A Discriminative Lexicon Model for Complex Morphology

Minwoo Jeong[*]
Saarland University
66123 Saarbrücken, Germany
m.jeong@mmci.uni-saarland.de

Kristina Toutanova, Hisami Suzuki, Chris Quirk
Microsoft Research
One Microsoft Way, Redmond WA 98052 USA
{kristout, hisamis, chrisq}@microsoft.com

Abstract

This paper describes successful applications of discriminative lexicon models to the statistical machine translation (SMT) systems into morphologically complex languages. We extend the previous work on discriminatively trained lexicon models to include more contextual information in making lexical selection decisions by building a single global log-linear model of translation selection. In offline experiments, we show that the use of the expanded contextual information, including morphological and syntactic features, help better predict words in three target languages with complex morphology (Bulgarian, Czech and Korean). We also show that these improved lexical prediction models make a positive impact in the end-to-end SMT scenario from English to these languages.

1 Introduction

Statistical machine translation (SMT) aims to capture the process of translating content from one language to another in a statistical model. This has several potential advantages over rule-based approaches, including the ability to quickly build systems in new languages and domains, assuming the existence of parallel data. The question of how to model this process remains, however.

Initial models (Brown et al. 1993) treated each word as an independent unit in the channel model, leaving contextual modeling to a target language model. Subsequent approaches achieved substantial improvements by modeling over larger units such as phrases (e.g. Koehn et al. 2003). Translating multiple words with a single phrase can lead to better translation quality when such matches can be found, but have significant issues with data sparsity. This problem is exacerbated when translating with morphologically complex languages, as the number of word tokens is inflated. Furthermore, if we treat each word as an independent unit, we fail to share information between related stems. Also, the correct inflection of a word might be determined by syntactic context rather than lexical context, which may extend across phrase boundaries.

Our paper investigates discriminative lexicon models to address these problems within the context of syntax-based MT. By building discriminative, feature-rich models for selecting translations, we can capture the behavior of lemmas, affixes, and surface forms as separate parameters. Therefore we have the ability to pick the appropriate stem based on lexical context and the appropriate affix based on morphological or syntactic context. In addition, these models allow for modeling contextual information without requiring the harsh partitioning commonly used in phrase-based and syntax-based systems. Many systems first partition the input into blocks, then translate each block independently. Instead, we carve the input into its smallest non-decomposible units, and pick a translation for each unit according to not only the source side, but also any additional conditioning information. This may be helpful in language pairs without complex morphology due to its increased robustness and generality of parameters.

We are not the first to investigate discriminative lexicon models. Approaches in the past include treating lexical selection as a word sense disambiguation problem, as a re-ranking problem, or by breaking the translation into separate factors; we will review these approaches in Section 2. Our approach is novel in that we build a single global log-

[*] This research was conducted during the author's internship at Microsoft Research.

linear model of translation selection that allows parameter sharing between different source sides, and integrates this directly into translation. By having a single log-linear model, we can learn more general parameters about affix selection independent of the words involved, or learn information about the best lemma translation of a given word regardless of its context. Integrating this information directly into the decoder helps prevent potential search errors introduced by re-ranking while leading to a faster overall pipeline.

For the rest of the paper, we first give a review of the literature in Section 2. Section 3 describes our approach in more detail. We describe our experiments with several feature sets and language pairs, both with intrinsic evaluations of lexical selection and in end-to-end BLEU scores in Sections 4 and 5.

2 Related Work

Much of the work on discriminative lexicon models has focused on target word selection, under the assumption that sentence-level global information is useful in finding good translation candidates. Bangalore et al. (2007) framed the problem of selecting translated words from a target lexicon as a binary classification task, where each word in the target vocabulary included in or excluded from the translation according to its binary decision. In a similar vein, Mauser et al. (2009) integrated discriminative and trigger-based lexicon models into phrase-based Chinese-English MT system, showing an improvement of the BLEU score. The goal of both of these papers on lexicon modeling was to propose appropriate translated words, looking beyond word- and phrase-based translation pairs by including source context information as bag-of-words features and trigger-based models.

The same objective has motivated work on word sense disambiguation (WSD) for machine translation, in which different *senses* of a word in the source language are defined as its possible translations in the target language (Berger et al., 1996). Again the correct sense, and therefore the correct translation, depends on the specific meaning of the word in context. Recently there has been quite a bit of research on integrating discriminatively trained WSD systems into SMT (Cabezas and Resnik, 2005; Carpuat and Wu, 2005; Vickrey et al., 2005; Chan et al., 2007). Although their integration efforts have shown promising results, none of these works have focused on the problem of translating from morphologically poor languages into morphologically rich languages. Since such languages generally have complicated rules for generating inflections, modeling a morphologically rich lexicon and disambiguating its senses is particularly challenging.

In SMT in general, there has only been a limited amount of work applying morphological processing for translating from English to morphologically rich languages. Of those, Bojar (2007) and Avramidis and Koehn (2008) used morphological features in the factored representation of words implemented in the Moses system. However, this approach is difficult to apply when the syntax-based SMT framework is used, and it tends to focus on simple generative translation models that partition the input into chunks, along with target language models to help model fluency, rather than discriminative models for picking aspects of translation based on varying amounts of context. Other studies closely related to ours are the works on Japanese case-marker generation (Toutanova and Suzuki, 2007) and morphological inflection prediction for Russian and Arabic (Toutanova et al., 2008). They built probabilistic models for morphology generation and applied them to rescore *n*-best outputs from an SMT engine, augmenting those outputs with additional inflectional variations. In contrast, we focus on direct integration of our discriminative lexicon model with a dependency tree-to-string translation system, in which syntatic features obtained from an English parser are naturally incorporated in the first-pass SMT decoder.

3 Discriminative Lexicon Model

This section describes our discriminative lexicon model for a tree-to-string translation system. The strength of our model over a plain tree-to-string translation system is that a discriminative model can easily incorporate rich contextual features, such as neighboring words and dependency relations, which are often absent in the generative translation models.

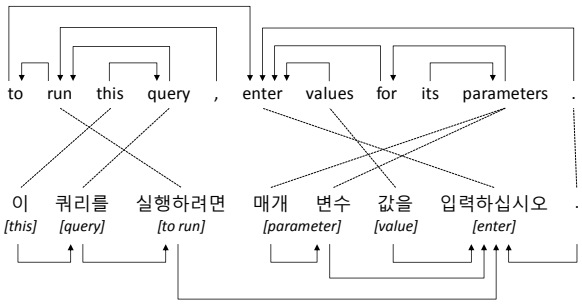

Figure 1: Aligned English-Korean sentence pair

3.1 Treelet Translation System

In this work, we build a discriminative lexicon model over a treelet translation system, a dependency tree-to-string SMT system (Quirk et al., 2005). A *treelet* is defined as a connected subgraph of a dependency tree, which acts as a unit in the channel model for decoding much like phrases in phrasal SMT models (Koehn et al., 2003). A *treelet translation pair* is a pair of source and target language treelets, which are extracted from word-aligned sentence pairs. Figure 1 shows an example of an aligned English-Korean sentence pair: a directed arc indicates a dependency relation, which is derived from an English dependency parser and is projected onto the Korean side. For instance, from the phrase "to run this query" in Figure 1, we can extract the following two treelet pairs: (to (run)) → (실행하려면), and ((this) query) → ((이) 쿼리를). By extracting all treelet translation pairs from the entire training corpus, we can build a treelet table which will be used to form the possible translations at runtime. As we will describe below, we also use this table to generate candidates for our discriminative lexicon model.

3.2 Log-linear Ranking Model

Formally, we define our discriminative lexicon model as follows. Assume we are given a source sentence $\boldsymbol{e}$, a target sentence $\boldsymbol{f}$, and an alignment relation function, ALIGN(f) which defines the set of source words aligned to each target word f. For each target word f in the target sentence, our model predicts the word f given the set of words from $\boldsymbol{e}$ aligned to f, as well as surface and syntactic context from the sentence $\boldsymbol{e}$. Each target word f is predicted independently from other words in the target sentence.

Our model is defined using a candidate generator function GEN, which computes a set of target words which are possible translations of a given set of POS-tagged aligned source words. For most words f in the target sentence, there is exactly one aligned source word, but groups of two or more source words are also possible. The GEN function is computed by considering all translation options that appear in the MT system's treelet table, with significance pruning using a log-odds significance test.

For example, given the group of tagged source words "the/DT necessary/JJ" our English-to-Bulgarian system learns a GEN function value {необходимия, необходимата, необходимият, необходимото}, which are the forms of the adjective necessary in Bulgarian that differ in gender and definiteness.

Our discriminative lexicon model estimates the conditional probability of a target word f given ALIGN(f), and additional context from $\boldsymbol{e}$. It is defined as follows:

$$P(f|\boldsymbol{e}, ALIGN(f); \boldsymbol{\lambda}) = \frac{\exp(\boldsymbol{\lambda}^{\mathsf{T}}\boldsymbol{\phi}(f, \boldsymbol{e}, ALIGN(f)))}{\sum_{f' \in \mathbf{GEN(e)}} \exp(\boldsymbol{\lambda}^{\mathsf{T}}\boldsymbol{\phi}(f', \boldsymbol{e}, ALIGN(f)))}$$

where $\boldsymbol{\phi}$ is a feature mapping and $\boldsymbol{\lambda}$ is a corresponding real-valued weight vector. We do not currently model null-aligned target words.

The features we used in this work are listed in Table 1. There are three types of features: (i) basic local context features that use the orthographic form and POS tags of source-aligned words and their neighboring contexts; (ii) dependency tree-based features, including dependency relations, modifier positions, and tree paths; (iii) morphological features of source and target words. The features are binary and compute predicates from the source context and candidate target translations. In the table below, if the feature descriptions do not mention a condition on the target word, they implicitly include the identity of the target word. If predicates on the target are explicitly mentioned, they are separated from the predicates on the source by &. Conjunctions of predicates for the source or target are denoted with a + symbol.

For the morphology-based features, we used manually curated lexicons of the source and target languages. The lexicons contain information about the lemma and the set of language-specific grammatical features of words. In case of ambiguity, we used an arbitrary analysis, namely the first analysis

Local context features (Local)
Conjunction of all AlignedWords with POS tags
AlignedWord, AlignPOSTag
PreviousWord+POS, Word-2+POS, NextWord, Next-Word+POS,Word+2+POS
PreviousWord+POS+AlignedWord
NextWord+POS+AlignedWord
Word-2+POS+AlignedWord
Word+2+POS+AlignedWord
POS -2 through +2 +AlignedWord
Bag of words: AlignedWord and Words -1 and +1 without indication of position with respect to AlignedWord
Dependency tree-based features (Deptree)
Computed with respect to AlignedWord:
Leftmost-child-Word+POS
Leftmost-child-Word+POS+AlignedWord
Parent-Word+POS+PreOrPostPosition
Parent-Word+POS+PreOrPostPosition+AlignedWord
Grandparent-Word+POS
Grandparent-Word+POS+AlignedWord
POS tags of parent,grandparent, left-child, aligned+AlignedWord
POS tags of parent,grandparent, left-child, aligned+AlignedWord+ParentWord
Morphology-based features (Morph)
GramFeat(AlignedWord)+GramFeat(Word-1) +GramFeat(Word+1)+GramFeat(Parent)+GramFeat(Grand parent) & TargetWord
GramFeat(AlignedWord)+GramFeat(Word-1) +GramFeat(Word+1)+GramFeat(Parent)+GramFeat(Grand parent)+AlignedWord & TargetWord
GramFeat(AlignedWord)+GramFeat(Word-1) +GramFeat(Word+1)+GramFeat(Parent)+GramFeat(Grand parent) & GramFeat(Target)
GramFeat(AlignedWord)+GramFeat(Word-1) +GramFeat(Word+1)+GramFeat(Parent)+GramFeat(Grand parent)+AlignedWord & GramFeat(Target)
Lemma(AlignedWord) & Lemma(Target)
Lemma(Word-1) & Lemma(Target)
Lemma(Word+1) & Lemma(Target)

Table 1:Features for discriminative lexicon model

appearing in our lexicon. For the words that are not in the lexicon, we used the suffix of a word as a grammatical feature by considering the last three characters as the suffix, and the word itself as its lemma. The function GramFeat(w) in Table 1 denotes a conjunction of all values of grammatical features as well as the POS tag. For Bulgarian, we created the lexicon from the Multext-East (Version 3) data (Erjavec, 2004), which contains about 41K distinct surface word forms and 23K lemmas. Our Czech lexicon was created using the training and development section of the CzEng corpus data (Bojar and Žabokrtský, 2009), resulting in about 800K distinct word forms and 431K lemmas. For English, we used the lexicon of the dependency parser, which includes about 150K unique word forms and 72K lemmas. [1]

	Train	Dev	Test	Random/ Oracle Acc@1
Bulgarian	345K (4.4M)	2K (51K)	3K (72K)	5.8 / 96.2
Czech	138K (2.5M)	2K (39K)	3K (60K)	7.8 / 90.5
Korean	2.8M (24M)	2K (15K)	3K (25K)	6.0 / 95.7

Table 2: Parallel data for discriminative lexicon training statistics: # sentence pairs (extracted example size in parentheses)

The main difficulty in learning our log-linear ranking models is scalability. Training with a large amount of parallel data is the norm in modern SMT systems; therefore, our model needs to accommodate a large set of training examples and features. To do this, we used an online learning approach, namely stochastic gradient descent (SGD) with L1-regularization. In this learning procedure, each example is evaluated sequentially for parameter updates so that the learning algorithm requires a minimal memory footprint. In addition, using an L1-regularizer in the log-linear model has the desirable property of reducing the number of parameters (Gao et al., 2007). We adopt an efficient gradient calculation method for L1-regularized log-linear model proposed in (Langford et al., 2009; Tsuruoka et al., 2009). Two hyper-parameters, learning rate and regularization prior, are determined using a development set in our experiments.

The training time depends on the number of features of the model. On the biggest Korean dataset, using a single CPU, training took about 12 hours per iteration for the local feature set, and about 24 hours per iteration for the local+deptree+morph feature set. About 2 to 4 iterations were sufficient for good performance. The independent models can be straightforwardly parallelized on thousands of CPUs and can thus be much faster. We are currently working on parallelizing the single-global model; batch optimization may be effective.

4 Experiments: Word Translation Task

4.1 Data and Settings

[1] Due to the large training data size, we have not yet completed the Korean experiment that uses morphological features.

In order to test our hypothesis that the discriminative method can take advantage of contextual features, we first evaluated our model on a word translation task, which is a simpler task than the end-to-end MT task. In this setting, our goal is to predict the target word for each target word position, given the aligned source word(s) and its context, compared with the reference word selection. Table 2 shows the data we used for testing our models in three target languages. This data is a subset of the data used for MT system training, which is described in detail in Section 5 in Table 7. We used 2K sentences for development and 3K sentences for testing for each language. In the table we report the number of sentence pairs in each set, as well as the number of examples available for the model, as there is a training/test example for each target word. The size of the GEN set for each source word group defines the accuracy at random of our model. Because we do some pruning in the definition of the GEN function, which is in addition to pruning performed by the MT system, we have an upper bound (oracle) accuracy of less than 100%. The random and oracle accuracies are also included in the table.

4.2 Results

Tables 3 through 5 summarize the result of applying our model to this task. In addition to the random baseline, we compare the results of our models with another baseline: a model in which the only feature type is the identity of the aligned source group of words with their POS tags. As evaluation metrics, we used the accuracy at the top k outputs (Acc@k for k=1,3), and the mean reciprocal rank of the correct target candidate (MRR).

The tables show the performance of the baseline model (labeled *source*), the model that additionally uses the local context features (+*local*), dependency tree features (+*deptree*) and morphological features (+*morph*). The feature additions are cumulative in this order; the number of features for each model (also cumulative) is also found in the tables. The number of features roughly doubles from the local to local+deptree and local+deptree to local+deptree+morph feature sets. Depending on the size of the training set, a model using the full feature set has between 30 and 100 million features. According to all of these performance metrics, our model performs substantially better than the two baselines. In addition, we see that the features based on the syntactic analysis of the source and the morphological features bring significant additional gains, as compared to a model using only local word and POS tag context.

Model	Acc@1	Acc@3	MRR
source	56.76	77.91	67.74
+local	66.47	83.92	75.38
+deptree	66.70	83.91	75.54
+morph	67.03	84.17	75.77

Table 3: Results on word translation task for English-Bulgarian

Model	Acc@1	Acc@3	MRR
source	43.74	63.08	54.03
+local	47.91	66.46	57.69
+deptree	48.36	66.79	58.02
+morph	48.69	67.21	58.38

Table 4: Results on word translation task for English-Czech

Model	Acc@1	Acc@3	MRR
source	59.16	77.98	69.25
+local	70.05	85.03	77.76
+deptree	71.05	85.75	78.55

Table 5: Results on word translation task for English-Korean

4.3 Single Log-linear Model versus Independent Models

In previous work on word sense disambiguation for MT, the disambiguation problem is solved independently for each distinct source word group (or phrase). In contrast, our model makes use of features which are shared among source word groups. This makes it possible to learn generalized knowledge: for example, if there is a definite article in the local neighborhood of the source word, the target word is more likely to be inflected for definiteness. Such information is learned from all training examples and not only the ones limited to a particular source word group. Other kinds of shared features are the ones that predict the target word from a bag-of-words representation of the context around the source word, without any indication of the particular aligned source word, which is expected to add some robustness to alignment errors; and features which share information across source groups sharing one or more words in common.

Model	Acc@1	MRR
Source	56.2	67.10
local-g	65.66	74.21
local-i	65.08	73.63
+deptree-g	65.87	74.43
+deptree-i	65.11	73.67
+morph-g	66.11	74.69
+morph-i	65.34	74.00

Table 6: Results on global vs. independent log-linear models on English-Bulgarian

To test the impact of having a single global model versus having separate models for each source group, we performed an experiment testing the two model set-ups using the same set of feature templates. We can train separate models for each source word group either by estimating and doing inference with thousands of models trained separately, or by still training a single log-linear model but appending the identity of the source word group to each feature template. We used the latter implementation.

Table 6 lists the 1-best accuracy and MRR of models using different templates on a smaller English-Bulgarian training/test split with 30K training sentences and 3K test sentences. Lines labeled with –g denote runs of the global model. Lines labeled with –i denote runs of a model with independently trained features. As seen from the table, the gains due to training a single global model are significant and consistent across feature sets.

5 Integrating the Discriminative Lexicon Model with SMT

5.1 Baseline System

We integrated the discriminative lexicon models explained so far in the framework of tree-to-string SMT system (Quirk et al., 2005) to measure their contribution in an end-to-end SMT scenario. In the treelet translation model, decoding is a process of syntax-directed translation. First the input sentence is broken into tokens, each token is assigned a part-of-speech tag, and finally the sentence is parsed to produce a single dependency tree (or a forest of dependency trees; though in this work we use only the one-best parse). Next, we find all treelet translation pairs from the treelet table where the source side of the treelet matches some contiguous subgraph of the dependency parse. These treelet translation pairs are combined with order templates to produce sentence-specific tree transduction rules. Finally the decoder searches for the best translation of the input according to these transduction rules using a CKY-style decoder. Let us explore the baseline method in a bit more detail.

From the word-aligned training corpus, we extract two distinct types of translation information. The first type of unit is the *treelet translation pair*. Here, a *treelet* is a connected subgraph of the dependency tree. Therefore, a treelet translation pair is a source and target treelet pair that is consistent with the word alignment. All such pairs are gathered from each training sentence pair, aggregated, and counted to form a set of possible translation units.

Lexical translations are provided by these pairs, as is the relative ordering of the target words within the pair. However, the ordering of target words from different treelets is not well specified. To order separate treelets, we rely on a set of *order templates*, which are single level tree transduction rules relying on part of speech tags. In English to Korean translation, for instance, a common order template specifies that a verb with a right subtree headed by a noun should be reordered to place the noun first: ($\star$/V (x1:N)) → ((x1) $\star$).

At translation time, then, we first look up all treelet translation pairs that match the input. Next we construct a sentence-specific transduction rule from every treelet. Starting with a single treelet, we visit every source word to find an order template that determines the target ordering of all its children. Note that this template must be consistent with the input tree and with the ordering of the target words in the treelet. Unifying the treelet translation pair with the set of order templates produces a transduction rule that specifies both the lexical translations and the relative ordering of all children.

As an example, consider the input sentence "Run this query." This can be parsed as the tree (*run*/Verb ((*this*/Det) *query*/Noun))), parentheses are used to denote tree structure. Say we also have the following treelet translation pairs:

run → 실행하시오
query → 쿼리를
this → 이

Also we have the following order templates, which specify post-modifying nouns become premodifiers, and premodifying determiners remain premodifiers:

($\star_1$/Verb (x1:$\star$/Noun)) → ((x1) $\star_1$)
((x1:$\star$/Det) $\star_1$/Noun) → ((x1) $\star_1$)

These treelet translation pairs and order templates can be combined to form the following transduction rules:

(run/Verb (x1:⋆/Noun)) → ((x1) 실행하시오)

((x1:⋆/Det) query/Noun) → ((x1) 쿼리를)

(this) → (이)

Given rules of this form, finding the best translation is a matter of searching for the best derivation according to a sentence specific grammar. In the absence of a language model or other models that score based on context, we may simply use a standard parsing algorithm such as CKY to find the best derivation. In the presence of context sensitive features, we resort to the approximate search technique of cube pruning (Chiang, 2007).

The baseline treelet system scores candidates with a weighted linear combination of features, with weights trained by Minimum Error Rate Training (Och, 2003), hereafter referred to as MERT. The baseline set of feature functions are: log probabilities of the source treelet given the target treelet and vice versa (maximum likelihood estimates); forward and backward lexical weighting; a target language log probability from a Kneser-Ney smoothed language model; word and phrase count feature functions, and order template log probabilities (maximum likelihood estimates).

For all features except the language model, we may pre-compute the weighted score of each model. Then, during decoding, we only need to update the score of the language model as larger hypothesis are composed together. This allows for more efficient search: we pay the computational cost of computing a context-independent feature value only once, regardless of how often that rule is applied during translation. Here, context-independent implies that a feature's value does not change regardless of its greater *target* context. The source is fixed at this point during search, therefore we may safely compute source-dependent functions prior to decoding.

5.2 Integration of Discriminative Lexicon Model

Two new features based on the discriminative lexicon model were added to the baseline system: (i) for all target words that were in the GEN set for their aligned source words, the log-probability of the target word according to the model, and (ii) for all target words that were not seen with their aligned source words, an indicator of this event. Since both feature values can be computed independently of the target context, we compute the feature values once for each transduction rule instead of during the search, saving computation cycles. These features participate in the search and optimization as would any other feature: as stated above, we use MERT over a development set to find the feature weights that optimize the BLEU score of this development set.

	Train	MERT Dev	Test
Bulgarian	350K (28.8)	500 (23.9)	1000 (18.4)
Czech	143K (23.4)	500 (24.6)	1026 (21.3)
Korean	2.8M (15.4)	500 (19.0)	1000 (15.2)

Table 7: Data for MT experiments: number of sentence pairs (average number of English words per sentence)

5.3 Data

Our full datasets for the three languages are described in Table 7. They consist of training sets (train), dev sets for tuning the weights of the MT component models (MERT dev), and final test sets for evaluating the translation performance (test). As mentioned earlier, the training data for the discriminative lexicon models is a subset of the MT training data: it includes almost all MT training data, excluding 5K sentences for development and testing.

For Bulgarian, we used a 300K sentence subset of the JRC-Aquis corpus (Steinberger et al., 2006) for training. The MERT development set and the test sets are from a variety of sources from more general domains and are thus out-of-domain with respect to the training set. The MT system for Bulgarian used a maximum treelet size of 4.

For Czech we used data from the EACL 2009 fourth workshop on SMT. Our test set is news-dev2009b, our MERT dev set is 500 sentences from news-dev2009a, and our training set is the union of the news-commentary09 corpus and the news portion of the CzEng corpus data (Bojar and Žabokrtský, 2009) from sections 0 to 7, with duplicates removed. The MT system used a maximum treelet size of 7.

For Korean we used data from a technical domain of software manuals. We used 2.8 million sentence pairs for training, 500 sentences as a de-

	MERT Dev		Test	
	Baseline	+DL	Baseline	+DL
Bulgarian	21.78	22.44	19.00	19.63
Czech	11.87	12.45	11.90	12.38
Korean	61.23	62.04	59.04	59.52

Table 8: Results (BLEU) on MT task

velopment set for MERT, and 1K sentences for a test set. The MT system used a maximum treelet size of 4.

5.4 Results and Discussion

BLEU score results for the baseline model (Baseline) and the model including the discriminative lexicon model (+DL) are shown in Table 8. We used case-insensitive four-gram BLEU. The table shows that the use of discriminative lexicon models improve the end-to-end MT results in all languages we experimented.

Although the BLEU gains on each test set are not gigantic, we find it very encouraging that the system can achieve considerable gains on a broad range of datasets for the languages with rich and diverse morphology. Furthermore BLEU may not be sensitive to some of the gains. Say the baseline translation system selects a lemma different than that of the reference translation with an incorrect surface form. If the system with the discriminative lexicon model succeeds in picking the correct surface form but does not change the stem, the BLEU score will not improve. Given that we see BLEU improvements even in the presence of such issues, we think the model is very likely to be helpful.

In terms of intrinsic accuracy, we find that across all language pairs the set of surface features is the single most important feature set to include. After some reflection we were not overly surprised by this result: as in n-gram language modeling, the neighboring words provide powerful and robust contextual indicators, especially because they are not subject to the error rate of any parser or morphological analyzer. That said, we found that including additional features about syntactic and morphological information produced consistent and notable gains in intrinsic accuracy; it would be interesting to note the impact of parser accuracy here. Including features from multiple parses or from the collapsed forest might lead to improvements by softening the impact of error rate.

One of the original motivations for the research in this paper was to improve the generation of complex morphology via better translation selection. An automatic assessment of the impact to the morphology generation is difficult – as mentioned above, the BLEU metric may not capture improvements in this regard. One possible automatic evaluation, however, is to measure the BLEU improvements in fully inflected forms as compared with the gains in the selection of word lemmas only. In other words, if the discriminative lexicon model helps select the correct word stems (as in WSD) rather than the correct inflection of the stems, the BLEU gain is expected to be larger when we lemmatize both the reference and the system output. We therefore computed the BLEU scores in the lemmatized versions of the reference and system output, where lemmatization was provided by the lexicons described in Section 3.2.

The results are shown in Table 9. They show different patterns for the three languages: the BLEU improvement is larger for the lemmatized version in Czech, while it is smaller in Bulgarian. In Korean, they are about the same. This suggests that in Bulgarian, much of the gains in the translations are due to picking the correct morphological ending, which we find very encouraging. A manual inspection of the English-Bulgarian results suggested that about half of the improvements were in grammar and half were in word sense disambiguation. One class of morphological ending errors that were reduced by the model was the errors in definiteness of nouns and adjectives: definiteness is marked by a suffix in Bulgarian and can be well-predicted by surface and syntactic context in the source sentence. For example, if a source noun is preceded the “the”, this makes it more likely that the Bulgarian corresponding word will have a definite suffix.

For Czech, on the other hand, the lemmatized gains are much greater. This may be due to the fact that Czech morphology is quite complex: it has many more surface word forms per lemma, due to both a larger case system and a larger number of morphological features. Therefore it may be difficult to get all of these features correct at once, which is required to improve the fully inflected BLEU number. In the future, it may be interesting to inspect specific attributes of the morphology, to see if our model is more effective in predicting individual morphological features such as case, number and gender.

Finally in Korean, our results indicate that the improvements include both in the selection of

	Baseline	DiscLex	Diff
Bulgarian			
Fully inflected	19.00	19.63	+0.63
Lemmatized	22.61	22.78	+0.17
Czech			
Fully inflected	11.90	12.38	+0.48
Lemmatized	16.18	17.11	+0.93
Korean			
Fully inflected	59.04	59.52	+0.48
Lemmatized	61.34	61.77	+0.43

Table 9: BLEU results for fully inflected forms vs. lemmas (using the test data in Table 8)

lemmas as well as in predicting the correct inflection. Since our current model for Korean does not use morphological features, it is also interesting to see the difference when such features are used in the model in the future.

6 Conclusion and Future Work

We have presented a discriminative lexicon model that improves over the prior state of the art in two substantial ways. First, we focus on models of morphologically rich languages, using feature rich methods to capture contextual and syntactic dependencies in translation. Especially in languages where the syntactic or thematic roles of words are indicated with morphological endings (such as Czech and Korean), we find the addition of features from the source language dependency tree helpful. Although this result is perhaps to be expected, we were gratified to find that this relatively simple linear model could effectively capture some of these generalizations and improve the accuracy of translation selection.

Second, we found that using a single global model, as opposed to a host of independently trained per-word classifiers, leads to additional improvements in translation quality. Therefore the model appears to learn some generalizations that are portable across input words. This result is particularly encouraging as we hope to scale these models to cover very broad domains – the portability of classifiers will be very useful.

Along these lines, our treelet table and GEN function are currently limited to the forms of data seen at training time. For common words this is no particular limitation: given even a moderately sized corpus, we are likely to see all the morphological forms in heavy use. As we find words toward the tail of the distribution, though, we are quite likely to encounter words for which only a very few forms were seen in training data. Of course this issue is only exacerbated as the richness of the morphology increases. Thus we are investigating methods of expanding both the treelet table and the GEN function to cover forms not seen in the parallel data, perhaps confirming these forms using monolingual data. We believe that more substantial gains will be achieved once this expansion is incorporated into the mainline system.

We have also not fully exploited the power of our feature-rich approach. Using features that depend solely on the source side has computational advantages, but it is likely that target-side features will garner further improvements. Especially relevant to morphologically rich languages is that phenomena such as head-modifier agreement are most easily modeled in terms of target contextual information. We plan to include some of these features in future work.

Using an online L1-regularized training method has its advantages. Learning sparse vectors is very helpful for training, as it reduces the size and computational requirements of the runtime system. It is also easy to use for single processor systems. However, for greater scalability, we would prefer to use methods that scale to take advantage of more processors, either within the same machine with low-latency access to memory, or in a cluster scenario where communicating parameter updates between threads is potentially very expensive. We have performed some initial investigation in this area, and hope to achieve significant gains in training speed.

Acknowledgements

We thank the reviewers of the AMTA as well as SSST-4 workshop program committees for their valuable comments on the earlier versions of this paper.

References

E. Avramidis, and P. Koehn. 2008. Enriching morphologically poor languages for statistical machine translation. In *Annual Meeting of the Association for Computational Linguistics*, pages 763-770.

S. Bangalore, P. Haffner, and S. Kanthak. 2007. Statistical machine translation through global lexical selection and sentence reconstruction. In *Annual Meeting of the Association of Computational Linguistics*, pages 152-159.A. Berger, S. Della Pietra, and V. Della

Pietra. 1996. A maximum entropy approach to natural language processing. In *Computational Linguistics*, 22(1).

O. Bojar. 2007. English-to-Czech Factored Machine Translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 232-239.

O. Bojar and Z. Žabokrtský. 2009. CzEng 0.9: Large Parallel Treebank with Rich Annotation. *Prague Bulletin of Mathematical Linguistics*, 92.

C. Cabezas and P. Resnik. 2005. Using WSD techniques for lexical selection in statistical machine translation. Technical report, University of Maryland.

M. Carpuat and D. Wu. 2005. Word sense disambiguation vs. statistical machine translation. In *Annual Meeting of the Association of Computational Linguistics*, pages 387-394.

David Chiang. 2007. Hierarchical phrase-based translation. In *Computational Linguistics*, 33(2):201–228.

Y. S. Chan, H. T. Ng, and D. Chiang. 2007. Word sense disambiguation improves statistical machine translation. In *Annual Meeting of the Association of Computational Linguistics*, pages 33-40.

T. Erjavec. 2004. MULTEXT-East Version 3: Multilingual morphosyntactic specifications, lexicons and corpora. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Paris.

J. Gao, G. Andrew, M. Johnson, and K. Toutanova. 2007. A comparative study of parameter estimation methods for statistical natural language processing, In *Annual Meeting of the Association for Computational Linguistics*, pages 824-831.

J. Langford, L. Li, and T. Zhang. 2009. Sparse online learning via truncated gradient, *Journal of Machine Learning Research*, 10(Mar):777-801.

A. Mauser, S. Hasan and H. Ney. 2009. Extending statistical machine translation with discriminative and trigger-Based lexicon models. In *Conf. of Empirical Methods in Natural Language Processing*, pages 210-218.

F. J. Och. 2003. Minimum Error Rate Training for Statistical Machine Translation. In *Proceedings of ACL*.

C. Quirk, A. Menezes, and C. Cherry. 2005. Dependency tree translation: Syntactically informed phrasal SMT. In *Annual Meeting of the Association for Computational Linguistics*, page 271-279.

R. Steinberger, B. Pouliquen, A. Widiger, C. Ignat, T. Erjavec, D. Tufiş and D. Varga. 2006. The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'2006)*. Genoa, Italy.

K. Toutanova, and H. Suzuki. 2007. Generating case markers in machine translation. In *Conf. of the North American Chapter of the Association for Computational Linguistics*, pages 49-56.

K. Toutanova, H. Suzuki, and A. Ruopp. 2008. Applying morphology generation models to machine translation. In *Annual Meeting of the Association for Computational Linguistics*, pages 514-522.

Y. Tsuruoka, J. Tsujii, and S. Ananiadou. 2009. Stochastic Gradient Descent Training for L1-regularized Log-linear Models with Cumulative Penalty. In *Annual Meeting of the Association for Computational Linguistics*, page 477-485.

D. Vickrey, L. Biewald, M. Teyssier, and D. Koller. 2005. Word-sense disambiguation for machine translation. In *Conf. of Empirical Methods in Natural Language Processing*, pages 771-778.