# ABSTRACT

Title of thesis: STRUCTURED LOCAL EXPONENTIAL MODELS FOR MACHINE TRANSLATION

Michael Vladimir Subotin, Master of Arts, 2010

Thesis directed by: Professor Philip Resnik
Department of Linguistics

This thesis proposes a synthesis and generalization of local exponential translation models, the subclass of feature-rich translation models which associate probability distributions with individual rewrite rules used by the translation system, such as synchronous context-free rules, or with other individual aspects of translation hypotheses such as word pairs or reordering events. Unlike other authors we use these estimates to replace the traditional phrase models and lexical scores, rather than in addition to them, thereby demonstrating that the local exponential phrase models can be regarded as a generalization of standard methods not only in theoretical but also in practical terms. We further introduce a form of local translation models that combine features associated with surface forms of rules and features associated with less specific representation – including those based on lemmas, inflections, and reordering patterns – such that surface-form estimates are recovered as a special case of the model. Crucially, the proposed approach allows estimation of parameters for the latter type of features from training sets that include multiple source phrases, thereby overcoming an important training set fragmenta-

tion problem which hampers previously proposed local translation models. These proposals are experimentally validated. Conditioning all phrase-based probabilities in a hierarchical phrase-based system on source-side contextual information produces significant performance improvements. Extending the contextually-sensitive estimates with features modeling source-side morphology and reordering patterns yields consistent additional improvements, while further experiments show significant improvements obtained from modeling observed and unobserved inflections for a morphologically rich target language.

# STRUCTURED LOCAL EXPONENTIAL MODELS FOR MACHINE TRANSLATION

by

Michael Vladimir Subotin

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Master of Arts
2010

Advisory Committee:
Professor Philip Resnik, Chair
Professor Amy Weinberg
Professor Hal Daumé III

# Acknowledgments

I would like to thank the members of my advisory committee – Philip Resnik, Amy Weinberg, and Hal Daumé III – for their time and comments, which have helped me to improve this thesis. Philip Resnik and Amy Weinberg have helped me in many other ways during my time in graduate school. This work has also benefited from discussions with and help from David Chiang, Chris Dyer, Lise Getoor, Kevin Gimpel, Adam Lopez, Nitin Madnani, Smaranda Muresan, Noah Smith, Daniel Zeman, and several anonymous reviewers.

# Table of Contents

# List of Tables

# Chapter 1

# Contributions and related work

## 1.1 Generalizing local translation models

In the field of machine translation hierarchical phrase-based systems currently produce some of the best results. These translation models are derived by purely statistical methods from parallel corpora. Several recent studies have proposed methods for extending phrase-based translation to use results of treebank parsers and other linguistic annotation tools. This thesis contributes to the development of the subclass of annotation-sensitive translation models which associate probability distributions with individual rewrite rules used by the translation system, such as paired n-grams or synchronous context-free rules, or with other individual aspects of translation hypotheses such as word pairs or reordering events. We call these models *local* to distinguish them from *global* probabilistic models that seek to estimate a single probability distribution over a set of translation hypotheses.

Local exponential translation models generalize currently used approaches. In standard translation models each rule is associated with closed-form maximum likelihood estimates computed according to a word alignment and an extraction heuristic. These estimates calculate conditional probabilities of one half of a rewrite rule given the other half, and their particular form can be viewed as a rudimentary solution to a multi-class classification problem. The idea of refining these estimates by de-

veloping more sophisticated solutions to these classification tasks is not new. Its first appearance dates back to Brown et al [7]. More recently several studies applied it to phrase-based translation. Carpuat & Wu [8, 9], working within a flat-phrase system, trained classifiers of different types to condition phrase probabilities on surrounding source-language context. Chan et al [10] developed a similar elaboration within a hierarchical phrase system, applying it only to a subset of flat phrases from its grammar. The first contribution of the present work, originally reported in [33], generalized this approach by conditioning all phrase-based probabilities in a hierarchical phrase-based system on source-side contextual information using an exponential model framework, demonstrating consistent significant improvements for 2 language pairs and 8 test sets. He et al [17] independently studied a similar elaboration of hierarchical phrase-based translation. Unlike the other authors Subotin [33] used the new estimates to replace the traditional phrase models and lexical scores, rather than in addition to them, thereby demonstrating that the local exponential phrase models can be regarded as a generalization of standard methods not only in theoretical but also in practical terms.

The view of exponential estimates as a generalization of standard translation models also motivated further elaborations. Previous extensions of local translation models were limited either to training independent classifiers associated with individual source phrases or to training independent classifiers for less specific representations of rewrite rules, such as their reordering patterns. Thus, Carpuat & Wu and Chan et al train a separate classifier for each source phrase or word pair. Conversely, Xiong et al [35] and later He et al [18] train specialized classifiers that

predict reordering patterns of target phrases without giving estimates for target phrases themselves. This thesis studies a form of local translation models that combine features associated with surface forms of rules (i.e., rules themselves) and features associated with their less specific representation (i.e., those based on lemmas, inflections, and reordering patterns), such that surface-form estimates are recovered as a special case of the model, while parameters for the latter type of features could be estimated from training sets spanning multiple source phrases. In contrast to the above-mentioned studies, where the individual models are interpolated using one weight per model, this approach permits to optimize trade-offs between these types of features on a per-feature basis while at the same time counteracting data sparsity problems resulting from excessive training set fragmentation. Extending the contextually-sensitive estimates with features modeling source-side morphology and reordering patterns yielded additional improvements, while a second series of experiments, reported for the first time in this thesis, shows significant improvements obtained from modeling observed and unobserved inflections for a morphologically rich target language.

## 1.2 Alternative training methods

Several recent studies propose alternative modeling frameworks capable of incorporating complex feature sets into phrase-based machine translation. Factored translation models [19] is a popular framework that can use information from different levels of representation in a phrase-based system, which has been explored

in several recent studies [2, 29, 36]. However, it is normally based on interpolating different relative frequency estimates, and thus cannot learn complex interactions between features of different kinds. Chiang, Marton & Resnik [13] and Chiang, Wang & Knight [14] use an alternative algorithm in place of the traditional minimum error rate training (MERT) [28], which enables them to incorporate several thousand additional features into the model, tuning their weights to increase a translation quality score. The nature and capabilities of their approach differs from local exponential models in several respects. In contrast to the methods discussed above, they optimize the additional parameter weights on a small held-out set rather than on all of the training data. Consequently, they have to compute the standard translation models in the usual way and interpolate their aggregate scores with the new features. This is fundamentally different from local exponential models, which, by virtue of using a convex objective and comparatively inexpensive computations, can be trained on the entire data set and handle feature sets that are larger by several orders of magnitude. Aside from using a larger variety of features, this also allows us to optimize the trade-off between rule features and other features on a per-feature basis, rather than forcing us to compute rule feature weights by aggregate heuristics. On the other hand, the method proposed by Chiang et al is able to use some feature types which cannot be effectively incorporated into local models. An example of such features are measures of structural similarity between translation derivations and treebank parse trees of the source sentence [13], which cannot be effectively modeled by conditional estimates for target phrases, since these features distinguish between different *source* phrases. Furthermore, the use of standard eval-

uation metrics as training objectives has been shown to improve the score, at least for the respective metrics [28]. Hence, the two approaches may be viewed as being complementary rather than in direct competition with each another.

Other authors have introduced alternative training methods that utilize the entire training set. Liang et al [22] study updating methods based on the averaged perceptron, where the features include both the relative frequency aggregate scores used in phrase-based translation and additional features such as indicator functions for part-of-speech tags. Tillmann & Zhang [34] develop an alternative method based on stochastic gradient updates closely related to the perceptron. Finally, Blunsom & Osborne [5] study a global probabilistic model based on synchronous context-free grammar and incorporating n-gram language model features, where the synchronous parse is treated a hidden variable. All of these methods in their present form present considerable practical challenges. The problem is more fundamental for proposed methods based on the perceptron, since their lack of robust feature selection leads to reduced decoding speed, thereby not only adding to the already long training times, but also introducing run-time delays that may be unacceptable in industrial applications. Hidden-variable exponential models, like that proposed by Blunsom & Osborne [5], can in principle be trained with weight-pruning regularizers, but difficulties would remain, since estimation would still involve computational costs comparable to or exceeding the costs of decoding the entire training set, repeated over multiple iterations, which are exacerbated by the non-convex form of hidden-variable likelihood. As a result, all of these authors have to carefully restrict dimensionality of their feature space and are able to report only small improvements over

a relatively weak baseline, or no improvements at all.

Thus, the framework presented here stands apart in offering capabilities that go beyond alternative approaches, both in terms of the number, and hence variety of features that can be used in practice, and in terms of the size of parallel sets that can be used to train the models, while being complementary to alternative training methods with respect to additional feature types and training objectives that the latter can use.

## 1.3 Thesis summary

The main contribution of the thesis is to address the following limitations of previously proposed local translation models:

1. In one line of previous work [8, 9, 10], context-based features are tied to particular source phrases, which prevents generalization beyond surface forms of words required for morphological and syntactic phenomena. Even if less specific features were introduced into these models, their parameters would have to be estimated separately in data sets associated with different source phrases, leading to training set fragmentation that would prevent generalizations these features were meant to capture.

2. In another line of previous work [35, 18] context-based features for reordering patterns are used in a separate classifier, trained independently from the standard phrase models. This prevents the model from learning the individual trade-offs between surface forms of the rules and reordering-based features.

Furthermore, this approach has not been extended to other phenomena, such as inflections of a morphologically rich target language, which may present additional difficulties. The use of relative frequency estimates in factored translation models can be seen as a form of this approach.

The proposed solution can be schematically represented as follows:

1. We start by identifying exponential likelihood functions that underlie the standard relative frequency phrase models and classifier-based models.

2. We then identify a more general and flexible exponential model that yields these models as special cases.

3. Finally, we apply the proposed model to different morphological and syntactic phenomena by varying its feature and normalization sets.

The rest of the thesis is organized as follows. Chapter 2 presents the baseline approach and discusses several proposed types of local exponential models. Chapter 3 presents experiments studying classifier models, lexical models, source-side morphology models, and reordering models. Chapter 4 presents experiments studying target-side morphology models. Finally, chapter 5 summarizes the main points of the thesis and outlines several possible extensions of the proposed framework.

# Chapter 2

# Local exponential translation models

## 2.1 Hierarchical phrase-based translation

We take as our starting point David Chiang's Hiero system [12], which generalizes phrase-based translation to substrings with gaps. Consider for instance the following set of context-free rules with a single non-terminal symbol:

$\langle\, A\,,\, A\,\rangle \rightarrow \langle\, A_1\, A_2\,,\, A_1\, A_2\,\rangle$

$\langle\, A\,,\, A\,\rangle \rightarrow \langle\, d'\, A_1\, idées\, A_2\,,\, A_1\, A_2\, ideas\,\rangle$

$\langle\, A\,,\, A\,\rangle \rightarrow \langle\, incolores\,,\, colorless\,\rangle$

$\langle\, A\,,\, A\,\rangle \rightarrow \langle\, vertes\,,\, green\,\rangle$

$\langle\, A\,,\, A\,\rangle \rightarrow \langle\, dorment\, A\,,\, sleep\, A\,\rangle$

$\langle\, A\,,\, A\,\rangle \rightarrow \langle\, furieusement\,,\, furiously\,\rangle$

It is one of many rule sets that would suffice to generate the English translation 1b for the French sentence 1a.

1a. *d' incolores idées vertes dorment furieusement*

1b. *colorless green ideas sleep furiously*

As shown by Chiang [12], a weighted grammar of this form can be collected and scored by simple extensions of standard methods for phrase-based translation and efficiently combined with a language model in a CKY decoder to achieve large

improvements over a state-of-the-art phrase-based system. The translation is chosen to be the target-side yield of the highest-scoring synchronous parse consistent with the source sentence. Although a variety of scores interpolated into the decision rule for phrase-based systems have been investigated over the years, only a handful have been discovered to be consistently useful, as is in our experience also the case for the hierarchical variant. Setting aside specialized components such as number translators, we concentrate on the essential sub-models[1] comprising the translation model: the phrase models and lexical models.

## 2.2 Exponential models

The computation of generalized local translation models is based on the familiar equation for conditional exponential models, which are also known as log-linear models, and whose maximum likelihood estimates are equivalent to maximum entropy models (see, e.g., [31]):

$$p(Y|X) = \frac{e^{\mathbf{w}\cdot\mathbf{f}(X,Y)}}{\sum_{Y'\in GEN(X)} e^{\mathbf{w}\cdot\mathbf{f}(X,Y')}} \tag{2.1}$$

where $\mathbf{f}(X,Y)$ is a vector of feature functions, $\mathbf{w}$ is a corresponding weight vector, so that $\mathbf{w}\cdot\mathbf{f}(X,Y) = \sum_i w_i f_i(X,Y)$, and $GEN(X)$ is a set of values corresponding to $Y$.

Maximum likelihood estimation for exponential model finds the values of

[1] To avoid confusion with features of the exponential models described below we shall use the term "model" for the terms interpolated using MERT.

weights that maximize the likelihood of the training data, or, equivalently, its logarithm:

$$LL(\mathbf{w}) = \log \prod_{m=1}^{M} p(Y_m|X_m) = \sum_{m=1}^{M} \log p(Y_m|X_m) \tag{2.2}$$

where the expressions range over all training instances $\{m\}$. Standard optimization algorithms enable us to find the unique maximum likelihood solution for exponential models as long as we can compute the value of the objective and its gradient for a given set of weights. Estimation thus also involves evaluating the standard expression for log-likelihood gradient, whose $n$-th component is shown below:

$$\frac{\partial \log LL(\mathbf{w})}{\partial w_n} = \sum_{m=1}^{M} \left[ f_n(X_m, Y_m) - \sum_{Y' \in GEN(X_m)} f_n(X_m, Y')\ p(Y'|X_m) \right] \tag{2.3}$$

Parameter estimates can be made more statistically reliable by use of regularization. The present work uses $\ell_1$ and $\ell_2$ regularization [27, 15]. The first type of regularization involves optimizing a function where the sum of absolute values of the weights multiplied by a regularization trade-off $C$ is subtracted from the log-likelihood:

$$\sum_{m=1}^{M} \log p(Y_m|X_m) - C\,|\mathbf{w}| \tag{2.4}$$

This form of regularization has the additional benefit of driving a large number of feature weights to zero, thereby reducing the computational costs of computing model predictions. The discontinuities in the absolute value function make the

optimization problem more involved, but efficient methods for its solution have been recently introduced (e.g., Andrew & Gao [1]).

In $\ell_2$ regularization the sum of *squares* of the weights, or, equivalently, the squared norm of the weight vector, is subtracted from the objective instead.

$$\sum_{m=1}^{M} \log\, p(Y_m|X_m) - C\,||\mathbf{w}||^2 \tag{2.5}$$

Unlike $\ell_1$ regularization, $\ell_2$ regularization causes relatively few feature weights to go to zero. However, it gives an optimization problem with a smooth objective, and smooth optimization problems have been studied more extensively than problems with non-smooth objectives. In applications where feature pruning is not important, this may make $\ell_2$ preferable due to the increased robustness of available software.

As can be seen from the formulas given above, the form and solution of a model are completely determined by answers to the following questions:

- What are the training instances?
- What are the features?
- What are the normalization sets $GEN(X)$?
- How is the model regularized?

It is in these details that the proposed generalizations of local translation models differ from familiar maxent classifiers.

## 2.3 Classifier translation models

The simplest form of local exponential translation models is similar to standard multiclass classifiers. They generalize phrase-based translation, whose specifics should be assumed to be adopted in what follows without change unless stated otherwise. An independent exponential phrase model corresponds to each source-side phrase $r^x$ observed in training data, according to a standard phrase extraction heuristic. Here the $X$ in eq. 2.1 corresponds to a particular source-side phrase together with its context. The set $GEN(X)$ enumerates all the target-side phrases $r^y$ co-occurring with $r^x$. As in standard classifier models, each feature consists of two parts, one picking out a potentially predictive aspect of $X$ and the other one associated with one of the possible outcomes $Y$. The simplest translation model would thus include only the features combining indicator functions for both halves of some rule $\langle r^x, r^y \rangle$. The maximum likelihood solutions for these models are equivalent to the standard relative frequency phrase models:

$$p(r^y|r^x) = \frac{count(\langle r^x, r^y \rangle)}{\sum_{r^{y'} \in GEN(r^x)} count(\langle r^x, r^{y'} \rangle)} \tag{2.6}$$

To see this, consider the n-th component of the gradient for the entire training set $(X_m, Y_m)$, $m = 1, \ldots M$, corresponding to rule $r$

$$\frac{\partial \log LL(\mathbf{w})}{\partial w_n} = \sum_{m=1}^{M} f_n(X_m, Y_m) - \sum_{m=1}^{M} \sum_{r^{y'} \in GEN(r^x)} f_n(X_m, Y_m) p(r^{y'}|X_m) \tag{2.7}$$

At the maximum likelihood solution the gradient is zero and its two terms are equal. The first term will collect the counts for a given rule in the training set. In

turn, the inner sum in the second term will have exactly one non-zero term equal to $p(r^y|X_m)$ whenever a factor matches the source side of the rule. The maximum likelihood solution thus reduces to

$$\frac{\partial \log LL(\mathbf{w})}{\partial w_n} = count\left(\langle r^x, r^y \rangle\right) - count\left(\langle r^x, r^{y'} \rangle : r^{y'} \in GEN(r^x)\right)\ p(r^y|X_m) = 0$$

which is equivalent to (2.6). Thus classifier local phrase models include the standard phrase model estimates as a special case.

This special case is useful for illustrating several basic properties of the models. Table 2.1 shows maximum likelihood and $\ell_1$-regularized probability estimates for a toy training set consisting of 4 rules, with a feature corresponding to each rule. One can see that the regularizer introduces a form of smoothing of the probabilities. Table 2.2 shows the parameter weights for two of the models, illustrating the weight-pruning property of $\ell_1$ regularization, which in this case uses only a single non-zero parameter. Table 2.3 demonstrates that increasing the number of training instances while preserving their proportions decreases the amount of smoothing produced by the regularizer and prevents any of the estimated weights from being driven to zero. Computationally this occurs because additional terms in the log-likelihood sum reduce the effect of the regularizer in the trade-off. This provides an intuitive account of why regularization makes the estimates more statistically reliable: it increases the contribution of predictors to the model to the degree that empirical evidence warrants it.

We can now generalize these relative frequency estimates by relaxing the restrictions they implicitly place on the form of permissible feature functions. The

| Features | Count | $p_{ML}$ | $p_{\ell_1, C=0.1}$ | $p_{\ell_1, C=0.5}$ |
|---|---|---|---|---|
| $(r_1^x, r_1^y)$ | 2 | 0.5 | 0.47 | 0.45 |
| $(r_1^x, r_2^y)$ | 1 | 0.25 | 0.26 | 0.275 |
| $(r_1^x, r_3^y)$ | 1 | 0.25 | 0.26 | 0.275 |

Table 2.1: Counts and probability estimates for training instances with rule features only.

| Feature | $w_{ML}$ | $w_{\ell_1, C=0.1}$ |
|---|---|---|
| $(r_1^x, r_1^y)$ | 0.46 | 0.59 |
| $(r_1^x, r_2^y)$ | -0.23 | 0 |
| $(r_1^x, r_3^y)$ | -0.23 | 0 |

Table 2.2: Estimated parameter weights for training instances with rule features only.

| Features | Count | $p_{ML}$ | $p_{\ell_1, C=0.1}$ | $w_{\ell_1, C=0.1}$ |
|---|---|---|---|---|
| $(r_1^x, r_1^y)$ | 20 | 0.5 | 0.493 | 0.53 |
| $(r_1^x, r_2^y)$ | 10 | 0.25 | 0.254 | -0.14 |
| $(r_1^x, r_3^y)$ | 10 | 0.25 | 0.254 | -0.14 |

Table 2.3: Counts, probability estimates, and weights for training instances with rule features only and higher counts.

simplest elaboration involves allowing indicator functions for rules to be conjoined with indicator functions for arbitrary attributes of the source sentence or its annotation. We may, for example, conjoin an indicator function for the rule $\langle\, A\,,\, A\,\rangle \rightarrow \langle\, d'\, A_1\, idées\, A_2\,,\, A_1\, A_2\, ideas\,\rangle$ with a function telling us whether a part-of-speech tagger has identified the word at the left edge of the source-side gap $A_2$ as an adjective, which would provide additional evidence for the target side of this rule. Because training instances associated with different source phrases have no features in common, parameter estimation can be decomposed into independent optimization problems, one for each source phrase, without affecting the solution (aside from differences in regularization trade-off). Combining a grammar-based formalism with contextual features raises a subtle question of whether rules which have gaps at the edges and can match at multiple positions of a training example should be counted as having occurred together with their respective contextual features once for each possible match. To avoid favoring monotone rules, which tend to match at many positions, over reordering rules, which tend to match at a single span, we randomly sample only one of such multiple matches for training. Unlike conventional phrase models, contextually-conditioned probabilities cannot be stored in a pre-computed phrase table. Instead, we store information about features and their weights and compute the normalization factors at run-time at the point when they are first needed by the decoder.

At the expense of more complicated decoding procedures we could also apply the same line of reasoning to generalize the "noisy channel" phrase model $p(r^x|r^y)$ to be conditioned on local target-side context in a translation hypothesis, possibly

combining target-side annotation of the training set with surface form of rules. We do not pursue this elaboration in part because we are skeptical about its potential for success. The current state of machine translation rarely permits constructing well-formed translations, so that most of the contextual features on the target side would be rarely if at all observed in the training data, resulting in sparse and noisy estimates.

Unlike other authors, who train arbitrary off-the-shelf classifiers for some phrase types and interpolate their predictions with the standard translation models, the framework presented here recovers the standard models as a special case. This makes it straightforward to replace all main translation models, including lexical scores, with classifier variants and extend them further. In particular, standard classifiers crucially restrict the possible features to be associated with the surface form of some rule. The target side of some rule has to be included because traditional classifier features are associated with one of the outcomes, and the source side has to be included because different source-side phrases correspond to different normalization sets $GEN(X)$. This causes fragmentation and sparsity of training data and prevents us from modeling phenomena that generalize beyond specific lexical items. We describe extensions that overcome this limitation below.

## 2.4 Lexical models

The use of conditional probabilities in standard lexical models gives us a straightforward way to generalize them in the same way as phrase models. Consider

the lexical model $p_w(r^y|r^x)$, defined following Koehn et al [20], with $a$ denoting the most frequent word alignment observed for the rule in the training set.

$$p_w(r^y|r^x) = \prod_{i=1}^{n} \frac{1}{|j|(i,j) \in a|} \sum_{(i,j)\in a} p(w_i^y|w_j^x) \tag{2.8}$$

We replace $p(w_i^y|w_j^x)$ with context-conditioned probabilities computed at the level of individual words. Our experience suggests that, unlike the analogous phrase model, the standard lexical model $p_w(r^x|r^y)$ is not made redundant by this elaboration, and we use its baseline variant in all our experiments.

## 2.5 Source-side inflection models

The simplest extension redefines the classifier models to apply to source-side lemmas, adding features conjoined with indicator functions for Arabic lemmas instead of surface word forms to the lexical models in Arabic-English translation. This preserves decomposition of parameter estimation, with separate optimization subproblems now associated with individual source-side lemmas rather than words. Formally, these models are equivalent to classifiers, except that they treat source-side *lemmas* in the same way the basic classifier models use source-side inflected forms, and use source-side *inflected forms* in the same way the basic classifier models use source-side context. In particular, each normalization set $GEN(X)$ consists of all target-side words co-occurring with a given source-side *lemma*. We apply these estimates to lexical translation models.

## 2.6 Target-side inflection models

Translation into languages with rich morphology presents special challenges for phrase-based methods. Thus, Birch, Osborne & Koehn [4] find that translation quality achieved by a popular phrase-based system correlates significantly with a measure of target-side, but not source-side morphological complexity. Recently, Avramidis & Koehn [2] and Ramanathan et al [29] proposed modeling target-side morphology in a phrase-based factored models framework. Under this approach linguistic annotation of source sentences is analyzed using heuristics to identify relevant structural phenomena, whose occurrences are in turn used to compute additional relative frequency estimates predicting target-side inflections.

These studies demonstrate that modeling target-language inflection can lead to improvement, although their evidence is so far limited to small training sets. The improvements are made possible by consistent structural correspondences between languages. For example, the accusative case is usually preserved in translation, so that nouns appearing in object position of English clauses tend to be translated to words with accusative case markings in languages with richer morphology, and vice versa. However, there are exceptions. For example, some verbs that place their object in the accusative case in Czech may be rendered as prepositional constructions in English [26]:

David was looking for Jana

David hledal Janu

*David searched Jana-ACC*

Conversely, direct objects of some English verbs can be translated by nouns with genitive case markings in Czech:

David asked Jana where Karel was

David zeptal se Jany kde je Karel

*David asked SELF Jana-GEN where is Karel*

Furthermore, English noun modifiers are often rendered by Czech possessive adjectives and a verbal complement in one language is commonly translated by a nominalizing complement in another language, so that the part of speech (POS) of its head need not be preserved. These complications make it difficult to model these phenomena using closed-form estimates. In contrast, exponential models are well suited for capturing the complex interplay of source-side predictors that correlate with target-level inflections. Exponential translation models can be extended to model target-side morphology by use of non-lexicalized features with parameters shared over multiple lexical items. Their form may be clarified by the following toy example.

Suppose our training set contains 6 distinct rules: $(r_1^x, r_1^y)$, $(r_1^x, r_2^y)$, $(r_1^x, r_3^y)$, $(r_2^x, r_4^y)$, $(r_2^x, r_5^y)$, and $(r_2^x, r_6^y)$, the first 3 occurring 10 times each and the last 3 occurring once each, so that maximum likelihood estimates $p(Y|X)$ – which in this case could be computed in closed form – would give each a probability of 1/3. The features observed for this simple case are shown in table 2.4, while table 2.5 summarizes the training set. The normalization sets are $GEN(r_1^x) = \{r_1^y, r_2^y, r_3^y\}$ and $GEN(r_2^x) = \{r_4^y, r_5^y, r_6^y\}$. We have made the indices of target-side phrases, the

| Feature index | Tracked attribute |
|---|---|
| $f_1$ | $(r_1^x, r_1^y)$ |
| $f_2$ | $(r_1^x, r_2^y)$ |
| $f_3$ | $(r_1^x, r_3^y)$ |
| $f_4$ | $(r_2^x, r_4^y)$ |
| $f_5$ | $(r_2^x, r_5^y)$ |
| $f_6$ | $(r_2^x, r_6^y)$ |

Table 2.4: Observed rule features.

indices of features, and the indices of training instances all match for easy reference, but the correspondence between these three indices is arbitrary.

The training set consists of 33 instances, and there is a term for each of them in the summations over $m$ in eqs. 2.2 and 2.3 ($M = 33$). Let us write out in full the term of eq. 2.2 corresponding to the last training instance (omitting the log):

$$p(Y_6|X_6) = \frac{e^{\mathbf{w}\cdot\mathbf{f}(X,Y)}}{\sum_{Y' \in GEN(X)} e^{\mathbf{w}\cdot\mathbf{f}(X,Y')}} = \frac{e^{w_6 f_6}}{e^{w_4 f_4} + e^{w_5 f_5} + e^{w_6 f_6}} \tag{2.9}$$

Now suppose that we associate additional features with certain case-related grammatical phenomena associated with these rules, for example, subject or object position of words on the source side ($c_1^x$ and $c_2^x$, respectively) and nominative or accusative markings on the target side ($m_1^y$ and $m_2^y$). The additional observed features and the new form of the training instances are shown in tables 2.6 and 2.7.

| Instance ($m$) | Features | Count | $GEN(X)$ |
|---|---|---|---|
| 1 | $f_1$ | 10 | $\{r_1^y, r_2^y, r_3^y\}$ |
| 2 | $f_2$ | 10 | $\{r_1^y, r_2^y, r_3^y\}$ |
| 3 | $f_3$ | 10 | $\{r_1^y, r_2^y, r_3^y\}$ |
| 4 | $f_4$ | 1 | $\{r_4^y, r_5^y, r_6^y\}$ |
| 5 | $f_5$ | 1 | $\{r_4^y, r_5^y, r_6^y\}$ |
| 6 | $f_6$ | 1 | $\{r_4^y, r_5^y, r_6^y\}$ |

Table 2.5: Training instances with rule features only.

| Feature index | Tracked attribute |
|---|---|
| $f_7$ | $(c_1^x, m_1^y)$ |
| $f_8$ | $(c_2^x, m_2^y)$ |
| $f_9$ | $(c_1^x, m_2^y)$ |

Table 2.6: Observed shared features.

| Instance ($m$) | Attributes | Features | Count | $GEN(X)$ |
|---|---|---|---|---|
| 1 | $(r_1^x, r_1^y)$ $(c_1^x, m_1^y)$ | $f_1$ $f_7$ | 10 | $\{r_1^y, r_2^y, r_3^y\}$ |
| 2 | $(r_1^x, r_2^y)$ $(c_2^x, m_2^y)$ | $f_2$ $f_8$ | 10 | $\{r_1^y, r_2^y, r_3^y\}$ |
| 3 | $(r_1^x, r_3^y)$ $(c_2^x, m_2^y)$ | $f_3$ $f_8$ | 10 | $\{r_1^y, r_2^y, r_3^y\}$ |
| 4 | $(r_2^x, r_4^y)$ $(c_1^x, m_1^y)$ | $f_4$ $f_7$ | 1 | $\{r_4^y, r_5^y, r_6^y\}$ |
| 5 | $(r_2^x, r_5^y)$ $(c_1^x, m_1^y)$ | $f_5$ $f_7$ | 1 | $\{r_4^y, r_5^y, r_6^y\}$ |
| 6 | $(r_2^x, r_6^y)$ $(c_1^x, m_2^y)$ | $f_6$ $f_9$ | 1 | $\{r_4^y, r_5^y, r_6^y\}$ |

Table 2.7: Training instances with rule features and shared features.

With these additional features the likelihood term corresponding to the last training instance takes the form:

$$p(Y_6|X_6) = \frac{e^{w_6 f_6 + w_9 f_9}}{e^{w_4 f_4 + w_7 f_7} + e^{w_5 f_5 + w_7 f_7} + e^{w_6 f_6 + w_9 f_9}} \tag{2.10}$$

In this case conjoining the contextual attribute $c_1^x$ of instance 6 with the morphological attributes of phrases $r_4^y$ and $r_5^y$ in both cases produces the feature $f_7$. However, such conjunctions of feature halves do not always produce features observed in the training set. For example, in computing the normalization factor for instance 2 we would encounter an unobserved combination of its contextual attribute $c_2^x$ with the morphological attribute $m_1^y$ of the phrase $r_1^y$. The question thus arises whether the model should include additional features and weights for these unobserved combinations. Interestingly, although this issue is relevant in almost

all maxent applications, it does not seem to be commonly discussed. Consider the term of the likelihood corresponding to one occurrence of the second training instance which includes the additional feature $f_{10}$ tracking the attribute combination $(c_1^x, m_2^y)$:

$$p(Y_2|X_2) = \frac{e^{w_2 f_2 + w_8 f_8}}{e^{w_1 f_1 + w_{10} f_{10}} + e^{w_2 f_2 + w_8 f_8} + e^{w_3 f_3 + w_8 f_8}} \tag{2.11}$$

The feature $f_{10}$ appears only in the denominator. Thus, for a given choice of other parameters, we would increase the likelihood by giving it as small a value as possible. This means that a likelihood expression with unobserved features would be maximized by letting all of their weights go to $-\infty$. In other words, even though a maximum of the objective may be found, the solution would not converge. Although adding a regularizer makes the solution convergent, we shall omit unobserved features from all of the models.

The notion of feature sharing can be further illustrated by looking closer at the expression for the log-likelihood gradient. Rearranging the sums in eq. 2.3 we obtain two terms, one equal to the count of a feature in the training set and another one giving its expected count under the current model:

$$\begin{aligned} \frac{\partial \log LL(\mathbf{w})}{\partial w_n} &= \sum_{m=1}^{M} f_n(X_m, Y_m) - \sum_{m=1}^{M} \sum_{Y' \in GEN(X_m)} f_n(X_m, Y')\, p(Y'|X_m) \\ &= \sum_{m=1}^{M} f_n(X_m, Y_m) - \sum_{m=1}^{M} E_{p(Y|X_m)}\left[f_n(X_m, Y)\right] \end{aligned}$$

At the maximum of the objective the gradient is zero, so that these two terms are equal for all features. It is easy to see that all the terms corresponding to

instances where a shared feature appears *or* is generated in the normalizer are pooled together in computing its expectation under the model. In that sense its weight is estimated as though all the rules were part of a single exponential classifier, although in this case the normalization sets $GEN(X_m)$ change from one value of $m$ to another.

Given the distribution of features given in the table 2.7, shared-parameter exponential models with $\ell_1$ regularization ($C = 0.1$) would produce the estimates shown in table 2.8. The last row in the table, calculated according to eq. 2.10, illustrates particularly clearly how feature sharing allows the estimate associated with a rule to be affected by non-lexicalized feature counts observed for rules with which it may not have any lexical items in common. A particularly attractive feature of these models is that they naturally define estimates for inflected forms that do not appear in training data, which can be generated by a straightforward extension of the phrase table, as described below.

## 2.7 Reordering models

Another application of shared features, one especially suited to hierarchical phrase-based translation, involves phrase representations limited to the patterns formed by gaps and words, allowing the model to generalize reordering information beyond individual tokens. We study two types of ordering patterns. For rules with two gaps we form features by conjoining contextual indicator functions with functions indicating whether the gap pattern is monotone or inverting.

| Features | Count | $p_{\ell_1,C=0.1}$ |
|---|---|---|
| $(r_1^x, r_1^y)\ (c_1^x, m_1^y)$ | 10 | 0.987 |
| $(r_1^x, r_2^y)\ (c_2^x, m_2^y)$ | 10 | 0.494 |
| $(r_1^x, r_3^y)\ (c_2^x, m_2^y)$ | 10 | 0.494 |
| $(r_2^x, r_4^y)\ (c_1^x, m_1^y)$ | 1 | 0.356 |
| $(r_2^x, r_5^y)\ (c_1^x, m_1^y)$ | 1 | 0.356 |
| $(r_2^x, r_6^y)\ (c_1^x, m_2^y)$ | 1 | 0.289 |

Table 2.8: Counts and estimates for training instances with shared features.

We also use another type of ordering features, representing the pattern formed by gaps and contiguous subsequences of words. For example, the rule with the right-hand side $\langle\ d'\ A_1\ \textit{idées}\ A_2\ ,\ A_1\ A_2\ \textit{ideas}\ \rangle$ might be associated with the pattern $\langle\ a\ A_1\ a\ A_2\ ,\ A_1\ A_2\ a\ \rangle$. We apply the second type of reordering features to rules with a single gap only. Because some source-side patterns of this type apply to many different rules it is no longer possible to decompose parameter estimation into small independent optimization subproblems. For practical convenience we enforce a coarser-grained decomposition in the experiments reported below in the following way. We define indicator functions for sequences of closed-class words and the most frequent part-of-speech tag for open-class words on the source side. For the rule above and a simple tag-set the pattern tracked by such an indicator function would be $d'\ A_1\ N\ A_2$. We require all reordering features to be conjoined with an indicator

function of this type, ensuring that each corresponds to a separate optimization subproblem.

# Chapter 3

# Experiments: Series I

Two series of experiments were performed. In the first series, described in this section, classifier models were applied to Arabic-English and Chinese-English translation tasks and extended with lexical models, source-side inflection models and reordering models. The main results are shown in tables 3.2-3.4. Table 3.2 shows significant improvements obtained using classifier models with simple lexical scores, classifier-based lexical scores, and lexical scores incorporating source-side inflection models for a small-scale Arabic-English translation task. Tables 3.3 and 3.4 show significant improvements obtained using classifier models and further consistent improvements for models with shared reordering features for a large-scale Chinese-English translation task.

## 3.1 Corpora and baselines

We apply the models to Arabic-English and Chinese-English translation, with training sets consisting of 108,268 and 1,017,930 sentence pairs, respectively.[1] All

[1]The Arabic-English data came from Arabic News Translation Text Part 1 (LDC2004T17), Arabic English Parallel News Text (LDC2004T18), and Arabic Treebank English Translation (LDC2005E46). Chinese-English data came from Xinhua Chinese English Parallel News Text Version 1 beta (LDC2002E18), Chinese Treebank English Parallel Corpus (LDC2003E07), Chinese English News Magazine Parallel Text (LDC2005T10), FBIS Multilanguage Texts (LDC2003E14),

conditions use word alignments produced by sequential iterations of IBM model 1, HMM, and IBM model 4 in GIZA++ , followed by "diag-and" symmetrization [20]. Thresholds for phrase extraction and decoder pruning were set to values typical for the baseline system [12]. Unaligned words at the outer edges of rules or gaps were disallowed. A trigram language model with modified interpolated Kneser-Ney smoothing [11] was trained by the SRILM toolkit [32] on the Xinhua portion of the Gigaword corpus and the English side of the parallel training set. Evaluation was based on the BLEU score with 95% bootstrap confidence intervals for the score and difference between scores, calculated by scripts in version 11a of the NIST distribution. The 2002 NIST MT evaluation sets was used for development. The 2003, 2004, 2005, and 2006 sets were used for testing.

The decision rule was based on the standard log-linear interpolation of several models, with weights tuned by MERT [28] on the development set. The baseline consisted of the language model, two phrase translation models, two lexical models, and a brevity penalty. In the runs where generalized exponential models were used they replaced both of the baseline phrase translation models.

## 3.2 Contextual features and parameter estimation

The feature set used for exponential phrase models in the experiments included all the rules in the grammar and all aligned word pairs for lexical models. Elementary

Chinese News Translation Text Part 1 (LDC2005T06), and the HKNews portion of Hong Kong Parallel Text (LDC2004T08). Some sentence pairs were excluded from the training sets due to large length discrepancies.

contextual features were based on Viterbi parses obtained from the Stanford parser. Word features included identities of word unigrams and bigrams adjacent to a given rule, possibly including rule words. Part-of-speech features included similar ngrams up to the length of 3 and the tags for rule tokens. These features were collected for training by a straightforward extension of rule extraction algorithms implemented in the baseline system for each possible location of ngrams with respect to the rule: namely, at the outer edges of the rule and at the edges of any gaps that it has. Our models also included pairs of contextual features formed by features of the same type (e.g., word-based or POS-based) at the edges of a sequence of one or more non-terminal symbols of a rule. A final type of contextual features in these experiments was the sequence of the highest nodes in the parse tree that fill the span of the rule and the sequences that fill its gaps. We used an Arabic tokenizer based on a Java implementation of Buckwalter's morphological analyzer[2] and incorporating simple statistics from the Penn Arabic treebank, also extending it to perform lemmatization.

The total number of candidate features thus defined is very large, and we use a number of simple heuristics to reduce it prior to training. They are not essential to the estimates and were chosen so that the models could be trained in a few hours on a small cluster. With the exception of discarding all except the 10 most frequent

[2] `http://www.cs.cmu.edu/~cdyer/jbuck.jar`

target phrases observed with each source phrase,[3] which benefits performance, none of these heuristics were applied to the baselines, and we expect that relaxing these restrictions would improve the score. These limitations included count-based thresholds on the frequency of contextual features included into the model, the frequency of rules and reordering patterns conjoined with other features, and the size of optimization subproblems to which contextual features are added. We don't conjoin contextual features to rules whose source phrase terminals are all punctuation symbols. For subproblems of size exceeding a certain threshold, we train on a subsample of available training instances. For the Chinese-English task we do not add reordering features to problems with low-entropy distributions of inversion and reordering patterns and discard rules with two non-terminals altogether if the entropy of their reordering patterns falls under a threshold. Finally, we solve only those optimization subproblems which include parameters needed in the development and training sets. This leads to a reduction of costs that is similar to phrase table filtering and likewise does not affect the solution. The pre-training feature pruning heuristics are summarized in table 3.1. At decoding time all features for the translation models and their weights are accessed from a disk-mapped trie.

We optimize the objective with an an $\ell_1$ regularizer using a variant of the orthant-wise limited-memory quasi-Newton algorithm proposed by Andrew & Gao [1].[4] All values $C_i$ are set to 1 in the experiments below, although we apply stronger

[3]This has prompted us to add an additional target-side token to lexical models, which subsumes the discarded items under a single category. During decoding, the probabilities estimated for it were divided by the appropriate number of unique discarded items.

[4]Our implementation of the algorithm as a SciPy routine is available at

| | Arabic-English | Chinese-English |
|---|---|---|
| sentence pairs | 108,268 | 1,017,930 |
| *phrase-min* | 50 | 100 |
| *rule-min* | 5 | 10 |
| *y-cutoff* | 10 | 10 |
| *problem-max* | 2000 | 2000 |
| *feature-min* | 3 | 5 |

Table 3.1: Feature pruning settings for contextual features used in the experiments: 1) the minimum number of distinct target phrases which a source phrase must co-occur with to be part of contextual features (*phrase-min*); 2) the minimum number of times a rule must occur to be part of contextual features (*rule-min*); 3) the number of target sides a rule can have (keep *y-cutoff* most frequent, including ties); 4) the maximum size of training instances in an optimization subproblem solved without subsampling and the size of subsample for problems exceeding this threshold (*problem-max*); 5) the minimum number of occurrences of a contextual feature in the training set after subsampling (*feature-min*)

regularization ($C_i = 3$) to reordering features. Tuning the regularization trade-off for these models presents special challenges. Optimizing its value for translation performance as measured by BLEU should ideally involve repeating MERT training for each value, which is computationally very expensive. A more efficient alternative is to optimize the trade-off for the cross-entropy computed on a held-out subsample of rules. However, our experimentation showed that this procedure hurt translation accuracy in comparison to setting the trade-off to the value of 1 – an effect that was both surprising and clear-cut. Following the approach of Mann et al [25], the larger training sets for reordering features were split into many approximately equal portions, for which parameters were estimated separately and then averaged for features observed in multiple portions. Parameter estimation was performed using a modified version of the maximum entropy module from SciPy.

## 3.3 Results and discussion

The results are shown in tables 3.2 and 3.3. For both language pairs we had a choice between using a baseline that is computed in the same way as the other exponential models, with the exception of its use of relative frequency estimates and a baseline that incorporates averaged fractional counts for phrase models and lexical models, as used by Chiang [12]. For the sake of completeness we report both (though without performing statistical comparisons between them). Statistical tests for experimental conditions were performed in comparison to the baseline which

`http://www.umiacs.umd.edu/~msubotin/owlqn.py`

| Condition | MT03 | MT04 | MT05 | MT06 |
|---|---|---|---|---|
| Rel. freq. | 48.24 | 43.92 | 47.53 | 37.94 |
| **Frac.** | 48.34 | 45.68 | 47.95 | 39.41 |
| Context | 49.47* | 45.65 | 48.76 | 39.49 |
| +lex | **50.42*** | 46.07* | **49.66*** | 39.32 |
| +lex+lemma | 49.86* | **47.02*** | 49.29* | **40.81*** |

Table 3.2: Arabic-English translation, BLEU scores on testing. Conditions include two baselines: simple relative frequency (rel. freq.) and fractional estimates (frac.). Experimental conditions: contextual features in phrase models (context); same and contextual features in lexical models (+lex); same and lemma based features in lexical models (+lex+lemma). Stars mark statistically significant improvements over the fractional baseline which produced a higher score on the dev-test MT02 set than the other baseline (59.75 vs. 59.66).

| Condition | MT03 | MT04 | MT05 | MT06 |
|---|---|---|---|---|
| **Rel. freq.** | 32.82 | 27.42 | 30.68 | 22.55 |
| Frac. | 32.21 | 27.94 | 30.82 | 23.35 |
| Context | 33.21* | 28.88* | 31.34* | 23.97* |
| +lex | 33.13* | 28.52* | 31.72* | 23.60 |
| +lex+reord | **33.86*** | **29.47*** | **32.09*** | **24.52*** |

Table 3.3: Chinese-English translation, BLEU scores on testing. Conditions include two baselines: simple relative frequency (rel. freq.) and fractional estimates (frac.). Experimental conditions: contextual features in phrase models (context); same and contextual features in lexical models (+lex); same and reordering features in phrase models (+lex+reord). Stars mark statistically significant improvements over the simple relative frequency baseline which produced a higher score on the dev-test MT02 set than the other baseline (33.74 vs. 33.26).

achieved higher score on the test-dev MT02 set: the fractional count baseline for Arabic-English and the simple relative count baseline for Chinese-English.

We test models with classifier solutions for phrase models alone and for phrase models together with lexical models in both language pairs. For Arabic-English translation we also experiment with adding features based on lemmas to lexical models, while for Chinese-English we add "reordering" features – features based on the ordering pattern of gaps for rules with two gaps and features based on ordering of gaps and words for rules with a single gap.

For both language pairs the results show consistent distinctions in behavior of different models between the test sets giving rise to generally higher scores (MT03 and MT05) and generally lower scores (MT04 and MT06). The fractional counts seem to be more helpful for test sets with poorer coverage, although the reason for this is not immediately clear. For exponential models the two type of sets present two possible sources of difference. The lower-performing sets have poorer coverage in the training data, and they also may suffer from lower-quality annotation, since the training sets for both the translation models and the annotation tools are dominated by text in the same, newswire domain. Overall, the use of features based on surface forms is more beneficial for MT03 and MT05. In contrast, using features based on less specific representations is more beneficial on test sets with poorer coverage. This agrees with our intuitions and also suggests that the differences in coverage of training data for the translation models may be playing a more important role in these trends than coverage for annotation tools.

These experiments do not use the largest or optimally selected data sets avail-

able for training either the translation or language models. Consequently, the obtained scores are lower than those reported in other recent work, such as Chiang et al [13] and Setiawan et al [30]. In this regard, it should be emphasized that the primary goal of present study is not to beat the current strongest baselines for these particular language pairs, but rather to demonstrate scalability of the proposed framework in conjunction with consistency of improvements it can produce over a varied range of data scenarios and model types. The ability of a model to yield improvements on a training set of modest size and/or for a testing set with poor coverage in training data is no less important than improvement it may yield on training and testing data derived from the largest extant parallel corpus, since for the vast majority of world's languages the available resources belong to the former end of the spectrum. We argue that the promise held by this framework crucially depends on the variety of feature types it can incorporate, and it is beyond the scope of the present study to find the variation that is optimal for any given language pair. At the same time, its scalability and consistency of reported improvements set it apart from other proposed models capable of incorporating a similar variety of features. Thus, Liang et al [22], who use a training set of 67K sentence pairs, precomputed aggregate features, and monotonic translation, report a 0.1 BLEU point improvement over a non-hierarchical phrase system with a 3-gram language model. Tillmann & Zhang [34], who use a training set of 230K sentences, do not present comparisons with a standard baseline. Blunsom & Osborne [5], who use a training set of 38K sentence pairs with an average length of under 10 words per sentence, report improvements over a hierarchical phrase-based baseline with a 3-gram language model only for

some forms of BLEU, and do not report significance tests.

Among the different ways in which a baseline can be improved, the n-gram limit of language model arguably presents a reasonable exception to the line of argument in the preceding paragraph, since monolingual data is easier to obtain than parallel data, and using 5-gram language models generally does not present additional difficulties for any language, given sufficient random-access memory. We therefore briefly consider results for the Chinese-English models trained with a 5-gram language model, shown in table 3.4. The pattern of results is similar for models with simple contextual features, except for the generally higher scores. In contrast, the version with reordering features shows a surprising degradation of BLEU on test. Upon closer examination, the reason for this lies in the selection of MT02 for tuning MERT weights. Evidently, its relatively high coverage causes it to learn weights that are appropriate only for another set with similarly high coverage like MT05. The last line in the table, which shows results for the same model, but with MERT weights borrowed from a simple contextual model (+lex) supports this conjecture. These results – which provide a conservative estimate for the model performance, since they can only be improved by better selection of MERT weights – demonstrate the same pattern of results observed above, although the degree of improvements obtained from reordering features is smaller. A more thorough investigation of domain adaptation issues for these translation models is beyond the scope of the present study and is left for future research.

Finally, table 3.5 illustrates the effects of run-time exponential model normalization and $\ell_1$ regularization on decoding speed using the example of a Chinese-

| Condition | MT03 | MT04 | MT05 | MT06 |
|---|---|---|---|---|
| Baseline | 33.78 | 28.70 | 32.35 | 23.44 |
| Context | 34.27 | 29.24* | 33.38* | 23.74 |
| +lex | 34.51* | 31.44* | 33.18* | 26.32* |
| +lex+reord+mert | 34.44 | 29.18* | **33.76*** | 23.61 |
| +lex+reord-mert | **34.83*** | **31.94*** | 33.45* | **26.81*** |

Table 3.4: Chinese-English translation, 5-gram language model, BLEU scores on testing. Experimental conditions: contextual features in phrase models (context); same and contextual features in lexical models (+lex); same and reordering features in phrase models (+lex+reord+mert); same but with MERT weights taken from the model marked +lex (+lex+reord-mert). Asterisks mark statistical significance over the relative frequency baseline.

| Model type | Run-time norm. | Rules | Context features | Avg. time (sec) |
|---|---|---|---|---|
| Baseline | No | 5,539,989 | 0 | 4.3 |
| No contexts | Yes | 5,539,989 | 0 | 6.8 |
| Non-zero contexts | Yes | 5,539,989 | 5,024,917 | 28.1 |
| All contexts | Yes | 5,539,989 | 18,889,683 | 80.7 |

Table 3.5: Effects of feature pruning on decoding speed (average decoding time per sentence). Baseline: precomputed grammar, no run-time normalization. No contexts: phrase models normalized at decoding time, no context features. Non-zero contexts: standard model with context features. All contexts: same, but including context features with zero weights.

English model with contextual phrase-model features. For consistency of comparison, the same pre-training pruning heuristics were applied to all the models shown in the table. The table shows that normalization of the models during decoding accounts for only a small portion of increased computational expenses for the contextually-sensitive models. This is because only a single normalization factor needs to be computed for each source phrase matched in the translated sentence. In contrast, checking contextual features incurs considerable expenses even with feature match caching in the decoder.

# Chapter 4

# Experiments: Series II

This section describes an application of target-side morphology model to English-Czech translation, including modeling of inflected variants unobserved in training data. The main results, presented in tables 4.2 and 4.3, show significant improvements for target-side inflection models obtained for small-scale and large-scale English-Czech translation tasks with a 3-gram language model, and consistent improvements obtained with a 5-gram language model.

## 4.1 Features for target-side inflection models

The feature space for target-side inflection models used in this work consists of features tracking the source phrase and the corresponding target phrase together with its complete morphological tag, which will be referred to as *rule features* for brevity. The feature space also includes features tracking the source phrase together with the lemmatized representation of the target phrase, called *lemma features* below. Since there is little ambiguity in lemmatization for Czech, the lemma representations were for simplicity based on the most frequent lemma for each token. Finally, we include features associating aspects of source-side annotation with inflections of aligned target words. Inflection features fall into three classes corresponding to the POS for the source word aligned to the target word whose inflection is being

predicted.

### 4.1.1 Nouns

Among the inflection types of Czech nouns, the only type that is not generally observed in English and does not belong to derivational morphology is inflection for case. Czech marks seven cases: nominal, genitive, dative, accusative, vocative, locative, and instrumental. Not all of these forms are overtly distinguished for all nouns, and some lexical items that function syntactically as nouns do not inflect at all. The following feature types were included:

- The structural position of the aligned source word or the head of the smallest noun phrase containing the aligned source word. Features were included for the roles of subject, direct object, and nominal predicate.

- The preposition governing the smallest noun phrase containing the aligned source word, if it is governed by a preposition.

- An indicator for the presence of a possessive marker modifying the aligned source word or the head of the smallest noun phrase containing the aligned source word.

- An indicator for the presence of a numeral modifying the aligned source word or the head of the smallest noun phrase containing the aligned source word.

- An indication that aligned source word modified by quantifiers *many*, *most*, *such*, and *half*. These features would be more properly defined based on the

identity of the target word aligned to these quantifiers, but little ambiguity seems to arise from this substitution in practice.

Features corresponding to aspects of the source word itself and features corresponding to aspects of the head of a noun phrase containing it were treated as separate types.

### 4.1.2 Adjectives

Czech adjectives inflect for case, number, gender, animacy, negation, and degree of comparison. The features we use for adjectives include all the same types depending on properties of the smallest noun phrase containing the aligned source word which we have described for nouns above, in addition to the number marking of its head. Gender and animacy present a more difficult inflection type to model, since they depend on the lexical choice for the head of the *target* noun phrase which contains the predicted inflection. Thus, the decoder needs to split states until this choice can be known in the course of the derivation. These features are also unlikely to make an impact on translation quality, since Czech adjectives are generally adjacent to their head noun, or to another word in the same noun phrase. Thus, this type of agreement is especially likely to be captured by a language model. We leave these types of inflection aside. Negation and the degree of comparison is also not modeled by special features, although they are taken into account in generating morphological variants, as described below.

### 4.1.3 Verbs

Czech verbs inflect for person, number, gender, negation, and aspect. We add features marking the number of the subject of the verb aligned to the predicted inflected form, with separate features for noun and pronoun subjects. The considerations outlined above for adjective gender agreement also apply to verbs. Although verbs are more commonly separated from their subject noun phrase, verbal features are also less likely to affect translation quality since Czech marks gender agreement only in the past tense, and verbs appear in formal text less frequently than adjectives and nouns in terms of total counts. The other forms of inflection are left aside since they are even more infrequent.

We add inflection features for all words aligned to at least one English verb, adjective, noun, pronoun, or determiner, excepting definite and indefinite articles. A separate feature type marks cases where an intended inflection category is not applicable to a target word falling under these criteria due to a POS mismatch between aligned words.

## 4.2 Modeling unobserved target inflections

As a consequence of translating into a morphologically rich language, some inflected forms of target words are unobserved in training data and cannot be generated by the decoder under standard phrase-based approaches. Exponential models with shared features provide a straightforward way to estimate probabilities of unobserved inflections. This is accomplished by extending the sets of target phrases

$GEN(X)$ over which the model is normalized by including some phrases which have not been observed in the original sets. When additional rule features with these unobserved target phrases are included in the model, their weights will be estimated even though they never appear in the training examples (i.e, in the denominator of their likelihoods)[1].

We generate unobserved morphological variants for target phrases starting from a generation procedure for target words. Morphological variants for words were generated using the ÚFAL MORPHO tool [21]. The forms produced by the tool from the lemma of an observed inflected word form were subjected to several restrictions:

- For nouns, generated forms had to match the original form for number.
- For verbs, generated forms had to match the original form for tense and negation.
- For adjectives, generated forms had to match the original form for degree of comparison and negation.
- For pronouns, excepting relative and interrogative pronouns, generated forms had to match the original form for number, case, and gender.

[1]The fact that the additional inflections share lemma features with observed forms may suggest an alternative method where the model is trained in the usual way and only lemma and inflection features are used for unobserved forms. However, this approach would be incorrect. It implicitly assumes that unobserved rule features have zero weights, which would make their weights larger than the negative weights assigned to many observed rule features.

- Non-standard inflection forms for all POS were excluded.

The following criteria were used to select rules for which expanded inflection sets were generated:

- The target phrase had to contain exactly one word for which inflected forms could be generated according to the criteria given above.
- If the target phrase contained prepositions or numerals, they had to be in a position not adjacent to the inflected word. The rationale for this criterion was the tendency of prepositions and numerals to determine the inflection of adjacent words.
- The lemmatized form of the phrase had to account for at least 25% of target phrases extracted for a given source phrase.

The standard relative frequency estimates for the $p(X|Y)$ phrase model and the lexical models do not provide reasonable values for the decoder scores for unobserved rules and words. In contrast, exponential models with surface and lemma features can be straightforwardly trained for all of them. For the experiments described below we trained an exponential model for the $p(Y|X)$ lexical model. For greater speed we estimate the probabilities for the other two models using interpolated Kneser-Ney smoothing [11], where the surface form of a rule or an aligned word pair plays to role of a 3-gram, the pairing of the source surface form with the lemmatized target form plays the role of a bigram, and the source form alone plays the role of a unigram.

## 4.3 Parameter estimation

Although the number of contextual features in these experiments was much smaller than in the experiment described above, optimization nevertheless presented a challenge due to the size of the training sets. Several strategies were pursued to reduce the computational expenses. As above, we used the parallelization strategy of Mann et al [25]. The sets of target phrases for each source phrase prior to generation of additional inflected variants were truncated by discarding extracted rules which were observed with frequency less than the 200-th most frequent target phrase for that source phrase. Exponential models included an $\ell_2$ regularizer with $C = 1$. We do not use $\ell_1$ regularization as in the other experiments, because for these models dimensionality reduction gives no practical advantage, while the combination of weight-averaging parallelization of training with $\ell_1$ regularization is not well researched and may show undesirable effects. Additional computational challenges remained due to an important difference between models with shared features and models discussed in earlier sections. Features appearing with source phrases found in development and testing data share their weights with features appearing with other source phrases, so that filtering the training set for development and testing data affects the solution. Although there seems to be no reason why this would positively affect translation accuracy, to be methodologically strict we estimate parameters for rule and lemma features without inflection features for larger models, and then combine them with weights for inflection features estimates from a smaller portion of training data. This should affect model performance negatively, since

it precludes learning trade-offs between evidence provided by the different kinds of features, and therefore it gives a conservative assessment of the results that could be obtained at greater computational costs. Inflection features from the small data set models were used in the the other models.

## 4.4 Corpora and baselines

We investigate the models using the 2009 edition of the parallel treebank from ÚFAL [6]. The set is very large and contains texts from a number of genres (table 4.1), which gives us an opportunity to study performance of the models in a variety of settings. The English side follows the standards of the Penn Treebank and includes dependency parses and semantic role labels. The Czech tags follow the standards of the Prague Dependency Treebank. While the Czech side includes several layers of annotation, only the morphological tags and lemmas are used in this study.

The impact of the models on translation accuracy was investigated for 2 experimental conditions:

- Small data set: trained on the news portion of the data; development and testing sets containing 1500 sentences of news text each.

- Large data set: trained on all the training data; developing and testing sets each containing 1500 sentences of EU, news, and fiction data in equal portions. The other genres were excluded from the development and testing sets because manual inspection showed them to contain a considerable proportion of non-parallel sentences pairs.

| Genre | Sentence pairs | English words | Czech words |
|---|---|---|---|
| EU Legislation | 1,271,413 | 25,374,862 | 22,719,663 |
| Prose fiction | 830,354 | 13,624,545 | 12,006,426 |
| News | 111,176 | 2,527,090 | 2,328,075 |
| Movie Subtitles | 2,840,595 | 21,245,195 | 17,735,011 |
| Technical Documentation | 968,658 | 7,272,451 | 6,745,935 |
| Parallel Web Pages | 372,563 | 3,955,808 | 3,796,210 |
| Project Navajo | 37,239 | 485,993 | 427,293 |
| Total | 6,424,476 | 74,485,944 | 65,758,613 |

Table 4.1: Descriptive statistics for the CzEng09 corpus

The training of baselines and evaluation of the results was performed using the procedures described above for the experiments of series I, with 3-gram and 5-gram language models estimated from a set of 208 million running words of text obtained by combining the monolingual Czech text distributed by the 2010 ACL MT workshop with the Czech portion of the training data.

## 4.5 Results and discussion

Table 4.2 shows the results obtained with a 3-gram language model. For both data sets translation accuracy was significantly higher for the experimental condition than for the baseline. Results with a 5-gram language model are shown in table 4.3. We can see that for the smaller data set the gains obtained over the baseline remain almost the same for both language models, with slightly larger improvements obtained with a 5-gram language model. For the large data set the improvement with a 5-gram language model is smaller. To gain better insight into the role played by different elements of the model (grammar expansion, phrase model scores incorporating tag information, and non-standard estimates for all models), we also evaluated a system that was identical to the experimental condition, except that the grammar was restriced to observed rules and the main exponential model was replaced by a relative frequency phrase model based on counts of rules labeled with target-side tags. The last column in the table shows that inflection-based features account for about half of the gains obtained over the baseline for the small data set and for all of the gain obtained for the large data set. The fact that the

| Condition | Baseline | Exponential |
|---|---|---|
| Small data set | 0.1905 | **0.2116*** |
| Large data set | 0.2429 | **0.2503*** |

Table 4.2: BLUE scores on testing, 3-gram language model.

| Condition | Baseline | Exponential | W/out infl. |
|---|---|---|---|
| Small data set | 0.1964 | **0.2184*** | 0.2067 |
| Large data set | 0.2562 | **0.2573** | 0.2522 |

Table 4.3: BLUE scores on testing, 5-gram language model. The last column shows scores for a system that is identical to the experimental condition except in being restriced to observed rules and in having the main exponential model replaced with a relative frequency phrase model.

other elements of the model actually degrade performance for the large data set also suggests that the results obtained by the full model could be improved by tuning various aspects of these elements.

Tables 4.4 and 4.5 show a summary of the grammars and feature spaces for the models. The weights learned for inflection features generally conformed with intuitive expectations. Table 4.6 shows a typical example for the feature which associates constituents of a noun phrase headed by a subject and case markings for the aligned target word.

We now illustrate general properties of these models using toy examples and

| Condition | Total rules | Added rules | Source phrases | w/ added rules |
|---|---|---|---|---|
| Small data set | 11,811,041 | 7,827,430 | 1,270,064 | 503,331 |
| Large data set | 38,040,661 | 12,123,583 | 2,179,650 | 749,831 |

Table 4.4: Summary of grammars (all filtered for dev/test sets). The columns show the total number of rules with target tags, the number of rules added through generation of inflected forms, the number of distinct source phrases, and the number of distinct source phrases with some rules added through generation of inflected forms.

| Condition | Rule features | Lemma features | Inflection features |
|---|---|---|---|
| Small data set | 54,896,451 | 19,592,256 | 750 |
| Large data set | 37,562,137 | 19,117,426 | 0 |

Table 4.5: Distinct features included in the models. Only the large training set was filtered for dev/test sets. Parameters from inflection features from the small data set model were used in the large data model during decoding.

| Case code | Case | Feature weight |
|---|---|---|
| 1 | Nominative | 1.6614028807 |
| 2 | Genitive | 0.0266661206 |
| 3 | Dative | -0.6682866795 |
| 4 | Accusative | -0.1597880022 |
| 5 | Vocative | -3.3165760566 |
| 6 | Locative | -1.2214837635 |
| 7 | Instrumental | -0.8131854644 |
| X | Any | 1.2539379959 |
| - | n/a | -0.5932031147 |

Table 4.6: Feature weights for target case and source constituents of a noun phrases headed by a subject. The coded representations for case forms used in the corpus annotation are included for future reference.

the actual parameters estimated from the large data set. Table 4.7 shows a summary of rule sets for 2 source phrases extracted from the large training set. The columns show the counts for total and unique rule features extracted from the training data, the number of lemmatized target phrase forms selected for generation of inflection variants, and the number of added inflection variants. Table 4.8 provides additional details about the rulesets, illustrated using 3 representative rules selected for each rule set: a relatively rare rule, a relatively frequent rule, and an unobserved rule added to the grammar through generation of inflection variants.

The next two tables illustrate the probabilities estimated for these rule sets. Table 4.9 shows the baseline $p(Y|X)$ phrase model estimated from the counts of tagged rules and compares it to estimates given by the exponential model with rule and lemma features computed on the large data set. The values for the estimated parameters are also shown. One can see from the table that for both rule sets there is little difference in the estimated probabilities for rules observed a single time and generated unobserved rules, although this similarity originates from two different underlying factors: from the greater smoothing caused by the lower frequencies in the case the first rule set and from the lower probability alloted to singletons rules in the second rule set due to the greater total number of observed rules in the set. The probability of the more frequent example rule in the first rule set is additionally pushed down by the negative value of the lemma feature, which results from the addition of terms with that feature in the denominator of the likelihood for generated inflected variants. Although this effect is partially compensated by the surface feature for that rule, to which the model is thus forced to give additional

| | Source phrase | Observed rules | Expanded forms | Added variants |
|---|---|---|---|---|
| 1 | ( $A_1$ ) $A_2$ cooperation | 9 (7 distinct) | 1 | 6 |
| 2 | ( $A_1$ cooperation | 131 (22 distinct) | 2 | 12 |

Table 4.7: Example rule sets for two source phrases extracted from the large data set.

credence, the smoothing of probabilities for observed rules in this particular rule set seems excessive on intuitive level, and suggests that it might be benefitial to add inflected variants for *all* rules in any ruleset to which any generated inflected forms are added, despite practical complications this presents. On the other hand, this effect is virtually unnoticeable in the second ruleset, where the additional terms in the normalization factor are well compensated by the prevalence of the lemma form corresponding to both of the observed rules chosen for illustration.

The same models were then recomputed in the decoder for artificially constructed examples with and without inflection features with the results shown in table 4.10. There is a grammatical match between nominative case for the target word and subject position for the aligned source word and between the accusative case for the target word and the object role for the aligned source word. The other pairings represent grammatical mismatches. One can see that the probabilities for rules leading to correct case matches are several times greater than the alternatives with incorrect case matches.

| Source # | Target # | Target phrase | Target tag | Observed |
|---|---|---|---|---|
| 1 | 1 | ( $A_1$ ) , $A_2$ spolupráci | NNFS3——A- | 1 |
| 1 | 2 | ( $A_1$ ) $A_2$ spolupráci | NNFS4——A- | 3 |
| 1 | 3 | ( $A_1$ ) $A_2$ spolupráce | NNFS1——A- | 0 |
| 2 | 1 | ( $A_1$ spoluprací | NNFS7——A- | 1 |
| 2 | 2 | ( $A_1$ spolupráce | NNFS1——A- | 31 |
| 2 | 3 | ( $A_1$ spolupráci | NNFS4——A- | 0 |

Table 4.8: Representative target phrases for the two rulesets, the tags of the target word subject to morphological generation and their observed counts. The case is marked in the fifth position of the tag using the coding given in table 4.6.

| Source # | Target # | Baseline | Exponential | Weights (rule/lemma) |
|---|---|---|---|---|
| 1 | 1 | 0.111111111 | 0.0849776054 | 0.0334523185 / 0.0669046369 |
| 1 | 2 | 0.333333333 | 0.0859253624 | 0.2470487544 / -0.1356005141 |
| 1 | 3 | 0 | 0.0628478066 | -0.0657142498 / -0.1356005141 |
| 2 | 1 | 0.007633588 | 0.0066588695 | -0.4731434663 / 0.0285786938 |
| 2 | 2 | 0.236641221 | 0.2115539786 | 2.9853869979 / 0.0285786938 |
| 2 | 3 | 0 | 0.0047809538 | -0.8044531177 / 0.0285786938 |

Table 4.9: Model details for example rule sets introduced in tables 4.7 & 4.8. The columns show the baseline relative frequency probabilities, the probabilities estimated by the exponential model without inflection features, and values of weights for rule and lemma features corresponding to each rule.

| Source # | Target # | Rule case | No infl. | Sb | Obj |
|---|---|---|---|---|---|
| 1 | 1 | Dat | 0.084978 | 0.037419 | 0.035458 |
| 1 | 2 | Acc | 0.085925 | 0.091517 | 0.203561 |
| 1 | 3 | Nom | 0.062848 | 0.416277 | 0.062669 |
| 2 | 1 | Instr | 0.006659 | 0.001585 | 0.003247 |
| 2 | 2 | Nom | 0.211554 | 0.623763 | 0.168965 |
| 2 | 3 | Acc | 0.004781 | 0.002267 | 0.009468 |

Table 4.10: The effect of inflection features on estimated probabilities. The estimates are shown for models with and without inflection features, computed for translated sentences with the aligned word with subject and object structural positions.

# Chapter 5

# Conclusion

## 5.1 Summary

We reiterate the main contributions of the thesis below.

- This thesis contributes to the development of the subclass of annotation-sensitive translation models which associate probability distributions with individual rewrite rules used by the translation system, such as synchronous context-free rules, or with other individual aspects of translation hypotheses such as word pairs or reordering events.

- Unlike other authors we used the new estimates to replace the traditional phrase models and lexical scores, rather than in addition to them, thereby demonstrating that the local exponential phrase models can be regarded as a generalization of standard methods not only in theoretical but also in practical terms.

- We have introduced a form of local translation models that combine features associated with surface forms of rules and features associated with less specific representation – including those based on lemmas (sections 2.5 & 2.6), inflections (*ibid.*), and reordering patterns (section 2.7) – such that surface-form estimates were recovered as a special case of the model, while parameters

for the latter type of features could be estimated from training sets spanning multiple source phrases.

- These proposals were experimentally validated. Conditioning all phrase-based probabilities in a hierarchical phrase-based system on source-side contextual information showed significant improvements for 2 language pairs and 8 test sets (section 3). Extending the contextually-sensitive estimates with features modeling source-side morphology and reordering patterns yielded additional improvements (*ibid.*), while further experiments showed significant improvements obtained from modeling observed and unobserved inflections for a morphologically rich target language (section 4).

## 5.2 Further extensions

### 5.2.1 Other possible feature types

The general form of proposed models can straightforwardly support other feature types besides those described above. Thus, any version of the model could incorporate features tracking aligned word pairs inside grammar rules, thereby taking over some of the work normally done by the standard lexical models. Conversely, features tracking aspects of the target side of rules without dependence of the source side would play a role analogous to the language model, but optimizing individual weight trade-off with the translation model while performing parameter estimation on the parallel data set alone. Instead of using the output of monolingual annotation

tools, one could use features that depended on character n-grams seeking to capture morphological inflections or classes learned by clustering algorithms seeking to capture semantic or syntactic generalizations. Finally, a model of reordering could be extended to include features tracking the topological ordering of function words studied by Setiawan et al [30], taking the heuristic they use to count reordering patterns and adapting it to identify instances of these reordering patterns within the span of non-terminals in a parallel corpus and at decoding time.

### 5.2.2 Other possible applications

The general form of exponential model with shared features can be combined with other methods besides standard phrase-based machine translation. Thus, it could be applied to other translation models based on local conditional probabilities conditioned on source-side information, such as a reversed (target given source) variant of the syntax-based translation model used in the GHKM system [16]. It could also be used in systems of this kind designed for other tasks, such as paraphrase generation [24]. Local exponential models with $\ell_1$ regularization could be used for the purposes of selecting contextual features to include in the global translation model by Blunsom & Osborne [5] or perceptron-based approaches [22, 34]. As was shown in section 3.3 (table 3.5), checking contextual features is computationally expensive, and it is reasonable to hypothesize that for important classes of contextual features lack of discriminative power in a local translation model, as evidenced by an estimated zero weight, implies low discriminative power in a global translation

model. These two approaches could be futher combined with the feature-rich locally normalized version of the EM algorithm proposed by Berg-Kirkpatrick et al [3] to further reduce computational expenses of training. With this combined approach, the synchronous parsing would be used to collect expected counts, with the results providing the basis for an iterative M-stem.

We leave these possibilities for future research.

# Bibliography


[1] G. Andrew and J. Gao. Scalable training of L1-regularized log-linear models. International Conference on Machine Learning (ICML), 2007.

[2] E. Avramidis and P. Koehn. Enriching Morphologically Poor Languages for Statistical Machine Translation, The Annual Meeting of Association of the Association of Computational Linguistics (ACL), 2008.

[3] T. Berg-Kirkpatrick, J. DeNero, and D. Klein.Painless Unsupervised Learning with Features. The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), 2010.

[4] A. Birch, M. Osborne and P. Koehn. Predicting Success in Machine Translation. The Conference on Empirical Methods in Natural Language Processing (EMNLP), 2008.

[5] P. Blunsom and M. Osborne. Probabilistic Inference for Machine Translation. The Conference on Empirical Methods in Natural Language Processing (EMNLP), 2008.

[6] O. Bojar and Z. Žabokrtský. Large Parallel Treebank with Rich Annotation. Charles University, Prague. http://ufal.mff.cuni.cz/czeng/czeng09/, 2009.

[7] P. F. Brown, S. A. Della Pietra, V. J. Della Pietra, R. L. Mercer. Word-Sense Disambiguation Using Statistical Methods. The Annual Meeting of Association of Computational Linguistics (ACL), 1991.

[8] M. Carpuat and D. Wu. Improving Statistical Machine Translation using Word Sense Disambiguation. Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), 2007.

[9] M. Carpuat and D. Wu. How Phrase Sense Disambiguation outperforms Word Sense Disambiguation for Statistical Machine Translation. 11th Conference on Theoretical and Methodological Issues in Machine Translation (TMI), 2007.

[10] Y. S. Chan, H. T. Ng, and D. Chiang. Word sense disambiguation improves statistical machine translation. The Annual Meeting of Association of Computational Linguistics (ACL), 2007.

[11] S. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. Technical report TR-10-98, Harvard University, 1998.

[12] D. Chiang. Hierarchical phrase-based translation. Computational Linguistics 33(2):201-228, 2007.

[13] D. Chiang, Y. Marton and P. Resnik. Online large-margin training of syntactic and structural translation features, The Conference on Empirical Methods in Natural Language Processing (EMNLP), 2008.

[14] D. Chiang, W. Wang and K. Knight. 11,001 new features for statistical machine translation, The Annual Conference of the North American Chapter of the Association for Computational Linguistics and Human Language Technologies (NAACL HLT), 2009.

[15] J. Gao, G. Andrew, M. Johnson and K. Toutanova. A Comparative Study of Parameter Estimation Methods for Statistical Natural Language Processing. The Annual Meeting of the Association of Computational Linguistics (ACL), 2007.

[16] M. Galley, J. Graehl, K. Knight, D. Marcu, S. DeNeefe, W. Wang, and I. Thayer. Scalable Inference and Training of Context-Rich Syntactic Models. The Annual Meeting of Association of Computational Linguistics and the International Conference on Computational Linguistics (ACL-COLING), 2006.

[17] Z. He, Q. Liu and S. Lin. Improving Statistical Machine Translation using Lexicalized Rule Selection. The International Conference on Computational Linguistics (COLING), 2008.

[18] Z. He, Y. Meng and H. Yu. Maximum Entropy Based Phrase Reordering for Hierarchical Phrase-based Translation. The Conference on Empirical Methods in Natural Language Processing (EMNLP), 2010.

[19] P. Koehn and H. Hoang.Factored translation models. The Conference on Empirical Methods in Natural Language Processing (EMNLP), 2007).

[20] P. Koehn, F. J. Och, D. Marcu. Statistical Phrase-Based Translation. The Annual Conference of the North American Chapter of the Association for Computational Linguistics and Human Language Technologies (HLT-NAACL), 2003.

[21] D. Kolovratník & L. Přikryl. Programátorská dokumentace k projektu Morfo. http://ufal.mff.cuni.cz/morfo/, 2008.

[22] P. Liang, A. Bouchard-Cote, D. Klein and B. Taskar. An End-to-End Discriminative Approach to Machine Translation. The Annual Meeting of the Association for Computational Linguistics (ACL), 2006.

[23] D.C. Liu and J. Nocedal. On the Limited Memory Method for Large Scale Optimization, Mathematical Programming B, 45(3), pp. 503-528, 1989.

[24] N. Madnani and B. Dorr. Generating Phrasal & Sentential Paraphrases: A Survey of Data-Driven Methods. Computational Linguistics, 36(3), 2010.

[25] G. Mann, R. McDonald, M. Mohri, N. Silberman, D. Walker. Efficient Large-Scale Distributed Training of Conditional Maximum Entropy Models. Advances in Neural Information Processing Systems (NIPS), 2009.

[26] J. Naughton (2005) *Czech. An Essential Grammar.* Routledge, 2005.

[27] A. Y. Ng. Feature selection, L1 vs. L2 regularization, and rotational invariance. The International Conference on Machine Learning (ICML), 2004.

[28] F. J. Och. Minimum Error Rate Training in Statistical Machine Translation. The Annual Meeting of Association of Computational Linguistics (ACL), 2003.

[29] A. Ramanathan, H. Choudhary, A. Ghosh, P. Bhattacharyya (2009) Case markers and Morphology: Addressing the crux of the fluency problem in English-Hindi SMT. The Annual Meeting of Association of Computational Linguistics (ACL), 2009.

[30] H. Setiawan, M.-Y. Kan, H. Li and P. Resnik. Topological Ordering of Function Words in Hierarchical Phrase-based Translation. The Annual Meeting of Association for Computational Linguistics (ACL), 2009.

[31] N. Smith. Log-linear models. Unpublished manuscript, 2004.

[32] A. Stolcke. SRILM – An Extensible Language Modeling Toolkit. International Conference on Spoken Language Processing, 2002.

[33] M. Subotin. Generalizing local translation models. The Annual Meeting of Association for Computational Linguistics, Second Workshop on Syntax and Structure in Statistical Translation (ACL SSST-2), 2008.

[34] C. Tillmann and T. Zhang. A Discriminative Global Training Algorithm for Statistical MT. The Annual Meeting of Association for Computational Linguistics (ACL), 2006.

[35] D. Xiong, Q. Liu, and S. Lin. Maximum entropy based phrase reordering model for statistical machine translation. The Annual Meeting of Association for Computational Linguistics (ACL), 2006.

[36] R. Yeniterzi and K. Oflazer. Syntax-to-Morphology Mapping in Factored Phrase-Based Statistical Machine Translation from English to Turkish. The Annual Meeting of Association for Computational Linguistics (ACL), 2010.