An Investigation on the Effectiveness of Features for Translation Quality Estimation

Kashif Shah, Trevor Cohn and **Lucia Specia**
Department of Computer Science
University of Sheffield
Sheffield, United Kingdom
{kashif.shah,t.cohn,l.specia}@sheffield.ac.uk

Abstract

We describe a systematic analysis on the effectiveness of features commonly exploited for the problem of predicting machine translation quality. Using a feature selection technique based on Gaussian Processes, we identify small subsets of features that perform well across many datasets for different language pairs, text domains, machine translation systems and quality labels. In addition, we show the potential of the reduced feature sets resulting from our feature selection technique to lead to significantly better performance in most datasets, as compared to the complete feature sets.

1 Introduction

As Machine Translation (MT) systems become widely adopted both for gisting purposes and to produce professional quality translations, automatic methods are needed for predicting the quality of translations. This is referred to as Quality Estimation (QE). Different from standard MT evaluation metrics, QE metrics do not have access to reference (human) translations; they are aimed at MT systems in use. Applications of QE include:

- Decide which segments need revision by a translator (quality assurance);
- Decide whether a reader gets a reliable gist of the text;
- Estimate how much effort it will be needed to post-edit a segment;
- Select among alternative translations produced by different MT systems.

Work in QE started with the goal of estimating automatic metrics such as BLEU (Papineni et al., 2002) and WER (Blatz et al., 2004). However, these metrics are difficult to interpret, particularly at the sentence-level, and results proved unsuccessful. A new surge of interest in the field started recently, motivated by the widespread use of MT systems in the translation industry, as a consequence of better translation quality, more user-friendly tools, and higher demand for translation. In order to make MT maximally useful in this scenario, a quantification of the quality of translated segments similar to "fuzzy match scores" from translation memory systems is needed. QE work addresses this problem by using more complex metrics that go beyond matching the source segment against previously translated data. QE can also be useful for end-users reading translations for gisting, particularly those who cannot read the source language. Recent work focuses on estimating more interpretable metrics, where "quality" is defined according to the task at hand: post-editing, gisting, etc. A number of positive results have been reported (Section 2).

QE is generally addressed as a supervised machine learning task using algorithms to induce models from examples of translations described through a number of **features** and annotated for quality. One of most challenging aspects of the task is the design of feature extractors to capture relevant aspects of quality.

A wide range of features from source and translation texts and external resources and tools have been used. These go from simple, language-independent features, to advanced, linguistically motivated features. They include features that rely

Sima'an, K., Forcada, M.L., Grasmick, D., Depraetere, H., Way, A. (eds.)
Proceedings of the XIV Machine Translation Summit (Nice, September 2–6, 2013), p. 167–174.

on information from the MT system that generated the translations, and features that are oblivious to the way translations were produced. This variety of features plays a key role in QE, but it also introduces a few challenges. Datasets for QE are usually small because of the cost of human annotation. Therefore, large feature sets bring sparsity issues. In addition, some of these features are more costly to extract as they depend on external resources or require time-consuming computations. Finally, it is generally believed that different datasets (i.e. language pair, MT system or specific quality annotation such as post-editing time vs translation adequacy) can benefit from different features.

Feature selection techniques can help not only select the best features for a given dataset, but also understand which features are in general effective. While recent work has exploited selection techniques to some extent, the focus has been on improving QE performance on individual datasets (Section 2). As a result, no general conclusions can be made about the effectiveness of features across language pairs, text domains, MT systems and quality labels.

In this paper we propose to use Gaussian Processes for feature selection, a technique that has proved effective in ranking features according to their discriminative power (Specia et al., 2013). We benchmark with this technique on two settings: (i) nine datasets for three language pairs, seven Statistical MT (SMT) systems and three types of quality scores with the same feature sets; (i) one dataset (same language pair and quality scores) with seven feature sets produced in a completely independent fashion (by participants in a shared task on the topic) (Section 3). The experiments showed the potential of feature selection to improve overall regression results, often outperforming published results on features that had been previously selected using other methods. They also allowed us to identify a small number of well-performing features across datasets (Section 4). We discuss the feasibility of extracting these features based on their dependence on external resources or specific languages.

2 Related work

Examples of successful cases of QE include improving post-editing efficiency by filtering out low quality segments which would require more effort or time to correct than translating from scratch (Specia et al., 2009; Specia, 2011), selecting high quality segments to be published as they are, without post-editing (Soricut and Echihabi, 2010), selecting a translation from either an MT system or a translation memory for post-editing (He et al., 2010), selecting the best translation from multiple MT systems (Specia et al., 2010), and highlighting sub-segments that need revision (Bach et al., 2011). For an overview of various algorithms and features we refer the reader to the WMT12 shared task on QE (Callison-Burch et al., 2012).

Most previous work on QE use machine learning algorithms such as SVMs, which are robust to redundant/noisy features to a certain extent. In what follows we summarise recent work using explicit feature selection methods in the WMT12 QE shared task.

González-Rubio et al. (2012) performed feature selection on a set of 475 sentence- and sub-sentence level features. Principal Component Analysis and a greedy selection algorithm to iteratively create subsets of increasing size with the best-scoring individual features were exploited. Both selection methods yielded better performance than all features, with greedy selection achieving the best MAE scores with 254 features.

Langlois et al. (2012) reported positive results with a greedy backward selection algorithm that removes 21 poor features from an initial set of 66 features based on error minimisation on a development set.

In an oracle-like experiment, Felice and Specia (2012) use a sequential forward selection method, which starts from an empty set and adds one feature at a time as long as it decreases the model's error, evaluating the performance of the feature subsets on the test set directly. 37 features out of 147 are selected, and these significantly improved the overall performance.

Avramidis (2012) tested a few feature selection methods using both greedy stepwise and best first search to select among their 266 features with 10-fold cross-validation on the training set. These resulted in sets of 30-80 features, all outperforming the complete feature set. Correlation-based selection with best first search strategy was reported to perform the best. Conversely, Moreau and Vogel (2012) reported no improvements in performance in experiments with several selection methods.

Finally, (Soricut et al., 2012), the winning system in the WMT12 QE shared task, used a computationally-intensive method on a development set. For each of the official evaluation metrics (e.g. MAE), from an initial set of 24 features, all 2^{24} possible combinations were tested, followed by an exhaustive search to find the best combinations. The 15 features belonging to most of the top combinations were selected. Other rounds were added to deal with POS features, but the final feature sets included 14-15 features depending on the evaluation metric. This technique outperformed the complete feature set by a very large margin.

3 Experimental settings

3.1 Datasets with common feature sets

All datasets used in our experiments are available for download.[1] The statistics of these datasets are shown in Table 1.

WMT12 English-Spanish news sentence translations produced by a phrase-based (PB) Moses "baseline" SMT system,[2] and judged for post-editing effort in 1-5 (highest-lowest), taking a weighted average of three annotators.

EAMT11 English-Spanish (EAMT11-en-es) and French-English (EAMT11-fr-en) news sentence translations produced by a PB-SMT Moses baseline system and judged for post-editing effort in 1-4 (highest-lowest).

EAMT09 English sentences from the European Parliament corpus translated by four SMT systems (two Moses-like PB-SMT systems and two fully discriminative training systems) into Spanish and scored for post-editing effort in 1-4 (highest-lowest). Systems are denoted by s_1-s_4.

GALE11 Arabic newswire sentences translated by two Moses-like PB-SMT systems into English and scored for adequacy in 1-4 (worst-best). Systems are denoted by s_1-s_2.

The features for these datasets are extracted using an open source toolkit QuEst.[3] We differentiate between *black-box* (BB) and *glass-box* (GB) features, as only BB are available for all datasets (we did not have access to the MT systems that produced the other datasets). For the WMT12 and GALE11 datasets, we experimented with both BB and GB features. The BB feature sets are the same for all datasets, except for one language pair (Arabic-English), where language-specific features supplement the initial 80 features.

Data	Training	Test
WMT12 (en-es)	1,832	422
EAMT11 (en-es)	900	64
EAMT11 (fr-en)	2,300	225
EAMT09-s_1-s_4 (en-es)	3,095	906
GALE11-s_1-s_2 (ar-en)	2,198	387

Table 1: Number of sentences in our datasets

We also distinguish one special feature: the *pseudo-reference* (PR), as this is not a standard feature in that it requires another MT system to be extracted. This feature consists in translating the source sentence using another MT system (in our case, Google Translate) to obtain a *pseudo-reference*. The geometric mean of (lambda-smoothed) 1-to-4-gram precision scores (i.e. a smoothed version of BLEU to avoid 0-counts without the brevity penalty) is then computed between the original MT and this pseudo-reference. We note that the better the external MT system, the closer the pseudo-reference translation is to a human translation, and therefore the more reliable this feature becomes.

For each dataset we built five systems:

- **BL**: 17 features that performed well across languages in previous work and were used as baseline in the WMT12 QE task.
- **AF**: All features available for the dataset, a superset of the above. For a comprehensive list, we refer the reader to QuEst website.[3]
- **BL+PR**: 17 baseline features along with a pseudo reference feature.
- **AF+PR**: All features along with a pseudo reference feature.
- **FS(GP)**: Feature selection for automatic ranking and selection of top features with Gaussian Process on set **AF+PR**.

3.2 WMT12 feature sets

These very diverse feature sets were provided by the participants in the WMT12 shared task on QE.[4]

[1] http://www.dcs.shef.ac.uk/~lucia/resources.html

[2] http://www.statmt.org/moses/?n=Moses.Baseline

[3] http://www.quest.dcs.shef.ac.uk

[4] These feature sets were made available by the task organisers at http://www.dcs.shef.ac.uk/~lucia/

We note that in a few cases these are a subset of the datasets used in the shared task, e.g. **UU**. This explains the difference between the official scores reported in (Callison-Burch et al., 2012) and our figures. This difference can also be explained by the learning algorithms: while we used GPs, participants have used SVRs, M5P and other algorithms. Some of these feature sets already result from feature selection techniques.

SDL (Soricut et al., 2012): 15 features selected after an exhaustive search algorithm based on all possible combinations of features. This is the optimal set used by the winning submission. It includes many of the baseline features, the pseudo-reference feature, phrase table probabilities, and a few part-of-speech tag alignment features.

UU (Hardmeier et al., 2012): 82 features, a subset of those used in the shared-task as the parse tree features (based on tree-kernels) were not provided by the participants. These are similar to the common BL and BB features presented above and include various source and target LM features, average number of translations per source word, number of tokens matching certain patterns (hyphens, ellipsis, etc.), percentage of n-grams seen in corpus, percentage of non-aligned words, etc.

UEdin (Buck, 2012): 56 black-box features including source translatability, named entities, LM back-off features, discriminative word-lexicon, edit distance between source sentence and the SMT source training corpus, and word-level features based on neural networks to select a subset of relevant words among all words in the corpus.

Loria (Langlois et al., 2012): 49 features including 1-5gram LM and back-off LM features, interlingual and cross-lingual mutual information features, IBM1 model average translation probability, punctuation checks, and out-of-vocabulary rate.

TCD (Moreau and Vogel, 2012): 43 features based on the similarity between the (source or target) sentence and a reference set (the SMT training corpus or Google N-grams) with n-grams of different lengths, including the TF-IDF metric.

WLV-SHEF (Felice and Specia, 2012): 147 features which are a superset of the common 80 BB features above. The additional features include a number of linguistically motivated features for source or target sentences (percentage) or their comparison (ratio), such as content words and function words, width and depth of constituency and dependency trees, nouns, verbs and pronouns.

UPC (Pighin et al., 2012): 56 features on top of the baseline features. Most of these features are based on different language models estimated on reference and automatic Spanish translations.

3.3 Gaussian Processes for feature selection and model learning

Gaussian Processes (GPs; Rasmussen and Williams (2006)) are an advanced machine learning framework incorporating Bayesian non-parametrics and kernels, and are widely regarded as state of the art for many regression tasks. Despite that, GPs have been under-exploited for language applications. Most of the previous work on QE uses kernel-based Support Vector Machines for regression (SVR), based on experimental findings that non-linear models significantly outperform linear models. Like SVR, GPs can describe non-linear functions using kernels such as radial basis function (RBF). However in contrast, inference in GP regression can be expressed analytically and the kernel hyper-parameters optimised directly using gradient descent. This avoids the need for costly grid search while also allowing the use of much richer kernel functions with many more parameters. Further differences between the two techniques are that GPs are probabilistic models, and take a fully Bayesian approach by integrating out the model parameters to represent the posterior distribution.

GPs allow for many different kernels. Here we consider the RBF with automatic relevance determination,

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}\sum_i^D \frac{x_i - x_i'}{l_i}\right) \quad (1)$$

where the $k(\mathbf{x}, \mathbf{x}')$ is the kernel function between two data points $\mathbf{x}$, $\mathbf{x}'$ and D is the number of features, and σ_f and $l_i \geq 0$ are the kernel hyper-parameters, which control the covariance magnitude and the *length scales* of variation in each dimension, respectively. This is closely related to the

resources.html

RBF kernel used with SVR, except that each feature is scaled independently from the others, i.e., $l_i = l$ for SVR, while GPs allow for a vector of independent values. Following standard practice we also include an additive white-noise term in the kernel with variance σ_s^2. The kernel hyper-parameters $(\sigma_f, \sigma_n, \mathbf{l})$ are learned from data using a maximum likelihood estimates.

The learned length scale hyper-parameters can be interpreted as the per-feature RBF widths which encode the importance of a feature: the narrower the RBF (the smaller is l_i) the more important a change in the feature value is to the model prediction. Therefore, a model trained using GPs can be viewed as a list of features ranked by relevance, and this information can be used for feature selection by discarding the lowest ranked (least useful) features. GPs on their own do not provide a cut-off point on this ranked list of features, instead this needs to be determined by evaluating loss on a separate set to determine the optimal number of features.

In our experiments, learning and feature ranking are performed with an open source implementation of GP[5] regression. Each feature is centred and scaled to have zero mean and unit standard deviation. For feature ranking, the models are trained on the full training sets. The RBF widths, scale and noise variance are initialised with an isotropic kernel (with a single length scale, $l_i = l$) which helps to avoid local minima. The hyper-parameters are learned using gradient descent with a maximum of 100 iterations and cross-validation on the training set. A forward selection approach is then used to select features ranked from top to worst and train models with increasing numbers of features. In an oracle-like experiment, we analyse the performance of models with different sizes of feature sets directly on the test set. The subset of top ranked features that minimises error in each test set is selected to report optimal results and therefore the potential of feature selection using GPs.

4 Results

We use Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) to evaluate the models.

[5]http://sheffieldml.github.io/GPy/

4.1 Results on the common feature sets

The error scores for all datasets with common BB features are reported in Table 2, while Table 3 shows the results with GB features for a subset of these datasets, and Table 4 the results with BB and GB features together for the same datasets. For each Table and dataset, bold-faced figures are significantly better than all others (paired t-test with $p \leq 0.05$).

It can be seen from the results that adding more BB features (systems **AF**) improves the results in most cases as compared to the baseline systems **BL**, however, in some cases the improvements are not significant. This behaviour is to be expected as adding more features may bring more relevant information, but at the same time it makes the representation more sparse and the learning prone to overfitting.

It is interesting to note that adding a single feature, the pseudo-reference (systems **BL+PR**) to our baseline improves results in all datasets, often by a large margin. Similar improvements are observed by adding this feature to the set with all available features (systems **AF+PR**) .

Our experiments with feature selection using GPs lead to significant further improvements in most cases. We note that the **FS(GP)** figures are produced from selecting the ideal number of top-ranked features based on the test set results, and therefore should be interpreted as *oracle-like* optimal results. These results show the potential of feature selection with GPs: **FS(GP)** outperforms other systems despite using considerably fewer features (10-20 in most cases, with up to 31 in the Arabic-English datasets). These are very promising results, as they show that it is possible to reduce the resources and overall computational complexity for training the models, while achieving similar or better performance. For a more realistic overview of the results of feature selection using GPs, we plot the learning curves for some of our datasets.

The learning curves for top-ranked features according to our forward selection method for two of our feature sets is given in Figures 1. The y axis shows the MAE scores, while the x axis shows the number of features selected. Generally, we observe a very fast error decrease in the beginning as features are added until approximately 20 features, where the minimum (optimal) error scores

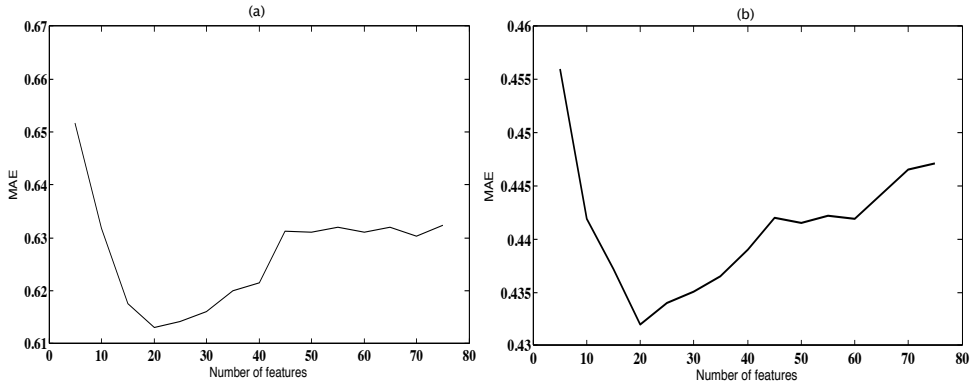

Figure 1: Error on (a) WMT12 and (b) EAMT11-en-es datasets with 80 BB features ranked by GPs

are found, and as more features are added, the error starts to quickly increase again, until a plateau is reached (approximately 45 features). This shows that while a very small number of features is naturally insufficient, adding features ranked lower by GPs degrades performance. Similar curves were observed for all datasets with slightly different ranges for optimal numbers of features and best score. It is interesting to note that the best performance gains on most datasets are observed within the 10-20 top-ranked features. Therefore, even though our optimal results rely on the test as oracle, this range of features could be used to find optimal results across datasets without an oracle.

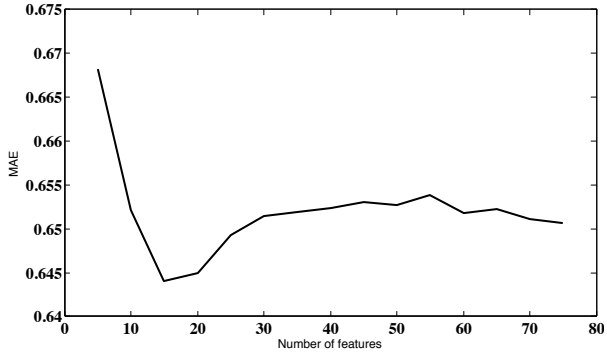

Figure 2: Error on 82 UU ranked features

GB features on their own perform worse than BB features, but in all three datasets, the combination of GB and BB followed by feature selection resulted in significantly lower errors than using only BB features with feature selection, showing that the two features sets are complementary.

4.2 Results on the WMT12 dataset

In order to investigate whether our feature selection results hold for other feature sets, we experimented with the feature sets provided by most teams participating in the WMT12 QE shared task. These feature sets are very diverse in terms of the types of features, resources used, and their sizes. As shown in Table 5, we observed similar results: feature selection with GPs has the potential to outperform models with all initial feature sets. Improvements were observed even on feature sets which had already been produced as a result of some other feature selection technique. Table 5 also shows the official results from the shared task (Callison-Burch et al., 2012), which are often different from the results obtained with GPs even before feature selection, simply because of differences in the learning algorithms used. In some cases results with GPs before feature selection are better, notably for WLV-SHEF, showing the potential of GPs as a learning algorithm for QE.

The learning curves with the performance on the test sets for different numbers of top-ranked features have a similar shape to those with the common feature sets. As an example, Figure 2 shows the Uppsala University feature set, with the lowest error score for the 15 top-ranked features.

4.3 Commonly selected features

Next we investigate whether it is possible to identify a common subset of features which are selected for the optimal feature sets in most datasets. In our experiments with the common feature sets, we found that the following features appear ranked at the top set that maximises the performance of the models in at least 75% of the times out of all datasets where they appear:

- LM perplexities and log probabilities for source and target sentences;
- size of source and target sentences;
- average number of possible translations of

Team	System	#feats.	Official WMT12 score		Score with GP	
			MAE	RMSE	MAE	RMSE
SDL	AF	15*	0.61	0.75	0.6030	0.7510
	FS(GP)	10	-	-	0.6015	0.7474
UU	AF	82	0.64	0.79	0.6507	0.8012
	FS(GP)	10			**0.6419**	**0.7931**
Loria	AF	49	0.68	0.82	0.6866	0.8340
	FS(GP)	10	-	-	**0.6824**	**0.8395**
UEdin	AF	56	0.68	0.82	0.6949	0.8540
	FS(GP)	20	-	-	**0.6795**	0 **8323**
TCD	AF	43	0.68	0.82	0.6906	0.8367
	FS(GP)	10	-	-	0.6904	0.8370
WLV-SHEF	AF	147	0.69	0.85	0.6665	0.8219
	FS(GP)	15	-	-	**0.6592**	**0.8088**
UPC	AF	57	0.84	1.01	0.8365	0.9601
	FS(GP)	15	-	-	**0.8092**	**0.9288**
DCU	AF	308	0.75	0.97	0.6782	0.8394
	FS(GP)	15	-	-	**0.6137**	**0.7602**
PRHLT	AF	497	0.70	0.85	0.6733	0.8297
	FS(GP)	30	-	-	**0.6647**	**0.8179**

Table 5: Results on WMT12 feature sets. * indicates initial feature sets resulting from feature selection

source words (IBM 1 with thresholds);
- ratio of target by source lengths in words;
- percentage of numbers in the target sentence;
- percentage of distinct unigrams seen in the MT source training corpus;
- pseudo-reference.

Interestingly, not all top ranked features are among the 17 reportedly good baseline features. All of these features are language-independent. Also, most of them are simple and straightforward to extract: they either do not rely on external resources, or use resources that are easily available, such as tools for LM (e.g., SRILM), or word-alignment (e.g., GIZA++).

The same analysis on the feature sets from the WMT12 shared task is not possible, given the very little overlap in features used by the different feature sets.

5 Conclusion

We have presented a number of experiments showing the potential of a promising feature ranking technique based on Gaussian Processes for translation quality estimation. Using an oracle to select the number of top-ranked features to train quality estimation models, this technique has been shown to outperform strong baseline systems with only a small fraction of their features. In addition, it allowed us to identify a common set of features which perform well across many datasets with different language pairs, machine translation systems, text domains and quality labels.

Acknowledgments

This work was supported by the QTLaunchPad project (EU FP7 CSA No. 296347).

References

Avramidis, Eleftherios. 2012. Quality estimation for machine translation output using linguistic analysis and decoding features. In *WMT12*, pages 84–90, Montréal, Canada.

Bach, Nguyen, Fei Huang, and Yaser Al-Onaizan. 2011. Goodness: a method for measuring machine translation confidence. In *ACL11*, pages 211–219, Portland, Oregon.

Blatz, John, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence Estimation for Machine Translation. In *Coling04*, pages 315–321, Geneva.

Buck, Christian. 2012. Black-box Features for the WMT 2012 Quality Estimation Shared Task. In *WMT12*, pages 91–95, Montréal, Canada.

Callison-Burch, Chris, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2012. Findings of the 2012 WMT. In *WMT12*, pages 10–51, Montréal, Canada.

Felice, Mariano and Lucia Specia. 2012. Linguistic features for quality estimation. In *WMT12*, pages 96–103, Montréal, Canada.

González-Rubio, Jesús, Alberto Sanchís, and Francisco Casacuberta. 2012. PRHLT Submission to the WMT12 Quality Estimation Task. In *WMT12*, pages 104–108, Montréal, Canada.

Dataset	System	#feats.	MAE	RMSE
WMT12	BL	17	0.6821	0.8117
	AF	80	0.6717	0.8103
	BL+PR	18	**0.6290**	**0.7729**
	AF+PR	81	0.6324	0.7735
	FS(GP)	19	**0.6131**	**0.7598**
EAMT11-en-es	BL	17	0.4857	0.6178
	AF	80	**0.4719**	**0.5418**
	BL+PR	18	**0.4490**	**0.5329**
	AF+PR	81	0.4471	0.5301
	FS(GP)	20	**0.4320**	**0.5260**
EAMT11-fr-en	BL	17	0.4401	0.6301
	AF	80	**0.4292**	**0.6192**
	BL+PR	18	**0.4183**	**0.6192**
	AF+PR	81	0.4169	0.6181
	FS(GP)	10	**0.4110**	**0.6099**
EAMT09-s_1	BL	17	0.5313	0.6655
	AF	80	0.5265	0.6538
	BL+PR	18	**0.5123**	0 **.6492**
	AF+PR	81	0.5109	0.6441
	FS(GP)	13	**0.5025**	**0.6391**
EAMT09-s_2	BL	17	0.4614	0.5816
	AF	80	0.4741	0.5953
	BL+PR	18	**0.4493**	**0.5692**
	AF+PR	81	0.4609	0.5821
	FS(GP)	12	**0.4410**	**0.5625**
EAMT09-s_3	BL	17	0.5339	0.6619
	AF	80	0.5437	0.6827
	BL+PR	18	**0.5113**	**0.6492**
	AF+PR	81	0.5309	0.6771
	FS(GP)	15	**0.5060**	**0.6410**
EAMT09-s_4	BL	17	0.3591	0.4942
	AF	80	0.3578	0.4960
	BL+PR	18	**0.3401**	**0.4811**
	AF+PR	81	0.3409	0.4816
	FS(GP)	19	0.3370	0.4799
GALE11-s_1	BL	17	0.5462	0.6885
	AF	123	**0.5399**	**0.6805**
	BL+PR	18	0.5301	0.6814
	AF+PR	81	**0.5249**	**0.6766**
	FS(GP)	27	0.5210	0.6701
GALE11-s_2	BL	17	0.5540	0.7117
	AF	123	**0.5401**	**0.6911**
	BL+PR	18	0.5401	0.7014
	AF+PR	81	**0.5249**	**0.6806**
	FS(GP)	31	**0.5194**	0.6779

Table 2: Results with common BB features

Dataset	System	#feats.	MAE	RMSE
WMT12	AF	47	0.7066	0.8445
	FS(GP)	21	**0.6755**	**0.8298**
GALE11-s_1	AF	39	0.5736	0.7402
	FS(GP)	19	0.5702	0.7361
GALE11-s_2	AF	48	0.5540	0.6979
	FS(GP)	13	0.5491	0.6974

Table 3: Results with common GB features

Dataset	System	#feats.	MAE	RMSE
WMT12	AF	128	0.7185	0.8451
	FS(GP)	29	**0.6101**	**0.7561**
GALE11-s_1	AF	163	0.5455	0.6722
	FS(GP)	30	**0.5150**	**0.6681**
GALE11-s_2	AF	172	0.5239	0.6529
	FS(GP)	17	**0.5109**	**0.6431**

Table 4: Results with common BB & GB features

Hardmeier, Christian, Joakim Nivre, and Jörg Tiedemann. 2012. Tree Kernels for Machine Translation Quality Estimation. In *WMT12*, pages 109–113, Montréal, Canada.

He, Yifan, Yanjun Ma, Josef van Genabith, and Andy Way. 2010. Bridging SMT and TM with Translation Recommendation. In *ACL2010*, pages 622–630, Uppsala, Sweden.

Langlois, David, Sylvain Raybaud, and Kamel Smaïli. 2012. LORIA System for the WMT12 Quality Estimation Shared Task. In *WMT12*, pages 114–119, Montréal, Canada.

Moreau, Erwan and Carl Vogel. 2012. Quality estimation: an experimental study using unsupervised similarity measures. In *WMT12*, pages 120–126, Montréal, Canada.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL02*, pages 311–318, Philadelphia, Pennsylvania.

Pighin, Daniele, Meritxell González, and Lluís Màrquez. 2012. The UPC Submission to the WMT 2012 Shared Task on Quality Estimation. In *WMT12*, pages 127–132, Montréal, Canada.

Rasmussen, Carl E. and Christopher K.I. Williams. 2006. *Gaussian processes for machine learning*, volume 1. MIT press Cambridge, MA.

Soricut, Radu and Abdessamad Echihabi. 2010. TrustRank: Inducing Trust in Automatic Translations via Ranking. In *ACL11*, pages 612–621, Uppsala, Sweden.

Soricut, Radu, Nguyen Bach, and Ziyuan Wang. 2012. The SDL Language Weaver Systems in the WMT12 Quality Estimation Shared Task. In *WMT12*, pages 145–151, Montréal, Canada.

Specia, Lucia, Marco Turchi, Nicola Cancedda, Marc Dymetman, and Nello Cristianini. 2009. Estimating the Sentence-Level Quality of Machine Translation Systems. In *EAMT09*, pages 28–37, Barcelona.

Specia, Lucia, Dhwaj Raj, and Marco Turchi. 2010. Machine translation evaluation versus quality estimation. *Machine Translation*, pages 39–50.

Specia, Lucia, Kashif Shah, José G. C. De Souza, and Trevor Cohn. 2013. QuEst - A translation quality estimation framework. In *Proceedings of ACL Demo Session (to appear)*.

Specia, Lucia. 2011. Exploiting objective annotations for measuring translation post-editing effort. In *EAMT11*, pages 73–80, Leuven.