A Computational Model for the Linguistic Notion of Morphological Paradigm

Miikka Silfverberg and **Ling Liu** and **Mans Hulden**
Department of Linguistics
University of Colorado
first.last@colorado.edu

Abstract

In supervised learning of morphological patterns, the strategy of generalizing inflectional tables into more abstract paradigms through alignment of the longest common subsequence found in an inflection table has been proposed as an efficient method to deduce the inflectional behavior of unseen word forms. In this paper, we extend this notion of morphological 'paradigm' from earlier work and provide a formalization that more accurately matches linguist intuitions about what an inflectional paradigm is. Additionally, we propose and evaluate a mechanism for learning full human-readable paradigm specifications from incomplete data—a scenario when we only have access to a few inflected forms for each lexeme, and want to reconstruct the missing inflections as well as generalize and group the witnessed patterns into a model of more abstract paradigmatic behavior of lexemes.

1 Introduction

Most work in phonology and morphology assumes that the ability to inflect previously unseen or unheard word forms is through a model of analogy (Blevins and Blevins, 2009) to known forms. In languages with poor inflectional morphology, this often manifests itself in the capacity to inflect previously unwitnessed lexemes, e.g. **wug** $\mapsto$ **wugs** by analogy to some other lexeme, perhaps a noun like **bug** $\mapsto$ **bugs**. A closely related problem, relevant for languages that have complex inflectional morphology and where a single part-of-speech can be inflected in perhaps thousands of ways, is the capacity to inflect a lexeme with a previously unheard combination of inflectional features. For example, Finnish nouns are inflected in 2,253 different ways,[1] and no speaker will have witnessed all variants for any given noun. This capacity to fill in missing forms has been dubbed the **paradigm cell filling problem** (PCFP) (Ackerman et al., 2009).

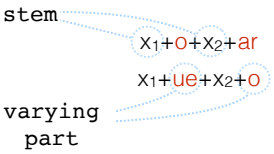
The frequency of references to the nebulous label 'analogy' and its responsibility for human performance in these tasks stands in contrast to the dearth of proposed exact formal or computational models of analogical word-formation in the literature. Even in second-language teaching material, speakers are assumed to possess the capacity to directly generalize a pattern from a collection of inflected forms. Table 1 shows a partial inflection table for the Spanish verb **tostar** ('to toast') given in the resource *501 Spanish Verbs* (Kendris and Kendris, 2007). By contrast, the verb entry for **colar** ('to strain') contains no table at all, but simply provides a pointer to the inflection table for **tostar**. In other words, it is presupposed that, for example, given the analogy **tostar:tuesto::colar:x**, the astute reader can solve for **x** without undue problems, though no explicit procedure for doing so is given.[2]

One formalization of this implicit procedure is offered in Ahlberg et al. (2014) and Hulden (2014). In those works, a collection of inflected forms is first subjected to identifying the *longest common subsequence* (LCS) among the given forms. For example, the set of words {**tostar, tuesto, . . .** } would contain

[1] http://www.ling.helsinki.fi/~fkarlsso/genkau2.html

[2] Note that to perform this 'analogy', it is not sufficient to merely identify a monolithic stem and possible suffix in the pair **tostar/tuesto** and add the discovered suffix to the presupposed stem of **colar**. This strategy works for simple cases like **bug:bugs::wug:wugs**, but obviously cannot be the basis for general analogical modeling in morphology.

Proceedings of the 27th International Conference on Computational Linguistics, pages 1615–1626
Santa Fe, New Mexico, USA, August 20-26, 2018.



(a) Stem and varying part extracted for word forms **tostar** and **tuesto**.

			P1	P2
NOM	koira	mäyrä	x_1	x_1
ADE	koiralla	mäyrällä	x_1+**ll**a	x_1+**ll**ä
ELA	koirasta	mäyrästä	x_1+**st**a	x_1+**st**ä

(b) Paradigms extracted by LCS do not correspond to linguistic generalizations because they do not factor out regular phonological alternations.

Figure 1: Illustration of the LCS method for paradigm extraction in 1a. In 1b, we show the inability of the LCS method to factor out vowel harmony.

the LCS **tst**, as that subsequence occurs in every form. After locating this, the generalization is set up so that the LCS part is defined as a discontinuous *stem* that recurs throughout the inflectional paradigm, while the remainder can vary. In the above example, this would yield the generalization in Figure 1a where the $\mathbf{x_i}$s represent (in this case) the discontinuous 'stem'. Fitting another word such as **colar** into the first form would implicitly yield that the stem of that word is $\mathbf{x_1} = \mathbf{c}$, $\mathbf{x_2} = \mathbf{l}$. Having identified the stem, we can now produce the form $\mathbf{x_1}$ + **ue** + $\mathbf{x_2}$ + **o** which becomes **cuelo**, concluding the analogy as **tostar:tuesto::colar:cuelo**.

This strategy works reasonably well for supervised learning of morphological paradigms and provides a human-readable generalization as a result (Ahlberg et al., 2014; Ahlberg et al., 2015; Forsberg and Hulden, 2016). However, we identify two weaknesses in the model. First, the approach undergeneralizes in that it produces separate generalizations of paradigms for what is arguably very similar inflectional behavior. Secondly, it does not offer a method to learn paradigms from partial data—it operates on full inflectional tables—something that would address the paradigm cell filling problem.

To illustrate the first point, consider the two Finnish inflection tables and resulting paradigms in Figure 1b. Here, the only difference between the two learned patterns is that whenever one (P1) has an **a**-segment in its slots, the other (P2) has **ä**. This difference prevents the two patterns from being deemed alike. Naturally, a more thorough linguistic analysis would uncover that this is an instance of back/front vowel harmony dictated by the vowels found in the stem, the sequences **lla/llä** and **sta/stä** in effect being allomorphs of the same morpheme.

In this paper we propose a method that will discover such allomorphy through a purely data-driven approach—i.e. it will not be necessary to postulate any phonological evidence or analysis to be able to discover that several paradigms learned through the LCS method are actually alike and to deduce all the allomorphs. This, as we will show, will substantially cut down on the number of different inflectional paradigms discovered by the algorithm to something very close to that postulated by linguists after thorough phonological and morphological analysis.

Further, we propose a method on top of the LCS-paradigm generalization to discover partial paradigms from partially given inflection tables and to collapse these into full-size paradigms that allow us to address the paradigm cell filling problem.

tostar (474)		**colar**
tuesto	tostamos	*[inflect like 474]*
tuestas	tostáis	
tuesta	tuestan	
...	...	

Table 1: Information given in Kendris & Kendris (2007) about Spanish verb inflection. Note the reference from **colar** to **tostar**.

1.1 Terminology

To avoid terminological confusion, we will adhere to some conventions in this paper. First, we make a distinction between an **inflection table** and **paradigm**: an **inflection table** is a concrete manifestation of forms such as the two in the left hand box in Figure 1b. By contrast, a **paradigm** is the result of generalizing into stems (the $\mathbf{x_i}$s) and affixes (the remainder), such as the two paradigms in the right-hand box. This is in line with the idea that a mere collection of inflected forms is not a generalization per se, and we therefore reserve the term **paradigm** to the generalized version of an inflection table.[3] A single **slot** in an inflection table which contains a **form** (such as 'koiralla' in Figure 1b) will be generalized into

[3]Some authors call the generalization an **inflectional class** (Haspelmath and Sims, 2013).

a **form pattern**, such as $\mathbf{x_i}$**+lla**. The features corresponding to a form, such as NOM;SG will be referred to as the **morphosyntactic description** (MSD).

2 Related Work

As noted above, we build on the work of Ahlberg et al. (2014), Ahlberg et al. (2015), and Forsberg and Hulden (2016) which introduced the LCS as a core strategy in formal definitions of morphological paradigm and analogy. Further linguistic arguments favoring the LCS-model are given in Lee (2015).

Learning to generalize from inflection tables to unseen forms has attracted much recent interest in natural language processing (Dreyer and Eisner, 2011; Durrett and DeNero, 2013). In addition, two recent shared tasks hosted by SIGMORPHON and CoNLL have contributed to the research by producing comparable large multilingual data sets (Cotterell et al., 2016; Cotterell et al., 2017) consistently annotated with the Unimorph scheme (Sylak-Glassman et al., 2015; Kirov et al., 2018). Overwhelmingly, most recent work in inflection generation has employed neural sequence-to-sequence models with attention (Sutskever et al., 2014; Bahdanau et al., 2015) in supervised scenarios (Kann and Schütze, 2016; Faruqui et al., 2016; Östling, 2016; Makarov et al., 2017; Aharoni and Goldberg, 2017), often with data augmentation mechanisms in low-resource settings (Bergmanis et al., 2017; Silfverberg et al., 2017; Kann and Schütze, 2017). The PCFP has been explicitly addressed by recurrent neural generators as well (Malouf, 2016; Malouf, 2017). While neural models perform quite well on such tasks—even in low-resource settings—their parameter opacity makes it difficult to extract concise linguistic generalizations that can be interpreted and compared with linguist analyses.

Since morphological annotation is usually done at the word-level (**flowers** $\leftrightarrow$ ❀**+PL**),[4] recent work has also attempted to learn the latent segmentation and recover the different allomorphs for stems and affixes (**flower** = ❀, **s=PL**). This is arguably similar to what an L1-learner does, and the kind of evidence L1 learners have—words with semantic (❀) and grammatical (**+PL**) content deducible from context, but crucially missing information about how these correspond to some subsequence of phonemes in an inflected word form. Silfverberg and Hulden (2017a) and Silfverberg and Hulden (2017b) propose a data-driven method that searches the space of possible allomorph and grammatical tag associations, and favors a small model. Similar approaches are found in Hayes (2018) who similarly argues that allomorphs can be detected "even if we don't yet understand the phonology". Our work, apart from providing an explicit model for paradigms, also implicitly extracts the different allomorphs used, something that falls out as a byproduct of the paradigm generalizations in section 4 and inference from partial inflection tables in section 5.

3 Data

We use two different data sets for experiments. The first one is the training and development parts from the data set developed by Durrett and DeNero (2013) for inflection generation, including Finnish, German and Spanish.[5] The data set was scraped from the English Wiktionary. It is used both for experiments on paradigm generalization and learning of full paradigms from partial paradigms. We describe the data set in Table 2.

The second data set comes from the CoNLL-SIGMORPHON 2017 joint shared task on morphological reinflection (Cotterell et al., 2017). We use the paradigm completion subtask (subtask 2) data. We use this data exclusively for the paradigm generalization task.[6] The CoNLL-SIGMORPHON shared task data set contains full morphological inflection tables. We use the task's train-dev-test splits. Like the data set by Durrett and DeNero (2013), the CoNLL-SIGMORPHON data set is also scraped from Wiktionary. It contains data from 52 languages representing a wide variety of language families. However, we perform experiments on a selection of 33 languages out of these 52 languages because of the variance in inflection table size for some languages as explained below.[7]

[4] An illustration of a semantic representation.

[5] `http://www.cs.utexas.edu/~gdurrett/`

[6] We do not use CoNLL-SIGMORPHON data for paradigm learning because of the limited size of the data set.

[7] These 33 languages are Albanian, Bengali, Catalan, Danish, English, Estonian, Faroese, Finnish, French, Georgian, Ger-

The paradigm extraction method developed by Ahlberg et al. (2014), as well as the paradigm generalization method to be proposed in section 4, operates on full paradigms, requiring that all of the inflection tables for a given part-of-speech have the same number of MSDs. However, this is not the case in the CoNLL-SIGMORPHON shared task data: inflection tables of German nouns, for instance, encompass between four and eight forms. There are several reasons for this discrepancy. For example, some nouns may lack plural or singular forms altogether (as an example, consider the English *plurale tantum* noun **pants**). We, therefore, first determine the maximal number of MSDs for inflection tables of each part-of-speech in each language and perform experiments only on those inflection tables where all MSDs are present. This is not possible for all languages because many do not contain a single table which would give forms corresponding to the full set of MSDs. Consequently, we limit our experiments to those languages where we can successfully find lexemes with a maximal number of MSDs.

4 Generalized Paradigms

We now turn to the method for making further generalizations over the basic paradigm model given in Ahlberg et al. (2014). This method also extracts sets of allomorphs.

As mentioned in section 1, the paradigm extraction procedure presented in Ahlberg et al. (2014) cannot group inflection tables under the same paradigm unless the non-stem parts of corresponding forms in the tables are identical. For example, the two Finnish stems **lauk** and **hyp** in Figure 2 cannot belong to the same paradigm because back vowels (**a**) and front vowels (**ä**) vary across the two paradigms, though the form patterns are similar in other respects. This does not adhere to standard linguistic assumption whereby regular phonological alternations—such as vowel harmony in this case—should not affect paradigm membership.

Our method for paradigm generalization groups together paradigms which exhibit such regular alternations, under the same *generalized paradigm*. The method is based purely on a formal comparison of the paradigms, not linguistic considerations; it is in effect 'phonology-free'. We discover regular alternations using a variable substitution described below, and group multiple paradigms under the same *generalized paradigm* whenever they exhibit regular alternation without regard to the identity of the variants.

4.1 Method

The basis for paradigm generalization is identification of regular alternations between paradigms. Consider the example in Figure 2. Both of the nominal lexemes **laukku** ('bag') and **hyppy** ('jump') undergo consonant gradation. In what is called the 'strong grade', the last stem consonant is doubled both in the nominative (NOM) and partitive (PART) cases. The alternation between the paradigms is regular in the sense that whenever the form for **laukku** has an additional **k**, **hyppy** will have an additional **p**. We abstract out the differences between variants by replacing the varying parts in form patterns with consecutively numbered new variables $\mathbf{y_i}$, where multiple occurrences of the same varying part all get the same variable $\mathbf{y_i}$. For the current case, the **k** in the paradigm for **laukku** and the **p** in the paradigm extracted from **hyppy** in the strong grade are replaced by $\mathbf{y_1}$ both in the nominative and partitive forms, **lla** and **llä** are both replaced with $\mathbf{y_2}$, **sta** and **stä** are both replaced with $\mathbf{y_3}$, and **a** and **ä** are both replaced with $\mathbf{y_4}$. Applying this second-order generalization procedure on top of the LCS-generalization results in identical generalized paradigm structure for **laukku** and **hyppy**, seen on the right in Figure 2.

4.2 Experiments and Results

As mentioned in Section 3 we use two data sets for experiments: the data set from Durrett and DeNero (2013), and part of the data (33 languages) for the second subtask of the CoNLL-SIGMORPHON 2017 shared task on universal morphological reinflection. The number of paradigms and generalized paradigms in the Durrett and DeNero (2013) data set is shown in Table 2 in the last two columns. The number of generalized paradigms agrees much better with linguist analyses regarding the number of

man, Hebrew, Hindi, Hungarian, Icelandic, Irish, Italian, Latin, Latvian, Lithuanian, Lower Sorbian, Northern Sami, Persian, Portuguese, Romanian, Slovak, Slovene, Spanish, Swedish, Turkish, Urdu, and Welsh, which are from the 11 different language groups (based on the original data source grouping given by Cotterell et al. (2017)) including Romance, Germanic, Indo-Aryan, Uralic, Kartvelian, Semitic, Celtic, Baltic, Iranian, Slavic, and Turkic

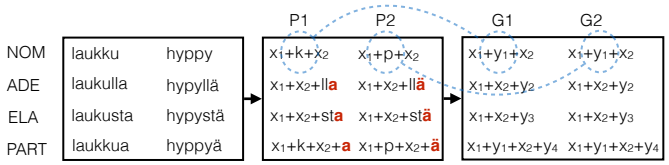


Figure 2: Forming a generalized paradigm (right) based on a paradigm discovered by the LCS method (Ahlberg et al., 2014) (middle) from raw inflection tables (left).

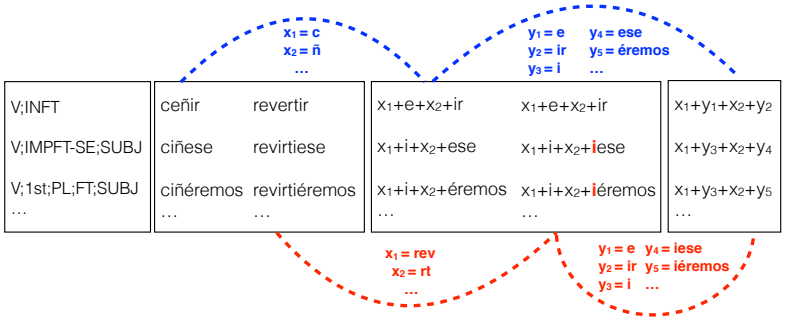


Figure 3: Illustration of a latent phonological generalization in Spanish.

types of inflection tables within languages. For example, Nykysuomen sanalista,[8] a lexical resource developed by the Institute for the Languages of Finland, recognizes 51 nominal inflection types and 27 verb inflection types for Finnish. The number of generalized paradigms for Finnish nouns and adjectives[9] produced by the current method is 40 for nouns and 30 for verbs. The average ratio between the number of generalized paradigms and the number of original LCS-paradigms across languages are 39.07% for verbs, 26.29% for nouns and adjectives, and 35.02% for all parts-of-speech together, showing a significant reduction in number of disctinct patterns identified.

In addition, generalized paradigms capture linguistically regular alternations. Figure 3 illustrates how the generalized paradigm captures a phenomenon surrounding the grapheme **ñ** in Spanish. Verbs that don't contain stem-final **ñ** [ɲ] insert an **i** in some forms, while verbs with **ñ** forgo the **i** as the **ñ** itself can be interpreted as a coalesced form that already contains an **i**, i.e. [ni] ↦ [ɲ]. Spanish verbs like **ceñir** ('to cling to') and verbs like **revertir** ('to revert to') would fall into two separate paradigms using the paradigm extraction method proposed in Ahlberg et al. (2014). These two LCS-paradigms are provided in the second box from the right in the figure. The dotted blue curve on top shows the first-order generalization provided by the LCS-strategy that assigns $\mathbf{x_i}$-variables, and our subsequent second-order abstraction that assigns $\mathbf{y_i}$-variables. After these two steps, the behavior of the the two verbs is seen to be identical in the new representation shown in the rightmost box. Other words which belong to the same LCS-paradigm as **ceñir** that all have **ñ** as their second variable include words like **reñir** ('to scold'), **constreñir** ('to restrict'), **teñir** ('to stain'). The second variable for words which get the same paradigm as **revertir** can be **rt**, **r**, **nt** etc. Words in this paradigm include **convertir** ('to convert'), **ingerir** ('to ingest'), **sentir** ('to feel') etc. The generalized paradigm groups the two paradigms together because the two paradigms become identical after $\mathbf{y_i}$-replacement following the method described in section 4.1. The $\mathbf{y_i}$s from the first paradigm are given along the blue dotted curve connecting the paradigm with the generalized paradigm, and the $\mathbf{y_i}$s from the second paradigm are added to the red dotted curve at the bottom combining the second paradigm with the generalized paradigm. By grouping the two paradigms together as one generalized paradigm, the alternation between **ñ** and any other consonants followed by **i** emerges.

With the second data set of 33 languages, we combine all the training, development and test sets together as the data set is small (less than 300 inflection tables altogether) for each language. For all the

[8] `http://kaino.kotus.fi/sanat/nykysuomi`

[9] Finnish nouns and adjectives are inflected in the same way in the data set.

languages we experiment with, the number of paradigms decreases consistently after generalization: the average ratio between the number of generalized paradigms and the number of LCS-paradigms across languages on this data set is 0.69 for verbs, 0.55 for nouns and adjectives, and 0.59 for all parts-of-speech together. The ratio is higher on this data set than on the Durrett and DeNero (2013) data set because the data size here is much smaller, and yields fewer paradigms in general.

LANGUAGE	POS	TABLE COUNT	TABLE SIZE	PARADIGM COUNT	GENERALIZED PARADIGM COUNT
FINNISH	NOUN & ADJ	6,400	28	259	40
FINNISH	VERB	7,249	53	276	30
GERMAN	NOUN	2,764	8	70	26
GERMAN	VERB	2,027	27	135	107
SPANISH	VERB	4,055	57	96	26

Table 2: Data sets by Durrett and DeNero (2013). We show counts of inflection tables and inflection table sizes for nouns, adjectives and verbs. Additionally, we show the number of paradigms found the LCS method (Ahlberg et al., 2014) and the resulting number of 'generalized paradigms' found by subsequently applying our method.

An example of how generalization over paradigms collapses them and catches regularities from the CoNLL-SIGMORPHON data set can be found by examining English verbal inflections. Different paradigms are extracted from verbs like **contest**, **establish**, **benefit**, **occur** and **bag**, though the paradigm for **establish** differs from that for **contest** only in the present tense for the third-person singular, with the last variable as **-es** for **establish**, but **-s** for **contest**; the paradigms for **benefit**, **occur** and **bag** differ from that of **contest** in the present participle, past participle and past tense, for which the last variables for **benefit**, **occur** and **bag** are **-ting**, **-ring** and **-ging** (for the present participle) and **-ted**, **-red** and **-ged** (for the past participle and past tense) respectively, but **-ing** and **-ed** for **contest**. The generalized paradigm approach groups all these patterns together, which completely agrees with the linguistic regularity.

5 Learning Full Paradigms from Partial Paradigms

In this section we present an algorithm for learning full morphological paradigms given partial inflection tables. Our algorithm takes as input a set of partial inflection tables, that is, inflection tables with a number of missing forms. It first derives a morphological LCS-paradigm for each partial inflection table and then imputes missing form patterns into the paradigm. At test time, form patterns are substituted by concrete word forms (see Figure 4a). A key component of the proposed algorithm is a stem sampling step which counteracts bad hypotheses about stems caused by missing information when performing paradigm extraction on only partial inflection tables.

V;INF	speak		V;INF	speak
V;PRES	?		V;PRES	speaks
V;PAST	?	$\Longrightarrow$	V;PAST	spoke
V;PAST;PCPLE	spoken		V;PAST;PCPLE	spoken
V;PRES;PCPLE	?		V;PRES;PCPLE	speaking

(a) Our inputs are partial inflection tables and the final outputs are completed inflection tables.

V;INF	-		V;INF	-
V;PRES	-		V;PRES	-
V;PAST	-	$\Longrightarrow$	V;PAST	-
V;PAST;PCPLE	eaten		V;PAST;PCPLE	$\mathbf{x_1{+}e{+}x_2}$
V;PRES;PCPLE	eating		V;PRES;PCPLE	$\mathbf{x_1{+}i{+}x_2{+}g}$

(b) Paradigm extraction is applied to a partial inflection table. The stem is incorrectly identified as **eatn**. This makes it impossible to correctly infer the infinitive **eat** and past tense **ate**.

Figure 4: We illustrate the paradigm cell filling task in 4a and show an example of over-generalization effects due to data sparsity in 4b.

We evaluate the presented algorithm with regard to its ability to reconstruct missing forms in partial paradigms and compare it against two baselines: majority voting and matrix completion. Our experiments demonstrate that the proposed algorithm delivers clear improvements over the baselines.

5.1 Methods

As mentioned above, our input consists of partial inflection tables. The first step is to apply the LCS procedure for paradigm extraction (Ahlberg et al., 2014) presented in Section 1. This results in a partial

paradigm as shown in Figure 4. We now present four different methods for imputing the missing forms into the resulting partial paradigm.

First Baseline using Majority Voting For each partial paradigm P, we apply a simple variant of majority voting in order to infer missing form patterns. We form the set of all partial paradigms which have a form pattern corresponding to a missing MSD, m, and perform a majority vote among the stem patterns. A slight variant of standard majority voting is, however, required because we do not want to infer a candidate with an incorrect number of stem variables. For example, if the PAST and PRES;PCPLE forms in P are $\mathbf{x_1}$**+ed** and $\mathbf{x_1}$**+ing** respectively, we do not want to infer a form $\mathbf{x_1}$**+**$\mathbf{x_2}$**+n** for PAST;PCPLE because the stem corresponding to P has only one variable $\mathbf{x_1}$. Therefore, we restrict the vote to paradigms which have the same number of variables ($\mathbf{x_i}$s) as P.

We want to further restrict the set of candidates, because whenever two paradigms P and P' disagree on a known form, it is likely that they will disagree on missing forms as well. For example, if the PAST form in P is $\mathbf{x_1}$**+ed** and the PAST form for P' is $\mathbf{x_1}$**+n**, that is, the paradigms disagree on the PAST MSD, then the PAST;PCPLE forms are unlikely to agree either. Therefore, we further restrict voting to those candidate paradigms P', where all the shared MSDs with P correspond to identical form patterns. For each MSD m, we now take a majority vote in the restricted set of candidate paradigms.

Priority Voting As a small additional restriction to the majority voting process, we consider restricting the set of candidates even further. Whereas we include candidates with no overlap, i.e. common forms, in majority voting, we now exclude such paradigms from voting. That is, only include candidates which have some common MSDs with the target paradigm P. These are, after all, more likely to agree other forms as well. This set may, however, be empty. We, therefore, resort to applying majority voting as fall-back.

Stem Sampling We apply the paradigm extraction procedure outlined in Section 1 to partial inflection tables. Unfortunately, this frequently leads to incorrectly identified stems as shown in Figure 4b. This happens when all of the given forms (for example, **eating** and **eaten**) share a substring (**n**) which, nevertheless, is a part of the inflectional affixes and not the actual word stem. The common substring then becomes incorrectly incorporated in the word stem (**eat**, **n**), because the stem is identified greedily as the longest common subsequence of all of the forms in the partial inflection table. This gives rise to implausible form patterns ($\mathbf{x_1}$**+i+**$\mathbf{x_2}$**+g** and $\mathbf{x_1}$**+e+**$\mathbf{x_2}$, from **eat**<u>**i**</u>**ng**, **eat**<u>**e**</u>**n**). To counteract this problem, we perform a sampling step before imputing missing forms.

By examining all partial paradigms extracted from partial inflection tables at once, we can make a better guess concerning the identity of the stems. It is a reasonable assumption that an MSD such as PAST;PCPLE have few distinct realizations. In English, most words take a past participle ending of either **-n** or **-ed**. Although there are examples of other PAST;PCPLE markers, the majority of words belong to one of these classes. Therefore, we would expect most form patterns in the past perfect slot to look similar to $\mathbf{x_1}$**+ed** or $\mathbf{x_1}$**+**$\mathbf{x_2}$**+n**. However, incorrectly identified stems will frequently result in implausible variants like $\mathbf{x_1}$**+**$\mathbf{x_2}$, $\mathbf{x_1}$**+d** or $\mathbf{x_1}$**+e+**$\mathbf{x_2}$. We can expect to find better stem candidates if we attempt to find a set of stems which minimize the number of realizations for each MSD over the whole data set, i.e. all partial paradigms.

The general plan of attack is to sample stems in a way that favors fewer realizations for each MSD. Let the partial inflection table be as in Figure 4b. The longest common subsequence between all of the given forms is **eatn**, which leads us to postulate it as the stem. We also get forms $\mathbf{x_1}$**+e+**$\mathbf{x_2}$ and $\mathbf{x_1}$**+i+**$\mathbf{x_2}$**+g**. We can consider additional stem candidates, for example, **ea** and **eat**, which are substrings of the original stem. Each stem candidate leads to different realizations for the inflections. For example, the candidate **eat** will give realizations $\mathbf{x_1}$**+en** and $\mathbf{x_1}$**+ing** for the PAST;PCPLE and PRES;PCPLE forms, respectively.

At the start of the sampling process, each paradigm P has a stem S_0, which is a collections of stem parts $S_0(1), ..., S_0(n)$ (for example $S_0(1) =$ **eat** and $S_0(2) =$ **n** in Figure 4b). Each stem part is a sequence of letters $S_0(i)^1, ..., S_0(i)^m$. Intuitively, we sample a new stem S_1 by replacing a random stem part $S_0(i)$ by a random continuous substring of $S_0(i)^k, ..., S_0(i)^{k+l}$. This gives us a new stem proposal

S_1, where each stem part $S_1(j) = S_0(j)$, except $S_1(i)$, which is a substring of $S_0(i)$. For example we could sample a new stem part $S_1(2) = \epsilon$ based on the existing stem part $S_0(2) = \mathbf{n}$. The new stem will now define new realizations for each of the MSDs in paradigm P (for example, $\mathbf{x_1}$**+en** and $\mathbf{x_1}$**+ing**) and we can use statistics about MSD realizations over the entire set of partial paradigms to determine whether to accept or discard the new stem.

Note that whenever we sample a new stem S_k, we sample a substring of the original stem part $S_0(i)$. This allows stem parts to both shrink and grow during the sampling process but they can never grow beyond the stem parts in the original stem S_0.

We set up stem sampling as a product of Chinese Restaurant Processes (CRP) and use Gibbs sampling to optimize the stem assignments for all inflection tables in the data set. This has the effect of preferring MSD realizations which are common. Therefore, sampling will indirectly minimize the number of distinct MSD realizations, which is our goal.

Let P be a randomly selected paradigm, S_i be a stem for P and S_{i+1} a sampled stem. Let $F_i = \{F_i(m_1), ..., F_i(m_k)\}$ and $F_{i+1} = \{F_{i+1}(m_1), ..., F_{i+1}(m_k)\}$ be the set of form patterns, determined by S_i and S_{i+1}, respectively, for each MSD m_1,...,m_k in paradigm P. Further, let $\#(f, m)$ be the count of form pattern f for MSD m over all paradigms $\neq P$. We now accept the new stem setting S_{i+1} with probability $p_a = \prod_j \frac{G_{i+1}(m_j)}{G_i(m_j)}$, where $G_k(m_j) = \alpha$, if $\#(F_k(m_j), m_j) = 0$, and $G_k(m_j) = \#(F_k(m_j), m_j)$, otherwise. Whenever, $p_a > 1$, we automatically accept the new setting S_{i+1}. If we reject the new setting, we set $S_{i+1} := S_i$. Based on preliminary experiments, we set α to 0.1.

We run the sampler for 100 steps over the entire training set. In order to ensure that each paradigm is sampled, we do not randomly select paradigms but instead loop through and sample all paradigms shuffling the data set after each epoch.

Second Baseline using Matrix Completion As an additional baseline, we frame the task of learning complete paradigms as a matrix completion task. We first infer the paradigm structure for each partial inflection table. We then encode each form pattern present in paradigm P into a one-hot representation $\mathbf{v_m} \in \mathbb{R}^N$, where N is the number of distinct realizations of MSD m as shown in Figure 5.[10]

Collecting the rows for each paradigm P, we then form a matrix M, whose rows $\mathbf{r_P} = [\mathbf{v_{m_1}}; ...; \mathbf{v_{m_n}}]$, where m_1, ..., m_n are all the distinct MSDs in the data set. Since the paradigms P are incomplete, M will have a number of missing entries. We complete these entries using the SOFT-IMPUTE algorithm introduced by Mazumder et al. (2010).[11]

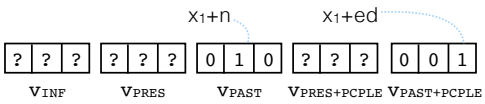


Figure 5: Form patterns are encoded into one-hot vectors and concatenated.

Completed entries $\mathbf{v_m}$ may not be one-hot vectors. Therefore, we transform them into one-hot vectors by applying the transformation $\mathbf{v_m}'[\arg\max_i \mathbf{v_m}[i]] = 1$ and $\mathbf{v_m}'[i] = 0$ otherwise. Finally, we decode the one-hot vectors defined by M into form patterns.

5.2 Experiments and Results

We start with complete inflection tables and uniformly sample a fixed number of forms n from each to form partial inflection tables (n=2,3,4,5 and 6). This gives us collections of partial inflection tables where each one has exactly n forms. We then extract paradigms from the partial inflection tables and use the different methods presented in Section 5.1 to impute the missing forms into the paradigms. More specifically, we apply the following methods (1) sampling of stems followed by priority voting, (2) Priority voting, (3) Majority voting (baseline I), and (4) matrix completion (baseline II).

To evaluate the method, we, after inferring the complete paradigms, substitute stem parts in to form patterns, for example $\mathbf{x_1}$**+ed** $\rightarrow$ **eased**. Thus, we end up with a complete inflection table. We evaluate

[10] In preliminary experiments, we encoded each form pattern as a distinct integer. This, however, delivered poorer performance.

[11] We use the implementation in the `fancyimpute` Python package `https://pypi.python.org/pypi/fancyimpute` by Alex Rubinsteyn and Sergey Feldman.

the result against the gold standard inflection tables. using two metrics: how many of the missing forms were correctly reconstructed and how many of the entire tables were correctly reconstructed.

We perform experiments separately on nouns (nouns and adjectives for Finnish) and verbs. We do this because (1) the performance varies quite a lot between nouns and verbs for the same language, and (2) we only have data for Spanish verbs but we have data both for nominals and verbs for Finnish and German.

Table 3 in Appendix A presents results of the experiment. Overall, we can see that the number of successfully reconstructed forms increases as the number of given forms in the paradigm increases. The best results are attained for German nouns where the number of MSDs in the paradigm is the smallest (8 MSDs per paradigm). Conversely, results are the poorest for Finnish verbs, which have a large number of MSDs per paradigm (54). However, all methods except matrix completion fare quite well on full paradigm learning for Spanish verbs which have even more MSDs per paradigm than Finnish verbs (57).

Matrix completion performs very poorly in comparison to the other methods on all data sets except German nouns. Even for German nouns it, however, performs worse than Baseline I, namely, majority voting. Priority voting consistently outperforms majority voting (by over 30%-points for Spanish verbs when three forms are given). We can also see that sampling gives a consistent improvement over other methods except for German nouns which have the smallest inflection tables (only nine forms per table).

6 Discussion and Conclusions

Our method for paradigm generalization results in a marked decrease in paradigm counts in both Durrett and DeNero (2013) and CoNLL-SIGMORPHON data sets. For Finnish, our paradigm counts for verbs and nouns very closely match the counts arrived at by linguists (40 vs. 51 and 30 vs. 27 for verbs and nouns+adjectives respectively). Qualitative analysis of generalized paradigms shows that the method is clearly capable of capturing regularities like the examples illustrated about Finnish, Spanish, and English.

On the full morphological paradigm learning task, the baseline voting methods (majority and priority voting) are surprisingly effective. Priority voting is able to reconstruct 84% of missing Spanish verb forms and 96% of missing German noun forms when six forms are given in the input inflection table. However, stem sampling on top of priority voting still results in sizable gains over the baseline methods especially in the presence of few given forms in the input table. It is to be expected that the effect diminishes when the number of input forms increases as this reduces the risk of mis-identifying stems. Nevertheless, even with six input forms, stem sampling still results in relatively large gains of around 4%-points for Finnish, German and Spanish verbs which have the largest table sizes in the data set.

Both methods proposed above can be useful for developers of linguistic resources. Paradigm generalization from partial paradigms can be valuable for development of morphological analyzers. Generalized paradigms can guide a linguist working on the phonological and morphological description of a language, and also extract all the allomorphs that are related to a grammatical category.

A clear future research direction for full paradigm learning is exploration of neural methods for the task. While it is to be expected that neural methods will reach high performance, it will be substantially more difficult to draw clear linguistic generalizations from the models they produce. As an added advantage of the current method, we automatically split word forms into stems and inflectional material. Such a split can be difficult to derive from a neural model which can reduce its usefulness as an aid in writing linguistic descriptions.

We have presented a model for automatically inducing generalized morphological paradigms which agrees well with the linguist intuitions concerning morphological pattern generalization since it factors out regular phonological alternations, without actually being 'phonology-aware'. Moreover, we have presented a method for learning full morphological paradigms given training data of partial inflection tables only, and have shown that the method is capable of inferring 90% of missing forms for paradigms as large as 57 forms from only six given forms.

Acknowledgements We want to thank the anonymous reviewers for their valuable comments. The first author is supported by The Society of Swedish Literature in Finland (SLS).

References

Farrell Ackerman, James P. Blevins, and Robert Malouf. 2009. Parts and wholes: Implicative patterns in inflectional paradigms. In James P. Blevins and Juliette Blevins, editors, *Analogy in Grammar*, pages 54–82. Oxford University Press.

Roee Aharoni and Yoav Goldberg. 2017. Morphological inflection generation with hard monotonic attention. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2004–2015, Vancouver, Canada, July. Association for Computational Linguistics.

Malin Ahlberg, Markus Forsberg, and Mans Hulden. 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, Gothenburg, Sweden, April. Association for Computational Linguistics.

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

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Toms Bergmanis, Katharina Kann, Hinrich Schütze, and Sharon Goldwater. 2017. Training data augmentation for low-resource morphological inflection. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 31–39, Vancouver, August. Association for Computational Linguistics.

James P. Blevins and Juliette Blevins, editors. 2009. *Analogy in Grammar*. Oxford University Press.

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, Berlin, Germany, August. Association for Computational Linguistics.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 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, Vancouver, August. Association for Computational Linguistics.

Markus Dreyer and Jason Eisner. 2011. Discovering morphological paradigms from plain text using a Dirichlet process mixture model. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 616–627. 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.

Manaal Faruqui, Yulia Tsvetkov, Graham Neubig, and Chris Dyer. 2016. Morphological inflection generation using character sequence to sequence learning. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 634–643, San Diego, California, June. Association for Computational Linguistics.

Markus Forsberg and Mans Hulden. 2016. Learning transducer models for morphological analysis from example inflections. In *Proceedings of the SIGFSM Workshop on Statistical NLP and Weighted Automata (StatFSM)*, pages 42–50, Berlin, Germany, August. Association for Computational Linguistics.

Martin Haspelmath and Andrea Sims. 2013. *Understanding Morphology*. Routledge.

Bruce Hayes. 2018. Allomorph discovery as a basis for learning alternations. In *Proceedings of the Society for Computation in Linguistics: Vol. 1 , Article 33.*

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. Association for Computational Linguistics.

Katharina Kann and Hinrich Schütze. 2016. Single-model encoder-decoder with explicit morphological representation for reinflection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 555–560, Berlin, Germany, August. Association for Computational Linguistics.

Katharina Kann and Hinrich Schütze. 2017. The LMU system for the CoNLL-SIGMORPHON 2017 shared task on universal morphological reinflection. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 40–48, Vancouver, August. Association for Computational Linguistics.

Christopher Kendris and Theodore Kendris. 2007. *501 Spanish verbs*. Barron's.

Christo Kirov, Ryan Cotterell, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sebastian J. Mielke, Arya McCarthy, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2018. Unimorph 2.0: Universal morphology. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. European Language Resources Association (ELRA).

Jackson L. Lee. 2015. Morphological Paradigms: Computational Structure and Unsupervised Learning. In *Proceedings of NAACL-HLT 2015 Student Research Workshop (SRW)*, pages 161–167, Denver, Colorado, June. Association for Computational Linguistics.

Peter Makarov, Tatiana Ruzsics, and Simon Clematide. 2017. Align and copy: UZH at SIGMORPHON 2017 shared task for morphological reinflection. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 49–57, Vancouver, August. Association for Computational Linguistics.

Robert Malouf. 2016. Generating morphological paradigms with a recurrent neural network. *San Diego Linguistic Papers*, 6:122–129.

Robert Malouf. 2017. Abstractive morphological learning with a recurrent neural network. *Morphology*, 27:431–458.

Rahul Mazumder, Trevor Hastie, and Robert Tibshirani. 2010. Spectral regularization algorithms for learning large incomplete matrices. *Journal of Machine Learning Research*, 11:2287–2322.

Robert Östling. 2016. Morphological reinflection with convolutional neural networks. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 23–26, Berlin, Germany, August. Association for Computational Linguistics.

Miikka Silfverberg and Mans Hulden. 2017a. Automatic morpheme segmentation and labeling in universal dependencies resources. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 140–145, Gothenburg, Sweden, May. Association for Computational Linguistics.

Miikka Silfverberg and Mans Hulden. 2017b. Weakly supervised learning of allomorphy. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pages 46–56, Copenhagen, Denmark, September. Association for Computational Linguistics.

Miikka Silfverberg, Adam Wiemerslage, Ling Liu, and Lingshuang Jack Mao. 2017. Data augmentation for morphological reinflection. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 90–99, Vancouver, August. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3104–3112.

John Sylak-Glassman, Christo Kirov, David Yarowsky, and Roger Que. 2015. A language-independent feature schema for inflectional morphology. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 674–680, Beijing, China, July. Association for Computational Linguistics.

Appendix A. Detailed Results

FINNISH NOUNS AND ADJECTIVES					
Method	forms = 2	forms = 3	forms = 4	forms = 5	forms = 6
1. Sampling	**38.08** ±0.36 (0.09 ±0.11)	**53.71** ±3.27 (4.88 ±1.31)	**60.39** ±0.37 (8.25 ±1.23)	**66.57** ±0.26 (11.59 ±0.80)	**71.32** ±0.95 (21.81 ±1.25)
2. Priority Voting	24.64 ±0.79 (0.00 ±0.00)	45.95 ±3.11 (1.75 ±0.59)	56.09 ±0.45 (7.47 ±0.94)	62.80 ±0.33 (12.59 ±0.95)	68.85 ±1.11 (20.50 ±1.35)
3. Majority Voting (Baseline I)	15.40 ±0.75 (0.00 ±0.00)	27.02 ±1.10 (0.00 ±0.00)	31.40 ±0.28 (0.03 ±0.07)	31.12 ±0.22 (0.38 ±0.30)	29.61 ±0.47 (1.20 ±0.35)
4. Soft Impute (Baseline II)	1.23 ±0.16 (0.00 ±0.00)	2.14 ±0.51 (0.00 ±0.00)	2.53 ±0.12 (0.00 ±0.00)	3.11 ±0.14 (0.00 ±0.00)	3.26 ±0.18 (0.00 ±0.00)
FINNISH VERBS					
Method	forms = 2	forms = 3	forms = 4	forms = 5	forms = 6
1. Sampling	**27.98** ±0.25 (0.00 ±0.00)	**44.29** ±0.16 (0.00 ±0.00)	**54.56** ±0.36 (0.00 ±0.00)	**57.27** ±0.17 (0.00 ±0.00)	**63.31** ±0.20 (0.00 ±0.00)
2. Priority Voting	13.65 ±0.20 (0.00 ±0.00)	30.95 ±0.19 (0.00 ±0.00)	44.68 ±0.28 (0.00 ±0.00)	52.02 ±0.20 (0.00 ±0.00)	59.91 ±0.20 (0.00 ±0.00)
3. Majority Voting (Baseline I)	10.79 ±0.19 (0.00 ±0.00)	22.12 ±0.15 (0.00 ±0.00)	22.10 ±0.22 (0.00 ±0.00)	20.92 ±0.12 (0.00 ±0.00)	20.07 ±0.14 (0.00 ±0.00)
4. Soft Impute (Baseline II)	0.38 ±0.03 (0.00 ±0.00)	0.80 ±0.04 (0.00 ±0.00)	1.02 ±0.04 (0.00 ±0.00)	1.34 ±0.06 (0.00 ±0.00)	1.54 ±0.05 (0.00 ±0.00)
GERMAN NOUNS					
Method	forms = 2	forms = 3	forms = 4	forms = 5	forms = 6
1. Sampling	55.55 ±4.53 (10.64 ±2.10)	65.22 ±7.24 (26.70 ±3.13)	83.45 ±1.99 (66.93 ±2.00)	92.75 ±0.92 (85.02 ±2.90)	95.26 ±1.26 (92.15 ±1.46)
2. Priority Voting	55.28 ±4.40 (10.82 ±2.09)	64.33 ±6.80 (25.43 ±2.63)	82.39 ±1.79 (65.41 ±2.64)	92.12 ±0.85 (84.33 ±2.89)	95.53 ±1.22 (92.40 ±1.41)
3. Majority Voting (Baseline I)	43.64 ±6.32 (2.71 ±1.07)	44.10 ±11.40 (1.45 ±0.99)	50.10 ±7.14 (14.04 ±1.88)	51.95 ±1.26 (21.78 ±2.33)	52.42 ±1.98 (33.32 ±3.20)
4. Soft Impute (Baseline II)	18.59 ±0.99 (0.18 ±0.20)	32.34 ±8.27 (1.45 ±0.81)	32.45 ±5.95 (6.15 ±1.87)	35.06 ±0.73 (10.53 ±2.16)	36.14 ±2.32 (19.57 ±2.33)
GERMAN VERBS					
Method	forms = 2	forms = 3	forms = 4	forms = 5	forms = 6
1. Sampling	**61.24** ±2.41 (**11.74** ±2.27)	**76.23** ±0.90 (**34.68** ±2.76)	**81.54** ±0.52 (**40.16** ±2.40)	**82.79** ±0.99 (**42.03** ±3.19)	**84.49** ±0.34 (**44.89** ±2.42)
2. Priority Voting	40.55 ±3.79 (0.00 ±0.00)	64.39 ±0.86 (11.15 ±2.84)	70.16 ±0.59 (16.18 ±2.01)	75.49 ±0.91 (24.12 ±3.33)	78.65 ±0.44 (28.52 ±2.87)
3. Majority Voting (Baseline I)	14.57 ±2.43 (0.00 ±0.00)	32.12 ±0.38 (0.00 ±0.00)	47.31 ±0.41 (15.98 ±2.99)	72.63 ±0.92 (40.11 ±3.20)	74.43 ±0.42 (41.74 ±2.52)
4. Soft Impute (Baseline II)	8.55 ±1.57 (0.00 ±0.00)	8.35 ±0.42 (0.00 ±0.00)	11.65 ±0.45 (0.00 ±0.00)	12.34 ±0.37 (0.00 ±0.00)	15.67 ±0.35 (0.00 ±0.00)
SPANISH VERBS					
Method	forms = 2	forms = 3	forms = 4	forms = 5	forms = 6
1. Sampling	**49.69** ±2.83 (0.00 ±0.00)	**72.38** ±1.23 (25.38 ±1.72)	**86.23** ±0.15 (**53.88** ±3.37)	**87.26** ±0.14 (**58.35** ±2.07)	**89.29** ±0.11 (**67.08** ±2.38)
2. Priority Voting	34.22 ±1.55 (0.00 ±0.00)	60.01 ±1.02 (25.33 ±1.75)	72.02 ±0.18 (35.71 ±2.74)	80.01 ±0.42 (45.77 ±2.69)	83.64 ±0.21 (53.91 ±2.45)
3. Majority Voting (Baseline I)	21.41 ±1.81 (0.00 ±0.00)	31.00 ±0.59 (0.00 ±0.00)	46.31 ±0.26 (0.00 ±0.00)	53.81 ±0.09 (47.79 ±2.73)	57.94 ±0.05 (51.94 ±2.65)
4. Soft Impute (Baseline II)	0.57 ±0.07 (0.00 ±0.00)	1.28 ±0.06 (0.00 ±0.00)	2.48 ±0.08 (0.00 ±0.00)	3.63 ±0.15 (0.00 ±0.00)	4.46 ±0.09 (0.00 ±0.00)

Table 3: Results for learning full paradigms from partial paradigms. The columns give results in presence of n forms in the partial paradigms. We give the amount of missing forms which were successfully reconstructed. In parentheses, we give the amount of complete paradigms which were successfully reconstructed. We give confidence intervals at the 99% level from a one-sided t-test and show the cases where sampling gives significantly better results than the other methods in bold typeface.