Inflection Generation as Discriminative String Transduction

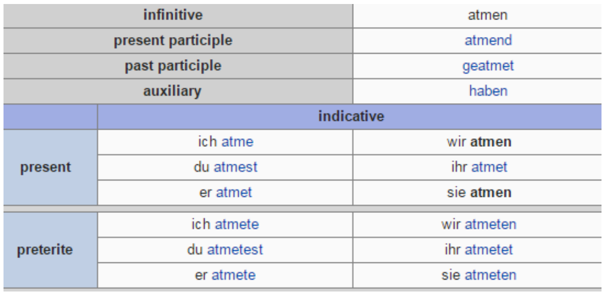
Garrett Nicolai[†] **Colin Cherry**[‡] **Grzegorz Kondrak**[†]

[†]Department of Computing Science
University of Alberta
Edmonton, AB, T6G 2E8, Canada
{nicolai,gkondrak}@ualberta.ca

[‡]National Research Council Canada
1200 Montreal Road
Ottawa, ON, K1A 0R6, Canada
Colin.Cherry@nrc-cnrc.gc.ca

Abstract

We approach the task of morphological inflection generation as discriminative string transduction. Our supervised system learns to generate word-forms from lemmas accompanied by morphological tags, and refines them by referring to the other forms within a paradigm. Results of experiments on six diverse languages with varying amounts of training data demonstrate that our approach improves the state of the art in terms of predicting inflected word-forms.



infinitive		atmen
present participle		atmend
past participle		geatmet
auxiliary		haben
	indicative	
present	ich atme	wir **atmen**
	du atmest	ihr atmet
	er atmet	sie **atmen**
preterite	ich atmete	wir atmeten
	du atmetest	ihr atmetet
	er atmete	sie atmeten

Figure 1: A partial inflection table for the German verb *atmen* “to breathe” in Wiktionary.

1 Introduction

Word-forms that correspond to the same lemma can be viewed as paradigmatically related instantiations of the lemma. For example, *take, takes, taking, took,* and *taken* are the word-forms of the lemma *take*. Many languages have complex morphology with dozens of different word-forms for any given lemma: verbs inflect for tense, mood, and person; nouns can vary depending on their role in a sentence, and adjectives agree with the nouns that they modify. For such languages, many forms will not be attested even in a large corpus. However, different lemmas often exhibit the same inflectional patterns, called *paradigms*, which are based on phonological, semantic, or morphological criteria. The paradigm of a given lemma can be identified and used to generate unseen forms.

Inflection prediction has the potential to improve Statistical Machine Translation (SMT) into morphologically complex languages. In order to address data sparsity in the training bitext, Clifton and Sarkar (2011) and Fraser et al. (2012) reduce diverse inflected forms in the target language into the corresponding base forms, or lemmas. At test time, they predict an abstract inflection tag for each translated lemma, which is then transformed into a proper word-form. Unfortunately, hand-crafted morphological generators such as the ones that they use for this purpose are available only for a small number of languages, and are expensive to create from scratch. The supervised inflection generation models that we investigate in this paper can instead be trained on publicly available inflection tables.

The task of an inflection generator is to produce an inflected form given a base-form (e.g., an infinitive) and desired inflection, which can be specified as an abstract inflectional tag. The generator is trained on a number of inflection tables, such as the one in Figure 1, which enumerate inflection forms for a given lemma. At test time, the generator predicts inflections for previously unseen base-forms. For example, given the input *atmen* + *1SIA*, where the tag stands for “first person singular indicative preterite,” it should output *atmete*.

Recently, Durrett and DeNero (2013) and Ahlberg et al. (2014) have proposed to model inflection generation as a two-stage process: an input base-form is first matched with rules corresponding to a paradigm seen during training, which is then used to generate all inflections for that base-form simultaneously. Although their methods are quite different, both systems account for paradigm-wide regularities by creating rules that span all inflections within a paradigm. We analyze both approaches in greater detail in Section 2.

In this paper, we approach the task of supervised inflection generation as discriminative string transduction, in which character-level operations are applied to transform a lemma concatenated with an inflection tag into the correct surface word-form. We carefully model the transformations carried out for a single inflection, taking into account source characters surrounding a rule, rule sequence patterns, and the shape of the resulting inflected word. To take advantage of paradigmatic regularities, we perform a subsequent reranking of the top n word-forms produced by the transducer. In the reranking model, soft constraints capture similarities between different inflection slots within a table. Where previous work leveraged large, rigid rules to span paradigms, our work is characterized by small, flexible rules that can be applied to any inflection, with features determining what rule sequence works best for each pairing of a base-form with an inflection.

Since our target application is machine translation, we focus on maximizing inflection form accuracy, rather than complete table accuracy. Unlike previous work, which aims at learning linguistically-correct paradigms from crowd-sourced data, our approach is designed to be robust with respect to incomplete and noisy training data, which could be extracted from digital lexicons and annotated corpora. We conduct a series of experiments which demonstrate that our method can accurately learn complex morphological rules in languages with varying levels of morphological complexity. In each experiment we either match or improve over the state of the art reported in previous work. In addition to providing a detailed comparison of the available inflection prediction systems, we also contribute four new inflection datasets composed of Dutch and French verbs, and Czech verbs and nouns, which are made available for future research.

2 Inflection generation

Durrett and DeNero (2013) formulate the specific task of supervised generation of inflected forms for a given base-form based on a large number of training inflection tables, while Ahlberg et al. (2014) test their alternative method on the same Wiktionary dataset. In this section, we compare their work to our approach with respect to the following three subtasks:

1. character-wise alignment of the word-forms in an inflection table (Section 2.1),
2. extraction of rules from aligned forms (2.2),
3. matching of rules to new base-forms (2.3).

2.1 Table alignment

The first step in supervised paradigm learning is the alignment of related inflected forms in a table. Though technically a multiple-alignment problem, this can also be addressed by aligning each inflected form to a base-form. Durrett & DeNero do exactly this, aligning each inflection to the base with a paradigm-aware, position-dependent edit distance. Ahlberg et al. use finite-state-automata to implement a multiple longest-common-subsequence (LCS) alignment, avoiding the use of an explicit base-form. Both systems leverage the intuition that character alignment is mostly a problem of aligning those characters that remain unchanged throughout the inflection table.

Our alignment approach differs from previous work in that we use an EM-driven, many-to-many aligner. Instead of focusing on unchanged characters within a single paradigm, we look for small multi-character operations that have statistical support across all paradigms. This includes operations that simply copy their source into the target, leaving the characters unchanged.

2.2 Rule extraction

The second step involves transforming the character alignments into inflection rules. Both previous efforts begin addressing this problem in the same way: by finding maximal, contiguous spans of changed characters, in the base-form for Durrett & DeNero, and in the aligned word-forms for Ahlberg et al. Given those spans, the two methods diverge quite substantially. Durrett & DeNero extract a rule for

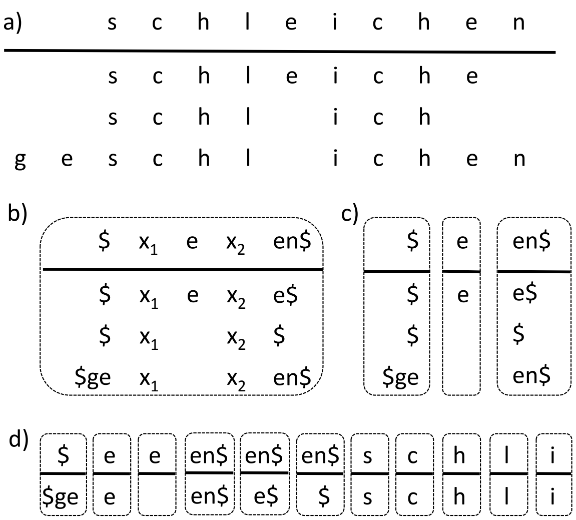


Figure 2: Competing strategies for rule extraction: (a) an aligned table; (b) a table-level rule; (c) vertical rules; (d) atomic rules. $ is a word boundary marker.

each *changed* span, with the rule specifying transformations to perform for each inflection. Ahlberg et al. instead replace each *unchanged* span with a variable, creating a single rule that specifies complete inflections for the entire table. The latter approach creates larger rules, which are easier to interpret for a linguist, but are less flexible, and restrict information sharing across paradigms.

We move in the opposite direction by extracting a rule for each minimal, multi-character transformation identified by our aligner, with no hard constraint on what rules travel together across different inflections. We attempt to learn atomic character transformations, which extends the flexibility of our rules at the cost of reduced interpretability.

The differences in rule granularity are illustrated on the German verb *schleichen* "to sneak" in Figure 2. The single rule of Ahlberg et al. comprises three vertical rules of Durrett & DeNero, which in turn correspond to eleven atomic rules in our system. Note that this is a simplification, as alignments and word boundary markers vary across the three systems.

2.3 Rule selection

The final component of an inflection generation system is a mechanism to determine what rules to apply to a new base-form, in order to generate the inflected forms. The strongest signal for this task comes from learning how the training base-forms use the rules. With their highly restrictive rules, Ahlberg et al. can afford a simple scheme, keeping an index that associates rules with base-forms, and employing a longest suffix match against this index to assign rules to new base-forms. They also use the corpus frequency of the inflections that would be created by their rules as a rule-selection feature. Durrett & DeNero have much more freedom, both in what rules can be used together and in where each rule can be applied. Therefore, they employ a more complex semi-Markov model to assign rules to spans of the base-form, with features characterizing the n-gram character context surrounding the source side of each rule.

Since our rules provide even greater flexibility, we model rule application very carefully. Like Durrett & DeNero, we employ a discriminative semi-Markov model that considers source character context, and like Ahlberg et al., we use a corpus to reevaluate predictions. In addition, we model rule sequences, and the character-shape of the resulting inflected form. Note that our rules are much more general than those of our predecessors, which makes it easy to get statistical support for these additional features. Finally, since our rules are not bound by paradigm structure, we employ a reranking step to account for intra-paradigm regularities.

3 Discriminative Transduction

In this section, we describe the details of our approach, including the affix representation, the string alignment and transduction, and the paradigm reranking.

3.1 Affix representation

Our inflection generation engine is a discriminative semi-Markov model, similar to a monotonic phrase-based decoder from machine translation (Zens and Ney, 2004). This system cannot insert characters, except as a part of a phrasal substitution, so when inflecting a base form, we add an abstract affix representation to both provide an insertion site and to indicate the desired inflection.

Abstract tags are separated from their lemmas with a single '+' character. Marking the morpheme boundary in such a way allows the transducer to generalize the context of a morpheme boundary. For example, the third person singular indicative present of the verb *atmen* is represented as *atmen+3SIE*. We use readable tags throughout this paper, but they are presented to the transducer as indivisible units; it cannot translate them character-by-character.

German and Dutch past participles, as well as several Czech inflections, are formed by circumfixation, a special process of simultaneous prefixation and suffixation. We represent such inflections with separate copies of the circumfix tag before and after the lemma. For example, the past participle *gebracht* "brought" is represented as *PPL+bringen+PPL*. In the absence of language-specific information regarding the set of inflections that involve circumfixation, the system can learn to transduce particular affixes into empty strings.

During development, we experimented with an alternative method, in which affixes are represented by a default allomorph. Allomorphic representations have the potential advantage of reducing the complexity of transductions by the virtue of being similar to the correct form of the affix. However, we found that allomorphic affixes tend to obfuscate differences between distinct inflections, so we decided to employ abstract tags instead.

3.2 String transduction

We perform string transduction adapting the tool DIRECTL+, originally designed for grapheme-to-phoneme conversion (Jiampojamarn et al., 2010). DIRECTL+ is a feature-rich, discriminative character transducer, which searches for a model-optimal sequence of character transformation rules for its input. The core of the engine is a dynamic programming algorithm capable of transducing many consecutive characters in a single operation, also known as a semi-Markov model. Using a structured version of the MIRA algorithm (McDonald et al., 2005), training attempts to assign weights to each feature so that its linear model separates the gold-standard derivation from all others in its search space.

DIRECTL+ uses a number of feature templates to assess the quality of a rule: source context, target n-gram, and joint n-gram features. Context features conjoin the rule with indicators for all source character n-grams within a fixed window of where the rule is being applied. Target n-grams provide indicators on target character sequences, describing the shape of the target as it is being produced, and may also be conjoined with our source context features. Joint n-grams build indicators on rule sequences, combining source and target context, and memorizing frequently-used rule patterns. Durrett & DeNero also use source context features, but we are the first group to account for features that consider rule sequences or target word shape.

Following Toutanova and Cherry (2009), we modify the out-of-the-box version of DIRECTL+ by implementing an abstract copy feature that indicates when a rule simply copies its source characters into the target, e.g. $p \rightarrow p$. The copy feature has the effect of biasing the transducer towards preserving the base-form within the inflected form.

In addition to the general model that is trained on all inflected word-forms, we derive tag-specific models for each type of inflection. Development experiments showed the general model to be slightly more accurate overall, but we use both types of models in our reranker.

3.3 String alignment

DIRECTL+ training requires a set of aligned pairs of source and target strings. The alignments account for every input and output character without the use of insertion. Derivations that transform the input substrings into the desired output substrings are then extracted from the alignments.

We induce the alignments by adapting the M2M aligner of (Jiampojamarn et al., 2007), which uses Expectation-Maximization to maximize the joint likelihood of its input under a pairwise alignment scheme. Previous work creates alignments based upon entire inflection tables, while ours considers each inflection paired with its base form independently. M2M goes beyond linking single characters by aligning entire substrings instead. In practice, the base-form serves as a pivot for the entire inflection table, leading to consistent multiple alignments.

We modify the M2M aligner to differentiate between stems and affixes. The alignments between stem letters rarely require more than a 2-2 alignment. A single tag, however, must align to an entire affix, which may be composed of four or more letters. The distinction allows us to set different substring length limits for the two types.

In order to encourage alignments between identical letters, we augment the training set by pairing each inflected form with itself. In addition, we modify the aligner to generalize the identity alignments into a single operation, which corresponds to the copy feature described in Section 3.2.

3.4 Reranking

Morphological processes such as stem changes tend to be similar across different word-forms of the same lemma. In order to take advantage of such paradigmatic consistency, we perform a reranking of the n-best word-forms generated by DIRECTL+. The correct form is sometimes included in the n-best list, but with a lower score than an incorrect form. We propose to rerank such lists on the basis of features extracted from the 1-best word-forms generated for other inflection slots, the majority of which are typically correct.

We perform reranking with the Liblinear SVM (Fan et al., 2008), using the method of Joachims (2002). An initial inflection table, created to generate reranking features, is composed of 1-best predictions from the general model. For each inflection, we then generate lists of candidate forms by taking the intersection of the n-best lists from the general and the tag-specific models.

In order to generate features from our initial inflection table, we make pairwise comparisons between a prediction and each form in the initial table. We separate stems from affixes using the alignment. Our three features indicate whether the compared forms share the same stem, the same affix, and the same surface word-form, respectively. We generate a feature vector for each aligned pair of related word-forms, such as past participle vs. present participle. In addition, we include as features the confidence scores generated by both models.

Two extra features are designed to leverage a large corpus of raw text. A binary indicator feature fires if the generated form occurs in the corpus. In order to model the phonotactics of the language, we also derive a 4-gram character language model from the same corpus, and include as a feature the normalized log-likelihood of the predicted form.

Language / POS	Set	Base forms	Infl.
German Nouns	DE-N	2764	8
German Verbs	DE-V	2027	27
Spanish Verbs	ES-V	4055	57
Finnish Nouns	FI-N	6400[1]	28
Finnish Verbs	FI-V	7249	53
Dutch Verbs	NL-V	11200	9
French Verbs	FR-V	6957	48
Czech Nouns	CZ-N	21830	17
Czech Verbs	CZ-V	4435	54

Table 1: The number of base forms and inflections for each dataset.

4 Experiments

We perform five experiments that differ with respect to the amount and completeness of training data, and whether the training is performed on individual word-forms or entire inflection tables. We follow the experimental settings established by previous work, as much as possible.

The parameters of our transducer and aligner were established on a development set of German nouns and verbs, and kept fixed in all experiments. We limit stem alignments to 2-2, affix alignments to 2-4, source context to 8 characters, joint n-grams to 5 characters, and target Markov features to 2 characters.

4.1 Inflection data

We adopt the Wiktionary inflection data made available by Durrett and DeNero (2013), with the same training, development, and test splits. The development and test sets contain 200 inflection tables each. and the training sets consist of the remaining data. Table 1 shows the total number of tables in each language set. We convert their inflectional information to abstract tags for input to our transducer.

We augment the original five datasets with four new sets: Dutch verbs from the CELEX lexical database (Baayen et al., 1995), French verbs from Verbiste, an online French conjugation dictionary[2], and Czech nouns and verbs from the Prague Dependency Treebank (Böhmová et al., 2003). For each of

[1] Durrett & DeNero report 40589 forms, but only use 6000 for training, and 200 each for development and testing

[2] http://perso.b2b2c.ca/sarrazip/dev/verbiste.html

Case	Singular	Plural
Nominative	Buch	Bücher
Accusative	Buch	Bücher
Dative	Buch	Büchern
Genitive	Buches	Bücher

Table 2: All word-forms of the German noun *Buch*.

Set	DDN	Ours	10-best
DE-V	94.8	**97.5**	99.8
DE-N	88.3	**88.6**	98.6
ES-V	99.6	**99.8**	100
FI-V	97.2	**98.1**	99.9
FI-N	92.1	**93.0**	99.0
NL-V	90.5*	**96.1**	99.4
FR-V	98.8*	**99.2**	99.7

Table 3: Prediction accuracy of models trained and tested on individual inflections.

these sets, the training data is restricted to 80% of the inflection tables listed in Table 1, with 10% each for development and testing. Each lemma inflects to a finite number of forms that vary by part-of-speech and language (Table 1); German nouns inflect for number and case (Table 2), while French, Spanish, German, and Dutch verbs inflect for number, person, mood, and tense.

We extract Czech data from the Prague Dependency Treebank, which is fully annotated for morphological information. This dataset contains few complete inflection tables, with many lemmas represented by a small number of word-forms. For this reason, it is only suitable for one of our experiments, which we describe in Section 4.5.

Finnish has a morphological system that is unlike any of the Indo-European languages. There are 15 different grammatical cases for nouns and adjectives, while verbs make a number of distinctions, such as conditional vs. potential, and affirmative vs. negative. We derive separate models for two noun classes (singular and plural), and six verb classes (infinitive, conditional, potential, participle, imperative, and indicative). This is partly motivated by the number of individual training instances for Finnish, which is much larger than the other languages, but also to take advantage of the similarities within classes.

For the reranker experiments, we use the appropriate Wikipedia language dump. The number of tokens in the corpora is approximately 77M for Czech, 200M for Dutch, 6M for Finnish, 425M for French, 550M for German, and 400M for Spanish.

4.2 Individual inflections

In the first experiment, we test the accuracy of our basic model which excludes our reranker, and therefore has no access to features based on inflection tables or corpus counts. Table 3 compares our results against the Factored model of Durrett & DeNero (DDN), which also makes an independent prediction for each inflection. The numbers marked with an asterisk were not reported in the original paper, but were generated by running their publicly-available code on our new Dutch and French datasets. For the purpose of quantifying the effectiveness of our reranker, we also include the percentage of correct answers that appear in our 10-best lists.

Our basic model achieves higher accuracy on all datasets, which shows that our refined transduction features are consistently more effective than the source-context features employed by the other system. Naturally, their system, as well as the system of Ahlberg et al., is intended for whole-table scenarios, which we test next.

4.3 Complete paradigms

In this experiment, we assume the access to complete inflection tables, as well as to raw corpora. We compare our reranking system to the Joint model of Durrett & DeNero (DDN), which is trained on complete tables, and the full model of Ahlberg et al. (AFH), which is trained on complete tables, and matches forms to rules with aid of corpus counts. Again, we calculated the numbers marked with an asterisk by running the respective implementations on our new datasets.

The results of the experiment are shown in Table 4. Our reranking model outperforms the Joint model of DDN on all sets, and the full model of AFH on most verb sets. Looking across tables to Table 3, we can see that reranking improves upon our independent model on 5 out of 7 sets, and is equivalent on the remaining two sets. However, accord-

Set	DDN	AFH	Ours
DE-V	96.2	**97.9**	**97.9**
DE-N	88.9	**91.8**	89.9
ES-V	99.7	99.6	**99.9**
FI-V	96.4	96.6	**98.1**
FI-N	93.4	**93.8**	93.6
NL-V	94.4*	87.7*	**96.6**
FR-V	96.8*	98.1*	**99.2**

Table 4: Individual form accuracy of models trained on complete inflection tables.

Set	DDN	AFH	Ours
DE-V	85.0	76.5	**90.5**
DE-N	79.5	**82.0**	76.5
ES-V	95.0	98.0	**99.0**
FI-V	87.5	92.5	**94.5**
FI-N	83.5	**88.0**	82.0
NL-V	79.5*	37.7*	**82.1**
FR-V	92.1*	96.0*	**97.1**

Table 5: Complete table accuracy of models trained on complete inflection tables.

ing to single-form accuracy, neither our system nor DDN benefits too much from joint predictions. Table 5 shows the same results evaluated with respect to complete table accuracy.

4.4 Incomplete paradigms

In this experiment, we consider a scenario where, instead of complete tables, we have access to some but not all of the possible word-forms. This could occur for example if we extracted our training data from a morphologically annotated corpus. We simulate this by only including in our training tables the forms that are observed in the corresponding raw corpus. We then test our ability to predict the same test forms as in the previous experiments, regardless of whether or not they were observed in the corpus. We also allow a small held-out set of complete tables, which corresponds to the development set. For Durrett & DeNero's method, we include this held-out set in the training data, while for our system, we use it to train the reranker.

The Joint method of DDN and the methods of AFH are incapable of training on incomplete tables, and thus, we can only compare our results against the Factored model of DDN. However, unlike their Factored model, we can then still take advantage of paradigmatic and corpus information, by applying our reranker to the predictions made by our simple model.

The results are shown in Table 6, where we refer to our independent model as *Basic*, and to our reranked system as *Reranked*. The latter outperforms DDN on all sets. Furthermore, even with only partial tables available during training, reranking improves upon our independent model in every case.

4.5 Partial paradigms

We run a separate experiment for Czech, as the data is substantially less comprehensive than for the other languages. Although the number of 13.0% observed noun forms is comparable to the Finnish case, the percentages in Table 6 refer only to the training set: the test and held-out sets are complete. For Czech, the percentage includes the testing and held-out sets. Thus, the method of Durrett & DeNero and our reranker have access to less training data than in the experiment of Section 4.4.

The results of this experiment are shown in Table 7. Our Basic model outperforms DDN for both nouns and verbs, despite training on less data. However, reranking actually decreases the accuracy of our system on Czech nouns. It appears that the reranker is adversely affected by the lack of complete target paradigms. We leave the full investigation into the effectiveness of the reranker on incomplete data to future work.

4.6 Seed paradigms

Dreyer and Eisner (2011) are particularly concerned with situations involving limited training data, and approach inflection generation as a semi-supervised task. In our last experiment we follow their experimental setup, which simulates the situation where we obtain a small number of complete tables from an expert. We use the same training, development, and test splits to test our system. Due to the nature of our model, we need to set aside a hold-out set for reranking. Thus, rather than training on 50 and 100 tables, we train on 40 and 80, but compare the results

Set	% of Total	DDN	Ours	
			Basic	Reranked
DE-V	69.2	90.2	96.2	**97.9**
DE-N	92.7	88.3	88.4	**89.8**
ES-V	36.1	97.1	95.9	**99.6**
FI-V	15.6	73.8	78.7	**85.6**
FI-N	15.2	71.6	78.2	**80.4**
DU-V	50.5	89.8	94.9	**96.0**
FR-V	27.6	94.6	96.6	**98.9**

Table 6: Prediction accuracy of models trained on observed forms.

with the models trained on 50 and 100, respectively. For reranking, we use the same German corpus as in our previous experiments, but limited to the first 10M words.

The results are shown in Table 8. When trained on 50 seed tables, the accuracy of our models is comparable to both the basic model of Dreyer and Eisner (DE) and the Factored model of DDN, and matches the best system when we add reranking. When trained on 100 seed tables, our full reranking model outperforms the other models.

5 Error analysis

In this section, we analyze several types of errors made by the various systems. Non-word predictions are marked with an asterisk.

German and Dutch are closely-related languages that exhibit similar errors. Many errors involve the past participle, which is often created by circumfixation. For the German verb *verfilmen* "to film," we predict the correct *verfilmt*, while the other systems have *verfilmen**, and *geverfilmt**, respectively. DDN simply select an incorrect rule for the past participle. AFH choose paradigms through suffix analysis, which fails to account for the fact that verbs that begin with a small set of prefixes, such as *ver-*, do not take a *ge-* prefix. This type of error particularly affects the accuracy of AFH on Dutch because of a number of verbs in our test set that involve infixation for the past participle. Our system uses its source and target-side n-gram features to match these prefixes with their correct representation.

The second type of error is an over-correction by the corpus. The past participle of the verb *dimmen* is *gedimmt*, but AFH predict *dimmt**, and then change it to *dummen* with the corpus. *Dummen* is indeed a valid word in German, but unrelated to the verb *dimmen*. It is also far more common, with 181 occurrences in the corpus, compared with only 28 for *gedimmt*. Since AFH use corpus frequencies, mistakes like this can occur. Our system is trained to balance transducer confidence against a form's existence in a corpus (as opposed to log frequency), which helps it ignore the bias of common, but incorrect, forms.

Set	% of Total	DDN	Ours	
			Basic	Reranked
CZ-N	13.0	91.1	**97.7**	93.5
CZ-V	6.8	82.5	83.6	**85.8**

Table 7: Prediction accuracy of models trained on observed Czech forms.

The German verb *brennen* "to burn" has an irregular past participle: *gebrannt*. It involves both a stem vowel change and a circumfix, two processes that only rarely co-occur. AFH predict the form *brannt**, using the paradigm of the similar *bekennen*. The flexibility of DDN allows them to predict the correct form. Our basic model predicts *gebrennt**, which follows the regular pattern of applying a circumfix, while maintaining the stem vowel. The reranker is able to correct this mistake by relating it to the form *gebrannt* in the corpus, whose stem is identical to the stem of the preterite forms, which is a common paradigmatic pattern.

Our system can also over-correct, such as with the second person plural indicative preterite form for the verb *reisen*, which should be *reistet*, and which our basic model correctly predicts. The reranker, however, changes the prediction to *rist*. This is a nominal form that is observed in the corpus, while the verbal form is not.

An interesting example of a mistake made by the Factored model of DDN involves the Dutch verb *aandragen*. Their model learns that stem vowel *a* should be doubled, and that an *a* should be included as part of the suffix *-agt*, which results in an incorrect form *aandraaagt**. Thanks to the modelling of phonotactics, our model is able to correctly rule out the tripling of a vowel.

Seed Tables	DE		DDN		Ours	
	Basic	Full	Factored	Joint	Basic	Full
50	89.9	**90.9**	89.6	90.5	89.7	**90.9**
100	91.5	92.2	91.4	92.3	92.0	**92.6**

Table 8: Prediction accuracy on German verb forms after training on a small number of seed inflection tables.

Finnish errors tend to fall into one of three types. First, words that involve harmonically neutral vowels, such as "e" and "i" occasionally cause errors in vowel harmony. Second, all three systems have difficulty identifying syllable and compound boundaries, and make errors predicting vowels near boundaries. Finally, consonant gradation, which alternates consonants in open and closed syllables, causes a relatively large number of errors; for example, our system predicts **heltempien*, instead of the correct *hellempien* as the genitive singular of the comparative adjective *hellempi* "more affectionate".

6 Conclusion

We have proposed an alternative method of generating inflected word-forms which is based on discriminative string transduction and reranking. We have conducted a series of experiments on nine datasets involving six languages, including four new datasets that we created. The results demonstrate that our method is not only highly accurate, but also robust against incomplete or limited inflection data. In the future, we would like to apply our method to non-European languages, with different morphological systems. We also plan to investigate methods of extracting morphological tags from a corpus, including differentiating syncretic forms in context.

Acknowledgments

We thank Mans Hulden and Aki-Juhani Kyröläinen for their assistance in analyzing Finnish errors..

This research was supported by the Natural Sciences and Engineering Research Council of Canada, and the Alberta Innovates Technology Futures.

References

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.

Harald R. Baayen, Richard Piepenbrock, and Leon Gulikers. 1995. *The CELEX Lexical Database. Release 2 (CD-ROM)*. Linguistic Data Consortium, University of Pennsylvania, Philadelphia, Pennsylvania.

Alena Böhmová, Jan Hajič, Eva Hajičová, and Barbora Hladká. 2003. The Prague dependency treebank. In *Treebanks*, pages 103–127. Springer.

Ann Clifton and Anoop Sarkar. 2011. Combining morpheme-based machine translation with post-processing morpheme prediction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 32–42. 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 NAACL-HLT*, pages 1185–1195.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. LIBLINEAR: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874.

Alexander Fraser, Marion Weller, Aoife Cahill, and Fabienne Cap. 2012. Modeling inflection and word-formation in SMT. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 664–674. Association for Computational Linguistics.

Sittichai Jiampojamarn, Grzegorz Kondrak, and Tarek Sherif. 2007. Applying many-to-many alignments and hidden markov models to letter-to-phoneme conversion. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 372–379, Rochester, New York, April. Association for Computational Linguistics.

Sittichai Jiampojamarn, Colin Cherry, and Grzegorz Kondrak. 2010. Integrating joint n-gram features into a discriminative training framework. In *Proceedings of NAACL-2010*, Los Angeles, CA, June. Association for Computational Linguistics.

Thorsten Joachims. 2002. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142. ACM.

Ryan McDonald, Koby Crammer, and Fernando Pereira. 2005. Online large-margin training of dependency parsers. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

Kristina Toutanova and Colin Cherry. 2009. A global model for joint lemmatization and part-of-speech prediction. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 486–494. Association for Computational Linguistics.

Richard Zens and Hermann Ney. 2004. Improvements in phrase-based statistical machine translation. In *HLT-NAACL 2004: Main Proceedings*, pages 257–264, Boston, USA, May.