set of possibly-syllabic signs, together with a large number of logograms (Dahl 2019). Syllabic signs convey phonetic information that can provide crucial insights for decipherment, and are therefore a major focus of decipherment efforts on this script. In our Japanese experiments (see Section 9.4), we therefore train on the entire script, but only evaluate the model’s ability to recover the two syllabaries.



Figure 9.5: Samples of image sequences from our PE (top), En (middle) and Jp (bottom) datasets.

We use the same Cypro-Greek data as Corazza et al. 2022a. Unlike the other datasets, this uses manually-segmented images from hand-drawn copies of artifacts bearing the Cypro-Greek script. There is therefore a greater degree of variation between the character shapes in this data, and no two tokens ever have identical images. This means that our three datasets fall along a cline from fully naturalistic, handwritten sequences (Cypro-Greek), to synthetic sequences derived from handwritten images (English), to synthetic sequences derived from digital fonts (Japanese).

Table 9.1 summarizes the token count for these datasets. We trim extraneous whitespace from all character images and resize them to  $64 \times 64$  pixels with a single grayscale color channel before training.<sup>1</sup>

## 9.4 Experiments

We train each of the models from Section 9.2.2 on the four corpora detailed above (see Appendix B.1 for hyperparameters and additional training details). After training, we encode each image using the trained encoder and cluster the resulting encodings using (i) agglomerative clustering with varying numbers of clusters (English, Japanese, proto-Elamite) or (ii) DBSCAN (Ester et al. 1996) with varying  $\varepsilon$  (Cypro-Greek). We use DBSCAN for Cypro-Greek to enable a fair comparison against the results in Corazza et al. 2022a; however, we find that DBSCAN is generally not effective when clustering the other scripts. When clustering with DBSCAN, we follow Corazza et al. 2022a in using a minimum cluster size of 2, to imitate a decipherment setting where the true number and frequency of characters is unknown; for the other scripts we vary the number of clusters for the same reason.<sup>2</sup>

For English, Japanese, and Cypro-Greek, we report homogeneity, completeness, and V-measure (Rosenberg and Hirschberg 2007) relative to the gold labels. Homogeneity ranges from 0 to 1, where 1 means that each cluster contains instances from just one of the underlying characters, and smaller values imply that some clusters combine instances of two or more distinct characters. Similarly, a completeness of 1 means that each of the underlying characters is represented by a single cluster, while smaller values mean that some characters have been divided across multiple clusters. Intuitively, low homogeneity scores mean that a clustering merges together characters which are actually distinct, while low completeness means that it splits some characters into subgroups that are not underlyingly distinct. V-Measure is the harmonic mean of homogeneity and completeness.

DBSCAN can label samples as outliers (and thus, not part of any cluster): we only evaluate on tokens which it assigns to a cluster.

In our Japanese experiments, we only evaluate on hiragana and katakana, in imitation of the proto-Elamite setting where we eventually aim to understand the divisions of a putative syllabary comprising only a subset of the overall script.

<sup>1</sup>Rescaling the images to a fixed size obscures the height of the original character (see Fig. 9.5, where  $\text{っ}$  is indistinguishable from  $\text{つ}$ ). For this reason, our Japanese evaluation only tests the model’s ability to recover the gojūon, dakuon, and handakuon, ignoring the yōon, sokuon, and small vowels which are distinguished only by size.

<sup>2</sup>The present work will otherwise ignore the problem of selecting the correct number of clusters, for which a variety of heuristics have been proposed in prior work (Rousseeuw 1987; Thorndike 1953; Sugar and James 2003; Honarkhah and Caers 2010; Tibshirani et al. 2002 i.a.).

In our Cypro-Greek experiments, we compare against the Sign2Vec and DeepCluster-v2 results reported in Corazza et al. 2022a. For the other scripts, we compare against agglomerative clusterings over the input images.

In proto-Elamite, where the ground truth is not known, we perform a qualitative evaluation in collaboration with domain experts. We look for sets of tokens which are assigned to the same cluster by our VAE+Neighbor, VAE+LSTM, or VAE+Transformer model, but belong to *different* clusters in the vanilla VAE model. The vanilla VAE differs from the other models in that it lacks contextual information; therefore, any groupings which are absent from this model’s output likely reflect primarily contextual similarities. Contextual resemblances are harder for human annotators to notice than visual resemblances, and so we expect these groupings to reflect similarities which may have been overlooked in prior work. We collaborate with domain experts to assess how these token groupings relate to their intuitions about this script.

## 9.5 Results

### 9.5.1 Modern Scripts

Figure 9.6 plots V-Measure from agglomerative clusterings over our models’ representations of handwritten English letters (Appendix A.7 shows the breakdown into homogeneity and completeness scores). The curve for the baseline is obtained by clustering the raw character images, rather than their encodings.

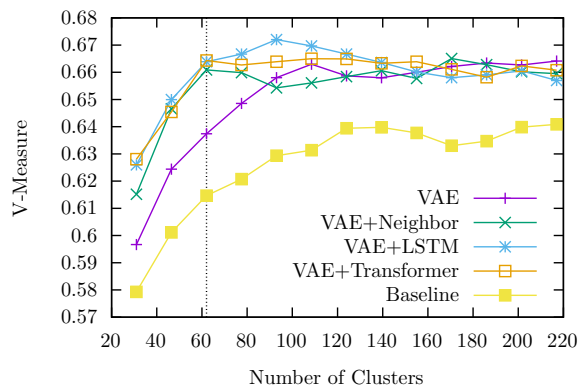


Figure 9.6: V-Measure on handwritten English. The true character inventory comprises 52 upper- and lowercase letters plus 10 digits.

All four of our proposed models are able to recover the underlying script more accurately than the baseline. When the number of clusters is close to the true size of the alphabet, our LSTM and Transformer-based models achieve the highest performance, which supports our hypothesis that wider context windows allow for more accurate script recovery.

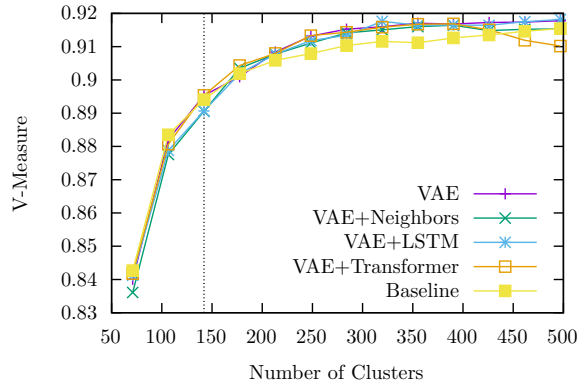


Figure 9.7: V-Measure on a synthetic mixture of Japanese fonts. The target character inventory comprises 142 hiragana and katakana (46 gojūon, 20 dakuon, and 5 handakuon each).

Figure 9.7 plots the same metrics for our synthetic Japanese dataset. In this setting, the differences between models are much less pronounced: the context-aware models do not exhibit the same advantage as in English, and in fact the VAE+LSTM model fails to outperform the naive baseline when the number of clusters exactly matches the true number of underlying characters. When the number of clusters is much larger than the true number of signs, our models do outperform the baseline, however, the contextual models continue to slightly underperform the contextless VAE on average. Regardless of the number of clusters chosen, the V-Measure for Japanese is always much higher than for English.

These differences between English and Japanese are likely due, in part, to the fact that there are only two distinct images per character in the Japanese data, compared to 55 in English. The Japanese data are also fully synthetic, whereas the English is handwritten. This may make the Japanese task too easy (despite covering a much larger number of unique characters) to the point that contextual models are not needed. This is nevertheless a useful result, as it suggests that the difficulty of script clustering depends less on the number of graphemes than on the degree of variation between allographs.

### 9.5.2 Cypro-Greek

Table 9.2 compares the best result from each of our models against the best results reported in Corazza et al. 2022a; Figure 9.8 shows our full results for different values of DBSCAN’s  $\epsilon$  parameter. Our best models (VAE+LSTM and VAE+Transformer) outperform the Sign2Vec baseline, and all of our models outperform DeepCluster-v2 (Caron et al. 2018) which was the inspiration for Sign2Vec. Although our V-measure gains are modest, we emphasize that our models use  **$\sim 98\%$  fewer parameters than Sign2Vec**, and  **$\sim 99\%$  fewer than DeepCluster-v2**. This supports our hypothesis that the ResNet encoder is

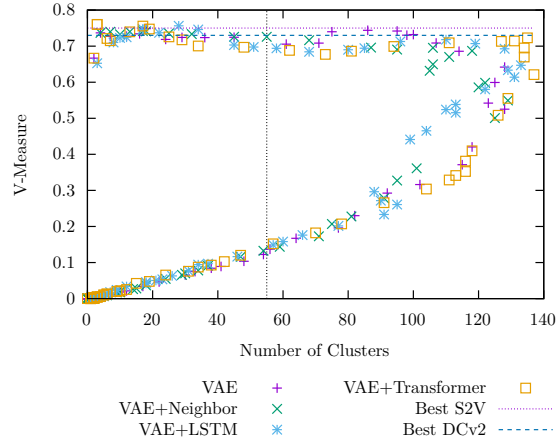


Figure 9.8: V-measure versus number of clusters for DBSCAN clusterings on Cypro-Greek. We evaluate over the interval  $0.1 \leq \varepsilon < 8$  in steps of 0.1. The dotted line represents the true number of signs in the script.

over-parameterized for the task of script clustering, and demonstrates that accurate script recovery is clearly possible even with much more lightweight architectures.

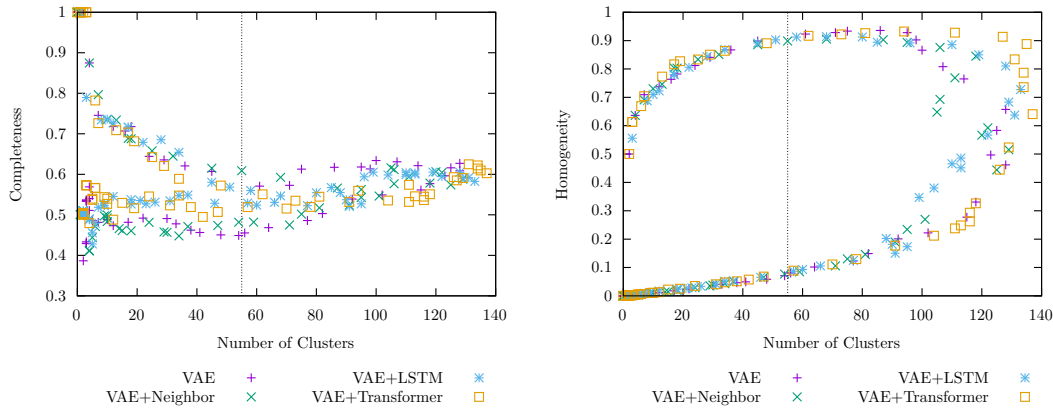


Figure 9.9: Completeness (left) and homogeneity (right) versus number of clusters for DBSCAN clusterings on Cypro-Greek. We evaluate over the interval  $0.1 \leq \varepsilon < 8$  in steps of 0.1. The dotted line represents the true number of signs in the script.

Figure 9.9 plots homogeneity and completeness versus number of clusters for each of our Cypro-Greek models. Each model exhibits a unique trend: the VAE+Transformer exhibits less variation in its completeness scores across a range of clustering sizes, while the other models exhibit a more pronounced fall and rise as the number of clusters increases. All models are capable of achieving comparable homogeneity in the neighborhood surrounding the true number of clusters, but the VAE+Transformer maintains high homogeneity up to a much higher number of clusters than the other models. Geshkovski et al. (2023) have argued that self-attention mechanisms cause tokens to cluster around certain attracting states in



	$V \uparrow$	Parameters $\downarrow$
DeepCluster-v2 (Corazza et al. 2022a)	0.73	> 23M
Sign2Vec (Corazza et al. 2022a)	0.75	> 11M
VAE (Ours)	0.75	<b>0.215M</b>
VAE+Neighbor (Ours)	0.74	0.215M
VAE+LSTM (Ours)	<b>0.76</b>	0.218M
VAE+Transformer (Ours)	<b>0.76</b>	0.227M

Table 9.2: V-measure ( $V$ ) and parameter counts for Cypro-Greek. Best results for each model from Figure 9.8 and Corazza et al. 2022a.

the representation space; pseudolabeling (Caron et al. 2018) is intended to have the same effect. We speculate that the VAE+Transformer’s strong performance may result in part from these behaviours reinforcing one another to perform a more effective soft clustering at training time.

### 9.5.3 Proto-Elamite

Table 9.3 shows pairs and triplets of proto-Elamite characters which have distinct labels in the working signlist and VAE clustering, yet occupy the same cluster in the VAE+Neighbor, VAE+LSTM, or VAE+Transformer clusterings.<sup>3</sup>


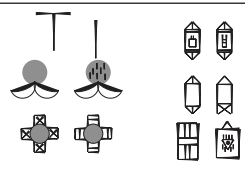
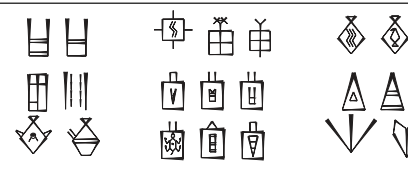






VAE+Neighbor	VAE+LSTM	VAE+Transformer
		

Table 9.3: Pairs/triplets of character images which have distinct labels in the working signlist, but which our models merge into single clusters.





The VAE+Neighbor model differs from the working signlist and VAE clustering in only two places, merging M332~g  with M297~b  and M356~b  with M327~n . Neither merger appears to reflect any known similarities in how these signs are used.









By contrast, the 6 mergers proposed by the VAE+LSTM model appear much more plausible. One cluster combines tokens which are currently labeled as M362  and M362~a , which adds hatching to the central circle in a manner resembling “gunuification” in early cuneiform. The ~ notation in the working sign name explicitly acknowledges that

<sup>3</sup>For the VAE model, we use 1306 clusters, which equals the number of unique sign images available at the time we created our training data. We cluster the other models using  $3.5\times$  this many clusters; using such a large number helps to guarantee that the observed groupings reflect legitimate similarities and are not simply a side effect of compressing too many tokens into too few groups.

experts believe M362~a may<sup>4</sup> be a graphic variant of M362; both signs have been glossed as “nanny goat” (Dahl 2005a), and previous scholarship has already acknowledged a likely equivalence between M362~a and another hatched variant called M362~b (Dahl 2005a)
















The output from our VAE+Transformer differs the most from the working signlist, suggesting 13 sets of shapes which may represent the same underlying character. A significant number of these are complex graphemes. We have previously suggested that the meaning of a complex grapheme is principally determined by its outer component (Chapter 6), and indeed most of the merges proposed by the VAE+Transformer occur between complex graphemes with the same outer part. This suggests that the model has rediscovered the same pattern identified in earlier chapters, and that these particular merges do reflect plausible groupings on the part of our model. Many of the other mergers occur between signs which are already labeled as possible variants in the working signlist (such as M029~a  and M029~b , or the possible syllables M387~h  and M387~ca ) and thus appear similarly plausible.

Of greater interest are cases such as M209~a , M210~f , and M195+M057 . The working signlist labels these as wholly distinct characters, and no explicit relationship between these signs has been proposed in prior work. However, the visual resemblance between M209~a and M210~f is undeniable; both signs occur in texts which contain the “yoke” character M054 , and both occur in texts which appear to record amounts of grain (M209~a appears with the speculative grain capacity sign M354 , while M210~f occurs with the more common container sign M288). Moreover, in one text M209~a is attested alongside a related variant of M210, labeled M210~d . Given these signs’ shared visual resemblance to a plant sprouting from a field, and their association with yokes and grain accounts, we believe it is reasonable for the model to have grouped these characters under the same umbrella. M195+M057 is also attested in texts alongside both M288 and M354; although it does not occur next to the yoke sign M054, it often occurs near the sign M003~b , which is speculated to be another field utensil and which experts note is “related to M054” (Dahl Unpublished). Both M195+M057 and M209~a are also attested as headers. While we are skeptical that M195+M057 is truly the same underlying character as M209~a and M210~f, they clearly share contextual similarities and are attested in comparable, apparently agricultural, contexts. The inner part of the complex grapheme has been compared to proto-cuneiform A , referring to water, on the basis of which this sign could denote an irrigated field. Then this group of signs may refer to distinct types of

<sup>4</sup>Specifically, ~ followed by a number marks one sign as a graphic variant of another; ~ followed by a letter means that the sign *may* be a variant of another, but experts remain agnostic in the absence of further evidence.

field (irrigated and non-irrigated), or to the same field at different points in time (before and after sprouting, perhaps seeding and harvest time respectively). This example clearly demonstrates our proposed model’s ability to detect contextual parallels which are helpful for understanding the possible relationships between signs in this script and which have not been remarked on previously.

By reducing the number of clusters to force additional merges, we can obtain yet more groupings of the sort reported in Table 9.3. For example, when we reduce the number of clusters in the VAE+Transformer model by a factor of  $\frac{1}{7}$ , a new merger appears between  $M175+M131 \sim d$   and  $M157+M131 \sim d$  . The outer components  $M157$   and  $M175$   differ only in the shape of the protrusion at the top of the box, and a merger between these signs is tentatively expected based on current understandings of the corpus. Other mergers which appear, and which are also expected based on current understandings of the corpus, include  $M056 \sim f$   with  $M056 \sim e$  , both signs being understood to depict a plow;  $M075 \sim ff$  ,  $M075 \sim g$  ,  $M075 \sim h$  , and  $M075 \sim o$   apparently depicting minor variations on a sprouting plant; and  $M111 \sim c$  ,  $M111 \sim d$  , and  $M111 \sim e$   which differ only in the direction of the internal hatching. Such cases serve as useful confirmations of experts’ current understanding of sign use in this script, as they show that there are no unremarked-on patterns of use which cause the model to keep these signs separate in its representation space.

The cases reported so far represent only a small fraction of the candidate mergers which can be extracted from our models, and we are optimistic that this work will continue to give rise to useful insights as we take the opportunity to investigate this space more fully together with domain experts.

## 9.6 Related Work

Scribal hand identification (Popović et al. 2021; Srivatsan et al. 2021) is a related task which seeks to cluster instances of characters from a known script according to the hand which wrote them. Like our work, the latter authors propose to learn sign representations using an autoencoder; they then factor these representations into disjoint character and scribal hand embeddings. Unlike our work, they consider signs in isolation, ignoring the surrounding text content which we found to be beneficial to our models.

Yin et al. (2019) describe a system for segmenting, transliterating, and deciphering images of historical manuscripts. In the transliteration step, their model implicitly learns an underlying script by clustering character representations obtained from a Siamese neural network trained to discriminate between characters from known scripts. This network learns character representations without access to context, similarly to our vanilla VAE and the DeepCluster-v2 baseline in Corazza et al. 2022a.

In a setting where the underlying script is already known, Dencker et al. (2020) and Gordin et al. (2020) also describe systems for automated transliteration from images of cuneiform text.

Kelley et al. 2022a describes our own initial attempts to bridge the gap between the models described in this chapter and those from Chapter 6. In that work, we did not use a variational model; rather we used softmax decoding over a fixed vocabulary, which introduced label bias and made it challenging to assess the efficacy of the proposed model. Our models in this chapter use a deconvolutional decoder, which sidesteps this bias by allowing an open vocabulary.

## 9.7 Conclusion

We have described four models which add varying degrees of contextual information to a VAE, and have shown how these can be used to cluster token images to recover a script’s character inventory. On the ancient Cypro-Greek script, our best models meet or outperform the state-of-the-art Sign2Vec baseline using just  $\sim 2\%$  as many parameters, which supports our claim that written text lacks the visual complexity to warrant models of the depth used in other image processing applications. Our English and Cypro-Greek experiments also demonstrate that contextual models are more effective for script recovery than contextless models. On synthetic Japanese data, which contains many distinct graphemes but little variation between allographs, our models achieve extremely high V-Measure ( $>0.91$ ), suggesting that they handle large character inventories more easily than they handle allography.

We have applied our models to study the undeciphered proto-Elamite script, and have shown that they capture existing intuitions about this script as well as suggest new parallels between signs which have never been noted in prior work. Our best insights for proto-Elamite come from the LSTM and Transformer models, while for Cypro-Greek our VAE+Neighbor model is the only one which produces a clustering with precisely the same number of clusters as there are signs in the underlying script. This indicates that it is useful to consider models with varying access to contextual information according to the number of long-distance contextual dependencies the input script is expected to exhibit.

## Part IV

# Additional Results

## Chapter 10

# Fast Cognate Alignment on Imbalanced Data

### 10.1 Background

Cognate alignment models seek to align monolingual word lists from related languages, in order to identify pairs of cognate lexemes. If one of the word lists comes from an undeciphered language, this process will yield a decipherment for whichever words are correctly mapped to their cognates.

As of this publication, the current state-of-the-art for cognate detection comes from Tamburini 2023, which uses coupled simulated annealing (Xavier-de-Souza et al. 2010) to solve a search problem over the set of all  $k$ -permutations mapping  $n$  lost-language words into  $k$  known-language buckets. Prior to this, the state-of-the-art came from Luo et al. 2019 which alternated between searching for character-level and word-level pairings using an iterative, expectation-maximization-style training procedure.

Both of these models achieve strong results (upwards of 95% accuracy in the best case) on existing cognate detection datasets. However, these results come from settings where every word in the dataset has at least one valid cognate in the language it has been paired with. This is artificially clean compared to true decipherment settings, where not only are there likely to be unpaired words in both languages, but there may also be uncertainties about underlying word boundaries or character inventories which introduce further kinds of noise. Luo et al. (2019) acknowledge this limitation and include an evaluation on less-clean Ugaritic-Old Hebrew data; in this setting their alignment accuracy drops to just 65.9%, from 93.5% in the clean setting. Tamburini (2023) does not include any evaluations in a noisy setting, likely because their technique involves a computationally intensive search process that we do not anticipate to scale effectively to a setting with many unpaired words.

Although the search process underlying Luo et al. 2019 is less intensive than that behind Tamburini 2023, it nonetheless involves multiple iterations of EM-style optimization which are both time and parameter intensive. We hypothesize that it may be possible to make

this process faster and more parameter-efficient by shifting to a language modeling based approach. This is motivated by our recent observation (Kambhatla et al. 2023) that it is possible to eliminate brute-force search from *sentence*-level decipherment tasks by training a language model to “translate” from a script- and language-agnostic input representation back to plain Latin characters. At inference time, our approach proposed in that work required only straightforward beam-search decoding, making it both fast and effective compared to the then-state-of-the-art. More efficient alignment techniques would be useful for real decipherment settings where the correct target language is not known beforehand. In such a setting, it may be necessary to align to multiple languages before a true solution can be found, and this scattershot approach is most feasible when each individual language pair can be aligned efficiently.

Moreover, we note that in the noisy setting Luo et al. (2019: 3152) “found it beneficial to train [their] model only on a randomly selected subset (10%) of the entire corpus with the same percentage of noncognates, and test it on the full dataset”, but observe that this kind of data filtering is impossible in a realistic setting, where it is not known which words have cognates and thus the relative proportion cannot be maintained. In a realistic setting, the word-lists from both languages would need to be sampled independently, meaning that a 10% subset of the corpus should be expected to contain a mere 1% of the original cognate pairs, destroying most of the signal that the model would learn from. It is not clear whether Luo et al.’s filtering was “beneficial” for reasons of efficiency, or of accuracy. If the former, this emphasizes all the more clearly the need for faster solutions to this task. If the latter, then the reported numbers are actually unfair from the perspective of *de novo* decipherment, and more robust methods will be required to deal with large, unfiltered word lists.

In this chapter we will demonstrate a fast and simple approach to learning the character-level mapping between two scripts by adapting a monolingual language model; the proposed technique is robust against noisy data and is significantly more efficient than the state-of-the-art in Luo et al. 2019. We leave it to future work to extend this to word-level cognate identification, but we demonstrate promising preliminary results using edit distances derived from the learned character-level mapping.

While there is good reason to doubt that proto-Elamite is glottographic, it is nonetheless worth exploring the possibility that the “syllabary” signs may truly be syllabic, as establishing phonetic values for these signs would be the most significant breakthrough in the decipherment of this script to date. Our results in this chapter set the groundwork for such an exploration, by providing a model that is demonstrably robust against realistic noisy data and that can be rapidly applied to a wide battery of candidate languages in future work.

## 10.2 Character-Level Alignment

### 10.2.1 Motivation

Tran (2020) considers the problem of transferring a pretrained English model to another language assuming a limited computational budget. Given a matrix of pretrained English (sub)word embeddings  $\mathbf{E}_e$ , they propose to model the embeddings of foreign-language words  $\mathbf{E}_f$  as linear combinations of the English vectors:

$$\mathbf{E}_f[i] = \sum_{j=1}^{|V_e|} \alpha_{ij} \mathbf{E}_e[j] = \alpha_i \mathbf{E}_e$$

where  $\alpha_i$  is a sparse weight vector satisfying  $\sum_j^{V_e} \alpha_{ij} = 1$  (i.e. it can be interpreted as a probability distribution over English words). The authors describe two offline approaches for estimating the weights in  $\alpha$  by using existing word alignment techniques (Dyer et al. 2013; Conneau et al. 2017). Otherwise, the authors do not devote much focus to this mapping, as it is only used to initialize the embeddings of unseen words before fine-tuning.

Cognate detection datasets are simple word lists with no surrounding context, meaning the proposed techniques for estimating  $\alpha$  are not directly applicable in this setting. However, we propose that a similar approach could be applied at the character level. Concretely, we propose to first train a monolingual character-level language model on the known-language word list, and then to model each lost-language character as a weighted sum over the known-language character embeddings. Next, rather than estimating the weights for this mapping in an offline step, as in Tran 2020, we propose a novel training technique that allows us to directly fine-tune the known-language model on lost-language data. By performing this fine-tuning with the mapping  $\alpha$  as the only tunable parameter, we will show that the model comes to learn the correct character-level alignment between the two scripts. After this fine-tuning step, we can use the inferred mapping to estimate word-level alignment probabilities as part of a separate process outlined in Section 10.3.<sup>1</sup>

### 10.2.2 Methodology

Concretely, the first stage of our proposed character alignment process entails training a character-level language model on the known language. The inputs to the model are individual words from the known-language wordlist, with no additional context; words are tokenized at the character level (we do not use subwords). Given the small size of cognate

<sup>1</sup>To some extent, the approaches explored in this chapter can be viewed as a continuous relaxation of the MATCHER system from Berg-Kirkpatrick and Klein 2011, which uses a similar approach of learning character-level mappings and then aligning words based on edit distance. In that work, word- and character-level mappings are learned jointly, and the character-level mapping was discrete. In this work, we learn a *continuous* character-level mapping in the form of a weighted sum, and we show that this mapping can be effectively learned *independent* of any word-level alignments.



detection datasets, we use a shallow Transformer with a small feature dimension (2 layers, 4 heads, and 32 dimensions) as larger models fail to converge. We use positional encodings (Vaswani et al. 2017) and apply dropout at a rate of 0.5. This model is trained with SGD to minimize categorical cross-entropy on an autoregressive language modeling task using a causal attention mask.

Let  $E \in \mathbb{R}^{k \times 32}$  be the embedding layer from the now fully-trained known-language model, where  $k$  is the number of known-language characters. Let  $M : \mathbb{R}^{n \times 32} \rightarrow \mathbb{R}^{n \times k}$  be a black-box representation for the remainder of the language model, which maps a sequence of  $n$  32-dimensional character embeddings onto a sequence of log-probabilities over  $k$  known-language characters.

To allow this model to accept lost-language inputs, we introduce a mapping  $\alpha \in \mathbb{R}^{l \times k}$  following Tran 2020. The product  $\alpha E \in \mathbb{R}^{l \times 32}$  can be seen as an embedding matrix for  $l$  distinct lost-language characters, and  $M \circ \alpha E$  can be seen as a hybrid language model which accepts lost-language inputs and predicts known-language outputs.

To convert the outputs from  $M \circ \alpha E$  into a distribution over lost-language characters, we introduce the following conditional probability distribution inspired by t-SNE (Hinton and Roweis 2002; van der Maaten and Hinton 2008):

$$\log p_{j|i} = \frac{-|\mathbf{x}_i - \alpha E_j|^2 / 2\sigma_i}{\sum_{h \neq i} -|\mathbf{x}_i - \alpha E_h|^2 / 2\sigma_i} \quad (10.1)$$

where  $1 \leq i \leq k$ ,  $1 \leq j \leq l$ ,  $\mathbf{x}_i$  is the embedding for the  $i$ th known character,  $\alpha E_j$  is the embedding for the  $j$ th lost character, and  $\sigma_i$  is a per-character density estimate. Given a known-language character  $i$ , suppose we sample neighboring characters in the embedding space based on their distance from  $\mathbf{x}_i$ , with Gaussian falloff. Assuming we are only allowed to sample neighbors from the lost language,  $p_{j|i}$  models the probability that the lost-language character  $j$  will be the one sampled. This is equivalent to the sampling procedure used in t-SNE (van der Maaten and Hinton 2008) with the modification that points are divided into two classes, and each class can only sample points from the other class.

Given a distribution  $\mathbf{p} = [p_1, \dots, p_k]$  over known-language characters returned as output by  $M \circ \alpha E$ , we model the corresponding distribution  $\tilde{\mathbf{p}} = [\tilde{p}_1, \dots, \tilde{p}_l]$  over *lost*-language characters as  $\tilde{p}_j = \sum_{i=1}^k p_i p_{j|i}$ . With this transformation in hand, we have now successfully converted the original known-language model to both accept lost-language inputs *and* predict lost-language outputs, using nothing but the character-level mapping  $\alpha$ .

We now proceed with fine-tuning the adapted model on the lost-language word list, following exactly the same procedure that was used to train the underlying known-language model. During this fine-tuning step, however, we make  $\alpha$  the only tunable parameter, meaning that the only way for the model to improve the language modeling loss at this stage is by learning the correct mappings between characters in the two scripts.

We emphasize that this procedure makes the mapping  $\alpha$  *trainable*, whereas the mapping from Tran 2020 was static and estimated offline.

**Permutation Loss** It is generally assumed that the mapping from characters in one script to characters in another will be sparse: it may be almost one-to-one when aligning two alphabets, and it is unlikely to exceed two- or three-to-one when aligning a syllabary to an alphabet. In light of this, we hypothesize that our model may learn the mapping more effectively if we constrain  $\alpha$  to be approximately one-to-one.

To achieve this, we generalize the continuous matrix penalty function introduced by Lyu et al. (2020) to the case of non-square  $k \times l$  matrices:

$$\mathcal{L}_{sparse} = \sum_{i=1}^k \left[ \sum_{j=1}^l |\alpha_{ij}| - \left( \sum_{j=1}^l \alpha_{ij}^2 \right)^{1/2} \right] + \sum_{j=1}^l \left[ \sum_{i=1}^k |\alpha_{ij}| - \left( \sum_{i=1}^k \alpha_{ij}^2 \right)^{1/2} \right]$$

When applied to a square matrix  $\alpha$ , this quantity approaches zero as the matrix approaches a permutation; by extension, in the non-square setting it approaches zero as  $\alpha$  approaches some non-square *projection* of a permutation. We hypothesize that fine-tuning to jointly minimize the sum of the cross-entropy language modeling loss with this sparsity loss will yield more accurate character-level alignments, and by extension, more accurate cognate identifications, than fine-tuning on the language modeling loss by itself.

### 10.2.3 Experimental Results

We train the proposed model on the complete Ugaritic-Old Hebrew dataset from Luo et al. 2019, without pruning any of the distractor vocabulary items. We consider two settings, one where we learn a model on the known language and fine-tune the alignment on the lost language, and the reverse where we learn a model on the lost language and tune on the known. In each setting, we compare the outcome of training without the permutation loss  $\mathcal{L}_{sparse}$ , with the permutation loss for just the first 50 iterations as a kind of warm-up, and with the permutation loss for the entire duration of training.

The mapping  $\alpha$  captures a nuanced view of the relationship between characters in the two scripts in the form of a weighted sum. We can attempt to convert  $\alpha$  to a more concrete, one-to-one mapping for evaluation purposes, but we note that this process will necessarily lose some of the information inherent in the full set of weights. For example, Table 10.1 reports top-1 and top-5 precision for mappings derived by aligning each character in the known script to the character(s) with the highest likelihood  $p_{j|i}$  according to Eq. (10.1).

From these results it is clear that the most effective way to learn the mapping  $\alpha$  is by pretraining on *known* language data, then fine-tuning on the lost language. In this setting,  $\alpha$  appears to perfectly capture the mapping between the two scripts, achieving 100% top-5 precision in the best case. In the opposite direction, the best result is just 74% top-5

Pretraining Language	Permutation Loss					
	None		Warm-Up		Always	
	top 1	top 5	top 1	top 5	top 1	top 5
Old Hebrew (Known)	26%	57%	<b>83%</b>	<b>100%</b>	13%	30%
Ugaritic (Lost)	48%	74%	48%	70%	0%	17%

Table 10.1: Top-1 and top-5 precision of character-level mappings derived from  $\alpha$ .

precision. We speculate that this asymmetry derives from the fact that there is much more known-language data than lost-language data, meaning that the initial model will learn higher-quality character representations when trained on the known language.

Our proposed permutation loss does appear to improve the character-level alignment, but only in certain settings. Specifically, when we pretrain on the known language, and apply the permutation loss for just the first 50 iterations of the alignment step, we see significant improvements in the quality of the learned alignment. In other settings, we find that this term has no effect or is actively detrimental to the quality of the learned mapping. This suggests that it is useful to “prime” the model to expect a roughly one-to-one mapping at the start of training, but that this requirement must eventually be relaxed. This makes sense given that the true mapping between these scripts is not bijective, and some many-to-one or one-to-many relationships do exist (as between Ugaritic  $a$ ,  $u$ ,  $i$  and Old Hebrew  $a$  in most contexts).

## 10.3 Towards Word-Level Alignment

### 10.3.1 Methodology

After fine-tuning on the lost language data,  $\alpha$  contains a representation for each lost-language character as a linear combination of known-language characters. We now wish to use these combinations to infer likely pairings between cognates at the word level.

**Edit Distances** To do this, we first compute the Levenshtein edit distance (Levenshtein 1966) from every lost word to every known word, where the cost of substituting lost character  $j$  with known character  $i$  is

$$C(j, i) = 1 - \frac{p_{i|j}}{\max_i p_{i|j}}$$

where  $p_{i|j}$  is the conditional probability that known character  $i$  would be sampled by lost character  $j$  paralleling Eq. (10.1). Note that  $C(j, i) = 0$  whenever known character  $i$  is the nearest neighbor to lost character  $j$  in the embedding space, while for all other pairings the cost rises proportionally to the distance between  $i$  and  $j$ . Thus there is no penalty

for replacing a lost-language character with its most likely known-language correspondent, while there is increasing cost for less likely substitutions.

**Alignment Likelihoods** Now, for lost- and known-language vocabularies  $V_l$  and  $V_k$ , let  $\Delta \in \mathbb{R}^{|V_l| \times |V_k|}$  be a matrix where  $\Delta_{mn}$  is the weighted edit distance between lost word  $m$  and known word  $n$  as computed using the weights described above. Let  $D_m = \text{softmax}(-\Delta_m)$  be a probability distribution where  $D_{mn}$  is the probability that lost word  $m$  aligns to known word  $n$ , and note that the resulting probabilities are inversely proportional to the original edit distances.

To obtain a more sophisticated model for the word-level alignments, we can incorporate a prior estimate for the likelihood that known word  $n$  is cognate to some lost word, as opposed to being a distractor that has no mapping into the other language. Existing approaches to cognate detection excel in clean settings where there are few or no such distractors, so the ability to identify and prune these words would be a useful result in itself.

To this end, we propose to learn a language model on the *lost* word list, then fine-tune on the known word list following the same procedure outlined in Section 10.2. In the resulting model, the average perplexity when producing known word  $n$  should be low if  $n$  is cognate to a lost word, as in this case the underlying lost-language model will have seen that cognate during pretraining and the corresponding known word will look “in-domain”. By contrast, if known word  $n$  is not cognate to any lost words, it should be “out-of-domain” and therefore incur a higher average perplexity. We construct a vector  $\Pi \in \mathbb{R}^k$  where  $\Pi_n$  is the average perplexity when producing the characters in known word  $n$ . We convert this to a probability distribution  $P = \text{softmax}(-\Pi)$ , and estimate the probability that lost word  $m$  aligns to known word  $n$  as  $P_n^r D_{mn}$  where  $r$  is a smoothing term.

## 10.4 Results

**Identifying Candidate Cognate Pairs** The Ugaritic-Old Hebrew dataset contains 38898 total Hebrew words, meaning that *a priori* each Ugaritic word could be aligned to any of 38898 possible targets. We wish to narrow this to a small pool of candidate cognates for each word: ideally, we want just the most likely cognate in each case, but even tens of candidates per word is a manageable amount for reranking or to allow human analysts to evaluate whether a proposed alignment has been successful.

To this end, for each lost word, we select the  $k$  known words with the highest probability according to the distribution  $P_n^r D_{mn}$  described above. We call these *candidate cognate pairs*, and consider the alignment successful if the true cognate is among the  $k$  chosen terms. Figure 10.1 plots the top- $k$  precision for different values of  $k$  and the smoothing parameter  $r$ .

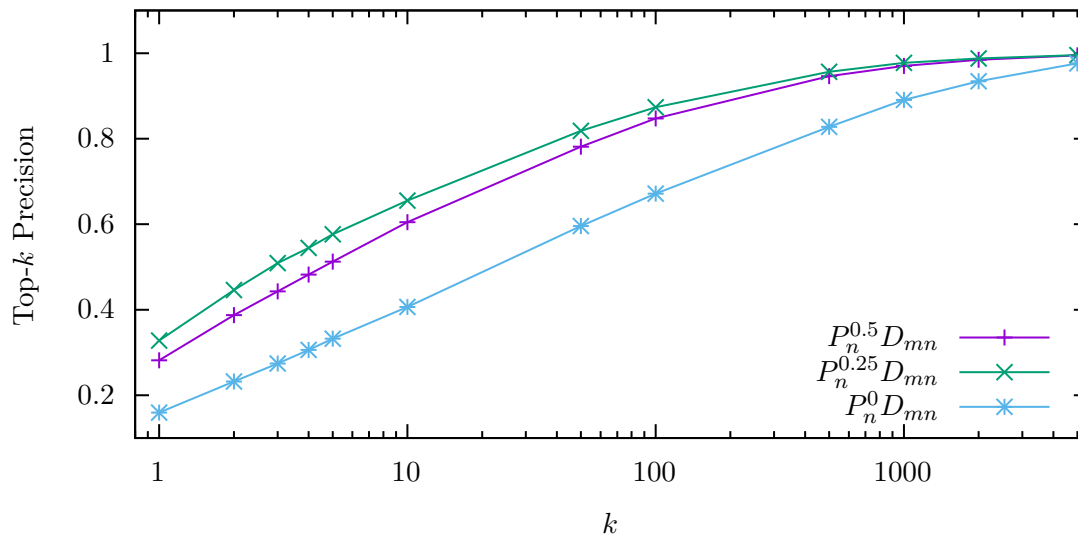


Figure 10.1: Top- $k$  precision on Ugaritic-Old Hebrew cognate detection for various thresholds  $k$  and values of the smoothing parameter  $r$ .

The best results overall come from values of the smoothing parameter  $r$  around 0.25. In this setting, nearly half (46%) of true cognates are either the most likely or the second-most likely candidate according to  $P_n^r D_{mn}$ , and 66% fall within the top 10 most likely candidates. 66% is the same accuracy reported by Luo et al. (2019) for their noisy evaluation. These results suggest an avenue for future work whereby the top- $k$  candidates are reranked to promote the *true* cognates to the top of the ranking. Under this approach, the correct cognate would only need to be selected from a few tens or hundreds of candidates, rather than the full set of many thousands.

We also note that top- $k$  precision is highest when we exploit  $P$  as a prior estimate for whether a word has a cognate or not. In other words, the distribution  $P$  provides a usable signal to help assess whether or not a given word is a distractor. This is encouraging, as it is otherwise difficult to tell which words are “out-of-domain” when dealing with undeciphered data.

## 10.5 Conclusion

In this chapter, we have demonstrated a novel technique for learning the character-level mapping between two related languages by training a monolingual language model on one, then fine-tuning on the other. We show that this technique appears to perform best when there is a large amount of known-language data available, which makes it particularly suitable for large, imbalanced cognate detection datasets where existing models currently struggle. The proposed technique uses a novel t-SNE-inspired approach to derive predictions over

lost-language characters from the original known-language model outputs; we suggest that this technique may also be applicable at the word level, where it would serve as a fully trainable generalization of Tran 2020.

We also describe a first attempt at extrapolating to word level alignments from these character-level results. We find that using the learned character alignment scores to compute weighted edit distances provides a very strong signal for aligning between two word lists, with nearly half of all cognate pairs having the smallest or second-smallest edit distance to their cognate, and 95% falling within the top 400. We therefore propose to reframe cognate alignment in noisy settings as a reranking task, where an initial set of words numbering in the thousands or tens of thousands is pruned, using our technique, to a set of candidate cognate pairs numbering in the dozens or hundreds. These can then be rescored to find the best overall match, potentially using an existing approach to cognate detection (which could be applied much more efficiently than usual in this setting, as the model would no longer need to search over the full space of possible pairings). Model speed is useful in true decipherment contexts where it can be necessary to align to multiple candidate languages before finding a true match, and this is precisely the setting where we intend to apply our proposed approach in upcoming experiments.

We have also shown that perplexity scores from a model trained on one language can provide an effective signal for identifying words that have no cognate in the other language. This is significant as it enables a kind of data cleaning or pruning which would otherwise be impossible for undeciphered data. We believe that this signal could be incorporated into any new or existing cognate detection model to improve its performance in noisy settings.

# Chapter 11

## Additional Results



### 11.1 Linear Elamite Sound Values

This section reproduces parts of Kelley et al. (2022b), where we have provided an initial response to the possible proto-Elamite sound values suggested in Desset et al. 2022. In that work, Desset et al. propose sound values for Linear Elamite signs which yield a candidate decipherment of that script into the Elamite language, and suggest that it may be possible to “proceed in a regressive way, starting from the [sound] values established for the LE signs [...] and trying to apply these “readings” to their graphic counterparts in the earlier PE writing [...] The same signs may have been used with similar or identical phonemic values [...] in the late 4th millennium BCE PE tablets” (Desset et al. 2022: 53).

We have already highlighted (Chapter 2) some of the difficulties surrounding this line of argument, but we expand on these now to better frame the following discussion. At present there is understood to be a wide gap in time between the proto-Elamite texts and the appearance of Linear Elamite, on the order of eight centuries (Englund 2004: 104), with no evidence for the continual transmission of proto-Elamite writing traditions across this period. Thus the appearance of Linear Elamite does not look like a natural temporal development of proto-Elamite, so much as it looks like the revival of a dead technology by users centuries removed from any scribe who could have known the original readings of the signs. There is therefore scant reason to expect those readings to have been maintained, particularly when we consider that the signs in question are predominantly abstract geometrical shapes, and not iconic representations of objects that users may be expected to call by the name of the thing they depict. The only clear connection between the two scripts is a visual resemblance between the shapes of some signs, but Englund (2004: 143-144) notes that such “seeming graphic correspondences are notoriously inaccurate and can only be pursued as an avenue of decipherment **within the framework of a continuous writing tradition** such as that of Babylonia, but even then must be considered highly tentative” (emphasis added).

In the interest of exploring every possible avenue, we nonetheless consider the implications should the sound values from Desset et al. (2022) actually reflect some part of the proto-Elamite reality. This exploration rests on the following assumptions:

1. that the sound values proposed in Desset et al. 2022 are substantially correct for reading the Linear Elamite texts; and
2. that the mapping between Linear Elamite and proto-Elamite sign shapes, outlined in Table 7 of Desset et al. 2022, is sufficiently complete and correct to allow these sound values to be mapped back to proto-Elamite.

The second assumption in particular remains unproven. The proposed mapping occasionally assigns several proto-Elamite signs to the same Linear Elamite correlate, and some of these mergers appear unlikely when we consider the proto-Elamite contexts where those signs occur. For example, M002  and M387  are both equated with Linear Elamite *na*. As we argue in Kelley et al. 2022b: 6, the former consistently appears as an object sign introducing capacity measures, while the latter has more varied uses as a possible syllabic sign and as an owner sign (Dahl 2019: 77, 84). There is little evidence in the proto-Elamite corpus that these should be treated as graphical variants of the same underlying sign, so the implication that they should both map to the same Linear Elamite sign appears suspect. We also note that 18 of the proposed correlates are *hapax legomena* in the proto-Elamite corpus: we reiterate our claim (Kelley et al. 2022b: 6) that “It would be quite remarkable if a sign with a single attestation across 26 thousand tokens in the currently known corpus [...] were to prove resilient enough to be transferred with a similar value into a much later writing system.” Overall, we believe the mapping in Desset et al. 2022 is meant only as a general suggestion based on visual cues, but these cases highlight that more work must be done to obtain a properly nuanced mapping if a connection between these scripts is to be explored.

We have also considered the result of replacing each proto-Elamite sign in the corpus with the corresponding Linear Elamite sound value based on the proposed mapping, to assess the frequencies of individual sounds and of syllable *n*-grams in the resulting phoneticised data. The resulting sequences include some, like *m-t* and *m-la-a*, that appear phonetically implausible unless we assume additional vocalic segments that are not reflected in the proposed sound values.

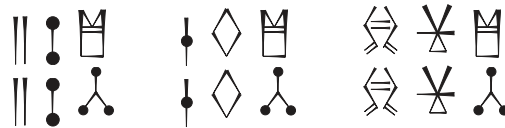
We refer the reader to our original publication of these results in Kelley et al. 2022b for additional details, including discussion of the resulting word forms from a more Assyriological point of view. We omit that discussion from this work, as it is both out of scope from a technical perspective and is not wholly original to the author of this thesis.



## 11.2 Kober's Triplets

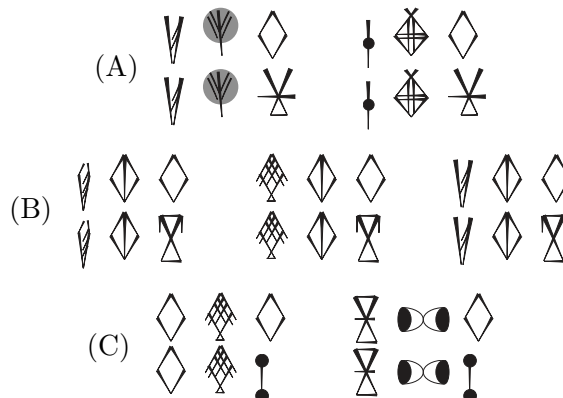
In Section 5.2 we raised the possibility that the proposed anthroponyms or strings of so-called syllabic signs may have some internal structure, as certain signs exhibit an apparent preference for what resemble word-final positions. That discussion brings to mind the work of Alice Kober in the decipherment of Linear B (Kober 1945), whose discovery of affixal morphology ultimately proved crucial to the later identification of that script as encoding a form of Greek (Chadwick 1958). Kober identified words which differed only in their last character or characters; by then grouping words which exhibit similar sets of endings, it became possible to derive tables of the sort depicted in Chadwick 1958: 35. Such tables resemble inflectional paradigms as may be found in any language textbook, and indeed the alternations involved would later prove to encode aspects of Greek inflectional morphology.

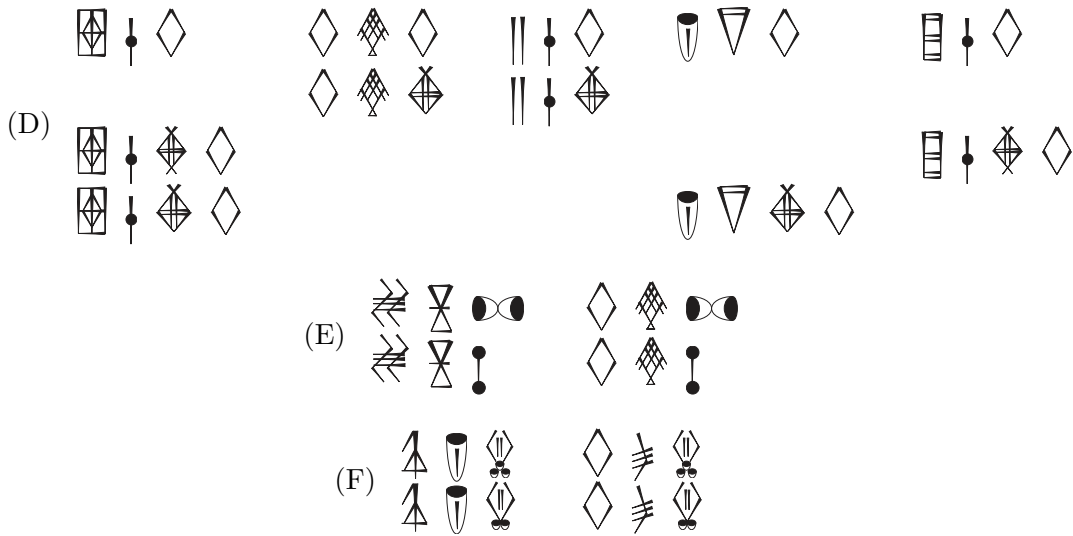
We may automate Kober's technique by compiling, for every  $n$ -gram in a corpus, a list of candidate suffixes up to some maximum length, and looking for sets of  $n$ -grams whose suffix sets share some non-empty intersection. Applying this technique to proto-Elamite indeed reveals alternations which are at least superficially similar to those identified in Linear B, for example:



Care must be taken, however: in Kober's own words, "in any language [...] a certain number of words can be found that have many signs in common and still are not related—e.g., in English, the pairs "heavy" and "heaven" [...] are not related, although a careless alien might conclude that they showed suffixal [...] inflection" (Kober 1945: 144). In proto-Elamite the alien faces the additional confound that word boundaries are not marked, so candidate affixes must be distinguished from the beginnings of freestanding words. This is precisely the pattern exhibited in the above "paradigm" from proto-Elamite, which actually shows an alternation between different object signs in parallel contexts.

There exist other candidate paradigms which appear more promising, however. We illustrate a number of these below:

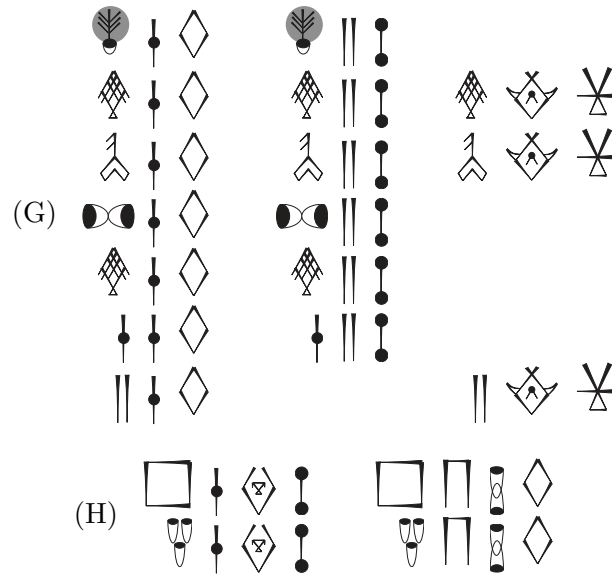




In these examples, the suffixes are formed exclusively from signs in the proposed syllabary, and the same is true for nearly all of the signs in the stems (with the possible exception of M131  $\nabla$  and M146  $\boxplus$ , though the former is understood to be syllabic in the complex grapheme M131+M388  $\nabla$ ). More than this, there is significant overlap between the candidate suffixes identified here by Alice Kober’s method, and those proposed in Section 5.2 based on our partial ordering of signs and analysis of median positions within entries. Given that Kober’s method has already been vindicated in the decipherment of Linear B, this increases our confidence that this represents a legitimate signal for some kind of internal structure to the so-called syllable sequences in proto-Elamite, whether that be true inflection or another superficially similar process. It is noteworthy that most of these alternations involve relatively few distinct stems and suffixes, given that true affixal morphology should be expected to be productive and therefore to produce larger paradigms than we observe here. This supports the view that these paradigms may reflect internal structure within proto-Elamite syllable sequences without representing true morphology.

Even if these paradigms do not represent morphology, the parallels which they highlight remain useful as evidence of a functional equivalence between certain sign variants. For example, in paradigm (B), the first and third columns differ in that the first sign of one is a tilde-variant of the first sign in the other. The fact that both signs occur in identical contexts in this paradigm suggests that they are underlyingly the same character, despite minor graphical differences in these surface forms. On a more tentative note, the visual similarity between M096  $\nabla$  in paradigm (A) and M101  $\nabla$  in paradigm (B) may suggest a similar equivalence, given that both appear to alternate with M218  $\diamond$ , though in this case the preceding stems are not shared and so the equivalence is less certain. In paradigm (D), the equivalence between M263  $\boxplus$  and M263~1  $\boxplus$  is already acknowledged by the use of a numbered, rather than lettered, variant label.

Kober applied her method to the discovery of suffixes, but in an age where the search process can be trivially automated it is straightforward to extend her technique to identify alternations involving possible prefixes, for example:



In paradigm (H), Kober’s technique has recovered the same pairs of strings used to argue against the treatment of M388 as a true determinative in Section 5.2.

The construction of these paradigms is computationally intensive, involving pairwise comparisons between affix sets for all candidate  $n$ -grams. For this reason the examples in this work are limited to stems and affixes of length at most 3. We publish other common alternations within these length bounds in Appendix A.10, and will release the code by which these paradigms were derived to enable future explorations with different stem and affix lengths by interested parties. We leave further interpretation of these paradigms to future work, which can incorporate linguistic insights beyond the scope of a computer science thesis.

## Part V

# Summary

## Chapter 12

# Conclusion

By now it will no doubt be evident to the reader how uniquely complex a dataset is proto-Elamite, and how diverse the challenges it presents. A relatively small corpus by computer science standards, exhibiting many hundreds of often vanishingly rare sign shapes, multi-way ambiguous notations, and uncertain word and phrase boundaries, without even the guarantee of underlying linguistic content: such are the hurdles which must be overcome if this corpus is to be understood in depth. And yet, with tools furnished by natural language processing and computational linguistics, we have managed to undertake the first ever large-scale computational analyses of this data, and have demonstrated how eminently surmountable these hurdles can be when they are approached from the correct angle.

We began by taking a critical eye to questions of glottography versus semasiography, and sought to clarify just how language-like this data really is. By measuring the entropy of  $n$ -gram distributions (Section 5.1), we demonstrated that the ordering of character tokens is less random than in the proto-cuneiform accounts, but more random than later cuneiform prose. When we subsequently sought for a “grammar” governing entry construction (Section 5.2), we found clear evidence for signs preferring to occupy the beginning or end of an entry, with weaker trends governing the middle. Together these results paint the picture of a script with weak or incipient word or character ordering rules, which seems to occupy a transitional space between the more obviously non-linguistic proto-cuneiform script and the more obviously linguistic cuneiform.

We have also identified other areas where the script is more similar to later writing systems. Zipf’s Law provides a weak test which could help to rule out linguistic content, but when we examined the proto-Elamite character frequencies we found that they fell within the expected bounds for representations of language (Section 4.1). When we applied an LDA topic model to this corpus (Section 4.4), we found that it produced interpretable results, and even inspired hypotheses about headers and seal impressions which proved fruitful in followup studies. These results suggest that the script is sufficiently language-like to be studied through the lens of natural language processing and computational linguistics, even if it is not a glottographic writing system in the strictest sense. Moreover, these and

other exploratory methods (Section 4.3) successfully and independently replicated numerous observations from prior manual work, allaying fears that computer models might be hindered by a lack of historical or other domain knowledge.

With these results in hand, we proceeded to more focused studies of interesting and unique features of the script, beginning with the class of signs known as complex graphemes (Chapter 6). We developed an approach for multimodal, image-in-context language modeling which partially overcomes the rare word problem by adding sign images as a model input, in addition to or in replacement of a typical embedding lookup table. Using the representations learned by this and other models, we argued that some complex graphemes appear to have semantically compositional meanings, so that the decipherment of these signs will be incidental to decipherment of their components. We also used pairing consistency scores and four-part analogy to argue that these signs share the general meaning of their outermost component, with the inner part(s) playing a specificatory role. Finally, we demonstrated for the first time the existence of an apparently inviolable grammar or hierarchy governing the construction of these signs, which must somehow be accounted for as meanings are proposed or refined.

We next replicated the discovery of headers (Chapter 7) by demonstrating that this structure is salient to unsupervised sequence models. We also argued from two separate angles (HMM state sequences and transformer self-attention) that two- or three-sign headers are much more common than previously believed. Building on an intuition from our earlier LDA explorations, we investigated the link between headers, seals, and other features of a document to reveal a variety of previously undetected correlations. Taken together, these correlations suggest that the first sign of a document looks outward, conveying extra-textual information about the surrounding administrative context, while the second looks inward and correlates with features of the text itself.

Our study of numeral notations (Chapter 8) contributed two techniques for disambiguating the oft-opaque numeration conventions employed by this script. The first of these techniques allowed us to curate a test set for a new numeral disambiguation task, while the second solved this task with high accuracy using a highly interpretable bootstrap model and a novel approach to cautious rule selection. This allowed us to update the corpus with new annotations recording the known or predicted magnitude of each numeral in a format that is understandable to modern readers. These annotations, in turn, enabled focused studies of particular signs and texts, as well as new visualizations which convey clear qualitative information about numeral distributions without assuming any particular disambiguation. Given that every known text is an administrative “spreadsheet” with extensive numeral notations, we believe that these results, and future work which expands on them, will prove to be among the most significant contributions of our efforts in the long run.

The true size of the sign list remains one of the murkiest aspects of the proto-Elamite script, as does the nature of the relationship between signs with similar, but not identical,

shapes. To advance our understanding in these areas, we revisited our earlier attempts at multimodal language modeling and developed a family of VAE-inspired, *image-only* language models (Chapter 9). With these models, we were at last able to completely distance our analyses from the working sign names, and could thus explore results which were less prone to potential label bias. On a script-recovery task, we demonstrated that our models were competitive with existing architectures that required on the order of *50 times as many parameters*. In cases where experts remain agnostic about the relationship between visually-similar signs, the output from our models supported the hypothesis that some of those sign pairs represent the same underlying character. In another case, we showed how the output from this model highlighted an uncommented-on parallel between signs which spurred new hypotheses about their possible meanings and deepened our understanding of the signs in question.

If proto-Elamite should prove to have a truly linguistic component, cognate alignment models are one way by which this fact could be discovered and exploited. However, existing approaches to cognate detection are either very slow or rely on clean data, making them difficult to apply directly to proto-Elamite. We outlined (Chapter 10) a new approach to cognate detection which learns a character level alignment between two scripts by adapting a monolingual language model, and uses edit distances derived from these alignments to identify candidate cognate word pairs. This technique offered strong top- $k$  performance on a noisy Ugaritic-Old Hebrew dataset while also requiring just a fraction as much training time as the current state-of-the-art.

We concluded with a brief discussion of results that were less technologically novel than our main results, but which nonetheless contributed useful context or gave further support to our earlier results. Perhaps the most intriguing of these was the application of Alice Kober’s test for inflectional morphology to this dataset (Section 11.2), and our extension of this technique to look for prefixes rather than merely suffixes. Here we found additional support for the novel suggestion that some name-like strings possess affix-like appendages (but despite this terminology we repeat the caution that these may not represent true morphosyntactic affixation).

We have provided additional descriptive statistics throughout this work, to provide a rather general survey of the corpus from a more quantitative perspective than is typical of existing references. We have also demonstrated how the same or similar patterns recur across different types of analysis, which gives us confidence that our results reflect genuine signals from the corpus. This is crucially important as the script remains only partially deciphered (though hopefully less so as a result of this work), and the consequent lack of a ground truth makes much of this work challenging to evaluate except by way of internal consistency or manual assessment by domain experts. On the latter point, we reiterate that every step of this work was performed in direct collaboration, or following extensive consultation, with leading domain experts. Our contributions reflect attempts to directly answer questions

of interest to those stakeholders using the tools afforded by natural language processing, attempts which we understand to have been generally successful based on their feedback. This collaboration has been equally fruitful from a technological perspective, having led to the development of novel models and analytic techniques published at major NLP venues.

We look forward to continued work on this corpus, both that which builds on our results and that which tackles new questions which we have not yet considered. As the first authors to have applied computational techniques to this corpus, we have only been able to scratch the surface of possible inquiries, and there remain desperately many topics worthy of eventual focus. These include studies of individual number systems, particularly marginal systems like B#, and of the distributions associated with specific object signs; an extension of our sequence modeling analysis to look for “subheaders”, which may delimit subsections in very large tablets; extensions of our techniques to proto-cuneiform, which remains opaque despite having known descendant scripts (this will be an interesting technical challenge owing to the lack of linear sign order, and may require the adaptation of more techniques from computer vision); comparisons between parallel categories in proto-Elamite and proto-cuneiform, for instance to establish whether equivalent counted objects are associated with equivalent numeral distributions, or to determine whether colophons exhibit correlations with extra-textual features similarly to headers; direct alignments between proto-Elamite sign sequences and character, word, or sound sequences from other languages using techniques like the cognate alignment model which we have proposed; comparisons with glyptic traditions to further clarify the connection between seal imagery and tablet headers; and undoubtedly many more besides.

Our image-only, VAE-inspired language models also have potential applications to the study of other scripts, such as the evolution of character shapes in early Chinese bronze inscriptions (Anonymous, pers. comm.). We are also aware of potential interest in applying a similar model to the study of stamp impressions in Han Dynasty funerary bricks (Anonymous, pers. comm.). These bricks are stamped with decorative patterns, and subtle differences between the shapes on different bricks suggest that they were prepared using *distinct* stamps showing the same general image. Clustering images of stamp impressions to create a “stamplist” roughly parallels the task of clustering tokens to infer a sign list, and may help historians to more easily determine when two bricks come from the same workshop. Thus the models considered in our work promise to generalize to distinct contexts, and are not limited to the decipherment applications which have thus far been our focus.

To conclude, this work has served to deepen modern understandings of an important piece of humanity’s shared cultural heritage, and has shed new light on one of the earliest known forms of writing or proto-writing. We earnestly hope that this work will inspire renewed study of this uniquely interesting and challenging corpus, and will spur greater collaboration between computer scientists and experts from other domains by demonstrating how such collaborations serve to advance both fields.



# Bibliography

- Steven P. Abney. 2002. Bootstrapping. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 360–367. ACL.
- Nada Aldarrab and Jonathan May. 2021. Can sequence-to-sequence models crack substitution ciphers? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7226–7235, Online. Association for Computational Linguistics.
- Pierre Amiet. 1972. *Glyptique susienne des origines à l'époque des perses achéménides*, volume 43 of *Mémoires de la Délégation Archéologique en Iran*. Paris: Librairie orientaliste Paul Geuthner.
- Mikel Artetxe and Holger Schwenk. 2018. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *arXiv:1812.10464v2*.
- Tilman Becker, Aravind K. Joshi, and Owen Rambow. 1991. Long-distance scrambling and Tree Adjoining Grammars. In *Fifth Conference of the European Chapter of the Association for Computational Linguistics*, Berlin, Germany. Association for Computational Linguistics.
- Taylor Berg-Kirkpatrick and Dan Klein. 2011. Simple effective decipherment via combinatorial optimization. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 313–321, Edinburgh, Scotland, UK. Association for Computational Linguistics.
- Taylor Berg-Kirkpatrick and Dan Klein. 2013. Decipherment with a million random restarts. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 874–878.
- Taylor Berg-Kirkpatrick and Daniel Spokoyny. 2020. An empirical investigation of contextualized number prediction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4754–4764, Online. Association for Computational Linguistics.
- Wicher Bergsma. 2013. A bias-correction for cramér’s v and tschuprow’s t. *Journal of the Korean Statistical Society*, 42(3):323–328.
- David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

- Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008.
- Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.
- Logan Born and Kate Kelley. 2021. A quantitative analysis of proto-cuneiform sign use in archaic tribute. *Cuneiform Digital Library Bulletin*, 2021:6.
- Logan Born, Kate Kelley, Nishant Kambhatla, Carolyn Chen, and Anoop Sarkar. 2019. Sign clustering and topic extraction in Proto-Elamite. In *Proceedings of the 3rd Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 122–132, Minneapolis, USA. Association for Computational Linguistics.
- Logan Born, Kate Kelley, M. Willis Monroe, and Anoop Sarkar. 2022. Sequence models for document structure identification in an undeciphered script. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Abu Dhabi, UAE. Association for Computational Linguistics.
- Logan Born, Kathryn Kelley, M. Willis Monroe, and Anoop Sarkar. 2021. Compositionality of complex graphemes in the undeciphered Proto-Elamite script using image and text embedding models. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4136–4146, Online. Association for Computational Linguistics.
- Logan Born, M. Willis Monroe, Kathryn Kelley, and Anoop Sarkar. 2023a. Disambiguating numeral sequences to decipher ancient accounting corpora. In *Proceedings of the Workshop on Computation and Written Language (CAWL 2023)*, pages 71–81, Toronto, Canada. Association for Computational Linguistics.
- Logan Born, M. Willis Monroe, Kathryn Kelley, and Anoop Sarkar. 2023b. Learning the character inventories of undeciphered scripts using unsupervised deep clustering. In *Proceedings of the Workshop on Computation and Written Language (CAWL 2023)*, pages 92–104, Toronto, Canada. Association for Computational Linguistics.
- William Brice. 1963. A comparison of the account tablets of Susa in the proto-Elamite script with those of Hagia Triada in Linear A. *Kadmos*, 2(1):27–38.
- Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. 2018. Deep clustering for unsupervised learning of visual features. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIV*, volume 11218 of *Lecture Notes in Computer Science*, pages 139–156. Springer.
- Robert L. Cave and Lee P. Neuwirth. 1980. Hidden markov models for english. In J. D. Ferguson, editor, *Hidden Markov Models for Speech*. IDA-CRD, Princeton, NJ.
- John Chadwick. 1958. *The Decipherment of Linear B*. Vintage Books. Cambridge University Press.

- Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.
- Jacob Cohen. 1988. *Statistical Power Analysis for the Behavioral Sciences*, 2nd edition. Routledge.
- Michael Collins and Yoram Singer. 1999. Unsupervised models for named entity classification. In *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.
- Alexis Conneau, Guillaume Lample, Marc’Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.
- Michele Corazza, Fabio Tamburini, Miguel Valério, and Silvia Ferrara. 2022a. Contextual unsupervised clustering of signs for ancient writing systems. In *Proceedings of the Second Workshop on Language Technologies for Historical and Ancient Languages*, pages 84–93, Marseille, France. European Language Resources Association.
- Michele Corazza, Fabio Tamburini, Miguel Valério, and Silvia Ferrara. 2022b. Unsupervised deep learning supports reclassification of Bronze age cyriot writing system. *PLOS One*, 17(7):1–22.
- Silvio Cordeiro, Carlos Ramisch, Marco Idiart, and Aline Villavicencio. 2016. Predicting the compositionality of nominal compounds: Giving word embeddings a hard time. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1986–1997, Berlin, Germany. Association for Computational Linguistics.
- Harald Cramér and Herman H. Goldstine. 1946. *Mathematical methods of statistics*. Princeton: Princeton University Press.
- Jacob Dahl. 2005a. Animal husbandry in Susa during the proto-Elamite period. *Studi Micenei ed Egeo-Anatolici*, 47:81–134.
- Jacob Dahl. 2005b. Complex graphemes in proto-elamite. *Cuneiform Digital Library Journal*, 2005:3.
- Jacob Dahl. 2019. *Tablettes et fragments proto-élamites – Proto-Elamite tablets and fragments*, volume 32 of *Texts cunéiformes du Louvre*. Paris: Éditions Khéops.
- Jacob Dahl. Unpublished. Unpublished proto-Elamite sign list with notes including speculated meanings.
- Jacob Dahl, Laura Hawkins, and Kate Kelley. 2018. Labour administration in proto-Elamite Iran. In A. Garcia-Ventura, editor, *What’s in a Name? Terminology related to the Work Force and Job Categories in the Ancient Near East*, *Alt Orient und Altes Testament* 440, pages 15–44. Münster: Ugarit Verlag.
- Jacob L. Dahl. 2002. Proto-Elamite sign frequencies. *Cuneiform Digital Library Bulletin*, 2002/1.
- Jacob L. Dahl. 2005c. Complex graphemes in proto-elamite. *Cuneiform digital library journal*, 4(3).

- Jacob L. Dahl. 2009. Early writing in iran, a reappraisal. *Iran*, 47:23–31.
- Jacob L. Dahl. 2016. The production and storage of food in early Iran. In M.B. D’Anna, C. Jauß, and J.C. Johnson, editors, *Food and Urbanisation. Material and Textual Perspectives on Alimentary Practice in Early Mesopotamia*, volume 37, pages 45–50. Gangemi Editore.
- Peter Damerow. 1999. The origins of writing as a problem of historical epistemology. *Cuneiform Digital Library Journal*, 2006:1.
- Peter Damerow and Robert K. Englund. 1989. *The proto-Elamite texts from Tepe Yahya*, volume 39 of *American School of Prehistoric Research: Bulletin*. Peabody Museum, Cambridge, Massachusetts.
- Teófilo Emídio de Campos, Bodla Rakesh Babu, and Manik Varma. 2009. Character recognition in natural images. In *VISAPP 2009 - Proceedings of the Fourth International Conference on Computer Vision Theory and Applications, Lisboa, Portugal, February 5-8, 2009 - Volume 2*, pages 273–280. INSTICC Press.
- Roland de Mecquenem. 1949. *Épigraphie proto-élamite*, volume 31 of *Mémoires de la Mission Archéologique en Iran*. Paris: Presses Universitaires de France.
- Roland de Mecquenem. 1956. Notes proto-élamites. *Revue d’Assyriologie et d’archéologie orientale*, 50(4).
- Tobias Dencker, Pablo Klinkisch, Stefan M. Maul, and Björn Ommer. 2020. Deep learning of cuneiform sign detection with weak supervision using transliteration alignment. *PLOS One*.
- Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. IEEE.
- Leon Derczynski and Sean Chester. 2016. Generalised Brown clustering and roll-up feature generation. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI 2016*, pages 1533–1539. AAAI Press.
- François Desset. 2012. *Premières écritures iraniennes. Les systèmes proto-élamite et élamite linéaire*. Università degli studi di Napoli “L’Orientale”, Dipartimento Asia Africa e Mediterraneo.
- François Desset. 2016. Proto-Elamite writing in Iran. *Archéo-nil. Revue de la société pour l’étude des cultures prépharaoniques de la vallée du Nil*, 26:67–104.
- François Desset, Kambiz Tabibzadeh, Matthieu Kervran, Gian Pietro Basello, and Gianni Marchesi. 2022. The decipherment of linear Elamite writing. *Zeitschrift für Assyriologie und vorderasiatische Archäologie*, 112(1):11–60.
- Mary D’Imperio. 1979. An application of PTAH to the Voynich Manuscript (U). In *National Security Agency Technical Journal*, volume 24, pages 65–91.

- Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.
- Robert K. Englund. 2001. Grain accounting practices in archaic Mesopotamia. In J. Høyrup and Peter Damerow, editors, *Changing Views on Ancient Near Eastern Mathematics*, pages 1–35.
- Robert K. Englund. 2004. The state of decipherment of proto-Elamite. *The First Writing: Script Invention as History and Process*, pages 100–149.
- Robert K. Englund. 2011. Accounting in proto-cuneiform. In K. Radner and E. Robson, editors, *The Oxford Handbook of Cuneiform Culture*, pages 32–50.
- Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD’96, page 226–231. AAAI Press.
- Steve Farmer, Richard Sproat, and Michael Witzel. 2004. The collapse of the indus-script thesis: The myth of a literate harappan civilization. *Electronic Journal of Vedic Studies*, 11(2).
- Louis Fournier, Emmanuel Dupoux, and Ewan Dunbar. 2020. Analogies minus analogy test: measuring regularities in word embeddings. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 365–375, Online. Association for Computational Linguistics.
- Jöran Friberg. 1978. *The Third Millennium Roots of Babylonian Mathematics. I. A Method for the Decipherment, through Mathematical and Metrological Analysis, of Proto-Sumerian and Proto-Elamite Semi-Pictographic Inscriptions*. Department of Mathematics, Chalmers University of Technology.
- I. J. Gelb and R. M. Whiting. 1975. Methods of decipherment. *Journal of the Royal Asiatic Society of Great Britain and Ireland*, 2:95–104.
- Borjan Geshkovski, Cyril Letrouit, Yury Polyanskiy, and Philippe Rigollet. 2023. The emergence of clusters in self-attention dynamics.
- Shai Gordin, Gai Gutherz, Ariel Elazary, Avital Romich, Enrique Jiménez, Jonathan Berant, and Yoram Cohen. 2020. Reading Akkadian cuneiform using natural language processing. *PLOS One*.
- Kyle Gorman and Richard Sproat. 2023. Myths about writing systems in speech & language technology. In *Proceedings of the Workshop on Computation and Written Language (CAWL 2023)*, pages 1–5, Toronto, Canada. Association for Computational Linguistics.
- Gholamreza Haffari and Anoop Sarkar. 2007. Analysis of semi-supervised learning with the Yarowsky algorithm. In *Uncertainty in Artificial Intelligence (UAI 2007)*, pages 159 – 166. AUAI Press. Conference in Uncertainty in Artificial Intelligence 2007, UAI 2007 ; Conference date: 19-07-2007 Through 22-07-2007.

- Matthias Hartung, Fabian Kaupmann, Soufian Jebbara, and Philipp Cimiano. 2017. Learning compositionality functions on word embeddings for modelling attribute meaning in adjective-noun phrases. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 54–64, Valencia, Spain. Association for Computational Linguistics.
- Laura Hawkins. 2015. A new edition of the proto-Elamite text MDP 17, 112. *Cuneiform Digital Library Journal*, 2015:001.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.
- Geoffrey E Hinton and Sam Roweis. 2002. Stochastic neighbor embedding. In *Advances in Neural Information Processing Systems*, volume 15. MIT Press.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.
- Timo Homburg and Christian Chiarcos. 2016. Akkadian word segmentation. In *Tenth International Conference on Language Resource Evaluation (LREC 2016)*, pages 4067–4074.
- Mehrdad Honarkhah and Jef Caers. 2010. Stochastic simulation of patterns using distance-based pattern modeling. *Mathematical Geosciences*, 42:487–517.
- Alfred Inselberg. 1985. The plane with parallel coordinates. *The Visual Computer*, 1:69–91.
- Nishant Kambhatla, Logan Born, and Anoop Sarkar. 2023. Decipherment as regression: Solving historical substitution ciphers by learning symbol recurrence relations. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2136–2152, Dubrovnik, Croatia. Association for Computational Linguistics.
- Nishant Kambhatla, Anahita Mansouri Bigvand, and Anoop Sarkar. 2018. Decipherment of substitution ciphers with neural language models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 869–874, Brussels, Belgium. Association for Computational Linguistics.
- Kate Kelley. 2018. *Gender, age and labour organization in the earlier texts from Mesopotamia and Iran (c. 3300–2900 BC)*. Doctoral dissertation, Oxford University.
- Kathryn Kelley, Logan Born, M. Willis Monroe, and Anoop Sarkar. 2022a. Image-aware language modeling for proto-Elamite. *Lingue e linguaggio, Rivista semestrale*, 2/2022:261–294.
- Kathryn Kelley, Logan Born, M. Willis Monroe, and Anoop Sarkar. 2022b. On newly proposed proto-Elamite sign values. *Iranica Antiqua*, 57:1–25.
- Young-Bum Kim and Benjamin Snyder. 2013. Unsupervised consonant-vowel prediction over hundreds of languages. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1527–1536, Sofia, Bulgaria. Association for Computational Linguistics.

- Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.
- Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.
- Kevin Knight, Beáta Megyesi, and Christiane Schaefer. 2011. The copiale cipher. In *Proceedings of the 4th Workshop on Building and Using Comparable Corpora: Comparable Corpora and the Web*, pages 2–9, Portland, Oregon. Association for Computational Linguistics.
- Kevin Knight, Anish Nair, Nishit Rathod, and Kenji Yamada. 2006. Unsupervised analysis for decipherment problems. In *Proceedings of the COLING/ACL on Main Conference Poster Sessions, COLING-ACL '06*, pages 499–506, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Kevin Knight and Kenji Yamada. 1999. A computational approach to deciphering unknown scripts. In *Proceedings of the ACL Workshop on Unsupervised Learning in Natural Language Processing*.
- Alice E. Kober. 1945. Evidence of inflection in the “chariot” tablets from Knossos. *American Journal of Archaeology*, 49(2):143–151.
- V. I. Levenshtein. 1966. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10:707.
- Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312.
- Frederick Liu, Han Lu, Chieh Lo, and Graham Neubig. 2017. Learning character-level compositionality with visual features. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2059–2068, Vancouver, Canada. Association for Computational Linguistics.
- Jiaming Luo, Yuan Cao, and Regina Barzilay. 2019. Neural decipherment via minimum-cost flow: From Ugaritic to Linear B. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3146–3155, Florence, Italy. Association for Computational Linguistics.
- Jiancheng Lyu, Shuai Zhang, Yingyong Qi, and Jack Xin. 2020. Autosshufflenet: Learning permutation matrices via an exact lipschitz continuous penalty in deep convolutional neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '20*, page 608–616, New York, NY, USA. Association for Computing Machinery.
- H. B. Mann and D. R. Whitney. 1947. On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics*, 18(1):50–60.

- Mika Mäntylä, Maelick Claes, and Umar Farooq. 2018. Measuring LDA topic stability from clusters of replicated runs. In *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, ESEM '18*, New York, NY, USA. ACM.
- Piero Meriggi. 1971. *La scrittura e il contenuto dei testi*, volume I of *La scrittura proto-Elamica*. Rome: Accademia Nazionale dei Lincei.
- Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *CoRR*, abs/1609.07843.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *International Conference on Learning Representations (ICLR)*.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, NIPS'13*, page 3111–3119, Red Hook, NY, USA. Curran Associates Inc.
- Isabel Moreno-Sánchez, Francesc Font-Clos, and Àlvaro Corral. 2016. Large-scale analysis of Zipf’s law in English texts. *PloS One*, 11(1).
- Aakanksha Naik, Abhilasha Ravichander, Carolyn Rose, and Eduard Hovy. 2019. Exploring numeracy in word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3374–3380, Florence, Italy. Association for Computational Linguistics.
- Hans J. Nissen, Peter Damerow, and Robert K. Englund. 1993. *Archaic Bookkeeping: Early Writing and Techniques of Economic Administration in the Ancient Near East*. University of Chicago Press.
- Émilie Pagé-Perron, Maria Sukhareva, Ilya Khait, and Christian Chiarcos. 2017. Machine translation and automated analysis of the Sumerian language. In *Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature, ACL*, pages 10–16. Association for Computational Linguistics.
- Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.
- Mladen Popović, Maruf A. Dhali, and Lambert Schomaker. 2021. Artificial intelligence based writer identification generates new evidence for the unknown scribes of the Dead Sea scrolls exemplified by the great Isaiah scroll (1QIsa<sup>a</sup>). *PLOS ONE*, 16(4):1–28.
- L.R. Rabiner. 1989. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.
- Sujith Ravi and Kevin Knight. 2009. Learning phoneme mappings for transliteration without parallel data. In *Proceedings of human language technologies: The 2009 annual conference of the north american chapter of the association for computational linguistics*, pages 37–45.



- Sravana Reddy and Kevin Knight. 2011. What we know about the Voynich manuscript. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 78–86.
- A. Robinson. 2009. *Lost Languages: The Enigma of the World’s Undeciphered Scripts*. Thames & Hudson.
- Andrew Rosenberg and Julia Hirschberg. 2007. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 410–420, Prague, Czech Republic. Association for Computational Linguistics.
- Peter J. Rousseeuw. 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65.
- Bahar Salehi, Paul Cook, and Timothy Baldwin. 2015. A word embedding approach to predicting the compositionality of multiword expressions. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 977–983, Denver, Colorado. Association for Computational Linguistics.
- Vincent Scheil. 1900. *Textes élamites-sémitiques*, volume 2 of *Mémoires de la Délégation en Perse*. Paris: Ernest Leroux.
- Vincent Scheil. 1905. *Textes élamites-sémitiques*, volume 6 of *Mémoires de la Délégation en Perse*. Paris: Ernest Leroux.
- Vincent Scheil. 1923. *Textes de comptabilité proto-élamites*, volume 17 of *Mémoires de la Délégation en Perse*. Paris: Ernest Leroux.
- Vincent Scheil. 1935. *Textes de comptabilité proto-élamites*, volume 26 of *Mémoires de la Délégation en Perse*. Paris: Librairie Ernest Leroux.
- M. Schuster and K.K. Paliwal. 1997. Bidirectional recurrent neural networks. *Trans. Sig. Proc.*, 45(11):2673–2681.
- Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2022. Language models are multilingual chain-of-thought reasoners. *CoRR*, abs/2210.03057.
- Carson Sievert and Kenneth Shirley. 2014. LDAvis: A method for visualizing and interpreting topics. In *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*, pages 63–70. Association for Computational Linguistics.
- Benjamin Snyder, Regina Barzilay, and Kevin Knight. 2010. A statistical model for lost language decipherment. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1048–1057. Association for Computational Linguistics.

- Georgios Spithourakis and Sebastian Riedel. 2018. Numeracy for language models: Evaluating and improving their ability to predict numbers. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2104–2115, Melbourne, Australia. Association for Computational Linguistics.
- Richard Sproat. 2006. *A Computational Theory of Writing Systems (Studies in Natural Language Processing)*. Cambridge University Press, USA.
- Nikita Srivatsan, Jason Vega, Christina Skelton, and Taylor Berg-Kirkpatrick. 2021. Neural representation learning for scribal hands of Linear B. In *Document Analysis and Recognition – ICDAR 2021 Workshops: Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part II*, page 325–338, Berlin, Heidelberg. Springer-Verlag.
- Catherine A Sugar and Gareth M James. 2003. Finding the number of clusters in a dataset. *Journal of the American Statistical Association*, 98(463):750–763.
- Chi Sun, Xipeng Qiu, and Xuanjing Huang. 2019. VCWE: visual character-enhanced word embeddings. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2710–2719. Association for Computational Linguistics.
- Dhanasekar Sundararaman, Shijing Si, Vivek Subramanian, Guoyin Wang, Devamanyu Hazarika, and Lawrence Carin. 2020. Methods for numeracy-preserving word embeddings. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4742–4753, Online. Association for Computational Linguistics.
- Fabio Tamburini. 2023. Decipherment of lost ancient scripts as combinatorial optimisation using coupled simulated annealing. In *Proceedings of the Workshop on Computation and Written Language (CAWL 2023)*, pages 82–91, Toronto, Canada. Association for Computational Linguistics.
- Robert L. Thorndike. 1953. Who belongs in the family? *Psychometrika*, 18:267–276.
- Robert Tibshirani, Guenther Walther, and Trevor Hastie. 2002. Estimating the Number of Clusters in a Data Set Via the Gap Statistic. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 63(2):411–423.
- Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).
- Ke Tran. 2020. From english to foreign languages: Transferring pre-trained language models.
- Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

- Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. 2008. Extracting and composing robust features with denoising autoencoders. In *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, volume 307 of *ACM International Conference Proceeding Series*, pages 1096–1103. ACM.
- Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.*, 11:3371–3408.
- K. Wagensonner. 2015. Vessels and other containers for the storage of food according to the early lexical record. *Origini Preistoria e protostoria delle civiltà antiche*, XXXVII(1).
- Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. 2019. Do NLP models know numbers? probing numeracy in embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5307–5315, Hong Kong, China. Association for Computational Linguistics.
- Max Whitney and Anoop Sarkar. 2012. Bootstrapping via graph propagation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 620–628, Jeju Island, Korea. Association for Computational Linguistics.
- Qinzhao Wu, Qi Zhang, Zhongyu Wei, and Xuanjing Huang. 2021. Math word problem solving with explicit numerical values. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5859–5869, Online. Association for Computational Linguistics.
- Samuel Xavier-de-Souza, Johan A. Suykens, Joos Vandewalle, and Désiré Bolle. 2010. Coupled simulated annealing. *IEEE Trans Syst Man Cybern B Cybern*, 40(2):320–335.
- David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *33rd Annual Meeting of the Association for Computational Linguistics*, pages 189–196, Cambridge, Massachusetts, USA. Association for Computational Linguistics.
- Xusen Yin, Nada Aldarrab, Beáta Megyesi, and Kevin Knight. 2019. Decipherment of historical manuscript images. In *2019 International Conference on Document Analysis and Recognition, ICDAR 2019, Sydney, Australia, September 20-25, 2019*, pages 78–85. IEEE.
- Marcos Zampieri, Preslav Nakov, Shervin Malmasi, Nikola Ljubešić, Jörg Tiedemann, and Ahmed Ali, editors. 2019. *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*. Association for Computational Linguistics, Ann Arbor, Michigan.
- George K. Zipf. 1935. *The psycho-biology of language*. Houghton Mifflin.

# Appendix A

## Additional Figures

### A.1 Dendrograms

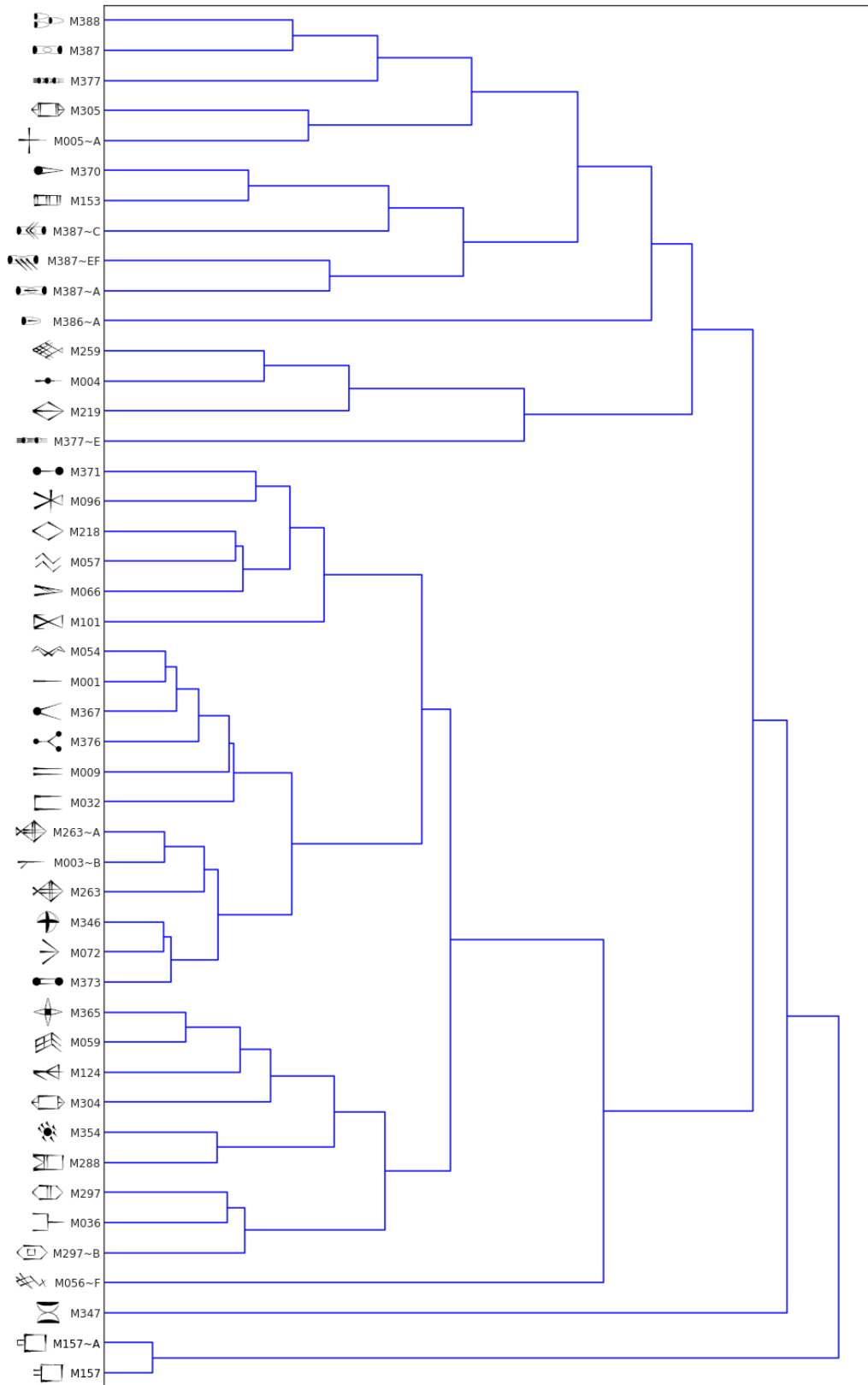


Figure A.1: Complete dendrogram for neighbor-based hierarchical clustering (Chapter 4), showing signs attested 50 times or more.

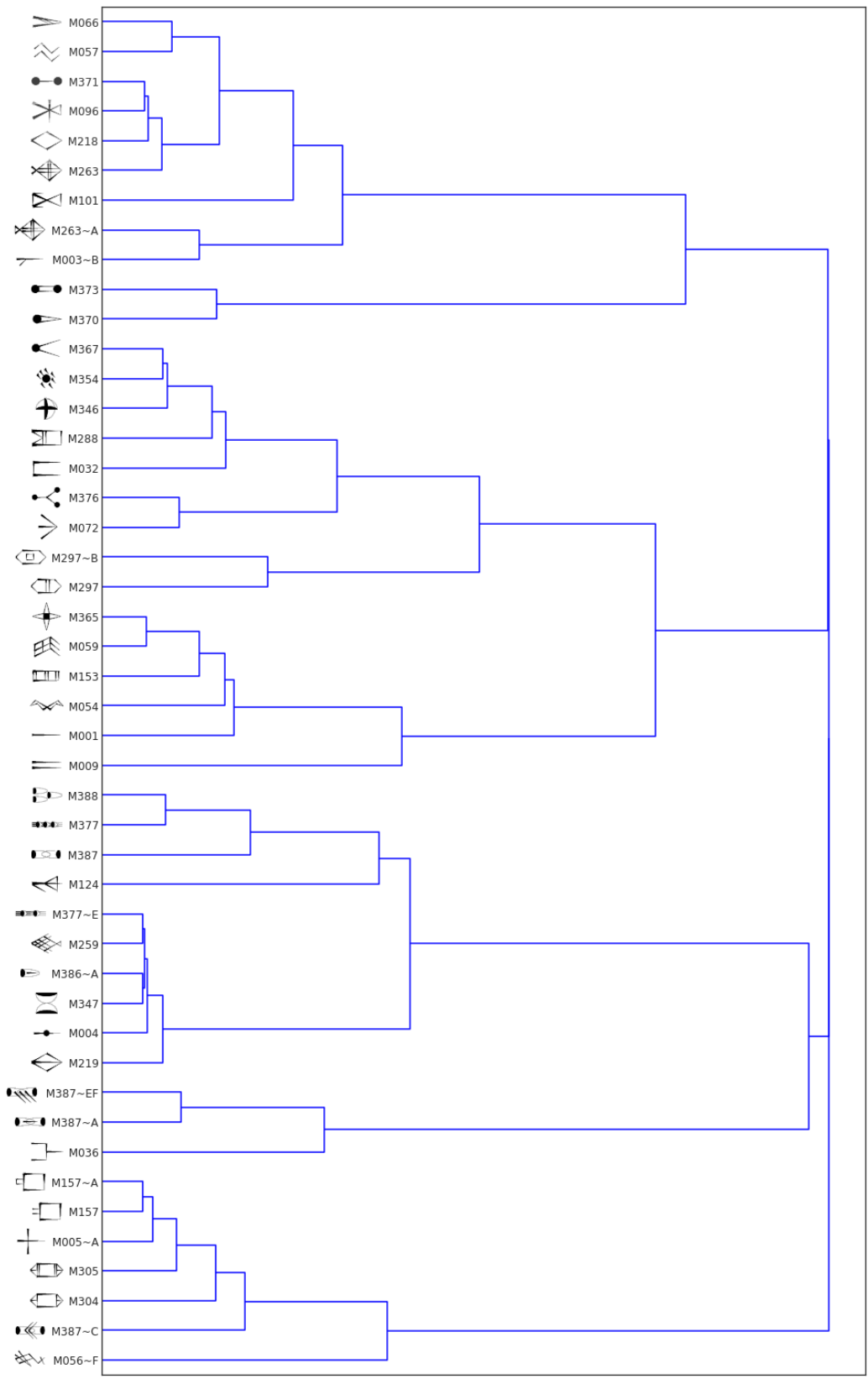


Figure A.2: Complete dendrogram for HMM-based hierarchical clustering (Chapter 4), showing signs attested 50 times or more.

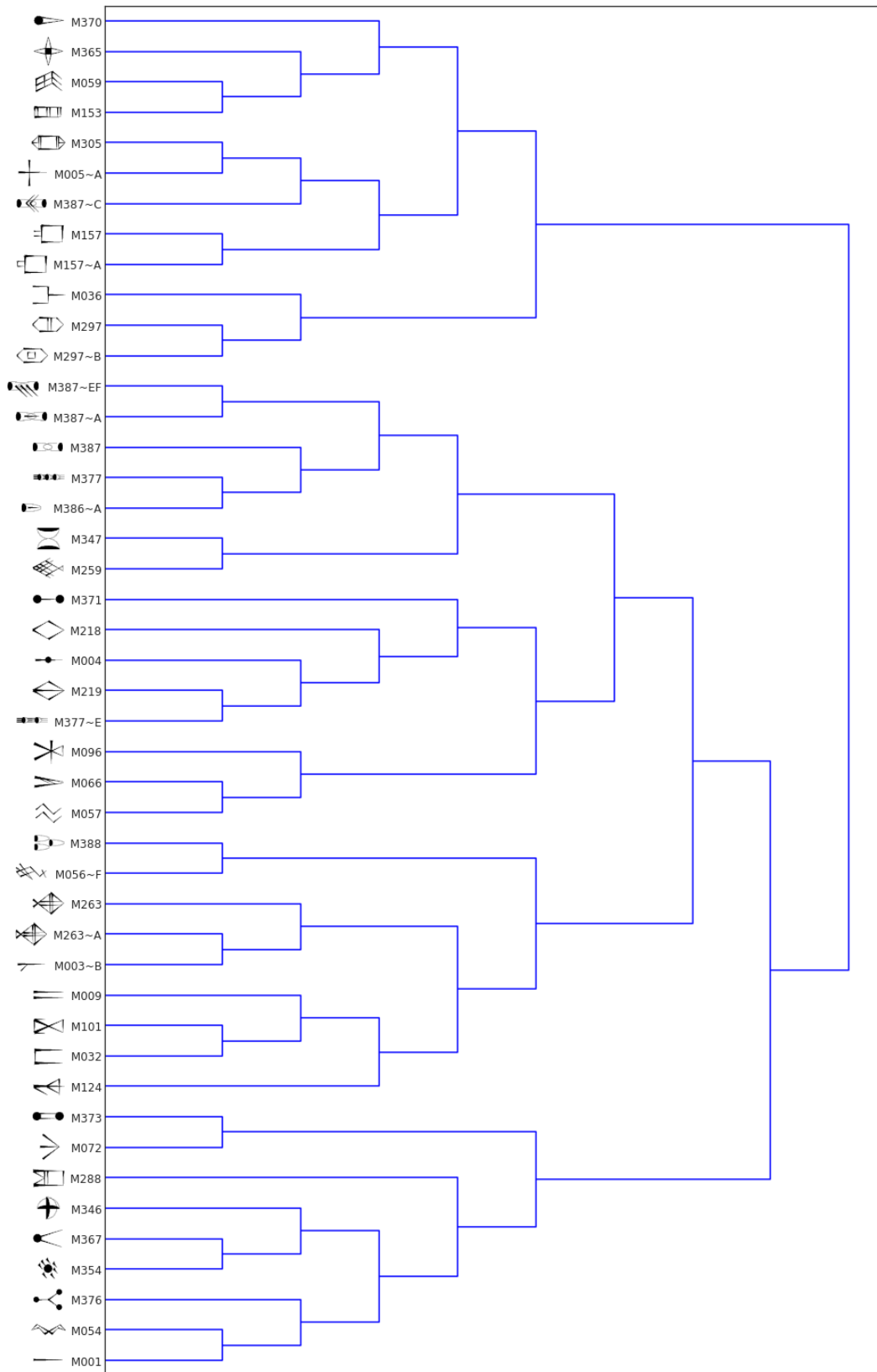


Figure A.3: Complete dendrogram for Brown clustering (Chapter 4), showing signs attested 50 times or more.

## A.2 LDA Topic Stability

This section reports the raw results from our topic stability analysis in Section 4.4. We cluster the topics from 30 random restarts of a 30-topic LDA model on the proto-Elamite corpus with numerals stripped out. For each cluster, we report the mean pairwise rank-biased overlap (Mäntylä et al. 2018) between topics in the cluster. We also count how frequently each sign occurs among the ten most predictive signs for the topics in the cluster; we list those which are most frequent, alongside the proportion of topics in the cluster in which that sign appears.

Mean Pairwise RBO: 0.845746

Term	Proportion
M106_M009	1.000
M009	1.000
M009_M206~G	1.000
M309~A	1.000
M106~A	1.000
M206~G	1.000
M102~E_M309~A	1.000
M102~E	0.967
M106	0.967
M260	0.567

Mean Pairwise RBO: 0.650776

Term	Proportion
M124	0.971
M370	0.941
M054	0.912
M072	0.853
M373	0.824
M376	0.794
M371	0.735
M288	0.559
M388	0.559
M003~B	0.559

Mean Pairwise RBO: 0.615730

Term	Proportion
M297	1.000
M263~A	1.000
M157	1.000
M157~A	0.667
M388	0.333
M341	0.333
M131~E	0.333
M175+M288	0.333
M265	0.333
M243~EE	0.333

Mean Pairwise RBO: 0.576265

Term	Proportion
M036	1.000
M387~EF	0.964
M266~B	0.929
M387~A	0.857
M263	0.786
M002	0.786
M387	0.429
M010~2	0.286
M263~B1	0.214
M297	0.179



Mean Pairwise RBO: 0.500974

<u>Term</u>	<u>Proportion</u>
M297	1.000
M388	0.938
M157	0.688
M380	0.500
M218	0.375
M218+M320	0.312
M226~C	0.250
M051~B	0.188
M323~I	0.125
M146~D	0.062

Mean Pairwise RBO: 0.485851

<u>Term</u>	<u>Proportion</u>
M054	1.000
M003~B	0.722
M072	0.722
M317	0.722
M373	0.722
M059	0.389
M388	0.278
M033	0.167
M269~B	0.056
M367~C	0.056

Mean Pairwise RBO: 0.465029

<u>Term</u>	<u>Proportion</u>
M218	0.936
M388	0.897
M371	0.859
M066	0.769
M096	0.718
M004	0.564
M057	0.385
M347	0.269
M377~E_M347	0.218
M387	0.218

Mean Pairwise RBO: 0.500370

<u>Term</u>	<u>Proportion</u>
M297	1.000
M388	0.900
M157	0.900
M218	0.600
M305	0.400
M066	0.400
M036+1(N30D)	0.300
M111~A	0.300
M380	0.300
M004	0.100

Mean Pairwise RBO: 0.474232

<u>Term</u>	<u>Proportion</u>
M288	1.000
M157	0.867
M203~C	0.467
M005~A	0.400
M391	0.400
M124	0.333
M106~2+M288	0.333
M136~B	0.300
M010~2	0.100
M217	0.100

Mean Pairwise RBO: 0.415863

<u>Term</u>	<u>Proportion</u>
M376	1.000
M149~A	0.633
M032	0.600
M157	0.600
M005	0.367
M136+X	0.300
M311~B	0.233
M149~A2	0.100
M383	0.033
M061	0.033

Mean Pairwise RBO: 0.406916

<u>Term</u>	<u>Proportion</u>
M367	0.962
M362	0.808
M362_M367	0.692
M006	0.462
M362_M367_M269~A_M269~2	0.385
M341~Q	0.385
M269~A_M269~2	0.346
M056	0.269
M106~2+M288	0.038
M377+M377	0.038

Mean Pairwise RBO: 0.359433

<u>Term</u>	<u>Proportion</u>
M346	1.000
M006	0.400
M346~A	0.200
M362~A	0.200
M362	0.133
M367~A	0.133
M006@G	0.133
M367~G	0.067
M346~D	0.067
M112	0.067

Mean Pairwise RBO: 0.354349

<u>Term</u>	<u>Proportion</u>
M354	1.000
M054	0.464
M265~F	0.179
M384~D	0.179
M157~A	0.179
M111~A	0.143
M346~A	0.071
M362~A	0.071
M323	0.071
M228~GA	0.036

Mean Pairwise RBO: 0.353889

<u>Term</u>	<u>Proportion</u>
M056~F	1.000
M288	0.727
M305	0.636
M075~G	0.545
M352~C	0.182
M222	0.091
M112	0.091
M038~B1	0.091
M206~FA	0.091
M080~B	0.091

Mean Pairwise RBO: 0.348825

<u>Term</u>	<u>Proportion</u>
M305	1.000
M038~BX	0.739
M388	0.609
M387~C	0.435
M131	0.348
M387~CA	0.348
M001	0.304
M387	0.261
M136+X	0.174
M203~C	0.043

Mean Pairwise RBO: 0.347237

<u>Term</u>	<u>Proportion</u>
M387	1.000
M081	0.522
M157	0.348
M263~B1	0.304
M297~B	0.217
M305	0.174
M038~A	0.087
M157~A	0.087
M136~C	0.043
M252~Q	0.043

Mean Pairwise RBO: 0.320113

<u>Term</u>	<u>Proportion</u>
M365	0.885
M054~I	0.769
M105~A	0.731
M003	0.692
M367	0.577
M006	0.538
M328~B	0.423
M136~I	0.308
M210	0.231
M080~A	0.038

Mean Pairwise RBO: 0.302056

<u>Term</u>	<u>Proportion</u>
M009	1.000
M371	0.200
M005~A	0.200
M384~AB	0.133
M125	0.133
M208	0.133
M380~B	0.067
M048~C	0.067
M036+1(N14)	0.067
M195+M038~A	0.067

Mean Pairwise RBO: 0.283433

<u>Term</u>	<u>Proportion</u>
M206~D	0.950
M157	0.700
M388	0.550
M223	0.300
M367~I	0.100
M351+X	0.100
M158~H	0.050
M112	0.050
M260~1+1(N24)	0.050
M103~2	0.050

Mean Pairwise RBO: 0.314599

<u>Term</u>	<u>Proportion</u>
M297~B	1.000
M157	0.708
M176	0.292
M044	0.083
M195	0.042
M069~A	0.042
M387_M069~A	0.042
M311~B	0.042
M146~D	0.042
M361~A	0.042

Mean Pairwise RBO: 0.286764

<u>Term</u>	<u>Proportion</u>
M157~A	1.000
M260~1	0.312
M327+M342	0.312
M005~A	0.188
M292	0.125
M305+M342	0.125
M038~A	0.062
M111~A	0.062
M264~A	0.062
M228~GA	0.062

Mean Pairwise RBO: 0.278930

<u>Term</u>	<u>Proportion</u>
M059	1.000
M325	0.200
M069~A	0.150
M387_M069~A	0.150
M288+1(N01)	0.050
M126	0.050
M367~E	0.050
M054+M365+M054~I	0.050
M054~C	0.050
M417~F	0.050

Mean Pairwise RBO: 0.278817

<u>Term</u>	<u>Proportion</u>
M032	1.000
M005	0.474
M391	0.316
M157	0.263
M376	0.211
M009	0.105
M327+X	0.105
M327+M059	0.053
M311~B	0.053
M180	0.053

Mean Pairwise RBO: 0.267829

<u>Term</u>	<u>Proportion</u>
M388	0.873
M157	0.618
M157_M195+M057	0.455
M195+M057	0.455
M009	0.418
M218	0.364
M314	0.200
M288	0.164
M122	0.055
M260~1	0.055

Mean Pairwise RBO: 0.209874

<u>Term</u>	<u>Proportion</u>
M327	1.000
M351+X	0.500
M365	0.333
M057	0.167
M009	0.167
M374~C	0.167
M247~G	0.167
M136	0.167
M153+X	0.167
M158	0.167

Mean Pairwise RBO: 0.276212

<u>Term</u>	<u>Proportion</u>
M263~A	0.714
M263	0.619
M263~B1	0.524
M387~A	0.476
M002	0.333
M048~D	0.286
M387~EF	0.238
M010~2	0.190
M057~B	0.143
M384~D	0.048

Mean Pairwise RBO: 0.214407

<u>Term</u>	<u>Proportion</u>
M242~B	0.857
M242~B_M096	0.857
M096	0.571
M066	0.429
M288+1(N01)	0.286
M387~I	0.143
M057~A	0.143
M387~I_M387~I	0.143
M218~D+M288	0.143
M003~B	0.143

Mean Pairwise RBO: 0.101756

<u>Term</u>	<u>Proportion</u>
M145	0.542
M264~D	0.500
M367~C	0.333
M317~A	0.292
M081	0.250
M038~E	0.208
M002	0.167
M036+1(N24)	0.083
M210~D	0.042
M064~A	0.042

Mean Pairwise RBO: 0.098027

<u>Term</u>	<u>Proportion</u>
M036+1(N30D)	0.392
M243~J	0.275
M305+M342	0.255
M033	0.176
M305+X	0.176
M149~A2	0.176
M051~B	0.157
M222	0.098
M384	0.098
M146~D	0.098

Mean Pairwise RBO: 0.031085

<u>Term</u>	<u>Proportion</u>
M124	0.096
M263~B	0.056
M073~B	0.051
M051~C	0.034
M260~1+1(N24)	0.034
M230~A	0.028
M010~6	0.028
M006@G	0.022
M312	0.017
M097~H_M004	0.011

### A.3 Precedence Relations

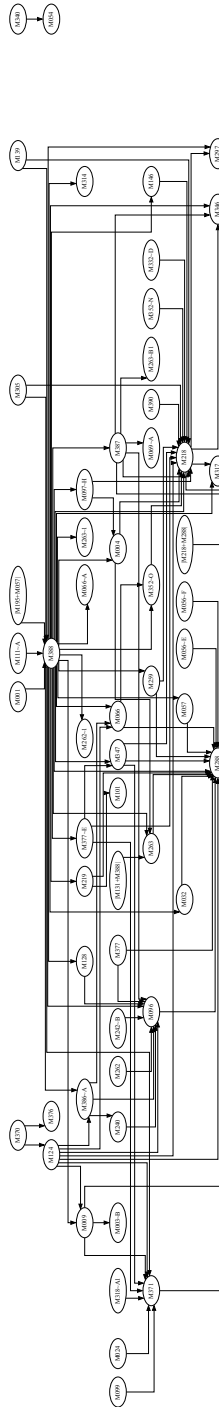
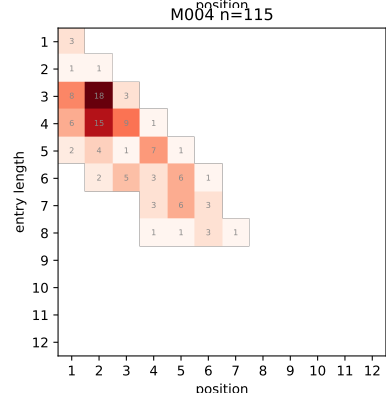
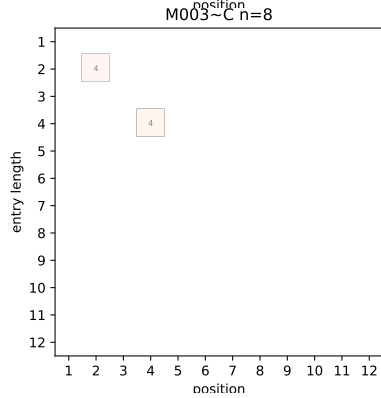
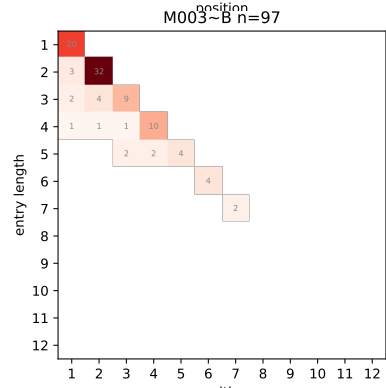
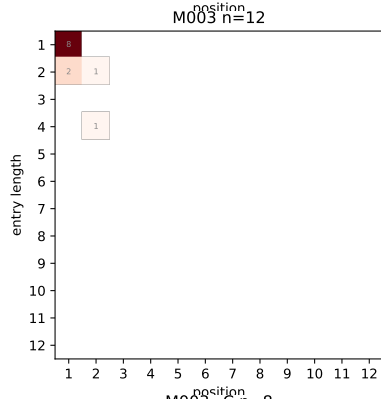
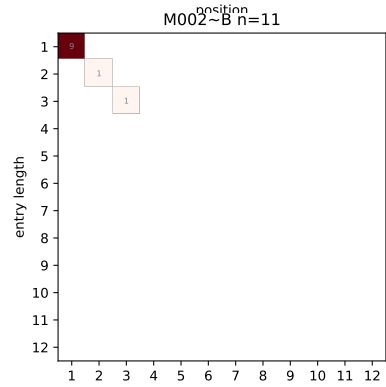
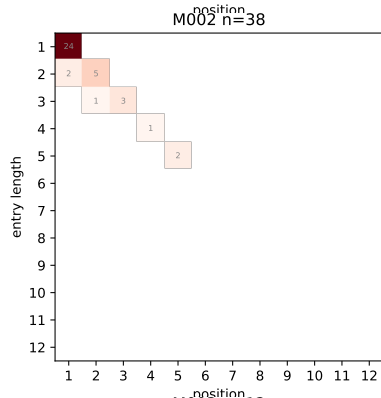
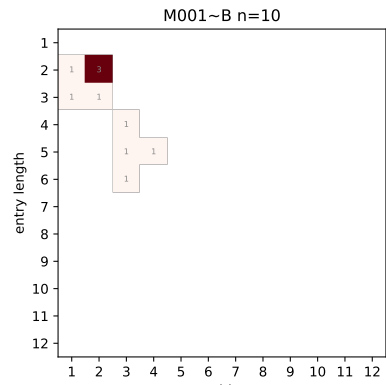
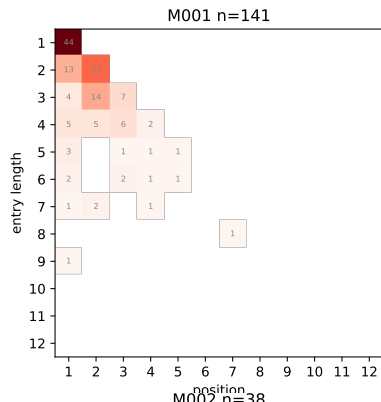


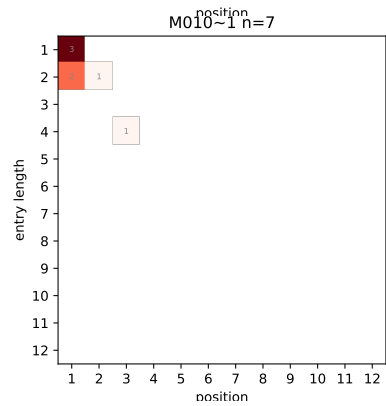
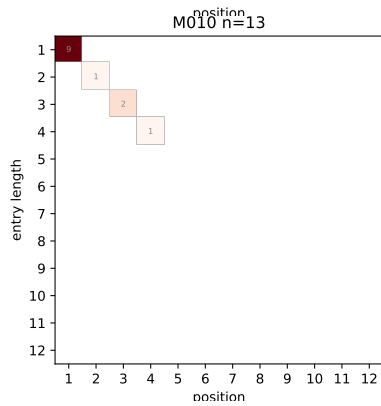
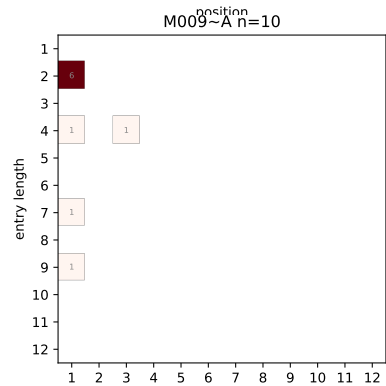
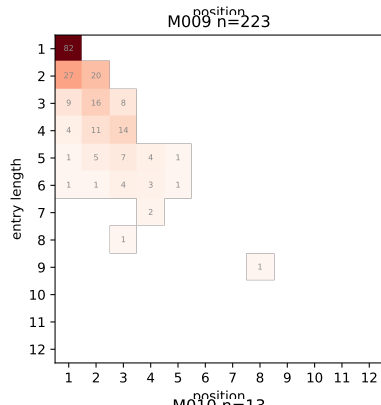
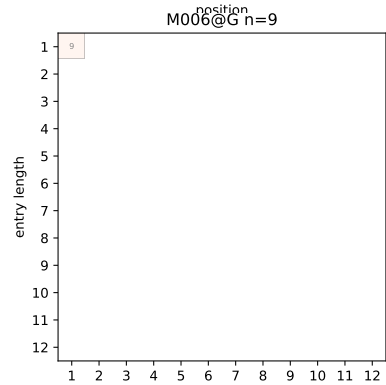
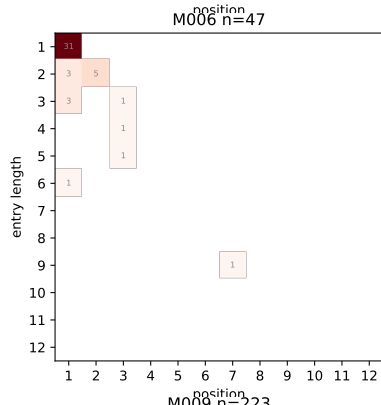
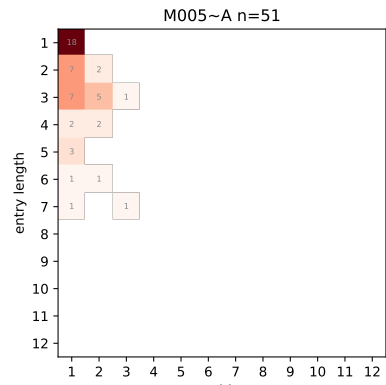
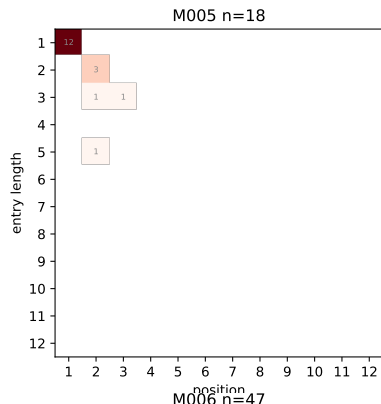
Figure A.4: Partial order over signs occurring together 10 times or more, inferred from  $\chi^2$  tests (Section 5.2). A directed arc from A to B implies that A is significantly more likely to occur before B than after, when both signs are present in an entry.

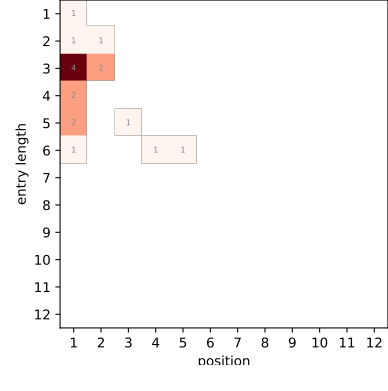
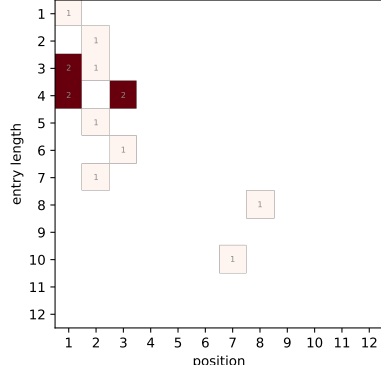
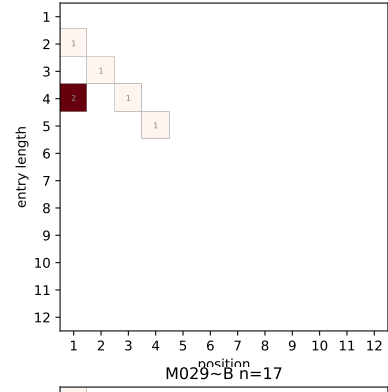
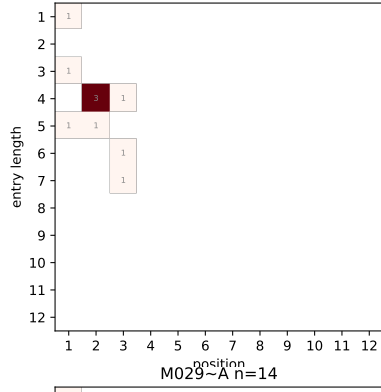
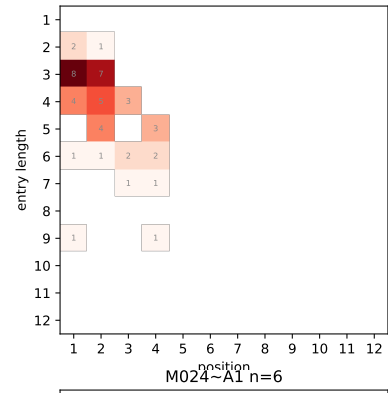
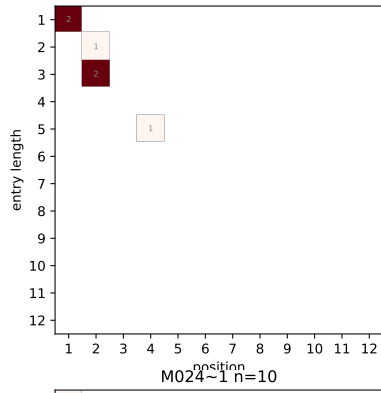
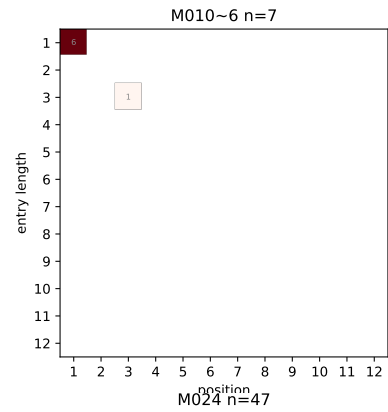
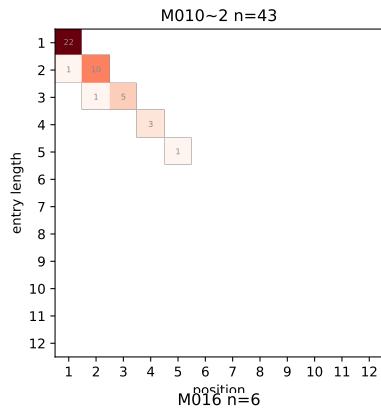
## A.4 Sign Position Heatmaps

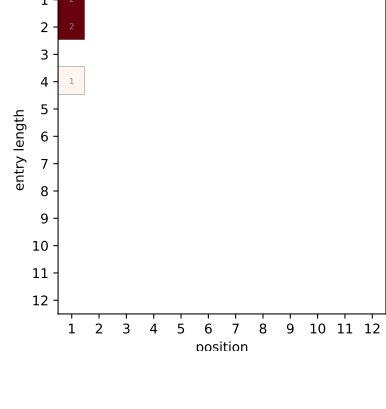
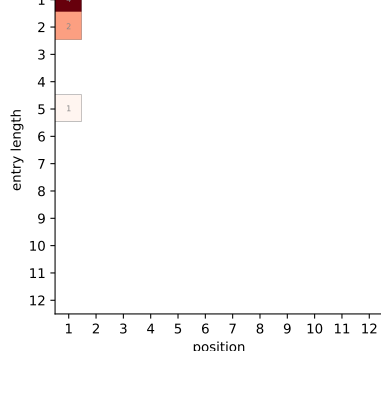
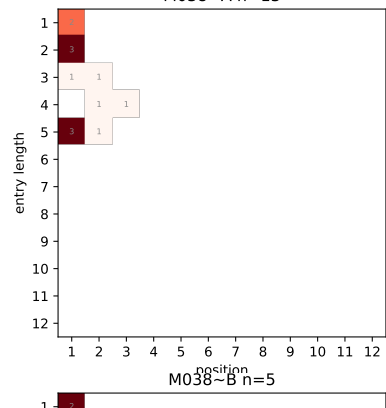
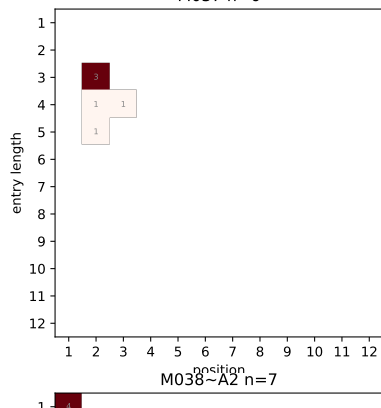
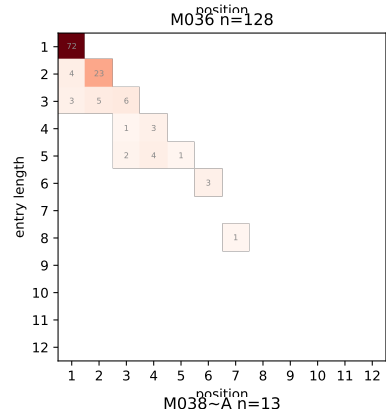
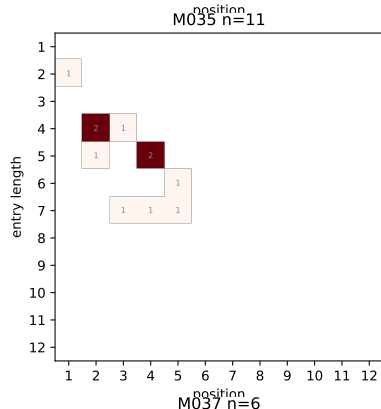
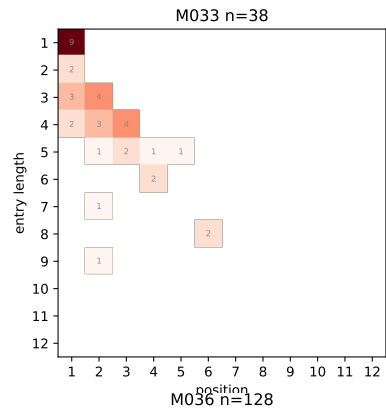
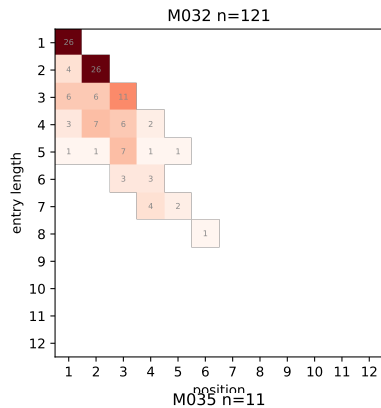
In the following heatmaps, each row represents a fixed entry length, and the intensity of the cells show how frequent a sign is at each position in entries of that length. Object signs can be recognized by a strong trend along the main diagonal, representing frequent occurrences at the very end of entries of all lengths (e.g. M288). The class of “suffixal” signs, which we identified as preferring penultimate position, can be similarly recognized by the presence of dark cells directly below the main diagonal (cf. M066).



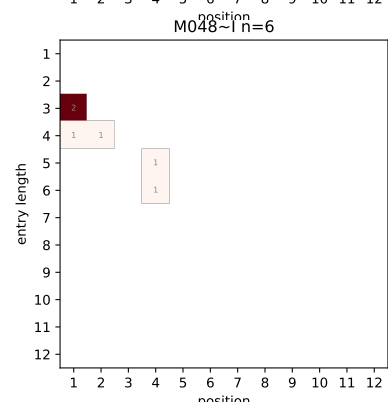
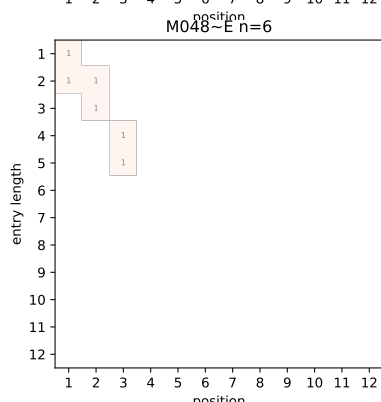
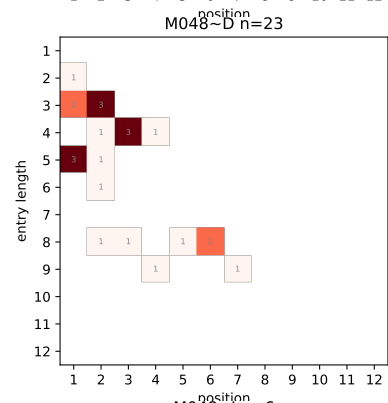
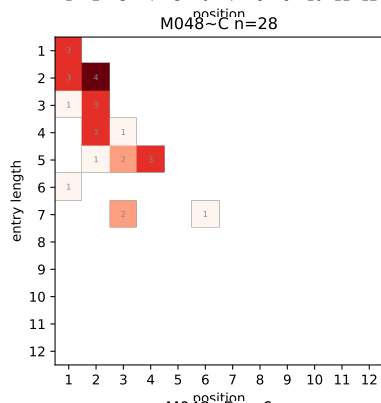
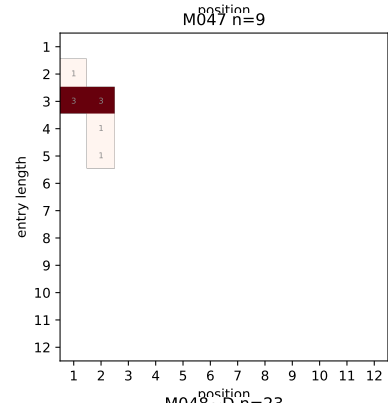
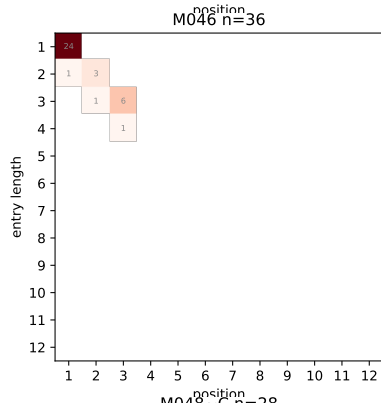
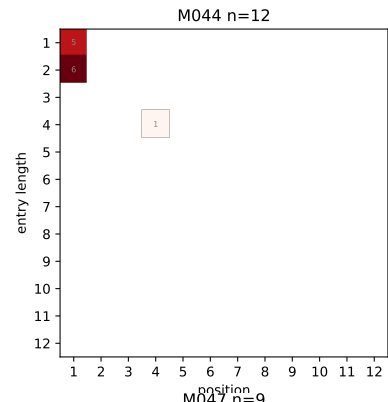
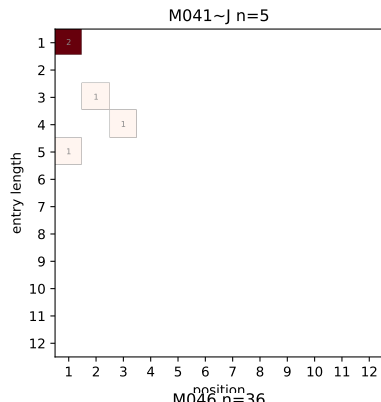


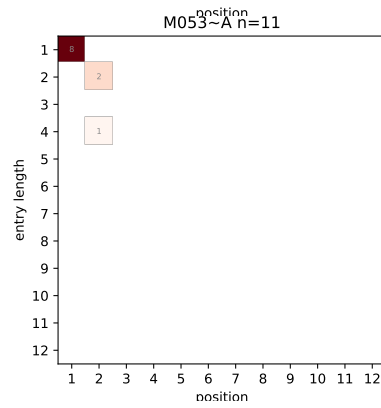
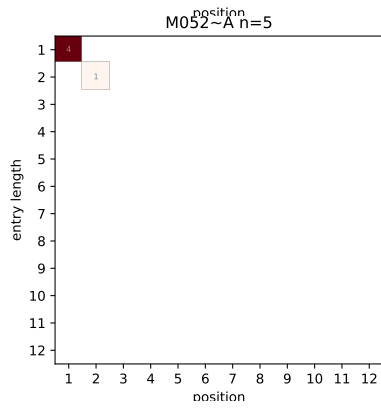
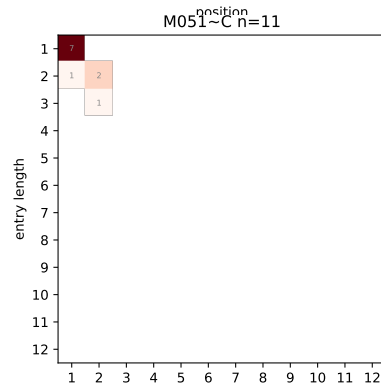
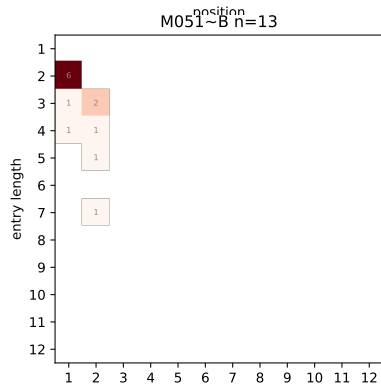
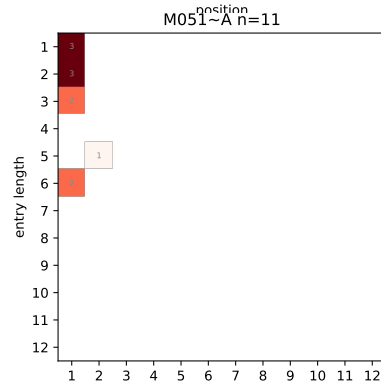
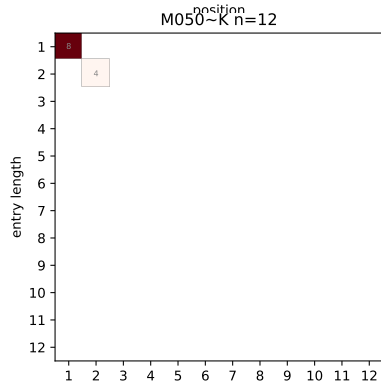
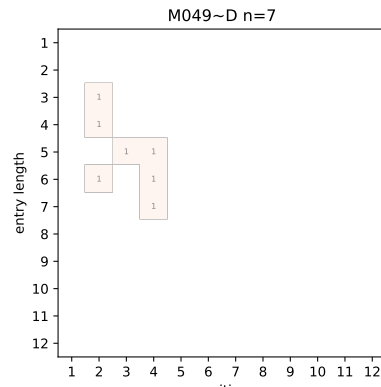
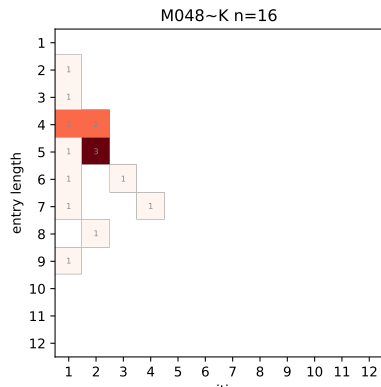


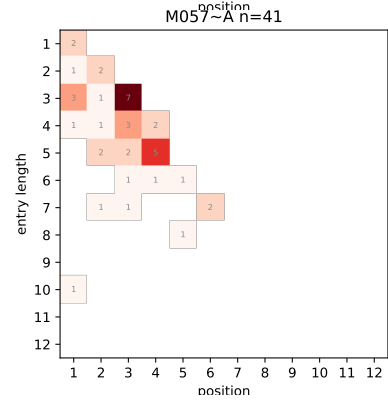
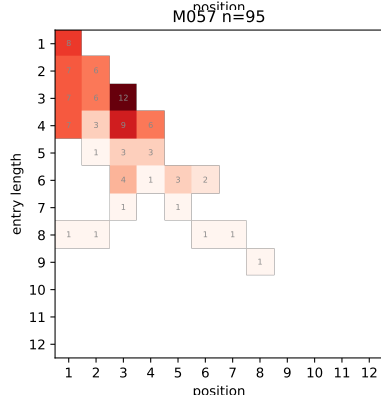
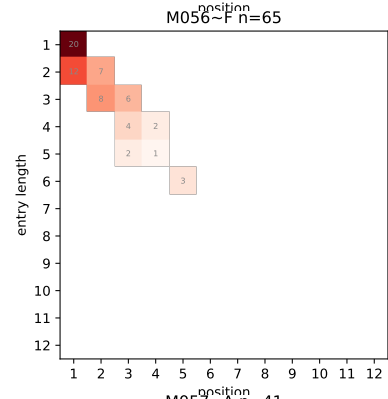
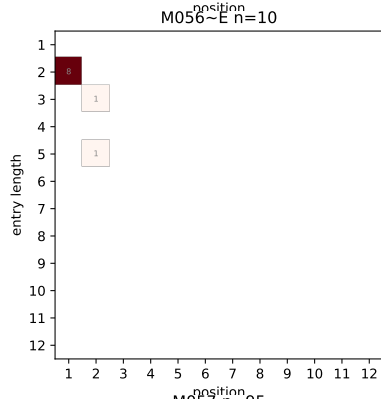
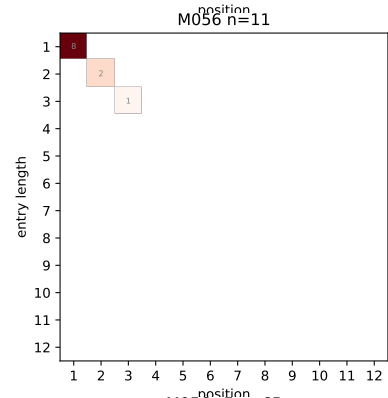
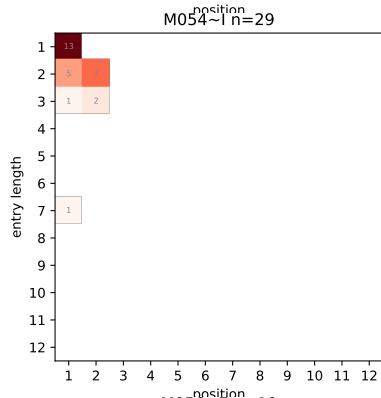
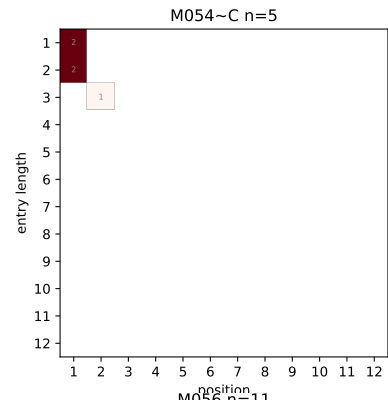
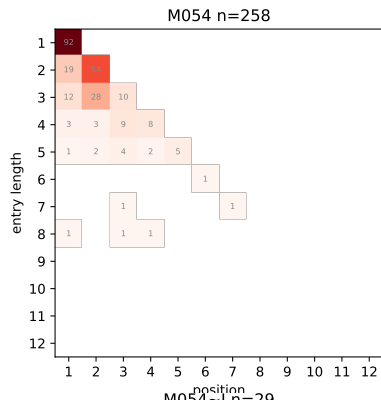


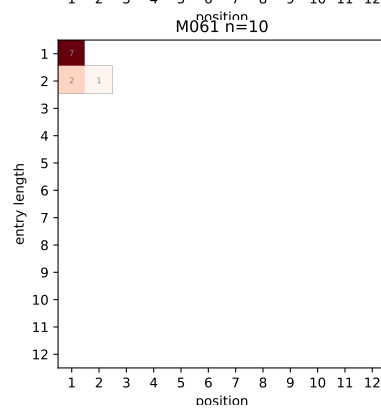
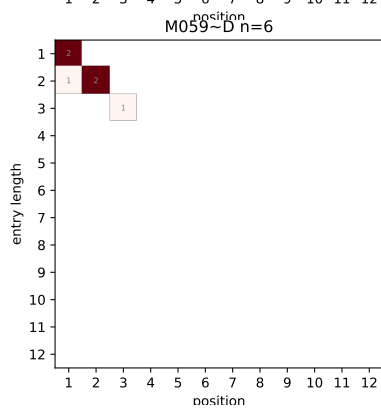
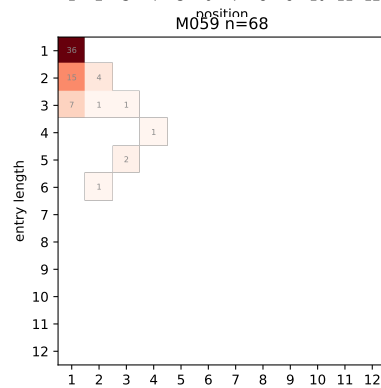
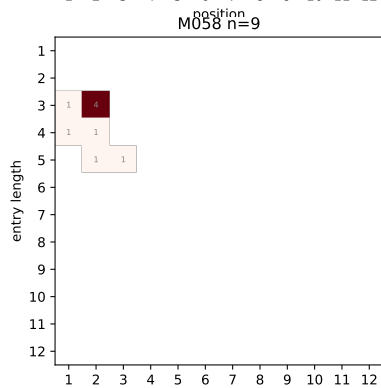
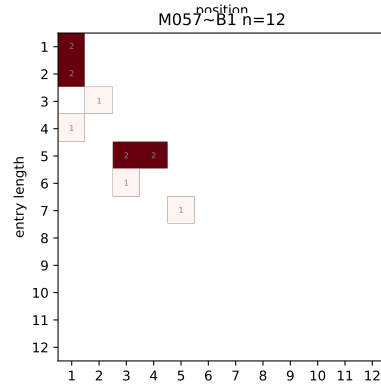
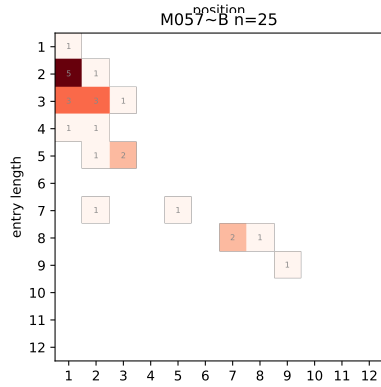
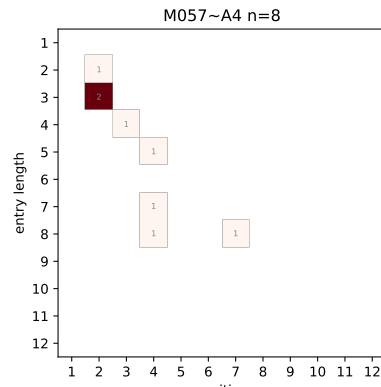
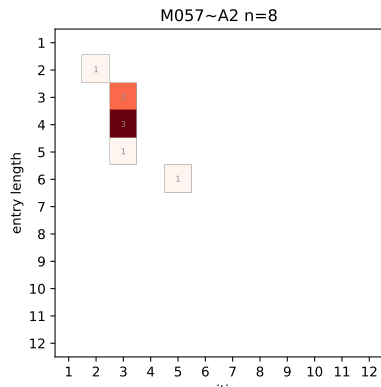




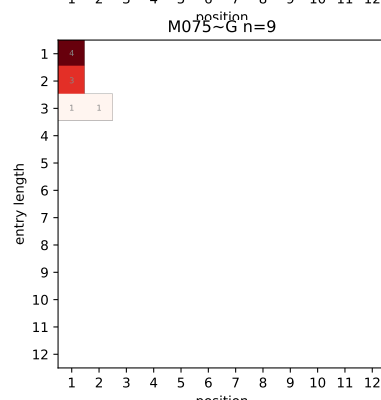
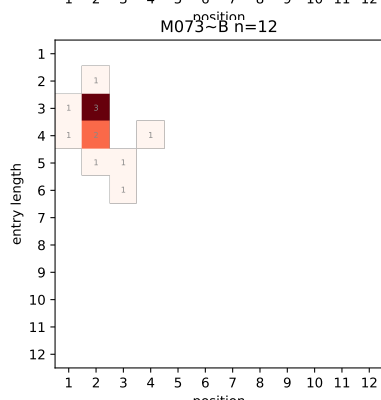
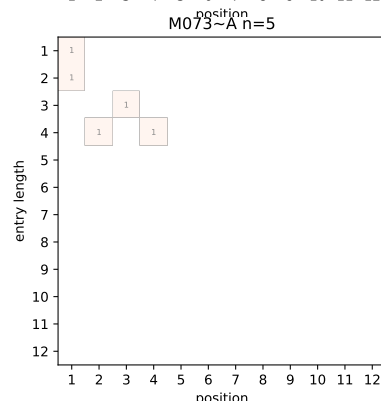
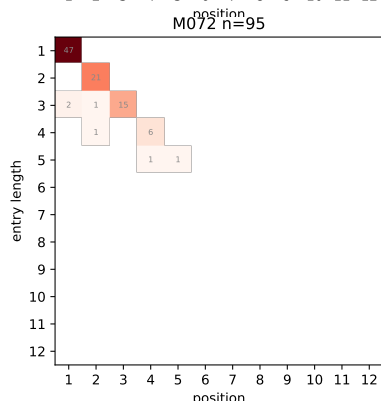
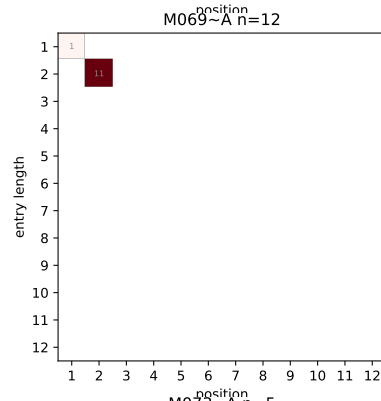
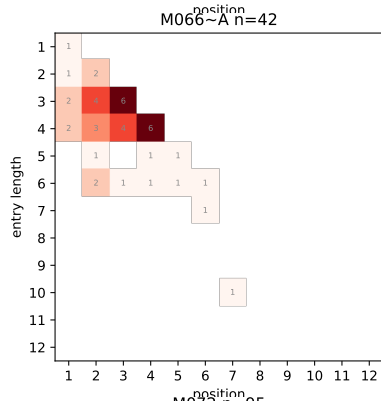
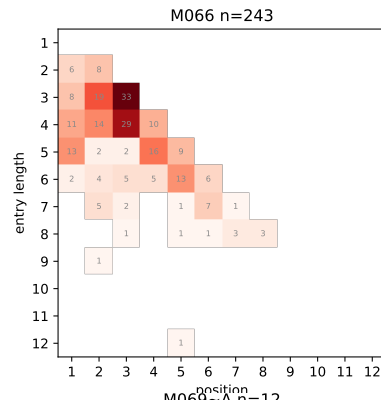
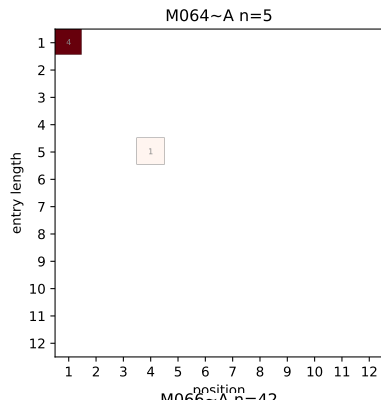


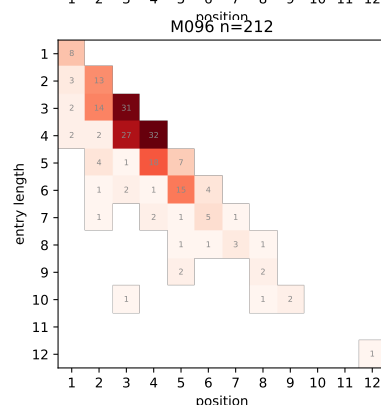
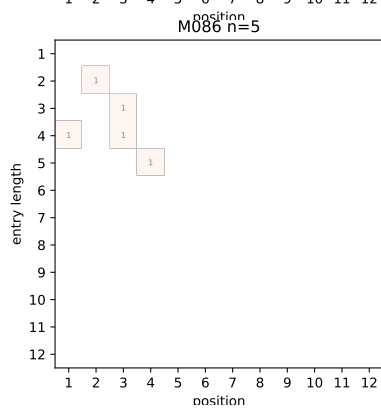
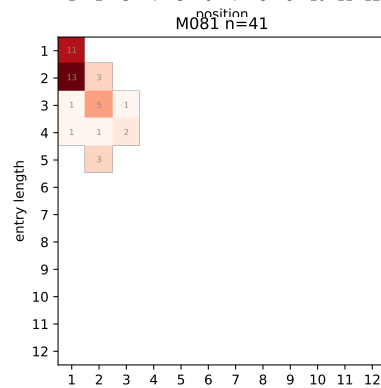
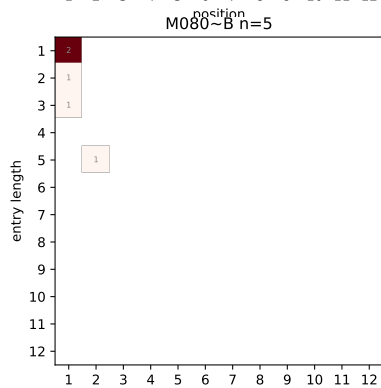
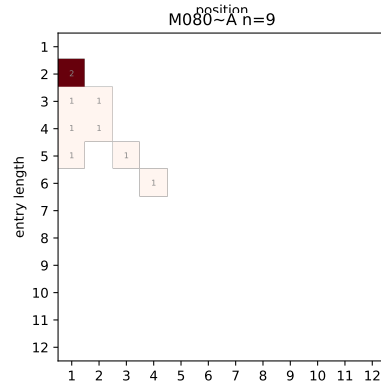
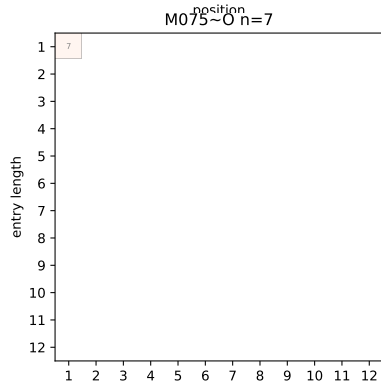
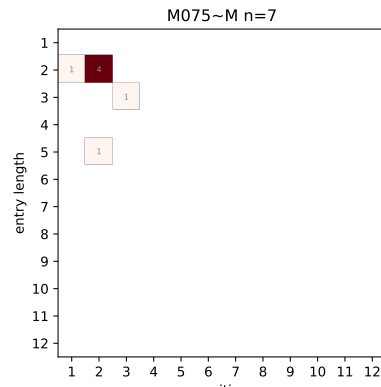
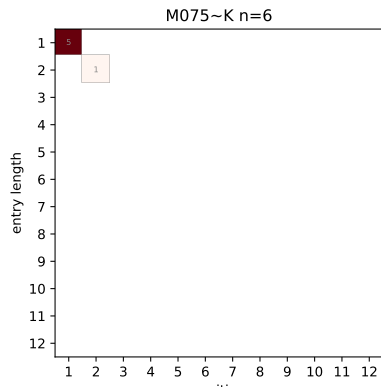


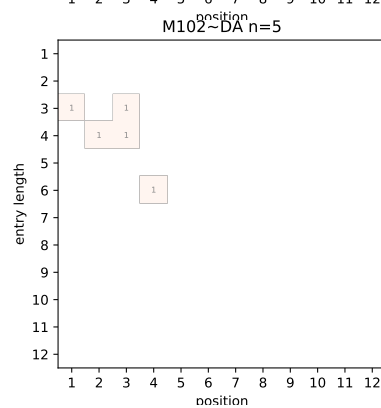
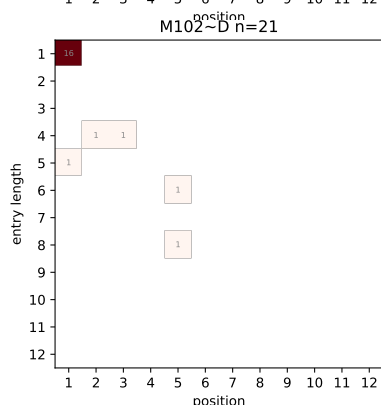
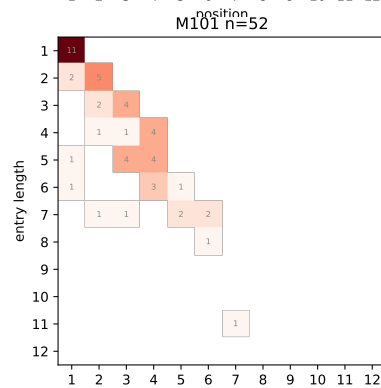
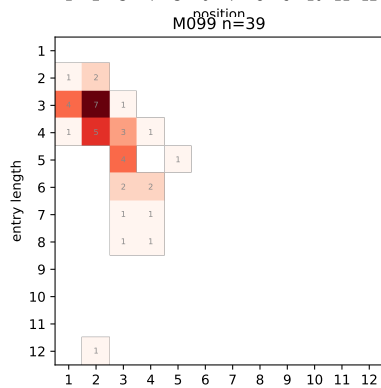
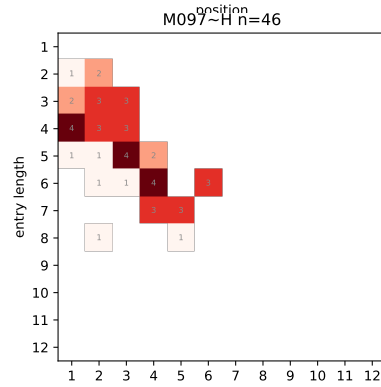
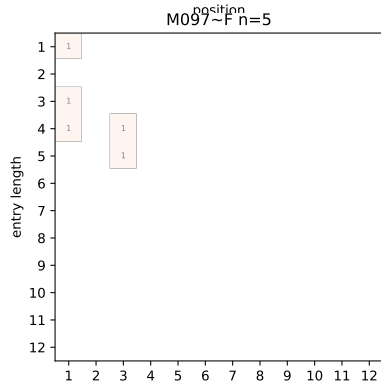
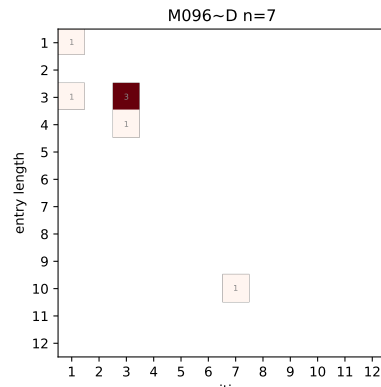
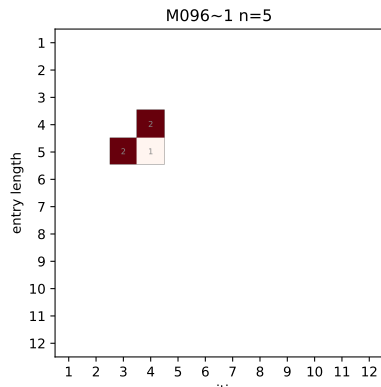


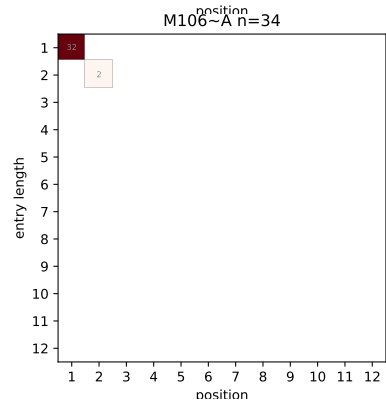
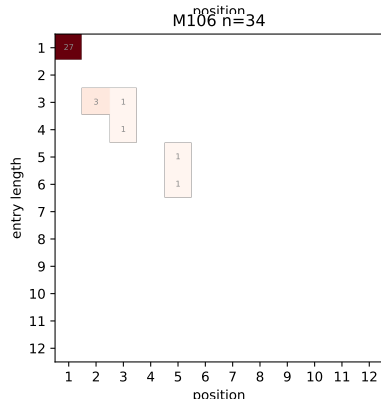
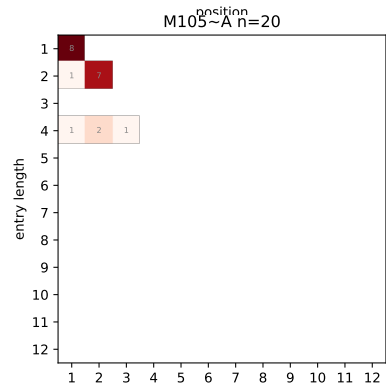
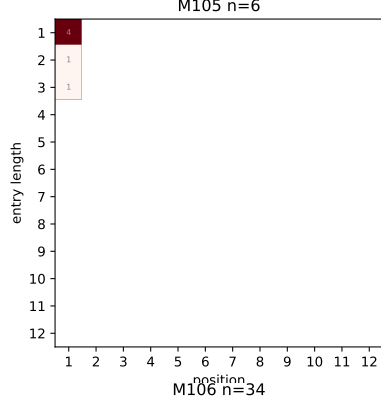
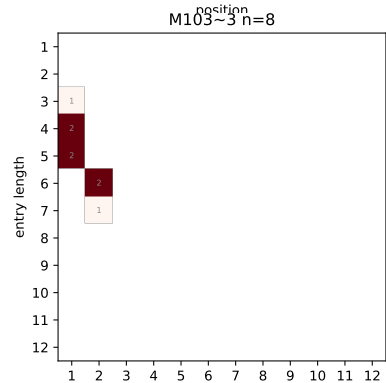
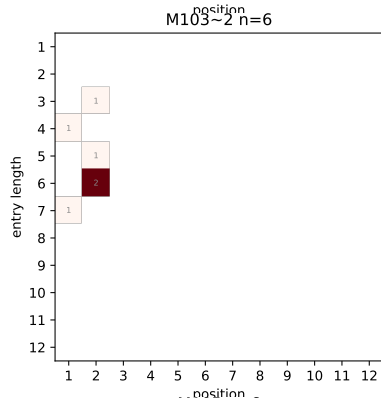
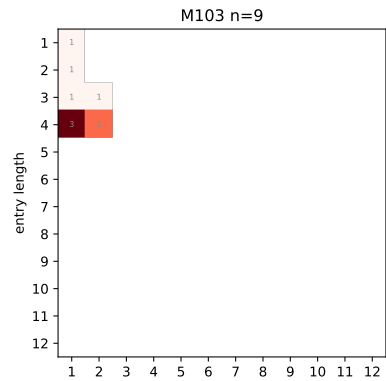
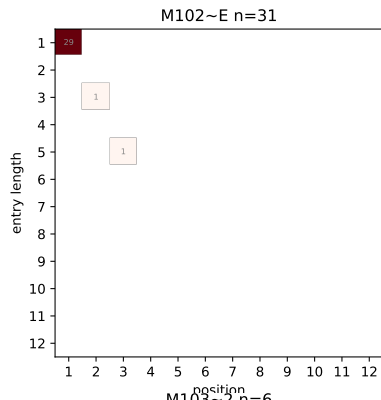




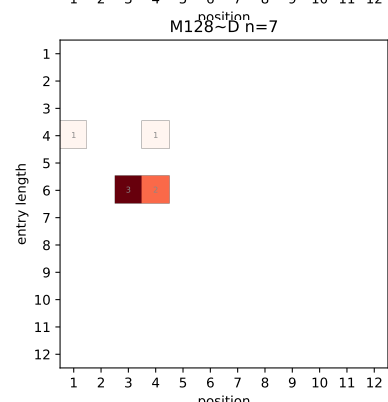
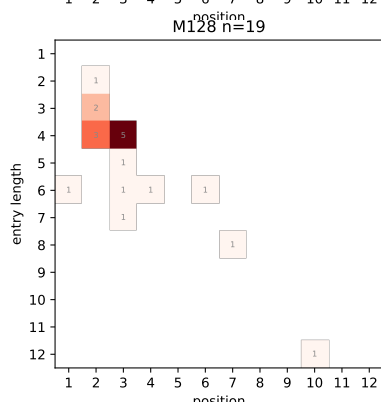
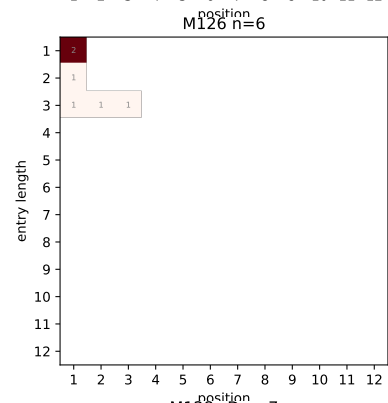
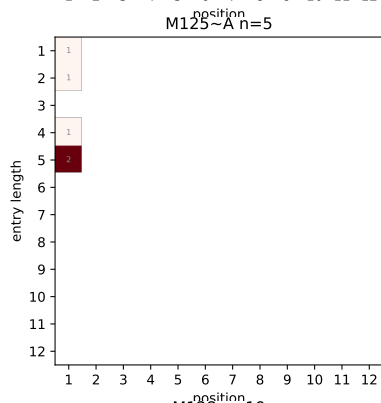
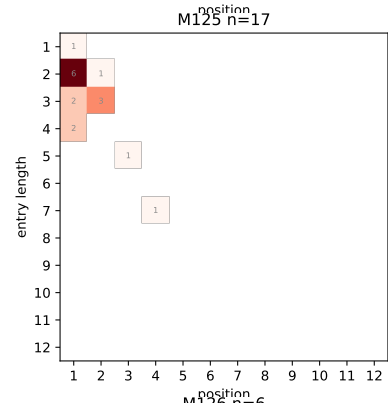
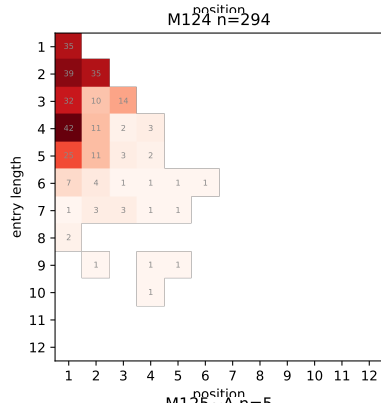
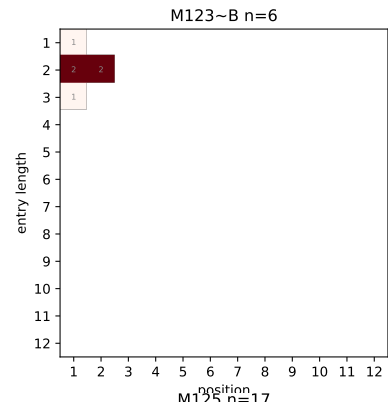
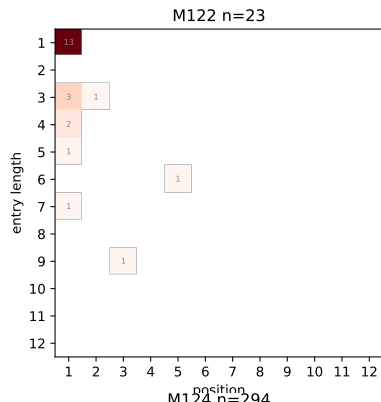


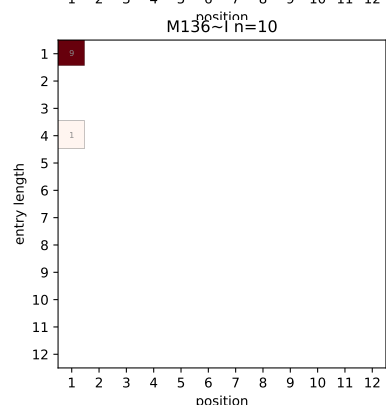
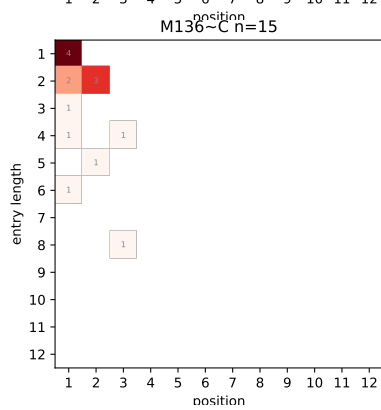
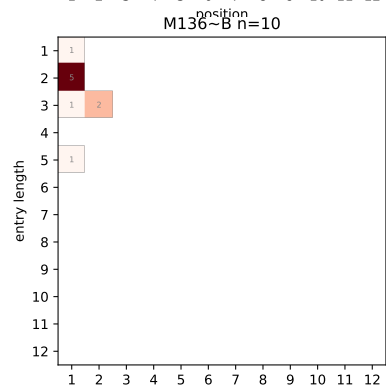
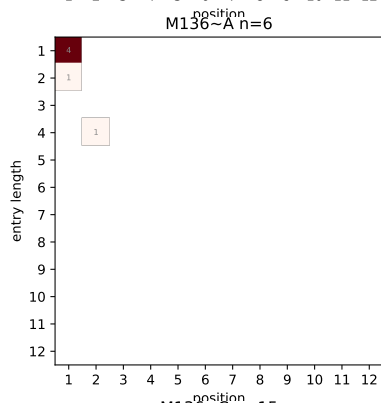
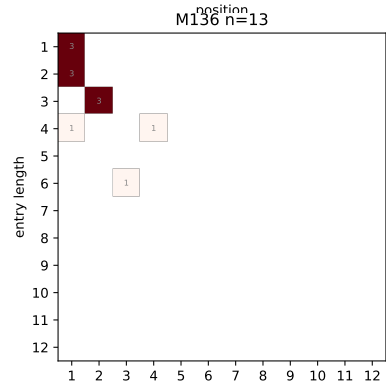
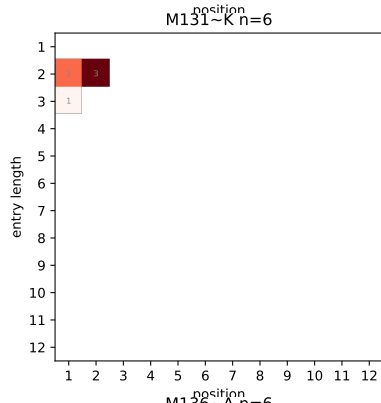
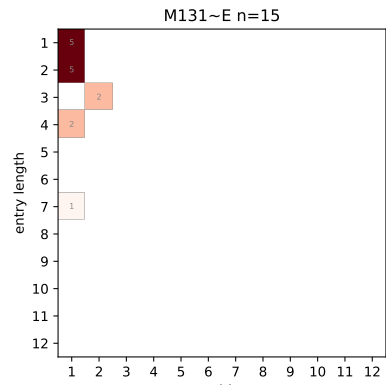
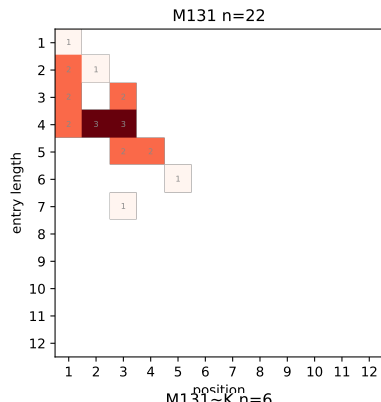


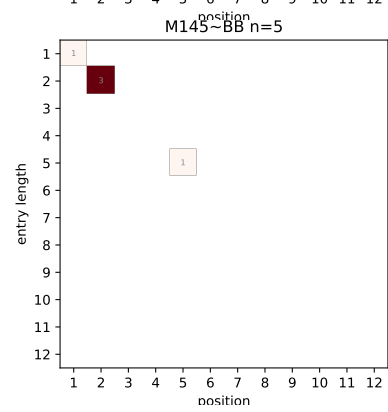
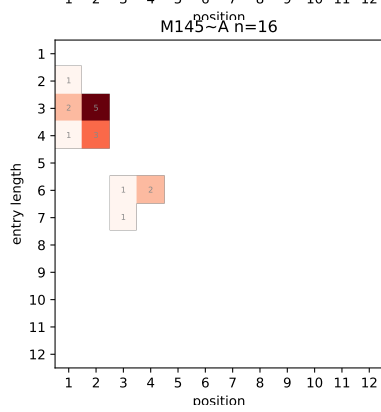
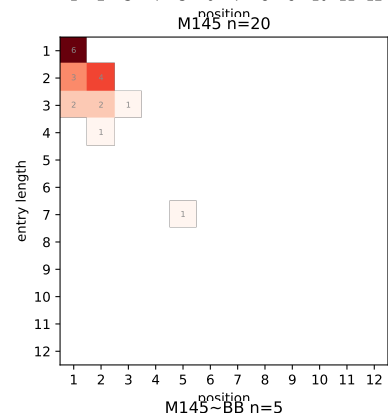
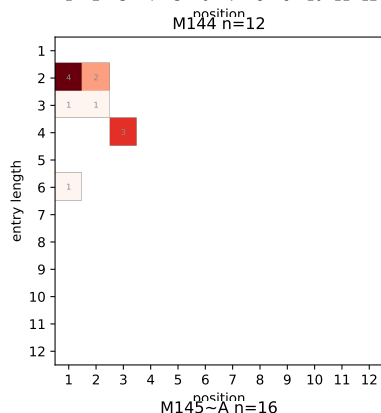
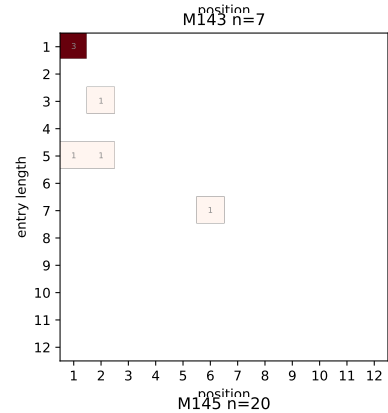
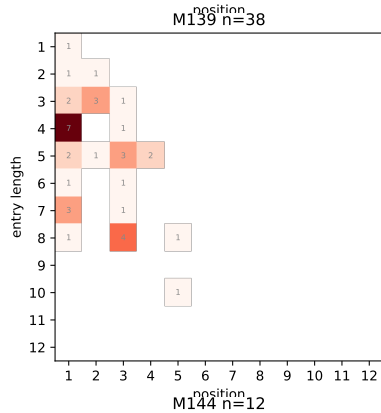
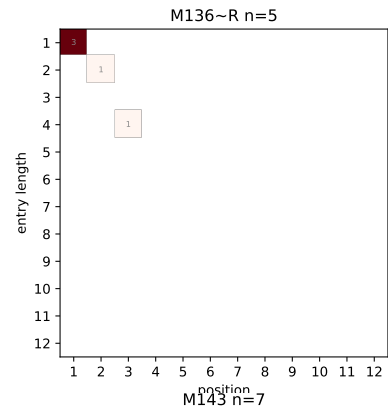
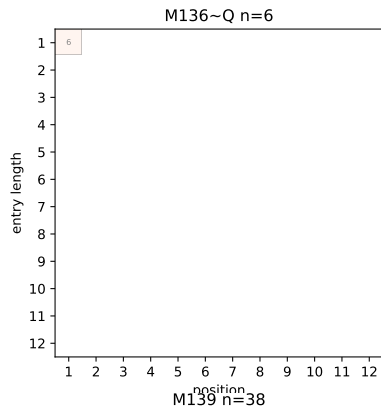




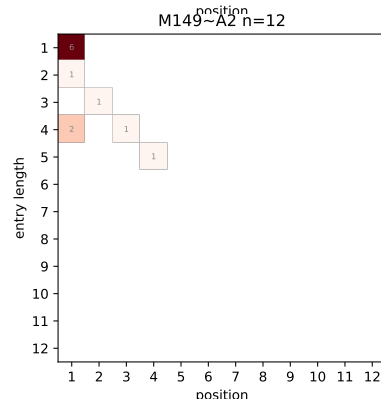
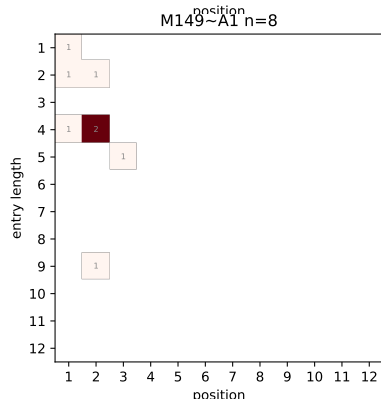
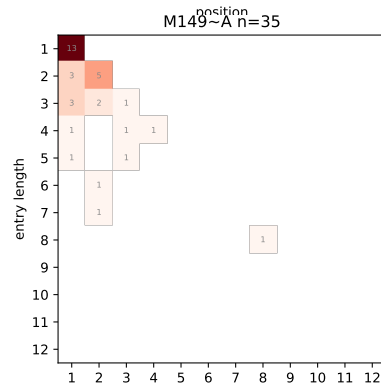
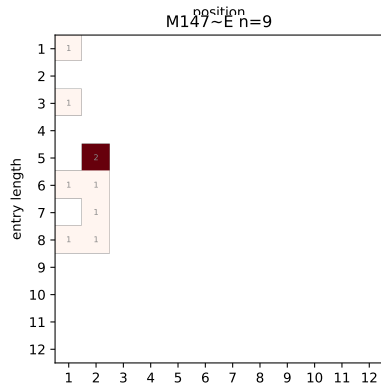
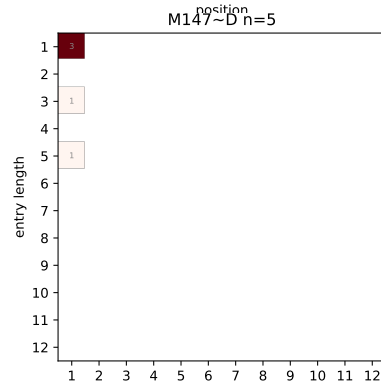
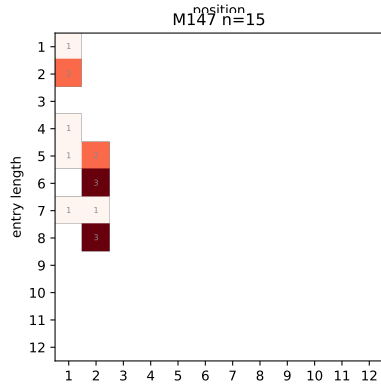
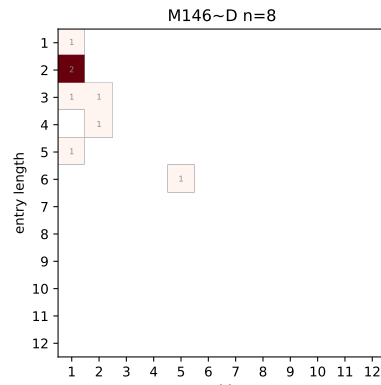
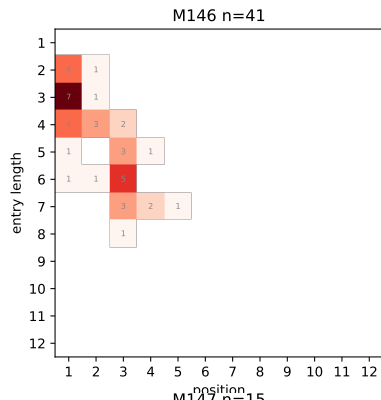


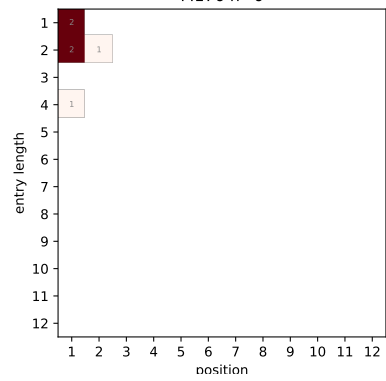
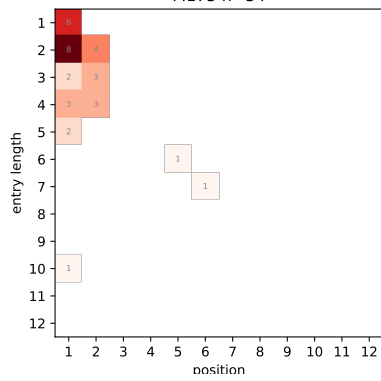
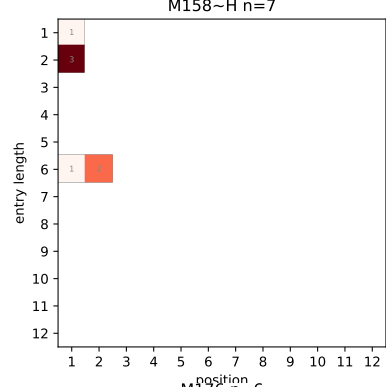
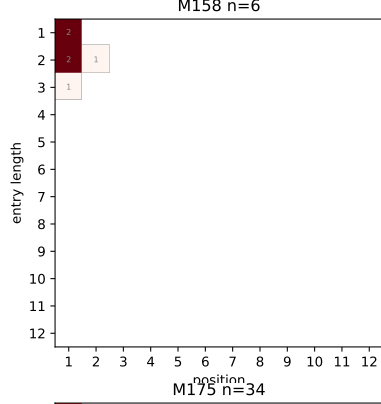
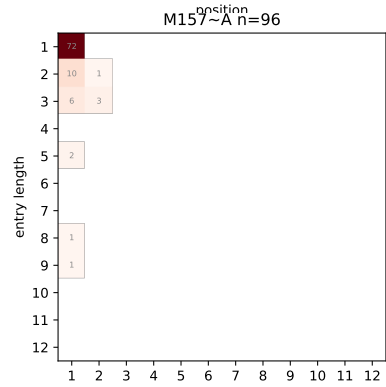
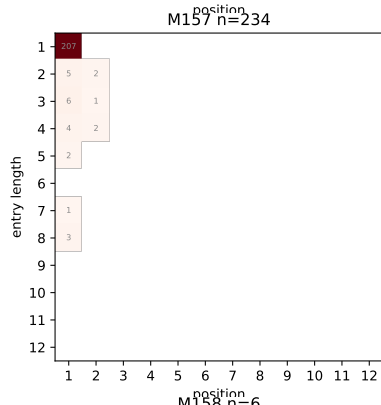
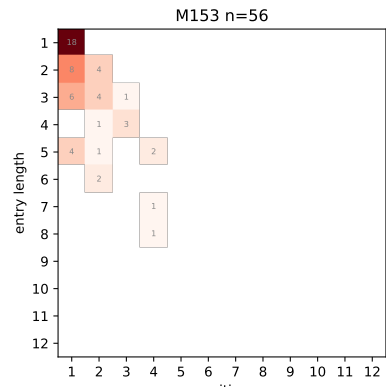
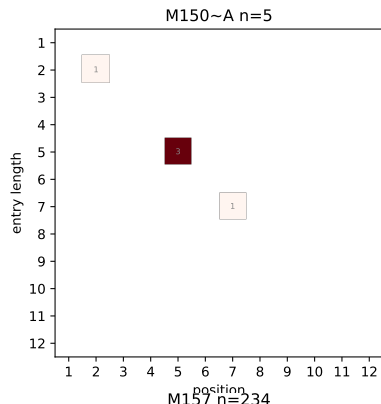


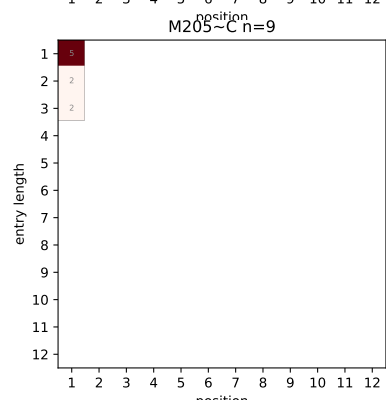
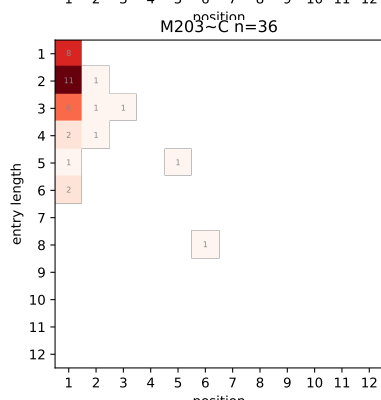
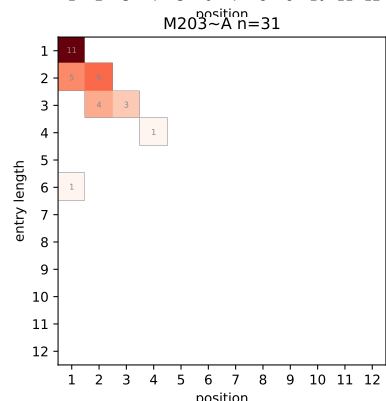
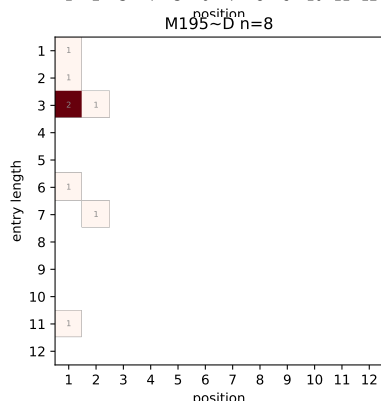
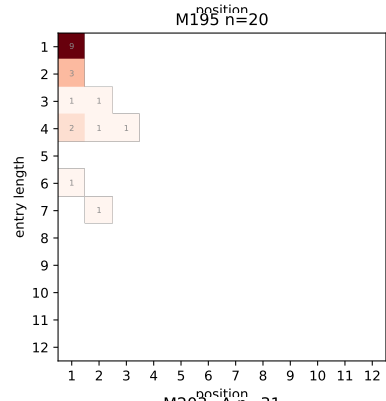
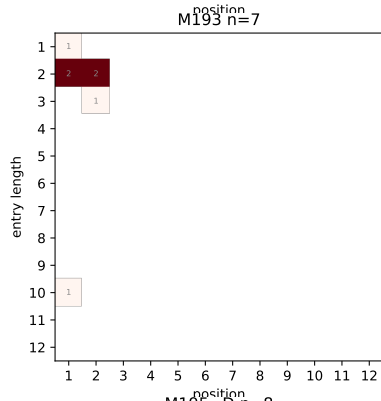
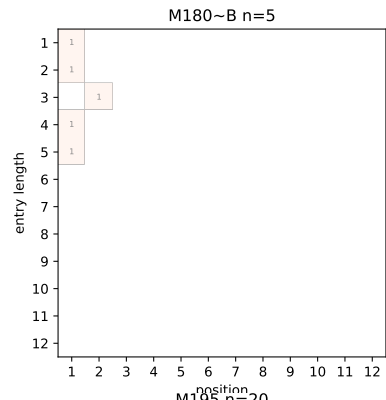
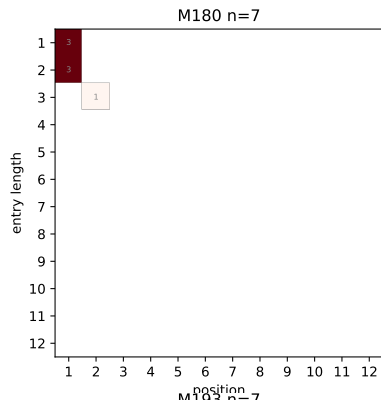


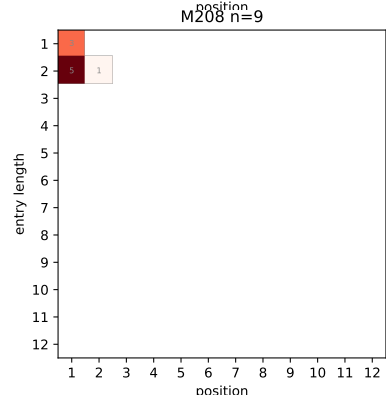
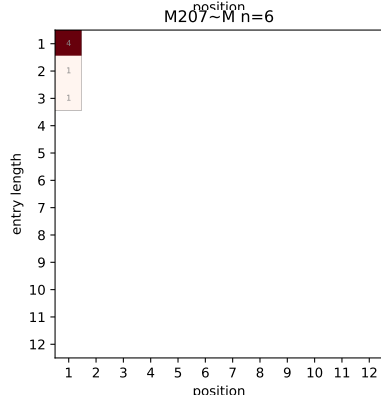
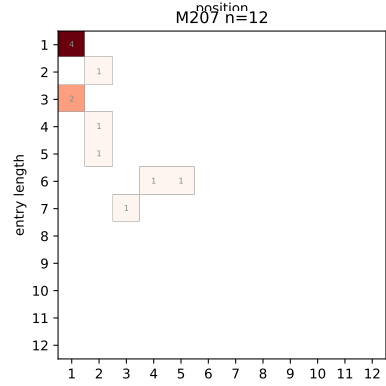
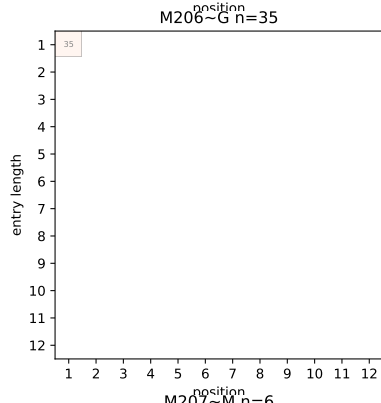
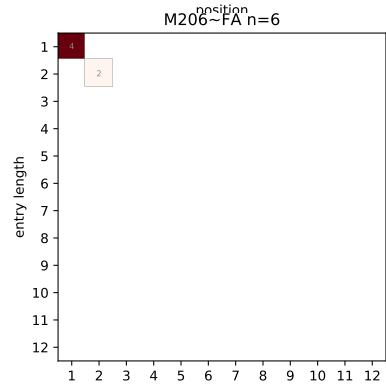
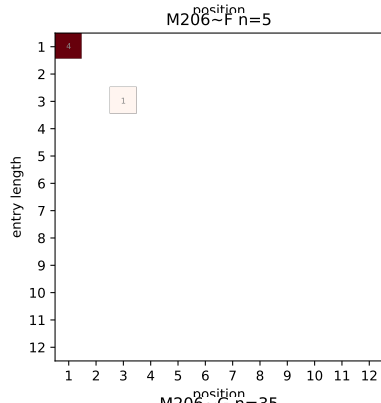
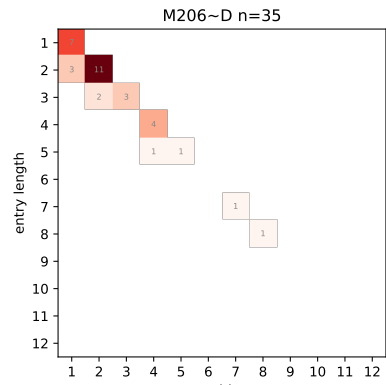
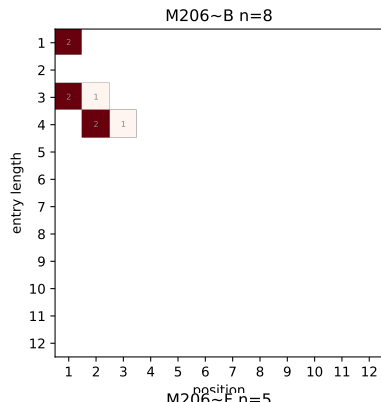


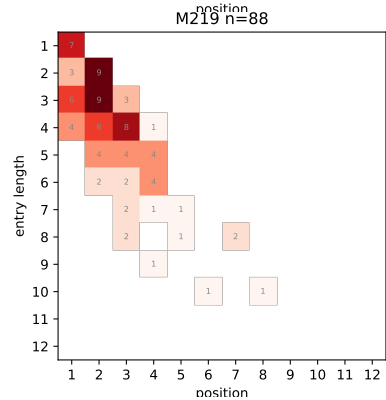
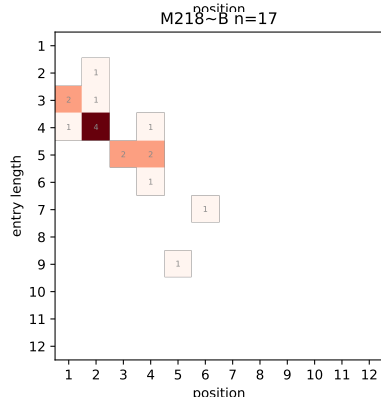
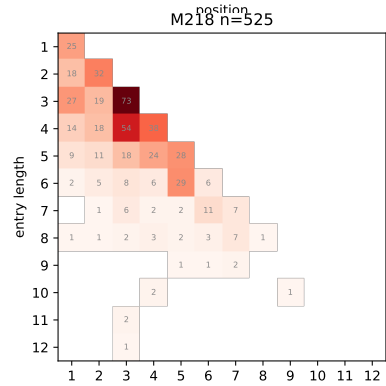
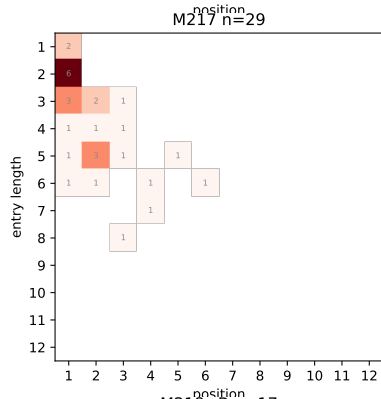
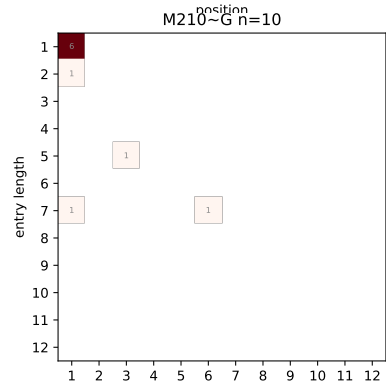
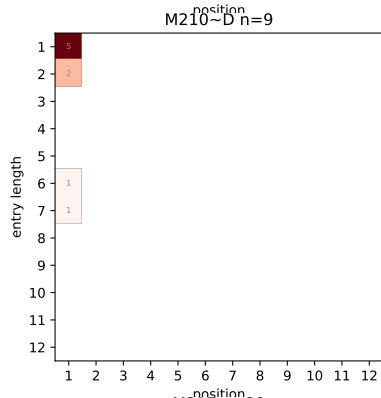
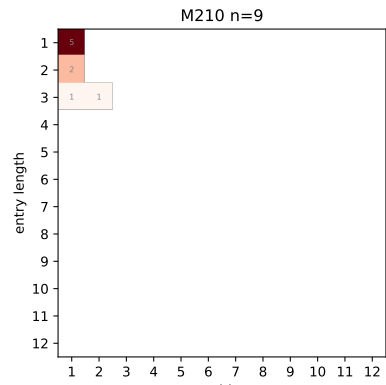
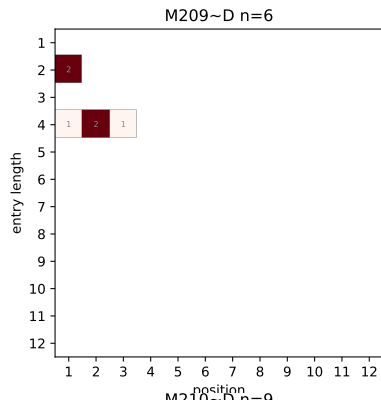


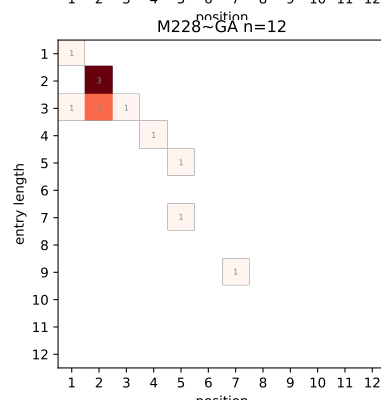
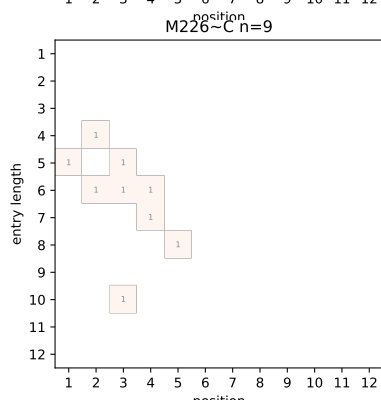
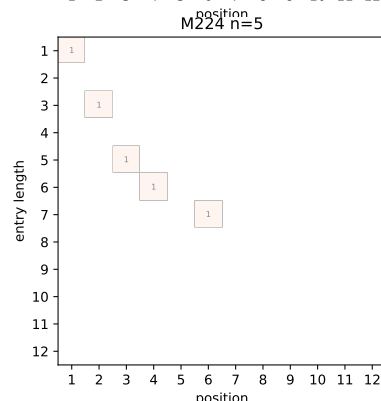
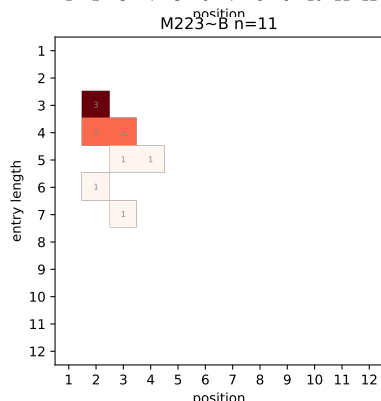
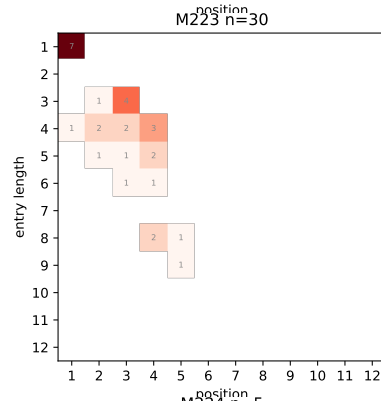
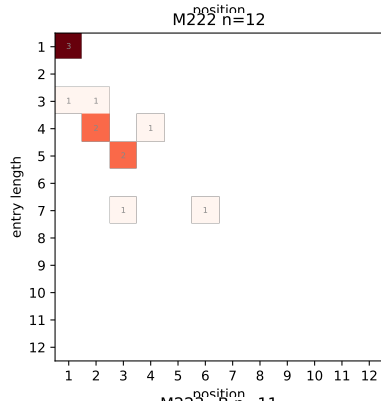
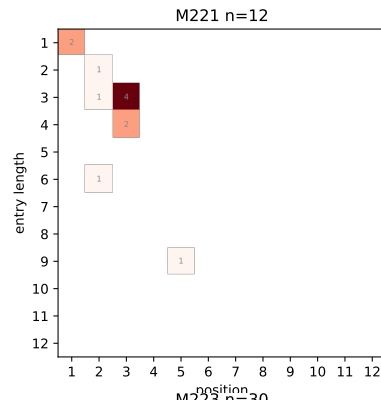
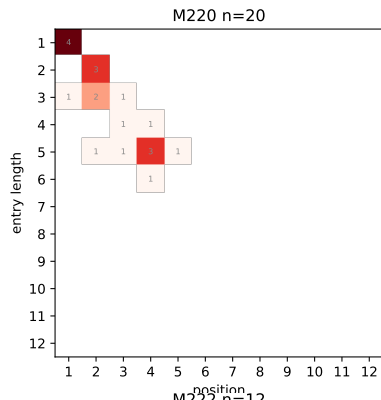


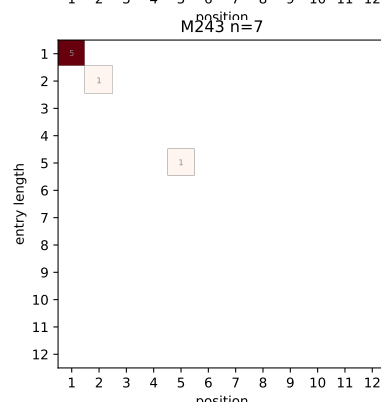
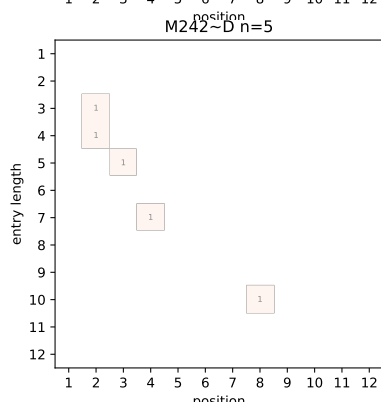
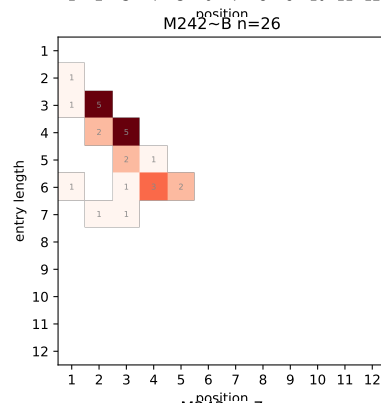
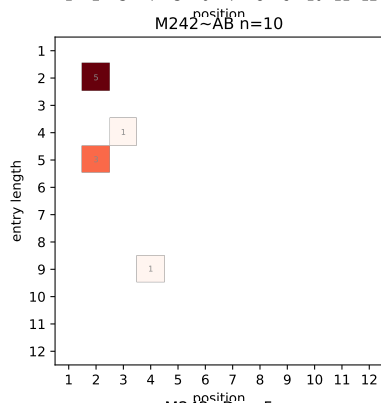
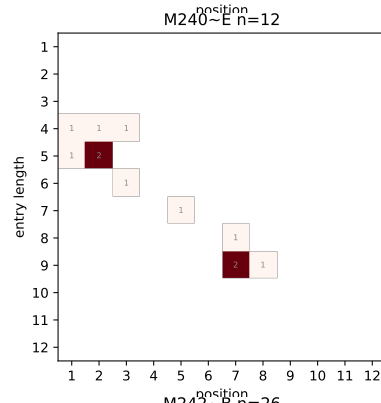
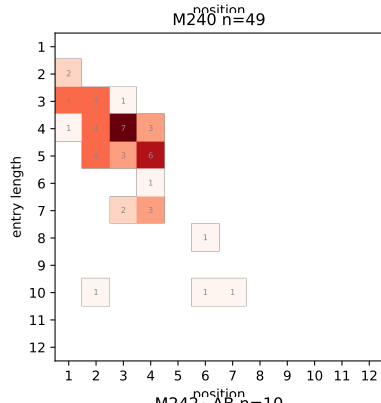
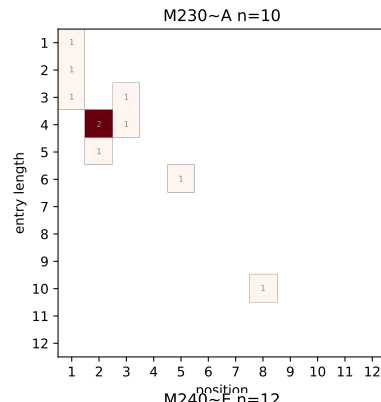
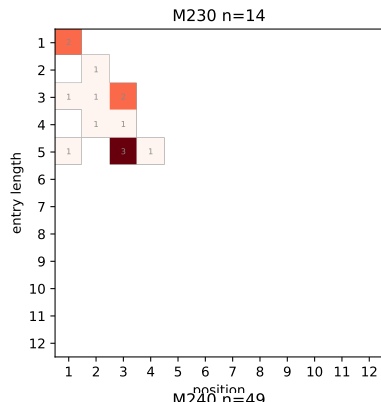


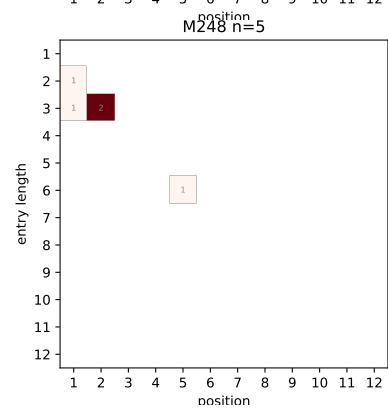
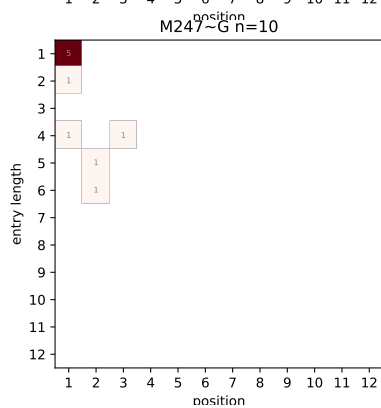
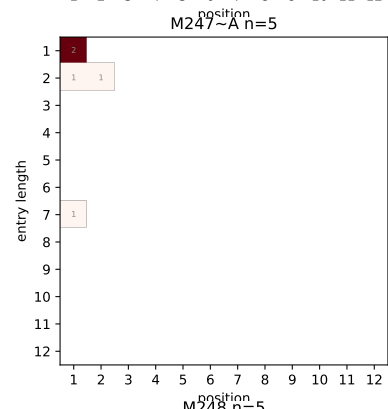
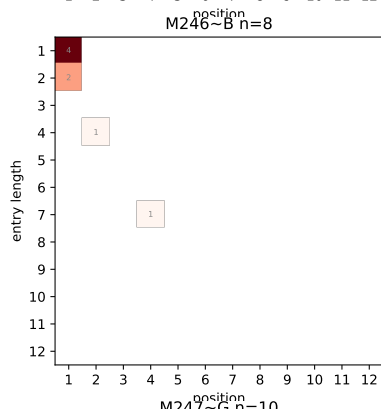
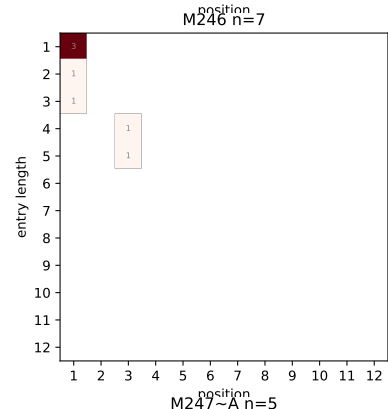
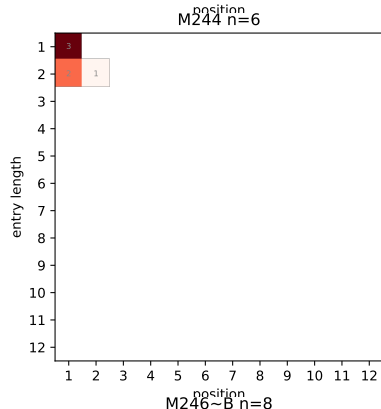
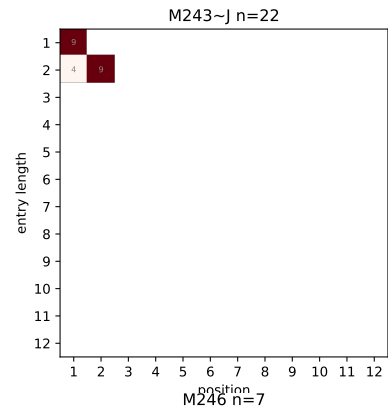
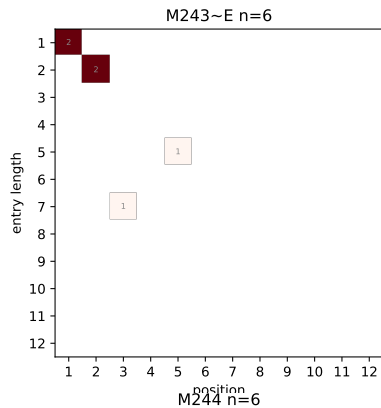






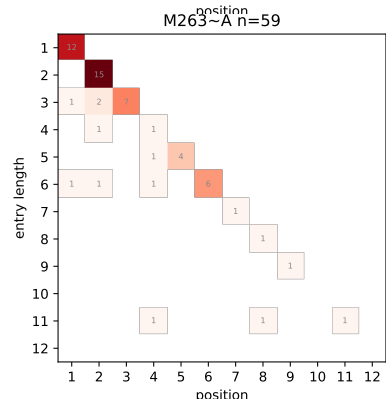
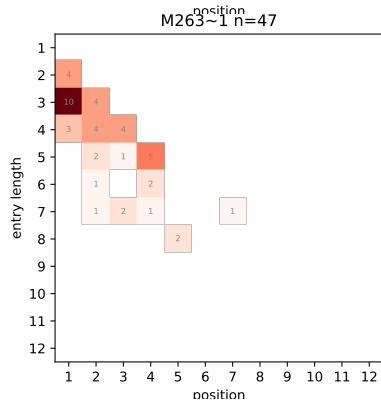
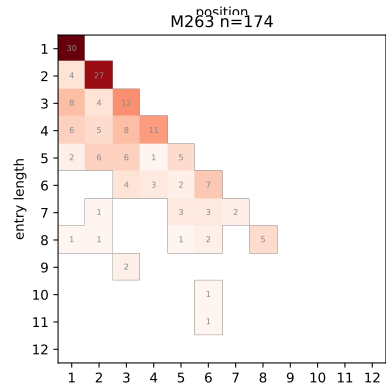
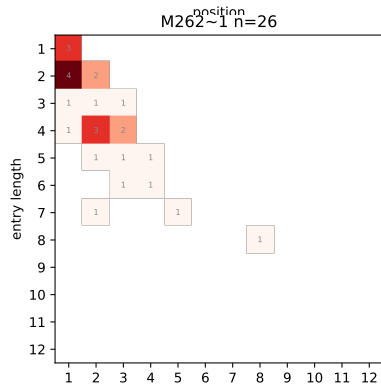
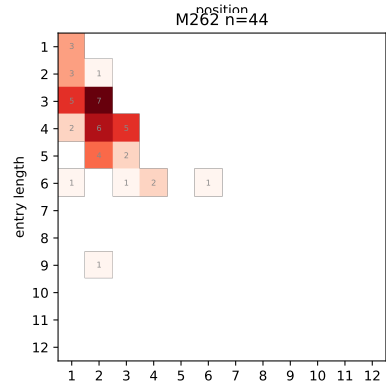
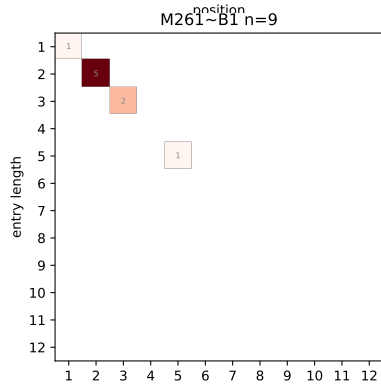
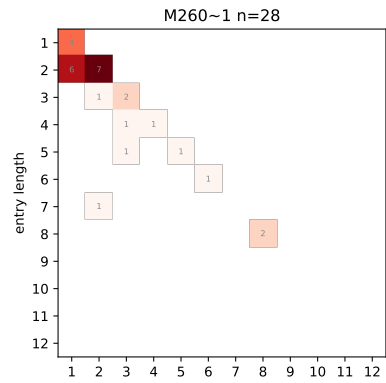
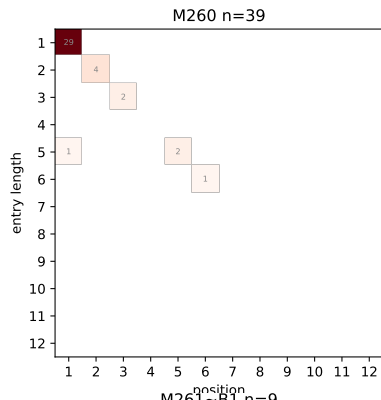


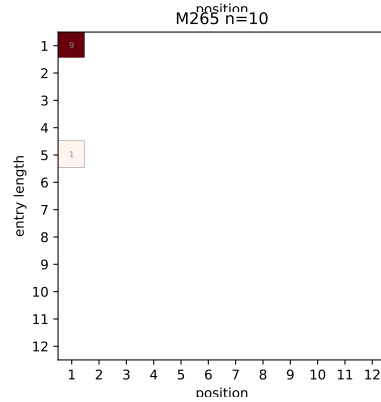
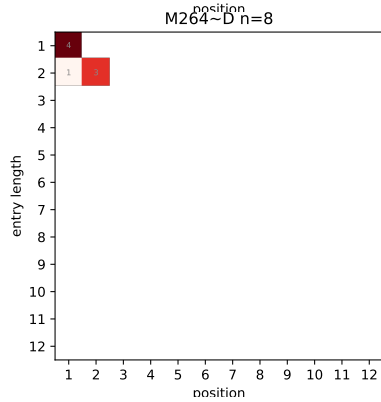
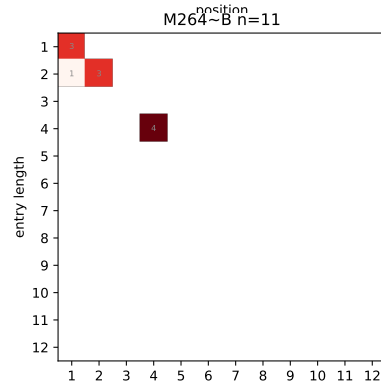
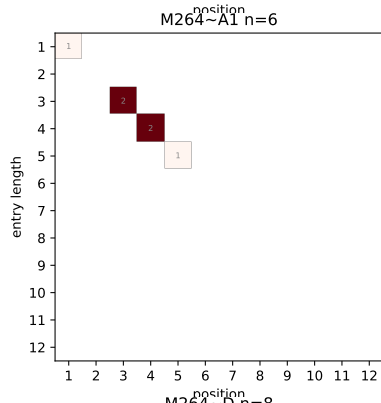
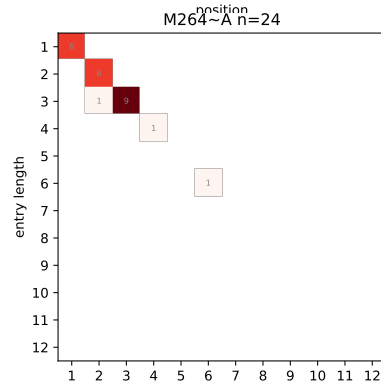
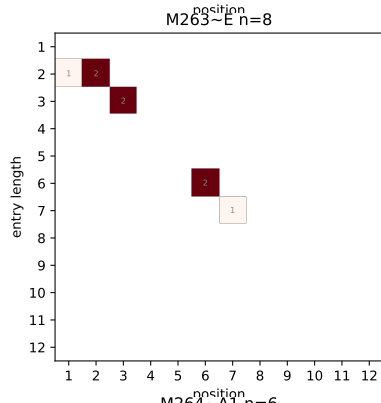
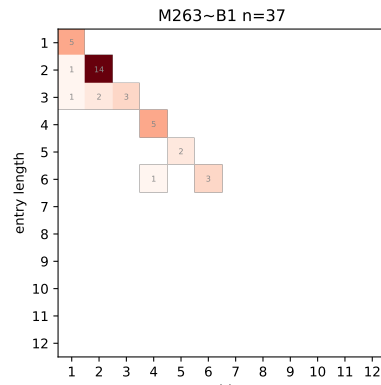
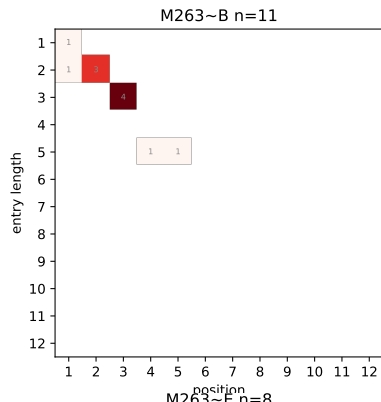


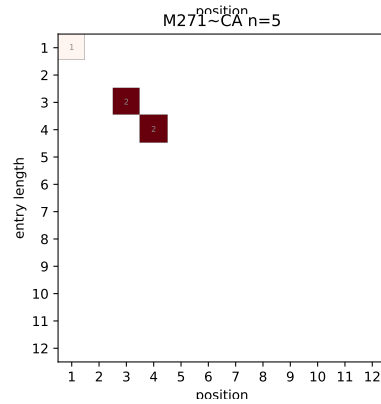
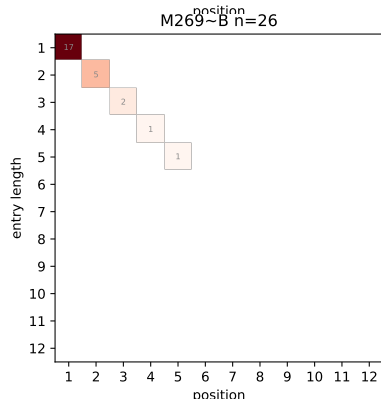
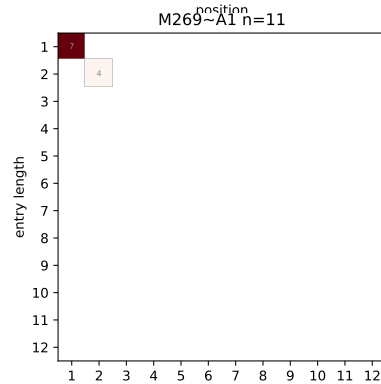
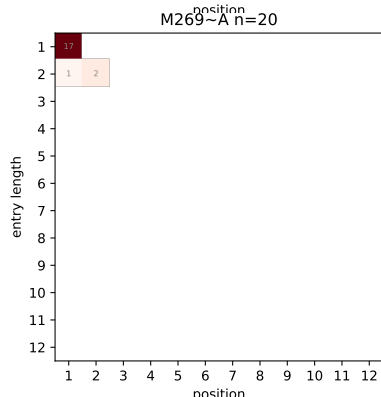
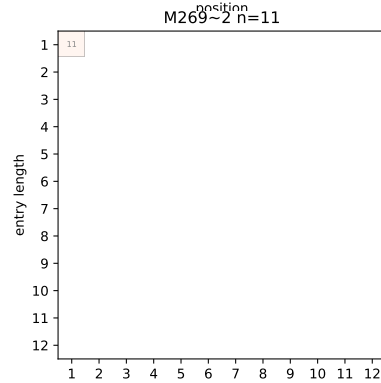
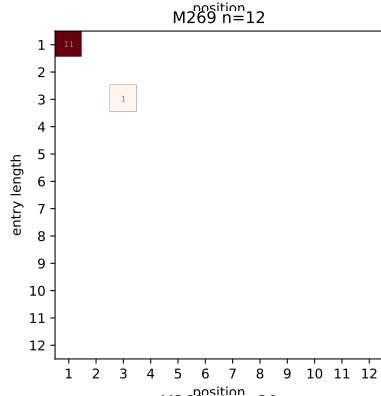
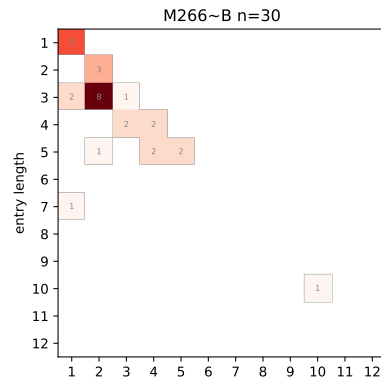
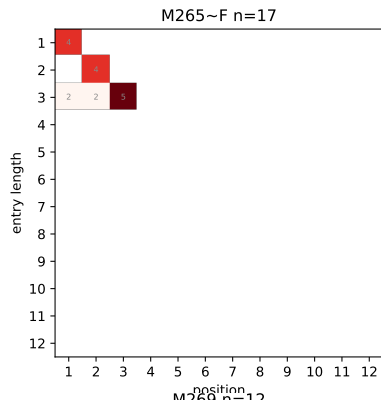


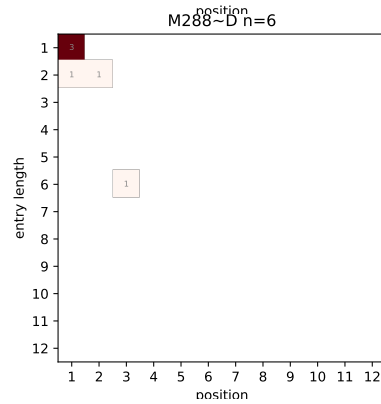
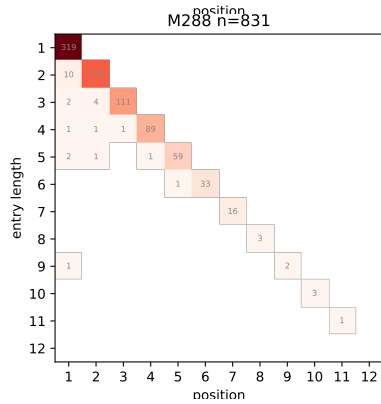
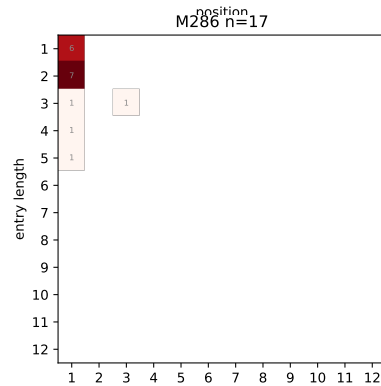
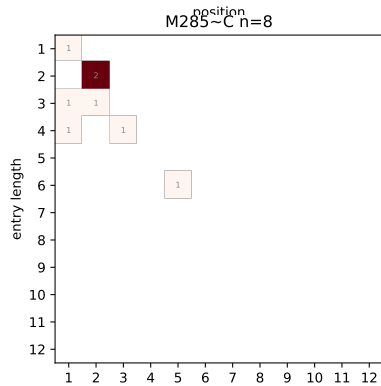
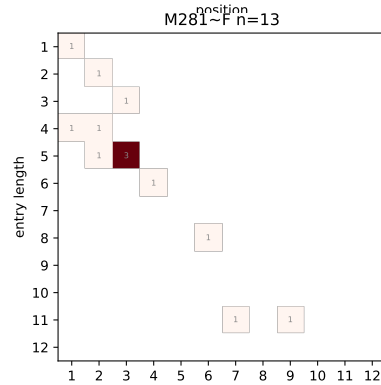
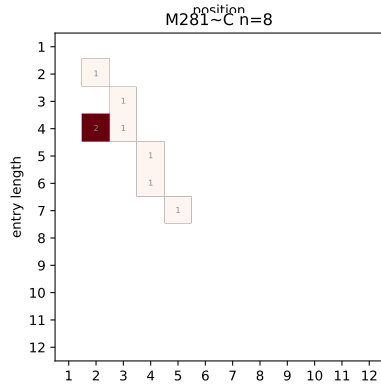
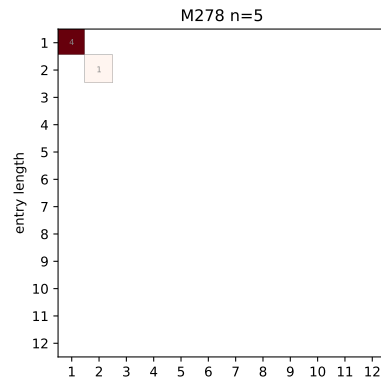
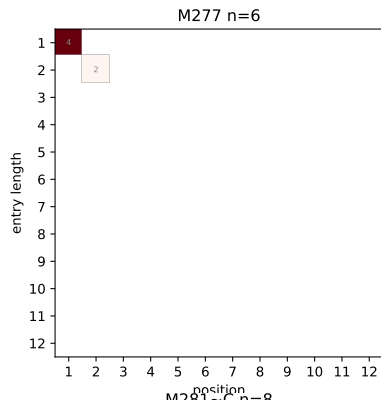


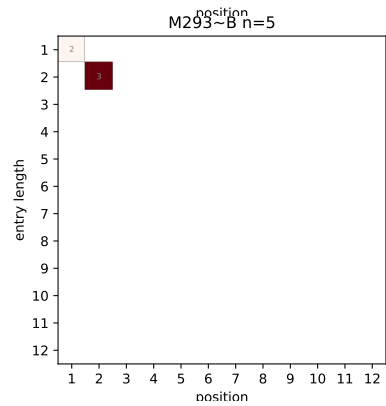
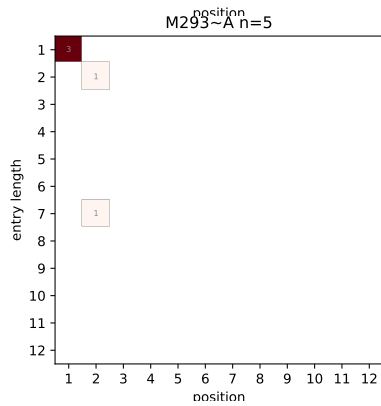
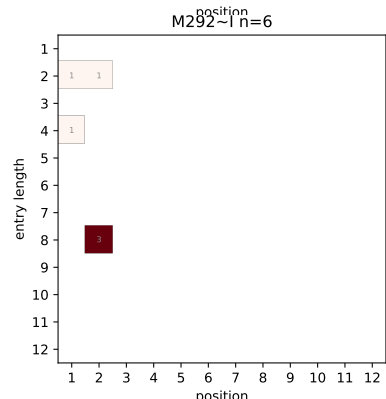
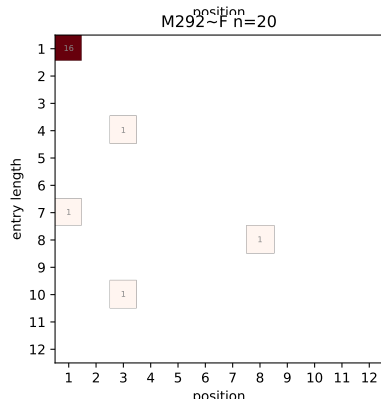
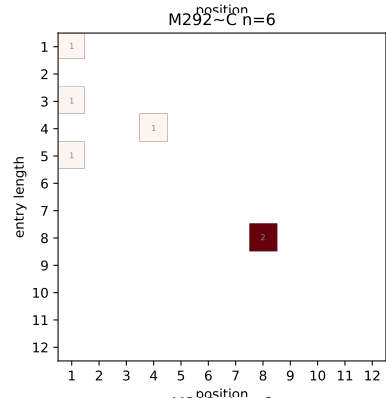
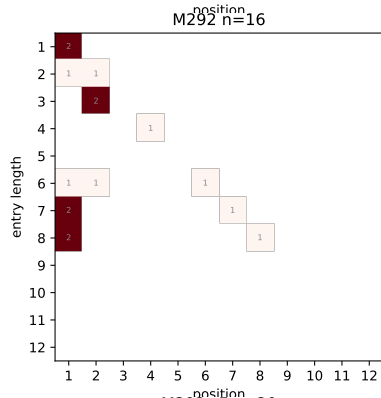
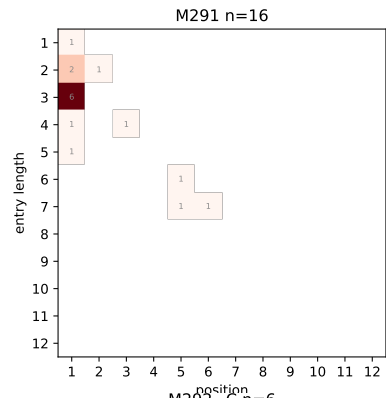
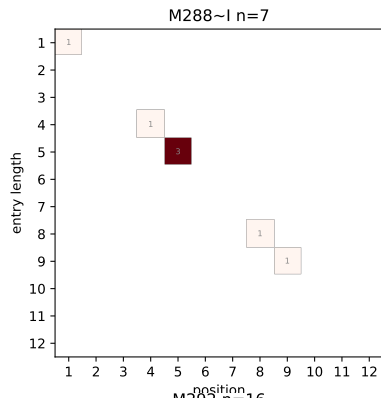


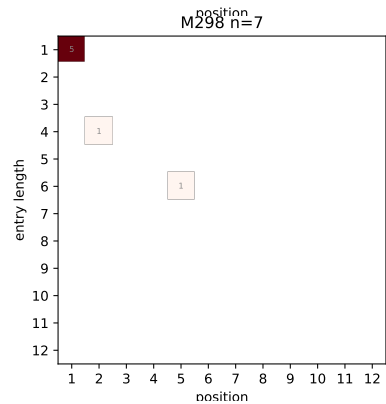
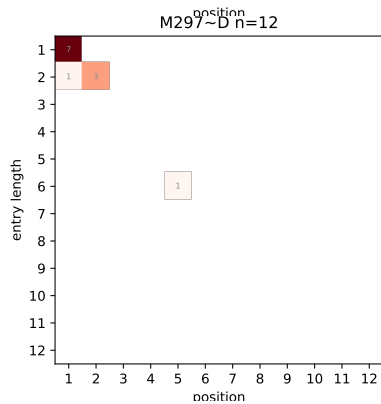
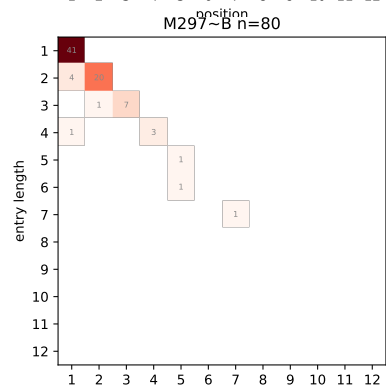
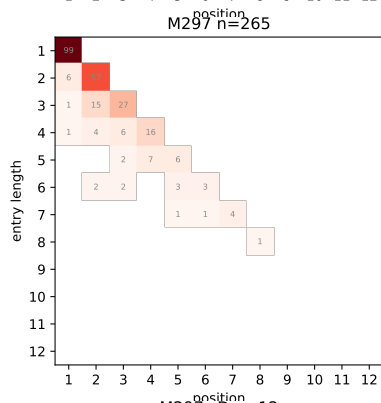
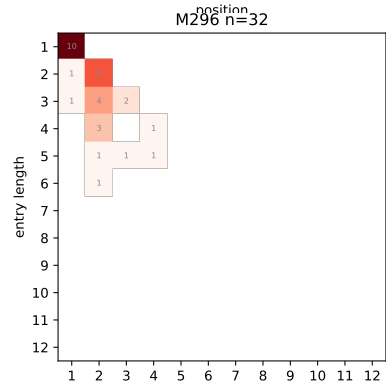
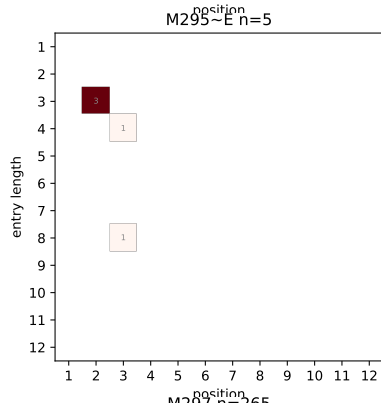
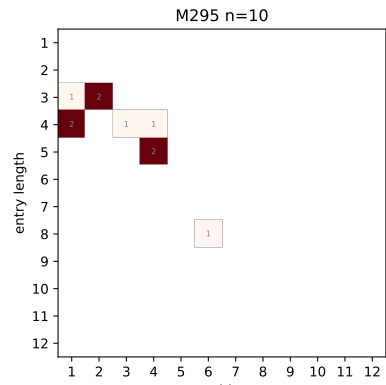
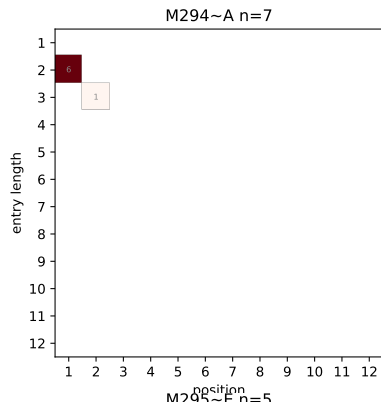


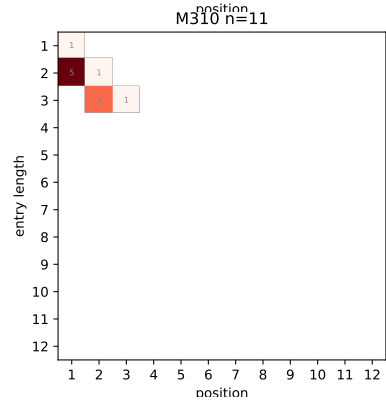
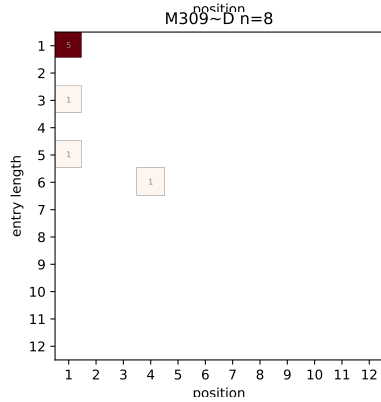
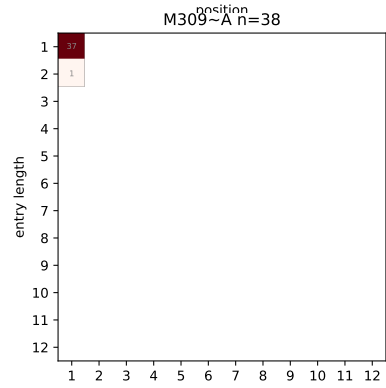
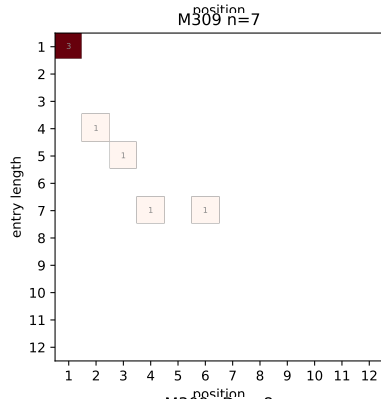
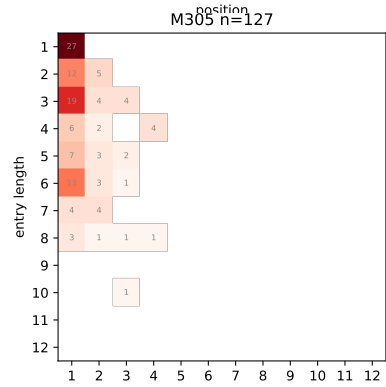
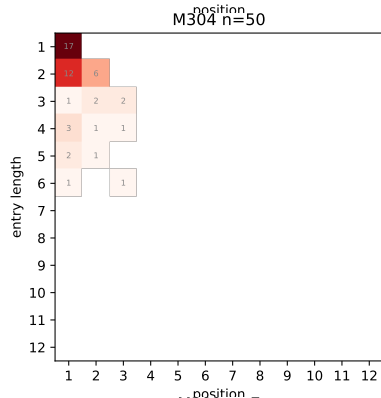
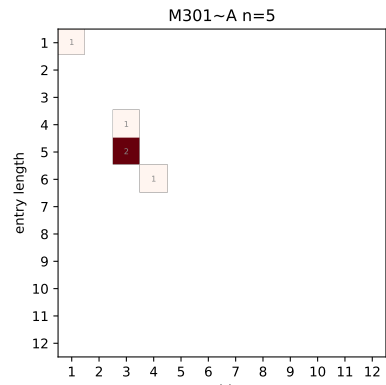
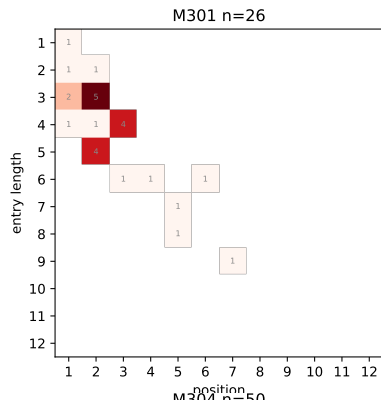






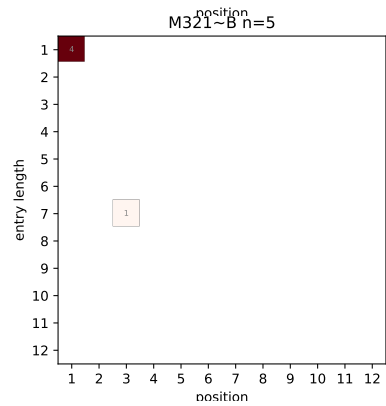
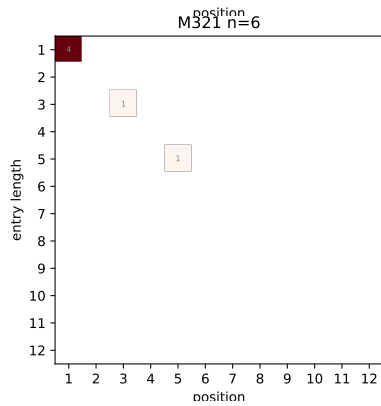
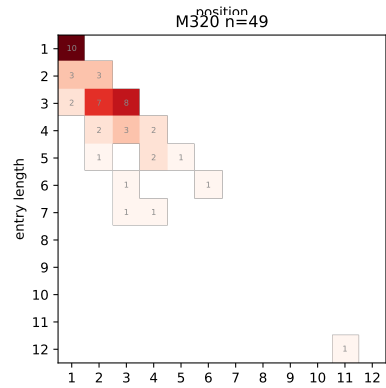
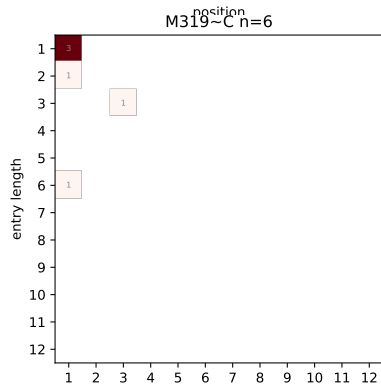
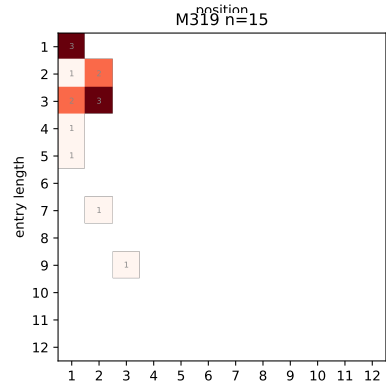
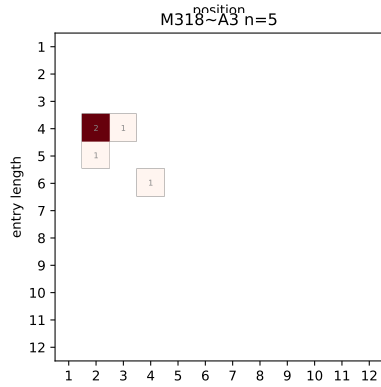
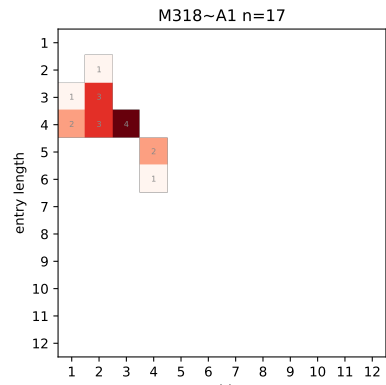
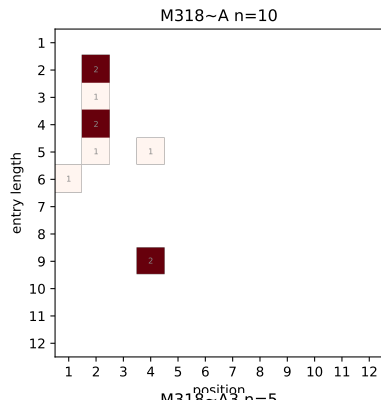


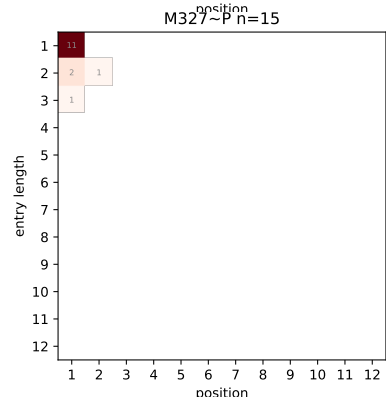
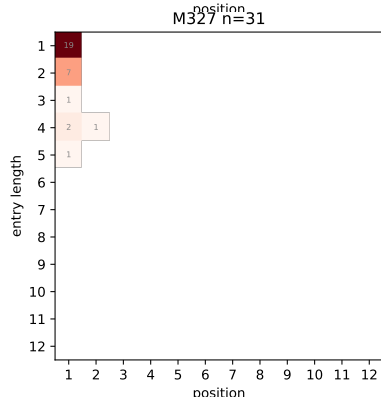
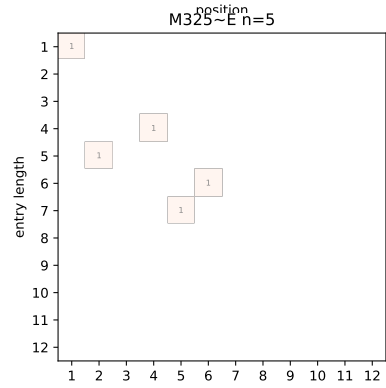
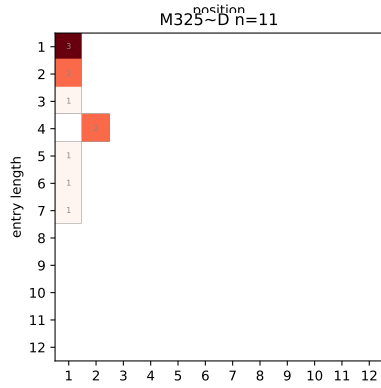
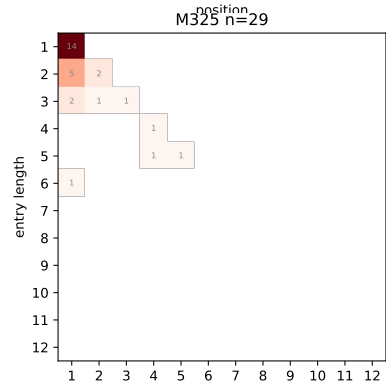
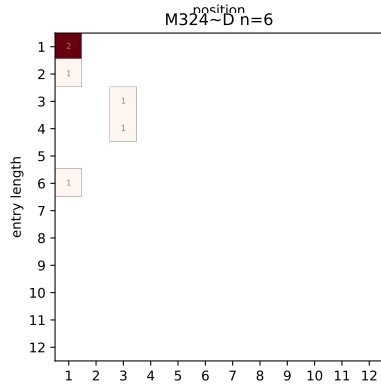
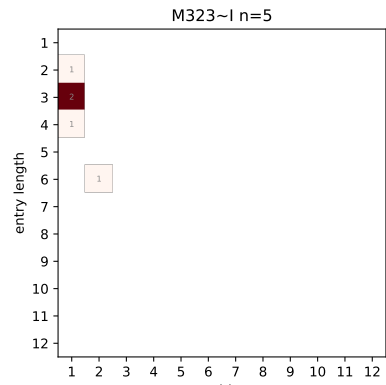
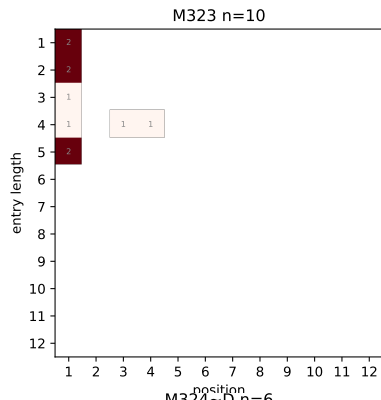


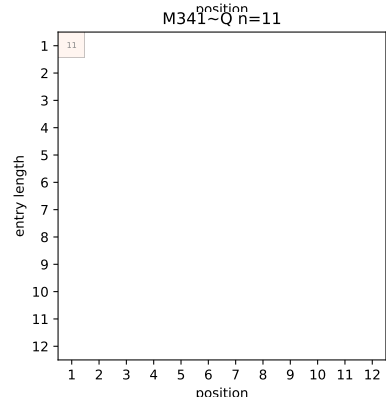
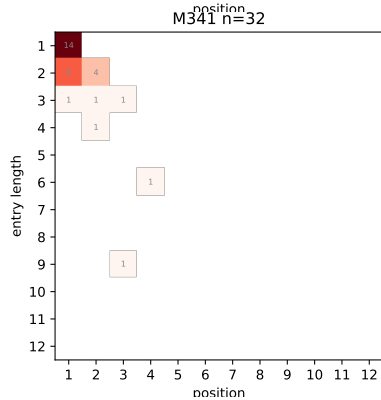
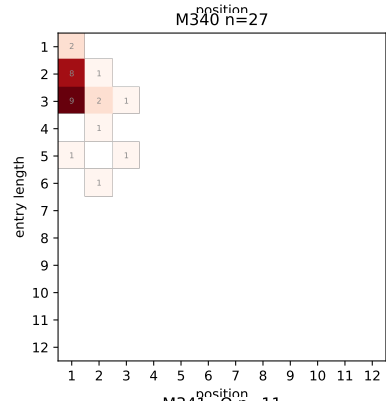
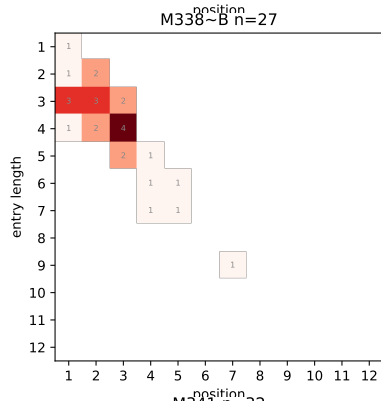
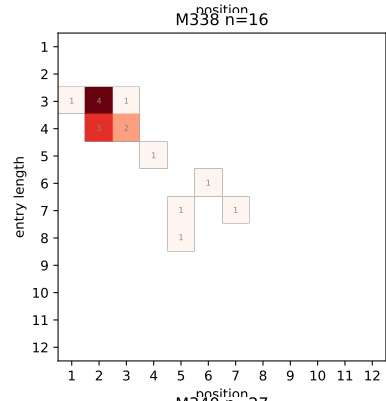
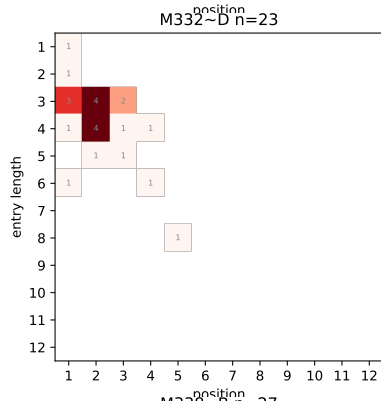
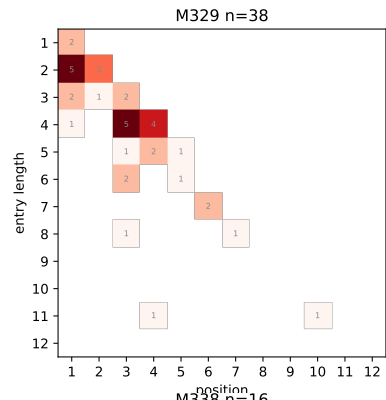
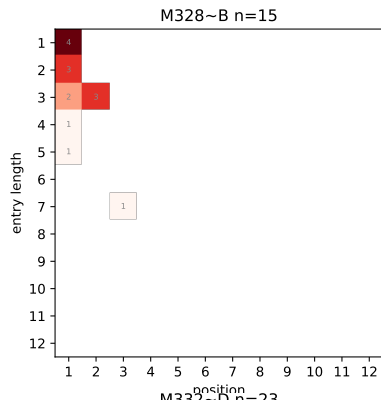


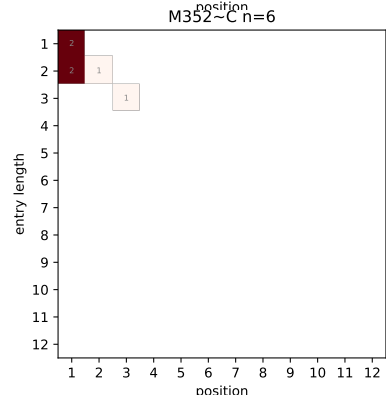
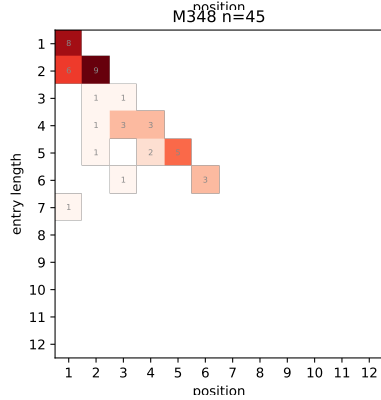
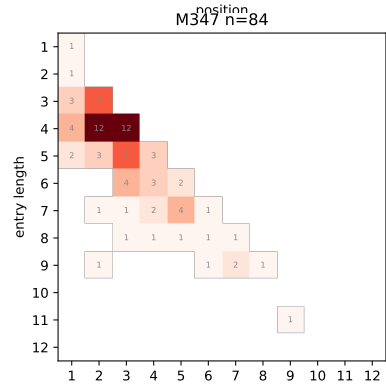
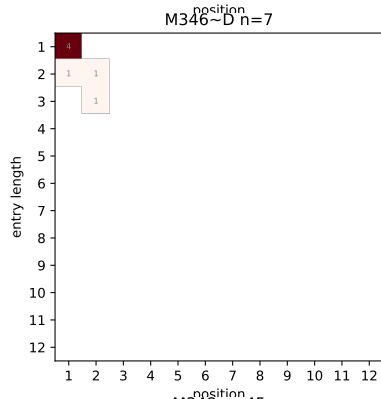
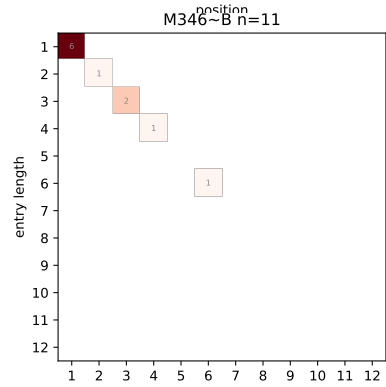
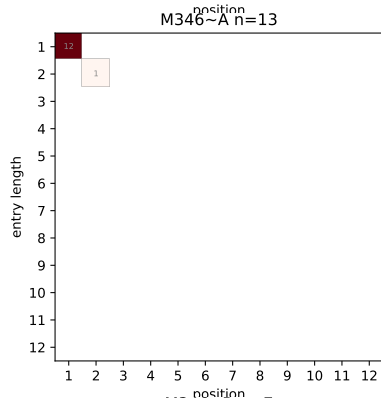
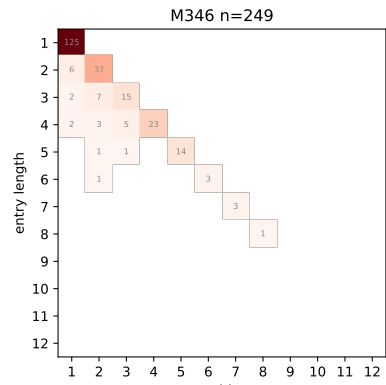
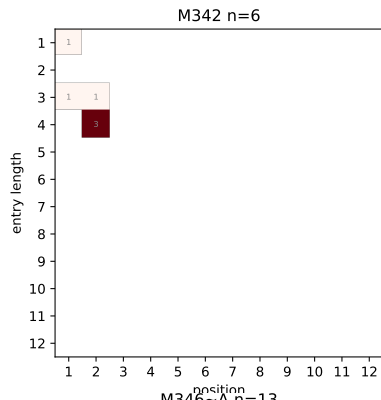


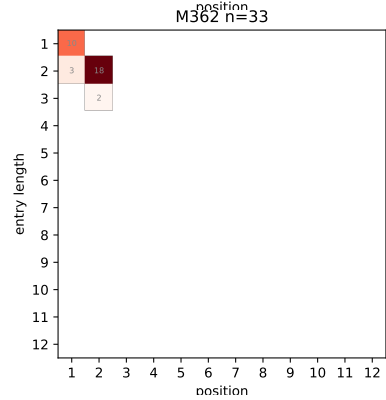
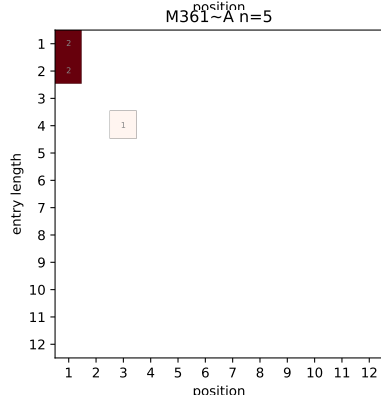
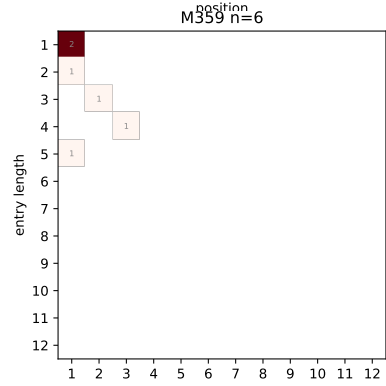
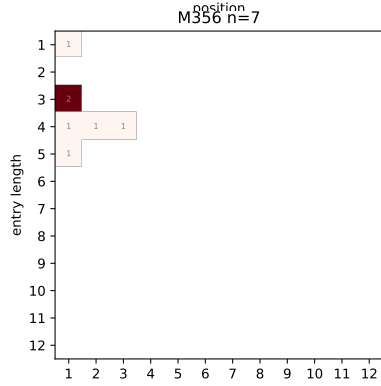
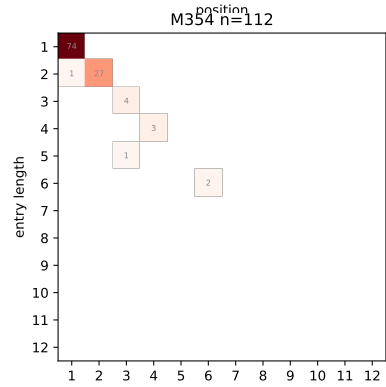
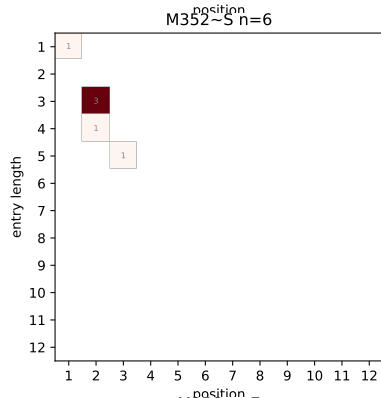
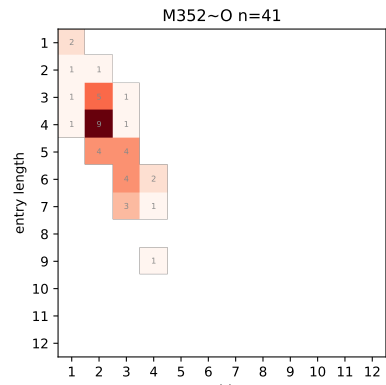
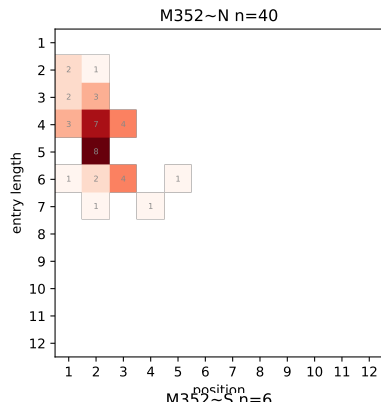


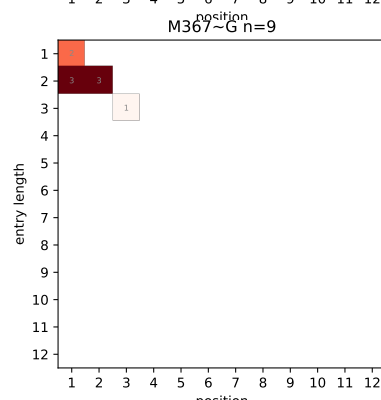
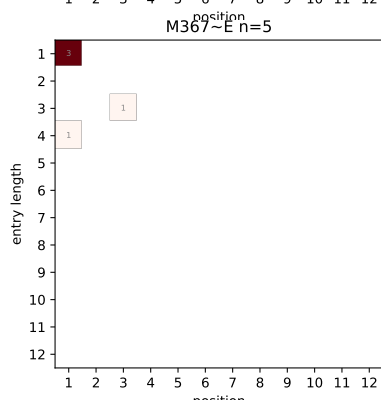
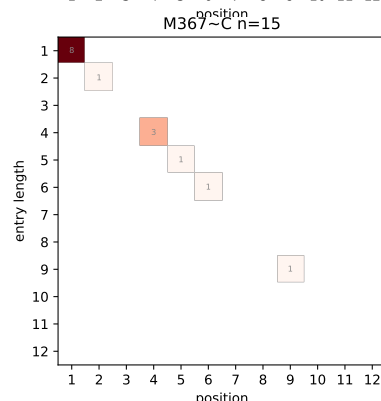
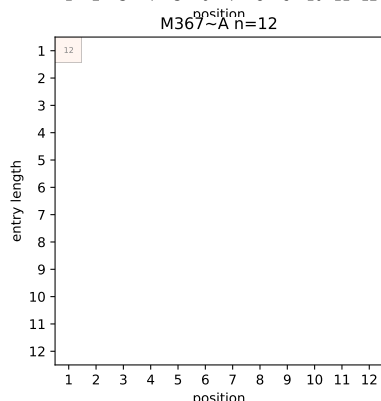
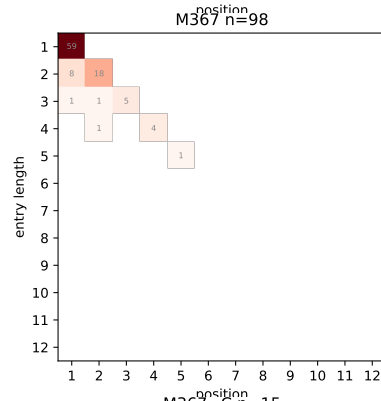
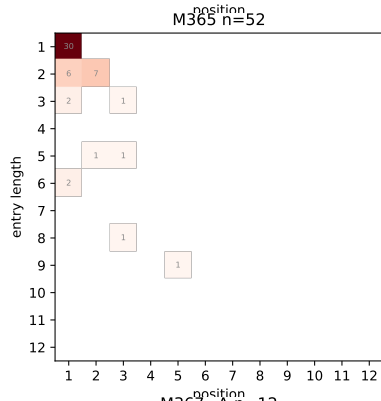
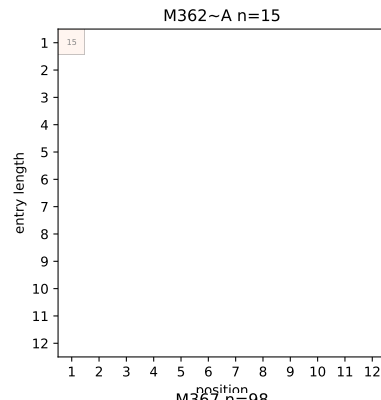
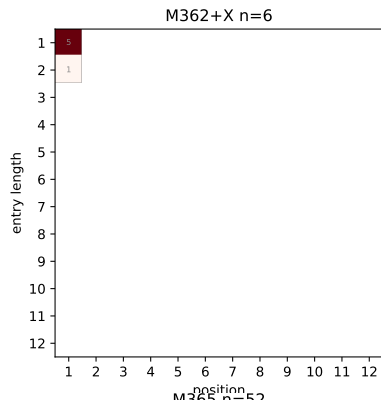


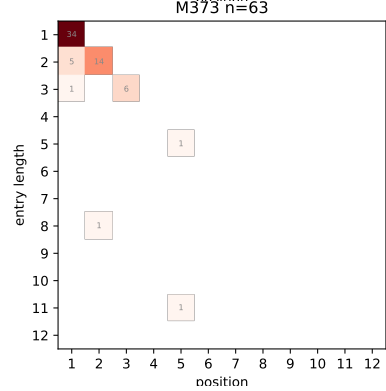
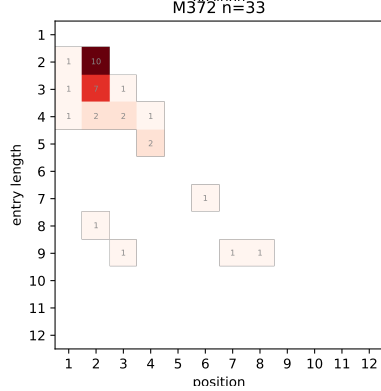
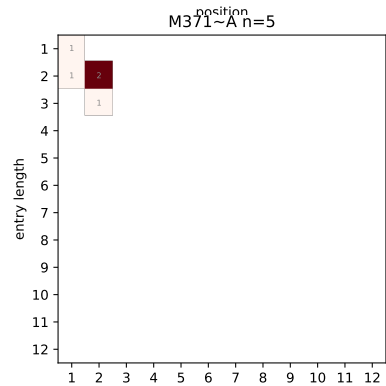
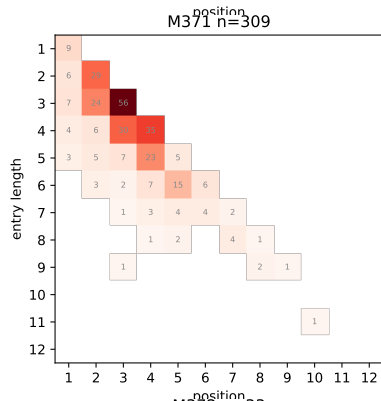
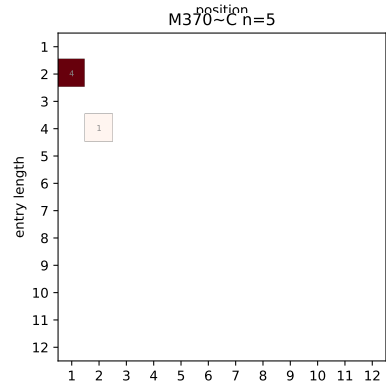
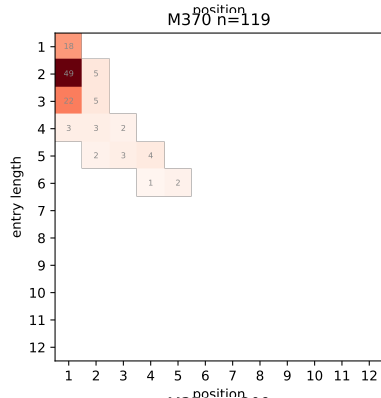
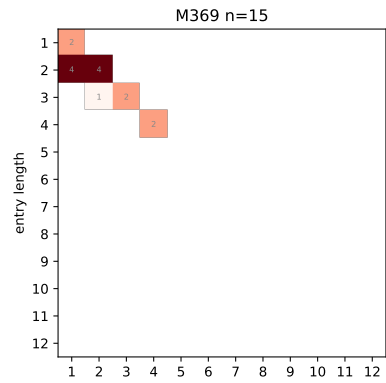
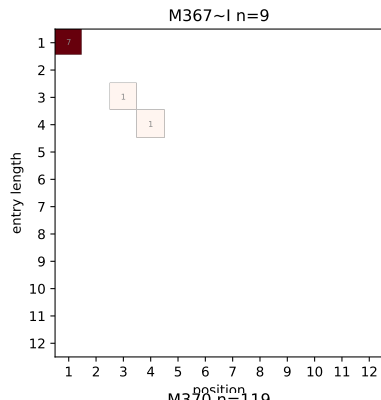




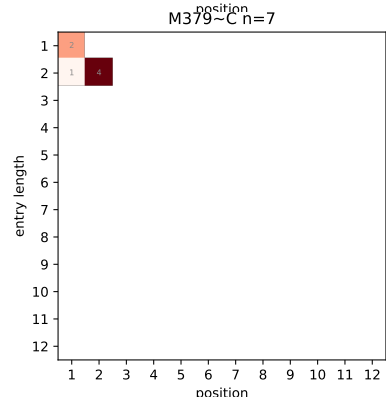
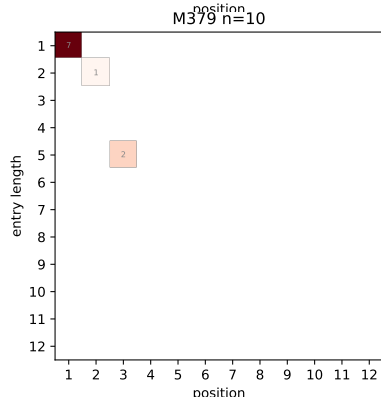
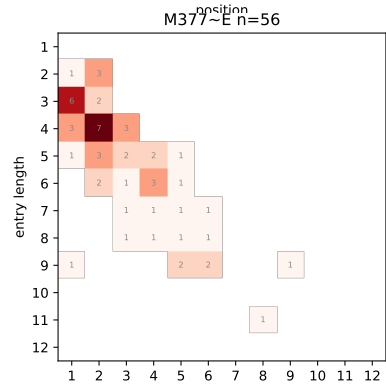
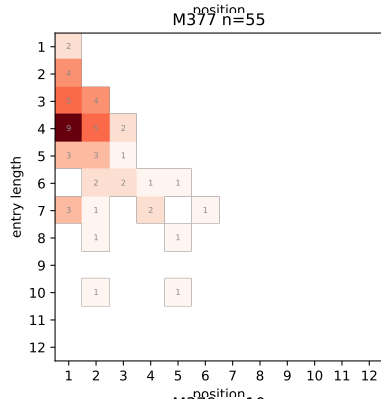
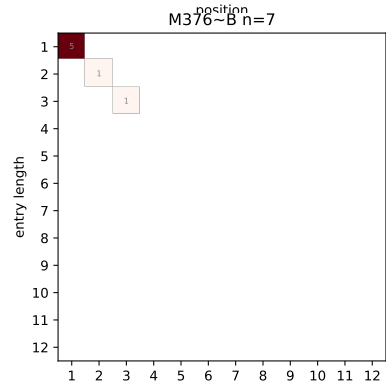
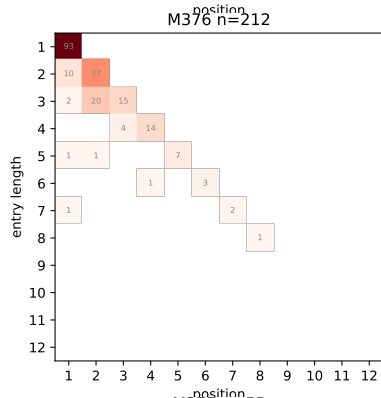
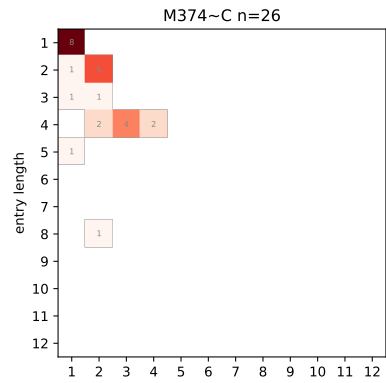
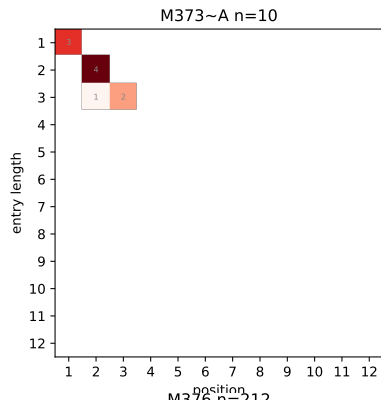


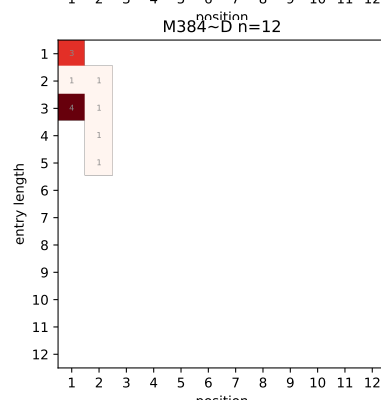
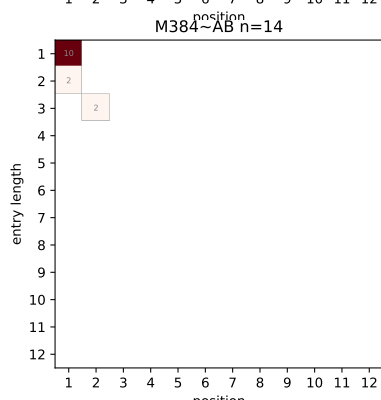
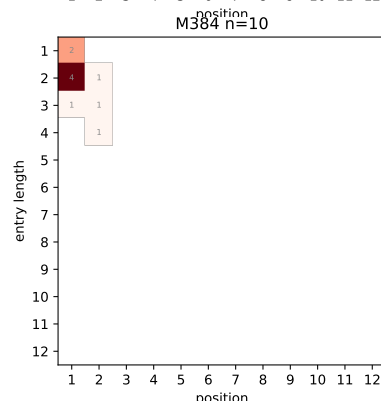
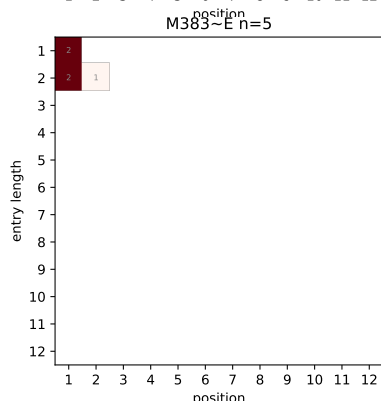
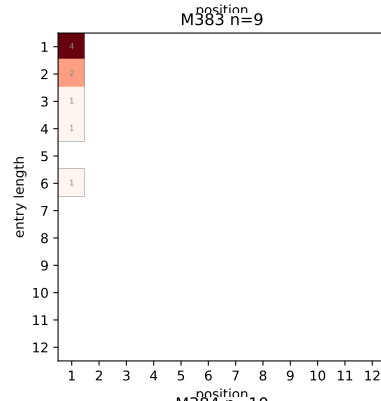
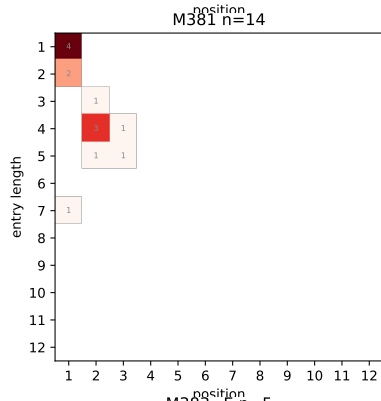
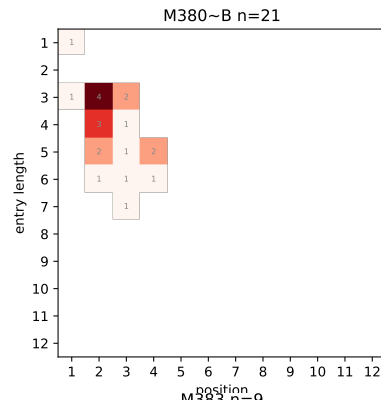
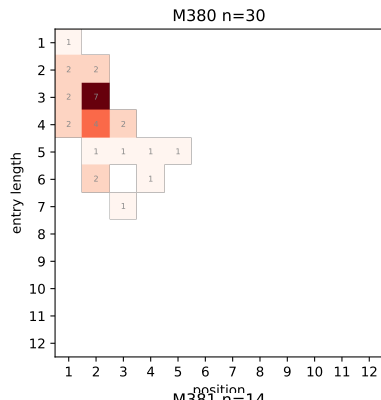


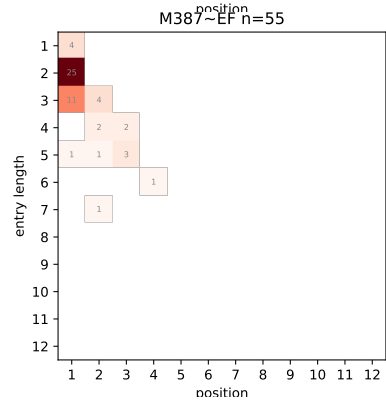
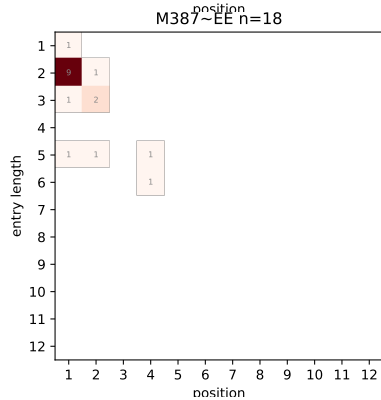
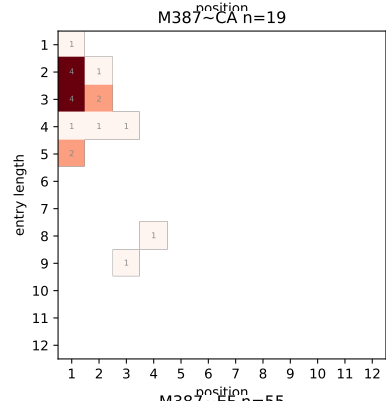
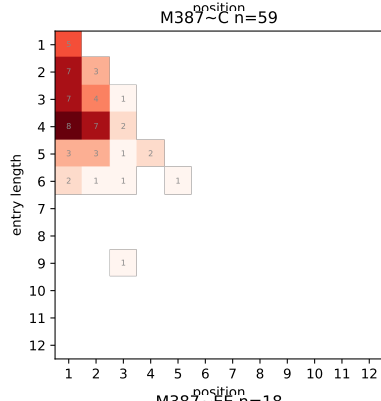
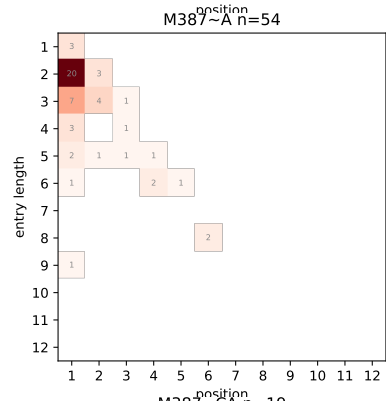
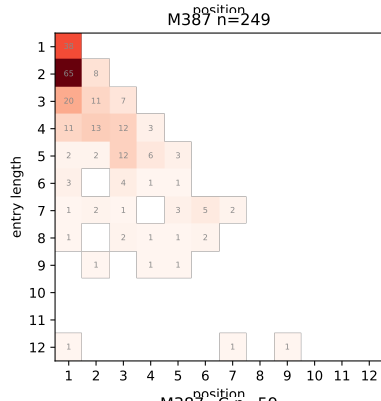
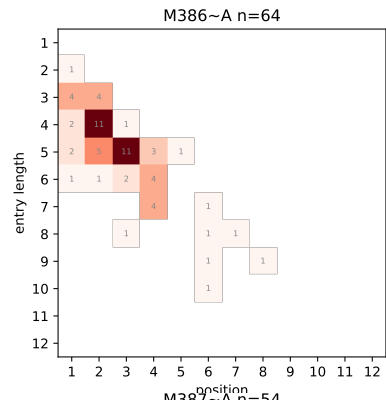
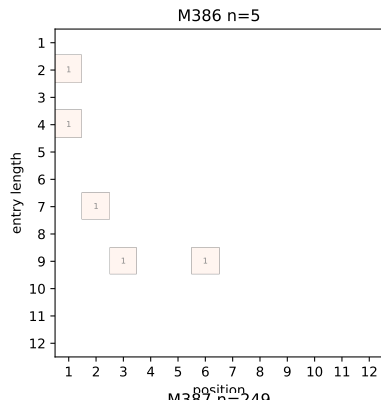


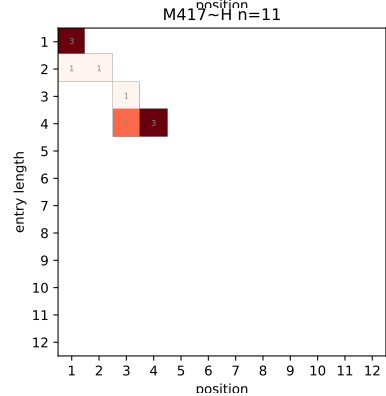
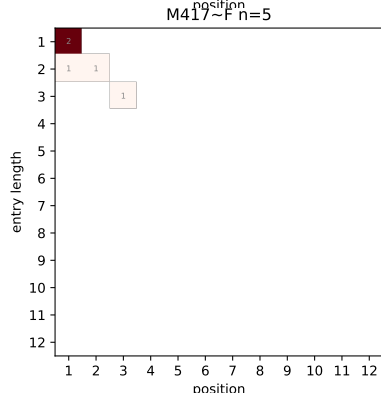
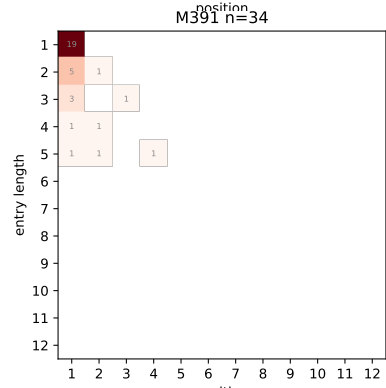
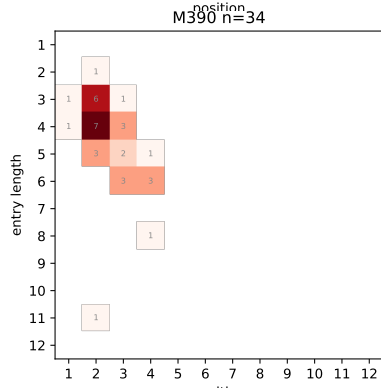
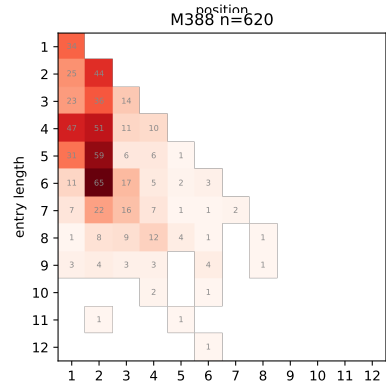
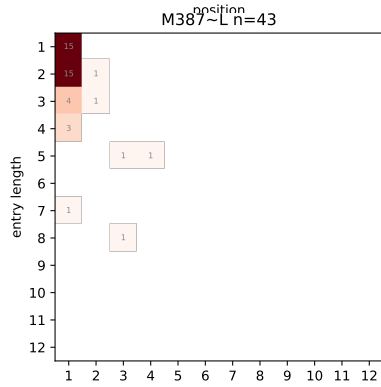
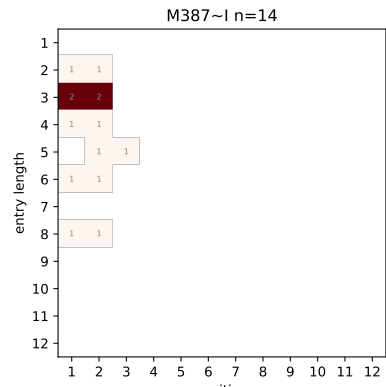
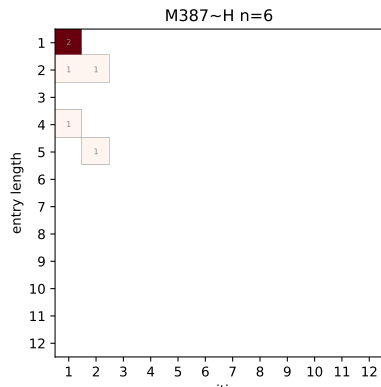


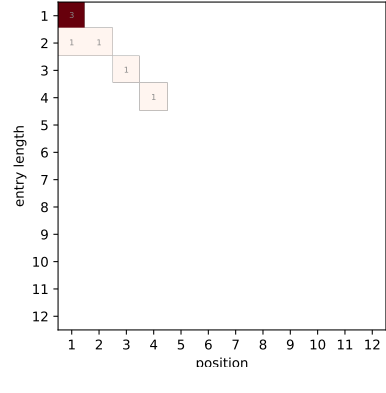
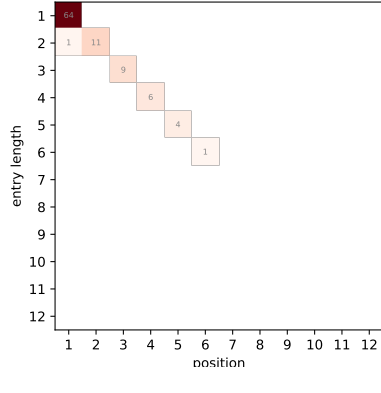
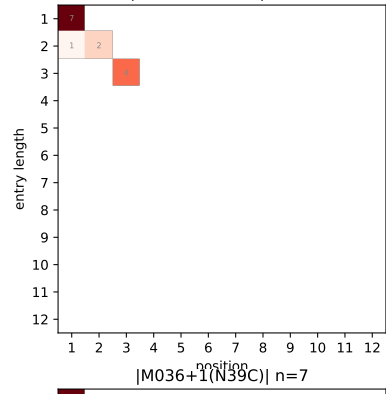
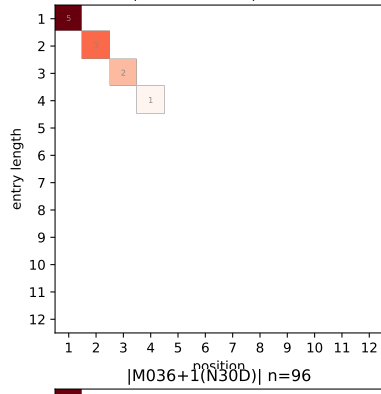
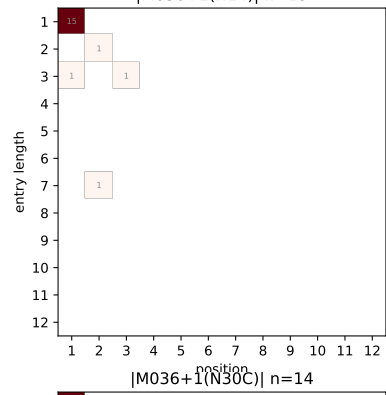
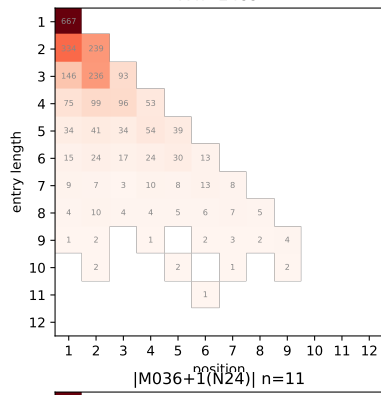
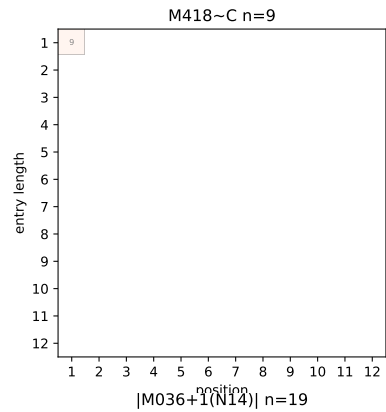
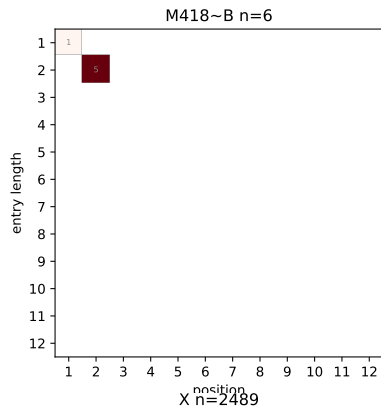


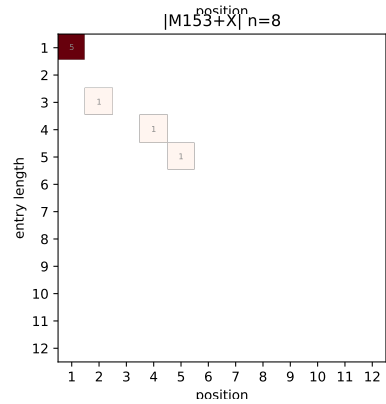
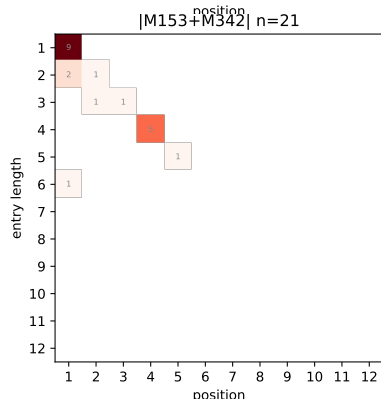
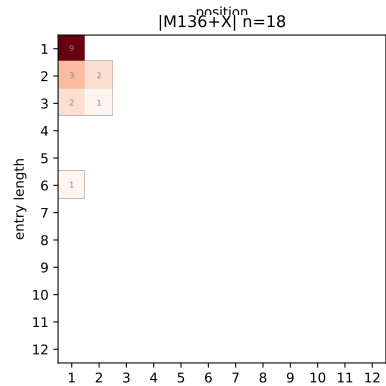
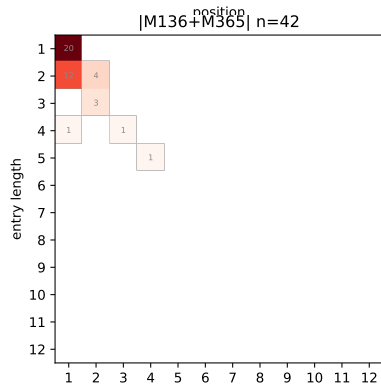
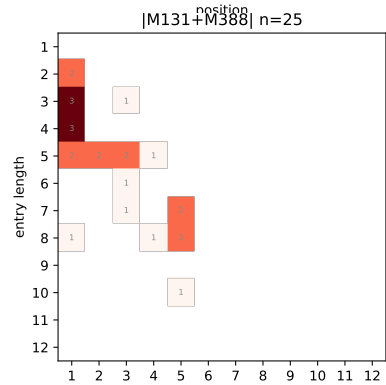
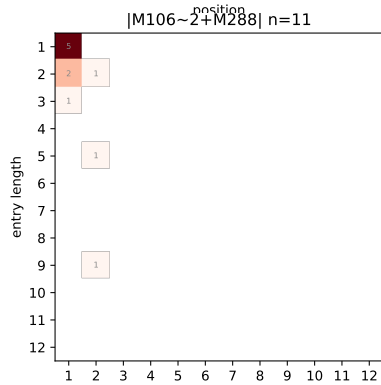
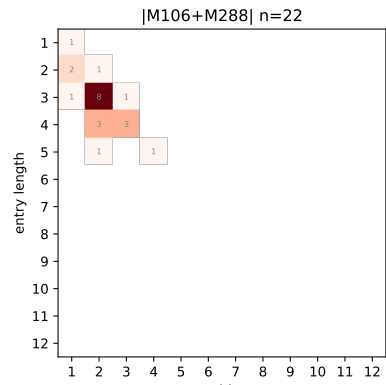
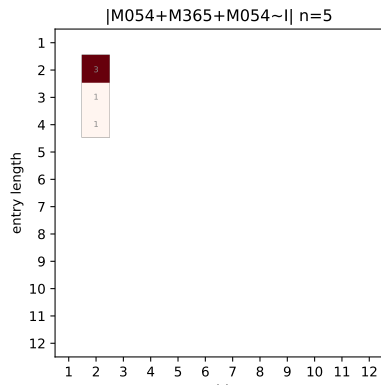


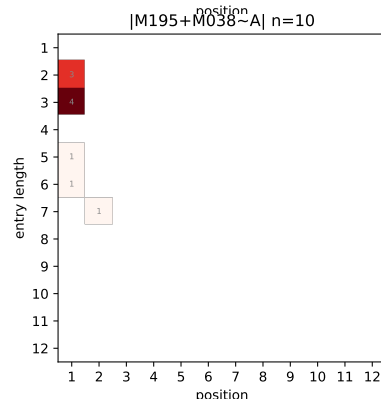
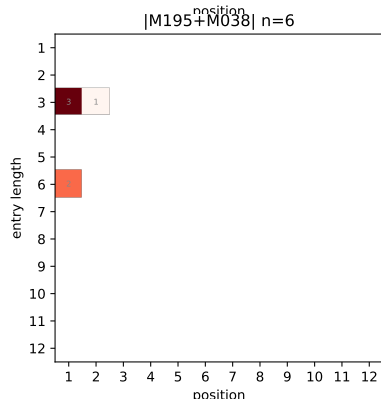
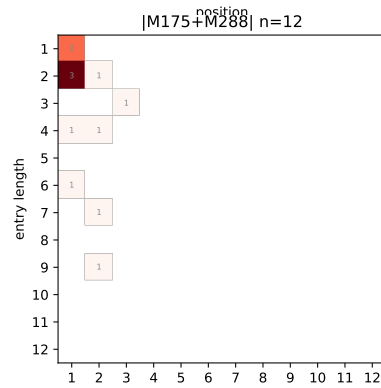
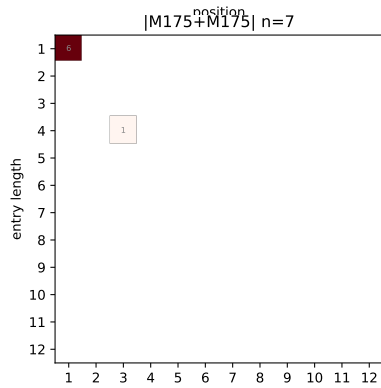
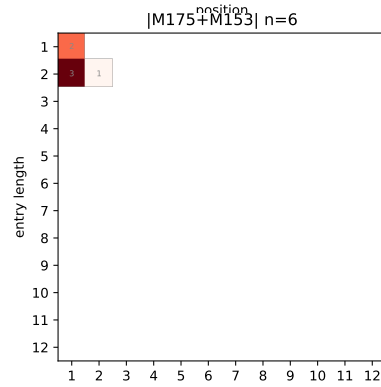
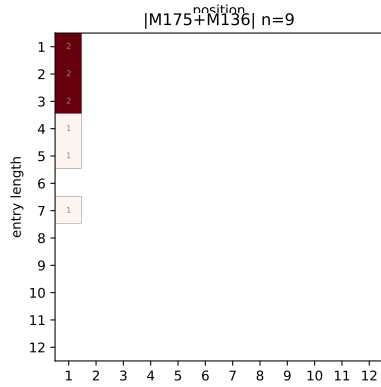
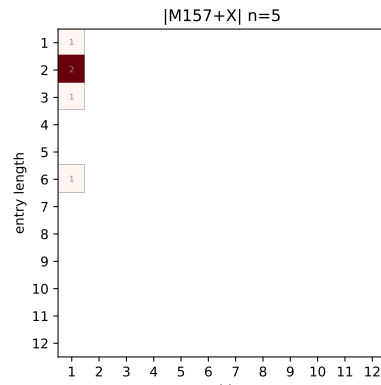
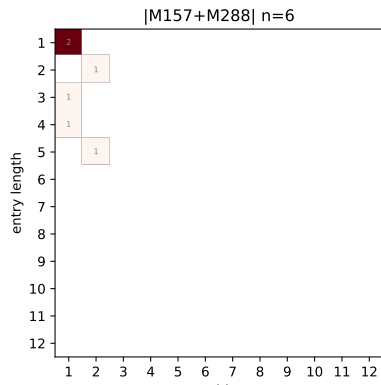


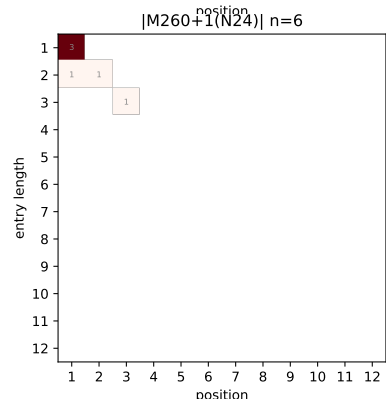
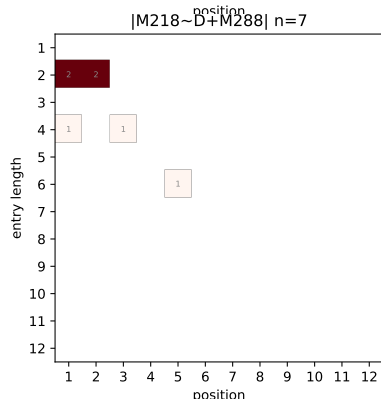
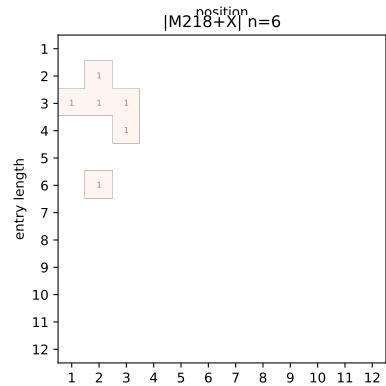
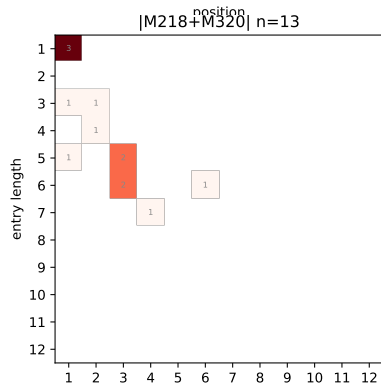
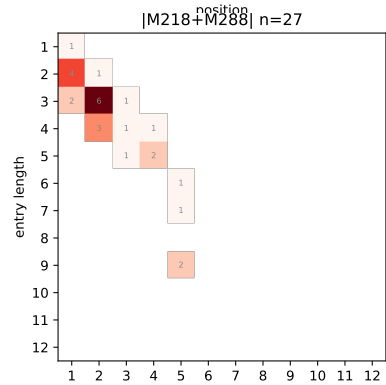
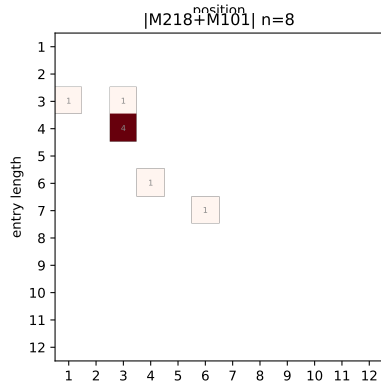
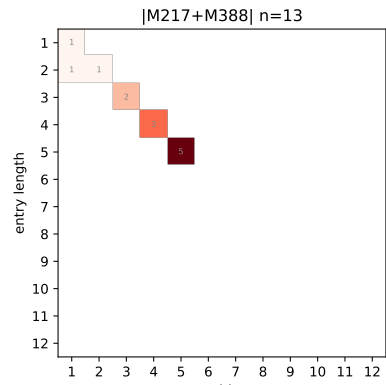
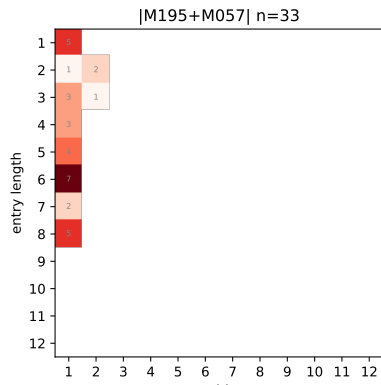




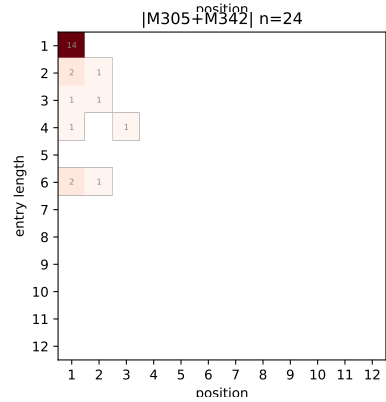
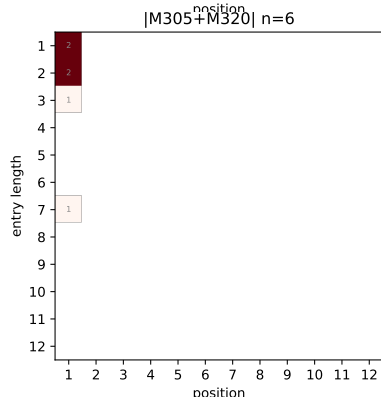
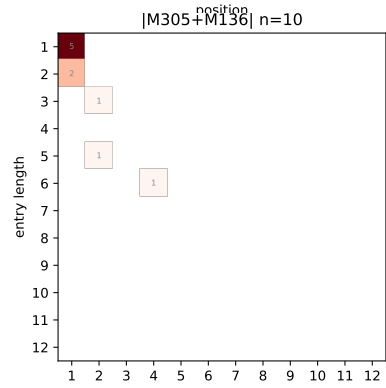
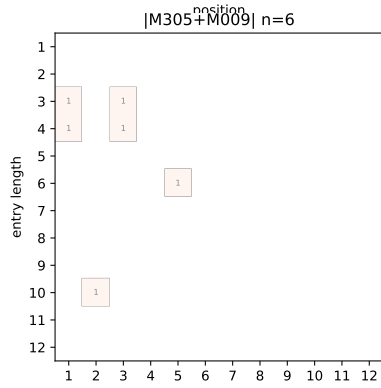
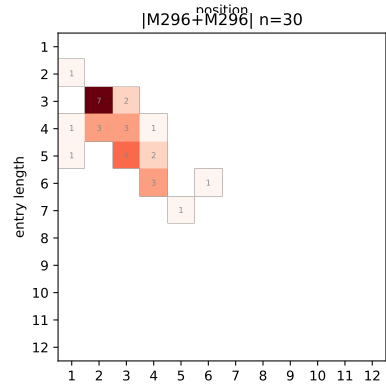
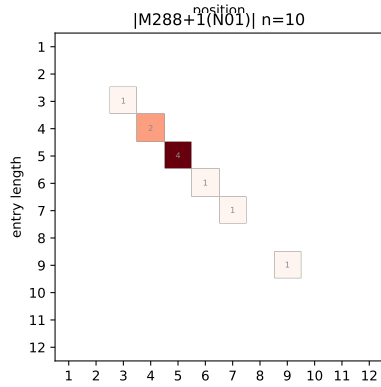
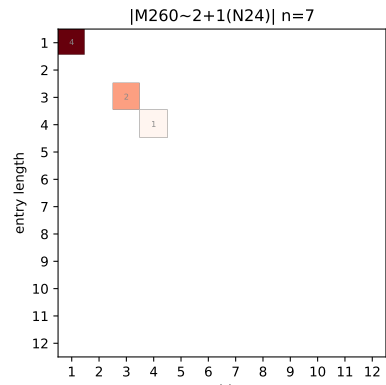
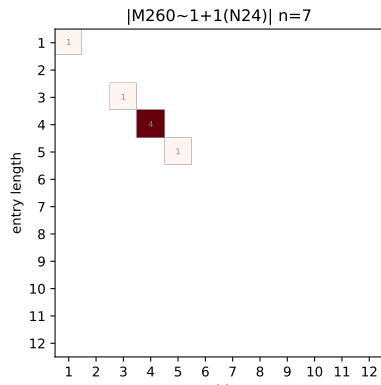


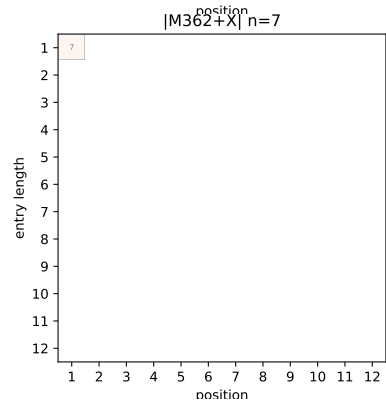
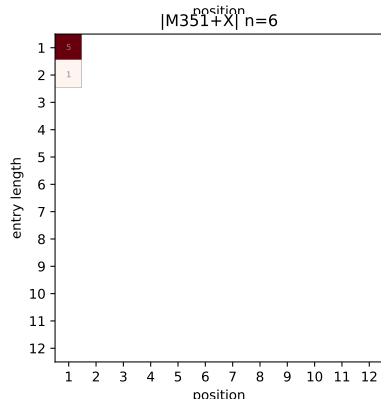
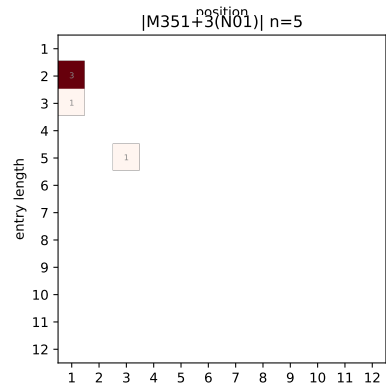
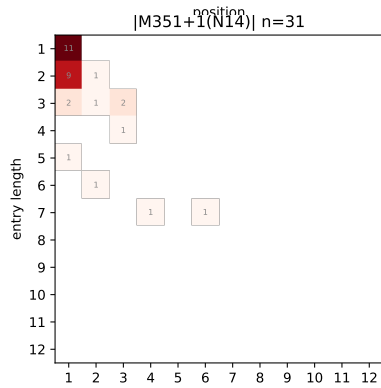
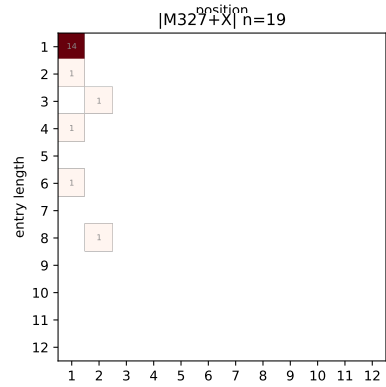
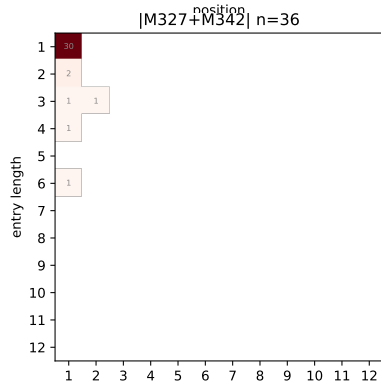
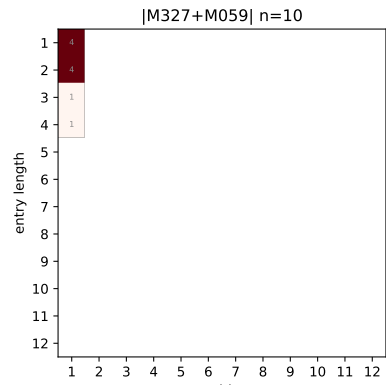
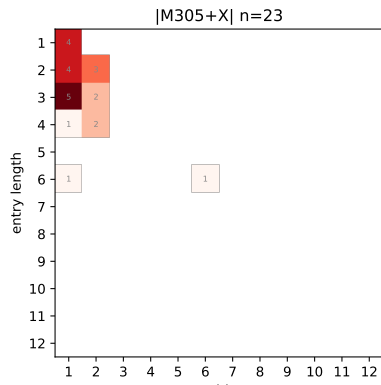


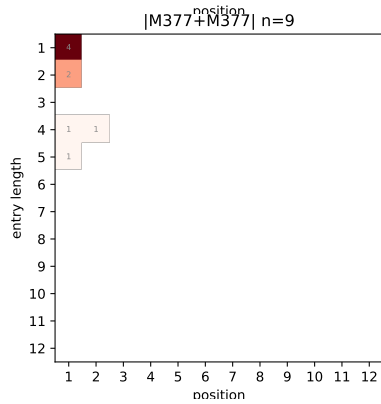
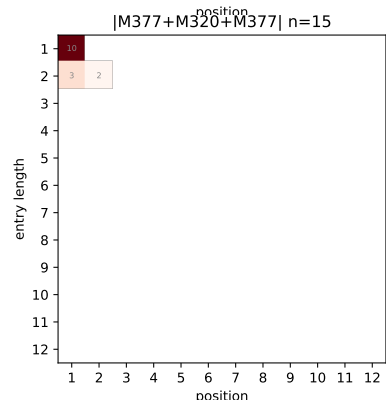
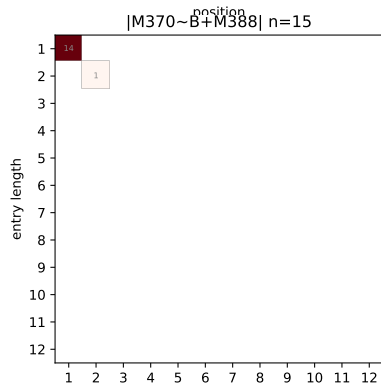
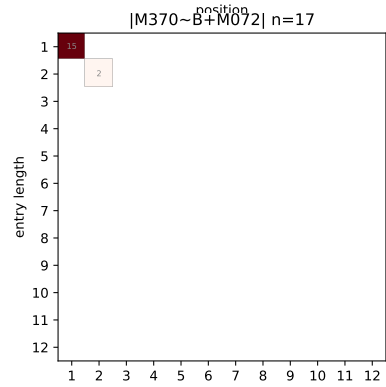
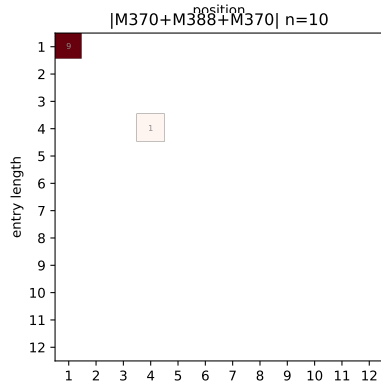
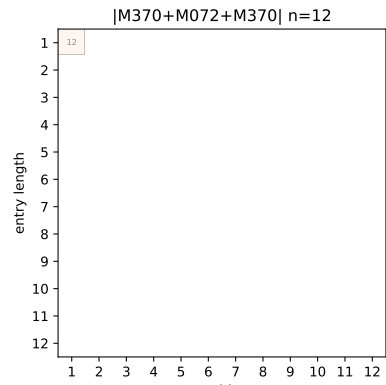
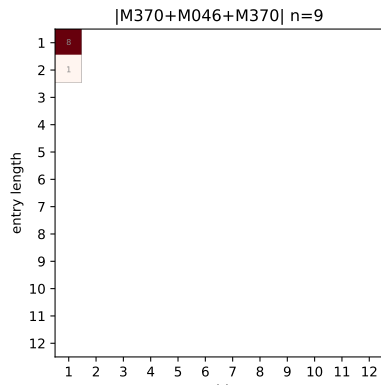












## A.5 Complex Grapheme Embedding Spaces

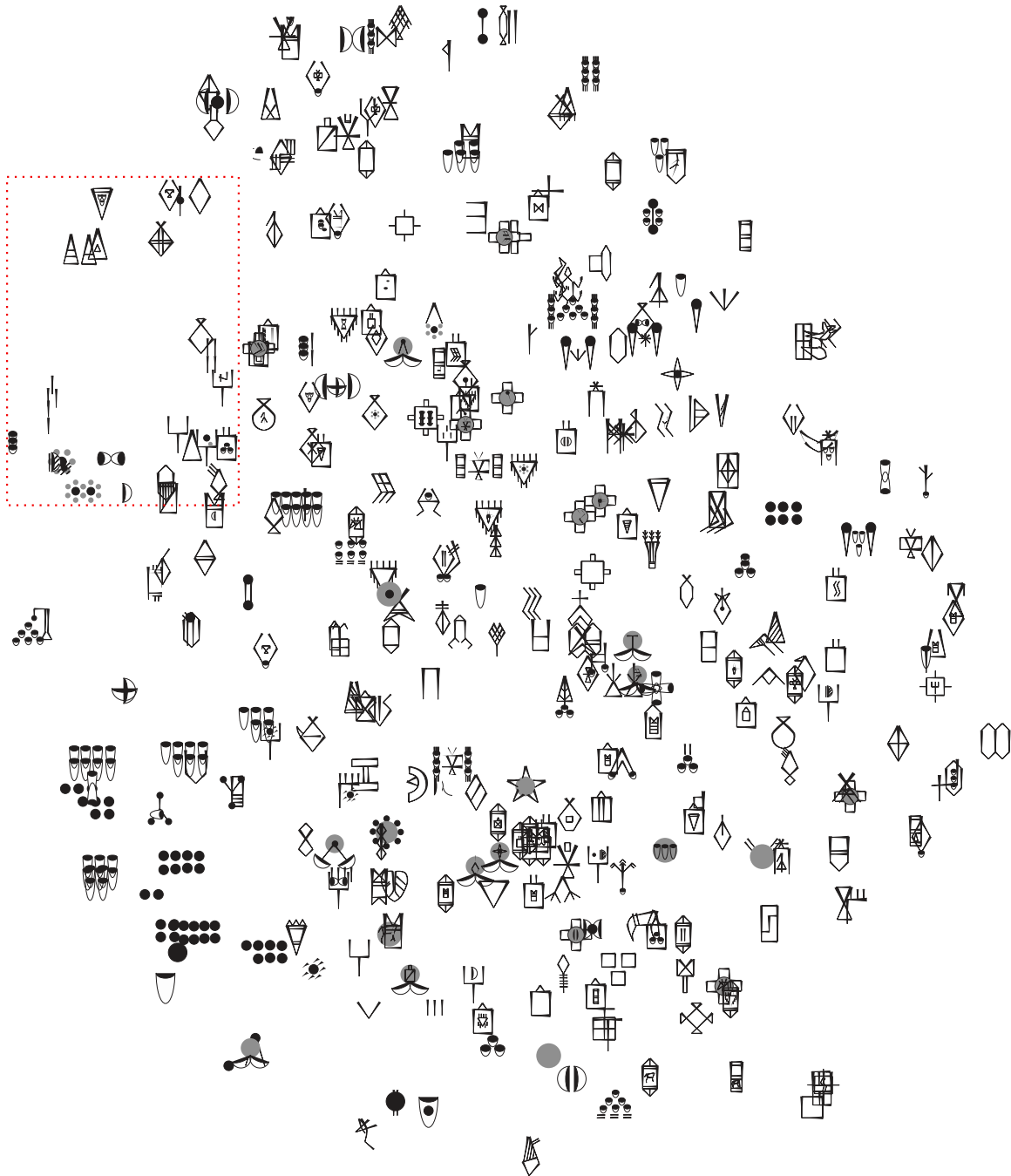


Figure A.5: t-SNE projection for the complete embedding space from the `glove.64` model from Chapter 6. Figure 6.3 excerpt shown in red.

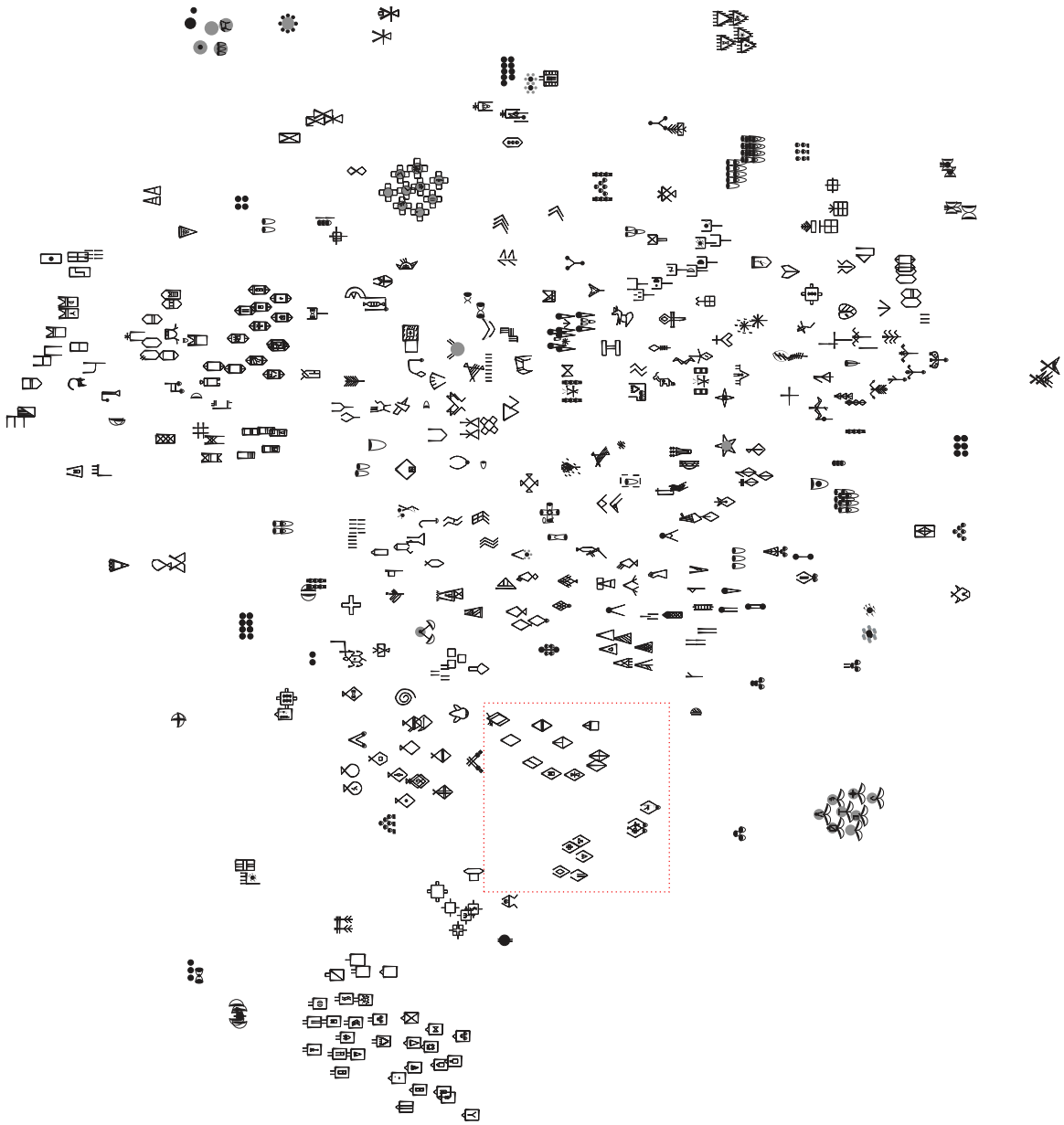


Figure A.6: t-SNE projection for the complete embedding space from the `image_recognition.64` model from Chapter 6. Figure 6.3 excerpt shown in red.

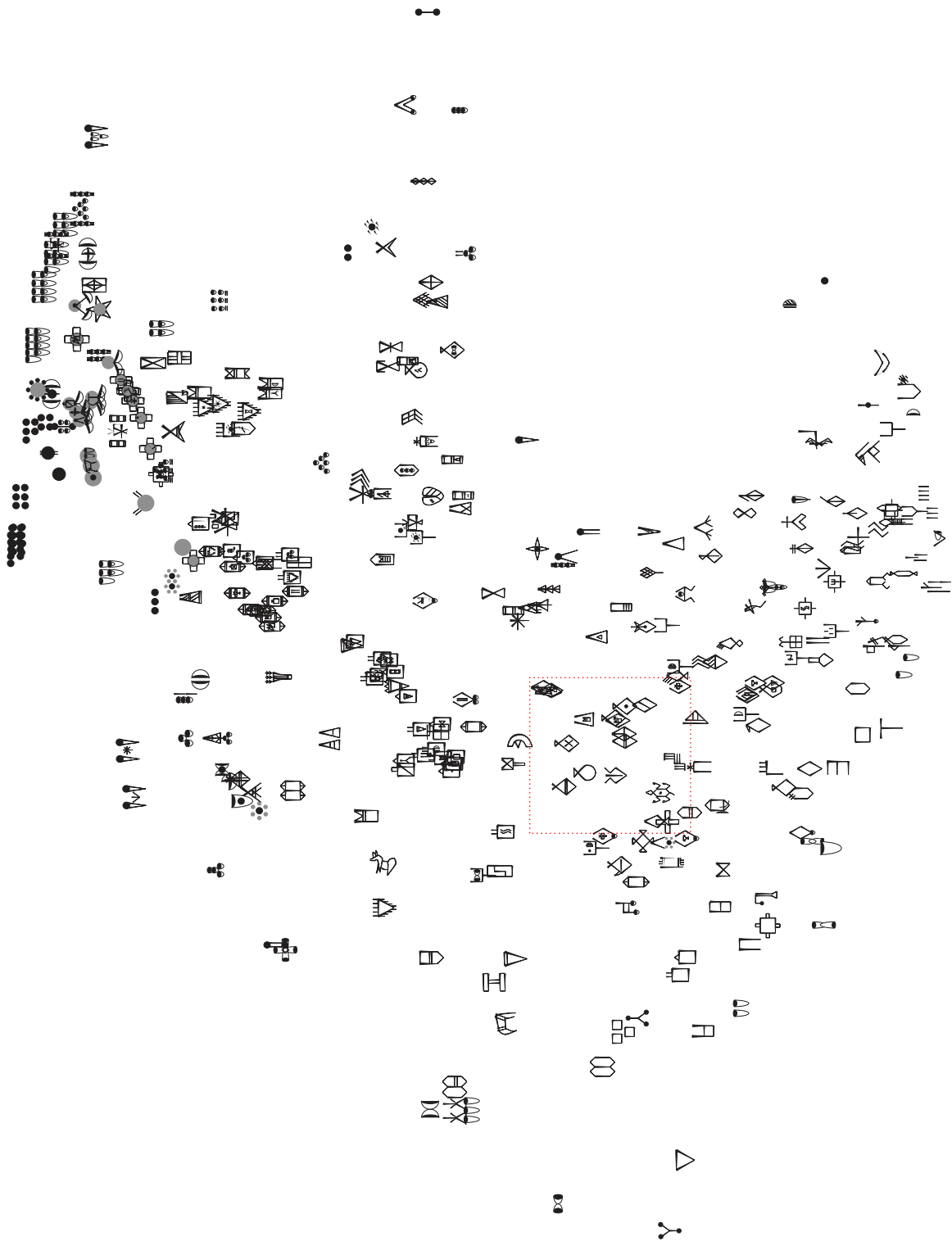


Figure A.7: t-SNE projection for the complete embedding space from the `lm.image.64` model from Chapter 6. Figure 6.3 excerpt shown in red.

## A.6 Complex Grapheme Containment Hierarchy

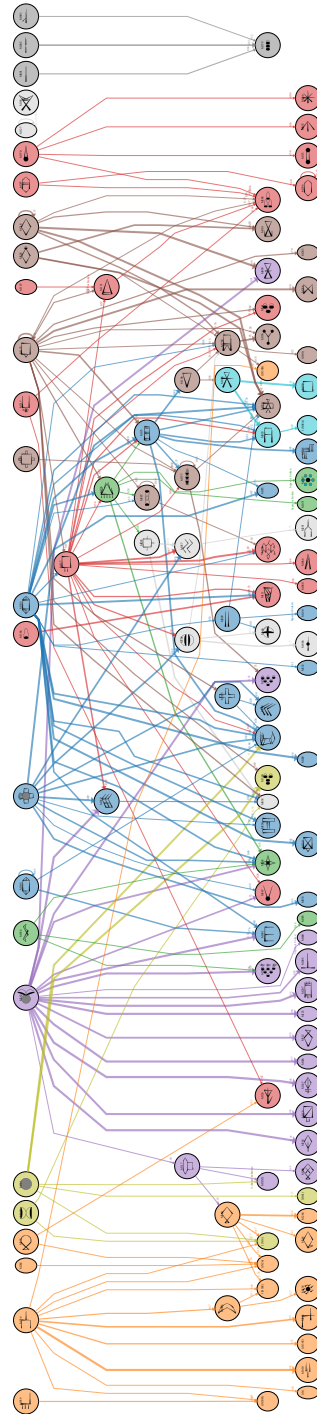


Figure A.8: Complete containment hierarchy which appears to govern the construction of complex graphemes (Chapter 6). (Figure is zoomable in digital editions of this work.)

## A.7 Sign Clustering

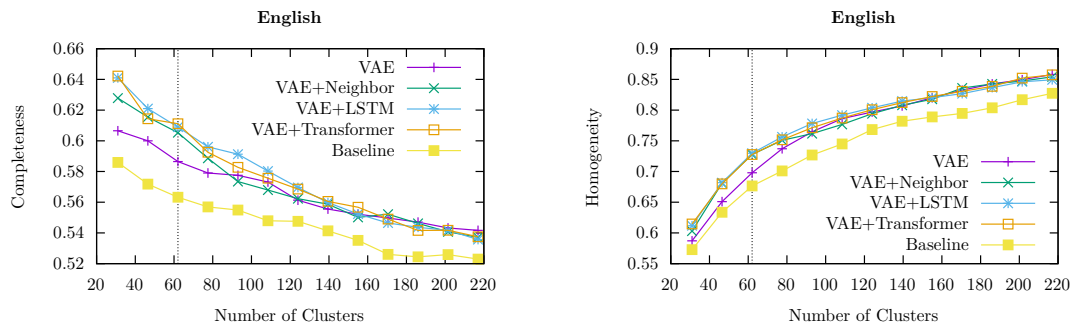


Figure A.9: Completeness and homogeneity for different clustering sizes using English models from Chapter 9

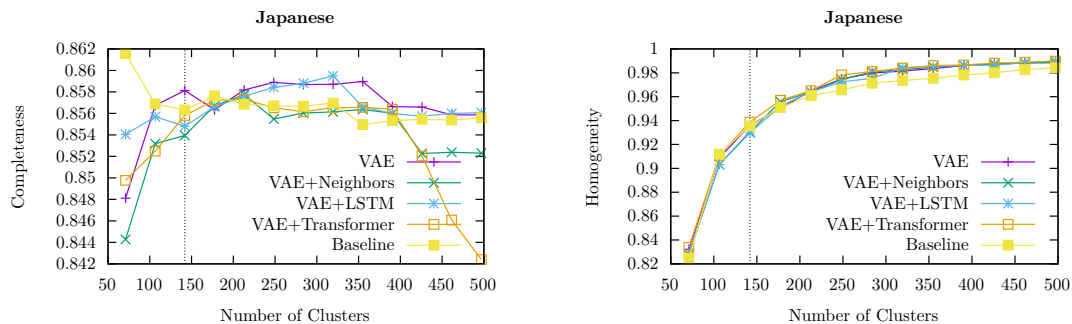
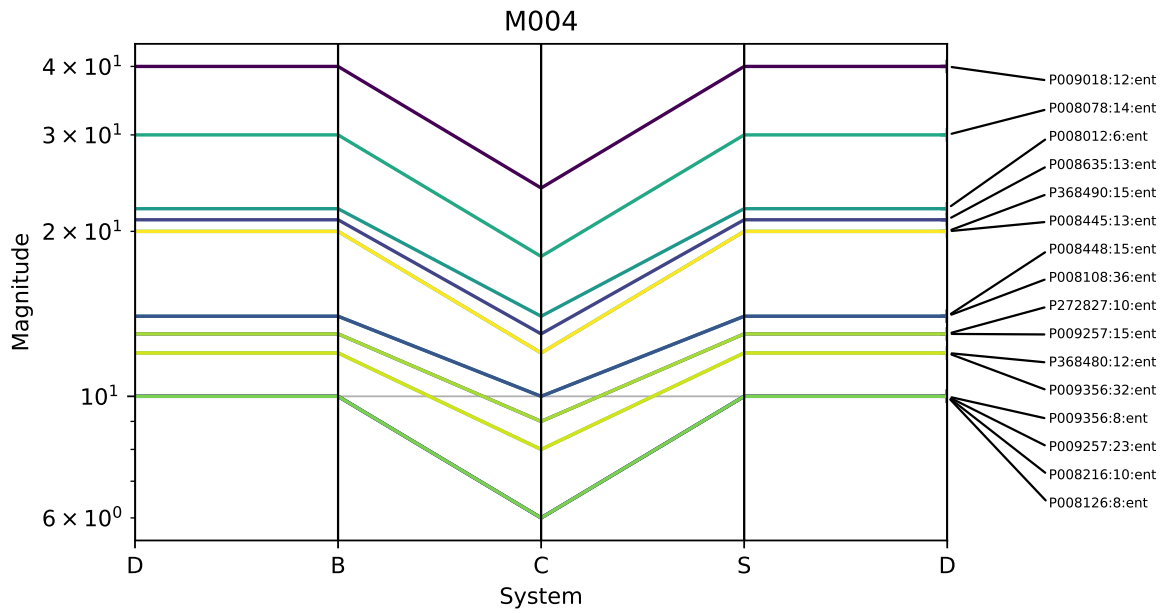
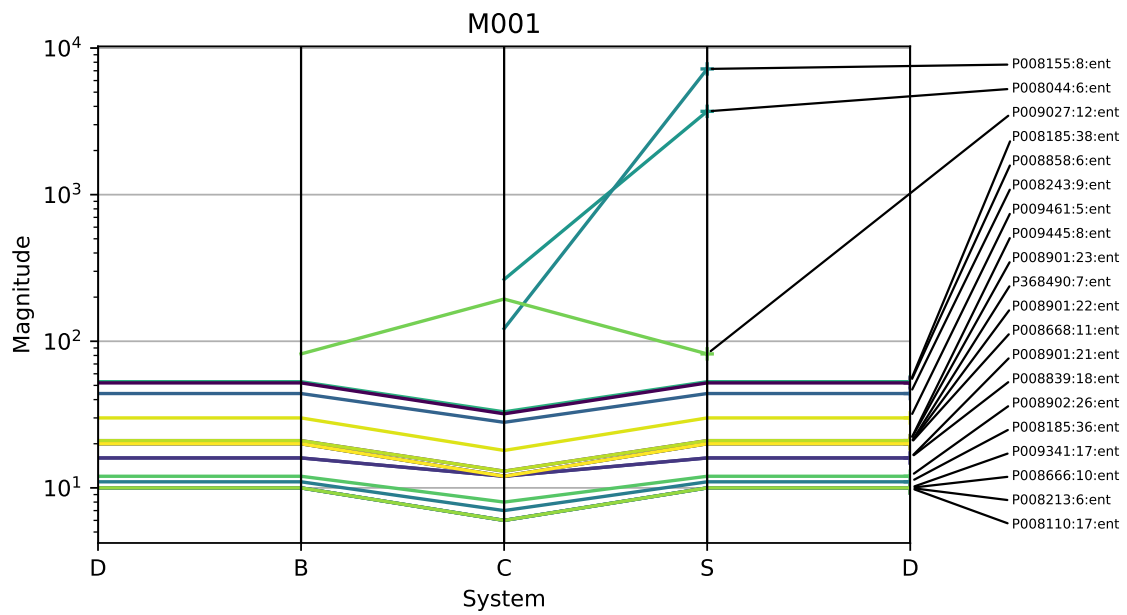


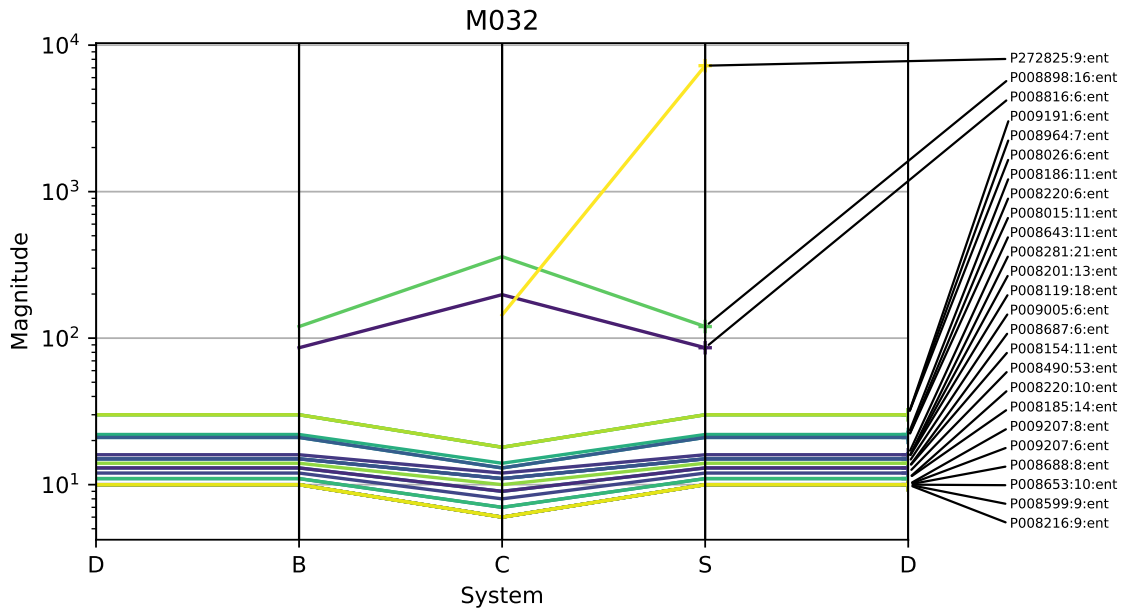
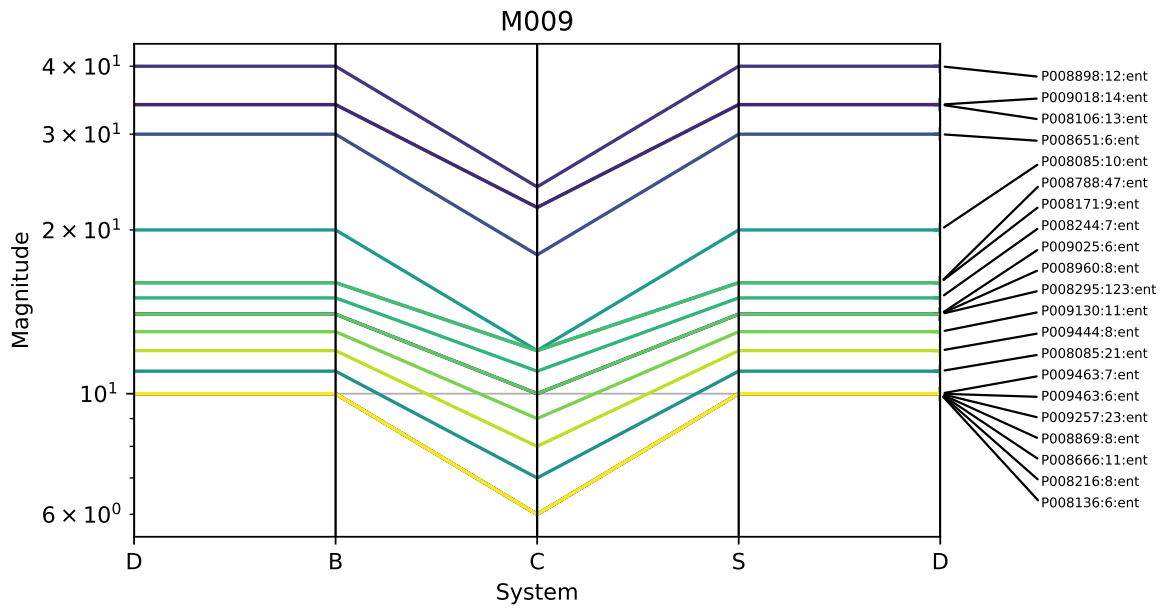
Figure A.10: Completeness and homogeneity for different clustering sizes using Japanese models from Chapter 9

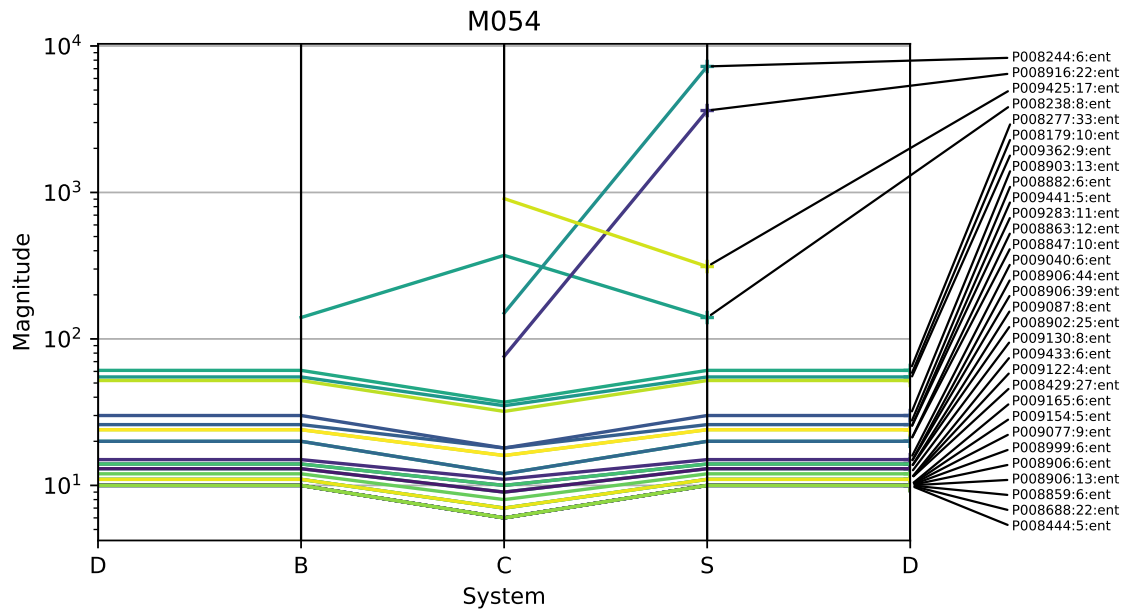
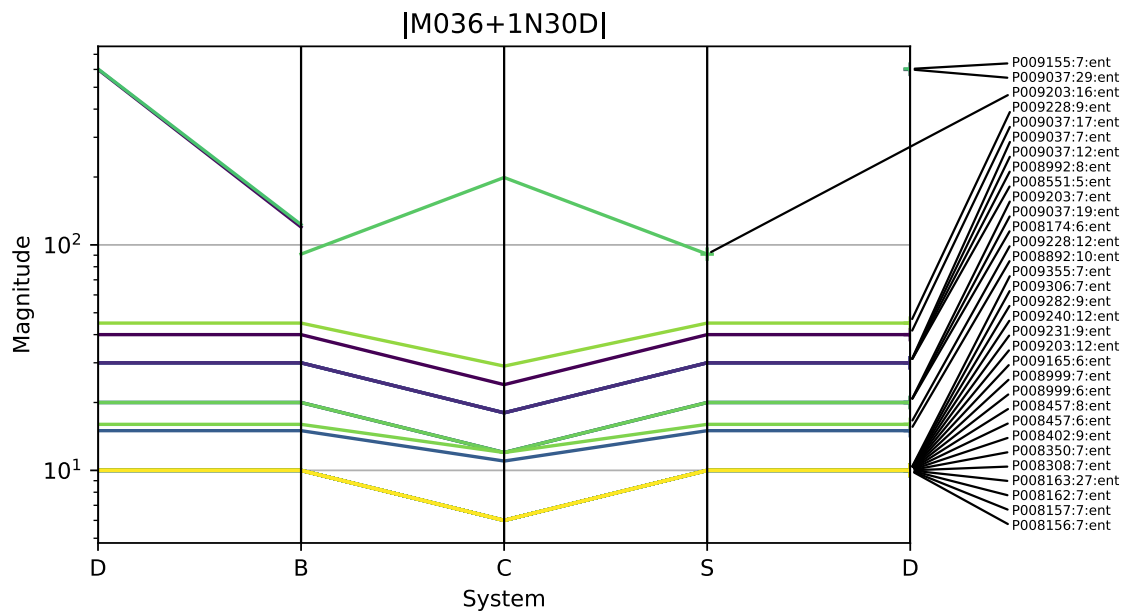
## A.8 Parallel Coordinates

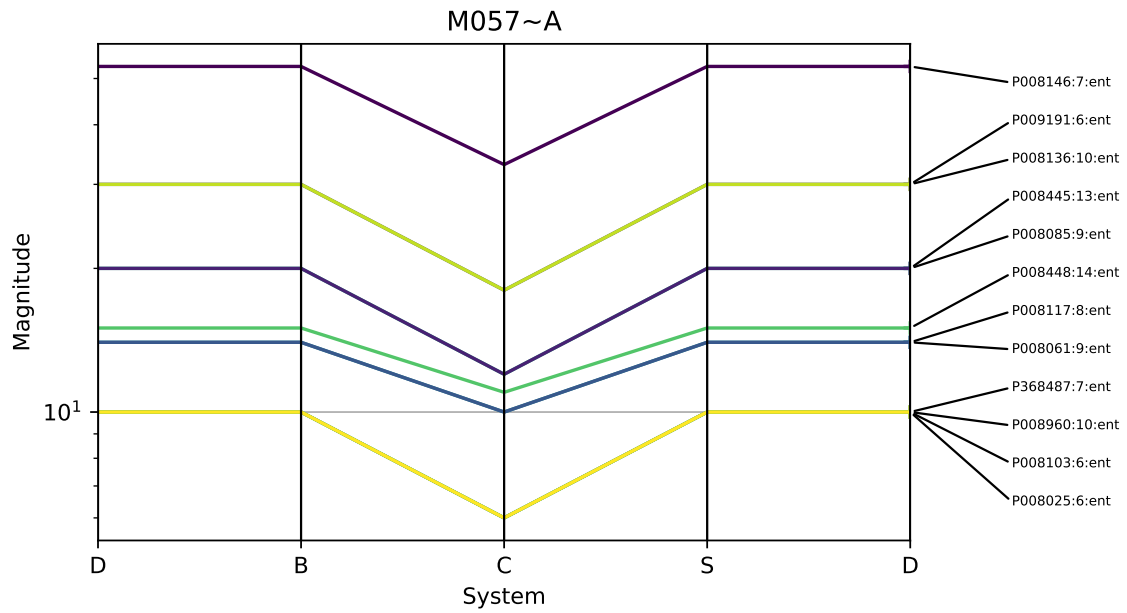
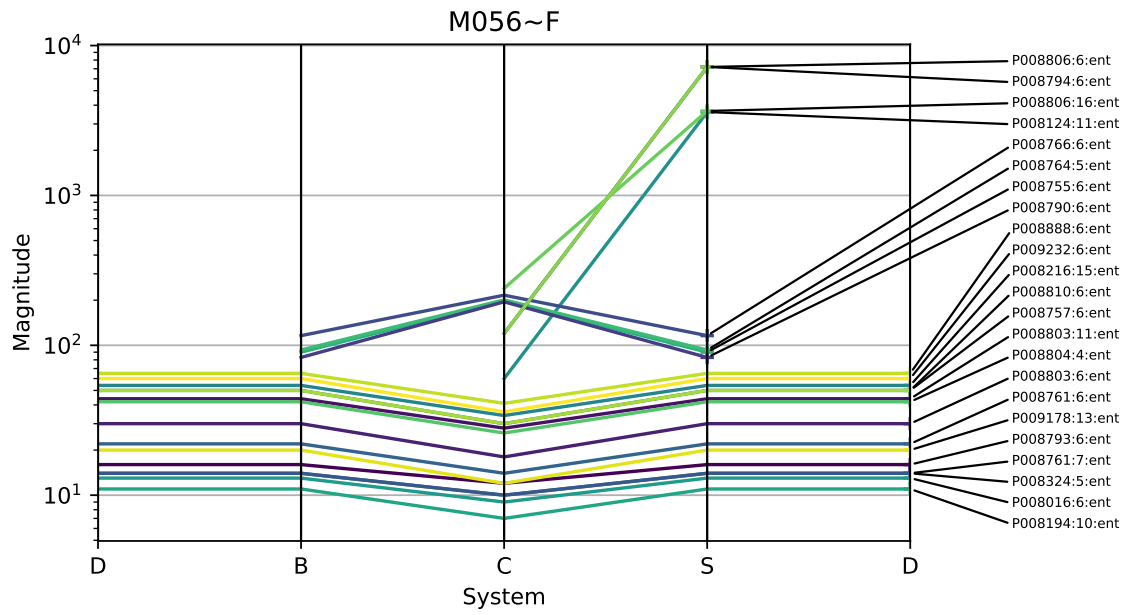
This section provides parallel coordinate plots for all counts with an ambiguous system, grouped according to which sign(s) occur in the preceding entry. For reasons of space, we only show signs that occur next to at least 10 ambiguous numerals, and for especially dense plots we leave some lines unlabeled to maintain readability. We will release the code to reproduce these figures separately, for parties interested in other signs or in other views of these signs.

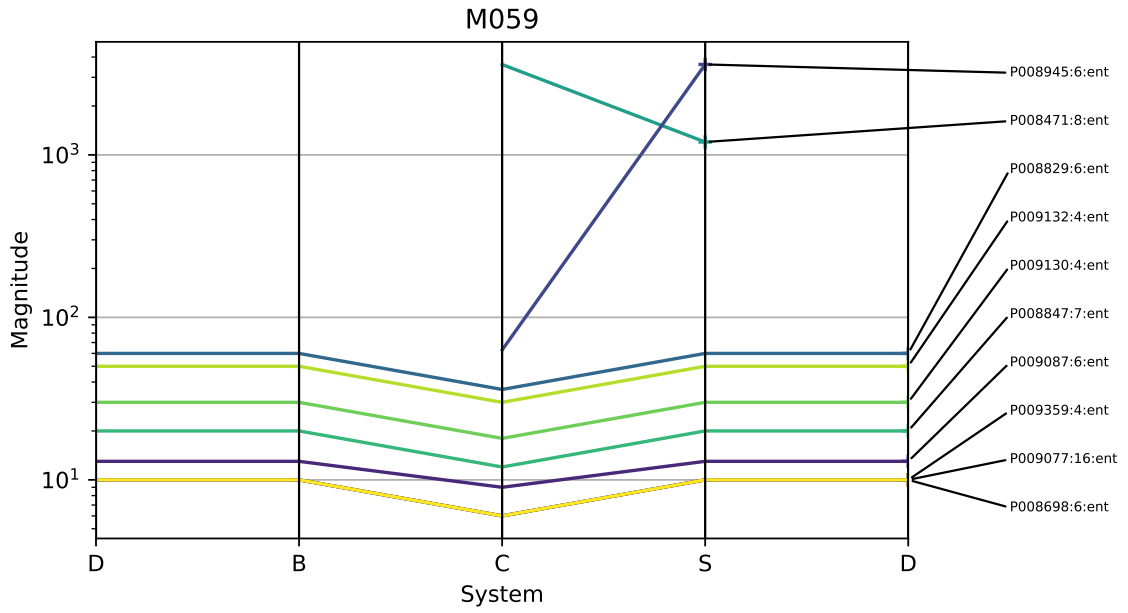
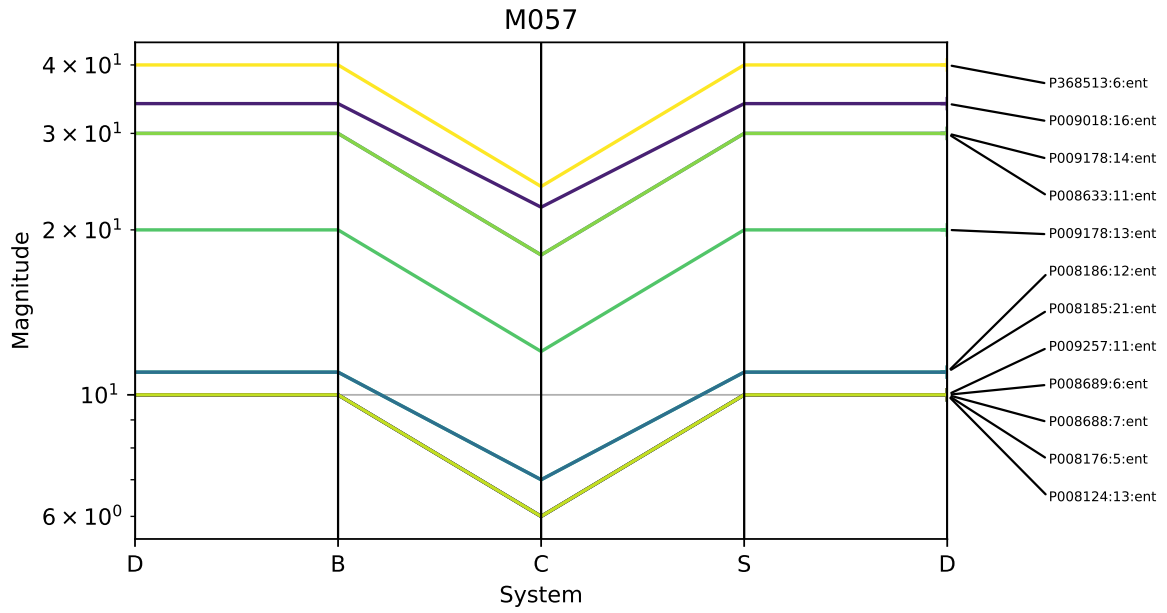


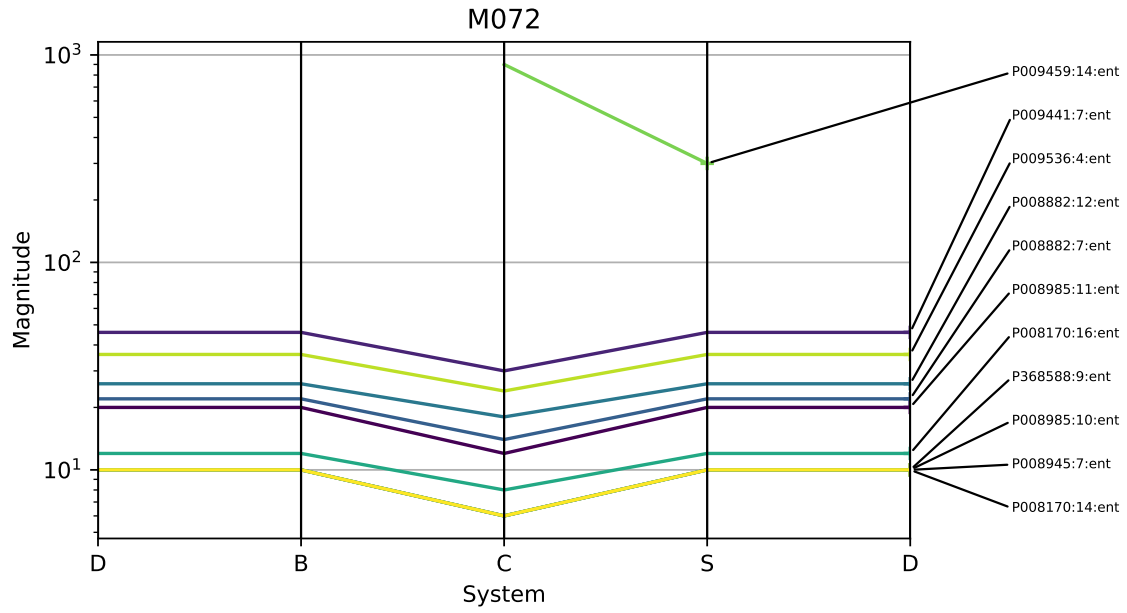
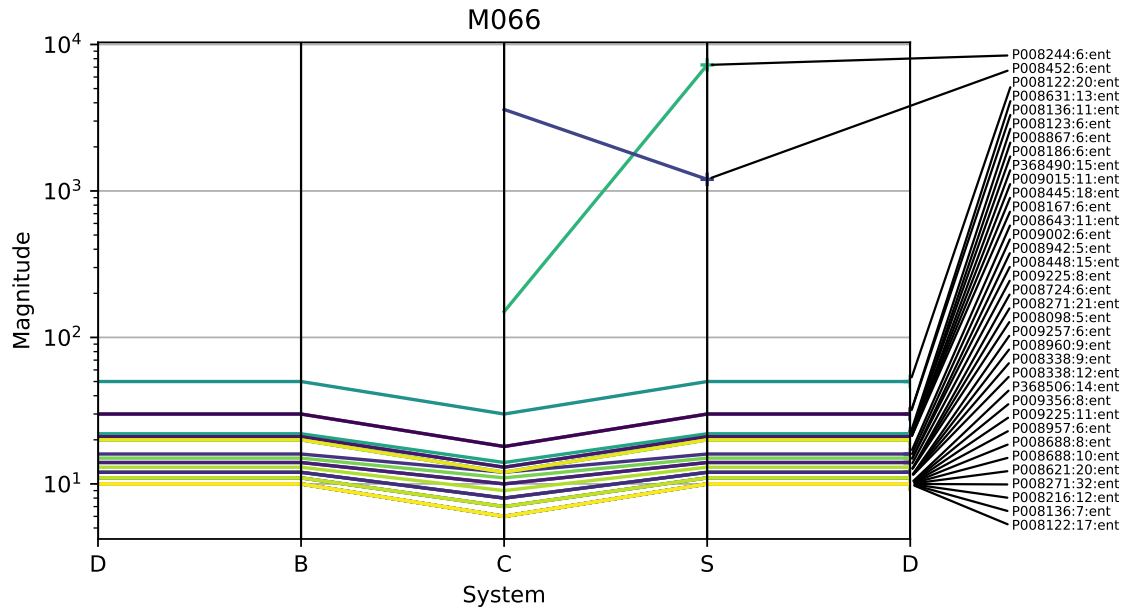


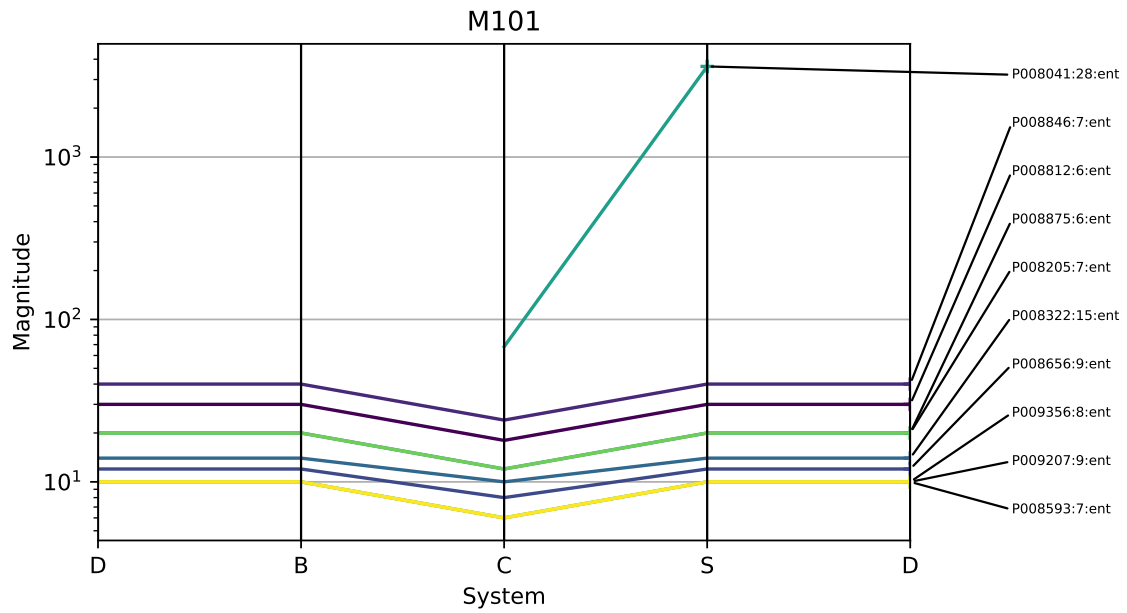
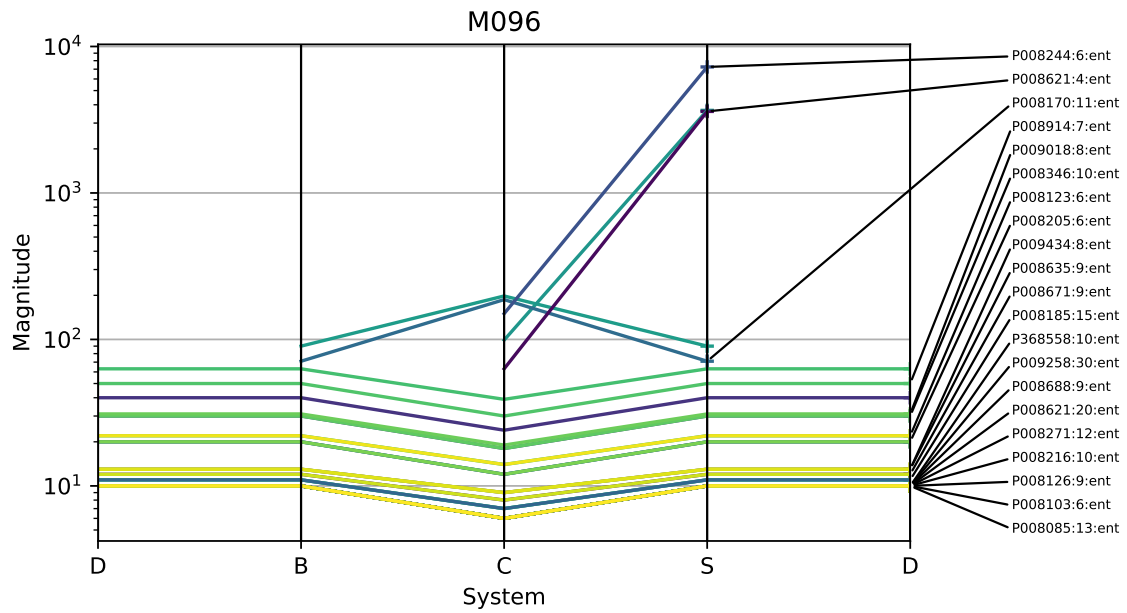


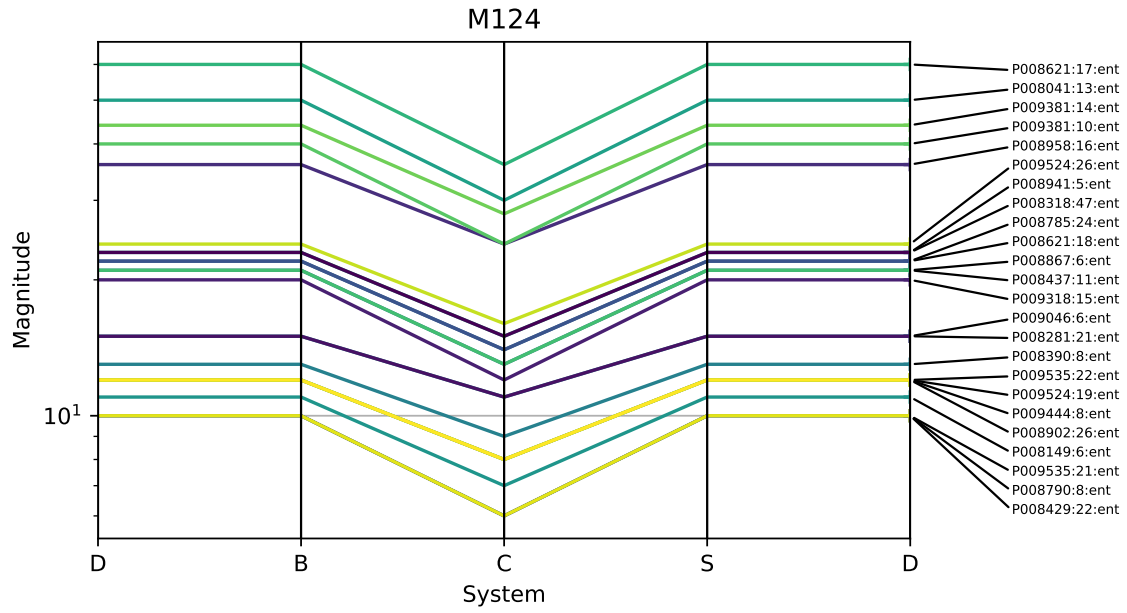
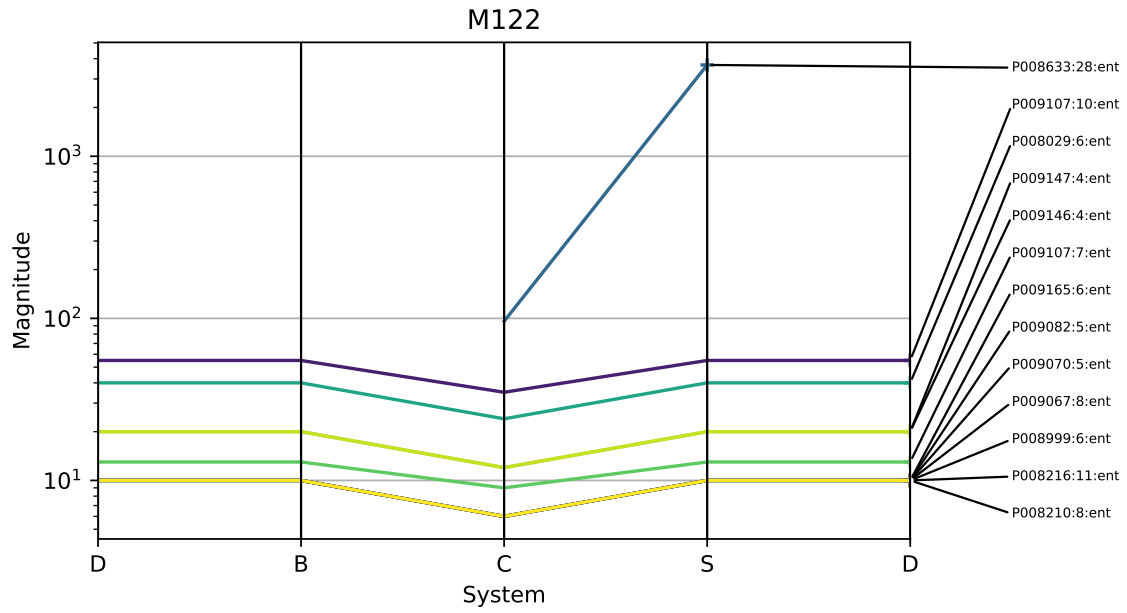




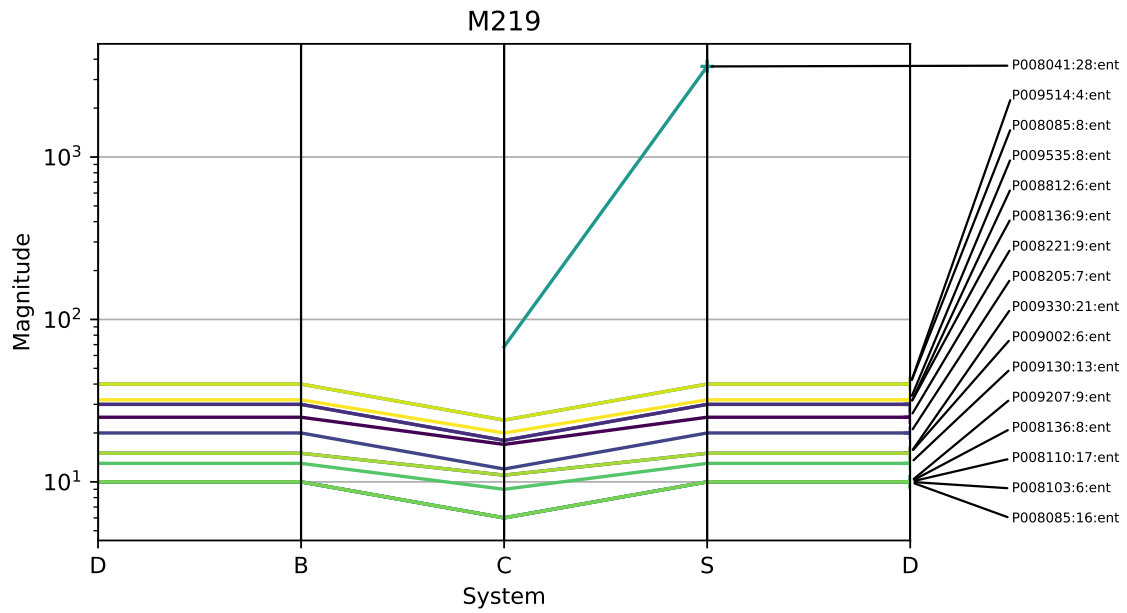
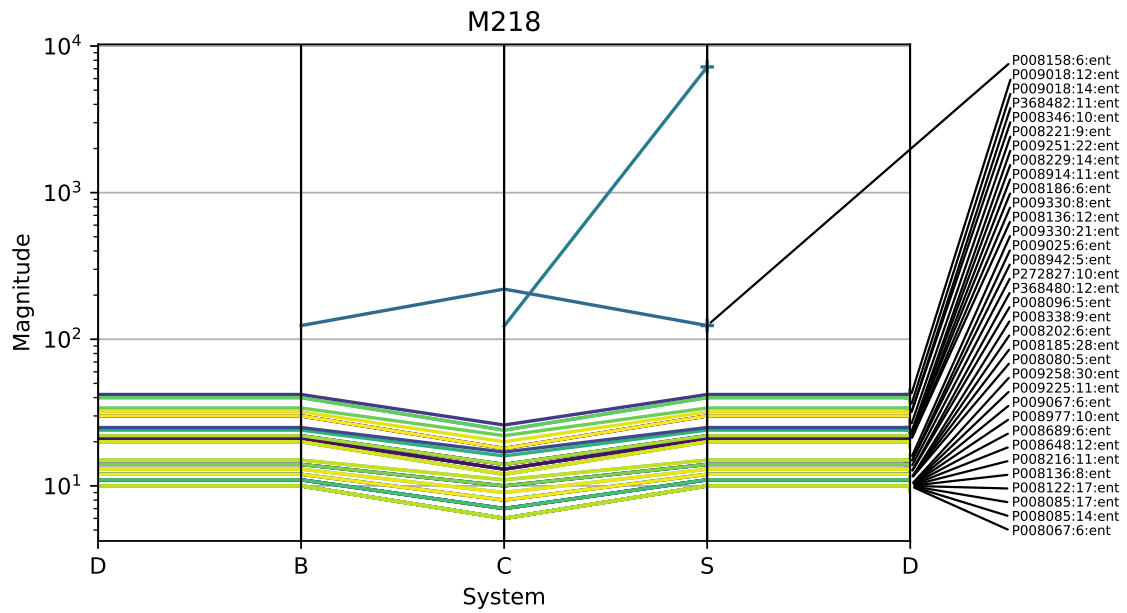


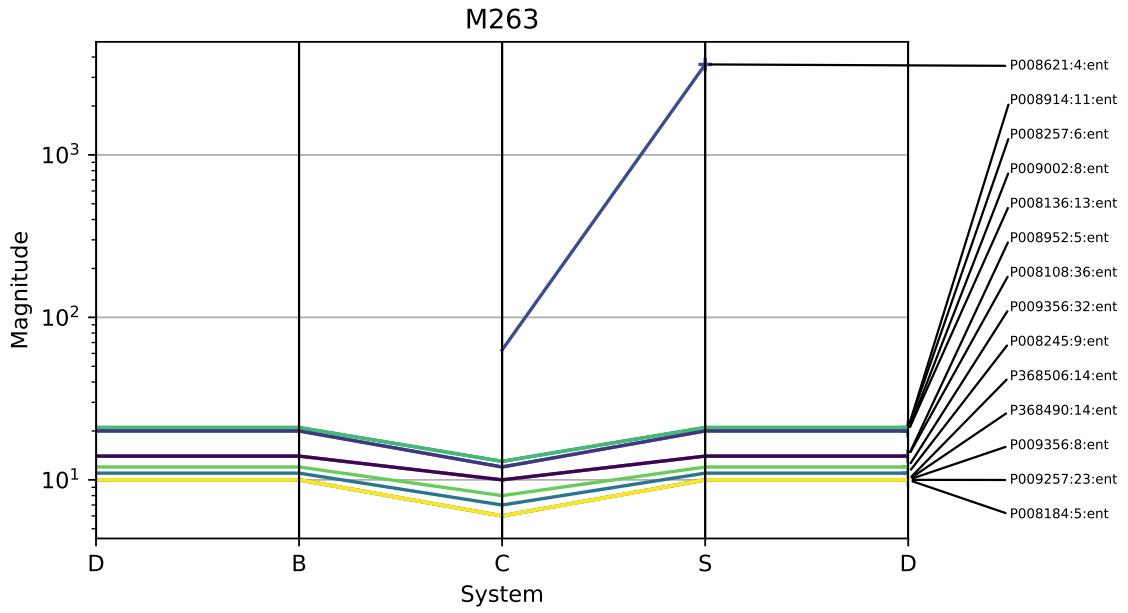
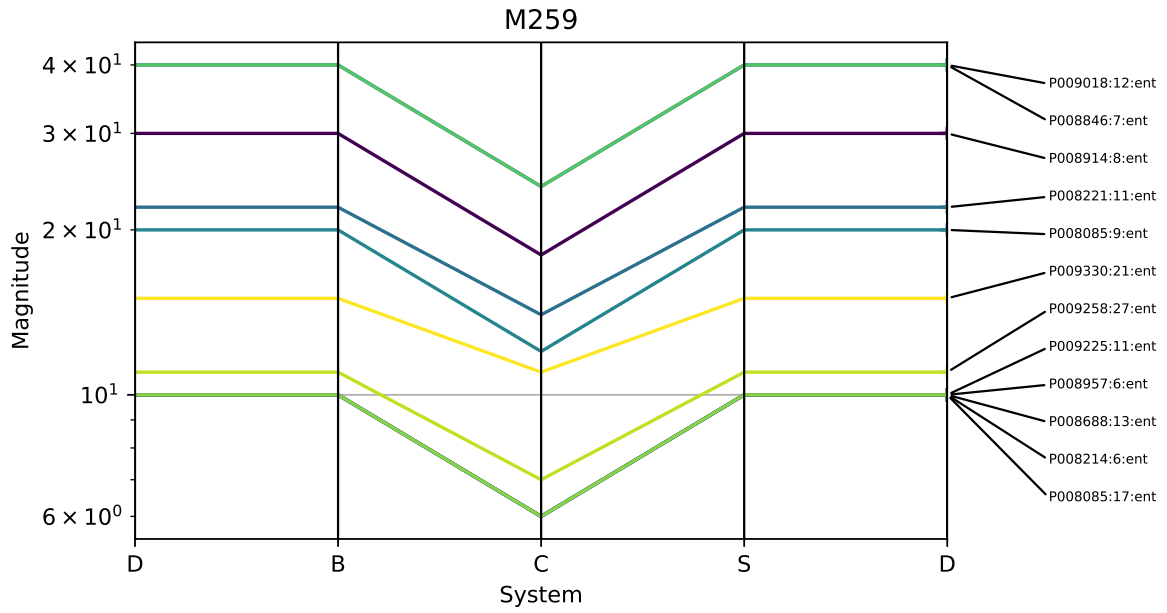


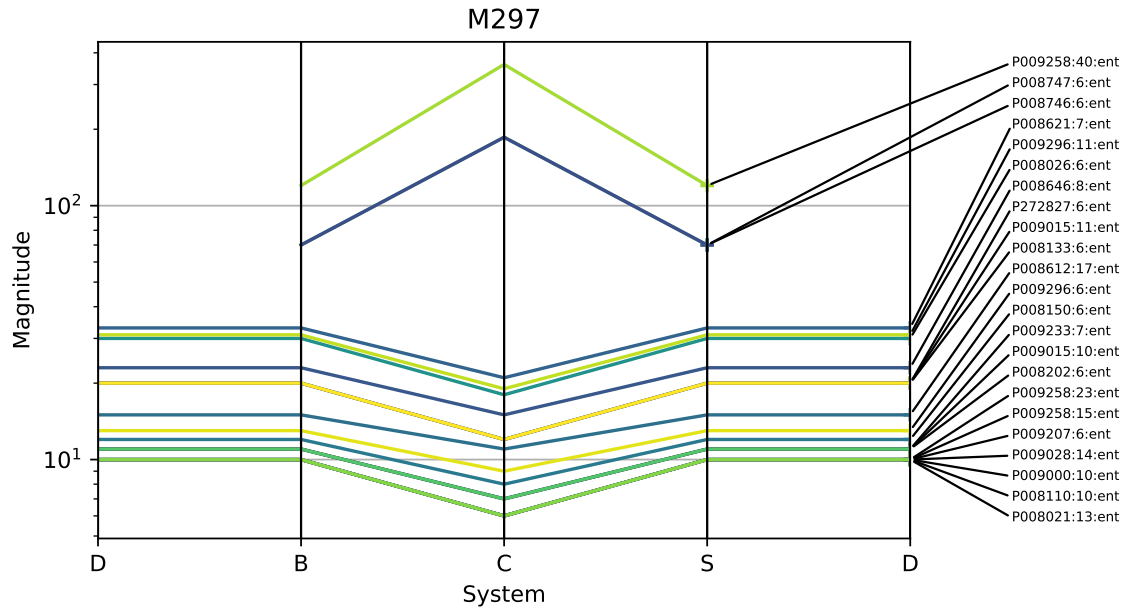
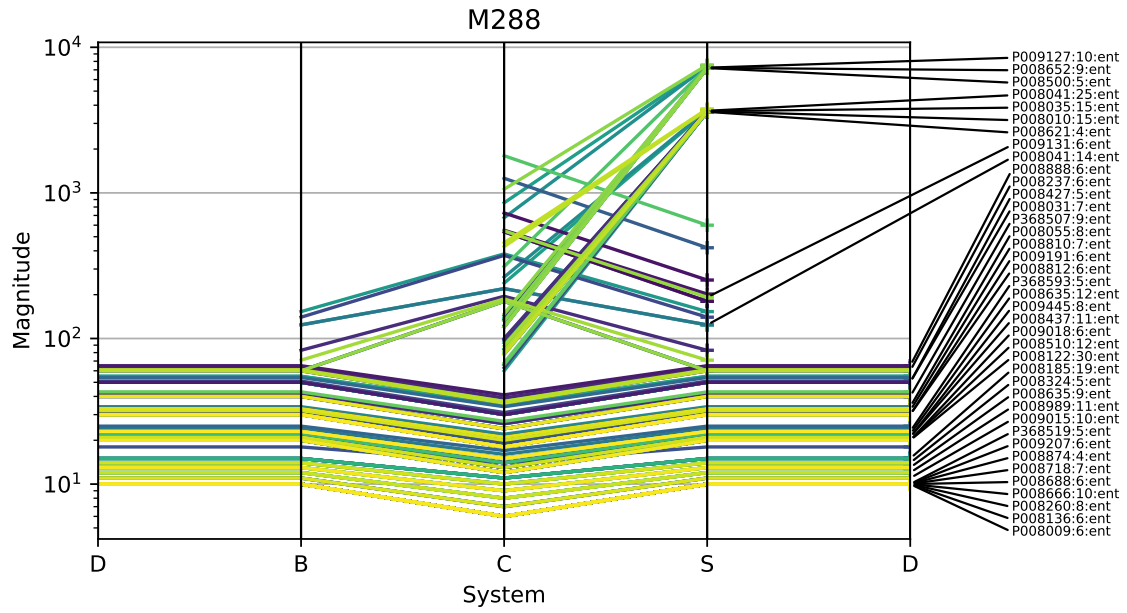


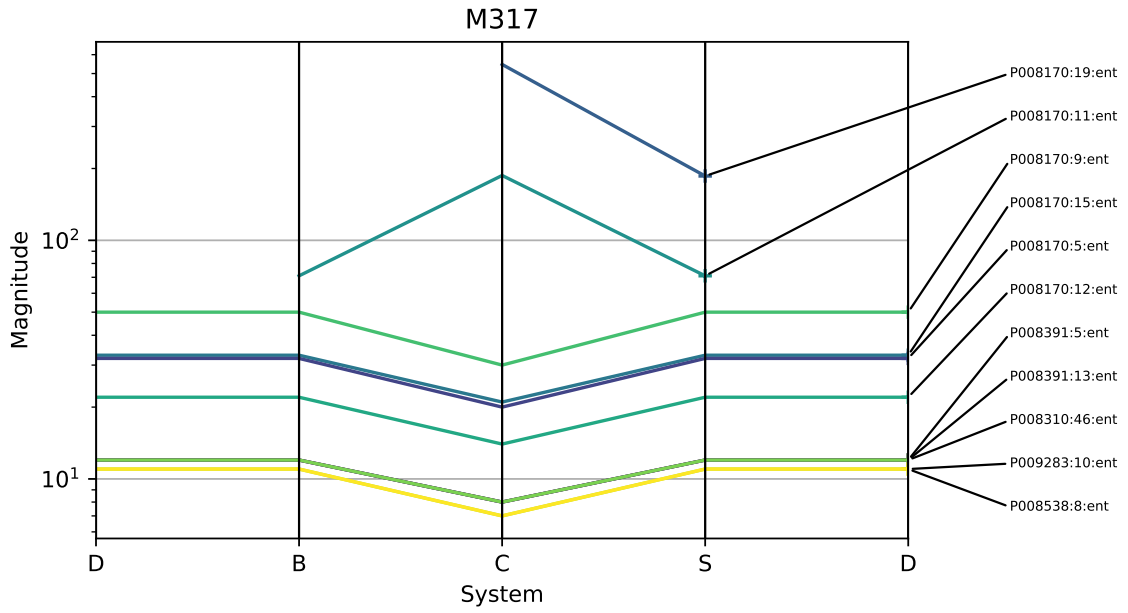
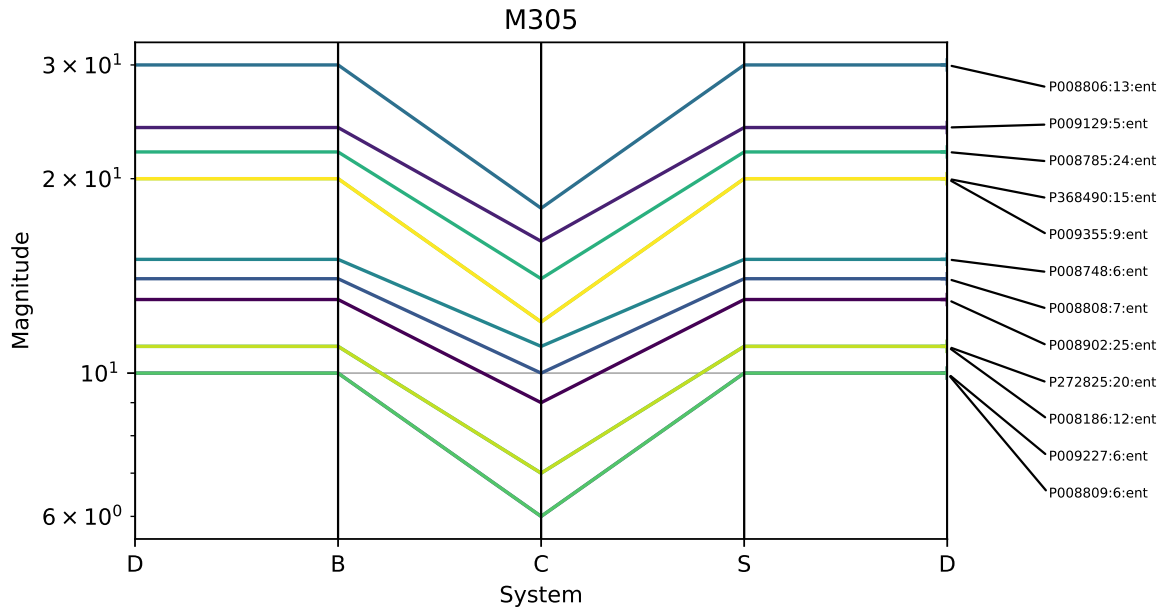




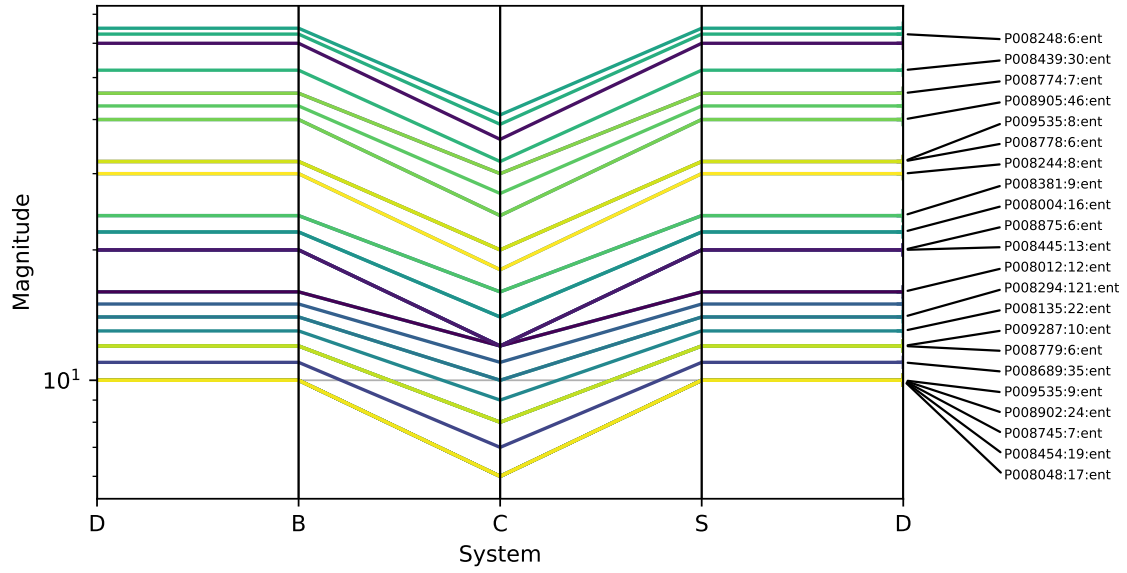




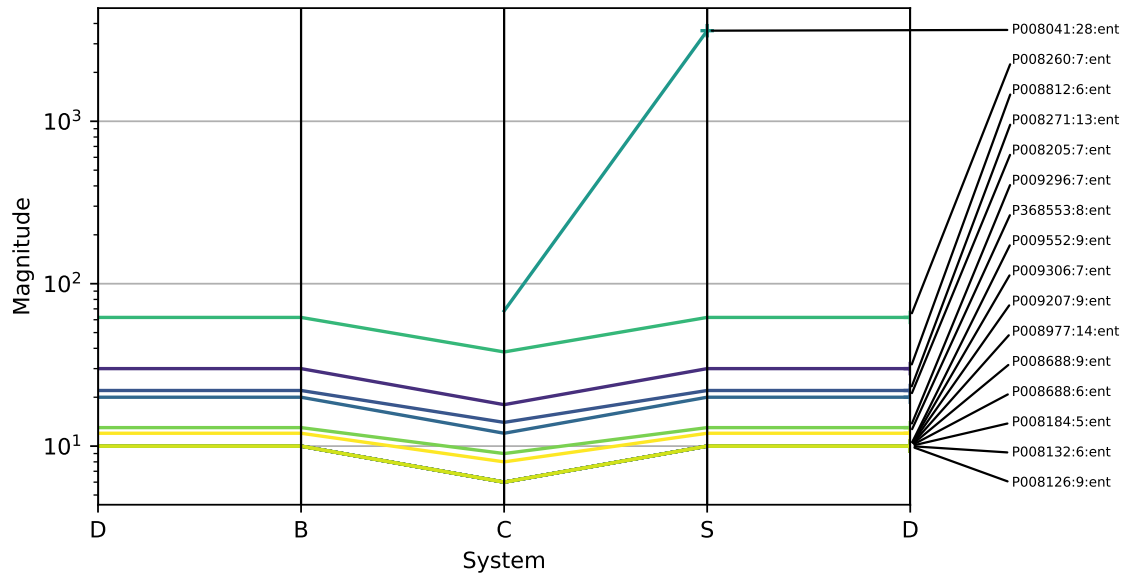


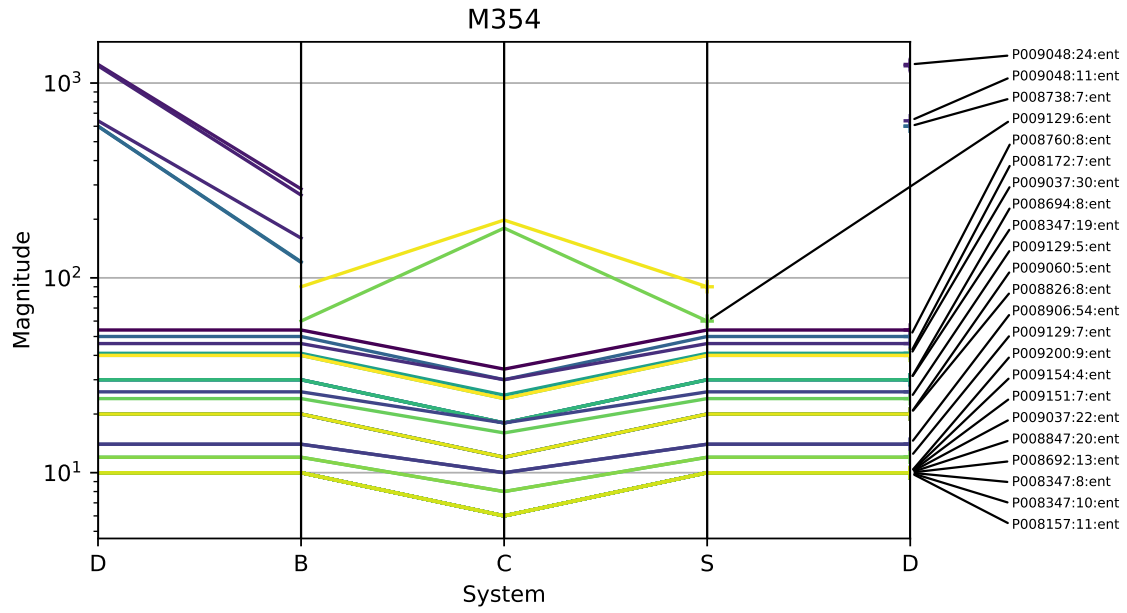
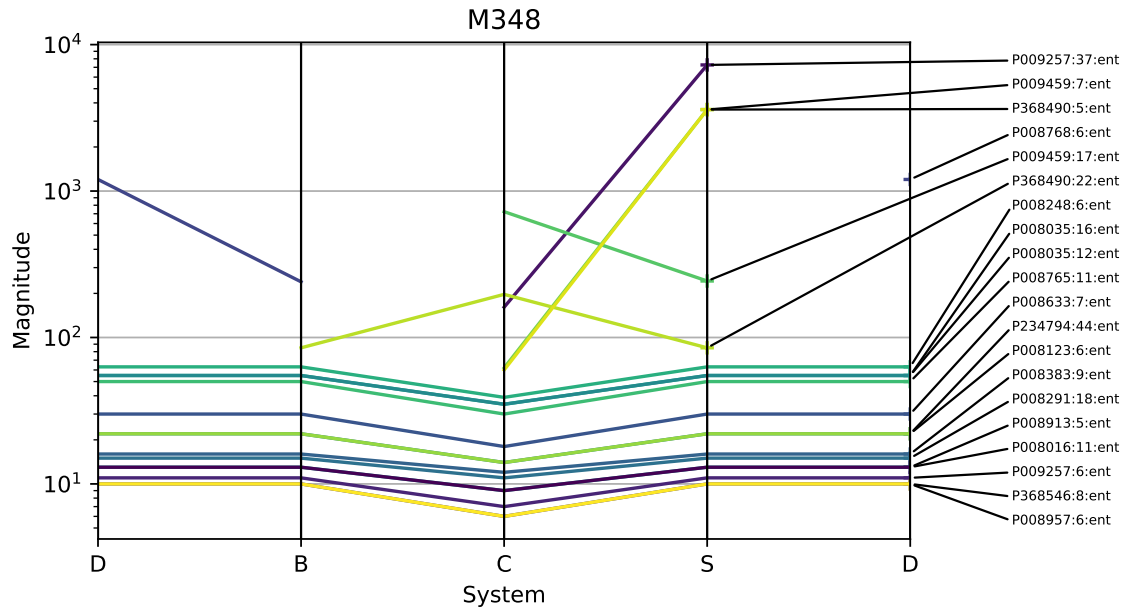


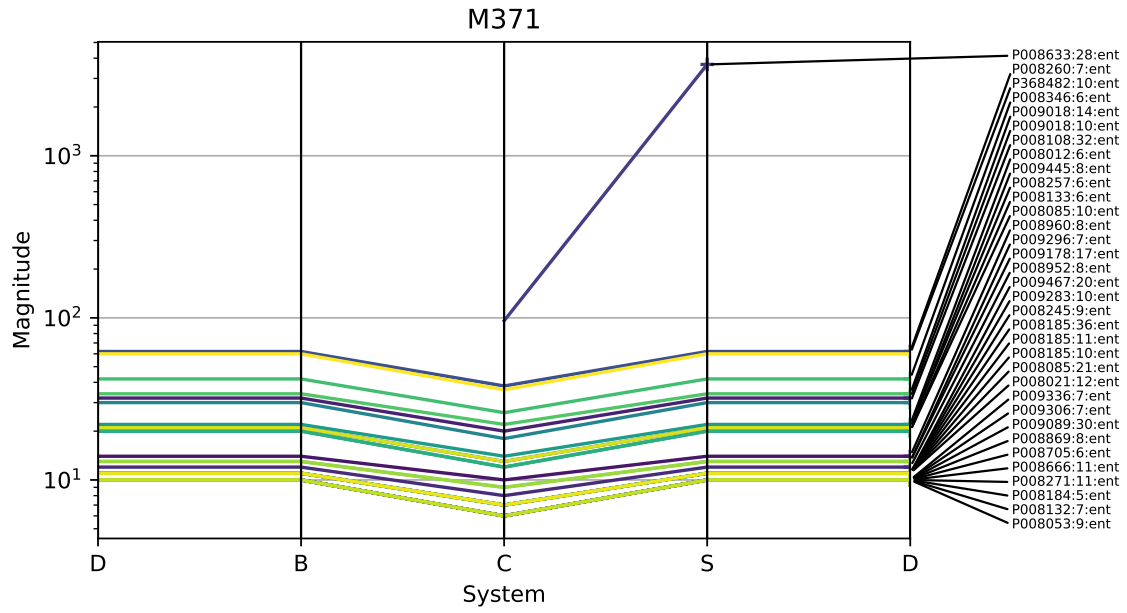
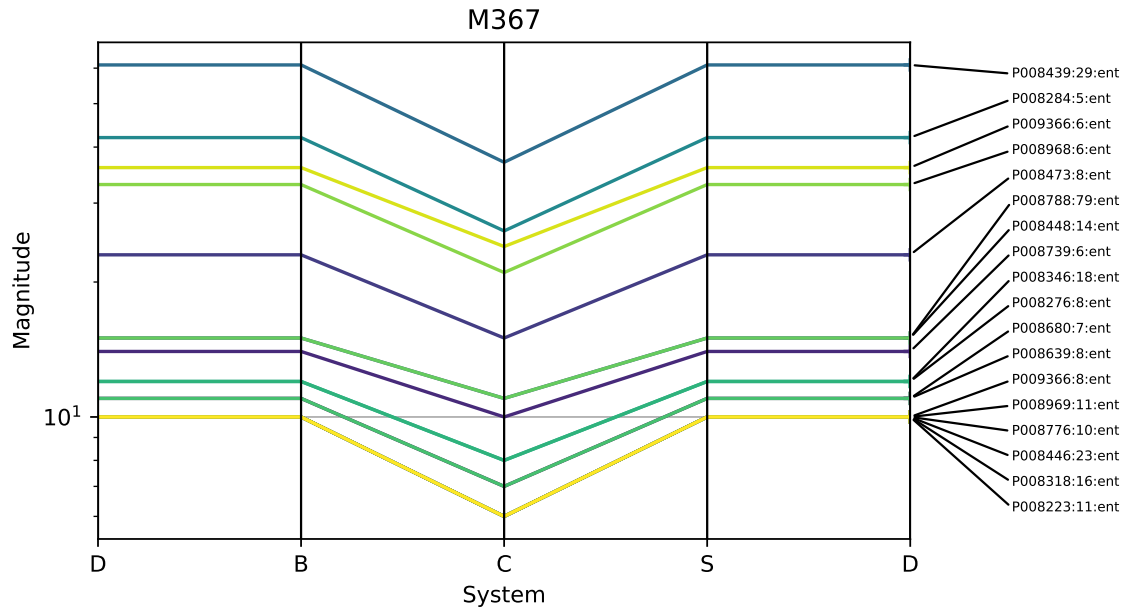
M346

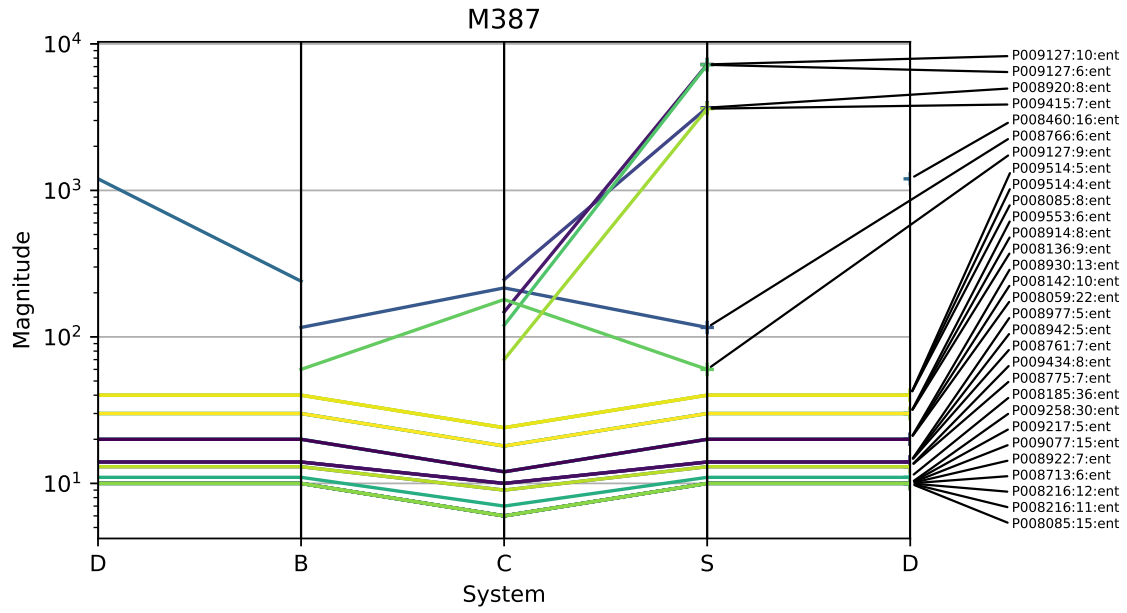
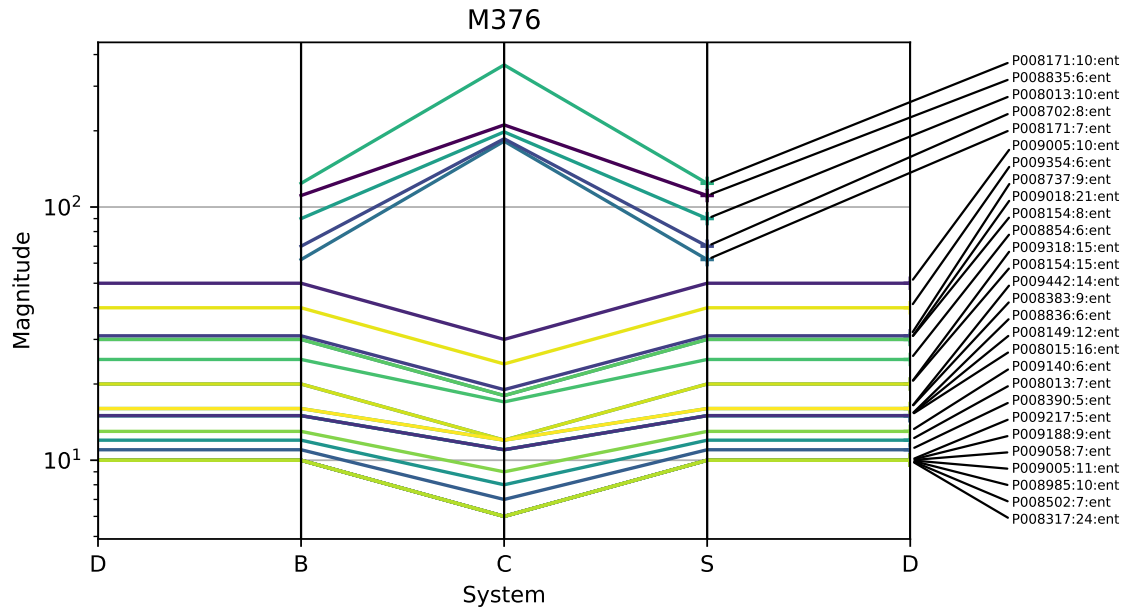


M347

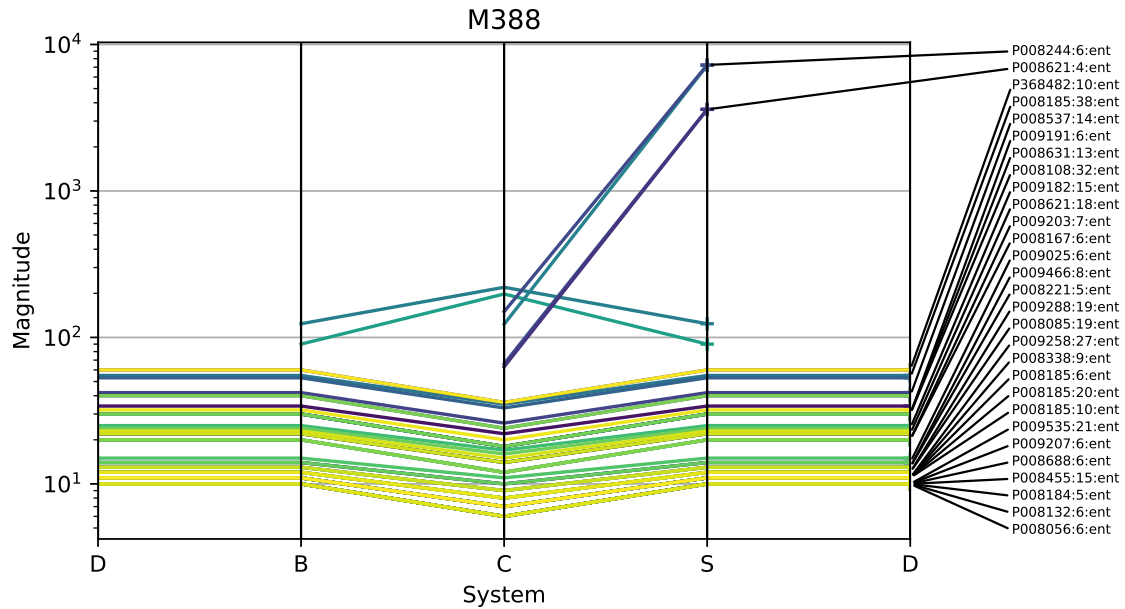












## A.9 Headers

This section summarizes which texts belong to certain categories identified in the body of the paper. Texts are identified by the P-number assigned by the CDLI.

### I. Texts with an implicit header, for which we have corrected the transliteration:

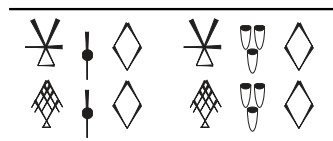
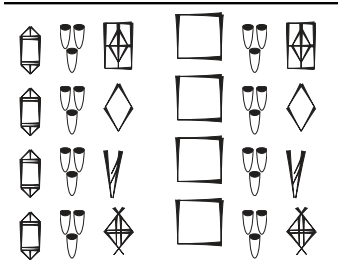
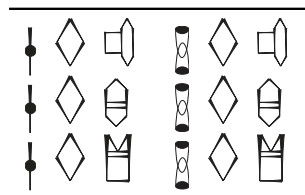
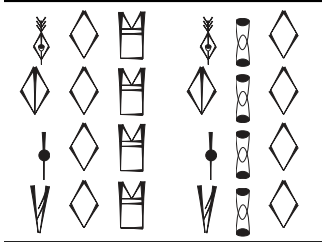
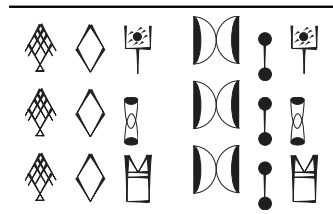
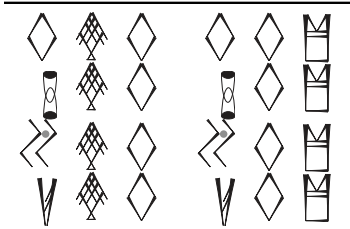
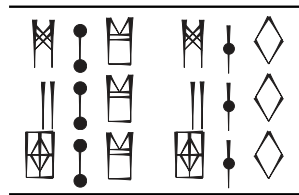
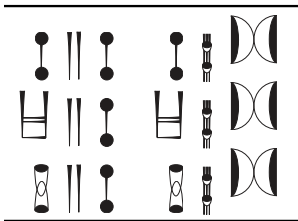
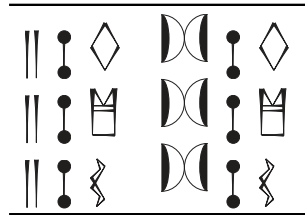
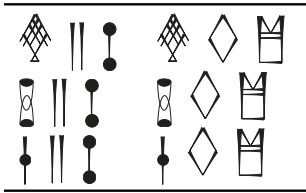
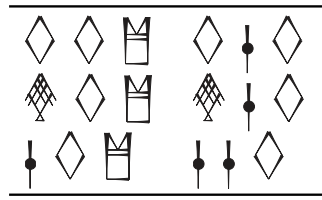
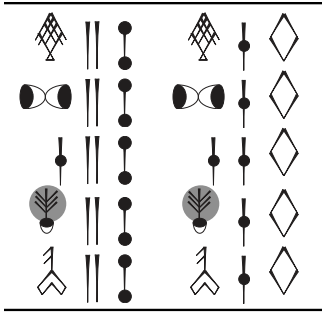
P008020, P008251, P008255, P008311, P008365, P008463, P008641, P008845, P008850, P008853, P008878, P008880, P009051, P009053, P009055, P009060, P009094, P009126, P009320, P009422, P009441, P009461, P009469, P393079, P393080

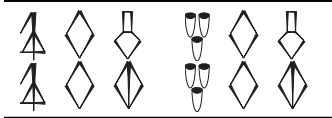
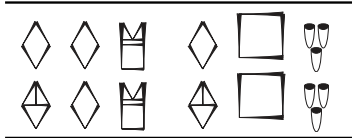
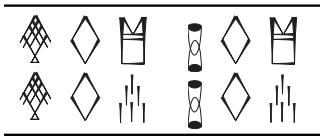
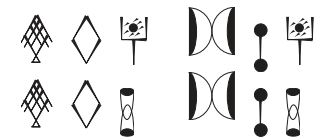
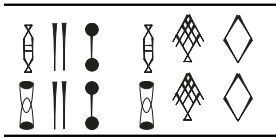
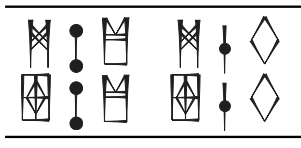
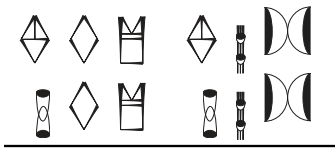
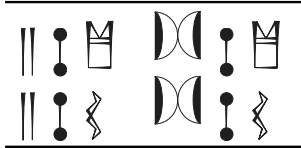
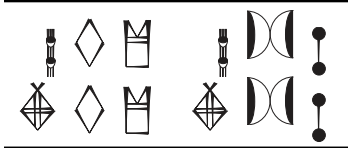
### II. Human-labeled two-sign headers:

P009524, P008220, P008258, P008281, P008702

## A.10 Kober Triplets

This section prints all of the alternations identified by Kober's method (Chapter 11.2) involving strings that occur at least 20 times across the corpus.





## Appendix B

# Reproducibility Details

### B.1 Signlist Revision

The code for our signlist revision models will be published at <https://github.com/MrLogarithm/cawl-clustering>. All of these models are implemented with PyTorch. All settings use an encoder with the following structure:

```
Sequential(  
    Dropout(0.5)  
    Conv2d(1 input channel, 3 output  
          channels, kernel size 8)  
    ReLU()  
    Conv2d(3 input channels, 6 output  
          channels, kernel size 8)  
    ReLU()  
    MaxPool2d(kernel size 3, stride  
              length 3)  
    Conv2d(6 input channels, 8 output  
          channels, kernel size 8)  
    ReLU()  
    MaxPool2d(kernel size 3, stride  
              length 3)  
    Flatten()  
    Dense(72 input dims, 16 output  
         dims)  
)
```

A pair of `Dense(16, 16)` layers project the encoded output to  $\mu$  and  $\sigma$ .

All settings use a decoder with the following structure:

```
Sequential(  
    ConvTranspose2d(16 input channels,
```

```

        60 output channels, kernel size
        8)
    BatchNorm2d(60 channels)
    ReLU()
    ConvTranspose2d(60 input channels,
        30 output channels, kernel size
        8, stride length 2)
    BatchNorm2d(30 channels)
    ReLU()
    ConvTranspose2d(30 input channels,
        15 output channels, kernel size
        8, stride length 2)
    BatchNorm2d(15 channels)
    ReLU()
    ConvTranspose2d(15 input channels,
        1 output channel, kernel size
        15, stride length 1)
    Sigmoid()
)

```

The VAE+LSTM model uses a single-layer unidirectional LSTM with a hidden dimension of size 16. The VAE+Transformer uses a 6-layer `TransformerEncoder` with 8 heads per layer, input and output dimensions of size 16, and 0.5 dropout. We apply a standard sinusoidal positional encoding to the Transformer inputs following Vaswani et al. (2017).

In the VAE+LSTM and VAE+Transformer models, we re-apply the reparameterization trick from Kingma and Welling 2014 to the LM outputs before decoding the image sequence. We add new `Dense(16,16)` layers to compute  $\mu$  and  $\sigma$  at this stage, separate from those used to compute  $\mu$  and  $\sigma$  within the VAE itself. When computing the overall KL divergence loss for these models, we sum the KL divergence from these projections with that of the VAE projections.

We train on sequences of length 50 using the Adam optimizer (Kingma and Ba 2015) with learning rate 0.001. When computing the loss, we scale the KL divergence loss term by 0.45. The LR and loss scaling hyperparameters were tuned via a small manual grid search. We recompute pseudolabel assignments at the start of every 600th training iteration.

## B.2 Headers

We implement our HMM using the `hmmlearn` package for Python. We train our Transformer LM following the instructions at [https://github.com/facebookresearch/fairseq/blob/main/examples/language\\_model/README.md](https://github.com/facebookresearch/fairseq/blob/main/examples/language_model/README.md). Our corpus is small, and this model trains on a single GTX 1070 for approximately one hour.

Our revisions to the corpus from Born et al. 2019 are available at <https://github.com/sfu-natlang/pe-headers>. We also include a `csv` listing the expert labels and the predictions from our models.

We preprocess the data by removing all comments and annotations (lines beginning in \$, &, or #) and deleting the , character which marks entry boundaries (entries are logical units delimited by explicit numeral notations). We remove annotations marking damage and corrected signs (characters matching the regular expression `[\\[\]<>#?!]`). We delete newlines from each text and compile the corpus into a file with one complete tablet per line. We shuffle the lines of this file and set aside 200 tablets as a validation set. The Transformer LM is trained directly on the data at this stage, tokenized on spaces only (we do not use a subword tokenizer). Before training the HMM, we circumfix beginning- and end-of-sequence tokens `<bos>` and `<eos>` to each line.

The embeddings which we use to evaluate compositionality are available upon request to the authors of Born et al. 2021.

## Appendix C

# Constructing a Validation Set for Numeral Disambiguation

This chapter describes a set of ambiguous numerals which we believe can be confidently disambiguated. Together, these numerals comprise the validation set used in Chapter 8 to evaluate approaches to automated disambiguation via bootstrapping.

### C.1 Capacity Measures

**P008014** Our subset-sum analysis (Section 8.3.1) reveals that the sole entry on the reverse of this tablet (P008014:18) exactly equals the sum of the entries on the obverse, provided the whole tablet is read in the capacity system. This is consistent with the fact that four of the entries on the text are unambiguously capacity measures (P008014:11, P008014:14, P008014:15, and P008014:18; the other entries are SDBC ambiguous). Moreover, the apparent summary line has an M288 object, and the first entry on the obverse ends in M288. This suggests that the entire text may record amounts of M288 (which is strongly associated with the capacity system), but that this object has been left implicit in all but the first and last entries.

On the basis of this evidence, we disambiguate the seven ambiguous entries in this text (P008014:6, P008014:7, P008014:8, P008014:9, P008014:10, P008014:12, P008014:15) to the capacity system.

### C.2 Sexagesimal Measures

**P008173** Our subset-sum analysis reveals that the first entry on the reverse (P008173:13), which is an unambiguous sexagesimal notation counting  $7.5 \times N01_S$  instances of M376, exactly equals the sum of the M376 entries on the obverse (P008173:5, P008173:7, P008173:8) if these entries are read in the sexagesimal system. On the basis of this evidence we disambiguate these entries to the sexagesimal system.

**M056~f/M288 Texts** P008798 is exemplary of a set of two-entry texts of comparable physical dimensions (approx. 43mm x 31mm x 18mm) which count M056~f in the first entry and M288 in the second. In many of these texts, the first entry is unambiguously sexagesimal and the second is unambiguously capacity; in these cases the amount of M056~f is always exactly 2.5 times the amount of M288.

If P008797:6 is read as a sexagesimal notation, then this text follows the same pattern as these other texts, and exhibits the expected 2.5:1 ratio of M056~f to M288. On the basis of this evidence we disambiguate this entry to the sexagesimal system.

By the same argument, we also disambiguate P008791:6, P008799:6, P008800:6, P008801:6, P008802:6, P008804:4 and P008810:7 to the sexagesimal system.

### C.3 Decimal Measures

**P008179** This tablet is broken: a fracture on the obverse renders one digit partially unreadable. The broken digit has been restored as 2(N23), but we propose that the correct reading should in fact be 1(N23).

To see that this should be the case, notice that the first entry on the reverse (P008179:14) unambiguously equals  $852 \times N01_D$ ; this is exactly the same sum obtained by reading the obverse entries in the decimal system, provided one uses our restoration instead of the current transliteration.

This reading is consistent with the fact that some entries (P008179:8, P008179:9, P008179:14) are already unambiguous decimal counts. Moreover, all but one of the entries on the obverse count the same object (M388), which further points towards their likely using the same number system. On the basis of this evidence, we disambiguate P008179:10 and P008179:11 to the decimal system.

The original restoration (2(N23)) presumably arose from the observation that the lacuna is wide enough to span two signs. The last visible sign before the lacuna is M388, which only occurs at the very end of entries elsewhere in this text, implying that all of the missing signs are likely digits. Our proposed reading requires that the N23 in the lacuna occupy a slightly wider space than would be typical, or perhaps that some of the lacuna be occupied by an erasure. If the original restoration is correct, there must instead be an arithmetic error in the summary line (which in this case differs from the true sum by 1(N23)).

**P008012** Following the claim that sheep and goats are counted decimally in proto-Elamite (Englund 2011), the entirety of this text should be expected to use the decimal system. Our subset-sum analysis confirms that the entry on the reverse (P008012:16) equals the sum of the entries on the obverse when read in this system, though in this case that would also be true for sexagesimal and bisexagesimal readings. Notably, the summation does *not* work out if the summary is read as a capacity measure. Thus, while we may confidently say that this text does not contain capacity measures, we can only tentatively assign it to the decimal system in particular.



**P008243** Kelley (2018) observes that this is a decimally-counted roster. We follow this author in disambiguating all ambiguous numerals in this text to the decimal system.

## C.4 Bisexagesimal Measures

**P009048** This text contains a large number of unambiguous bisexagesimal notations. Of these, P009048:16 counts the same object (M352~h) as P009048:19, on the basis of which we propose that P009048:19 is also a bisexagesimal notation. Similarly, the unambiguous entries P009048:10 and P009048:15 count the same objects (M351+X and M354, respectively) as P009048:13 and P009048:7, respectively, on the basis of which we also disambiguate these entries to B.