Maximum Entropy Translation Model in Dependency-Based MT Framework

David Mareček, Martin Popel, Zdeněk Žabokrtský
Charles University in Prague, Institute of Formal and Applied Linguistics
Malostranské nám. 25, Praha 1, CZ-118 00, Czech Republic
{marecek,popel,zabokrtsky}@ufal.mff.cuni.cz

Abstract

Maximum Entropy Principle has been used successfully in various NLP tasks. In this paper we propose a forward translation model consisting of a set of maximum entropy classifiers: a separate classifier is trained for each (sufficiently frequent) source-side lemma. In this way the estimates of translation probabilities can be sensitive to a large number of features derived from the source sentence (including non-local features, features making use of sentence syntactic structure, etc.). When integrated into English-to-Czech dependency-based translation scenario implemented in the TectoMT framework, the new translation model significantly outperforms the baseline model (MLE) in terms of BLEU. The performance is further boosted in a configuration inspired by Hidden Tree Markov Models which combines the maximum entropy translation model with the target-language dependency tree model.

1 Introduction

The principle of maximum entropy states that, given known constraints, the probability distribution which best represents the current state of knowledge is the one with the largest entropy. Maximum entropy models based on this principle have been widely used in Natural Language Processing, e.g. for tagging (Ratnaparkhi, 1996), parsing (Charniak, 2000), and named entity recognition (Bender et al., 2003). Maximum entropy models have the following form

$$p(y|x) = \frac{1}{Z(x)} exp \sum_i \lambda_i f_i(x,y)$$

where f_i is a feature function, λ_i is its weight, and $Z(x)$ is the normalizing factor

$$Z(x) = \sum_y exp \sum_i \lambda_i f_i(x,y)$$

In statistical machine translation (SMT), translation model (TM) $p(t|s)$ is the probability that the string t from the target language is the translation of the string s from the source language. Typical approach in SMT is to use backward translation model $p(s|t)$ according to Bayes' rule and noisy-channel model. However, in this paper we deal only with the forward (direct) model.[1]

The idea of using maximum entropy for constructing forward translation models is not new. It naturally allows to make use of various features potentially important for correct choice of target-language expressions. Let us adopt a motivating example of such a feature from (Berger et al., 1996) (which contains the first usage of maxent translation model we are aware of): "If *house* appears within the next three words (e.g., the phrases *in the house* and *in the red house*), then *dans* might be a more likely [French] translation [of *in*]."

Incorporating non-local features extracted from the source sentence into the standard noisy-channel model in which only the backward translation model is available, is not possible. This drawback of the noisy-channel approach is typically compensated by using large target-language n-gram models, which can – in a result – play a role similar to that of a more elaborate (more context sensitive) forward translation model. However, we expect that it would be more beneficial to exploit both the parallel data and the monolingual data in a more balance fashion, rather than extract only a reduced amount of information from the parallel data and compensate it by large language model on the target side.

[1] A backward translation model is used only for pruning training data in this paper.

Proceedings of the Joint 5th Workshop on Statistical Machine Translation and MetricsMATR, pages 207–212,
Uppsala, Sweden, 11-16 July 2010. ©2010 Association for Computational Linguistics

A deeper discussion on the potential advantages of maximum entropy approach over the noisy-channel approach can be found in (Foster, 2000) and (Och and Ney, 2002), in which another successful applications of maxent translation models are shown. Log-linear translation models (instead of MLE) with rich feature sets are used also in (Ittycheriah and Roukos, 2007) and (Gimpel and Smith, 2009); the idea can be traced back to (Papineni et al., 1997).

What makes our approach different from the previously published works is that

1. we show how the maximum entropy translation model can be used in a dependency framework; we use deep-syntactic dependency trees (as defined in the Prague Dependency Treebank (Hajič et al., 2006)) as the transfer layer,
2. we combine the maximum entropy translation model with target-language dependency tree model and use tree-modified Viterbi search for finding the optimal lemmas labeling of the target-tree nodes.

The rest of the paper is structured as follows. In Section 2 we give a brief overview of the translation framework TectoMT in which the experiments are implemented. In Section 3 we describe how our translation models are constructed. Section 4 summarizes the experimental results, and Section 5 contains a summary.

2 Translation framework

We use tectogrammatical (deep-syntactic) layer of language representation as the transfer layer in the presented MT experiments. Tectogrammatics was introduced in (Sgall, 1967) and further elaborated within the Prague Dependency Treebank project (Hajič et al., 2006). On this layer, each sentence is represented as a tectogrammatical tree, whose main properties (from the MT viewpoint) are following: (1) nodes represent autosemantic words, (2) edges represent semantic dependencies (a node is an argument or a modifier of its parent), (3) there are no functional words (prepositions, auxiliary words) in the tree, and the autosemantic words appear only in their base forms (lemmas). Morphologically indispensable categories (such as number with nouns or tense with verbs, but not number with verbs as it is only imposed by agreement) are stored in separate node attributes (grammatemes).

The intuition behind the decision to use tectogrammatics for MT is the following: we believe that (1) tectogrammatics largely abstracts from language-specific means (inflection, agglutination, functional words etc.) of expressing non-lexical meanings and thus tectogrammatical trees are supposed to be highly similar across languages,[2] (2) it enables a natural transfer factorization,[3] (3) and local tree contexts in tectogrammatical trees carry more information (especially for lexical choice) than local linear contexts in the original sentences.[4]

In order to facilitate transfer of sentence 'syntactization', we work with tectogrammatical nodes enhanced with the formeme attribute (Žabokrtský et al., 2008), which captures the surface morphosyntactic form of a given tectogrammatical node in a compact fashion. For example, the value n:před+4 is used to label semantic nouns that should appear in an accusative form in a prepositional group with the preposition před in Czech. For English we use formemes such as n:subj (semantic noun (SN) in subject position), n:for+X (SN with preposition *for*), n:X+ago (SN with postposition *ago*), n:poss (possessive form of SN), v:because+fin (semantic verb (SV) as a subordinating finite clause introduced by *because*), v:without+ger (SV as a gerund after *without*), adj:attr (semantic adjective (SA) in attributive position), adj:compl (SA in complement position).

We have implemented our experiments in the TectoMT software framework, which already offers tool chains for analysis and synthesis of Czech and English sentences (Žabokrtský et al., 2008). The translation scenario proceeds as follows.

1. The input English text is segmented into sentences and tokens.
2. The tokens are lemmatized and tagged with Penn Treebank tags using the Morce tagger (Spoustová et al., 2007).

[2] This claim is supported by error analysis of output of tectogrammatics-based MT system presented in (Popel and Žabok/rtský, 2009), which shows that only 8 % of translation errors are caused by the (obviously too strong) assumption that the tectogrammatical tree of a sentence and the tree representing its translation are isomorphic.

[3] Morphological categories can be translated almost independently from lemmas, which makes parallel training data 'denser', especially when translating from/to a language with rich inflection such as Czech.

[4] Recall the house-is-somewhere-around feature in the introduction; again, the fact that we know the dominating (or dependent) word should allow to construct a more compact translation model, compared to n-gram models.

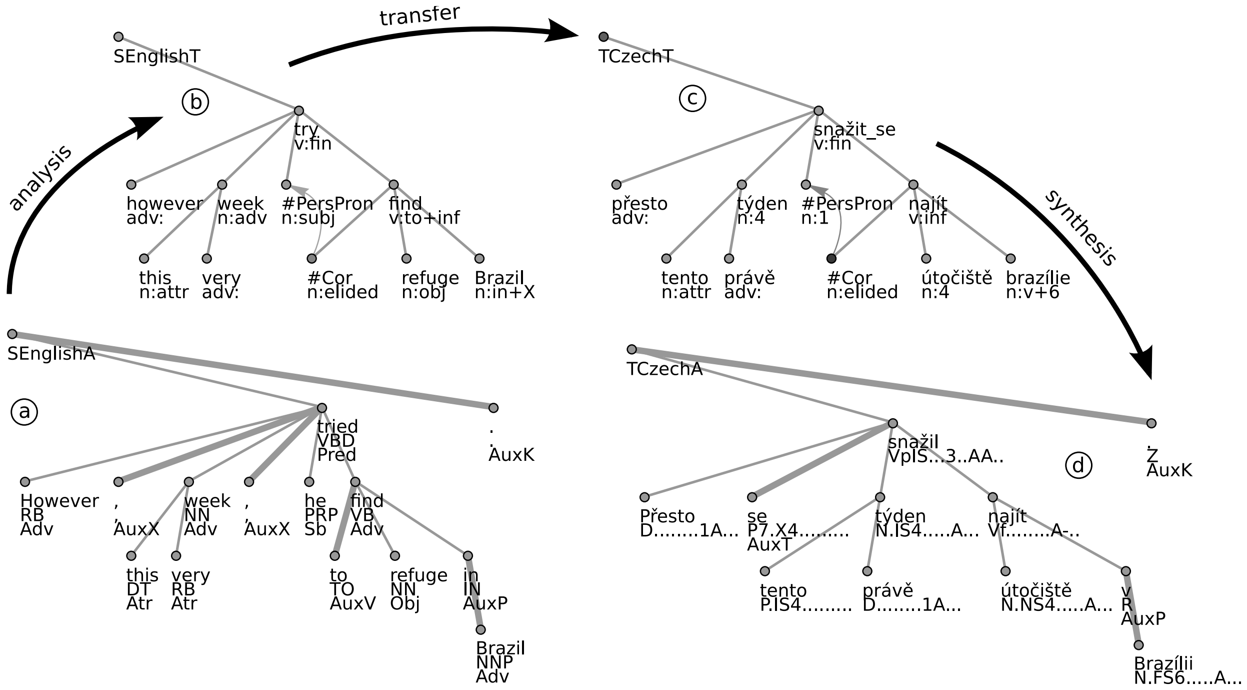

Figure 1: Intermediate sentence representations when translating the English sentence *"However, this very week, he tried to find refuge in Brazil."*, leading to the Czech translation *"Přesto se tento právě týden snažil najít útočiště v Brazílii."*.

3. Then the Maximum Spanning Tree parser (McDonald et al., 2005) is applied and a surface-syntax dependency tree (analytical tree in the PDT terminology) is created for each sentence (Figure 1a).

4. This tree is converted to a tectogrammatical tree (Figure 1b). Each autosemantic word with its associated functional words is collapsed into a single tectogrammatical node, labeled with lemma, formeme, and semantically indispensable morphologically categories; coreference is also resolved. Collapsing edges are depicted by wider lines in the Figure 1a.

5. The transfer phase follows, whose most difficult part consists in labeling the tree with target-side lemmas and formemes[5] (changes of tree topology are required relatively infrequently). See Figure 1c.

6. Finally, surface sentence shape (Figure 1d) is synthesized from the tectogrammatical tree, which is basically a reverse operation for the tectogrammatical analysis: adding punctuation and functional words, spreading morphological categories according to grammatical agreement, performing inflection (using Czech morphology database (Hajič, 2004)), arranging word order etc.

3 Training the two models

In this section we describe two translation models used in the experiments: a baseline translation model based on maximum likelihood estimates (3.2), and a maximum entropy based model (3.3). Both models are trained using the same data (3.1).

In addition, we describe a target-language tree model (3.4), which can be combined with both the translation models using the Hidden Tree Markov Model approach and tree-modified Viterbi search, similarly to the approach of (Žabokrtský and Popel, 2009).

3.1 Data preprocessing common for both models

We used Czech-English parallel corpus CzEng 0.9 (Bojar and Žabokrtský, 2009) for training the translation models. CzEng 0.9 contains about 8 million sentence pairs, and also their tectogrammatical analyses and node-wise alignment.

[5] In this paper we focus on using maximum entropy for translating lemmas, but it can be used for translating formemes as well.

We used only trees from training sections (about 80 % of the whole data), which contain around 30 million pairs of aligned tectogrammatical nodes.

From each pair of aligned tectogrammatical nodes, we extracted triples containing the source (English) lemma, the target (Czech) lemma, and the feature vector.

In order to reduce noise in the training data, we pruned the data in two ways. First, we disregarded all triples whose lemma pair did not occur at least twice in the whole data. Second, we computed forward and backward maximum likelihood (ML) translation models (target lemma given source lemma and vice versa) and deleted all triples whose probability according to one of the two models was lower than the threshold 0.01.

Then the forward ML translation model was reestimated using only the remaining data.

For a given pair of aligned nodes, the feature vector was of course derived only from the source-side node or from the tree which it belongs to. As already mentioned in the introduction, the advantage of the maximum entropy approach is that a rich and diverse set of features can be used, without limiting oneself to linearly local context. The following features (or, better to say, feature templates, as each categorical feature is in fact converted to a number of 0-1 features) were used:

- formeme and morphological categories of the given node,
- lemma, formeme and morphological categories of the governing node,
- lemmas and formemes of all child nodes,
- lemmas and formemes of the nearest linearly preceding and following nodes.

3.2 Baseline translation model

The baseline TM is basically the ML translation model resulting from the previous section, linearly interpolated with several translation models making use of regular word-formative derivations, which can be helpful for translating some less frequent (but regularly derived) lemmas. For example, one of the derivation-based models estimates the probability $p(\textit{zajímavě}|\textit{interestingly})$ (possibly unseen pair of deadjectival adverbs) by the value of $p(\textit{zajímavý}|\textit{interesting})$. More detailed description of these models goes beyond the scope of this paper; their weights in the interpolation are very small anyway.

3.3 MaxEnt translation model

The MaxEnt TM was created as follows:

1. training triples (source lemma, target lemma, feature vector) were disregarded if the source lemma was not seen at least 50 times (only the baseline model will be used for such lemmas),
2. the remaining triples were grouped by the English lemma (over 16 000 groups),
3. due to computational issues, the maximum number of triples in a group was reduced to 1000 by random selection,
4. a separate maximum entropy classifier was trained for each group (i.e., one classifier per source-side lemma) using `AI::MaxEntropy` Perl module,[6]
5. due to the more aggressive pruning of the training data, coverage of this model is smaller than that of the baseline model; in order not to loose the coverage, the two models were combined using linear interpolation (1:1).

Selected properties of the maximum entropy translation model (before the linear interpolation with the baseline model) are shown in Figure 2. We increased the size of the training data from 10 000 training triples up to 31 million and evaluated three relative quantities characterizing the translation models:

- *coverage* - relative frequency of source lemmas for which the translation model offers at least one translation,
- *first* - relative frequency of source lemmas for which the target lemmas offered as the first by the model ($argmax$) are the correct ones,
- *oracle* - relative frequency of source lemmas for which the correct target lemma is among the lemmas offered by the translation model.

As mentioned in Section 3.1, there are context features making use both of local linear context and local tree context. After training the MaxEnt model, there are about 4.5 million features with non-zero weight, out of which 1.1 million features

[6] `http://search.cpan.org/perldoc?AI::MaxEntropy`

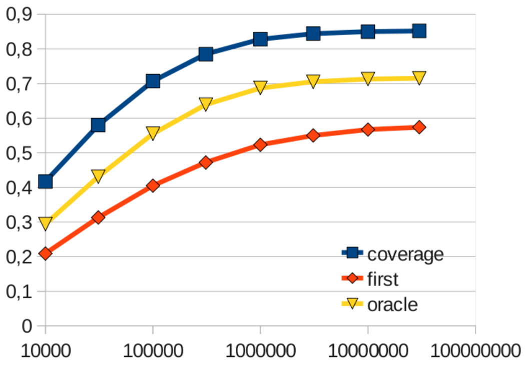

Figure 2: Three measures characterizing the MaxEnt translation model performance, depending on the training data size. Evaluated on aligned node pairs from the `dtest` portion of CzEng 0.9.

are derived from the linear context and 2.4 million features are derived from the tree context. This shows that the MaxEnt translation model employs the dependency structure intensively.

A preliminary analysis of feature weights seems to support our intuition that the linear context is preferred especially in the case of more stable collocations. For example, the most important features for translating the lemma *bare* are based on the lemma of the following noun: target lemma *bosý* (barefooted) is preferred if the following noun on the source side is *foot*, while *holý* (naked, unprotected) is preferred if *hand* follows.

The contribution of dependency-based features can be illustrated on translating the word *drop*. The greatest weight for choosing *kapka* (a droplet) as the translation is assigned to the feature capturing the presence of a node with formeme n:of+X among the node's children. The greatest weights in favor of *odhodit* (throw aside) are assigned to features capturing the presence of words such as *gun* or *weapon*, while the greatest weights in favor of *klesnout* (to come down) are assigned to features saying that there is the lemma *percent* or the percent sign among the children.

Of course, the lexical choice is influenced also by the governing lemmas, as can be illustrated with the word *native*. One can find a high-value feature for *rodilý* (native-born) saying that the source-side parent is *speaker*; similarly for *mateřský* (mother) with governing *tongue*, and *rodný* (home) with *land*.

Linear and tree features are occasionally used simultaneously: there are high-valued positive weights for translating *order* as *objednat* (reserve, give an order for st.) assigned both to tree-based features saying that there are words such as *pizza*, *meal* or *goods* and to linear features saying that the very following word is *some* or *two*.

configuration	BLEU	NIST
baseline TM	10.44	4.795
MaxEnt TM	11.77	5.135
baseline TM + TreeLM	11.77	5.038
MaxEnt TM + TreeLM	12.58	5.250

Table 1: BLEU and NIST evaluation of four configurations of our MT system; the WMT 2010 test set was used.

3.4 Target-language tree model

Although the MaxEnt TM captures some contextual dependencies that are covered by language models in the standard noisy-channel SMT, it may still be beneficial to exploit target-language models, because these can be trained on huge monolingual corpora. We use a target-language dependency tree model differing from standard n-gram model in two aspects:

- it uses tree context instead of linear context,
- it predicts tectogrammatical attributes (lemmas and formemes) instead of word forms.

In particular, our target-language tree model (TreeLM) predicts the probability of node's lemma and formeme given its parent's lemma and formeme. The optimal (lemma and formeme) labeling is found by tree-modified Viterbi search; for details see (Žabokrtský and Popel, 2009).

4 Experiments

When included into the above described translation scenario, the MaxEnt TM outperforms the baseline TM, be it used together with or without TreeLM. The results are summarized in Table 1. The improvement is statistically significant according to paired bootstrap resampling test (Koehn, 2004). In the configuration without TreeLM the improvement is greater (1.33 BLEU) than with TreeLM (0.81 BLEU), which confirms our hypothesis that MaxEnt TM captures some of the contextual dependencies resolved otherwise by language models.

5 Conclusions

We have introduced a maximum entropy translation model in dependency-based MT which enables exploiting a large number of feature functions in order to obtain more accurate translations. The BLEU evaluation proved significant improvement over the baseline solution based on the translation model with maximum likelihood estimates. However, the performance of this system still below the state of the art (which is around BLEU 16 for the English-to-Czech direction).

Acknowledgments

This research was supported by the grants MSM0021620838, MŠMT ČR LC536, FP7-ICT-2009-4-247762 (Faust), FP7-ICT-2007-3-231720 (EuroMatrix Plus), GA201/09/H057, and GAUK 116310. We thank two anonymous reviewers for helpful comments.

References

Oliver Bender, Franz Josef Och, and Hermann Ney. 2003. Maximum entropy models for named entity recognition. In *Proceedings of CoNLL 2003*, pages 148–151.

Adam L. Berger, Vincent J. Della Pietra, and Stephen A. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational linguistics*, 22(1):39–71.

Ondřej Bojar and Zdeněk Žabokrtský. 2009. CzEng 0.9, Building a Large Czech-English Automatic Parallel Treebank. *The Prague Bulletin of Mathematical Linguistics*, 92:63–83.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the 1st North American chapter of the ACL conference*, pages 132–139, San Francisco, USA.

George Foster. 2000. A maximum entropy/minimum divergence translation model. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 45–52, Morristown, USA. Association for Computational Linguistics.

Kevin Gimpel and Noah A. Smith. 2009. Feature-rich translation by quasi-synchronous lattice parsing. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 219–228, Morristown, USA. Association for Computational Linguistics.

Jan Hajič et al. 2006. Prague Dependency Treebank 2.0. CD-ROM, Linguistic Data Consortium, LDC Catalog No.: LDC2006T01, Philadelphia.

Jan Hajič. 2004. *Disambiguation of Rich Inflection – Computational Morphology of Czech*. Charles University – The Karolinum Press, Prague.

Abraham Ittycheriah and Salim Roukos. 2007. Direct translation model 2. In Candace L. Sidner, Tanja Schultz, Matthew Stone, and ChengXiang Zhai, editors, *HLT-NAACL*, pages 57–64. The Association for Computational Linguistics.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP*, volume 4, pages 388–395.

Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajič. 2005. Non-projective dependency parsing using spanning tree algorithms. In *Proceedings of HLT / EMNLP*, pages 523–530, Vancouver, Canada.

Franz Josef Och and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of ACL*, pages 295–302.

Kishore A. Papineni, Salim Roukos, and Todd R. Ward. 1997. Feature-based language understanding. In *European Conference on Speech Communication and Technology (EUROSPEECH)*, pages 1435–1438, Rhodes, Greece, September.

Martin Popel and Zdeněk Žabok/rtský. 2009. Improving English-Czech Tectogrammatical MT. *The Prague Bulletin of Mathematical Linguistics*, (92):1–20.

Adwait Ratnaparkhi. 1996. A maximum entropy model for part-of-speech tagging. In *In Proceedings of EMNLP'96*, pages 133–142.

Petr Sgall. 1967. *Generativní popis jazyka a česká deklinace*. Academia, Prague.

Drahomíra Spoustová, Jan Hajič, Jan Votrubec, Pavel Krbec, and Pavel Květoň. 2007. The Best of Two Worlds: Cooperation of Statistical and Rule-Based Taggers for Czech. In *Proceedings of the Workshop on Balto-Slavonic Natural Language Processing, ACL 2007*, pages 67–74, Praha.

Zdeněk Žabokrtský and Martin Popel. 2009. Hidden markov tree model in dependency-based machine translation. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 145–148, Suntec, Singapore.

Zdeněk Žabokrtský, Jan Ptáček, and Petr Pajas. 2008. TectoMT: Highly Modular MT System with Tectogrammatics Used as Transfer Layer. In *Proceedings of the 3rd Workshop on Statistical Machine Translation, ACL*, pages 167–170.