Findings of the SIGMORPHON 2021 Shared Task on Unsupervised Morphological Paradigm Clustering

Adam Wiemerslage[†] **Arya McCarthy**[‡] **Alexander Erdmann**[▽]
Garrett Nicolai[ψ] **Manex Agirrezabal** [ϕ] **Miikka Silfverberg**[ψ]
Mans Hulden[†] **Katharina Kann**[†]
[†]University of Colorado Boulder [‡]Johns Hopkins University [▽]Ohio State University
[ϕ]University of Copenhagen [ψ]University of British Columbia
{adam.wiemerslage,katharina.kann}@colorado.edu

Abstract

We describe the second SIGMORPHON shared task on unsupervised morphology: the goal of the SIGMORPHON 2021 Shared Task on Unsupervised Morphological Paradigm Clustering is to cluster word types from a raw text corpus into paradigms. To this end, we release corpora for 5 development and 9 test languages, as well as gold partial paradigms for evaluation. We receive 14 submissions from 4 teams that follow different strategies, and the best performing system is based on adaptor grammars. Results vary significantly across languages. However, all systems are outperformed by a supervised lemmatizer, implying that there is still room for improvement.

1 Introduction

In recent years, most research in the area of computational morphology has focused on the application of supervised machine learning methods to word inflection: generating the inflected forms of a word, often a lemma, in order to express certain grammatical properties. For example, a supervised inflection system for Spanish might be provided with a lemma *disfrutar* (English: *to enjoy*) and morphological features such as *indicative*, *present tense* & *1st person singular*, and generate the corresponding inflected form *disfruto* as output.

However, a supervised machine learning setup is quite different from a human first language (L1) acquisition setting. Young children must learn to segment a continuous speech signal into discrete words and perform unsupervised classification, decoding, and eventually, inference with incomplete feedback on this noisy input. The task of unsupervised paradigm clustering aims to replicate one of the steps in this process—namely, the grouping of word forms belonging to the same lexeme into inflectional paradigms. In this unsupervised task, a system does not know about lemmas. Furthermore, neither does it know (a) the features for which a lemma typically inflects, nor (b) the number of distinct inflected forms which constitute the paradigm.

Figure 1: Unsupervised morphological paradigm clustering consists of clustering word forms from raw text into paradigms.

A successful unsupervised paradigm clustering system leverages common patterns in the language's inflectional morphology while simultaneously ignoring regular circumstantial similarities along with derivational patterns. For example, an accurate unsupervised system must recognize that *disfrutamos* (English: *we enjoy*) and *disfruta* (English: *he/she/it enjoys*) are inflected variants of the same paradigm, but that the orthographically similar *disparamos* (English: *we shoot*), belongs to a separate paradigm. Likewise, a successful system for English will recognize that *walk* and *walked* belong to the same verbal paradigm but *walker* is a derived form belonging to a distinct nominal paradigm. Such fine-grained distinctions are difficult to learn in an unsupervised manner.

This paper describes the SIGMORPHON 2021 Shared Task on Unsupervised Morphological Paradigm Clustering. Participants are asked to submit systems which cluster words from the Bible into inflectional paradigms.[1] Participants are not allowed to use any external resources. Four teams submit at least one system for the shared task and

[1]Bible translations for five development and nine test languages were obtained from the Johns Hopkins University Bible Corpus introduced by McCarthy et al. (2020b).

Proceedings of the Seventeenth SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology,pages 72–81

all teams also submit a system description paper.

The shared task systems can be grouped into two broad categories: *similarity-based* systems experiment with different combinations of orthographic and embedding-based similarity metrics for word forms combined with clustering methods like k-means or agglomerative clustering. *Grammar-based* methods instead learn grammars or rules from the data and either apply these to clustering directly, or first segment words into stems and affixes and then cluster forms which share a stem into paradigms. Our official baseline, described in Section 2.3, is based on grouping together word forms sharing a common substring of length $\geq k$, where k is a hyperparameter. Grammar-based systems obtain higher average F1 scores (see Section 2.2 for details on evaluation) across the nine test languages than the baseline. The **Edinburgh** system has the best overall performance: it outperforms the baseline by 34.61% F1 and the second best system by 1.84% F1.

The rest of the paper is organized as follows: Section 2 describes the task of unsupervised morphological paradigm clustering in detail, including the official baseline and all provided datasets. Section 3 gives an overview of the participating systems. Section 4 describes the official results, and 5 presents an analysis. Finally, Section 6 contains a discussion of where the task can move in future iterations and concludes the paper.

2 Task Description

Unsupervised morphological paradigm clustering consists of, given a raw text corpus, grouping words from that corpus into their paradigms without any additional information. Recent work in unsupervised morphology has attempted to induce full paradigms from corpora with only a subset of all types. Kann et al. (2020) and Erdmann et al. (2020) explore initial approaches to this task, which is called unsupervised morphological paradigm *completion*, but find it to be challenging. Building upon the SIGMORPHON 2020 Shared Task on Unsupervised Morphological Paradigm Completion (Kann et al., 2020), our shared task is focused on a subset of the overall problem: sorting words into paradigms. This can be seen as an initial step to paradigm completion, as unobserved types do not need to be induced, and the inflectional categories of paradigm slots do not need to be considered.

2.1 Data

Languages The SIGMORPHON 2021 Shared Task on Unsupervised Morphological Paradigm Clustering features 5 development languages: Maltese, Persian, Portuguese, Russian, and Swedish. The final evaluation is done on 9 test languages: Basque, Bulgarian, English, Finnish, German, Kannada, Navajo, Spanish, and Turkish.

Our languages span 4 writing systems, and represent fusional, agglutinative, templatic, and polysynthetic morphologies. The languages in the development set are mostly suffixing, except for Maltese, which is a templatic language. And while most of the test languages are also predominantly suffixing, Navajo employs prefixes and Basque uses both prefixes and suffixes.

Text Corpora We provide corpora from the Johns Hopkins University Bible Corpus (JHUBC) (McCarthy et al., 2020b) for all development and test languages. This is the only resource that systems are allowed to use.

Gold Partial Paradigms Along with the Bibles, we also release a set of gold partial paradigms for the development languages to be used for system development. Gold data sets are also compiled for the test languages, but these test sets are withheld until the completion of the shared task.

In order to produce gold partial paradigms, we first take the set of all paradigms Π for each language from UniMorph (McCarthy et al., 2020a). We then obtain gold partial paradigms $\Pi_{\hat{G}} = \Pi \bigcap \Sigma$, where Σ is the set of types attested in the Bible corpus. Finally, we sample up to 1000 of the resulting gold partial paradigms for each language, resulting in the set Π_G according to the following steps:

1. Group gold paradigms in $\Pi_{\hat{G}}$ by size, resulting in the set G, where $g_k \in G$ is the group of paradigms with k forms in it.
2. Continually loop over all $g_k \in G$ and randomly sample one paradigm from g_k until we have 1000 paradigms.

Because not every token in the Bible corpora is in UniMorph, we can only evaluate on the subset of paradigms that exist in the UniMorph database. In practice, this means that for several languages, we are not able to sample 1000 paradigms, cf. Tables 1 and 2. Notably, for Basque, we can only provide 12 paradigms.

	Maltese	Persian	Portuguese	Russian	Swedish
# Lines	7761	7931	31167	31102	31168
# Tokens	193257	227584	828861	727630	871707
# Types	16017	11877	31446	46202	25913
TTR	.083	.052	.038	.063	.03
# Paradigms	76	64	1000	1000	1000
# Forms in paradigms	572	446	11430	6216	3596
Largest paradigm size	14	20	47	17	9

Table 1: Statistics for the development Bible corpora and the dev gold partial paradigms. TTR is the type-token ratio in the corpus. The statistics for the paradigms reflect only those words in our partial paradigms, not the full paradigms from Unimorph.

	English	Navajo	Spanish	Finnish	Bulgarian	Basque	Kannada	German	Turkish
# Lines	7728	5058	7337	31087	31101	7958	7863	31102	30182
# Tokens	236465	104631	251581	685699	801657	195459	193213	826119	616418
# Types	7144	18799	9755	54635	37048	18376	28561	22584	59458
TTR	.03	.18	.039	.08	.046	.094	.148	.027	.096
# Paradigms	1000	88	990	1000	1000	12	92	1000	1000
# Forms in paradigms	2475	214	5154	8509	5086	63	933	3628	9204
Largest paradigm size	7	13	34	31	27	25	44	15	49

Table 2: Statistics for the test Bible corpora and the test gold partial paradigms.

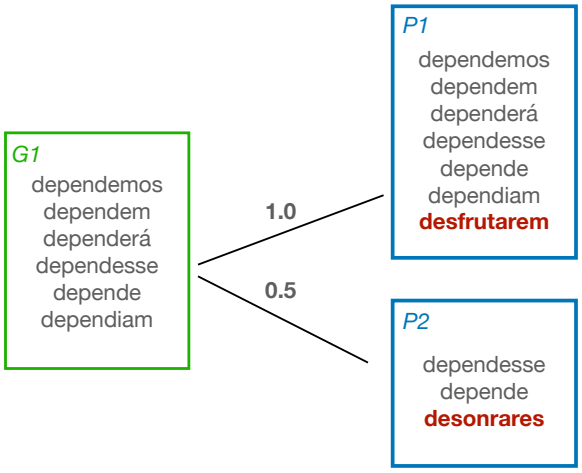

Figure 2: An example matching of predicted paradigms in blue, and a gold paradigm in green. Words in red do not exist in the gold set, and thus cannot be evaluated.

2.2 Evaluation

As our task is entirely unsupervised, evaluation is not straightforward: as in Kann et al. (2020), our evaluation requires a mapping from predicted paradigms to gold paradigms. Because our set of gold partial paradigms does not cover all words in the corpus, in practice we only evaluate against a subset of the clusters predicted by systems.

For these reasons, we want an evaluation that assesses the best matching paradigms, ignoring predicted forms that do not occur in the gold set, but still punishing for spurious predictions that *are* in the gold set. For example, Figure 2 shows two candidate matches for a gold partial paradigm. Each one contains a word that does not exist in the set of gold paradigms, and thus cannot be judged – these words are ignored and do not affect evaluation. In this example, the predicted P1 is a better match, resulting in a perfect F1 score. However, our evaluation punishes systems for predicting a second paradigm, P2, with words from G1, reducing the overall precision score of this submission.

Building upon BMAcc (Jin et al., 2020), we use best-match F1 score for evaluation. We define a paradigm as a set of word forms $f \in \pi$. Duplicate forms within π (*syncretism*) are discarded. Given a set of gold partial paradigms $\pi^g \in \Pi_G$, a set of predicted paradigms $\pi^p \in \Pi_P$, a gold vocabulary $\Sigma^g = \bigcup \pi^g$, and a predicted vocabulary $\Sigma^p = \bigcup \pi^p$, it works according to the following steps:

1. Redefine each predicted paradigm, removing the words that we cannot evaluate $\pi^{p'} = \pi^p \bigcap \Sigma^g$, to form a set of pruned paradigms Π'_P.

2. Build a complete Bipartite graph over Π'_P and Π_G, where the edge weight between π_i^g and $\pi_j^{p'}$ is the number of true positives $|\pi_i^g \bigcap \pi_j^{p'}|$.

3. Compute the maximum-weight full matching using Karp (1980), in order to find the optimal alignment between Π'_P and Π_G

4. Assign all predicted words $\Sigma^{p'}$ and all gold words Σ^g a label corresponding to the gold paradigm, according to the matching found in

3. Any unmatched $w_i^{p'} \in \Sigma^{p'}$ is assigned a label corresponding to a spurious paradigm.

5. Compute the F1 score between the sets of labeled words in $\Sigma^{p'}$ and Σ^{g}

2.3 Baseline System

We provide a straightforward baseline that constructs paradigms based on substring overlap between words. We construct paradigms out of words that share a substring of length $\geq k$. Since words can share multiple substrings, it is possible that multiple identical, redundant paradigms are created. We reduce these to a single paradigm. Words that do not belong to a cluster are assigned a singleton paradigm, that is, a paradigm that consists of only that word.

We tune k on the development sets and find that $k = 5$ works best on average. This means that a word of less than 5 characters can only ever be in one, singleton, paradigm.

3 Submitted Systems

The Boulder-Perkoff-Daniels-Palmer team (**Boulder-PDP**; Perkoff et al., 2021) participates with four submissions, resulting from experiments with two different systems. Both systems apply k-means clustering on vector representations of input words. They differ in the type of vector representations used: either orthographic or semantic representations. Semantic skip-gram representations are generated using word2vec (Mikolov et al., 2013). For the orthographic representations, each word is encoded into a vector of fixed dimensionality equaling the word length $|w_{max}|$ for the longest word w_{max} in the input corpus. They associate each character $c \in \Sigma$ in the alphabet of the input corpus with a real number $r \in [0, 1]$ and assign $v_i := r$ if the ith character of the input word w is c. If $|w| < |w_{max}|$, the remaining entries are assigned to 0.

The number of clusters is a hyperparameter of the k-means clustering algorithm. In order to set this hyperparameter, Perkoff et al. (2021) experiment with a graph-based method. The word types in the corpus form the nodes of a graph, where the neighborhood of a word w consists of all words sharing a maximal substring with w. The graph is split into highly connected subgraphs (HCS) containing n nodes, where the number of edges that need to be cut in order to split the graph into two disconnected components is $> n/2$ (Hartuv and Shamir, 2000). The number of HCSs is then taken to be the cluster number. In practice, however, the graph-clustering step proves to be prohibitively slow and results for test languages are submitted using fixed numbers of clusters of size 500, 1000, 1500 and 1900. In experiments on the dev languages, they find that the orthographic representations outperform the semantic representations for all languages, and thus submit four systems utilizing orthographic representations.

The Boulder-Gerlach-Wiemerslage-Kann team (**Boulder-GWK**; Gerlach et al., 2021) submits two systems based on an unsupervised lemmatization system originally proposed by Rosa and Zabokrtský (2019). Their approach is based on agglomerative hierarchical clustering of word types, where the distance between word types is computed as a combination of a string distance metric and the cosine distance of fastText embeddings (Bojanowski et al., 2017). Their choice of fastText embeddings is due to the limited size of the shared task datasets. Two variants of edit distance are compared to quantify string distance: (1) Jaro-Winkler edit distance (Winkler, 1990) resembles regular edit distance of strings but emphasizes similarity at the start of strings which is likely to bias the system toward languages expressing inflection via suffixation. (2) A weighted variant of edit distance, where costs for insertions, deletions and substitutions are derived from a character-based language model trained on the shared task data.

The **CU–UBC** (Yang et al., 2021) team provides systems that built upon the official shared task baseline – given the pseudo-paradigms found by the baseline, they extract inflection rules of multiple types. Comparing pairs of words in each paradigm, they learn both continuous and discontinuous character sequences that transform the first word into the second, following work on supervised inflectional morphology, such as Durrett and DeNero (2013); Hulden et al. (2014). Rules are sorted by frequency to separate genuine inflectional patterns from noise. Starting from a random seed word, paradigms are constructed by iteratively applying the most frequent rules. Generated paradigms are further tested for paradigm coherence using metrics such as graph degree calculation and fastText embedding similarity.

The **Edinburgh** team (McCurdy et al., 2021) submits a system based on adaptor grammars (John-

		English	Navajo	Spanish	Finnish	Bulgarian	Basque	Kannada	German	Turkish	Average
	Rec	28.93	32.71	23.90	18.43	20.55	28.57	25.19	25.50	15.70	24.39
Boulder-PDP-1	Prec	29.27	34.15	24.68	18.81	20.75	29.51	35.18	25.64	15.90	25.99
	F1	29.10	33.41	24.29	18.62	20.65	29.03	29.36	25.57	15.80	25.09
	Rec	36.57	36.92	28.52	23.38	26.37	30.16	25.83	33.21	19.53	28.94
Boulder-PDP-2	Prec	37.00	38.54	29.45	23.86	26.63	31.15	36.08	33.40	19.79	30.65
	F1	36.78	37.71	28.98	23.62	26.50	30.65	30.11	33.31	19.66	29.70
	Rec	42.79	37.85	29.41	26.01	28.73	26.98	25.94	38.18	21.38	30.81
Boulder-PDP-3	Prec	43.30	39.51	30.37	26.55	29.01	27.87	36.23	38.39	21.66	32.54
	F1	43.04	38.66	29.88	26.27	28.87	27.42	30.23	38.28	21.52	31.58
	Rec	45.45	40.19	30.64	26.60	29.79	28.57	24.54	39.86	21.65	31.92
Boulder-PDP-4	Prec	45.99	41.95	31.63	27.15	30.08	29.51	34.28	40.08	21.93	33.62
	F1	45.72	41.05	31.13	26.87	29.93	29.03	28.61	39.97	21.79	32.68
	Rec	28.81	10.75	19.27	22.02	30.02	19.05	18.54	31.92	20.63	22.33
Boulder-GWK-2	Prec	66.33	65.71	69.93	67.36	71.69	35.29	62.45	78.56	64.09	64.60
	F1	40.17	18.47	30.21	33.19	42.32	24.74	28.60	45.39	31.22	32.70
	Rec	24.53	11.21	18.30	22.69	31.18	25.40	16.93	30.98	21.16	22.49
Boulder-GWK-1	Prec	56.47	68.57	66.41	69.41	74.46	47.06	57.04	76.26	65.74	64.60
	F1	34.20	19.28	28.69	34.20	43.96	32.99	26.12	44.06	32.02	32.83
	Rec	76.69	59.81	72.18	76.73	73.02	25.40	38.48	77.62	65.82	62.86
Baseline	Prec	38.76	23.02	26.56	17.86	26.50	18.60	17.22	25.35	15.60	23.28
	F1	51.49	33.25	38.83	28.97	38.89	21.48	23.79	38.22	25.23	33.35
	Rec	66.95	50.93	60.52	45.96	65.08	17.46	30.33	66.57	43.25	49.67
CU–UBC-5	Prec	90.40	68.55	72.70	56.47	76.85	52.38	61.26	74.40	54.05	67.45
	F1	76.93	58.45	66.05	50.68	70.48	26.19	40.57	70.26	48.05	56.41
	Rec	63.76	51.867	63.62	48.75	63.84	17.46	33.12	65.05	45.81	50.36
CU–UBC-6	Prec	85.99	69.375	76.49	59.67	75.99	52.38	64.24	72.39	57.52	68.23
	F1	73.23	59.36	69.46	53.66	69.39	26.19	43.71	68.52	51.00	57.17
	Rec	60.36	53.74	64.05	51.51	58.18	22.22	35.37	59.32	47.74	50.28
CU–UBC-7	Prec	81.42	72.33	76.98	62.58	69.23	66.67	69.77	66.13	60.17	69.47
	F1	69.33	**61.66**	69.92	56.51	63.23	**33.33**	46.94	62.54	53.24	57.41
	Rec	83.39	47.66	76.48	52.06	73.14	25.40	36.33	74.28	46.50	57.25
CU–UBC-3	Prec	84.38	49.76	78.97	53.14	73.87	26.23	50.75	74.70	47.10	59.88
	F1	83.89	48.69	77.71	52.60	73.50	25.81	42.35	74.49	46.80	58.42
	Rec	80.69	47.66	78.35	57.29	73.77	28.57	40.73	74.06	50.93	59.12
CU–UBC-4	Rec	81.64	49.76	80.89	58.48	74.50	29.51	56.89	74.47	51.59	61.97
	F1	81.16	48.69	79.60	57.88	74.14	29.03	47.47	74.27	51.26	60.39
	Rec	75.96	47.66	75.73	65.35	69.07	28.57	49.52	65.08	60.58	59.73
CU–UBC-1	Prec	76.86	49.76	78.19	66.71	69.92	29.51	69.16	65.44	61.36	62.99
	F1	76.41	48.69	76.94	66.03	69.50	29.03	57.71	65.26	60.97	61.17
	Rec	88.16	41.59	81.90	72.68	76.58	28.57	50.91	73.98	67.37	64.64
CU–UBC-2	Prec	89.21	43.41	84.56	74.18	77.34	29.51	71.11	74.39	68.24	67.99
	F1	88.68	42.48	83.21	**73.42**	76.96	29.03	59.34	74.18	67.80	66.12
	Rec	89.54	41.59	82.38	59.58	80.22	31.75	58.95	78.97	72.82	66.20
Edinburgh	Prec	90.75	43.41	85.06	60.84	83.30	32.79	82.34	79.41	73.75	70.18
	F1	**90.14**	42.48	**83.70**	60.20	**81.73**	32.26	**68.71**	**79.19**	**73.28**	**67.96**
	Rec	95.31	-	85.49	86.21	84.74	65.08	-	79.19	86.80	83.26
stanza	Prec	93.87	-	85.84	85.91	82.79	50.62	-	71.57	86.87	79.64
	F1	94.59	-	85.66	86.06	83.75	56.94	-	75.19	86.84	81.29

Table 3: Results on all test languages for all systems in %; the official shared task metric is best-match F1. To provide a more complete picture, we also show precision and recall. *stanza* is a supervised system.

son et al., 2007) modeling word structure. Their work draws on parallels between the unsupervised paradigm clustering task and unsupervised morphological segmentation. Their grammars segment word forms in the shared task corpora into a sequence of zero or more prefixes and a single stem followed by zero or more suffixes.

Based on the segmented words from the raw text data, they then determine whether the language uses prefixes or suffixes for inflection. The final stem for words in a predominantly suffixing language then consists of the prefixes and stem identified by the adaptor grammar. For a predominantly prefixing language, the final stem instead contains all suffixes of the word form. The team notes that this approach is unsuitable for languages which extensively make use of both prefixes and suffixes, such as Basque.

Finally, they group all words which share the same stem into paradigms. However, because sampling from an adaptor grammar is a non-deterministic process – i.e., the system may return

multiple possible segmentations for a single word form – they construct preliminary clusters by including all forms which might share a given stem. Then they select the cluster that maximizes a score based on frequency of occurrence of the induced segment in all segmentations.

4 Results and Discussion

The official results obtained by all submitted systems on the test sets are shown in Table 3.

The Edinburgh system performs best overall with an average best-match F1 of 67.96%. In general, grammar-based systems attain the best results, with all of the CU–UBC systems and the Edinburgh system outperforming the baseline by at least 23.06% F1. The Boulder-GWK and Boulder-PDP systems, both of which perform clustering over word representations, approach but do not exceed baseline performance. Perkoff et al. (2021) found that clustering over word2vec embeddings performs poorly on the development languages, and their scores on the test set reflect clusters found with vectors based purely on orthography. The Boulder-GWK systems contain incomplete results, and partial evidence suggests that their clustering method, which combines both fastText embeddings trained on the provided bible corpora, and edit distance, *can* indeed outperform the baseline. However, it likely cannot outperform the grammar-based submissions.

For comparison, we also evaluate a supervised lemmatizer from the Stanza toolkit (Qi et al., 2020). The Stanza lemmatizer is a neural network model trained on Universal Dependencies (UD) treebanks (Nivre et al., 2020), which first tags for parts of speech, and then uses these tags to generate lemmas for a given word. Because there is no UD corpus in the current version for Navajo nor Kannada, we do not have scores for those languages. Stanza's accuracy on our task is far lower than that reported for lemmatization on UD data. We note, however, that 1) our data is from a different domain, 2) Biblical language in particular can differ strongly from contemporary text, and 3) we evaluate on only a partial set of types in the corpus, which could represent a particularly challenging set of paradigms for some languages. The Stanza lemmatizer outperforms all systems for all languages, except for German. This is unsurprising as it is a supervised system, though it is interesting that the German score falls short of that of the Edinburgh system.

naaghá	neiikai	naahkai
naashá	nijighá	nideeshaał
naayá	ninádaah	naniná
ninájídaah	nizhdoogaał	

Table 4: A paradigm from our gold set for Navajo.

Overgeneralization/Underspecification When acquiring language, children often overgeneralize morphological analogies to new, ungrammatical forms. For example, the past tense of the English verb *to know* might be expressed as *knowed*, rather than the irregular *knew*. The same behavior can also be observed in learning algorithms at some point during the learning process (Kirov and Cotterell, 2018). This is reflected to some extent in Table 3 by trade-offs between precision and recall. A low precision, but high recall indicates that a system is overgeneralizing: some surface forms are erroneously assigned to too many paradigms. In effect, these systems are hypothesizing that a substring is productive, and thus proposing a paradigmatic relationship between two words. For example, the English words *approach* and *approve* share the stem *appro-* with unproductive segments as suffixes. The baseline tends to overgeneralize due to its creation of large paradigms via a naive grouping of words by shared n-grams.

On the other hand, several systems seem to *underspecify*, indicated by their low recall. A low recall, but high precision indicates that a system does not attribute inflected forms to a paradigm that the form does in fact belong to. This can be caused by suppletion in systems based purely on orthography, for example, generating the paradigm with *go* and *goes*, but attributing *went* to a separate paradigm. Underspecification is apparent in the CU–UBC submissions that relied on discontinuous rules (CU–UBC 5, 6, and 7). This is likely because they filtered these systems down to far fewer rules than their prefix/suffix systems, in order to avoid severe overgeneralization that can result from spurious morphemes based on discontinuous substrings. Similarly, the Boulder-GWK systems both have reasonable precision, but very low recalls. They report that this is due to the fact that they ignore any words with less than a certain frequency in the corpus due to time constraints, thus creating small paradigms and ignoring many words completely.

Language and Typology In general, we find that Basque and Navajo are the two most difficult test languages. Both languages have relatively small

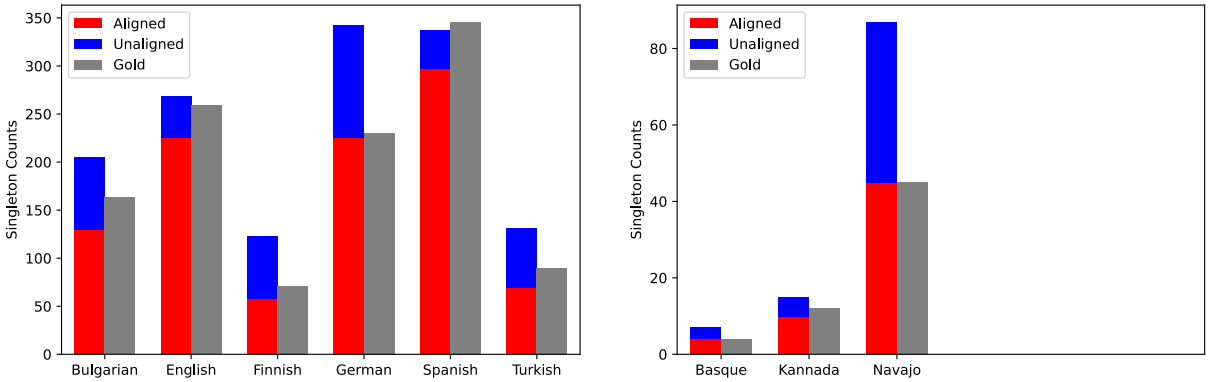

Figure 3: Singleton paradigm counts for the best performing system on all test languages. Languages for which we have more than 100 paradigms on the left, and those for which we have less than 100 paradigms on the right. Predicted singleton paradigms are in red and blue, gold singleton paradigms are in grey.

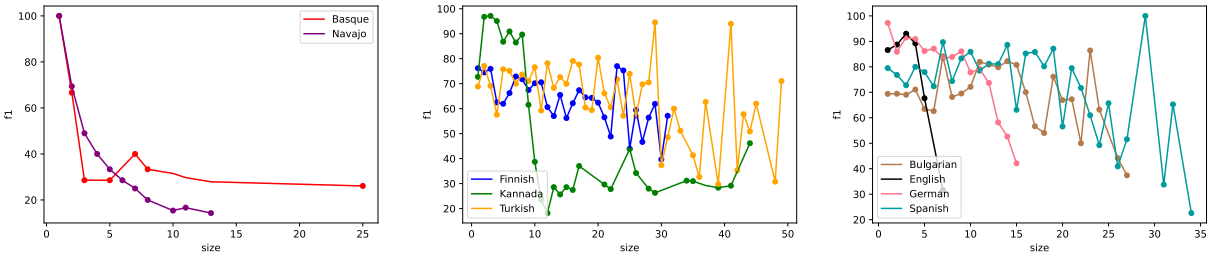

Figure 4: The F1 score across paradigm sizes for the best performing system on all test languages. From left to right, the graphs represent the groups of languages in increasing order of how well systems typically performed on them. F1 scores are interpolated for paradigm sizes that do not exist in a given language.

corpora, and are typologically agglutinative – that is, they express inflection via the concatenation of potentially many morpheme segments, which can result in a large number of unique surface forms. Both languages thus have relatively high type-token ratios (TTR) – especially Navajo, which has the highest TTR, cf. Table 2. It is also important to note that both Basque and Navajo have comparatively small sets of paradigms against which we evaluate. This leaves the possibility that the subset of paradigms in the gold set are particularly challenging. However, the differences between system scores indicates that these two languages do offer challenges related to their morphology.

Navajo is a predominantly prefixing language – the only one in the development and test sets – and Basque also inflects using prefixes, though to a lesser extent. The top two performing systems both obtain low scores for Navajo. The CU–UBC-2 system considers only suffix rules, which results in it being the lowest performing CU–UBC system on Navajo. The Edinburgh submission *should* be able to identify prefixes and consider the suffix to be part of the stem in Navajo. However, the large number of types, for a relatively small Navajo corpus may cause difficulties for their algorithm that builds clusters based on affix frequency. Notably, the CU-UBC-7 system, which learns discontinuous rules rather than rules that model strictly concatenative morphology, performs best on Navajo by a large margin when compared to the best performing system, which relies on strictly concatenative grammars. It also performs best on Basque, though by a smaller margin. Another difficulty in Navajo morphology is that it exhibits verbal stem alternation for expressing mood, tense, and aspect, which creates challenges for systems that rely on rewrite rules or string similarity, based on continuous substrings. For instance, our evaluation algorithm aligns a singleton predicted paradigm to the gold paradigm in Table 4 for nearly all systems.

On Basque, most systems perform poorly. McCurdy et al. (2021), the best performing system overall, obtains a low score for Basque, which may be due to their system assuming that a language inflects either via prefixation or suffixation, but not both, as Basque does. Other systems, however, attain similarly low scores for Basque.

The next tier of difficulty seems to comprise Finnish, Kannada, and Turkish, on which most sys-

tems obtain low scores. All of those languages are suffixing, but also have an agglutinative morphology. The largest paradigm of each of these 3 languages are all in the top 4 largest paradigms in Table 2. This implies that large paradigm sizes and large numbers of distinct inflectional morphemes – two properties often assumed to correlate with agglutinative morphology –, coupled with sparse corpora to learn from, offer challenges for paradigm clustering. Though agglutinative morphology, having relatively unchanged morphemes across words, might be simpler for automatic segmentation systems than morphology characterized as *fusional*, our sparse data sets are likely to complicate this.

Finally, systems obtain the best results for English, followed by Spanish, and then Bulgarian. These three languages are also strongly suffixing, but typically express inflection with a single morpheme. German appears to be a bit of an outlier, generally exhibiting scores that lie somewhere between the highest scoring languages, and the more difficult agglutinative languages. McCurdy et al. (2021) hypothesize that this may be due to non-concatenative morphology from German verbal circumfixes. This hypothesis *could* explain why the Boulder-GWK system performs better on German than other languages: it incorporates semantic information. However, the CU–UBC systems that use discontinuous rules (systems 5, 6, and 7), and thus should better model circumfixation, do not produce higher German scores than the continuous rules, including the suffix-only system.

5 Analysis: Partial Paradigm Sizes

The effect of the size of the gold partial paradigms on F1 score for the best system is illustrated in Figure 4. For Basque and Navajo, the F1 score tends to drop as paradigm size increases. We see the same trend for Finnish, Kannada, and German, with a few exceptions, but this trend does not exist for all languages. English resembles something like a bell shape, other than the low scoring outlier for the largest paradigms of size 7. Interestingly, Spanish and Turkish attain both very high and very low scores for larger paradigms.

An artifact of a sparse corpus is that many singleton paradigms arise. For theoretically larger paradigms, only a single inflected form might occur in such a small corpus. Of course, this also happens naturally for certain word classes. However, nouns, verbs, and occasionally adjectives typically form paradigms comprising several inflected forms. Figure 3 demonstrates that the best system tends to overgenerate singleton paradigms. We see this to some extent for all agglutinative languages, which may be due to the high number of typically long, unique forms. This is especially true for Navajo, which has a small corpus and extremely high type–token ratio. On the other hand, for the languages for which the highest scores are obtained, Spanish and English, the system does not overgenerate singleton paradigms. Of the large number of singleton paradigms predicted for both languages, the vast majority are correct. For other systems not pictured in the figure, singleton paradigms are typically *undergenerated* for Spanish and English. In the case of English, this could be due to words that share a derivational relationship. For example, the word *accomplishment* might be assigned to the paradigm for the verb *accomplish*, when, in fact, their relationship is not inflectional.

6 Conclusion and Future Shared Tasks

We presented the SIGMORPHON 2021 Shared Task on Unsupervised Morphological Paradigm Clustering. Submissions roughly fell into two categories: similarity-based methods and grammar-based methods, with the latter proving more successful at the task of clustering inflectional paradigms. The best systems significantly improved over the provided n-gram baseline, roughly doubling the F1 score – mostly through much improved precision. A comparison against a supervised lemmatizer demonstrated that we have not yet reached the ceiling for paradigm clustering: many words are still either incorrectly left in singleton paradigms or incorrectly clustered with circumstantially (and often derivationally) related words. Regardless of the ground still to be covered, the submitted results were a successful first step in automatically inducing the morphology of a language without access to expert-annotated data.

Unsupervised morphological paradigm clustering is only the first step in a morphological learning process that more closely models human L1 acquisition. We envision future tasks expanding on this task to include other important aspects of morphological acquisition. Paradigm slot categorization is a natural next step. To correctly categorize paradigm slots, cross-paradigmatic similarities must be considered, for example, the German words *liest* and *schreibt* are both 3$^{\text{rd}}$ person singular

present indicative inflections of two different verbs. This can occasionally be identified via string similarity, but more often requires syntactic information. Syncretism (the collapsing of multiple paradigm slots into a single representation) further complicates the task. A similar subtask involves lemma identification, where a canonical form (Cotterell et al., 2016b) is identified within the paradigm.

Likewise, another important task involves filling unrealized slots in paradigms by generating the correct surface form, which can be approached similarly to previous SIGMORPHON shared tasks on inflection (Cotterell et al., 2016a, 2017, 2018; McCarthy et al., 2019; Vylomova et al., 2020), but will likely be based on noisy information from the slot categorization – all previous tasks have assumed that the morphosyntactic information provided to an inflector is correct. Currently, investigations into the robustness of these systems to noise are sparse.

Another direction for this task is the expansion to more under-resourced languages. The submitted results demonstrate that the task becomes particularly difficult when the provided raw text is small, but under-documented languages are often the ones most in need of morphological corpora. The JHUBC contains Bible data for more than 1500 languages, which can potentially be augmented by other raw text corpora because morphology is relatively stable across domains. Future tasks may enable the construction of inflectional paradigms in languages that require them to construct further computational tools.

Acknowledgments

We would like to thank all of our shared task participants for their hard work on this difficult task!

References

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.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Arya D McCarthy, Katharina Kann, Sabrina J Mielke, Garrett Nicolai, Miikka Silfverberg, et al. 2018. The conll–sigmorphon 2018 shared task: Universal morphological reinflection. *arXiv preprint arXiv:1810.07125*.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sandra Kübler, David Yarowsky, Jason Eisner, et al. 2017. Conll-sigmorphon 2017 shared task: Universal morphological reinflection in 52 languages. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 1–30.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016a. 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, Tim Vieira, and Hinrich Schütze. 2016b. A joint model of orthography and morphological segmentation. 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.

Alexander Erdmann, Micha Elsner, Shijie Wu, Ryan Cotterell, and Nizar Habash. 2020. The paradigm discovery problem. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7778–7790, Online. Association for Computational Linguistics.

Andrew Gerlach, Adam Wiemerslage, and Katharina Kann. 2021. Paradigm clustering with weighted edit distance. In *Proceedings of the 18th Workshop on Computational Research in Phonetics, Phonology, and Morphology*. Association for Computational Linguistics.

Erez Hartuv and Ron Shamir. 2000. A clustering algorithm based on graph connectivity. *Information processing letters*, 76(4-6):175–181.

Mans Hulden, Markus Forsberg, and Malin Ahlberg. 2014. Semi-supervised learning of morphological paradigms and lexicons. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 569–578.

Huiming Jin, Liwei Cai, Yihui Peng, Chen Xia, Arya McCarthy, and Katharina Kann. 2020. Unsupervised morphological paradigm completion. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6696–6707, Online. Association for Computational Linguistics.

Mark Johnson, Thomas L Griffiths, Sharon Goldwater, et al. 2007. Adaptor grammars: A framework for specifying compositional nonparametric bayesian models. *Advances in neural information processing systems*, 19:641.

Katharina Kann, Arya D. McCarthy, Garrett Nicolai, and Mans Hulden. 2020. The SIGMORPHON 2020 shared task on unsupervised morphological paradigm completion. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 51–62, Online. Association for Computational Linguistics.

Richard M Karp. 1980. An algorithm to solve the m× n assignment problem in expected time o (mn log n). *Networks*, 10(2):143–152.

Christo Kirov and Ryan Cotterell. 2018. Recurrent neural networks in linguistic theory: Revisiting pinker and prince (1988) and the past tense debate. *Transactions of the Association for Computational Linguistics*, 6:651–665.

Arya D. McCarthy, Christo Kirov, Matteo Grella, Amrit Nidhi, Patrick Xia, Kyle Gorman, Ekaterina Vylomova, Sabrina J. Mielke, Garrett Nicolai, Miikka Silfverberg, Timofey Arkhangelskiy, Nataly Krizhanovsky, Andrew Krizhanovsky, Elena Klyachko, Alexey Sorokin, John Mansfield, Valts Ernštreits, Yuval Pinter, Cassandra L. Jacobs, Ryan Cotterell, Mans Hulden, and David Yarowsky. 2020a. UniMorph 3.0: Universal Morphology. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3922–3931, Marseille, France. European Language Resources Association.

Arya D. McCarthy, Ekaterina Vylomova, Shijie Wu, Chaitanya Malaviya, Lawrence Wolf-Sonkin, Garrett Nicolai, Christo Kirov, Miikka Silfverberg, Sabrina J. Mielke, Jeffrey Heinz, Ryan Cotterell, and Mans Hulden. 2019. The SIGMORPHON 2019 shared task: Morphological analysis in context and cross-lingual transfer for inflection. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 229–244, Florence, Italy. Association for Computational Linguistics.

Arya D. McCarthy, Rachel Wicks, Dylan Lewis, Aaron Mueller, Winston Wu, Oliver Adams, Garrett Nicolai, Matt Post, and David Yarowsky. 2020b. The Johns Hopkins University Bible corpus: 1600+ tongues for typological exploration. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2884–2892, Marseille, France. European Language Resources Association.

Kate McCurdy, Sharon Goldwater, and Adam Lopez. 2021. Adaptor grammars for unsupervised paradigm clustering. In *Proceedings of the 18th Workshop on Computational Research in Phonetics, Phonology, and Morphology*. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

E. Margaret Perkoff, Josh Daniels, and Alexis Palmer. 2021. Orthographic vs. semantic representations for unsupervised morphological paradigm clustering. In *Proceedings of the 18th Workshop on Computational Research in Phonetics, Phonology, and Morphology*. Association for Computational Linguistics.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics.

Rudolf Rosa and Zdenek Zabokrtský. 2019. Unsupervised lemmatization as embeddings-based word clustering. *CoRR*, abs/1908.08528.

Ekaterina Vylomova, Jennifer White, Elizabeth Salesky, Sabrina J Mielke, Shijie Wu, Edoardo Ponti, Rowan Hall Maudslay, Ran Zmigrod, Josef Valvoda, Svetlana Toldova, et al. 2020. Sigmorphon 2020 shared task 0: Typologically diverse morphological inflection. *arXiv preprint arXiv:2006.11572*.

William E. Winkler. 1990. String comparator metrics and enhanced decision rules in the fellegi-sunter model of record linkage. In *Proceedings of the Section on Survey Research*, pages 354–359.

Changbing Yang, Garrett Nicolai, and Miikka Silfverberg. 2021. Unsupervised paradigm clustering using transformation rules. In *Proceedings of the 18th Workshop on Computational Research in Phonetics, Phonology, and Morphology*. Association for Computational Linguistics.