Principal Parts Detection for Computational Morphology: Task, Models and Benchmark

Dorin Keshales, Omer Goldman, Reut Tsarfaty
Bar-Ilan University
{dorinkeshales14, omer.goldman}@gmail.com, reut.tsarfaty@biu.ac.il

Abstract

Principal parts—defined as the minimal set of cells from which all other forms within a lexeme's inflectional paradigm can be deduced—are an important concept in theoretical morphology. This concept, which outlines the minimal memorization needed for a perfect inflector, has been largely overlooked in computational morphology despite impressive advances in the field over the past decade. In this work, we formalize PRINCIPAL PARTS DETECTION as a computational task under the static scheme assumption, identifying a single set of cells as principal parts uniformly applicable across lexemes within a syntactic category. We construct a multilingual dataset of verbal inflection tables with gold principal parts annotations for ten typologically diverse languages. We evaluate several computational models for PRINCIPAL PARTS DETECTION, each implementing the same three-stage framework: characterizing relations between each pair of cells, clustering the resulting vector representations, and selecting a representative cell from each cluster as a predicted principal part. Our best-performing model, combining Edit Scripts between inflections with Hierarchical K-Means clustering, achieves an average F1 score of 55.05%, significantly outperforming a random baseline of 21.20%. While these results demonstrate initial success, further research is needed to advance PRINCIPAL PARTS DETECTION, which could optimize inputs for morphological inflection models and encourage exploration into the theoretical and practical significance of compact morphological representations.

1 Introduction

Morphological analysis is essential for understanding natural language, particularly in languages with complex inflectional systems. In both linguistic theory and language pedagogy, the concept of *principal parts* plays a central role in structuring and simplifying inflectional paradigms (Finkel and Stump, 2007; Stump and Finkel, 2013). Principal parts form the minimal subset of paradigm cells from which all other forms can be systematically derived.

By identifying these key forms, principal parts provide a compact representation of inflection tables and facilitate the analysis of morphologically rich languages. Despite their theoretical significance, the detection of principal parts remains largely unexplored in computational morphology. While they have inspired research in inflection and reinflection (Cotterell et al., 2017; Liu and Hulden, 2020), they are rarely used explicitly. Most computational approaches instead rely on a single citation form, the lemma (Cotterell et al., 2016; Goldman et al., 2023), or select input forms randomly (Cotterell et al., 2016; Kann et al., 2017). This reliance on suboptimal input representations overlooks the potential of principal parts as a more efficient foundation for inflectional modeling.

In this paper, we formalize PRINCIPAL PARTS DETECTION as a computational task under the static principal-parts scheme assumption: given a collection of inflection tables belonging to the same syntactic category, the goal is to identify a single, minimal set of cells that uniformly serve as principal parts across all lexemes. Crucially, inflection tables typically contain standard morphological annotations but are not explicitly labeled with principal parts, making this an unsupervised learning problem. To promote research in this area, we deliver a standardized dataset covering the verbal paradigms of ten diverse languages. We sourced principal parts for each language from online dictionaries, where they are often listed to aid language learners, and obtained full inflection tables from UniMorph (Batsuren et al., 2022).

We develop several computational approaches for PRINCIPAL PARTS DETECTION, leveraging the defining property of principal parts: their encapsulation of implicative relations existing among cells in the paradigm. Our models character-

Proceedings of the 29th Conference on Computational Natural Language Learning, pages 251–267

ize inter-cell similarity and cluster cells into *sub-paradigms*, selecting a representative cell from each sub-paradigm as predicted principal parts. We explore different methods for *characterizing* inter-cell relations, including Edit Distance, Edit Script, and Reinflection Accuracy, and we experiment with *clustering* techniques such as Affinity Propagation and a Hierarchical K-Means algorithm. Our best-performing model, combining Edit Script similarity measure + Hierarchical K-Means clustering, achieves an average F1 score of 55.05% across the ten languages in our dataset, significantly outperforming a random baseline of 21.20%.

By formalizing PRINCIPAL PARTS DETECTION as a computational task, we lay the groundwork for future research on more efficient morphological representations. To the best of our knowledge, this is the first work to deliver a standardized benchmark of PRINCIPAL PARTS DETECTION alongside a fully-operational detection framework. Successfully solving this task could enhance applications in morphological inflection and analysis by providing more informative input forms. Our findings suggest that principal parts can be computationally identified with reasonable accuracy, but further improvements are necessary to fully realize their potential.

2 The PRINCIPAL PARTS DETECTION Task and Dataset

The PRINCIPAL PARTS DETECTION Task. The task of PRINCIPAL PARTS DETECTION is defined as identifying the minimal set of cells within a paradigm that, when known, allow the derivation of all other paradigm forms. For instance, in English, the principal parts of the verbal paradigm are the cells corresponding to the infinitive, simple past and past participle (for example, *eat*, *ate*, and *eaten*), as these forms are not predictable from one another, especially for strong verbs. On the other hand, the forms corresponding to the present participle and the 3rd person singular present are deterministically predictable from the infinitive and they therefore provide no additional information for inflection if the infinitive is known.

Formally, the task of PRINCIPAL PARTS DETECTION is defined under the static principal-parts scheme assumption. Specifically, given a language L, a syntactic category POS, and their associated paradigm$P^L_{POS} = \{c_1, c_2, \ldots, c_n\}$, where each cell c_i corresponds exactly to one coherent morpho-syntactic feature set associated with POS, alongside a set of lexeme-specific inflection tables:

$$\mathcal{T}^L_{POS} = \{t^L_{P_{POS},\ell_1}, t^L_{P_{POS},\ell_2}, \ldots, t^L_{P_{POS},\ell_k}\},$$

each table instantiating the paradigm P^L_{POS} for a specific lexeme ℓ_i.

Then, the task is to identify the minimal subset of cells $C_{PP} \subseteq P^L_{POS}$ from which all remaining forms within each inflection table $t^L_{P_{POS},\ell_i} \in \mathcal{T}^L_{POS}$ can be deterministically deduced.

The PRINCIPAL PARTS DETECTION Dataset. To empirically evaluate methods for the detection of principal parts, we first need a dataset to evaluate against. To this end, we constructed the multilingual PRINCIPAL PARTS DETECTION dataset, containing verbal inflectional paradigms from ten typologically diverse languages: Hebrew, English, French, German, Spanish, Danish, Swedish, Finnish, Turkish, and Latin. These languages were selected based on the availability of comprehensive inflectional data and suitable resources for identifying principal parts.

The input side of the task comprises complete inflection tables sourced from the UniMorph corpus (Batsuren et al., 2022), a large-scale morphological resource providing comprehensive inflectional data across languages, organized by lexeme and morpho-syntactic features.

Gold principal parts annotations — the target output for evaluation — were primarily obtained from two online resources. For five languages (English, German, French, Latin, and Spanish), we directly adopted principal parts from Wikipedia's dedicated principal-parts page.[1] For the other languages, where principal parts were not explicitly documented, we identified them directly based on the forms presented in Wiktionary's standardized verb conjugation templates, except for Finnish, for which we consulted a specialized language-learning resource.[2]

The dataset preparation involved rigorous normalization and error correction applied specifically to the inflection tables. We retained only strictly inflectional forms, excluding derivational forms, and ensured exactly one form per feature set. Sparse, marginal, or inconsistent feature sets were removed, and problematic entries originating from the original sources were manually reviewed and corrected

[1] https://en.wikipedia.org/wiki/Principal_parts
[2] https://ielanguages.com

to ensure a reliable dataset (see Appendix A for details).

The PRINCIPAL PARTS DETECTION dataset provides a strong empirical foundation for computational modeling, bridging linguistic theory and practical applications, and constitutes a robust resource for future research on morphological inflection and principal parts detection. The next section shifts focus to computational methods for detecting principal parts, drawing on the linguistic insights outlined in the literature.[3]

3 Translating Linguistic Insights into Computational Methods

The linguistic principle underlying PRINCIPAL PARTS DETECTION is that principal parts encapsulate the implicative relations among cells within a lexeme's inflectional paradigm, constituting the minimal subset of cells from which all remaining cells can be deduced (Finkel and Stump, 2007; Stump and Finkel, 2013). In this work, we adopt the static principal-parts scheme, framing PRINCIPAL PARTS DETECTION as the automatic identification of a minimal, uniform subset of paradigm cells applicable consistently across all lexemes within a given syntactic category.

However, linguistic theory alone does not provide a direct computational method for systematically generalizing or approximating these implicative relations across multiple lexemes at the syntactic-category level. To operationalize PRINCIPAL PARTS DETECTION computationally, we hypothesize that implicative relationships across lexemes can be indirectly approximated through measurable morphological patterns observable within lexemes' inflectional paradigms. Specifically, we propose three types of measurable morphological patterns: (i) surface-level similarities, observed as orthographic overlap, shared morphological markers, or recurring affixation patterns across multiple cells; (ii) structural correspondences, represented by minimal transformations converting one cell's form into another; and (iii) predictive relations, characterized by the consistent ability of one cell's realization to predict another's.

By quantifying the morphological relations among cells based on these measurable patterns, we obtain empirical evidence enabling the organization of cells into meaningful subsets. We introduce the notion of sub-paradigms, computational abstractions (not formally defined in linguistic theory) that group cells whose realizations consistently display morphological and functional similarities. Cells grouped into a sub-paradigm thus implicitly share similar morphological and functional roles across lexemes, indirectly reflecting broader implicative trends, even though exact implicative relationships vary between individual lexemes.

Selecting principal parts thus naturally corresponds to choosing exactly one representative cell from each identified sub-paradigm. This ensures the resulting principal-part set compactly and effectively captures the generalized morphological roles identified through sub-paradigm formation.

This conceptualization leads directly to a three-phase computational methodology for PRINCIPAL PARTS DETECTION: First, we *characterize* morphological relationships between pairs of cells through similarity measures. Next, we *cluster* these cells into coherent sub-paradigms based on their quantified morphological similarities. Finally, we *select* exactly one representative cell from each sub-paradigm as its principal part. Together, these principal parts constitute a minimal and comprehensive set capable of systematically deriving all remaining paradigm cells across lexemes.

4 Framework and Task Empirical Design

The PRINCIPAL PARTS DETECTION framework we propose in this work is composed of three interconnected stages: *characterization*, *clustering*, and *principal-parts selection*, each implemented using well-defined computational methods. These stages operate independently, meaning that different configurations of the framework can mix and match methods in seeking the best combination. Below, we briefly review the computational methods considered for each stage.

4.1 Characterization: Quantifying Morphological Relations Between Cells

The characterization stage quantifies morphological relationships between paradigm cells by computing numerical similarity scores between them. This work explores three distinct characterization methods, each offering a different perspective on morphological relations between cells.

Edit Distance A metric that measures surface-level similarity between forms based on minimal

[3]The data is publicly available in https://github.com/DorinK/Principal-Parts-Detection.

edit operations — insertions, deletions, or substitutions — required to transform one form into another (Levenshtein, 1966). This method is implemented by computing the average Edit Distance from each paradigm cell to all others (calculated across all lexemes in the data), treating one as the source and the rest as destinations. The resulting vector representations store these averaged distances, capturing surface-level similarity between cells. Pairs of paradigm cells with low Edit Distance scores exhibit orthographic overlap.

Edit Script A metric that captures transformational diversity by analyzing character-level transformations between paradigm cells. Unlike traditional Edit Script approaches (Wagner and Fischer, 1974; Myers, 1986), which focus on the exact sequence of operations needed to transform one string into another, this approach computes the number of unique character-level transformations observed across all surface realizations of each paradigm-cell pair. Each transformation is counted only once per cell pair (calculated across all lexemes), capturing distinct transformational patterns rather than repeatedly occurring character changes. The result is a vector representation for each cell pair, where each entry encodes the number of unique transformations required to convert one cell to another, representing their transformational distance. This method provides insight into the variation in morphological transformations within a paradigm. Cells with lower transformation diversity may exhibit more stable morphological patterns, making them stronger principal part candidates. In contrast, higher transformation diversity may signal greater variability in inflectional behavior, affecting predictability.

Reinflection Accuracy A metric that evaluates the functional predictability of paradigm cells. It leverages the Base LSTM reinflection model (Goldman et al., 2021) trained to generate a target form given a source form and the morpho-syntactic features of the target. Unlike edit-based methods that focus on surface similarity and transformational diversity, Reinflection Accuracy captures functional dependencies between cells, reflecting their predictive capacity within a paradigm.

Reinflection Accuracy is particularly effective in languages with complex inflectional systems, where orthographic similarity alone is not a reliable predictor of implicative relations. By capturing functional dependencies rather than surface transformations, it provides a direct measure of a cell's ability to generate other forms. However, its performance depends on training data quality and resource availability. In low-resource settings, data sparsity may lead to biased results, and the approach is computationally intensive, as it requires training multiple models—one model per paradigm cell. Despite these challenges, its ability to model functional predictability makes it a valuable tool for identifying paradigm cells suitable as principal parts, particularly in morphologically complex languages.

Each characterization method produces a similarity table, where rows represent source paradigm cells and columns represent target paradigm cells, encoding pairwise morphological relationships (see Appendix B). Before clustering, all similarity tables are standardized by removing the mean and scaling to unit variance to ensure comparability across methods. These standardized characterization tables form the empirical basis for the clustering stage.

4.2 Clustering: Structuring Cells into Sub-Paradigms

The clustering stage groups paradigm cells based on their quantified morphological relationships, forming computational abstractions termed sub-paradigms. These sub-paradigms approximate the internal morphological organization of paradigms. The framework implements two clustering algorithms, each offering distinct computational properties. As with characterization, only one clustering algorithm is employed at a time.

Affinity Propagation A message-passing clustering algorithm that dynamically determines the number of clusters based on pairwise similarity scores (Frey and Dueck, 2007). Unlike traditional clustering methods, it does not require a predefined number of clusters. Instead, it iteratively updates responsibility and availability values, which determine how well a paradigm cell serves as an exemplar (cluster center), until the algorithm converges on a final set of exemplars. This property makes it particularly suitable for paradigms with high morphological variability. The algorithm is implemented using scikit-learn's AffinityPropagation module, with similarity scores computed as negative squared Euclidean distances. The preference parameter is set to the median similarity value, allowing clusters to emerge naturally. Additional

parameters include a convergence iteration limit of 30 and a random state value of 10.

Hierarchical K-Means A hierarchical variant of K-Means that recursively partitions paradigm cells into two clusters per iteration until a well-defined clustering structure is reached. The stopping criterion is determined using the Calinski–Harabasz Index (CHI) (Caliński and Harabasz, 1974), which evaluates clustering quality by comparing between-cluster dispersion to within-cluster cohesion. At each step, the CHI is computed across the entire clustering structure to assess how well-separated the clusters are relative to their internal cohesion. To prevent over-segmentation, clustering stops if the number of clusters in the new best CHI solution exceeds that of the previous best CHI solution by more than one cluster. The algorithm is implemented using scikit-learn's KMeans module with a random state value of 10. By grouping paradigm cells into sub-paradigms, the clustering stage provides a data-driven approximation of generalized morphological and functional roles. The resulting sub-paradigms form the structured basis for the principal-parts selection stage.

4.3 Principal Parts Selection: Identifying Representative Cells

The principal-parts selection stage finalizes the PRINCIPAL PARTS DETECTION framework by transforming sub-paradigms into a compact, generative summary of the paradigm structure. In this stage, exactly one representative cell from each sub-paradigm is selected, capturing the morphological and functional properties that characterize its sub-paradigm. These representative cells collectively constitute the principal parts, ensuring comprehensive morphological coverage while maintaining compactness and predictive capacity.

Specifically, we adopt the *Minimum Average Inflectional Length* criterion. Under this criterion, the principal part selected from each sub-paradigm is the paradigm cell whose realizations exhibit the minimal average length, computed across all lexemes. This selection ensures that the chosen cell is both structurally central and morphologically efficient within its sub-paradigm. Such a criterion aligns with linguistic insights suggesting that shorter inflectional paths often correspond to central morphological roles, enhancing their suitability as principal parts.

Together, the principal parts derived from this selection process form a minimal and comprehensive set capable of systematically deriving all remaining paradigm cells across lexemes, in accordance with the static principal-parts scheme assumption.

5 Experimental Setup and Results

We conduct a series of experiments to evaluate the effectiveness of the PRINCIPAL PARTS DETECTION framework across ten typologically diverse languages. The evaluation compares six model configurations, each formed by pairing one of three characterization methods—Edit Distance, Edit Script, and Reinflection Accuracy—with one of two clustering algorithms—Affinity Propagation and Hierarchical K-Means. To establish a performance threshold, we include a random baseline, selecting principal parts at random.

5.1 Dataset

The PRINCIPAL PARTS DETECTION dataset utilized in our experiments comprises ten typologically diverse languages, structured into two subsets to rigorously assess our framework's cross-linguistic generalization.

The first subset (Hebrew, English, French, German, Spanish) was used during method development, providing a broad and representative morphological foundation. Hebrew exhibits synthetic morphology, encoding multiple grammatical features within single inflected forms. English, in contrast, is predominantly analytic, relying primarily on word order and function words to indicate grammatical relations. French and Spanish, as fusional languages, embed tense, mood, and person distinctions within single inflectional forms, exhibiting varying degrees of morphological regularity. Finally, German presents a hybrid morphological system, integrating analytic and fusional characteristics.

The second subset (Danish, Swedish, Finnish, Turkish, Latin) was reserved exclusively for independent validation of the finalized methods' generalizability, evaluating their performance on languages not encountered during development. Finnish and Turkish exemplify agglutinative morphology, expressing grammatical information through clearly segmentable morphemes arranged sequentially. Latin, a highly inflected classical language, provides a challenging scenario due to extensive distinctions in case, number, and gender. Danish and Swedish, characterized by regular and

predictable inflectional paradigms, allow us to assess method robustness in languages with simpler morphological structures.

This structuring enables a rigorous and unbiased evaluation of our framework's adaptability, robustness, and cross-linguistic generalization across diverse morphological systems.

5.2 Evaluation Metric

To evaluate model effectiveness, we utilize the F1 score, balancing precision (correctness of predicted principal parts) and recall (coverage of gold principal parts) to assess both accuracy and completeness in PRINCIPAL PARTS DETECTION.

In addition to reporting F1 scores, we benchmark our models against a random baseline, which selects principal parts randomly within each paradigm. Given a paradigm with x cells and y gold principal parts, the probability of randomly selecting a correct principal part is $\frac{y}{x}$. As the baseline selects exactly y principal parts, the expected number of correct predictions is $y \times \frac{y}{x} = \frac{y^2}{x}$. Thus, the expected precision, recall, and consequently the F1 score, are all equal to $\frac{y}{x}$.

Since principal parts are inherently sparse within most paradigms, the random baseline represents a challenging threshold. Models that significantly exceed this baseline demonstrate an ability to detect principal parts systematically rather than relying on chance.

5.3 Reinflection Settings

For models utilizing Reinflection Accuracy, we train a separate reinflection model for each paradigm cell, treating it as the source while all other cells serve as targets. The model is based on the Base LSTM architecture (Goldman et al., 2021), a character-based sequence-to-sequence model comprising a one-layer bidirectional LSTM encoder and a one-layer unidirectional LSTM decoder with a global soft attention layer (Bahdanau et al., 2014). Each model is trained for 50 epochs, optimizing categorical cross-entropy.

The dataset is split 70%-30%, ensuring test lexemes remain unseen during training. Each paradigm cell is trained using a dedicated dataset, where it serves as the source inflection across different lexemes. Since each cell is evaluated on its ability to generate all other cells within the paradigm, corresponding test sets are created—one per target cell.

Model	Algorithmic Evaluation
Random Baseline	21.20
Edit Distance + Affinity Propagation	31.29
Edit Distance + Hierarchical K-Means	32.51
Reinflection Accuracy + Hierarchical K-Means	42.43
Edit Script + Affinity Propagation	44.62
Reinflection Accuracy + Affinity Propagation	45.56
Edit Script + Hierarchical K-Means	**55.05**

Table 1: Average F1 scores across the ten languages of our PRINCIPAL PARTS DETECTION dataset for different model configurations. The best-performing model configuration is highlighted.

Each trained model is evaluated on how accurately it inflects from its assigned source cell to each target cell. The resulting accuracy scores form a representation vector, capturing a cell's proficiency in generating others. Cells with high Reinflection Accuracy scores demonstrate strong predictive capacity, making them effective candidates for principal parts.

5.4 Results

Table 1 presents the average F1 scores across the ten languages, providing a comparative evaluation of model performance. All models outperform the random baseline, which achieves the lowest F1 score of 21.20%. The best-performing model, Edit Script + Hierarchical K-Means, achieves an F1 score of 55.05%, highlighting its ability to effectively characterize morphological relationships among paradigm cells and cluster these cells across diverse languages.

Reinflection Accuracy-based models perform competitively, with F1 scores of 45.56% (Affinity Propagation) and 42.43% (Hierarchical K-Means). In contrast, Edit Distance-based models yield lower scores of 31.29% and 32.51%, indicating that surface-level similarity alone is insufficient for PRINCIPAL PARTS DETECTION.

Overall, all tested methods surpass the random baseline by at least 10.09 points, with the best-performing model exceeding it by 33.85 points. These results confirm the effectiveness of the proposed methodology, highlighting a substantial improvement over random selection.

Table 2 provides a language-specific breakdown of F1 scores, offering further insights into models' performance across morphological typologies. Edit Script + Hierarchical K-Means, our best-performing model overall, achieves the high-

est scores in Hebrew, French, Spanish, Turkish, and Latin. This highlights its effectiveness in capturing systematic morphological transformations—particularly beneficial in languages with root-and-pattern morphology (e.g., Hebrew), fusional systems (e.g., French, Spanish, Latin), where single inflections encode multiple grammatical features simultaneously, and in Turkish, an agglutinative language characterized by clearly segmentable, predictable morphological sequences.

While the Reinflection Accuracy + Affinity Propagation model ranks second-best overall (45.56%), it does not consistently outperform other models across languages. Its strongest results appear specifically in languages characterized by relatively transparent, regular, and predictable inflectional paradigms, such as Danish and Swedish, where the exemplar-based clustering method effectively organizes paradigm cells. Conversely, its performance drops in morphologically opaque or fusional languages (e.g., Spanish, Finnish). However, the Reinflection Accuracy + Hierarchical K-Means model achieves notably stronger results in Finnish and English, indicating differences in how clustering methods handle morphological predictability. These contrasting patterns underscore the importance of carefully matching characterization methods and clustering algorithms to linguistic properties.

In contrast to the previously discussed models, the weaker performance of Edit Distance-based models is particularly evident in morphologically opaque or highly fusional languages (e.g., Spanish, Finnish), where subtle or irregular morphological variations encode multiple grammatical features simultaneously.

6 Analysis

We analyze how methodological factors shape model performance, focusing on transformations in characterization data and the effectiveness of clustering strategies. This evaluation highlights structural patterns influencing clustering quality and examines the extent to which clustering results align with ideal principal-parts selection.

6.1 Transpose Ablation: Evaluating the Impact of Data Orientation

The Transpose Ablation study investigates whether swapping the rows and columns of the characterization tables influences clustering quality and principal-parts selection. This transformation is particularly relevant for Reinflection Accuracy, where original tables encode directional relationships—rows indicate how easily a paradigm cell can inflect from itself to others, while columns represent the reverse relationship. Unlike Edit Distance and Edit Script methods, which produce symmetric similarity matrices, Reinflection Accuracy matrices are inherently asymmetric. Thus, transposing these tables meaningfully changes their directional structure and potentially impacts clustering results.

Transposition is applied only to Reinflection Accuracy models, as Edit Distance and Edit Script methods generate symmetric similarity tables, making transposition redundant. We evaluate two models: Reinflection Accuracy + Affinity Propagation and Reinflection Accuracy + Hierarchical K-Means, comparing their performance before and after transposition.

The results in Table 3 show that transposition affects models differently. Reinflection Accuracy + Affinity Propagation experiences a slight decrease in performance (45.56% $\rightarrow$ 44.05%), while Reinflection Accuracy + Hierarchical K-Means improves marginally (42.43% $\rightarrow$ 43.14%). This suggests that transposition does not universally enhance clustering effectiveness and that its impact depends on the underlying clustering strategy.

Despite the minor improvement for Hierarchical K-Means, transposed results are excluded from the main evaluation due to their limited effect and misalignment with the principal-parts definition. Because original (non-transposed) cells encode generative properties crucial for inflection, preserving this structure remains preferable. These findings suggest that alternative data transformations, better aligned with the linguistic task, may offer greater benefits.

6.2 Oracle Evaluation

To estimate the theoretical upper bound of our models' performance, we conduct an Oracle Evaluation, where principal parts are selected directly from the gold principal parts annotations rather than relying on clustering results. This evaluation disentangles the contribution of clustering quality from principal-parts selection effectiveness: a low Oracle score indicates fundamental limitations in clustering, while a significant gap between Oracle and Algorithmic scores highlights inefficiencies specifically in the principal-parts selection stage. By providing this performance ceiling, the Ora-

Model	Hebrew	English	French	German	Spanish	Danish	Swedish	Finnish	Turkish	Latin
Random Baseline	20.68	60.00	14.28	16.66	2.53	62.50	26.30	2.48	28.00	6.25
Edit Distance + Affinity Propagation	33.30	66.70	37.50	46.20	15.40	57.10	40.00	0.00	0.00	16.70
Edit Distance + Hierarchical K-Means	25.00	57.10	44.40	44.40	0.00	57.10	57.10	0.00	0.00	40.00
Reinflection Accuracy + Hierarchical K-Means	25.00	85.70	44.40	28.60	50.00	57.10	43.50	50.00	0.00	40.00
Edit Script + Affinity Propagation	50.00	80.00	54.50	66.70	36.40	50.00	60.00	23.50	6.90	18.20
Reinflection Accuracy + Affinity Propagation	36.40	80.00	26.70	60.00	16.70	75.00	75.00	46.20	17.40	22.20
Edit Script + Hierarchical K-Means	**50.00**	80.00	**54.50**	60.00	**50.00**	72.70	60.00	33.30	**50.00**	**40.00**

Table 2: Language-specific F1 scores illustrating variations in effectiveness of different model configurations across morphological typologies. Top results are marked, with a unique color used for each language.

Model	Transpose	Algorithmic Evaluation
Reinflection Accuracy + Affinity Propagation	✗	**45.56**
	✓	44.05
Reinflection Accuracy + Hierarchical K-Means	✗	42.43
	✓	**43.14**

Table 3: Algorithmic evaluation of Reinflection Accuracy models with and without transposition across ten languages. The averaged F1 scores highlight varying impacts depending on the clustering algorithm.

Model	Evaluation	
	Oracle	Algorithmic
Edit Distance + Affinity Propagation	40.08	31.29
Edit Distance + Hierarchical K-Means	50.57	32.51
Reinflection Accuracy + Affinity Propagation	58.78	45.56
Reinflection Accuracy + Hierarchical K-Means	65.64	42.43
Edit Script + Affinity Propagation	54.16	44.62
Edit Script + Hierarchical K-Means	**76.21**	**55.05**

Table 4: Oracle and Algorithmic evaluations of PRINCIPAL PARTS DETECTION models across languages. Oracle evaluation assumes perfect knowledge of principal parts, establishing an upper bound on performance; Algorithmic evaluation reflects actual model performance.

cle Evaluation identifies which components of the PRINCIPAL PARTS DETECTION framework require targeted improvement.

Table 4 reveals substantial gaps between Oracle and Algorithmic scores, underscoring clustering limitations and principal-parts selection inefficiencies. Edit Script + Hierarchical K-Means achieves the highest Oracle score (76.21%), confirming strong clustering performance. However, the 21.16-point gap suggests that principal-parts selection remains a limiting factor.

Conversely, Edit Distance + Affinity Propagation exhibits the lowest Oracle score (40.08%), indicating fundamental clustering challenges. Reinflection Accuracy + Hierarchical K-Means shows a notably large Oracle-Algorithmic gap (65.64% → 42.43%), highlighting that while clustering is effective, principal-parts selection still requires refinement.

These findings emphasize the importance of optimizing both clustering effectiveness and principal-parts selection to bridge the gap between Oracle and Algorithmic performance.

Model	Transpose	Evaluation	
		Oracle	Algorithmic
Reinflection Accuracy + Affinity Propagation	✗	**58.78**	**45.56**
	✓	58.51	44.05
Reinflection Accuracy + Hierarchical K-Means	✗	65.64	42.43
	✓	**67.70**	**43.14**

Table 5: Oracle and Algorithmic evaluations of Reinflection Accuracy models before and after transposition, assessing clustering quality under ideal (Oracle) and practical (algorithmic) conditions.

6.3 Interplay Between Transposition and Oracle Performance

Table 5 examines the impact of transposition on Reinflection Accuracy models under both Oracle and Algorithmic evaluations.

The results indicate that while transposition improves Oracle performance for Hierarchical K-Means (65.64% → 67.70%), it has a negligible effect on Algorithmic scores, indicating that while transposition enhances clustering under ideal conditions, it does not meaningfully improve principal-parts selection. Additionally, Affinity Propagation exhibits sensitivity to data orientation, showing a slight decline in Oracle performance (58.78% → 58.51%), suggesting that its clustering mechanism relies on specific directional patterns that transposition may disrupt. Conversely, Hierarchi-

cal K-Means benefits from transposed data, likely due to its iterative refinement of clusters. However, since Algorithmic scores remain largely unchanged across models, these findings reinforce that refining selection heuristics, rather than adjusting data orientation, is the key to improving model performance.

7 Related Work

Early computational approaches to paradigm completion predominantly relied on the lemma as the central reference form, treating it as the sole input for generating full inflectional paradigms (Durrett and DeNero, 2013; Hulden, 2014; Nicolai et al., 2015; Ahlberg et al., 2015; Faruqui et al., 2016). However, Cotterell et al. (2017) highlighted the limitations of this approach, noting that forcing transformations to pass exclusively through the lemma can introduce unnecessary complexity. Instead, more flexible models leveraging multiple inflected forms have been proposed, allowing transformations to occur directly or via intermediary forms, rather than constraining them to a single privileged form. This shift aligns with the concept of principal parts, defined as the minimal set of paradigm cells required to deduce all others (Finkel and Stump, 2007; Stump and Finkel, 2013).

Cotterell et al. (2017) introduced a directed graphical model that probabilistically generates missing inflected forms by modeling dependencies within paradigms. This approach enables the prediction of a form from multiple inflected forms rather than exclusively from the lemma. Around the same time, Kann et al. (2017) introduced multi-source reinflection, demonstrating that using multiple inflected forms as input improves accuracy. Their work explicitly references principal parts as a linguistic motivation, reinforcing the idea that certain cells within a paradigm hold stronger predictive capacity. Additionally, Cotterell et al. (2019) examined the structural complexity of inflectional paradigms, proposing a neural method for ordering paradigm slots based on their predictability—an indirect computational realization of the principal parts concept.

Liu and Hulden (2020) extended these ideas by reformulating morphological inflection as a Paradigm Cell Filling Problem (PCFP), where missing forms are inferred from a partially observed set of paradigm cells. While their work does not explicitly model principal parts, it aligns with their predictive role in improving inflectional accuracy, particularly in low-resource settings.

Despite these advancements, no prior work has proposed a systematic, data-driven approach to PRINCIPAL PARTS DETECTION. Existing studies have either assumed pre-defined principal parts or incorporated them indirectly within broader inflectional tasks. In contrast, we have introduced PRINCIPAL PARTS DETECTION as a formal computational task, developed a multilingual benchmark, and proposed a principled methodology for automatic PRINCIPAL PARTS DETECTION. By integrating linguistic insights with computational modeling, we establish a structured framework for PRINCIPAL PARTS DETECTION.

8 Conclusions

This work introduces PRINCIPAL PARTS DETECTION as a computational task, formalizing the detection of principal parts within inflectional paradigms under the static principal-parts scheme assumption. We construct a multilingual dataset covering ten typologically diverse languages and develop a structured framework to automatically detect principal parts uniformly applicable across all lexemes belonging the verb syntactic category.

Our empirical evaluation demonstrates that quantifying morphological relationships between cells, clustering these cells into sub-paradigms, and selecting representative cells from each sub-paradigm provide a viable strategy for identifying principal parts. Our best-performing approach — Edit Script similarity combined with Hierarchical K-Means clustering — achieves an F1 score of 55.05%, significantly surpassing the random baseline of 21.20%. However, results across evaluated models indicate that while clustering effectively organizes paradigm cells into meaningful subsets, principal-parts selection remains a key bottleneck.

Beyond theoretical interest, successfully addressing PRINCIPAL PARTS DETECTION has practical implications for computational morphology. By identifying compact, generative subsets of paradigm cells, principal parts can be leveraged to optimize morphological inflection models, reduce annotation costs, and improve low-resource language modeling. The structured computational approach presented here lays the foundation for future advancements, highlighting the relevance of linguistic insights in shaping more efficient NLP methodologies.

Limitations

Despite the progress demonstrated in this study, several open challenges remain. Irregular paradigms, as seen in Latin, continue to pose difficulties, highlighting the need for methods that can better capture morphological unpredictability. Additionally, our reliance on UniMorph, while offering broad linguistic coverage, exposes inconsistencies that impact model generalization. More curated linguistic resources could improve dataset reliability and refine the evaluation of principal parts across languages.

Future work could explore alternative clustering strategies better suited to morphological structures, such as graph-based methods or neural clustering approaches. Transformer-based models may hold potential for capturing deeper morphological dependencies, offering an avenue for enhancing both clustering accuracy and principal-parts selection. These challenges are beyond the scope of this paper and reserved for future research.

Finally, our dataset currently includes only ten languages. Expanding the dataset to include additional morphologically rich and underrepresented languages, such as polysynthetic languages, would more comprehensively capture typological diversity and potentially further validate the robustness of PRINCIPAL PARTS DETECTION methods.

Acknowledgments

This project is funded by a generous grant from the Israeli Science Foundation (#ISF-670/23), and a KAMIN grant (#81698) from the Israeli Innovation Authority (IAA), for which we are grateful. In addition, the first author is funded by a VATAT grant, and the second author is partially funded by the UniDive COST Action (#CA21167), which we thankfully acknowledge. We finally with to thank three anonymous reviewers for their insightful comments on an earlier draft.

References

Malin Ahlberg, Markus Forsberg, and Mans Hulden. 2015. Paradigm classification in supervised learning of morphology. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1024–1029, Denver, Colorado. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Khuyagbaatar Batsuren, Omer Goldman, Salam Khalifa, Nizar Habash, Witold Kieraś, Gábor Bella, Brian Leonard, Garrett Nicolai, Kyle Gorman, Yustinus Ghanggo Ate, Maria Ryskina, Sabrina Mielke, Elena Budianskaya, Charbel El-Khaissi, Tiago Pimentel, Michael Gasser, William Abbott Lane, Mohit Raj, Matt Coler, Jaime Rafael Montoya Samame, Delio Siticonatzi Camaiteri, Esaú Zumaeta Rojas, Didier López Francis, Arturo Oncevay, Juan López Bautista, Gema Celeste Silva Villegas, Lucas Torroba Hennigen, Adam Ek, David Guriel, Peter Dirix, Jean-Philippe Bernardy, Andrey Scherbakov, Aziyana Bayyr-ool, Antonios Anastasopoulos, Roberto Zariquiey, Karina Sheifer, Sofya Ganieva, Hilaria Cruz, Ritván Karahóǧa, Stella Markantonatou, George Pavlidis, Matvey Plugaryov, Elena Klyachko, Ali Salehi, Candy Angulo, Jatayu Baxi, Andrew Krizhanovsky, Natalia Krizhanovskaya, Elizabeth Salesky, Clara Vania, Sardana Ivanova, Jennifer White, Rowan Hall Maudslay, Josef Valvoda, Ran Zmigrod, Paula Czarnowska, Irene Nikkarinen, Aelita Salchak, Brijesh Bhatt, Christopher Straughn, Zoey Liu, Jonathan North Washington, Yuval Pinter, Duygu Ataman, Marcin Wolinski, Totok Suhardijanto, Anna Yablonskaya, Niklas Stoehr, Hossep Dolatian, Zahroh Nuriah, Shyam Ratan, Francis M. Tyers, Edoardo M. Ponti, Grant Aiton, Aryaman Arora, Richard J. Hatcher, Ritesh Kumar, Jeremiah Young, Daria Rodionova, Anastasia Yemelina, Taras Andrushko, Igor Marchenko, Polina Mashkovtseva, Alexandra Serova, Emily Prud'hommeaux, Maria Nepomniashchaya, Fausto Giunchiglia, Eleanor Chodroff, Mans Hulden, Miikka Silfverberg, Arya D. McCarthy, David Yarowsky, Ryan Cotterell, Reut Tsarfaty, and Ekaterina Vylomova. 2022. UniMorph 4.0: Universal Morphology. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 840–855, Marseille, France. European Language Resources Association.

Tadeusz Caliński and Jerzy Harabasz. 1974. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27.

Ryan Cotterell, Christo Kirov, Mans Hulden, and Jason Eisner. 2019. On the complexity and typology of inflectional morphological systems. *Transactions of the Association for Computational Linguistics*, 7:327–342.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. The sigmorphon 2016 shared task—morphological reinflection. In *Proceedings of the 14th SIGMORPHON workshop on computational research in phonetics, phonology, and morphology*, pages 10–22.

Ryan Cotterell, John Sylak-Glassman, and Christo Kirov. 2017. Neural graphical models over strings for principal parts morphological paradigm completion. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 759–765, Valencia, Spain. Association for Computational Linguistics.

Greg Durrett and John DeNero. 2013. Supervised learning of complete morphological paradigms. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1185–1195, Atlanta, Georgia. Association for Computational Linguistics.

Manaal Faruqui, Yulia Tsvetkov, Graham Neubig, and Chris Dyer. 2016. Morphological inflection generation using character sequence to sequence learning. *arXiv preprint arXiv:1512.06110*.

Raphael Finkel and Gregory Stump. 2007. Principal parts and morphological typology. *Morphology*, 17:39–75.

Brendan J. Frey and Dmitri Dueck. 2007. Clustering by passing messages between data points. *Science*, 315(5814):972–976.

Omer Goldman, Khuyagbaatar Batsuren, Salam Khalifa, Aryaman Arora, Garrett Nicolai, Reut Tsarfaty, and Ekaterina Vylomova. 2023. SIGMORPHON–UniMorph 2023 shared task 0: Typologically diverse morphological inflection. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 117–125, Toronto, Canada. Association for Computational Linguistics.

Omer Goldman, David Guriel, and Reut Tsarfaty. 2021. (un) solving morphological inflection: Lemma overlap artificially inflates models' performance. *arXiv preprint arXiv:2108.05682*.

Mans Hulden. 2014. Generalizing inflection tables into paradigms with finite state operations. In *Proceedings of the 2014 Joint Meeting of SIGMORPHON and SIGFSM*, pages 29–36, Baltimore, Maryland. Association for Computational Linguistics.

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2017. Neural multi-source morphological reinflection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 514–524, Valencia, Spain. Association for Computational Linguistics.

Vladimir I. Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet physics doklady*, 10(8):707–710.

Ling Liu and Mans Hulden. 2020. Leveraging principal parts for morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 153–161.

Eugene W Myers. 1986. An o (nd) difference algorithm and its variations. *Algorithmica*, 1(1):251–266.

Garrett Nicolai, Colin Cherry, and Grzegorz Kondrak. 2015. Inflection generation as discriminative string transduction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 922–931, Denver, Colorado. Association for Computational Linguistics.

Gregory Stump and Raphael A. Finkel. 2013. *Principal parts*, Cambridge Studies in Linguistics, page 9–39. Cambridge University Press.

Robert A Wagner and Michael J Fischer. 1974. The string-to-string correction problem. *Journal of the ACM (JACM)*, 21(1):168–173.

Appendix

A Technical Overview of the PRINCIPAL PARTS DETECTION Dataset

This section provides the technical details of the PRINCIPAL PARTS DETECTION dataset, including the number of samples per feature set in each language's verb paradigm and the total number of gold principal parts for each language. In some cases, specific feature sets were removed for various reasons, as explained in subsection A.2.

Additionally, we list the gold principal parts for each language, formatted as `feature_set (e.g., form)`. When two feature sets share the same form, the gold principal parts are listed in square brackets `[ ]`. The first feature set corresponds to the principal part identified in linguistic literature, while the second represents a feature set that consistently shares the same form across all samples in the dataset. In such cases, the second feature set is included as a possible principal part, as the algorithm's choice between them does not affect the analysis. To avoid redundancy, no principal part is counted more than once in these scenarios.

A.1 Dataset Summary and Illustrative Lexeme Examples

For each language, we provide an example lexeme to illustrate the principal parts, formatted as `feature_set (e.g., form)`. These examples are illustrative and may not share the same meanings across languages.

A.2 Explanatory Notes

The following explanatory notes clarify decisions made during dataset preparation and supplement the information presented in Table 6:

- **Spanish**: PRO feature sets, representing verbs with object clitic pronouns, were removed.
- **Swedish**: The V-IMP-PASS feature set was excluded due to insufficient samples (only three).
- **Latin**:
  - Passive feature sets were excluded.
  - Feature sets starting with V.PTCP (instead of V-V.PTCP) were removed.
  - Feature sets with 30 or fewer samples were excluded.
  - The first-person-singular-perfect-active-indicative feature set was excluded from the gold principal parts list due to insufficient data (only two samples).

B Characterization Tables for Selected Languages

To illustrate the structure of the characterization methods, we present detailed characterization tables for three representative languages from our dataset. These tables demonstrate how different feature sets relate within the verb morphology of each language, showcasing the variation across Edit Distance, Edit Script, and Reinflection Accuracy characterization methods.

Each language is represented by three tables, corresponding to the distinct characterization methods, with principal parts highlighted in yellow for clarity. Additionally, cases where two feature sets consistently share the same form and are interchangeable as principal parts are marked with a distinct color. Since these feature sets carry identical information, the model's selection between them does not impact the results.

Interpretation of Tables. The provided tables exemplify the structure of the characterization methods rather than an exhaustive display of all ten languages in our study. While specific lexeme examples are shown in the rows and columns, the quantified relationships they capture apply to the entire verb morphology of each language. These examples serve to illustrate the broader implicative patterns identified during the characterization process.

B.1 Characterization Tables for English

Figures 1, 2, and 3 illustrate the Edit Distance, Edit Script, and Reinflection Accuracy characterization tables for English, respectively.

B.2 Characterization Tables for German

Figures 4, 5, and 6 illustrate the characterization tables for German.

B.3 Characterization Tables for Swedish

Figures 7, 8, and 9 present the characterization tables for Swedish.

Language	Features	Samples per Feature Set	# of Gold Principal Parts	Gold Principal Parts
Hebrew	29	848–1,042	6	V-NFIN, (e.g., le'echol), V-2-SG-IMP-MASC, (e.g., echol!), V-3-SG-FUT-MASC, (e.g., yochal), V-3-SG-PST-MASC, (e.g., achal), V-SG-PRS-MASC, (e.g., ochel), V.MSDR (e.g., achila)
English	5	23,896–31,848	3	V-NFIN-IMP+SBJV (e.g., eat), V-PST (e.g., ate), V-V.PTCP-PST (e.g., eaten)
French	49	7,483–7,535	7	V-NFIN (e.g., mangier), V-IND-PRS-1-PL (e.g., manjons), V.PTCP-PST (e.g., mangié), V-IND-FUT-1-SG (e.g., mangerai), V-IND-PRS-1-SG (e.g., manju), V-IND-PRS-3-PL (e.g., manjüent), V-IND-PST-1-SG-PFV (e.g., manjai)
German	30	2,307–6,661	5	V-NFIN (e.g., essen), V.PTCP-PST (e.g., gegessen), [V-IND-SG-3-PST, V-IND-SG-1-PST (e.g., aß)], V-IND-SG-3-PRS (e.g., isst), [V-SBJV-SG-3-PST, V-SBJV-SG-1-PST (e.g., äße)]
Spanish	79	6,676–6,695	2	V-NFIN (e.g., comer), V-IND-PRS-1-SG (e.g., como)
Danish	8	162	5	V-ACT-NFIN (e.g., danse), V-ACT-IND-PRS (e.g., danser), V-ACT-IND-PST (e.g., dansede), V-ACT-IMP (e.g., dans), V.PTCP-PASS-PST (e.g., danset)
Swedish	19	2,114–2,536	5	[V-NFIN-ACT, V-IND-PL-ACT-PRS (e.g., äta)], V-IND-SG-ACT-PRS (e.g., äter), V-IND-SG-ACT-PST (e.g., åt), V-V.CVB-ACT (e.g., ätit), V-IMP-ACT (e.g., ät)
Finnish	161	7,221–7,226	4	V-NFIN-ACT+PASS (e.g., syödä), V-ACT-PRS-POS-IND-1-SG (e.g., syön), V-ACT-PST-POS-IND-3-SG (e.g., söi), V.PTCP-ACT-PST (e.g., syönyt)
Turkish	703	588	2	V-NFIN (e.g., içmek), V-IND-PRS-HAB-3-SG-POS-DECL (e.g., içer)
Latin	48	450–947	3	V-IND-ACT-PRS-1-SG (e.g., -pleō), V-NFIN-ACT-PRS (e.g., -plēre), V-V.MSDR-ACC-LGSPEC1 (e.g., -plētum)

Table 6: Summary of the PRINCIPAL PARTS DETECTION dataset by language, including gold principal parts and illustrative lexeme examples.

	Features	V-NFIN-IMP+SBJV - eat	V-PRS-3-SG - eats	V-PST - ate	V-V.PTCP-PRS - eating	V-V.PTCP-PST - eaten
1	V-NFIN-IMP+SBJV - eat	0	1.157683294	1.532683294	3.088508537	1.534943087
2	V-PRS-3-SG - eats	1.157683294	0	1.421493137	3.087504185	1.410905591
3	V-PST - ate	1.532683294	1.421493137	0	3.066078005	0.048627385
4	V-V.PTCP-PRS - eating	3.088508537	3.087504185	3.066078005	0	3.034273519
5	V-V.PTCP-PST - eaten	1.534943087	1.410905591	0.048627385	3.034273519	0

Figure 1: Average edit distances for the English verb paradigm. Values range from 0 to 3.088. Darker red shades indicate closer relationships between feature sets, while darker turquoise shades represent greater differences.

	Features	V-NFIN-IMP+SBJV - eat	V-PRS-3-SG - eats	V-PST - ate	V-V.PTCP-PRS - eating	V-V.PTCP-PST - eaten
1	V-NFIN-IMP+SBJV - eat	1	27	117	51	124
2	V-PRS-3-SG - eats	29	1	110	48	117
3	V-PST - ate	124	110	1	116	43
4	V-V.PTCP-PRS - eating	55	59	119	1	121
5	V-V.PTCP-PST - eaten	128	118	45	119	1

Figure 2: Edit Script scores for the English verb paradigm. Values range from 1 to 128. Darker purple shades indicate fewer unique character sets (closer relationships), while darker air-force-blue shades reflect greater variation.

	Features	V-NFIN-IMP+SBJV - eat	V-PRS-3-SG - eats	V-PST - ate	V-V.PTCP-PRS - eating	V-V.PTCP-PST - eaten
1	V-NFIN-IMP+SBJV - eat	0.95	0.96	0.92	0.94	0.92
2	V-PRS-3-SG - eats	0.95	0.96	0.92	0.94	0.91
3	V-PST - ate	0.9	0.91	0.96	0.94	0.95
4	V-V.PTCP-PRS - eating	0.91	0.92	0.92	0.95	0.92
5	V-V.PTCP-PST - eaten	0.91	0.91	0.96	0.95	0.96

Figure 3: Reinflection Accuracy scores for the English verb paradigm. Values range from 0.9 to 0.96. Darker teal shades indicate higher accuracy, while darker pink shades reflect lower performance.

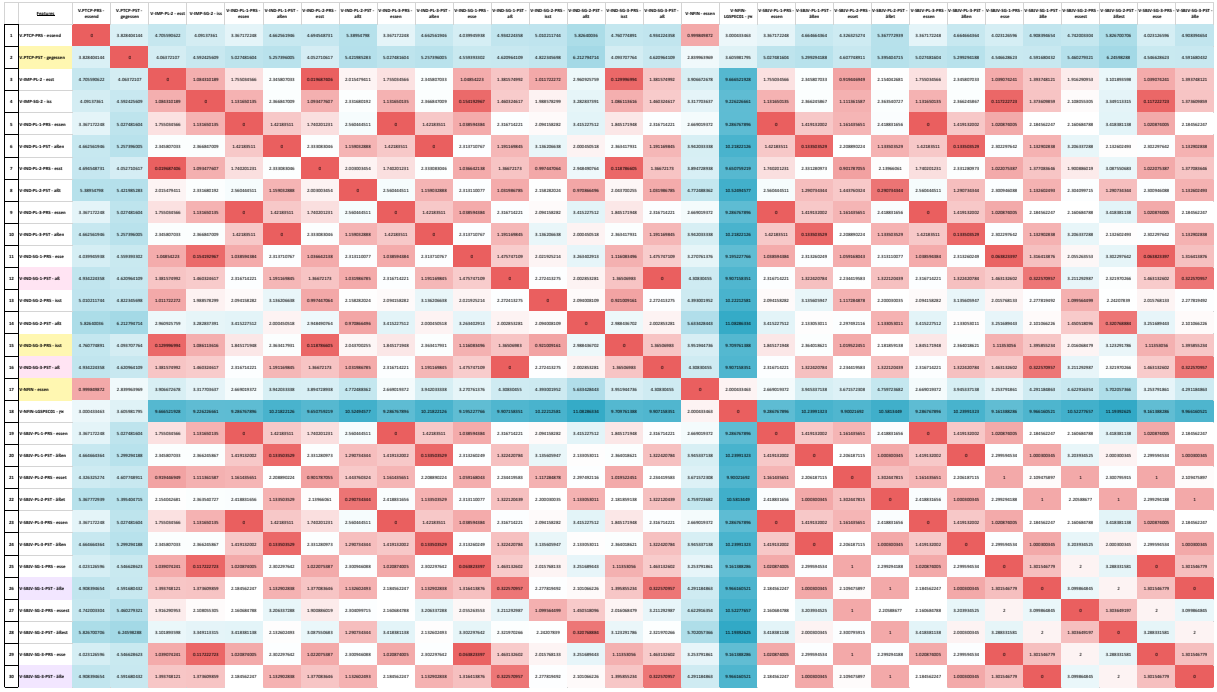


Figure 4: Average edit distances for the German verb paradigm. Values range from 0 to 11.19. Darker red shades indicate closer relationships between feature sets, while darker ball-blue shades represent greater distances.

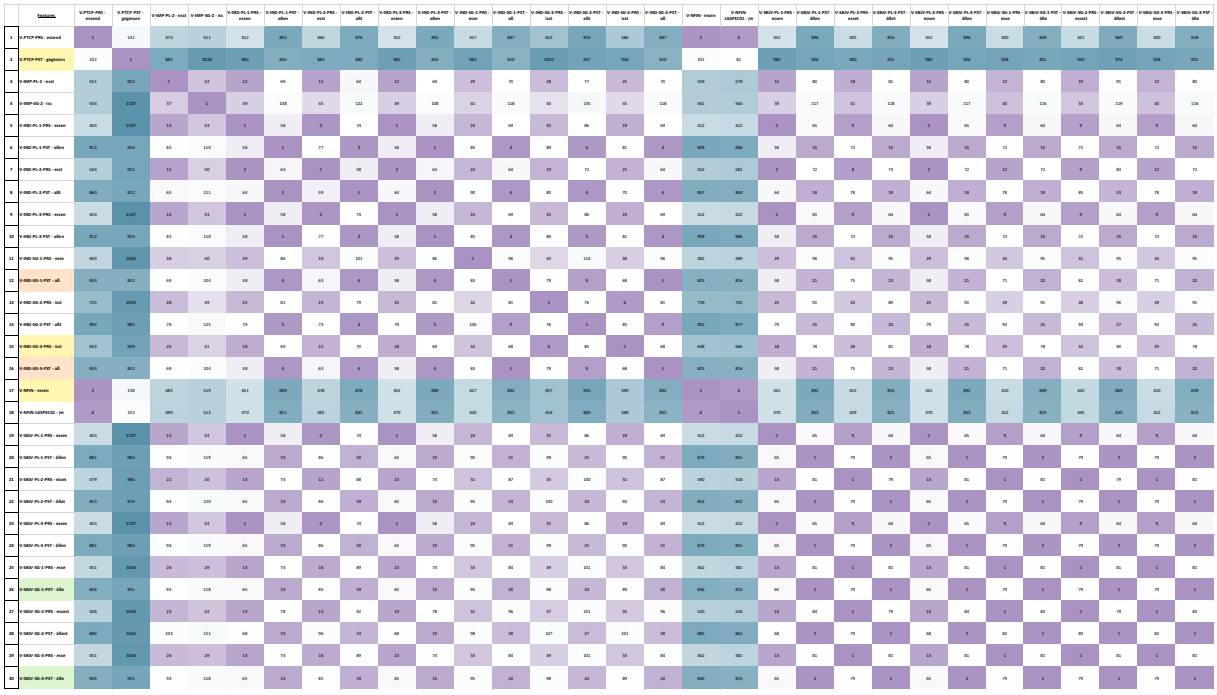


Figure 5: Edit Script scores for the German Verb Paradigm. Values range from 1 to 1,107. Darker purple shades indicate fewer unique character sets (closer relationships), while darker air-force-blue shades reflect greater variation.

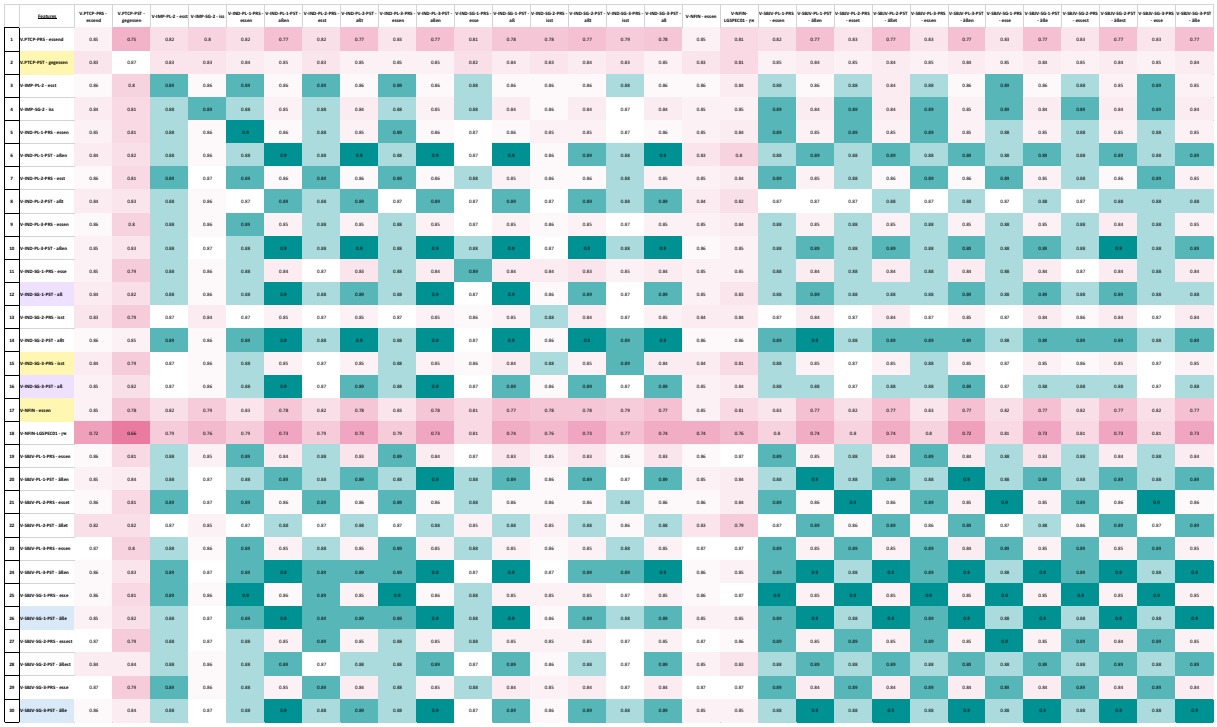
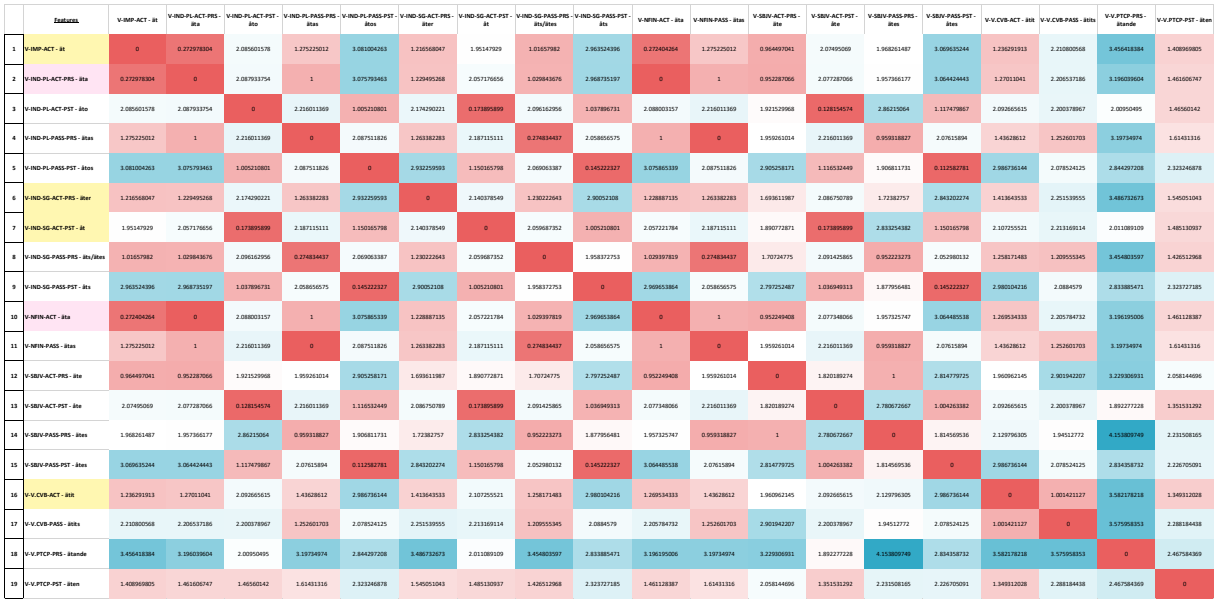
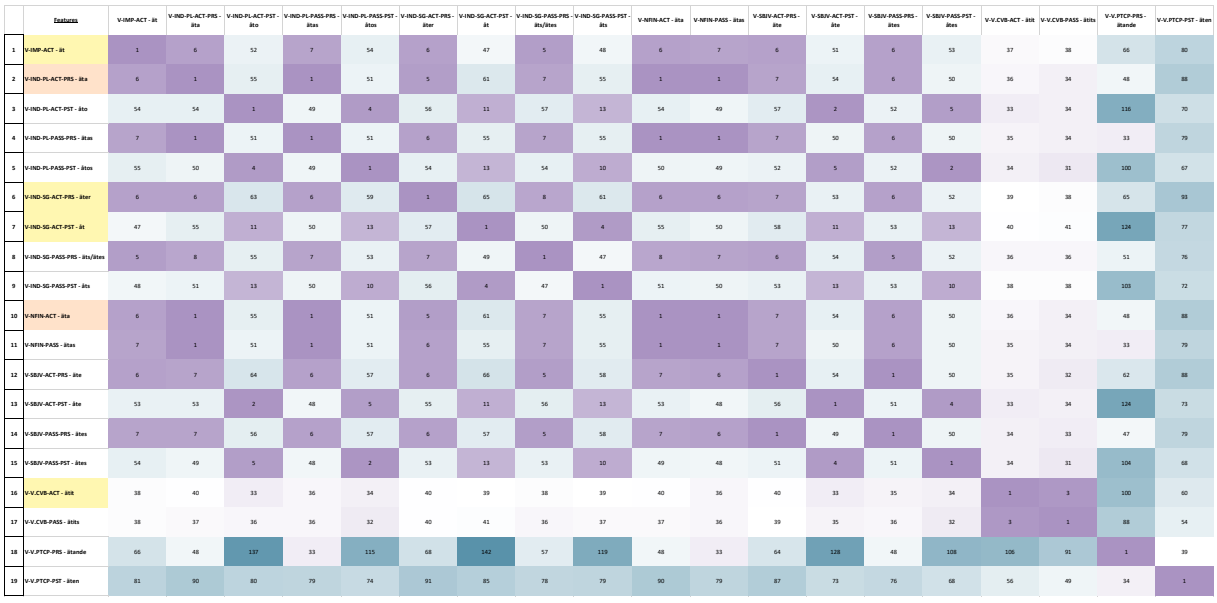


Figure 6: Reinflection Accuracy scores for the German verb paradigm. Values range from 0.66 to 0.9. Darker teal shades indicate higher accuracy, while darker pink shades reflect lower performance.



	Features	V-IMP-ACT - ät	V-IND-PL-ACT-PRS - äta	V-IND-PL-ACT-PST - äto	V-IND-PL-PASS-PRS - ätas	V-IND-PL-PASS-PST - ätos	V-IND-SG-ACT-PRS - äter	V-IND-SG-ACT-PST - ät	V-IND-SG-PASS-PRS - äts/ätes	V-IND-SG-PASS-PST - äts	V-NFIN-ACT - äta	V-NFIN-PASS - ätas	V-SBJV-ACT-PRS - äte	V-SBJV-ACT-PST - äte	V-SBJV-PASS-PRS - ätes	V-SBJV-PASS-PST - ätes	V-V.CVB-ACT - ätit	V-V.CVB-PASS - ätits	V-V.PTCP-PRS - ätande	V-V.PTCP-PST - äten
1	V-IMP-ACT - ät	0	0.272978304	2.085601578	1.275225012	3.081004263	1.216568047	1.95147929	1.01657982	2.963524396	0.272404264	1.275225012	0.964497041	2.07495069	1.968261487	3.069635244	1.236291913	2.210800568	3.456418384	1.408969805
2	V-IND-PL-ACT-PRS - äta	0.272978304	0	2.087933754	1	3.075793463	1.229495268	2.057176656	1.029843676	2.968735197	0	1	0.952287066	2.077287066	1.957366177	3.064424443	1.27011041	2.206537186	3.196039604	1.461606747
3	V-IND-PL-ACT-PST - äto	2.085601578	2.087933754	0	2.216011369	1.005210801	2.174290221	0.173895899	2.096162956	1.037896731	2.088003157	2.216011369	1.921529968	0.128154574	2.86215064	1.117479867	2.092665615	2.200378967	2.00950495	1.46560142
4	V-IND-PL-PASS-PRS - ätas	1.275225012	1	2.216011369	0	2.087511826	1.263382283	2.187115111	0.274834437	2.058656575	1	0	1.959261014	2.216011369	0.959318827	2.07615894	1.43628612	1.252601703	3.19734974	1.61431316
5	V-IND-PL-PASS-PST - ätos	3.081004263	3.075793463	1.005210801	2.087511826	0	2.932259593	1.150165798	2.069063387	0.145222327	3.075865339	2.087511826	2.905258171	1.116532449	1.906811731	0.112582781	2.986736144	2.078524125	2.844297208	2.323246878
6	V-IND-SG-ACT-PRS - äter	1.216568047	1.229495268	2.174290221	1.263382283	2.932259593	0	2.140378549	1.230222643	2.90052108	1.228887135	1.263382283	1.693611987	2.086750789	1.72382757	2.843202274	1.413643533	2.251539555	3.486732673	1.545051043
7	V-IND-SG-ACT-PST - ät	1.95147929	2.057176656	0.173895899	2.187115111	1.150165798	2.140378549	0	2.059687352	1.005210801	2.057221784	2.187115111	1.890772871	0.173895899	2.833254382	1.150165798	2.107255521	2.213169114	2.011089109	1.485130937
8	V-IND-SG-PASS-PRS - äts/ätes	1.01657982	1.029843676	2.096162956	0.274834437	2.069063387	1.230222643	2.059687352	0	1.958372753	1.029397819	0.274834437	1.70724775	2.091425865	0.952232273	2.052980132	1.258171483	1.209555345	3.454803597	1.426512968
9	V-IND-SG-PASS-PST - äts	2.963524396	2.968735197	1.037896731	2.058656575	0.145222327	2.90052108	1.005210801	1.958372753	0	2.969653864	2.058656575	2.797252487	1.036949313	1.877956481	0.145222327	2.980104216	2.0884579	2.833885471	2.323727185
10	V-NFIN-ACT - äta	0.272404264	0	2.088003157	1	3.075865339	1.228887135	2.057221784	1.029397819	2.969653864	0	1	0.952249408	2.077348066	1.957325747	3.064485538	1.269534333	2.205784732	3.196195006	1.461128387
11	V-NFIN-PASS - ätas	1.275225012	1	2.216011369	0	2.087511826	1.263382283	2.187115111	0.274834437	2.058656575	1	0	1.95926104	2.216011369	0.959318827	2.07615894	1.43628612	1.252601703	3.19734974	1.61431316
12	V-SBJV-ACT-PRS - äte	0.964497041	0.952287066	1.921529968	1.959261014	2.905258171	1.693611987	1.890772871	1.70724775	2.797252487	0.952249408	1.959261014	0	1.820189274	1	2.814779725	1.960962145	2.901942207	3.229306931	2.058144696
13	V-SBJV-ACT-PST - äte	2.07495069	2.077287066	0.128154574	2.216011369	1.116532449	2.086750789	0.173895899	2.091425865	1.036949313	2.077348066	2.216011369	1.820189274	0	2.780672667	1.004263382	2.092665635	2.200378967	1.892277228	1.351531292
14	V-SBJV-PASS-PRS - ätes	1.968261487	1.957366177	2.86215064	0.959318827	1.906811731	1.72382757	2.833254382	0.952232273	1.877956481	1.957325747	0.959318827	1	2.780672667	0	1.814569536	2.129796305	1.94512772	4.153809749	2.231508165
15	V-SBJV-PASS-PST - ätes	3.069635244	3.064424443	1.117479867	2.07615894	0.112582781	2.843202274	1.150165798	2.052980132	0.145222327	3.064485538	2.07615894	2.814779725	1.004263382	1.814569536	0	2.986736144	2.078524125	2.834358732	2.226705091
16	V-V.CVB-ACT - ätit	2.236291913	1.27011041	2.092665615	1.43628612	2.986736144	1.413643533	2.107255521	1.258171483	2.980104216	1.269534333	1.43628612	1.960962145	2.092665615	2.129796305	2.986736144	0	1.001421127	3.582178218	1.349312028
17	V-V.CVB-PASS - ätits	2.210800568	2.206537186	2.200378967	1.252601703	2.078524125	2.251539555	2.213169114	1.209555345	2.0884579	2.205784732	1.252601703	2.901942207	2.200378967	1.94512772	2.078524125	1.001421127	0	3.575958353	2.288184438
18	V-V.PTCP-PRS - ätande	3.456418384	3.196039604	2.00950495	3.19734974	2.844297208	3.486732673	2.011089109	1.454803597	2.833885471	3.196195006	3.19734974	3.229306931	1.892277228	4.153809749	2.834358732	3.582178218	3.575958353	0	2.467584369
19	V-V.PTCP-PST - äten	1.408969805	1.461606747	1.46560142	1.61431316	2.323246878	1.545051043	1.485130937	1.426512968	2.323727185	1.461128387	1.61431316	2.058144696	1.351531292	2.231508165	2.226705091	1.349312028	2.288184438	2.467584369	0

Figure 7: Average edit distances for the Swedish verb paradigm. Values range from 0 to 4.153. Darker red shades indicate closer relationships between feature sets, while darker ball-blue shades represent greater differences.



	Features	V-IMP-ACT - ät	V-IND-PL-ACT-PRS - äta	V-IND-PL-ACT-PST - äto	V-IND-PL-PASS-PRS - ätas	V-IND-PL-PASS-PST - ätos	V-IND-SG-ACT-PRS - äter	V-IND-SG-ACT-PST - ät	V-IND-SG-PASS-PRS - äts/ätes	V-IND-SG-PASS-PST - äts	V-NFIN-ACT - äta	V-NFIN-PASS - ätas	V-SBJV-ACT-PRS - äte	V-SBJV-ACT-PST - äte	V-SBJV-PASS-PRS - ätes	V-SBJV-PASS-PST - ätes	V-V.CVB-ACT - ätit	V-V.CVB-PASS - ätits	V-V.PTCP-PRS - ätande	V-V.PTCP-PST - äten
1	V-IMP-ACT - ät	1	6	52	7	54	6	47	5	48	6	7	6	51	6	53	37	38	66	80
2	V-IND-PL-ACT-PRS - äta	6	1	55	1	51	5	61	7	55	1	1	7	54	6	50	36	34	48	88
3	V-IND-PL-ACT-PST - äto	54	54	1	49	4	56	11	57	13	54	49	57	2	52	5	33	34	116	70
4	V-IND-PL-PASS-PRS - ätas	7	1	51	1	51	6	55	7	55	1	1	7	50	6	50	35	34	33	79
5	V-IND-PL-PASS-PST - ätos	55	50	4	49	1	54	13	54	10	50	49	52	5	52	2	34	31	100	67
6	V-IND-SG-ACT-PRS - äter	6	6	63	6	59	1	65	8	61	6	6	7	53	6	52	39	38	65	93
7	V-IND-SG-ACT-PST - ät	47	55	11	50	13	57	1	50	4	55	50	58	11	53	13	40	41	124	77
8	V-IND-SG-PASS-PRS - äts/ätes	5	8	55	7	53	7	49	1	47	8	7	6	54	5	52	36	36	51	76
9	V-IND-SG-PASS-PST - äts	48	51	13	50	10	56	4	47	1	51	50	53	13	53	10	38	38	103	72
10	V-NFIN-ACT - äta	6	1	55	1	51	5	61	7	55	1	1	7	54	6	50	36	34	48	88
11	V-NFIN-PASS - ätas	7	1	51	1	51	6	55	7	55	1	1	7	50	6	50	35	34	33	79
12	V-SBJV-ACT-PRS - äte	6	7	64	6	57	6	66	5	58	7	6	1	54	1	50	35	32	62	88
13	V-SBJV-ACT-PST - äte	53	53	2	48	5	55	11	56	13	53	48	56	1	51	4	33	34	124	73
14	V-SBJV-PASS-PRS - ätes	7	7	56	6	57	6	57	5	58	7	6	1	49	1	50	34	33	47	79
15	V-SBJV-PASS-PST - ätes	54	49	5	48	2	53	13	53	10	49	48	51	4	51	1	34	31	104	68
16	V-V.CVB-ACT - ätit	38	40	33	36	34	40	39	38	39	40	36	40	33	35	34	1	3	100	60
17	V-V.CVB-PASS - ätits	38	37	36	36	32	40	41	36	37	37	36	39	35	36	32	3	1	88	54
18	V-V.PTCP-PRS - ätande	66	48	137	33	115	68	142	57	119	48	33	64	128	48	108	106	91	1	39
19	V-V.PTCP-PST - äten	81	90	80	79	74	91	85	78	79	90	79	87	73	76	68	56	49	34	1

Figure 8: Edit Script scores for the Swedish verb paradigm. Values range from 1 to 142. Darker purple shades indicate fewer unique character sets (closer relationships), while darker air-force-blue shades reflect greater variation.

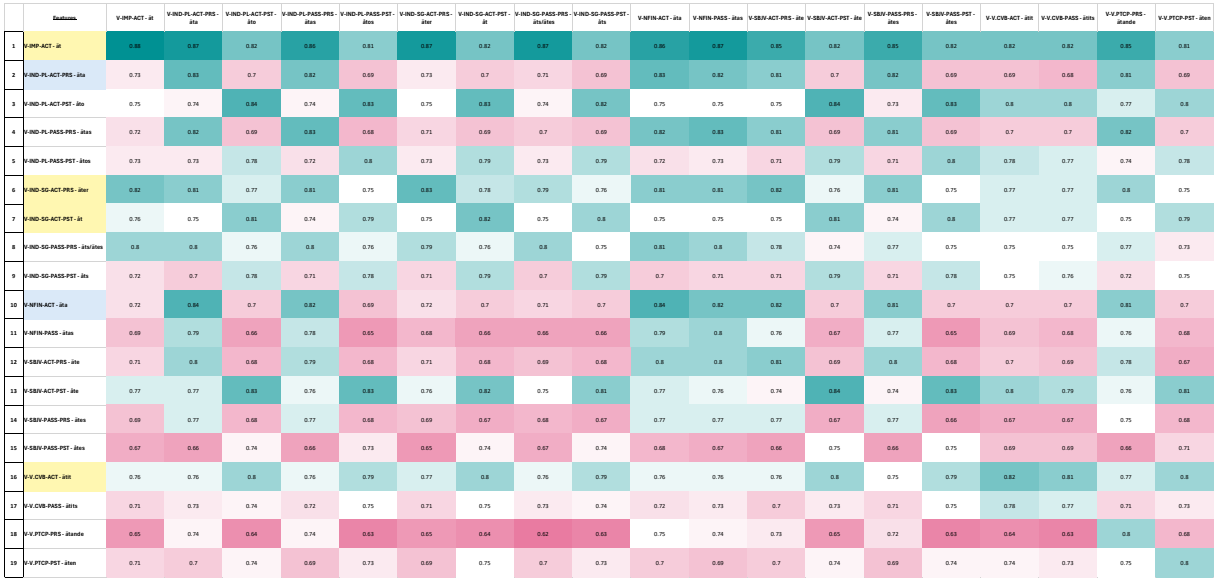


Figure 9: Reinflection Accuracy scores for the Swedish verb paradigm. Values range from 0.62 to 0.88. Darker teal shades indicate higher accuracy, while darker pink shades reflect lower performance.