Rich Linguistic Features for Translation Memory-Inspired Consistent Translation

Yifan He Yanjun Ma[†] Andy Way^{‡*} Josef van Genabith

CNGL, School of Computing, Dublin City University, Dublin, Ireland

{yhe, josef}@computing.dcu.ie

[†] Baidu Inc., China

yma@baidu.com

[‡] Applied Language Solutions, Delph, UK

andy.way@appliedlanguage.com

Abstract

We improve translation memory (TM)inspired consistent phrase-based statistical machine translation (PB-SMT) using rich linguistic information including lexical, part-of-speech, dependency, and semantic role features to predict whether a TM-derived sub-segment should constrain PB-SMT translation. Besides better translation consistency, for English-to-Chinese Symantec TMs we report a 1.01 BLEU point improvement over a regular state-of-the-art PB-SMT system, and a 0.45 BLEU point improvement over a TM-constrained PB-SMT system without access to rich linguistic information, both statistically significant (p < 0.01). We analyze the system output and summarize the benefits of using linguistic annotations to characterise the nature of translation consistency.

1 Introduction

In the world of localization and professional translation, translation consistency is a much desired property. Given a particular domain, consistency in translation is characterized not only by correctness and fluency, but also by adhering to specific terminology translation, language patterns, and even error patterns (which are easier to identify in the post-editing stage). However, state-of-the-art statistical machine translation (SMT) systems do not explicitly model translation consistency, as the objective of these systems is to produce translations that maximize a weighted combination of translation model and language model scores (among others). Translation memories (TMs), widely used in the localization industry, assist translators by retrieving and displaying previously translated similar "example" segments (displayed as source-target pairs, called "fuzzy matches"). When presented with fuzzy matches, translators can avail of useful complete matching sub-segments in previous translations while composing the translation of a new segment. This improves the consistency of translation, as new translations produced by translators are based on the target side of the fuzzy match they have consulted, and translators will build their translations around terminologies already used in the TM.

It is, therefore, natural to resort to TMs for consistent translation, and to incorporate fully matching sub-segments from fuzzy match examples into the SMT pipeline (cf. (Koehn and Senellart, 2010), (Zhechev and van Genabith, 2010), and (Ma et al., 2011)). Although these methods have led to improved translations, they only use very simple features (such as a threshold on the fuzzy match score of the complete TM segment) to determine whether matching sub-segments from the fuzzy match are suitable for use in the SMT pipeline. Here, we propose a rich set of linguistic features to select TM fuzzy matches that contain useful sub-segments that improve translation consistency in an SMT pipeline.¹ We assume that many factors are rele-

^{*}Work done while at CNGL, School of Computing, DCU.

¹In Ma et al. (2011), we considered a richer set of features – including features from the translation model and source-side dependency relations – but a thorough exploration of features is not conducted, and linguistically-motivated features are limited

vant in deciding whether a full TM segment contains sub-segments that should be reused in translation: translation model, lexical, syntactic (dependency), and semantic features can all be helpful in predicting the consistency of translation, and improve translation quality. We explore a rich set of linguistic features in a consistency-oriented constrained translation task following the paradigm proposed by Ma et al. (2011). We show that our method leads to translations of better quality, reflected by a 1.01 BLEU point improvement over an out-of-the-box phrasebased SMT (PB-SMT) system, and a 0.45 BLEU point improvement over the system of Ma et al. (2011), all statistically significant (p < 0.01). Furthermore, our approach provides insight into the linguistic properties of consistent translation pairs.

The paper is organized as follows: we review related work in Section 2, and summarize the approach of Ma et al. (2011) in Section 3. We introduce and compare features induced from translation models and linguistically-oriented features for consistent translation in Section 4. We present experimental results and analyze linguistic properties of consistent translation in Section 5. In Section 6, we conclude and point out some possible avenues for future work.

2 Related Work

Our approach extends the line of research proposed by Ma et al. (2011), which improves the consistency of translations in PB-SMT systems by constraining the SMT system with consistent phrase pairs induced from TMs. Whether the consistent phrase pairs should be used is determined through discriminative learning. As the research in this paper builds on this previous work of ours, we review it in detail in Section 3. Prior to Ma et al. (2011), several proposals used translation information derived from TM fuzzy matches, such as (i) adding such translations into a phrase table as in Bicici and Dymetman $(2008)^2$ and Simard and Isabelle (2009), or (ii) marking up the input segment using the relevant sub-segment translations in the fuzzy match, and using an MT system to translate the parts that are not marked up, as in Smith and Clark (2009), Koehn

and Senellart (2010), and Zhechev and van Genabith (2010). However, these do not include a classification step that determines whether consistent phrase pairs should be used.

3 Constrained Translation via Markup Classification

Ma et al. (2011) tightly integrate TM with MT at the sub-segment level in the following way: given a segment e to translate, the most similar segment e' from the TM associated with the target translation f' is retrieved, and the m longest common subsequences ("phrases") $\bar{\mathbf{e}}_1^{\mathbf{m}}$ between \mathbf{e} and \mathbf{e}' are identified. Then a set of "consistent phrase pairs" $\{(\bar{e}_i, \bar{f}_i)\}_{i=1}^m$ is derived using the word alignment information between e' and f'. These "consistent phrase pairs" are used to mark up the matched sub-segments in the source segment with the predetermined translations, as described in Ma et al. (2011). If a classifier predicts that the markup will lead to improved translation quality and translation, the consistent phrase translation will be reused directly in the translation process. Below we explain how consistent phrase pairs are defined, and how markup classification is performed. Based on these, we discuss why linguistic features are essential for this task.

3.1 Consistent Phrase Pair Identification

We use the method of Ma et al. (2011) to extract consistent phrase pairs: extracted phrase pairs are the intersections of bidirectional GIZA++ posterior alignments (Och and Ney, 2003) between the source and the target side of the TM fuzzy match. We use the intersected word alignment to minimize the noise introduced by word alignment in one direction only, in order to ensure translation consistency.

3.2 Markup Classification

Following (Ma et al., 2011), we use Support Vector Machines (SVMs, (Cortes and Vapnik, 1995)) to determine whether constraining translation with our consistent phrase pairs can help improve translation quality. We treat constrained translation as a binary classification problem, and use the SVM classifier to decide whether we should mark up a segment or not. We label training data using the automatic TER score (Snover et al., 2006), as in (1).

to dependency labels.

²Note that discontinuous phrase pairs are used in Biçici and Dymetman (2008), whereas we use continuous phrase pairs here.

$$y = \begin{cases} +1 & \text{if TER(w. markup)} < \text{TER(w/o markup)} \\ -1 & \text{if TER(w/o markup)} \ge \text{TER(w. markup)} \end{cases}$$
(1)

Each data point is associated with a set of features which are discussed in more detail in Section 4.

We perform our experiments with the Radial Basis Function (RBF) kernel, and use Platt's method (Platt, 1999) (as improved by (Lin et al., 2007)) to fit the SVM output to a sigmoid function, to obtain probabilistic outputs from the SVM.

3.3 Rich Linguistic Features for Markup Classification

A close look at the markup classification procedure shows that the accuracy of classification (and ultimately, the quality and consistency of the output) is determined by how well we can capture the characteristics of a sub-segment that is a "consistent translation".

When integrating sub-segments from TMs into the SMT pipeline, previous work focused mainly on using information from the translation models of the TM or MT systems (cf. Section 2).³ However, linguistic annotations are potentially stronger indicators as to whether a fuzzy match sub-segment can constitute a consistent translation. For example, in industrial translation, brand and product names are often kept constant in the original form. A markup classifier which is only informed by translation model features may fail to capture this information, but a sequence of NN POS-tags would be a strong indicator.

Following this intuition, we explore a rich set of linguistic features including lexical information, part-of-speech, dependency, and semantic roles to improve translation consistency prediction.

4 Translation Features and Linguistic Features

4.1 Translation Features

We use both the TM fuzzy match score and features derived from the SMT model to predict the quality of consistent phrase pairs, following (Ma et al., 2011).

4.1.1 The TM Feature

The TM feature is the fuzzy match score, which indicates the overall similarity between the input segment and the source side of the TM output. We compute fuzzy match cost $h_{\rm fm}$ as the minimum edit distance (Levenshtein, 1966) between the source and TM entry, normalized by the length of the source, as in (2).

$$h_{\rm fm}(\mathbf{e}) = \min_{\mathbf{s}} \frac{\text{LevenshteinDistance}(\mathbf{e}, \mathbf{s})}{\text{Len}(\mathbf{e})}$$
 (2)

where e is the segment to translate, and s is the source side of an entry in the TM. For fuzzy match scores F, $h_{\rm fm}$ roughly corresponds to 1 - F.

4.1.2 Translation Features

We use the six features induced from the SMT translation model, following (Ma et al., 2011): the phrase translation and lexical probabilities for the consistent phrase pairs (cf. Section 3.1) in both directions derived using the method in Section 3, a count feature (i.e. the number of phrases used to mark up the input segment), and a binary feature (i.e. whether the phrase table contains at least one phrase pair $\langle \bar{e}_m, \bar{f'}_m \rangle$ that is used to mark up the input segment).

4.2 Linguistic Features

The linguistic features measure how well the marked-up portion covers the source segment. The assessments could be (but are not limited to) coverage measures, such as the percentage of content words that are marked up (lexical level) or the number of covered nouns (Part-of-speech (POS) level), as well as position-related properties, such as whether the marked-up sub-segment is at the beginning or the end of the segment.

Lexical Features Lexical features capture the surface-level properties of the marked-up translation. We use the following indicators given a segment and its markup: *Coverage* measures the percentage of words covered by the marked-up segment, which we calculate on both the source and the target side; *Alphabetical Words* measures the percentage of words that are alphabetical (i.e. not numbers and punctuation marks) in the source side of

³In Ma et. al (2011), we tentatively used information deduced from dependency relations, and reported that these features are more beneficial to markup classification than plain translation model features.

marked-up sub-segments; *Punctuation Marks* measures the percentage of words in the source side of marked-up sub-segments that are punctuation marks; *Content Words* calculates the percentage of content words in the source side of marked up sub-segments, for which we use the snowball stop words list⁴ to identify function words; and finally *Position* comprises two binary features which fire if marked-up sub-segments cover the head or the tail of the source segment.

POS Features For the POS features, we simply extend the calculation of lexical features to the POS level. The POS tags in our experiments are obtained using the Stanford Parser.⁵ For ease of discussion, we define $POS_i(\bar{e}_m)$ as the number of words in the input segment e that are marked up with translations from TM, and have the POS tag POS_i . We also define $\#POS_i(e)$ as the number of words in e that have the POS tag POS_i . We calculate the POS Coverage for each POS tag in the input segment, where $POS_Coverage_POS_i = \#POS_i(\bar{e}_m) / \#POS_i(e).$ We also use a binary feature POS Position to indicate whether the consistent phrase pair covers the head or the tail of a segment, where POS_Head_POS_i=1 iff the first word of the source segment is marked up and has the POS tag POS_i .

Dependency Features We use dependency relations (obtained using the Stanford parser) to establish the roles of matched parts in the input sentence in terms of syntactic dependencies. The dependency features include *DEP Coverage*, *DEP Position*, and *DEP Consistency*, all of which follow the definitions in Ma et al. (2011).

Semantic Role Features Our semantic role labels are obtained using the Suda SRL labeler,⁶ with constituent trees produced by the Stanford parser as input. The labels follow the PropBank (Palmer et al., 2005) annotation. Following POS features, for each predicate identified in a segment, we define $\text{SEM}_i(\bar{e}_m)$ as the number of words in the input segment e having the role SEM_i that are marked up

⁴http://snowball.tartarus.org/

⁵http://nlp.stanford.edu/software/ lex-parser.shtml with translations from TM, and define $\#SEM_i(e)$ as the number of words in *e* that have the SEM role SEM_i. The features include SEM Partial Coverage which calculates the marked-up percentage for each argument label,⁷ SEM Complete Coverage – a binary feature that fires if the phrase with an argument label is completely covered by the markup (i.e. SEM_COMPLETE_SEM_i=1 iff SEM_PARTIAL_SEM_i=1.0) – SEM Position which fires if an argument at the beginning or the end of the segment is covered by the markup, and SEM Predicate which fires only if the segment has no predicate.

5 Experiments

We use the same data set as in Ma et al. (2011), an English–Chinese TM with technical translation from Symantec, consisting of 87K segment pairs. The average segment length of the English training set is 13.3 words and the size of the training set is comparable to the larger TMs used in the industry.

We obtain training samples using the cross-fold translation technique in Ma et al. (2011), so the word aligner, the translation models, and the classifier are all trained on the same training corpus. We train the SVM classifier using the libSVM (Chang and Lin, 2001) toolkit. As for SVM parameters, we set c = 2.0 and $\gamma = 0.125$.

We conducted experiments using a standard log-linear PB-SMT (Och and Ney, 2004) system Moses,⁸ which is capable of handling user-specified translations for portions of the input during decoding. The maximum phrase length is set to 7.

5.1 Evaluation

The performance of the phrase-based SMT system is measured by BLEU score (Papineni et al., 2002) and TER (Snover et al., 2006). Significance testing is carried out using approximate randomization (Noreen, 1989).

We also measure the quality of the classification using precision and recall. Let A be the set of predicted markup input segments, and B be the set of input segments where the markup version has a lower TER score than the plain version. We stan-

algorithms/english/stop.txt

⁶http://nlp.suda.edu.cn/~jhli/

⁷If more than one predicate is identified, the value of the feature is averaged among argument labels for each predicate. ⁸http://www.statmt.org/moses/

dardly define precision P and recall R as in (3):

$$P = \frac{|A \cap B|}{|A|}, R = \frac{|A \cap B|}{|B|}$$
(3)

5.2 Experimental Results

5.2.1 Feature Validation

Table 1.	Contribution of Features	(%)
raule r.	contribution of i catales	(70)

	Ter	BLEU	Р	R
BASELINE	39.82	45.80	N/A	N/A
TRANS	39.80	45.84	66.67	1.02
LEX	39.65	46.20	71.43	10.20
Pos	39.30	46.71*	61.54	28.57
Dep	39.81	46.14	58.25	30.61
Sem	39.74	46.35	59.09	19.90
LPDS	39.32	46.81*	61.36	41.33

We first validate the contribution of the feature sets proposed. The classification and translation results using different features are reported in Table 1. BLEU scores marked with "*" are statistically significantly better (p < 0.01) than the BASELINE.

We observe that using translation model-derived features (TRANS) only brings about a trivial difference in translation quality. In fact, very low recall indicates that the SVM actually cannot obtain enough information from this feature set, and has to take advantage of the prior distribution of the samples (where we have more negative examples than positive ones) and reject almost every markup attempt to obtain the best accuracy. This shows that these features cannot capture the properties of the TM subsegments that help translation consistency.

By contrast, we observe that the linguistic features improve classification accuracy and translation quality. The improvement in BLEU scores range from 0.34 (DEP) to a statistically significant 0.91 (POS), which is the highest BLEU score obtained using a single set of features. However, we also observe that using POS features leads to a lower recall than using DEP. When we put the LEX, POS, DEP, and POS features together in the LPDS setting, we achieve the best BLEU score among all the settings, which is also significantly better than the baseline. The TER and precision numbers are marginally inferior to those obtained using the POS features alone. However, the much higher recall enables us to perform more confidence threshold-based tuning and achieve better results (cf. Section 5.2.3).

5.2.2 The Impact of Constrained Translation and Linguistic Features

We aim to obtain some insight on how much constrained translation can improve translation quality, and how much improvement is brought about by the linguistic features.

Table 2 contains the translation results⁹ of the SMT system when we use discriminative learning with LPDS to mark up the input segment (LPDS), which we compare to three baselines. The first baseline (BASELINE) is the result of translating plain test sets without any markup, and the second baseline is the result when all test segments are marked up. We also report results on a third baseline: TRANS+DEP, which corresponds to the best result reported in Ma et al. (2011). Besides these baselines, we also report the oracle scores, i.e. the upperbound of using our discriminative learning approach. As is reported

Table 2: Performance	of l	Discrin	ninative	Learning	(%)
----------------------	------	---------	----------	----------	-----

	TER	BLEU
BASELINE	39.82	45.80
MARKUP	41.62	44.41
TRANS	39.80	45.84
TRANS+DEP	39.63	46.36
LPDS	39.32	46.81
ORACLE	37.27	48.32

in Ma et al. (2011), if we categorically mark up all the input segments using phrase pairs derived from fuzzy matches, this leads to an absolute 1.4 point drop in BLEU score and a 1.8 point deterioration in TER. In contrast, both the ORACLE BLEU and TER scores represent as much as a 2.5 point improvement over the baseline.

Our discriminative learning method with a full linguistic feature set (LPDS) leads to an increase

⁹Note that two of the baseline scores we report are slightly different from those in Ma et al. (2011) due to small differences in data processing, with our previous TRANS score slightly lower (BLEU:45.51%) than in this paper, and TRANS+DEP slightly higher (BLEU:46.46%) than ours. Comparing the TRANS+DEP output in Ma et al. (2011) and running a significance test, our LPDS setting still significantly outperformed the TRANS+DEP setting in that paper with respect to BLEU at p < 0.05.

of 1.01 absolute BLEU points over the BASELINE, which is statistically significant with p < 0.01. We also observe a 0.5 point improvement in TER compared to the BASELINE, which shows that the proposed method can clearly outperform a vanilla PB-SMT pipeline.

To examine the role of linguistic features in this task, we also compare the LPDS setting to the TRANS setting – which does not use any linguistic information – and to the TRANS+DEP setting, which corresponds to the setting in Ma et al. (2011). The LPDS setting outperforms both TRANS (0.97 BLEU points improvement) and TRANS+DEP (0.45 BLEU points improvement) with statistical significance at p < 0.01, which reiterates the essential role of rich linguistic features in the markup classification procedure; as is shown in these experiments, the more complete the set of linguistic features used in this task, the better the observed performance.

5.2.3 Translation Results with Confidence Threshold

To further analyze our discriminative learning approach, we also investigate the use of classification confidence (cf. Section 3.2) as a threshold to boost classification precision.

As can be seen from Table 3, increasing the classification confidence up to 0.65 leads to a steady increase in classification precision with a corresponding sacrifice in recall. The fluctuation in classification performance has an impact on the translation results as measured by BLEU and TER. We can see that the best BLEU and TER scores are achieved when we set the classification confidence to 0.55, representing in further improvements of 0.19 points in BLEU score and 0.22 points in TER score, compared to the default threshold of 0.50.

Compared to the BASELINE, we obtain improvements of 1.20 in BLEU and 0.72 in TER (with lower TER score), all statistically significant (p < 0.01), when we set the confidence to 0.55. Despite the higher precision when the confidence is set above 0.60, the dramatic decrease in recall cannot be compensated for by the increase in precision.

We also compare the effect of applying confidence thresholds to all linguistically motivated feature sets we have proposed in Figure 3. Note that the LPDS features obtain the best BLEU scores in the [0.5, 0.65] range and obtain the highest BLEU score at the confidence level of 0.55, which confirms our approach of combining a variety of linguistic features for this task. In addition, we observe that although the BLEU score of POS features is also competitive at the confidence level of 0.5, the translation quality will not improve as we set a higher threshold, because its recall is already low initially.

5.3 Translation Examples and Characteristics of Consistent Translation

From the output of our system, we identify three directions where consistent phrase pairs can improve on baseline SMT outputs: *segment skeleton, coherent concept*, and *consistent terminology*. We also see that using linguistic features helps to better capture these scenarios, such as *coherent concept* in our example.

Segment skeletons are consistent phrase pairs that cover most of the source segment, leaving only very few words to change. As the majority of the segment is covered by the fuzzy match from the TM, it is better to use the translation skeleton directly, rather than to using MT to recombine sub-segments from different sources from scratch.

Using the segment in Figure 1 as an example, the consistent phrase pair already covers the second part of the segment. Without accessing this information, the MT output introduces several small inconsistencies and errors in the translation, which are avoided when we reuse the TM fuzzy match.

Coherent concepts are consistent phrase pairs which convey a single, self-contained concept. As such phrases are relatively independent from the other parts of a segment, directly reusing their translations from the TM in the MT pipeline is less risky. If the whole sub-segment functions as a single semantic role in the segment, it is a strong indicator that it constitutes a coherent concept.

This also serves as evidence for the necessity of using rich linguistic features in this task. In Figure 2, the classifier using the TRANS+DEP feature set cannot identify the consistent phrase pair which actually covers A0, AM-MOD, V, and A1 of the segment, and rejects the markup. In contrast, when using a full set of linguistic features, the LPDS classifier does make

Table 3: The impact of applying classification confidence threshold on the LPDS setting. Scores marked with "*" are significantly better (p < 0.01) than the BASELINE. The default classification confidence is 0.5.

		BASELINE	0.50	0.55	0.60	0.65	0.70	0.75	
	BLEU	45.80	46.81*	47.00*	46.79*	46.47	46.11	46.03	
	Ter	39.82	39.32	39.10 *	39.28	39.45	39.66	39.70	
	Р	N/A	61.36	67.96	71.01	75.00	70.97	71.43	
	R	N/A	41.33	35.71	25.00	18.37	11.22	7.65	
as long as au	to-protect is	enabled , a daily q	uick scan and	lasingle we	eekly schedul	led scan of a	all files pro	vides sufficie	nt protection
_	-			-			_		-
as long as au	to-protect is	enabled, a daily ad	ctive scan and	d a single, w	eekly schedu	led scan of	all files pro	vides sufficie	nt protection .
只要 启用 了	· 自动 防护	,每日快速扫	- 描和每周	一次针对	所有 文件 的	调度 扫描	就会提供	共 充分 防护	•

RFF 快速 扫描 和 每 周 一 次 针对 所有 文件 的 调度 扫描 就 会 提供 充分 防护。 只要 启用 自动 防护 E 舟

BASE 只要 启用 了 自动 一次,快速扫描及所有文件的调度扫描,每周提供足够的防护。 防护 舟 天 运行

Figure 1: Consistent Translation Examples: Segment Skeleton. SRC: source segment. TM: target side of TM fuzzy match. LPDS: markup classification using combined linguistic features. REF: reference translation. BASE: plain PB-SMT. Links between SRC and TM indicate consistent phrase pairs.

the correct prediction that leads to improvement in translation quality.

SRC

ΤM

LPDS



Figure 3: Confidence Threshold on Various Feature Sets

Consistent Terminology represents phrase pairs that are the translations of terminologies. In Figure 4, both LPDS and BASE translations are correct in meaning, but in the industrial environment, the LPDS translation is preferred, as it is using exactly the same Chinese expression when translating "any option" (underlined) from the previously translated segment found in the TM. This demonstrates that for enterprise translation, consistency is a dimension of translation quality that is independent from the more commonly used adequacy and fluency metrics.

SRC	you can set any of the following options	8:
TM	you can check any of the following options	5:
LPDS	您可以设置下列任何选项:	
REF	您可以设置下列任何选项:	
BASE	您可以设置下列任一选项:	



6 Conclusion

We investigated a technique that exploits a rich linguistically-motivated feature set to find reusable translation sub-segments from TM fuzzy matches in a bid to improve translation consistency, extending the paradigm proposed by Ma et al. (2011). We show that by using rich linguistic features, we can better predict the reusability of consistent translation pairs derived from the TM: our method outperforms a PB-SMT baseline by 1.01 BLEU points, and a consistent translation-aware system reported in Ma et al. (2011) by 0.45 BLEU points, both statistically significant (p < 0.01). We also investigate the properties of consistent translations, and note that the consistent phrase pairs combine the strengths of keeping the segment skeleton, reusing coherent concept, and adhering to consistent terminology.

SRC	[for any update], [you] [can] [select] [whether the update occurs within minutes of the scheduled time]. AM-DIS A0 AM-MOD V A1
TM	for daily, weekly, or monthly updates, you can select whether the update occurs within minutes of the scheduled time.
LPDS	对于任何更新,您可以选择是否要在调度时间后的几分钟内执行更新。
REF	对于任何更新, 您都可以选择更新是否在调度时间的分钟数内执行。
BASE/ Trans_Dep	为任何更新, 您可以选择是否后的几分钟内执行更新。在调度时间

Figure 2: Consistent Translation Example:Coherent Concept. Notations follow those of Figure 1. Semantic role labels on the source side are annotated. TRANS_DEP: markup classification using the feature set in Ma et al. (2011)

There are many possibilities to explore along this line of research, such as testing this method on languages other than English, labelling the training examples using other segment-level evaluation metrics such as Meteor (Banerjee and Lavie, 2005), and testing our method on a hierarchical system (Chiang, 2005) to facilitate direct comparison with Koehn and Senellart (2010).

Acknowledgments

This work is supported by Science Foundation Ireland (Grant No 07/CE/I1142) and part funded under FP7 of the EC within the EuroMatrix+ project (Grant No 231720). We thank Junhui Li for his help with the Semantic Role Labeler.

References

- Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization.
- Ergun Biçici and Marc Dymetman. 2008. Dynamic translation memory: Using statistical machine translation to improve translation memory. In *CICLing*.
- Chih-Chung Chang and Chih-Jen Lin, 2001. *LIB-SVM: a library for support vector machines*. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.
- David Chiang. 2005. A hierarchical Phrase-Based model for Statistical Machine Translation. In *ACL*.
- Corinna Cortes and Vladimir Vapnik. 1995. Supportvector networks. *Machine learning*, 20(3).
- Philipp Koehn and Jean Senellart. 2010. Convergence of translation memory and statistical machine translation. In AMTA Workshop on MT Research and the Translation Industry.

- Vladimir Iosifovich Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8).
- Hsuan-Tien Lin, Chih-Jen Lin, and Ruby C. Weng. 2007. A note on Platt's probabilistic outputs for support vector machines. *Machine Learning*, 68(3).
- Yanjun Ma, Yifan He, Andy Way, and Josef van Genabith. 2011. Consistent translation using discriminative learning: A translation memory-inspired approach. In ACL.
- Eric W. Noreen. 1989. Computer-Intensive Methods for Testing Hypotheses: An Introduction. Wiley-Interscience, New York, NY.
- Franz Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1).
- Franz Josef Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4).
- Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1).
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.
- John C. Platt. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Advances in Large Margin Classifiers.
- Michel Simard and Pierre Isabelle. 2009. Phrase-based machine translation in a computer-assisted translation environment. In *MT Summit XII*.
- James Smith and Stephen Clark. 2009. EBMT for SMT: A new EBMT-SMT hybrid. In *The 3rd International Workshop on Example-Based Machine Translation*.
- Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *AMTA*.
- Ventsislav Zhechev and Josef van Genabith. 2010. Seeding statistical machine translation with translation memory output through tree-based structural alignment. In *SSST*.