

The application of structured learning in natural language processing

Yizhao Ni · Craig Saunders ·
Sandor Szedmak · Mahesan Niranjan

Received: 30 October 2009 / Accepted: 22 April 2010
© Springer Science+Business Media B.V. 2010

Abstract We propose a structured learning approach, max-margin structure (MMS), which is targeted at natural language processing (NLP) tasks. The architecture of our approach is shown to capture structural aspects of the problem domains, leading to demonstrable performance improvements on two NLP tasks: part-of-speech tagging and statistical machine translation (SMT). We present a perceptron-based online learning algorithm to train the model and demonstrate desirable computational scaling behavior over traditional optimisation methods.

Keywords Statistical machine translation · Max-margin structure

1 Introduction

Machine learning techniques have shown good performance in many diverse application areas where complex information needs to be learned from available data. This is very apparent also in natural language processing (NLP) and statistical machine translation (SMT) in recent years where methods from the field of machine learning have been increasingly applied. A particularly new and powerful paradigm in machine learning, *structured prediction*, is especially applicable to these tasks. Using this new paradigm we address two specific problems in NLP: *phrase translation* in *statistical machine translation* and *part-of-speech tagging*. In both we wish to devise a methodology that is specifically aimed at capturing structure in the output predictions.

Y. Ni (✉) · S. Szedmak · M. Niranjan
ISIS Group, School of Electronics and Computer Science, University of Southampton,
Southampton, SO17 1BJ, UK
e-mail: yizhao.ni@googlemail.com

C. Saunders
Xerox Research Centre Europe, 6 Chemin de Maupertuis, 38240 Meylan, France

1.1 Phrase translation

One state-of-the-art approach to machine translation is phrase-based SMT which involves the creation of a number of sub-models $\{h_m\}$, namely *Phrase Translation Probability* (PTP) model, language model and phrase reordering model. Then an SMT decoder is employed to solve the task where each source sentence \mathbf{f} is segmented into a sequence of I phrases $\bar{\mathbf{f}}^I$ and translated into a target sequence $\bar{\mathbf{e}}^I$ so as to maximise the posterior probability $\bar{\mathbf{e}}^I = \arg \max_{\hat{\mathbf{e}}^I \in E} \{p(\hat{\mathbf{e}}^I | \bar{\mathbf{f}}^I)\}$. In general $p(\hat{\mathbf{e}}^I | \bar{\mathbf{f}}^I)$ is modeled with a log-linear maximum entropy framework (Berger et al. 1996) to integrate all the sub-models h_m

$$p(\hat{\mathbf{e}}^I | \bar{\mathbf{f}}^I) = \frac{\exp(\sum_m \lambda_m h_m(\hat{\mathbf{e}}^I, \bar{\mathbf{f}}^I))}{\sum_{I', \hat{\mathbf{e}}^{I'}} \exp(\sum_m \lambda_m h_m(\hat{\mathbf{e}}^{I'}, \bar{\mathbf{f}}^I))}$$

As the denominator only depends on the source phrase sequence $\bar{\mathbf{f}}^I$, it is usually discarded and the solution is also represented as $\bar{\mathbf{e}}^I = \arg \max_{\hat{\mathbf{e}}^I \in E} \{\exp(\sum_m \lambda_m h_m(\hat{\mathbf{e}}^I, \bar{\mathbf{f}}^I))\}$, which is equivalent to searching a Viterbi-best string path according to the decoding information provided by individual sub-models. Therefore, the design of the sub-models used is critical for the success of an SMT system.

In this paper, we focus on developing a crucial component of SMT—the *Phrase Translation Probability* (PTP) model, whose main task is to predict the *target translation* from a finite candidate pool for a *source phrase*, based on the linguistic environment (or context) in which the source phrase is embedded (see Fig. 1a). Although a better phrase translation prediction can greatly improve the translation quality as well as ease the workload of the SMT decoder, the PTP model has changed little over the past few years. Following the assumption that the target phrase translations are conditionally independent given their source phrases, the traditional PTP model assigns a phrase translation probability for each phrase pair (\bar{f}_j, \bar{e}_i) using maximum likelihood estimation (MLE) $p(\bar{f}_j, \bar{e}_i) = \frac{\text{count}(\bar{f}_j, \bar{e}_i)}{\sum_{\bar{e}} \text{count}(\bar{f}_j, \bar{e})}$. This model is commonly used in current SMT systems (Koehn 2004; Koehn et al. 2005), but it has two major limitations: first, the phrase translation probability only depends on the frequency of the phrase pairs, the sentence context where phrases occur is completely ignored; secondly, the variance of MLE for low-frequency phrase pairs is considerably large, making the predictions over-fit the training data.

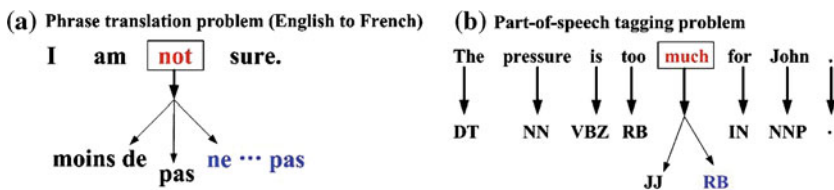


Fig. 1 Examples of two structured prediction tasks: phrase translation (a) and POS tagging (b). The target phrase (or POS tag) in blue is the correct output

As machine learning techniques become more pervasive in SMT, several discriminative methods have been applied for the PTP model. For example, a *word sense disambiguation* (WSD) model is proposed in [Vickrey et al. \(2005\)](#), that learns the translations of words based on the context, syntax and lemmatic information. A similar WSD model (namely global lexical selection model) is also used in [Bangalore et al. \(2007\)](#), in which a regularised maximum entropy classifier is trained on the bag-of-words features for resolving ambiguity in lexical selection. Extending from words to phrases, [Carpuat and Wu \(2007\)](#) uses an ensemble of four combined WSD models to predict phrase translation probabilities, which are then integrated into a traditional SMT system ([Koehn 2004](#)) to help the phrase disambiguation. Alternatively, [Giménez and Màrquez \(2007\)](#) deals with the phrase translation as a classification problem and uses the support vector machine (SVM) technique to perform the translation prediction. Although these approaches take some source context into account, the relationships between the target translations are still not considered at all. Theoretically, the structure of the target translations can be learned by state-of-the-art structured prediction techniques such as a structured SVM ([Tsochantaridis et al. 2004](#)); however in practice the runtime usually makes it infeasible to apply these methods to even medium-sized data sets. Therefore, we use the idea of structured SVMs that allows more flexible margins between classes, but apply a perceptron-based algorithm in order to reduce the computational complexity. Our aim is to show that it is practical to apply this *max-margin structure* (MMS) model to the MT field and produce reasonable results.

1.2 Part-of-speech (POS) tagging

POS tagging can be viewed as the process of “translating” words in a text into a particular part of speech. Since the translation from words to POS tags are one by one without word reordering (see Fig. 1b), it can be viewed as a simplified form of machine translation that only consists of a phrase translation probability model.

Among recent top performing methods for automatic assignment of POS tagging, *hidden markov models* ([Brants 2000](#)), *maximum entropy models* ([Toutanova et al. 2003](#)) and *conditional random field models* ([Lafferty et al. 2001](#)) are very popular. In these methods, a tag t_i assigned to a word f_i with the context feature function ϕ is connected via a conditional probability $p(t_i|\phi(f_i))$, while different parametric forms are applied for modeling this probability. The models are then “generating” the POS tag sequence for a given sentence by maximising the sequence probability $\prod_{i=1}^N p(t_i|\phi(f_i))$. Alternatively, one can view POS tagging as a multi-class classification problem, which predicts each word f_i 's tag label t_i in accordance with its linguistic features: $t_i = \arg \max_{\hat{t}} \mathbf{w}^T \phi(f_i, \hat{t})$. In this way, general classification techniques such as SVMs can be applied ([Joachims 1999](#)).

In POS tagging there are several problematic cases that come from ambiguous words in speech. For example, the word “good” in the phrase “good strategy” is an adjective (JJ) while in “the common good” it is a noun (NN). Current methods such as ([Toutanova et al. 2003](#)) try to solve the problematic cases by exploring richer feature sets, such as grammatical features and lemmas. However, such feature extensions are usually expensive to collect in advance.

In contrast to the above, we aim at improving the performance by exploiting the potential relationships between the tags for ambiguous words. We apply a word disambiguation technique to this problem and compare four learning methodologies—the *maximum entropy* (ME) framework, the *support vector machine* (SVM) technique¹, the *stochastic gradient descent* (SGD) classifier (Shalev-Shwartz et al. 2007) and the *max-margin structure* (MMS) model—on a POS classification task. In this paper we treat POS tagging as an MT task with a specific target language made of POS tags, where the goal is to show the effectiveness of the MMS model and pave the way for its application to the full machine translation task.

The remaining parts of this paper are organised as follows: a general framework of the MMS model² is given in Section 2, which specifies the motivations of utilising the structured prediction and the learning algorithm. Then in Section 3, we demonstrate the procedure for feature extraction and the training scheme of the MMS model. Section 4 evaluates the performance of the MMS model on POS tagging and MT tasks. Finally, we draw conclusions and mention areas for future work in Section 5.

2 Phrase translation with structure exploitation

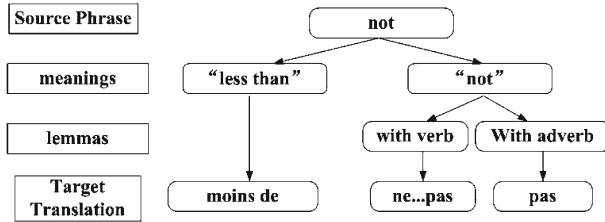
We define the source phrase as \bar{f}_j^n with \bar{f} denoting the phrase label, j denoting the phrase position in the phrase sequence $\bar{\mathbf{f}}^I$ and n denoting the n -th example. A similar notation \bar{e}_i^n is used for target phrases. Each unique source phrase \bar{f} is assigned to a cluster $\Omega_{\bar{f}}$ that includes all possible target translations (candidates) and the number of candidates is denoted as $C_{\bar{f}}$. Figure 2 demonstrates an English-to-French translation example, in which $\bar{f} = \text{“not”}$, $\Omega_{\bar{f}} = \{\text{“moins de”}, \text{“ne...pas”}, \text{“pas”}\}$ and $n = 1, \dots, 3$. Whenever this can be done without loss of clarity, the source and the target phrase examples are also abbreviated as \bar{f}^n and \bar{e}^n .

In our PTP model, we assign a separate sub-model for each unique source phrase \bar{f} . Assume a set of training instances $\mathcal{S}_{\bar{f}} = \{(\bar{f}^n, \bar{e}^n)\}_{n=1}^{N_{\bar{f}}}$ with the same source phrase \bar{f} , each of which consists of a structured feature vector $\phi(\bar{f}^n, \bar{e}^n) \in \mathbb{R}^{d \cdot C_{\bar{f}}}$ with d denoting the dimension of linguistic feature space. Then the goal is to learn a linear evaluation function $F := \mathbf{w}^T \phi(\bar{f}^n, \bar{e}^n) \rightarrow \mathbb{R}$ that can “generate” an appropriate translation score for (\bar{f}^n, \bar{e}^n) .

Instead of applying MLE, we adopt a discriminative framework, the max-margin formulation of Taskar et al. (2003), to find an operator $\mathbf{w} \in \mathbb{R}^{d \cdot C_{\bar{f}}}$ such that $\arg \max_{c \in \Omega_{\bar{f}}} \mathbf{w}^T \phi(\bar{f}^n, c) \approx \bar{e}^n, \forall n$. This is equivalent to minimising a risk function $J(\mathbf{w})$, which corresponds to the sum of the classification errors associated with the translation candidates

¹ The SVM technique mentioned in this paper is the classic SVM optimisation (Joachims 1999), which uses Sequential Minimal Optimisation as the SVM solver.

² Since we regard POS tagging as a special case of MT, in the system description our notation will follow the general MT notation as used in Koehn et al. (2005).



Example 1:

Source Sentence: Not five minutes ago.

Translation: Il y a moins de cinq minutes.

Example 2:

Source Sentence: I am not sure.

Translation: Je ne suis pas sûr.

Example 3:

Source Sentence: There is not even a pub.

Translation: Il n’y a même pas de bar.

Fig. 2 An English-to-French translation example. The latent connections between target translations are displayed in two levels: “meaning” and “lemma”

$$J(\mathbf{w}) = \frac{1}{N_{\bar{f}}} \sum_{n=1}^{N_{\bar{f}}} \rho(\bar{f}^n, \bar{e}^n, \mathbf{w}) + \frac{\lambda}{2} \|\mathbf{w}\|^2 \tag{1}$$

where ρ is a specific loss function and $\lambda \geq 0$ is a regularisation parameter. The risk function (1) can generate a set of discriminative models with different loss functions $\rho(\bar{f}^n, \bar{e}^n, \mathbf{w})$. For example, we can regard the phrase translation merely as a multi-class classification problem and let $\rho(\bar{f}^n, \bar{e}^n, \mathbf{w}) = \max\{0, 1 - \xi(\bar{f}^n, \bar{e}^n) - \mathbf{w}^T \phi(\bar{f}^n, \bar{e}^n)\}$, where $\xi(\bar{f}^n, \bar{e}^n)$ is a slack variable with similar function to an SVM. Essentially the slack variable allows some small error in training and can be used to trade-off against the regulariser; if a lot of slack is allowed, the algorithm performs poorly, if no slack variables are allowed then the algorithm could over-fit. If we view the model in this way, then it essentially is equivalent to a form of one-class SVM (Schölkopf et al. 2001).

As translations for the same source phrase tend to be interdependent, introducing flexible margins to separate different translation candidates sounds more reasonable. Consider Fig. 2, if the phrase “not” is translated into “pas” instead of “ne...pas”, intuitively the loss should be smaller than when it is translated into “moins de”. Indeed, the output (target translation) domain has an inherent structure (e.g. surface form and lemma) and the loss function should respect this. Hence, we model the phrase translation task as a *max-margin structure* (MMS) problem and apply a soft-margin loss on the structured labels:

$$\rho(\bar{f}^n, \bar{e}^n, \mathbf{w}) = \max\{0, \max_{c \neq \bar{e}^n} [\Delta(c, \bar{e}^n) + \mathbf{w}^T \phi(\bar{f}^n, c)] - \mathbf{w}^T \phi(\bar{f}^n, \bar{e}^n)\} \tag{2}$$

where $\Delta(c, \bar{e}^n)$ is applied to measure the “distance” between a pseudo candidate c and the correct translation \bar{e}^n . Theoretically, this loss requires that the pseudo candidate c

which is “far away” from the true translation \bar{e}^n must be classified with a large margin $\Delta(c, \bar{e}^n)$ while nearby candidates are allowed to be classified with a smaller margin.

This formulation is similar to the structured SVM (Tsochantaridis et al. 2004), which solves the following optimisation problem

$$\begin{aligned} \min_{\mathbf{w}, \xi} \quad & \frac{1}{2} \|\mathbf{w}\|^2 + \frac{C}{N} \sum_{n=1}^N \xi_n \\ \text{s.t.} \quad & \forall n, c \in \{\Omega_{\bar{f}} \setminus \bar{e}^n\} : \\ & \mathbf{w}^T \phi(\bar{f}^n, \bar{e}^n) - \mathbf{w}^T \phi(\bar{f}^n, c) \geq 1 - \frac{\xi_n}{\Delta(c, \bar{e}^n)} \\ & \forall n \quad \xi_n \geq 0 \end{aligned}$$

However, instead of loading $NC_{\bar{f}}$ constraints, MMS only consists of N constraints and hence speeds up the training procedure.

A variety of approaches have been suggested to evaluate the “distance” between strings, such as the “bag-of-words” method for string kernels (Shawe-Taylor and Cristianini 2004). Ideally, the measure function $\Delta(c, \bar{e}^n)$ should respect all aspects of influence on candidate connections, which is hard to achieve in practice however. To simplify the computation, we use a generalisation of the hamming distance—Levenshtein distance, that measures the minimum number of modifications required to change one string into another. The algorithm can be found in Gusfield (1997) and the distance function is of the form

$$\Delta(c, \bar{e}^n) = \frac{LevDist(c, \bar{e}^n)}{\max_{c' \in \Omega_{\bar{f}}} LevDist(c', \bar{e}^n)} \tag{3}$$

where $LevDist(c, \bar{e}^n)$ returns the Levenshtein distance between strings c and \bar{e}^n . This distance value satisfies $\Delta(c, \bar{e}^n) \in (0, 1]$ with $\Delta(c, \bar{e}^n) = 1$ indicating that c is the farthest from \bar{e}^n among the candidates. Note other distances, such as those based on factored representations (Hofmann et al. 2003; Cai and Hofmann 2004), could also be considered.

2.1 Perceptron-based structured learning (PSL)

If we do not consider the regularisation term in (1) (i.e. $\lambda = 0$), we can use a perceptron-based structured learning (PSL) algorithm to tune the parameters \mathbf{w} . Note that this algorithm is an extension of that provided by Collins (2002) where in the latter case the output structure is ignored (i.e. $\Delta(c, \bar{e}^n) = 1, \forall c$).

Table 1 gives the algorithm and shows that the computational complexity is $O(N_{\bar{f}}dC_{\bar{f}})$, while the complexity of multi-class SVM is somewhere between $O(N_{\bar{f}}^2d + N_{\bar{f}}C_{\bar{f}})$ and $O(N_{\bar{f}}^2d + N_{\bar{f}}^2C_{\bar{f}})$ (Bishop 2006). Since in practice the number of classes $C_{\bar{f}}$ is much smaller than the number of examples $N_{\bar{f}}$, this makes PSL substantially faster than the multi-class SVM used in Giménez and Màrquez (2007) and obviously the structured SVM proposed in Tsochantaridis et al. (2004). This time efficiency is also verified by the POS tagging experiment results shown in Table 3.

Table 1 Perceptron-based structured learning (PSL) algorithm

Input of the learner: Samples $\mathcal{S}_{\bar{f}} = \{(\bar{f}^n, \bar{e}^n)\}_{n=1}^{N_{\bar{f}}}$, learning rate η

Initialisation: $t = 0; \mathbf{w}_t = \mathbf{0};$

Repeat

randomly sample $(\bar{f}^n, \bar{e}^n) \in \mathcal{S}_{\bar{f}}$ **do**

$V = \max_{c \neq \bar{e}^n} \{\Delta(c, \bar{e}^n) + \mathbf{w}_t^T \phi(\bar{f}^n, c)\}$

$c^* = \arg \max_{c \neq \bar{e}^n} \{\Delta(c, \bar{e}^n) + \mathbf{w}_t^T \phi(\bar{f}^n, c)\}$

if $\mathbf{w}_t^T \phi(\bar{f}^n, \bar{e}^n) < V$ **then**

$\mathbf{w}_{t+1} = \mathbf{w}_t + \eta(\phi(\bar{f}^n, \bar{e}^n) - \phi(\bar{f}^n, c^*))$

$t = t + 1$

end if

until converge

Output of the learner: $\mathbf{w}_{t+1} \in \mathbb{R}^{d \cdot C_{\bar{f}}}$

Although PSL does not adopt the regularisation term, implying a potential risk of over-fitting, the *early-stopping strategy*,³ which involves a careful design of the maximum number of iterations, can usually help to avoid this problem. If one wished to add regularisation to the model to further guard against over-fitting, one could apply methods such as ALMA (Gentile 2001) or NORMA (Kivinen et al. 2004). However, the requirement of normalising \mathbf{w} at each step makes the implementation intractable for a large learning problem. As an alternative, the risk function (1) can be reformulated as a min–max optimisation problem which can be solved by a benchmark-based extra-gradient algorithm. Under mild conditions, the algorithm is guaranteed to converge linearly to a solution of \mathbf{w}^* (Taskar et al. 2006).

3 Training procedure

In this section, we describe two key steps for the method: feature extraction and model training.

3.1 Feature extraction

Following Vickrey et al. (2005), we consider different kinds of information extracted from the phrase environment (see Fig. 3). The types of features we used are depicted in Table 2.

To specify the difference with respect to each source environment position d_z , we express the features as $\phi_u(s_p^{|u|}) = \delta(s_p^{|u|}, u)$, with the indicator function $\delta(\cdot, \cdot)$, $p = \{j_l - d_l, \dots, j_l, j_r, \dots, j_r + d_r\}$ and string $s_p^{|u|} = [f_p, \dots, f_{p+|u|}]$ with $|u|$ denoting

³ The strategy selects the maximum number of iterations and the learning rate η by cross-validating on a validation set. In our experiments, this was done on the POS tagging data and the (max-iteration, learning rate) with the best performance was chosen for both POS tagging and MT experiments.

Fig. 3 Illustration of the phrase pair (“ne suis pas”, “am not”) (the word alignments are in *black boxes*). The linguistic features are extracted from a window environment (*red shadow boxes*) around the source phrase

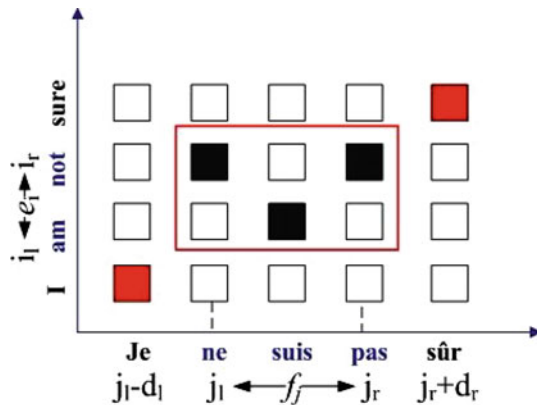


Table 2 Features extracted from the phrase environment

Types	Feature extraction
Context	Source word n-grams within a window (length d) around the phrase edge $[j_l]$ and $[j_r]$
Syntax	Source part of speech tag n-grams within a window (length d) around the phrase edge $[j_l]$ and $[j_r]$

the length of u . In this way, the phrase features are distinguished by both the content u and the start position p . For example, in Fig. 2 the word “not” in example 2 has the following context features $\{\delta(s_0^1, \text{“I”}), \delta(s_1^1, \text{“am”}), \delta(s_3^1, \text{“sure”}), \delta(s_0^2, \text{“I am”})\}$. As required by the PSL algorithm, we then *normalise* the feature vector $\bar{\phi}_t = \frac{\phi_t}{\|\phi\|}$.

3.2 Model training

To form the training sample pool, all consistent phrase pairs $\{(\bar{f}_n, \bar{e}_n)\}_{n=1}^{N_{\bar{f}}}$ with the corresponding features are derived from the training sentences using a phrase pair extraction procedure described in Koehn et al. (2005).⁴ Then the instances having the same source phrase \bar{f} are considered to be from the same cluster (see Fig. 2 for example) and a mapping operator $w_{\bar{f}}$ is tuned by the cluster samples only. When decoding, given a source phrase \bar{f}_j , we find the corresponding cluster model and predict the confidence-rated possibility for each candidate translation. For MT experiments, the confidence-rated values are then transformed to probabilities using the softmax function (Bishop 2006).

⁴ Since there is no word alignment problem in POS tagging, the word-to-tag samples with the linguistic features are derived directly from the training corpus.

Table 3 Data sizes of POS tagging experiments

	CoNLL2004 data set		CoNLL2009 data set	
	Tokens	Unknown tokens	Tokens	Unknown tokens
Training	211,727	0	976,567	0
Test	47,377	3,092 (6.5%)	14,964	261 (1.7%)

4 Experiments

4.1 Part-of-speech (POS) tagging

In this paper, we regard POS tagging as a special case of machine translation. The motivation of this experiment is to introduce the MMS model for capturing the relationships between the tags for ambiguous words, and we expect it to improve the performance of problematic cases described in Santorini (1990). In contrast to MT, the “distance” between POS tags for the same word are not clear and can not be measured by the Levenshtein distance. Hence, the distance matrix $\Delta(c, t_i)$ used is predefined heuristically, according to the problematic cases described in Santorini (1990). In general, the harder the problematic case is, the larger the distance will be.

The MMS model was trained and tested on two corpora: the CoNLL2004 and the CoNLL2009 data sets.⁵ The former is the POS tagged Wall Street Journal section of the Penn Treebank, where sections 15–18 are used for training and section 20 as a test set. The latter is a larger POS tagged set which matches sections 2–21 and 24 of the Penn Treebank, where 40,000 training sentences and 613 test sentences are sampled from the corpus and the experiments are repeated three times to access the variance. The sizes of both data sets are shown in Table 3.

To compare the performance, results derived from three other systems are also displayed. First is the Stanford POS Tagger (Toutanova et al. 2003) that utilises a maximum entropy model; next is a multi-class SVM model which is trained by SVM-Multiclass (Joachims 1999), the final is a stochastic gradient descent (SGD) classifier recently proposed in Shalev-Shwartz et al. (2007). The performance is measured by the word error rate (WER) as well as several class-specific F1 scores for the most problematic cases.

The feature set for the Stanford system is described in Toutanova et al. (2003) and a beam search decoder is applied to generate the predicted tag sequence. Alternatively, the other three models use the context features in Table 2. Observing that POS features might help the prediction, we also incorporate the syntactic features (see Table 2) by applying two-stage prediction. That is, first predicting the POS tags using the context features only, then predicting the POS tags again by incorporating the POS features predicted in the first stage. Since unknown words are unable to be assigned to certain word clusters, they are assigned to certain environment clusters instead. That is, for

⁵ Data supplied by Conference on Natural Language Learning (CoNLL) 2004 and CoNLL 2009 shared tasks.

Table 4 Test word error rate (WER) [%] for known words (K-WER.) and unknown words (UN-WER.) of the four models

Model	CoNLL2004 data set		CoNLL2009 data set		
	K-WER.	UN-WER.	K-WER.	UN-WER.	Runtime
Stanford	6.07	34.74	2.90 ± 0.15	31.6 ± 4.8	121.3
SVM	5.98	27.86	3.54 ± 0.15	23.8 ± 2.2	162.3
SVM + POS	5.73	29.06	2.80 ± 0.14	24.7 ± 1.5	167.5
MMS	5.92	27.65	3.20 ± 0.10	18.6 ± 1.4	37
MMS + POS	5.70	29.48	2.65 ± 0.10	23.0 ± 2.6	62.6
SGD	6.05	28.46	3.70 ± 0.26	22.2 ± 0.6	10.5
SGD + POS	5.83	29.27	3.00 ± 0.15	23.8 ± 0.8	15.3

In particular, the runtime (minutes) of each system is also presented for the CoNLL2009 data set. If not specified the models use the context features only; “POS” denotes using the predicted POS features as well. Bold numbers indicate the best results

Table 5 F1 scores for the most confusing POS classes, using “MMS + POS” (left) and “SVM + POS” (right)

Tag	F1 score	Tag	F1 score	Tag	F1 score
IN	98.2% / 98.1%	JJ	92.4% / 92.3%	VBD	80.6% / 80.5%
NN	93.7% / 93.7%	NNP	96.9% / 96.9%	VBN	69.9% / 69.2%
NNPS	52.4% / 51.2%	RB	90.9% / 90.9%	VBP	77.0% / 77.2%
RP	76.7% / 77.0%	VB	75.4% / 75.6%	VBZ	86.5% / 86.4%

Bold numbers indicate better results

a sample with an unknown word f_j , it is assigned to a cluster with samples having the same word environment $\{f_{j-1}, *, f_{j+1}\}$. For those unknown words which can't be assigned to any cluster, we simply regard them as unpredictable.

The results are shown in Table 4. The WER figure for the MMS model is the lowest. For the CoNLL2004 (small) data set, it achieves a relative improvement of 6.1% over the Stanford system on the known words and 20.4% on the unknown words. For the CoNLL2009 (large) data set, the relative improvements increase to 8.6% on the known words and a particular 41.4% on the unknown words. Similar improvements are observed over the multi-class SVM and the SGD classifier, showing the outstanding ability of MMS in exploiting linguistic features. Table 5 depicts the class-specific F1 scores for different POS tags that have the most confusing cases. In many cases, the MMS model performed better than the multi-class SVM.

Table 4 also displays the runtime of the four models on the CoNLL2009 data set.⁶ Compared with the SGD classifier that also has a linear computational complexity, the MMS model is slower due to the fixed learning rate used in the PSL algorithm. A faster PSL algorithm is achievable by incorporating the same updating strategy as used

⁶ The Stanford system is coded in Java, the MMS and the SGD models are coded in Python and SVM-multiclass is coded in C++.

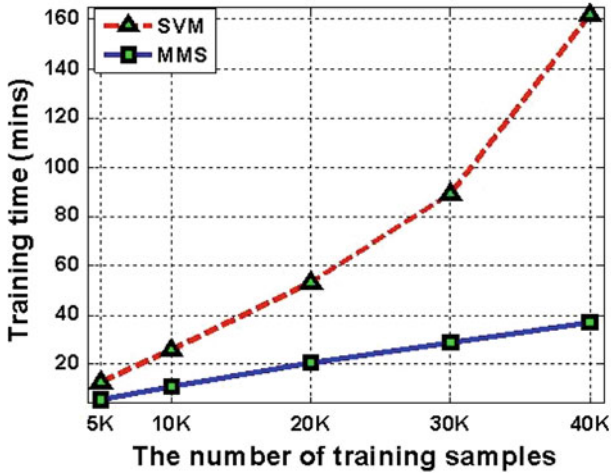


Fig. 4 The runtime for training examples from {5K, 10K, 20K, 30K, 40K} sentences using MMS (coded in Python) and SVM (coded in C++), respectively

in Shalev-Shwartz et al. (2007) and will be investigated in our future work. However, compared with the other two learning agents, PSL is substantially faster. Figure 4 further depicts the times for training examples from {5K, 10K, 20K, 30K, 40K} sentences using MMS and SVM respectively, demonstrating a linear time increase with MMS where in contrast a quadratic increase with SVM. This suggests that compared with SVM, MMS is more applicable to larger learning problems (e.g. MT problems).

Overall, most of our observations of MMS are desirable properties for the design of a PTP model, particularly the PSL algorithm provides a promising learning agent that has a good compromise between speed and performance.

4.2 Machine translation experiment

In this experiment, we use MMS for two complex MT tasks: French-to-English and English-to-French translation using the *EuroParl* corpus. Sentences of lengths between 1 and 100 words from the corpus were extracted where the ratio of source/target lengths was no more than 5:1. The training set is taken between {50K, 100K} sentences while the test set is fixed at 1K sentences. To compare the performance, two SMT systems—Pharaoh (Koehn 2004) and Moses (Koehn et al. 2005)—whose PTP models use maximum likelihood estimation (MLE), are taken as the baseline systems. To keep the comparison fair, our MT system just replaces their PTP models with our MMS prediction while sharing all other models (i.e. language model, phrase reordering model⁷ and beam search decoder).

For parameter tuning, minimum-error-rating training (Och 2003) is applied. Experiments are repeated three times to assess variance and the performance is evaluated by

⁷ For Moses, we only use the word distance-based reordering model to reduce the effect of the phrase reordering model.

Table 6 The classification precisions on the 50K-sentence tasks

Phrase classification	MLE	SVM	MMS
FR-to-EN	64.8% \pm 0.4%	65.1% \pm 0.4%	65.6% \pm 0.1%
EN-to-FR	52.8% \pm 0.6%	56.2% \pm 0.3%	56.8% \pm 0.3%

Bold numbers refer to the best results

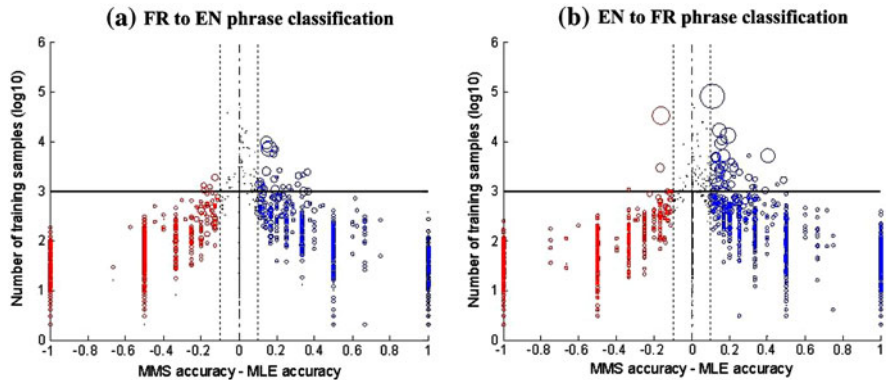


Fig. 5 Scatter-plots comparing the cluster accuracies of the MMS model with the MLE model on 50K-sentence French-to-English task (a) and English-to-French task (b). A cluster $\mathcal{S}_{\bar{f}}$ contains all phrase pairs with a unique source phrase \bar{f} . Those clusters for which the performance difference (x -axes) is greater than 0.1 are shown as *circles*, the areas of which are proportional to the number of target translations in them. The y -axes show the number of training samples (in log 10 form) for each cluster

four standard MT measurements, namely word error rate (WER), BLEU, NIST and METEOR (see Callison-Burch et al. 2007 for details).

We first demonstrate on Table 6 the phrase classification results on the 50K-sentence tasks, using MLE, SVM and MMS, respectively. In addition, Fig. 5 compares MMS with MLE on the basis of the overall accuracy of each cluster, suggesting when MMS works better and where it is better to apply:

- When given enough training samples (the black lines in Fig. 5), MMS is usually better than MLE. This verifies the advantage of the discriminative model and suggests using MMS when the training samples reach a reasonable size.
- The vocabulary in French is larger than that in English, which causes more polysemies when translating English into French (represented by the increasing numbers of large circles in Fig. 5b). The causes can be varied: the stylistic difference between English and French (e.g. English prefers the simple words from its Germanic wordstock where in contrast French uses learned words⁸); the different use of prepositions in French due to the grammatical gender (e.g. the English word

⁸ The “learned words” include a multitude of words which are comparatively seldom used in ordinary conversation. Their meanings are known to every educated person but there is little occasion to employ them at home or in the market-place (Greenough and Kittredge 1914). For example, the word “eye” is a popular word in the ordinary conversation, while “ocular” is a learned word. The readers are referred to Chapter III in Greenough and Kittredge (1914) for the concept of “learned words”; the stylistic difference

Table 7 Evaluations for MT experiments

Tasks	System	MT evaluations			
		BLEU [%]	WER [%]	NIST	METEOR [%]
FR-EN	MOSES	25.9 ± 0.2	39.1 ± 0.6	6.63 ± 0.07	48.6 ± 0.3
50K	MMS	27.1 ± 1.6	38.6 ± 0.7	6.74 ± 0.12	49.4 ± 0.7
EN-FR	MOSES	25.2 ± 0.3	43.2 ± 0.1	6.41 ± 0.03	47.7 ± 0.3
50K	MMS	27.1 ± 0.4	42.4 ± 0.3	6.58 ± 0.03	48.6 ± 0.2
FR-EN	Pharaoh	26.5 ± 0.4	39.0 ± 0.3	6.69 ± 0.04	50.8 ± 0.7
100K	MMS	27.1 ± 0.4	38.2 ± 0.3	6.80 ± 0.04	51.1 ± 0.7
EN-FR	Pharaoh	25.1 ± 0.4	42.6 ± 0.5	6.49 ± 0.06	47.9 ± 0.2
100K	MMS	26.0 ± 0.5	41.4 ± 0.5	6.65 ± 0.06	48.6 ± 0.3

Bold numbers refer to the best results

“the” can be translated as “la”, “le” and “les” in French, based on the noun it modifies); and the matter of French being more inflected than English. These situations make MMS a better choice, where the structured learning idea is beneficial. In contrast, when there is little ambiguity information in the target domain (i.e. less polysemies when translated French into English), MMS cannot benefit so much from the distance matrix and hence the positive effect of structured learning is not so high (represented by the smaller improvement on the French-to-English phrase classification task).

Table 7 depicts the translation results, where we observed consistent improvements in all evaluations. In addition, the improvements of MMS over the baseline on English-to-French MT tasks are usually better than those on French-to-English MT tasks, which is consistent with what has been observed on the phrase classification experiments. In particular, the WER and NIST scores concern more about the contents (rare n-grams), an improvement in both indicates that the MMS model is better in picking up correct words and phrases, which demonstrates the benefit of phrase disambiguation given by the MMS model.

5 Conclusion

In this paper, we applied a MMS model for two related NLP tasks: *POS tagging* and *phrase translation* in machine translation. We have shown that when using certain distance measures between output classes (e.g. tags or target translations), the MMS model showed improved performance for both tasks. Furthermore the PSL algorithm is faster than SVM without decreasing the performance in practice, making this model more applicable to the large scale learning problems (e.g. MT problems).

For future work, we will further develop our model for MT problems. We will refine the learning framework of MMS by carefully designing or automatically learn-

in using these words between English and French can be found in Sects. 2.1–2.2 in [Vinay and Darbelnet \(1995\)](#).

ing the distance matrix $\Delta(c, \bar{e}^n)$, aiming at capturing more complex structures in the target domain. We also intend to develop a further speed improvement in the PSL algorithm by incorporating a similar update strategy to that used in [Shalev-Shwartz et al. \(2007\)](#). Moreover, we will focus on the integration between the MMS model and other MT models (e.g. language model, phrase reordering model), as performance could be improved if the influence of these models are more effectively balanced in an end-to-end MT system. Finally, we will extend the MMS model to deal with larger corpora (e.g. the whole EuroParl corpus), with the purpose of verifying its ability in scaling up to large data collections.

Acknowledgment This work was supported by the European Commission under the IST Project SMART (FP6-033917).

References

- Bangalore S, Haffner P, Kanthak S (2007) Statistical machine translation through global lexical selection and sentence reconstruction. In: Proceedings of the 45th annual meeting of the association for computational linguistics (ACL 2007). Prague, Czech Republic
- Berger A, Pietra SD, Pietra VD (1996) A maximum entropy approach to natural language processing. *Comput Linguist* 22(1):39–72
- Bishop CM (2006) Pattern recognition and machine learning. Springer, New York
- Brants T (2000) TnT—a statistical part-of-speech tagger. In: Proceedings of the 6th applied natural language processing conference (ANLPS 2000). Seattle, WA, pp 224–231
- Cai L, Hofmann T (2004) Hierarchical document categorization with support vector machines. In: Proceedings of the ACM thirteenth conference on information and knowledge management (CIKM 2004). Hyatt Arlington Hotel, Washington, DC
- Callison-Burch C, Fordyce C, Koehn P, Monz C, Schroeder J (2007) (Meta-) evaluation of machine translation. In: Proceedings of the second workshop on statistical machine translation. Prague, Czech Republic, pp 136–158
- Carpuat M, Wu D (2007) Context-dependent phrasal translation lexicons for statistical machine translation. In: Proceedings of MT Summit XI. Copenhagen, Denmark
- Collins M (2002) Discriminative training methods for hidden markov models: theory and experiments with perceptron algorithms. In: Sammut C, Hoffmann AG (eds) Proceedings of the 19th international conference on machine learning (ICML 2002)
- Gentile C (2001) A new approximate maximal margin classification algorithm. *J Mach Learn Res* 2: 213–242
- Giménez J, Màrquez L (2007) Context-aware discriminative phrase selection for statistical machine translation. In: Proceedings of the second workshop on statistical machine translation. Prague, pp 159–166
- Greenough JB, Kittredge GL (1914) Words and their ways in English speech. The Macmillan Company, New York
- Gusfield D (1997) Algorithms on strings, trees, and sequences: computer science and computational biology. Cambridge University Press, New York
- Hofmann T, Cai L, Ciaramita M (2003) Learning with taxonomies: classifying documents and words. In: NIPS workshop on syntax, semantics, and statistics
- Joachims T (1999) Making large-scale SVM learning practical. In: Schölkopf B, Burges C, Smola A (eds) Advances in kernel methods—support vector learning. MIT Press, Cambridge
- Kivinen J, Smola AJ, Williamson RC (2004) Online learning with kernels. *IEEE Trans Signal Process* 52(8):2165–2176
- Koehn P (2004) Pharaoh: a beam search decoder for phrase-based statistical machine translation models. In: Proceedings of the 6th conference of the association for machine translation in the Americas (AMTA 2004), pp 115–124
- Koehn P, Axelrod A, Mayne AB, Callison-Burch C, Osborne M, Talbot D (2005) Edinburgh system description for the 2005 IWSLT speech translation evaluation. In: Proceedings of the international workshop on spoken language translation (IWSLT 2005). Pittsburgh, PA

- Lafferty JD, McCallum A, Pereira FCN (2001) Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proceedings of the eighteenth international conference on machine learning (ICML 2001). Morgan Kaufmann Publishers Inc, San Francisco
- Och FJ (2003) Minimum error rate training in statistical machine translation. In: Proceedings of the 41st annual meeting of the association for computational linguistics (ACL 2003). Japan
- Santorini B (1990) Part-of-speech tagging guidelines for the Penn Treebank project. In: Technical report MS-CIS-90-47, Department of Computer and Information Science, University of Pennsylvania
- Schölkopf B, Platt J, Shawe-Taylor J, Smola A, Williamson R (2001) Estimating the support of a high-dimensional distribution. *Neural Comput* 13(7):1443–1471
- Shalev-Shwartz S, Singer Y, Srebro N (2007) Pegasos: primal estimated sub-gradient solver for SVM. In: Proceedings of the twenty-fourth international conference (ICML 2007). Corvallis, OR
- Shawe-Taylor J, Cristianini N (2004) Kernel methods for pattern analysis. Cambridge University Press, Cambridge
- Taskar B, Guestrin C, Koller D (2003) Max-margin markov networks. In: Thrun S, Saul LK, Schölkopf B (eds) Proceedings of 7th annual conference on neural information processing systems (NIPS 2003). Vancouver, Canada
- Taskar B, Lacoste-Julien S, Jordan MI (2006) Structured prediction, dual extragradient and bregman projections. *J Mach Learn Res Spl Topic Mach Learn Optim*, pp 1627–1653
- Toutanova K, Klein D, Manning C, Singer Y (2003) Feature-rich part-of-speech tagging with a cyclic dependency network. In: Proceedings of the human language technology conference and meeting of the North American chapter of the association for computational linguistics (HLT-ACL 2003), pp 252–259
- Tsochantaridis I, Hofmann T, Joachims T, Altun Y (2004) Support vector machine learning for interdependent and structured output spaces. In: Greiner R, Schuurmans D (eds) Proceedings of the 21st international machine learning conference (ICML 2004). ACM Press
- Vickrey D, Biewald L, Teyssier M, Koller D (2005) Word-sense disambiguation for machine translation. In: Proceedings of the human language technology conference and conference on empirical methods in natural language processing (HLT-EMNLP 2005), pp 771–778
- Vinay JP, Darbelnet J (1995) Comparative stylistics of French and English: a methodology for translation. John Benjamins Publishing Company, Amsterdam