Neural Transductive Learning and Beyond: Morphological Generation in the Minimal-Resource Setting

Katharina Kann
Center for Data Science
New York University, USA
kann@nyu.edu

Hinrich Schütze
CIS
LMU Munich, Germany
inquiries@cislmu.org

Abstract

Neural state-of-the-art sequence-to-sequence (seq2seq) models often do not perform well for small training sets. We address paradigm completion, the morphological task of, given a partial paradigm, generating all missing forms. We propose two new methods for the minimal-resource setting: (i) *Paradigm transduction*: Since we assume only few paradigms available for training, neural seq2seq models are able to capture relationships between paradigm cells, but are tied to the idiosyncracies of the training set. Paradigm transduction mitigates this problem by exploiting the input subset of inflected forms at test time. (ii) *Source selection with high precision (SHIP)*: Multi-source models which learn to automatically select one or multiple sources to predict a target inflection do not perform well in the minimal-resource setting. SHIP is an alternative to identify a reliable source if training data is limited. On a 52-language benchmark dataset, we outperform the previous state of the art by up to 9.71% absolute accuracy.

1 Introduction

Morphological generation of previously unencountered word forms is a crucial problem in many areas of natural language processing (NLP). High performance can lead to better systems for downstream tasks, e.g., machine translation (Tamchyna et al., 2017). Since existing lexicons have limited coverage, learning morphological inflection patterns from labeled data is an important mission and has recently been the subject of multiple shared tasks (Cotterell et al., 2016, 2017a).

In morphologically rich languages, words inflect, i.e., they change their surface form in oder to express certain properties, e.g., number or tense. A word's canonical form, which can be found in a dictionary, is called the lemma, and the set of all inflected forms is referred to as the lemma's paradigm.

	Singular	Plural
NOM	*Schneemann*	*Schneemänner*
GEN	**Schneemannes**	*Schneemänner*
DAT	*Schneemann*	**Schneemännern**
ACC	*Schneemann*	*Schneemänner*
LEMMA	Schneemann	

Figure 1: The paradigm of the German noun "Schneemann" ("snowman"). In this running example, the input subset is bold, the output subset italic.

In this work, we address paradigm completion (PC), the morphological task of, given a partial paradigm of a lemma, generating all of its missing forms. For the partial paradigm represented by the input subset {("Schneemannes", GEN;SG), ("Schneemännern", DAT;PL)} of the German noun "Schneemann" shown in Figure 1, the goal of PC is to generate the output subset consisting of the six remaining forms.

Neural seq2seq models define the state of the art for morphological generation if training sets are large; however, they have been less successful in the low-resource setting (Cotterell et al., 2017a). In this paper, we address an even more extreme *minimal-resource* setting: for some of our experiments, our training sets only contain $k \approx 10$ paradigms. Each paradigm has multiple cells, so the number of *forms* (as opposed to the number of *paradigms*) is not necessarily minimal. However, we will see that generalizing from paradigm to paradigm is a key challenge, making the number of paradigms a good measure of the effective training set size.

We propose two PC methods for the minimal-resource setting: *paradigm transduction* and *source selection with high precision (SHIP)*. We define a learning algorithm as *transductive*[1] if its goal is to generalize from specific training examples to specific test examples (Vapnik, 1998). In contrast, in-

[1] In order to avoid ambiguity, "transduction" is never used in the sense of string-to-string transduction in this paper.

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 3254–3264
Brussels, Belgium, October 31 - November 4, 2018. ©2018 Association for Computational Linguistics

ductive inference learns a general model that is independent of any test set. Predictions of transductive inference for the same item are different for different test sets. There is no such dependence in inductive inference. Our motivation for transduction is that, in the minimal-resource setting, neural seq2seq models capture relationships between paradigm cells like affix substitution and umlauting, but are tied to the idiosyncracies of the k training paradigms. For example, if all source forms in the training set start with "b" or "d", a purely inductive model may then be unable to generate targets with different initials. By transductive inference on the information available in the input subset at test time, i.e., the given partial paradigm, our model can learn idiosyncracies. For example, if the input subset sources start with "p", we can learn to generate output subset targets that start with "p". Thus, we exploit the input subset for learning idiosyncracies at test time and then generate the output subset using a modified model. This setup employs standard inductive training (on the training set) for learning general rules of inflectional morphology and transductive inference (on the test set) for learning idiosyncracies. Our use of transduction is innovative in that most previous work has addressed unstructured problems whereas our problem is structured: we complete a paradigm, a complex structure of forms, each of them labeled with a morphological tag. Thus, the test set contains labels, whereas, in transduction for unstructured problems, the test set is a flat set of unlabeled instances. We view our work as an extension of transduction to the structured case, even though not all elements of the theory developed by Vapnik (1998) carry over.

The motivation for our second PC method for limited training data, SHIP, is as follows. Multi-source models can learn which combination of sources most reliably predicts the target in the high-resource, but less well in the minimal-resource setting. SHIP models the relationship between paradigm slots using edit trees (Chrupała et al., 2008), in order to measure how deterministic each transformation is. Then, it identifies the most deterministic source slot for the generation of each target inflection.

Paradigm transduction and SHIP can be employed separately or in combination. Our experiments show that, in an extreme minimal-resource setting, a combination of SHIP and a non-neural approach is most effective; for slightly more data, a combination of a neural model, paradigm transduction and SHIP obtains the best results.

Contributions. (i) We introduce neural paradigm transduction, which exploits the structure of the PC task to mitigate the negative effect of limited training data. (ii) We propose SHIP, a new algorithm for picking a single reliable source for PC in the minimal-resource setting. (iii) On average over all languages of a 52-language benchmark dataset, our approaches outperform state-of-the-art baselines by up to 9.71% absolute accuracy.

2 Paradigm Completion

In this section, we formally define our task, developing the notation for the rest of the paper.

Given the set of morphological tags $T(w)$ of a lemma w, we define the paradigm of w as the set of tuples of inflected form f_k and tag t_k:

$$\pi(w) = \big\{ \big(f_k[w], t_k\big) \big\}_{t_k \in T(w)} \tag{1}$$

The example in Figure 1 thus corresponds to: π(Schneemann) = $\big\{\big($"Schneemann", NOM;SG$\big)$ $\ldots\big($"Schneemänner", ACC;PL$\big)\big\}$.

A training set in our setup consists of complete paradigms, i.e., all inflected forms of each lemma are available. This simulates a setting in which a linguist annotates complete paradigms, as done, e.g., in Sylak-Glassman et al. (2016). In contrast, each element of the test set is a partial paradigm, which we refer to as the *input subset*. This simulates a setting in which we collect all forms of a lemma occurring in a (manually or automatically) annotated input corpus; this set will generally not be complete. The PC task consists of generating the *output subset* of the paradigm, i.e., the forms belonging to form-tag pairs which are missing from the collected subset.

3 Method

Our approach for PC is based on MED (*Morphological Encoder-Decoder*), a state-of-the-art model for morphological generation in the high-resource case, which was developed by Kann and Schütze (2016b). In this section, we first cover required background on MED and then introduce our new approaches.

3.1 MED

Input and output format. MED converts one inflected form of a paradigm into another, given the two respective tags. Thus, the input of MED is a sequence of subtags of the source and the target form (e.g., NOM and SG are subtags of NOM;SG), as well as the characters of the source form. All elements are represented by embeddings, which are trained together with the model. The output of MED is the character sequence of the target inflected form.

An example from the paradigm in Figure 1 is:

INPUT: DAT^S PL^S GEN^T SG^T Schneemännern
OUTPUT: Schneemannes

Encoder. The model's encoder consists of a bidirectional gated recurrent neural network (GRU) with a single hidden layer. It reads an input vector sequence $x=(x_1,\ldots,x_{X_t})$ and encodes it from two opposite directions into two hidden representations $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ as

$$\overrightarrow{h_t} = \text{GRU}(x_t, \overrightarrow{h_{t-1}}) \quad (2)$$

$$\overleftarrow{h_t} = \text{GRU}(x_t, \overleftarrow{h_{t+1}}) \quad (3)$$

which are concatenated to

$$h_t = \left[\overrightarrow{h_t}; \overleftarrow{h_t}\right] \quad (4)$$

Decoder. The decoder, another GRU with a single hidden layer, defines a probability distribution over the output vocabulary, which, for paradigm completion, consists of the characters in the language, as

$$p(y) = \prod_{t=1}^{T_y} \text{GRU}(y_{t-1}, s_{t-1}, c_t) \quad (5)$$

s_t denotes the state of the decoder at step t, and c_t is the sum of the hidden representations of the encoder, weighted by an attention mechanism.

Additional background on the general model architecture is given in Bahdanau et al. (2015); details on MED can be found in Kann and Schütze (2016b).

3.2 Semi-supervised MED

In order to make use of unlabeled data with MED, Kann and Schütze (2017) defined an auxiliary autoencoding task and proposed a multi-task learning approach.

For this extension, an additional symbol is added to the input vocabulary. Each input is then of the form $(\mathbf{A} \mid M^+)\,\Sigma^+$, with **A** being a novel tag for autoencoding, Σ being the alphabet of the language, and M being the set of morphological subtags of the source and the target. As for the basic MED model, all parts of the input are represented by embeddings.

The training objective is to maximize the joint likelihood for the tasks of paradigm completion and autoencoding:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) = & \sum_{(s,t^S,t^T,w)\in\mathcal{D}} \log p_{\boldsymbol{\theta}}(w \mid e_\theta(t^S,t^T,s)) \\ & + \sum_{a\in\mathcal{A}} \log p_{\boldsymbol{\theta}}(a \mid e_\theta(a))\end{aligned}$$

where $\mathcal{A}$ is a set of autoencoding examples, e_θ is the encoder, and $\mathcal{D}$ is a labeled training set of tuples of source s, morphological source tag t^S, morphological target tag t^T, and target w.

3.3 MED for Paradigm Completion

MED was originally developed for morphological reinflection. Thus, it operates on pairs consisting of a single source and a single target form. In order to use it for paradigm completion, where multiple source forms are given, and multiple target forms are expected, we convert the given data into a suitable format in the way described in the following.

For a lemma w, let $J(w)$ be the set of tags in the input subset. Recall that $J(w)$ is a subset of $T(w)$, the set of all tags, at test time, but that training paradigms are complete, i.e., $J(w)=T(w)$ for the training set.

For both training of the inductive model and paradigm transduction, we generate $|J(w)|(|J(w)|-1)$ training examples

$$(t_i, t_j, f_i[w]) \mapsto f_j[w]$$

one for each pair of different tags in $J(w)$. We also generate autoencoding training examples for all tags in $J(w)$ (removing duplicates):

$$\big(\mathbf{A}, f_i[w]\big) \mapsto f_i[w]$$

For the German lemma "Schneemann", assume:

$J(\text{Schneemann}) = \{\text{GEN;SG,DAT;PL}\}$

at test time. We then produce the following training examples for paradigm transduction:

(DAT^S PL^S GEN^T SG^T Schneemännern) ↦ Schneemannes
(GEN^S SG^S DAT^T PL^T Schneemannes) ↦ Schneemännern
(**A** Schneemannes) ↦ Schneemannes
(**A** Schneemännern) ↦ Schneemännern

For completing a partial paradigm, we then select one source form per target slot (the lemma, unless stated otherwise) and create all forms corresponding to the tags in $J(w)\setminus T(w)$ one by one.

3.4 Paradigm Transduction

Motivation. In the minimal-resource setting, parameter estimates are tied to the idiosyncracies of the lemmas seen in training, due to overfitting. Our example in §1 is that the model has difficulties producing initial letters not seen during training. However, within each paradigm, forms are generally similar; thus, input subset sources contain valuable information about how to generate output

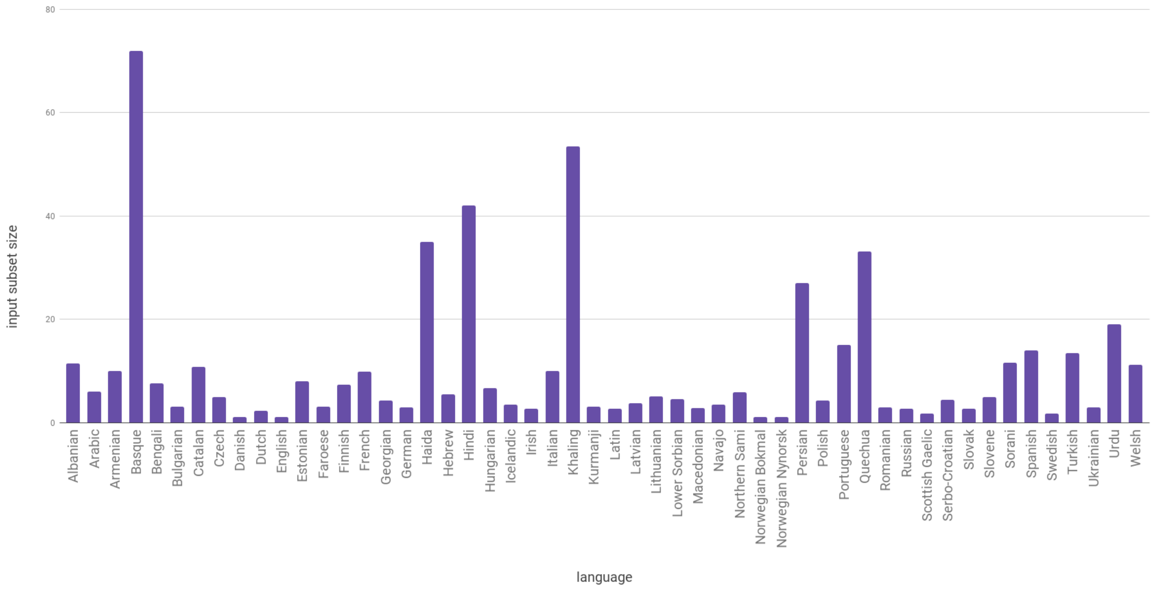

Figure 2: Average amount of sources in the input subset for paradigm transduction, per language.

subset targets. Based on this observation, we solve the problem of overfitting by transduction: we teach the model test idiosyncracies by training it on the input subset before generating the output subset.

Method description. We first train a general model on the training set in the standard supervised learning setup, i.e., the setup which is called inductive inference by Vapnik (1998). At test time, we take the general model as initialization and continue training on examples generated from the input subset as described in §3.3. We do this separately for each lemma, satisfying the defining criterion of transductive inference that predictions depend on the test data. Also, different input subsets (i.e., different subsets of the same paradigm) can in general make different predictions on an output subset target.

Paradigm transduction is expected to perform best in a setting in which many forms of each paradigm are given as input, i.e., when $|J(w)|$ is big. In Figure 2 we show the average sizes of the input subsets for all languages in our experiments.

3.5 Source Selection with High Precision

During PC, some sources contain more information relevant to generating certain targets than others. For instance, the nominative singular and accusative singular in German are generally identical (cf. Figure 1); thus, for generating the accusative singular, we should use the nominative singular as source if it is available—rather than, say, the dative plural.

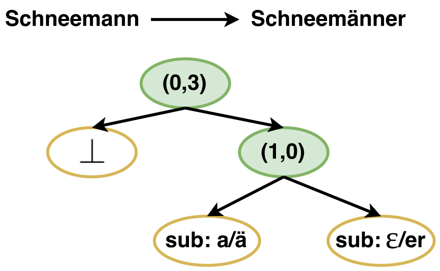

Figure 3: Edit tree example. Each node gives lengths of the parts before/after LCS, e.g., the root has LCS "Schneem", before part ϵ and after part "ann", thus the lengths are "(0,3)". "sub" = "substitution".

In fact, for many languages, the entire paradigm of most lemmas is deterministic if the right source forms are known and used for the right targets. A set of forms that determines all other inflected forms is called *principal parts* (Finkel and Stump, 2007). Based on this theory, Cotterell et al. (2017b) induce topologies and jointly decode entire paradigms, thus making use of all available forms. However, their method is only applicable if good estimates of the probabilities $p(f_j[w]|f_i[w])$ for source $f_i[w]$ and target $f_j[w]$ can be obtained, and they train on hundreds of paradigms per part of speech (POS) and language, which are not available in our setup.

We propose an alternative for the minimal-resource setting: SHIP, which selects a single best source for each target and is based on edit trees. An edit tree $e(f_i[w], f_j[w])$ is a transformation from a source $f_i[w]$ to a target $f_j[w]$ (Chrupała et al., 2008); see Figure 3. It is constructed by first determining

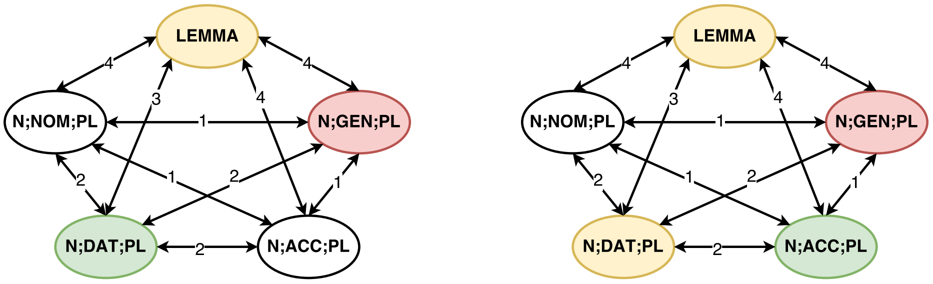

Figure 4: SHIP example for German plural forms (SET1). For the graph constructed in training (see §3.5), subgraphs are extracted in testing for input subset sizes two (left) and three (right). Input subset: yellow and green. Output subset: white and red. For generation of the target shown in red, SHIP selects the source shown in green.

the longest common substring (LCS) (Gusfield, 1997) of $f_i[w]$ and $f_j[w]$ and then modeling the prefix and suffix pairs of the LCS recursively. In the case of an empty LCS, $e(f_i[w], f_j[w])$ is the substitution operation that replaces $f_i[w]$ with $f_j[w]$.

We construct edit trees for each pair $(f_i[w], f_j[w])$ in the training set, count the number n_{ij} of different edit trees for $t_i \mapsto t_j$, and construct a fully connected graph. The tags are nodes of the graph, and the counts n_{ij} are weights. Edges are undirected, since edit trees are bijections (cf. Figure 4). We then interpret the weight of an edge as a measure of the (un)reliability of the corresponding two source-target relationships. Our intuition is that the fewer different edit trees relate source and target, the more reliable the source is for generating the target.

At test time, we find for each target t_j a source t_k such that $n_{kj} \leq n_{ij} \forall i \in J(w)$. We then use $f_k[w]$ to generate $f_j[w]$. Again, Figure 4 shows examples.

4 Experiments

4.1 Data

We run experiments on the datasets from task 2 of the CoNLL–SIGMORPHON 2017 shared task, which have been created using UniMorph (Kirov et al., 2018). We give a short overview here; see (Cotterell et al., 2017a) for details. The dataset contains, for each of 52 languages, a development set of 50 partial paradigms, a test set of 50 partial paradigms, and three training sets of complete paradigms. Training set sizes are 10 (SET1), 50 (SET2), and 200 (SET3). Recall that we view the number of paradigms (not the number of forms) as the best measure of the amount of training data available. Even for SET3, there are only 200 lemmas per language in the training set, which are additionally distributed over multiple POS tags, compared to >600 lemmas per POS used by Cotterell et al. (2017b). We, thus, want to emphasize that all settings—SET1, SET2, and SET3—can be considered low-resource.

We produce training sets for our encoder-decoder as described in §3.3, but limit the total number of training examples to 200,000.

4.2 Hyperparameters

With our hyperparameters, we follow Kann and Schütze (2016a). In particular, our encoder and decoder GRUs have 100-dimensional hidden states. Our embeddings are 300-dimensional. For training, we use stochastic gradient descent, ADADELTA (Zeiler, 2012), and minibatches of size 20. After experiments on the development set, we decide on training SET1, SET2, and SET3 models for 50, 30, and 20 epochs, respectively. For paradigm transduction, we train all models for 25 additional epochs.

4.3 Baselines

In the following, we describe our baselines. COPY, MED, and PT are used for ablation and SIG17 for comparison with the state of the art.

COPY. As targets in many paradigm cells in many languages are identical to the lemma, we consider a copy baseline that simply copies the lemma.

MED. This is the model by Kann and Schütze (2016b), which performed best at SIGMORPHON 2016. For decoding, the lemma is used. Since MED is designed for the high-resource setting, we do not expect good performance for our minimal-resource scenario, but the comparison shows how much our enhancements improve performance.

Pure paradigm transduction (PT). PT is a seq2seq model exclusively trained on the input subset. Its performance sheds light on the importance of the initial inductive training.

SIG17. SIG17 is the official baseline of the CoNLL–SIGMORPHON 2017 shared task, which was developed to perform well with very little

	SET1	SET2	SET3
BL: COPY	.0810	.0810	.0810
BL: MED	.0004	.0432	.4211
BL: PT	.0833	.0833	.0775
BL: SIG17	.5012	.6576	.7707
SIG17+SHIP	**.5971**	.7355	.8008
MED+PT	.5808	.7486	.8454
MED+PT+SHIP	.5793	**.7547**	**.8483**

Table 1: Accuracy on PC for SIG17+SHIP (the shared task baseline SIG17 with SHIP), MED+PT (MED with paradigm transduction), MED+PT+SHIP (MED with paradigm transduction and SHIP), as well as all baselines (*BL*). Results are averaged over all languages, and best results are in bold; detailed accuracies for all languages can be found in Appendix A.

training data. Its design follows Liu and Mao (2016): SIG17 first aligns each input lemma and output inflected form. Afterwards, it assumes that each aligned pair can be split into a prefix, a stem, and a suffix. Based on this alignment, the system extracts prefix (resp. suffix) rules from the prefix (resp. suffix) pairings. At test time, suitable rules are applied to the input string to generate the target; more details can be found in Cotterell et al. (2017a).

4.4 Results

Our results are shown in Table 1. For SET1, SIG17+SHIP obtains the highest accuracy, while, for SET2 and SET3, MED+PT+SHIP performs best. This difference can be easily explained by the fact that the performance of neural networks decreases rapidly for smaller training sets, and, while paradigm transduction strongly mitigates this problem, it cannot completely eliminate it. Overall, however, SIG17+SHIP, MED+PT, and MED+PT+SHIP all outperform the baselines by a wide margin for all settings.

Effect of paradigm transduction. On average, MED+PT clearly outperforms SIG17, the strongest baseline: by .0796 (.5808-.5012) on SET1, .0910 (.7486-.6576) on SET2, and .0747 (.8454-.7707) on SET3.

However, looking at each language individually (refer to Appendix A for those results), we find that MED+PT performs poorly for a few languages, namely Danish, English, and Norwegian (Bokmål & Nynorsk). We hypothesize that this can most likely be explained by the size of the input subset of those languages being small (cf. Figure 2 for average input subset sizes per language). Recall that the input subset is explored by the model during transduction. Most poorly performing languages have input subsets containing only the lemma; in this case paradigm transduction reduces to autoencoding the lemma. Thus, we conclude that paradigm transduction can only improve over MED if two or more sources are given.

Conversely, if we consider only the languages with an average input subset size of more than 15 (Basque, Haida, Hindi, Khaling, Persian, and Quechua), the average accuracy of MED+PT for SET1 is 0.9564, compared to an overall average of 0.5808. This observation shows clearly that paradigm transduction obtains strong results if many forms per paradigm are given.

Effect of SHIP. Further, Table 1 shows that SIG17+SHIP is better than SIG17 by .0959 (.5971-.5012) on SET1, .0779 (.7355-.6576) on SET2, and .0301 (.8008-.7707) on SET3. Stronger effects for smaller amounts of training data indicate that SHIP's strategy of selecting a single reliable source is more important for weaker final models; in these cases, selecting the most deterministic source reduces errors due to noise.

In contrast, the performance of MED, the neural model, is relatively independent of the choice of source; this is in line with earlier findings (Cotterell et al., 2016). However, even for MED+PT, adding SHIP (i.e., MED+PT+SHIP) slightly increases accuracy by .0061 (.7547-.7486) on SET2, and .0029 (.8483-.8454) on SET3 (L53).

Ablation. MED does not perform well for either SET1 or SET2. In contrast, on SET3 it even outperforms SIG17 for a few languages. However, MED loses against MED+PT in all cases, highlighting the positive effect of paradigm transduction.

Looking at PT next, even though PT does not have a zero accuracy for any setting or language, it performs consistently worse than MED+PT. For SET3, PT is even lower than MED on average, by .3436 (.4211-.0775). Note that, in contrast to the other methods, PT's performance is not dependent on the size of the training set. The main determinant for PT's performance is the size of the input subset during transductive inference. If the input subset is large, PT can perform better than MED, e.g., for Hindi and Urdu. For Khaling SET1, PT even outperforms both MED and SIG17. However, in most cases, PT does not perform well on its own.

MED+PT outperforms both MED and PT. This confirms our initial intuition: MED and PT learn complementary information for paradigm

input	output		
	MED	PT	MED+PT
Schneemann N;GEN;PL	GetGächen	Scnneeeeennnnnnnnnnnnnnnnnnnnnn	Schneemänner
dish V;V.PTCP;PRS	dising	dish	dishing
creer V;SBJV;PRS;1;PL	crezcamos	creyemos	creamos

Table 2: Analysis of the outputs of MED, PT, and MED+PT for SET2. Top to bottom: German, English, Spanish. MED and PT produce incorrect, MED+PT correct inflections.

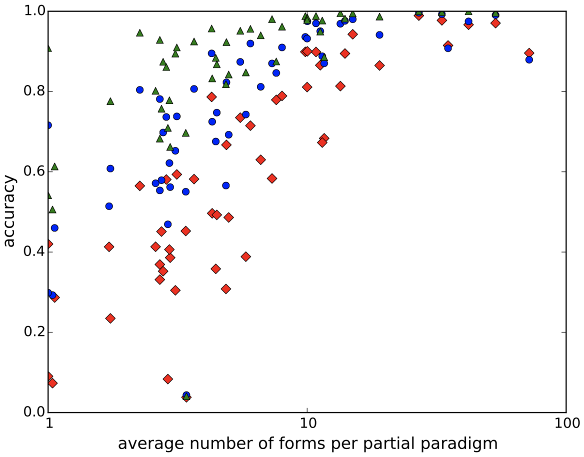

Figure 5: Accuracy of MED+PT as a function of the average input subset size. Red/diamonds: SET1; blue/circles: SET2; green/triangles: SET3.

completion. The base model learns the general structure of the language (i.e., correspondences between tags and inflections) while paradigm transduction teaches the model which character sequences are common in a specific test paradigm.

5 Analysis

5.1 On the Size of the Input Subset

We expect paradigm transduction to become more effective as the size of the input subset increases. Figure 5 shows the accuracy of MED+PT as a function of the average input subset size for SET1, SET2, and SET3. Accuracy for languages with input set sizes above 15 is higher than .8 in all settings. In general, languages with larger input set sizes perform better. The correlation is not perfect because languages have different degrees of morphological regularity. However, the overall trend is clearly recognizable.

The organizers of CoNLL–SIGMORPHON provided large input subsets in the development and test sets of languages with large paradigms. Thus, PT performs better for languages with many inflected forms per paradigm, i.e., large $|T(w)|$.

5.2 On the Effect of Paradigm Transduction

We further analyze why paradigm transduction improves the performance of the base model MED, using the German, English, and Spanish SET2 examples for MED, PT, and MED+PT given in Table 2.

German. MED generates an almost random sequence. However, it learns that the umlaut “ä” must appear in the target. PT only produces correct characters, but it produces far too many. The reason may be that the model is trained on both a double “e” and a double “n”, learning that “e” and “n” are likely to appear repeatedly. MED+PT generates the correct target.

English. MED fails to generate “h” because the bigram “sh” did not occur in training, and so the probability of “h” following “s” is estimated to be low. PT fails to produce the suffix “ing”, since it does not occur in the input subset, and, thus, PT has no way of learning it. Again, MED+PT generates the correct target.

Spanish. MED produces “crezcamos”, a form that has the correct tag V;SBJV;PRS;1;PL, but is a form of “crecer” (which appears in the training set), not of “creer” (which does not). This demonstrates the problems resulting from a lack of lemma diversity during training. PT produces a combination of several of the forms in the input subset: subjunctive forms beginning with “crey” and “creemos” V;IND;PRS;1;PL. Again, MED+PT generates the correct target.

Overall, this analysis confirms that MED learns relationships between paradigm cells, while paradigm transduction adds knowledge about the idiosyncracies of a partial test paradigm.

5.3 Comparison to Multi-Source Models

In this section, we explicitly compare our approach to neural multi-source models for morphological generation.

Following Kann et al. (2017a), we employ attention-based RNN encoder-decoder networks with two or four input sources. The input to a

	SET1				SET2				SET3			
	1	2	4	+PT	1	2	4	+PT	1	2	4	+PT
dutch	.00	.00	.00	**.49**	.04	.01	.00	**.78**	.43	.65	.72	**.87**
german	.00	.00	.00	**.65**	.00	.00	.01	**.75**	.44	.42	.59	**.88**
icelandic	.00	.00	.00	**.41**	.03	.02	.02	**.50**	.24	.33	.35	**.77**
spanish	.00	.00	.00	**.92**	.03	.09	.09	**.98**	.59	.63	.83	**.99**
welsh	.00	.00	.00	**.91**	.05	.14	.15	**.97**	.35	.53	.70	**.99**

Table 3: MED accuracy on five randomly selected languages with 1, 2, and 4 sources and combined with paradigm transduction ("+PT"). Best results in bold.

multi-source model is the concatenation of all sources and corresponding tags. During training, we randomly sample (with repetition) one or three additional forms from the paradigm of each example. At test time, we sample the additional forms from the given partial paradigm; without repetition first, but repeating if not enough inflected forms are available. For autoencoding examples in the training data, we simply concatenate two or four copies of the source and the autoencoding tag. We randomly select five languages for this experiment.

Table 3 shows that, for SET3, four sources (column header "4") are generally better than two sources ("2"), which in turn are better than one source ("1"); thus, as expected, making additional sources available in training improves results. We attribute one exception (German accuracy is .4391 for "1" and .4179 for "2") to the noisiness of the problem—training sets in terms of number of paradigms are relatively small, even for SET3.

The improvements we see for SET3 are large. This suggests that using more than four sources would further improve results and perhaps reach the level of performance of MED+PT, at the cost of a long training time. However, for SET1 and SET2, there is no consistent improvement from 1 to 2 to 4 sources. While it is possible that further optimization could improve the best multi-source result given in Table 3, the gap to MED+PT is very large, and the improvement from 2 to 4 is small. This indicates that multi-source methods cannot compete with transductive learning for SET1 and SET2.

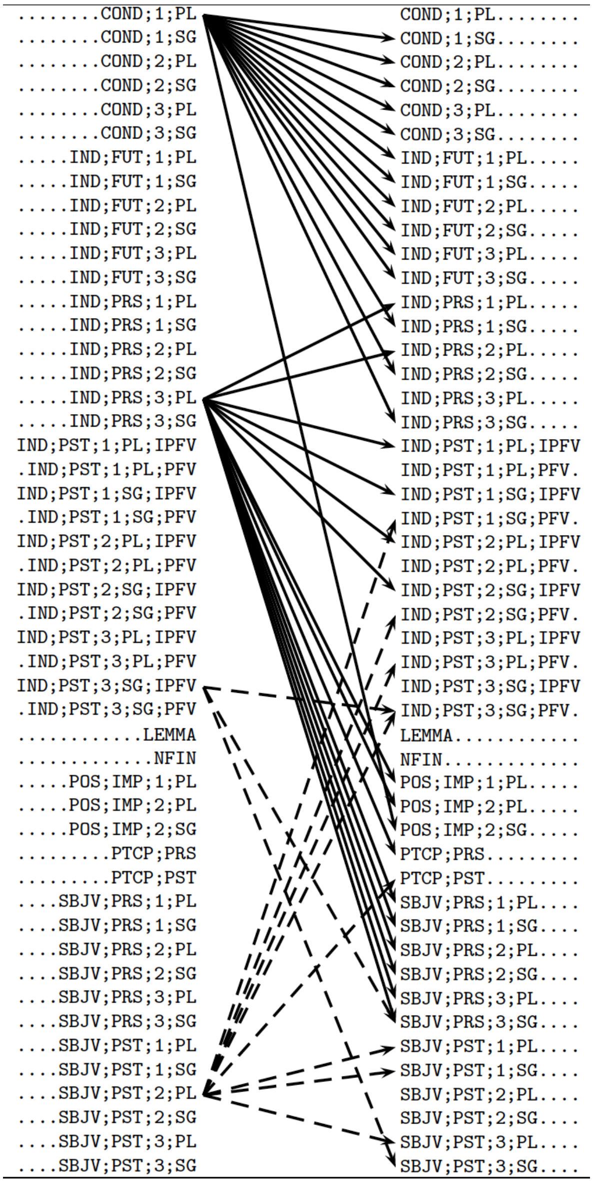

Figure 6: Right: output set target to be generated. Left: input set source selected by SHIP. Arrows for the two most frequently selected sources are solid, arrows for the two least frequently selected sources are dashed.

5.4 Qualitative Analysis of SHIP

For a qualitative analysis of SHIP, we look at the sources it selects for French verbs on the development set; the complete diagram is shown in Figure 6. For most verbs, future and conditional can be predicted from COND;1;PL (e.g., "finirions"), and indicative present, indicative imparfait and subjunctive present from IND;PRS;3;PL (e.g., "finissent"). In case of ties, SHIP selects the alphabetically first tag; this explains why COND;1;PL gets preference over IND;PRS;3;PL for indicative present singular. These two forms represent two of the *principal* *parts* of French conjugation, the infinitive (almost always derivable from COND;1;PL) and the stem that is used for plural indicative, imparfait, and other paradigm cells—which is sometimes not derivable from the infinitive as is the case for “finir”. In comparison, IND;PST;3;SG;IPFV and SBJV;PST;2;PL are less reliable sources. But they are still reasonably accurate if no better alternative is available; consider the following SBJV;PST;2;PL → IND;PST;1;SG;PFV generations: “parlassiez” ↦ “parlai”, “finissiez” ↦ “finis”, “missiez” ↦ “mis”, “prissiez” ↦ “pris”.

We thus conclude that SHIP indeed learns to select appropriate source forms.

6 Related Work

Morphological generation. In the last two years, most work on paradigm completion has been done in the context of the SIGMORPHON 2016 and the CoNLL–SIGMORPHON 2017 shared tasks (Cotterell et al., 2016, 2017a). Due to the success of neural seq2seq models in 2016 (Kann and Schütze, 2016b; Aharoni et al., 2016), systems developed for the 2017 edition were mostly neural (Makarov et al., 2017; Bergmanis et al., 2017; Zhou and Neubig, 2017). Besides the shared task systems, Kann and Schütze (2017) presented a paradigm completion model for a multi-source setting that made use of an attention mechanism to decide which input form to attend to at each time step. They used randomly chosen, independent pairs of source and target forms for training. This differs crucially from the setting we consider in that no complete paradigms were available in their training sets. Only Cotterell et al. (2017b) addressed essentially the same task we do, but they only considered the high-resource setting: their models were trained on hundreds of complete paradigms. The experiments reported in §5.3 empirically confirm that inductive-only models perform poorly in our setting.

Several ways to employ neural models for morphological generation with limited data have been proposed, e.g., semi-supervised training (Zhou and Neubig, 2017; Kann and Schütze, 2017) or simultaneous training on multiple languages (Kann et al., 2017b). The total number of sources in the training set in some of our settings may be comparable to this earlier work, but our training sets are less diverse since many forms come from the same paradigm. We argue in §1 that the number of paradigms (not the number of sources) measures the effective size of the training set.

Other important work on morphological generation—neural and non-neural—includes Dreyer et al. (2008); Durrett and DeNero (2013); Hulden et al. (2014); Nicolai et al. (2015); Faruqui et al. (2016); Yin et al. (2016).

Seq2seq models in NLP. Even though neural seq2seq models were originally designed for machine translation (Sutskever et al., 2014; Cho et al., 2014; Bahdanau et al., 2015), their application has not stayed limited to this area. Similar architectures have been successfully applied to many seq2seq tasks in NLP, e.g., syntactic parsing (Vinyals et al., 2015), language correction (Xie et al., 2016), normalization of historical texts (Bollmann et al., 2017), or text simplification (Nisioi et al., 2017). Transductive inference is similar to domain adaptation, e.g., in machine translation (Luong and Manning, 2015). One difference is that training set and test set can hardly be called different domains in paradigm completion. Another difference is that explicit structured labels (the morphological tags of the forms in the input subset) are available at test time in paradigm completion.

7 Conclusion

We presented two new methods for minimal-resource paradigm completion: paradigm transduction and SHIP. Paradigm transduction learns general inflection rules through standard inductive training and idiosyncracies of a test paradigm through transduction. We showed that paradigm transduction effectively mitigates the problem of overfitting due to a lack of diversity in the training data. SHIP is a robust non-neural method that identifies a single reliable source for generating a target. In the minimal-resource setting, this is an effective alternative to learning how to combine evidence from multiple sources. Considering the average over all languages of a 52-language benchmark dataset, we outperform the previous state of the art by at least 7.07%, and up to 9.71% absolute accuracy.

Acknowledgments

We would like to thank Samuel Bowman, Ryan Cotterell, Nikita Nangia, and Alex Warstadt for their feedback on this work.

References

Roee Aharoni, Yoav Goldberg, and Yonatan Belinkov. 2016. Improving sequence to sequence learning for morphological inflection generation: The biu-mit systems for the sigmorphon 2016 shared task for morphological reinflection. In *SIGMORPHON*.

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 *CoNLL–SIGMORPHON*.

Marcel Bollmann, Joachim Bingel, and Anders Søgaard. 2017. Learning attention for historical text normalization by learning to pronounce. In *ACL*.

Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. In *SSST*.

Grzegorz Chrupała, Georgiana Dinu, and Josef Van Genabith. 2008. Learning morphology with morfette. In *LREC*.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2017a. The CoNLL–SIGMORPHON 2017 shared task: Universal morphological reinflection in 52 languages. In *CoNLL–SIGMORPHON*.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. The SIGMORPHON 2016 shared task—morphological reinflection. In *SIGMORPHON*.

Ryan Cotterell, John Sylak-Glassman, and Christo Kirov. 2017b. Neural graphical models over strings for principal parts morphological paradigm completion. In *EACL*.

Markus Dreyer, Jason R Smith, and Jason Eisner. 2008. Latent-variable modeling of string transductions with finite-state methods. In *EMNLP*.

Greg Durrett and John DeNero. 2013. Supervised learning of complete morphological paradigms. In *NAACL-HLT*.

Manaal Faruqui, Yulia Tsvetkov, Graham Neubig, and Chris Dyer. 2016. Morphological inflection generation using character sequence to sequence learning. In *NAACL-HLT*.

Raphael Finkel and Gregory Stump. 2007. Principal parts and degrees of paradigmatic transparency. Technical report, Department of Computer Science, University of Kentucky, Lexington, KY.

Dan Gusfield. 1997. *Algorithms on strings, trees and sequences: computer science and computational biology*. Cambridge university press.

Mans Hulden, Markus Forsberg, and Malin Ahlberg. 2014. Semi-supervised learning of morphological paradigms and lexicons. In *EACL*.

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2017a. Neural multi-source morphological reinflection. In *EACL*.

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2017b. One-shot neural cross-lingual transfer for paradigm completion. In *ACL*.

Katharina Kann and Hinrich Schütze. 2016a. MED: The LMU system for the SIGMORPHON 2016 shared task on morphological reinflection. In *SIGMORPHON*.

Katharina Kann and Hinrich Schütze. 2016b. Single-model encoder-decoder with explicit morphological representation for reinflection. In *ACL*.

Katharina Kann and Hinrich Schütze. 2017. The LMU system for the CoNLL–SIGMORPHON 2017 shared task on universal morphological reinflection. In *CoNLL–SIGMORPHON*.

Katharina Kann and Hinrich Schütze. 2017. Unlabeled data for morphological generation with character-based sequence-to-sequence models. In *SCLeM*.

Christo Kirov, Ryan Cotterell, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sebastian J Mielke, Arya McCarthy, Sandra Kübler, et al. 2018. Unimorph 2.0: Universal morphology. In *LREC*.

Ling Liu and Lingshuang Jack Mao. 2016. Morphological reinflection with conditional random fields and unsupervised features. In *SIGMORPHON*.

Minh-Thang Luong and Christopher D Manning. 2015. Stanford neural machine translation systems for spoken language domains. In *IWSLT*.

Peter Makarov, Tatiana Ruzsics, and Simon Clematide. 2017. Align and copy: UZH at SIGMORPHON 2017 shared task for morphological reinflection. In *CoNLL–SIGMORPHON*.

Garrett Nicolai, Colin Cherry, and Grzegorz Kondrak. 2015. Inflection generation as discriminative string transduction. In *NAACL-HLT*.

Sergiu Nisioi, Sanja Stajner, Simone Paolo Ponzetto, Paolo Rosso, and Heiner Stuckenschmidt. 2017. Exploring neural text simplification models. In *ACL*.

Ilya Sutskever, Oriol Vinyals, and Quoc VV Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*.

John Sylak-Glassman, Christo Kirov, and David Yarowsky. 2016. Remote elicitation of inflectional paradigms to seed morphological analysis in low-resource languages. In *LREC*.

Aleš Tamchyna, Marion Weller-Di Marco, and Alexander Fraser. 2017. Modeling target-side inflection in neural machine translation. In *WMT*.

Vladimir N. Vapnik. 1998. *Statistical Learning Theory*. John Wiley.

Oriol Vinyals, Łukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. 2015. Grammar as a foreign language. In *NIPS*.

Ziang Xie, Anand Avati, Naveen Arivazhagan, Dan Jurafsky, and Andrew Y Ng. 2016. Neural language correction with character-based attention. *arXiv preprint arXiv:1603.09727*.

Wenpeng Yin, Hinrich Schütze, Bing Xiang, and Bowen Zhou. 2016. ABCNN: Attention-based convolutional neural network for modeling sentence pairs. *TACL*, 4:259–272.

Matthew D Zeiler. 2012. ADADELTA: An adaptive learning rate method. *arXiv:1212.5701*.

Chunting Zhou and Graham Neubig. 2017. Multi-space variational encoder-decoders for semi-supervised labeled sequence transduction. In *ACL*.