Real-life Translation Quality Estimation for MT System Selection

Lluís Formiga, **Lluís Màrquez** and **Jaume Pujantell**
TALP Research Centre - Universitat Politècnica de Catalunya
lluis.formiga@upc.edu lluism@lsi.upc.edu jaumepujantell@gmail.com

Abstract

Research on translation quality annotation and estimation usually makes use of standard language, sometimes related to a specific language genre or domain. However, *real-life* machine translation (MT) performed, for instance, by on-line translation services, has to cope with extra difficulties related to the usage of open, non-standard and noisy language. In this paper we study the learning of quality estimation (QE) models able to rank translations from real-life input according to their goodness without the need of translation references. For that, we work with a corpus collected from the 24/7 Reverso.net MT service, translated by 5 different systems, and manually annotated with quality scores. We define several families of features and train QE predictors in the form of regressors or direct rankers. The predictors show a remarkable correlation with gold standard rankings and prove to be useful in a system combination scenario, obtaining better results than any individual translation system.

1 Introduction

Automatic evaluation of machine translation (MT) quality is a crucial task for system development, combination and tuning, which has received increasing attention from the MT community in the recent years. Translation quality estimation has classically been addressed as a scoring task (Specia et al., 2010), where some scoring function predicts the absolute quality of the automatic translation of a source text compared to human references (Papineni et al., 2002; NIST, 2002; Denkowski and Lavie, 2011) or without comparison (Specia et al., 2010). In this paper, we will use the term Quality Estimation (QE) to refer to the latter case, that is, predicting the quality of the translated text avoiding the need of human correct translations. QE has recently evolved towards two separate subtasks (Callison-Burch et al., 2012) consisting in scoring itself (Specia et al., 2010) and ranking, where different MT outputs for a given source sentence have to be ranked according to their comparative quality. Results obtained so far on QE have been more satisfactory for the ranking approach (Specia et al., 2010; Avramidis, 2012; Callison-Burch et al., 2012).

System ranking based on human quality annotations has been established as a common practice for MT evaluation in shared tasks (Callison-Burch et al., 2012). Therefore, training corpora are available for researchers to train ranking functions with supervised machine learning methods to perform automatic ranking mimicking human annotations. Learned models can be reusable, provided they are system independent and based on a generic analysis (i.e., no system dependent features can be used for training), and applicable to other sets containing any input and multiple outputs. The applications of *QE-for-ranking* are diverse: from hybrid MT system combination to their internal optimization and evaluation. The most popular practical scenario of QE models (both rankers and regressors) consists of ranking alternative MT systems' outputs to predict the best translation at segment level.

It is worth noting that the research conducted in QE for training ranking models from human annotations has always been done in *controlled environments*, consisting of well-formed text with little presence of noise (such as News or EU Parliament acts). However, MT in real life has to deal with a more complex scenario, including non-standard usage of text (e.g., social media, blogs, reviews, etc.), which is totally open domain and prone to contain ungrammaticalities and errors (misspellings, slang, abbreviations, etc.). In that case, human-trained QE models would be most

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. 69–76.

useful for the end user. An example of noisy environment is found in the publicly available FAUST English-to-Spanish corpus[1] (Pighin et al., 2012), collected from the 24/7 Reverso.net MT service. This corpus is composed of 1,882 weblog source sentences translated with 5 independent MT systems. The systems were ranked according to human assessments of adequacy by several users using a graph-based methodology, obtaining considerably high agreement and quality indicators (Pighin et al., 2012).

In this paper, we study the supervised training of QE prediction models from the aforementioned FAUST corpus to rank alternative system translations. Our study focuses on different aspects, such as: *i*) the typology of the problem (regression vs. binary classification), *ii*) suitability of the learner (SVM vs. M5P regression trees), and *iii*) best combination of features to learn. In order to analyze the results, we compute the correlation and decision accuracy between the rankings estimated by the predictors and the gold standard, but we evaluate also the results obtained by the *combined* MT system resulting from selecting, for every sentence, the individual translation predicted as best. Results prove that is possible to build reliable QE models from a noisy annotated corpus. Concretely, correlation results are comparable to those described in the literature for standard text. Furthermore, we also observed that comparative (ranked–based) QE models fit better to the system selection task (i.e. predict always the best translation) compared to absolute (regression–based) QE models.

The rest of the paper is structured as follows. Section 2 presents the learning framework and the features we propose to train our QE models. In Section 3 we detail the different experiments that are conducted to better assess the proper building of the QE models and evaluate their quality. The results are compared to the related work in Section 4. Finally, conclusions and future research lines are presented in Section 5.

2 Learning QE comparative models

As previously said, we want to study the appropriateness of different learning strategies towards obtaining the best translation ranking models according to absolute correlation performance and task-oriented under the scenario of system selection. We considered two different strategies to train the models: *i*) Learning to predict absolute quality scores by means of regression models and *ii*) Learning to predict pairwise quality ranking decisions by means of binary classification. Concerning the absolute regression models we compared two different learning algorithms: *i*) M5P from Weka (Quinlan, 1992) which combines a decision tree with linear equations on the leaves and *ii*) Support Vector Regression (Joachims, 1999), which is based on Support Vector Machines (SVM). Two different approaches can be followed to train regression models: *a*) *system dependent*, where independent regression models are built to predict the quality of each MT system, and *b*) *system independent*, where a single absolute regression model is built across all systems in order to distinguish good and bad translations. Note that the former approach requires the MT systems to be known beforehand and it is only applicable to that fixed set of systems. As we wanted to build open and general purpose QE models we focused our experiments on the latter approach. Finally, concerning the pairwise ranking decisions we only used SVM learning for preference ranking based on binary decisions.

For comparative purposes, we do not limit the learning experiments to the usual prediction of the human quality assessments as gold standard, but we also considered the problem of predicting the scores of automatic evaluation metrics (and the corresponding rankings) without having the real human reference. For that, we considered BLEU (Papineni et al., 2002), NIST (NIST, 2002) and METEOR (Denkowski and Lavie, 2011) scores computed at sentence level on the training set and learned from them as the gold standard. Concerning human rankings, we computed the average position of the QE predicted translations within the real test human rankings. More details are given in Section 3.

We used a large set of features to characterize examples and perform learning. They are grouped in three different sets and described in the following subsections. Some of them are inspired by well-established features from the literature (*baseline*, Section 2.1), others apply the *pseudo-reference* idea from Soricut and Echihabi (2010)

[1] http://www.faust-fp7.eu/faust/Main/DataReleases

to a larger set of MT evaluation measures (Section 2.2) and, finally, others are language model–based features developed for the particular corpora of application (Section 2.3).

2.1 Baseline Features

Specia et al. (2010) defined a broad set of features covering important aspects for QE learning. Later, Callison-Burch et al. (2012) selected a subset of 17 features for the WMT shared task on QE. Furthermore, other evaluation suites exist which define several QE basic metrics. An example of those is the ASIYA toolkit (Giménez and Màrquez, 2010). In our work, we will take the union of both previous feature sets to train our baseline system.

- ***Baseline*:**

  1. *Specia Baseline* (Callison-Burch et al., 2012): subset of 17 baseline features from Specia et al. (2010), containing token counts and their ratio, LM probabilities, n-grams filtered by quartiles, punctuation marks and fertility ratios.
  2. ASIYA *QE based features* (Giménez and Màrquez, 2010): 26 ASIYA QE features, comprising bilingual dictionary ambiguity and overlap; ratios concerning chunks, named-entities and PoS; source and candidate language model perplexities and inverse perplexities over lexical forms, chunks and PoS and out-of-vocabulary word indicators.

2.2 Pseudo-reference based features.

Albrecht and Hwa (2007) introduced the concept of Pseudo-Reference (PR) based features for translation regression estimation, later extended to ranking (Soricut and Echihabi, 2010). Their hypothesis was based on previous findings showing that, in the absence of human-produced references, automatically produced ones were still good in differentiating good and bad translations. These features require one or more secondary MT systems, used to generate translations starting from the same input. It is also crucial to have a good-quality MT system among the candidates, as the pseudo-reference becomes a more solid reference.

Pseudo-references intend to identify translation convergence using classical reference-based metrics as feature values. Their rationale is: "if system X produced a translation A and system Y produced a translation B starting from the same input, and A and B are similar, then A is probably a good translation". In contrast, Soricut and Echihabi (2010) highlight also that systems X and Y need to be as different as possible from each other. This property ensures that a convergence on similar translations is not just due to the learning approach (e.g., all Moses-like phrase-based), but a true indication that the translations are correct. The FAUST corpus contains translations from systems of different type (open-source phrase-based (Moses), general purpose phrase-based (*Google* and *LanguageWeaver*), rule-based (*Systran*) and hybrid (*Bing*)) making the corpus suitable for the system ranking task. The practical implementation that we took of that approach is to compute one or several reference-based metrics to each translation candidate as QE metrics using the alternative translations from the other systems as references.

Soricut and Echihabi (2010) only considered BLEU in order to generate the pseudo-reference based features. In our case, we expand the initial pseudo-BLEU feature towards two separate levels: *i*) Classical lexical oriented evaluation measures (BLEU, NIST and METEOR) and *ii*) More complex (linguistically–based) evaluation measures obtained with the help of the ASIYA toolkit (Giménez and Màrquez, 2010). These are described next:

- ***PR-based***

  3. *Classical lexical-based measures*: 5 PR-based features calculated over the following measures: BLEU (4-grams and smoothed (Papineni et al., 2002)), NIST (5-grams and smoothed (NIST, 2002)), METEOR-EX -PA -ST: Denkowski & Lavie (2011) with exact matching and with variants (plus stem matching stem matching).
  4. ASIYA *provided features*: 23 PR-features calculated over GTM; ROUGE; WER; PER; TER; and all Syntax-based evaluation measures provided by ASIYA for Spanish (Giménez and Màrquez, 2010).

2.3 Adapted Language Model based Features.

As a last group, we considered specific language model-based features to deal with the weblog data, which is comprised of different domains. To that

effect, we interpolated different language models comprising WMT12 Monolingual corpora (EPPS, News and UN) with Spanish source sentences gathered from the weblog of Reverso.net. The interpolation weights were computed as to minimize the perplexity according to the Spanish FAUST development set[2]. Hence, the last features are as follow:

- ***LM-based***

  5. 2 features (LM_{WEB}, LMPOS_{WEB}) computing log-probabilities of the translation candidate with respect to the above described interpolated language models over word forms and Part-of-Speech labels.

3 Experimental framework

3.1 Corpora and learners

In this paper we use the FAUST corpus (Pighin et al., 2012) to train our models. That corpus is composed of 1,882 weblog source sentences submitted to reverso.net online portal and translated with 5 different systems (Bing[3], Google[4], LanguageWeaver[5], Systran[6] and Moses[7]). The systems were ranked according to human assessments of adequacy. In addition, they also can be ranked according to different automatic metrics as we have the references. These rankings (human and automatic) are the main target for our models as we want to use them for a system-selection task (Specia et al., 2010). The ranking ties in the training data were treated with a *min* {1,2,2,4} heuristic so that the scale of the rankings become constant throughout all the sentences. The {1,2,2,4} heuristic is defined by setting the ties score to their lower value while maintaining the other scores beyond the tied position. Similar heuristics are *max* {1,3,3,4}, *avg* {1,2.5,2.5,4}, *random* {1,2,3,4} and *shortened-scale* {1,2,2,3}.

We analyzed different learners (rankers and regressors) using the following implementations: As for the regressors, we used the M5P algorithm from Weka (Quinlan, 1992) and SVM-Light (Joachims, 1999) for Support Vector Regression. These learning algorithms are referred as 'M5Preg' and 'SVMreg' throughout this paper. The SVR algorithm was run according to the following parameters: Expanding the working set by 9 variables at each iteration, for a maximum of 50,000 iterations and with a cache size of 100 for kernel evaluations. As for the ranker, we performed SVM ranking by means of pairwise comparison using the same SVMLight toolkit (Joachims, 1999) but with the "`-z p`" option, which can provide system rankings for all the members of different groups. This method is named 'SVMrank' in the paper. The learner parameter C was empirically set to 0.001.

3.2 Experiments

For the experiments we proceeded in two stages. First, we studied the more appropriate learning framework (M5Preg, SVMreg or SVMrank) and afterwards, when the best framework is set, we study the contribution of each group of features (i.e., Base, PR Classic (PR_C), PR Asiya (PR_A), and FAUST trained LM (WEB)) to help the prediction task.

Secondly, we evaluated the performance achieved by the QE models without being bounded to a specific task. We did that with two types of indicators: *a*) the *correlation* between the rankings (real and predicted) and also *b*) the *accuracy* in predicting pairwise ranking decisions. More concretely, we used: *i*) Spearman, *ii*) Kendall's τ rank correlation, *iii*) accuracy ratio within all pairwise decisions given the real and predicted rankings, and *v*) accuracy of the trained model when predicting the best system. These performance indicators were computed independently at segment (sentence) level. Afterwards, they were averaged obtaining a final value for the studied learner.

On the other hand, we analyzed the capability of the trained models to perform system selection and, therefore, meet or improve the scores (human or automatic) achieved by the individual MT systems. In order to compare, we define two oracle scores that know the real scores: *i*) $\text{Oracle}_{Dominant}$ (O_D) which represents the score obtained when selecting always the best overall system (the dominant one) across all the segments (lower-bound oracle) and *ii*) Oracle_{Best} (O_B) which represents the score obtained when selecting the best translation for each segment (upper-bound oracle). We also

[2] ftp://mi.eng.cam.ac.uk/data/faust/FAUST-1.0.tgz
[3] http://translate.bing.com
[4] http://translate.google.com
[5] http://www.sdl.com
[6] http://www.systransoft.com
[7] http://www.statmt.org/moses/

E	T	ML	ρ	τ	Acc	Acc'
HUM	rank	SVM	32.51	28.20	39.06	46.67
	reg	M5P	**33.86**	**31.69**	**44.67**	**51.11**
	reg	SVM	24.60	21.18	36.56	39.44
$BLEU$	rank	SVM	**35.64**	**29.67**	**56.17**	**43.89**
	reg	M5P	32.07	29.13	56.22	40.00
	reg	SVM	29.39	25.12	54.00	40.00
$NIST$	rank	SVM	**37.27**	**31.78**	**56.89**	**39.44**
	reg	M5P	29.97	26.88	46.44	**39.44**
	reg	SVM	32.95	27.84	54.83	36.67
$METR$	rank	SVM	**38.43**	**33.02**	**58.11**	**41.67**
	reg	M5P	30.08	26.80	52.72	37.22
	reg	SVM	33.67	28.93	56.11	39.44

Table 1: Predicted ranking statistics according to different learning approaches and metrics. E column stands for the metric evaluated, T and ML stand for the type and implementation of the learner used. "reg" stands for regression approach and rank stands for "ranking" approach. ρ stands for spearman correlation, τ stands for Kendall's correlation. Acc stands for the accuracy, in percentage, over all pairwise decisions (better, worse, equal) and Acc' stands for the accuracy at predicting the best translation for each system

E	T	ML	BL_R	P	O_D	O_B
HUM	rank	SVM	2.18	**1.69**	1.77	1.00
	reg	M5P	2.02	1.79	1.77	1.00
	reg	SVM	2.22	1.86	1.77	1.00
$BLEU$	rank	SVM	33.64	**38.28**	37.57	44.91
	reg	M5P	29.99	35.94	37.57	44.91
	reg	SVM	31.00	38.25	37.57	44.91
$NIST$	rank	SVM	6.38	**6.83**	6.66	7.46
	reg	M5P	6.06	6.72	6.66	7.46
	reg	SVM	6.28	6.79	6.66	7.46
$METR$	rank	SVM	51.93	**57.27**	56.69	62.36
	reg	M5P	50.76	55.27	56.69	62.36
	reg	SVM	50.15	56.76	56.69	62.36

Table 2: Best predicted metrics (trained with Human Rankings). E, T and ML follow the same notation as Table 1. BL_R stands for the document metric baseline picking the system sentence at random. P stands for the document metric predicting the system sentence according to the models. O_D (dominant) stands for the document metric oracle while predicting all the sentences from the best dominant system and O_B (best) is the document metric real oracle while predicting the best sentence according to the metric

considered a baseline measure, Baseline_{Random} (BL_R), which computes the score obtained by randomly selecting the best translation for each source sentence. Those metrics were computed at document level after selecting the best translation among the 5 candidates.

For our experiments, we randomly split the FAUST corpus in two sets: training (90% – 1,694 sentences) and test (10% – 188 sentences).

In order to determine the best learning strategy we used all the features available to perform the task. In Table 1 we depict the correlation indicators obtained across different learning methods and paradigms. We also trained the QE models with the ranks from human assessments. The system selection performance is presented in Table 2. In that case, we try to predict the best system according to the human assessments and evaluate them with both, the human assessments and the automatic metrics.

From the metrics perspective (Table 2), it is clear that SVM rank is the best methodology across all the configurations. Therefore, we selected it as the appropriate learner to perform the study of features. The impact of feature types is presented in Table 3. Additionally, in Figure 1, we present relative bar charts showing the contribution to the final performance of each of the feature sets defined in Section 2. Discussion on all these aspects is detailed in the following Section 3.3.

3.3 Discussion

Method Analysis We found a different behavior depending whether we focus on the correlations or the task (Tables 1 and 2). While the automatic metrics (BLEU, NIST and METEOR) achieve the best correlations (Table 1) by means of SVMRank strategy, M5P regression is better suited for the task of predicting human rankings. However, when the models were applied to system selection task (Table 2) we observed that SVMRank also provided the best results overcoming the regression results. If we compare the regressors with themselves for this task, the SVM regression performed better compared to M5P regression.

The QE had the same performance if we trained them to learn automatic metrics. Confirming SVMrank as the best learner to perform system ranking. Concretely, when training SVMRank to learn the automatic metrics, we obtained: 38.73 for BLEU, 6.87 for NIST and 57.26 for METEOR. Therefore, no significant difference was found compared to Table 2. We want to clarify that this step involved three independently trained QE models. One for BLEU, one for NIST and one for METEOR.

The results lead to the following finding: not necessarily the correlation and accuracy indicators yield to predict the best systems for the task of system-selection. It is important to highlight the difference between selecting always the best

F	LM	BL_R	P	O_D	O_B	BL_R	P	O_D	O_B
		HUMAN				**BLEU**			
∅	WEB	2.23	2.02	1.77	1.00	31.92	34.31	37.57	44.91
+Base	–	2.15	2.07	1.77	1.00	29.75	31.29	37.57	44.91
	WEB	2.16	1.93	1.77	1.00	32.83	32.93	37.57	44.91
+PR_C	–	2.19	1.89	1.77	1.00	30.12	35.17	37.57	44.91
	WEB	2.28	1.75	1.77	1.00	31.33	37.64	37.57	44.91
+PR_A	–	2.14	1.82	1.77	1.00	30.87	37.55	37.57	44.91
	WEB	2.18	**1.69**	1.77	1.00	31.76	**38.73**	37.57	44.91
		NIST				**METEOR**			
∅	WEB	6.15	6.54	6.66	7.46	50.55	53.03	56.69	62.36
+Base	–	6.12	6.67	6.66	7.46	50.91	51.11	56.69	62.36
	WEB	6.44	6.80	6.66	7.46	50.07	53.22	56.69	62.36
+PR_C	–	6.31	6.61	6.66	7.46	52.55	54.31	56.69	62.36
	WEB	6.14	6.82	6.66	7.46	50.76	56.58	56.69	62.36
+PR_A	–	6.30	6.77	6.66	7.46	51.51	56.27	56.69	62.36
	WEB	6.09	**6.87**	6.66	7.46	51.67	**57.26**	56.69	62.36

Table 3: Predicted ranking metrics according to different feature subsets. The E, BL_R, P, O_D, O_B columns follow the same notation as Tables 1 and 2. F stands for the feature set used for training the model and LM stands for the use of additional language model perplexities as features. ∅ represents an empty set of features. '+X' represents a cumulative addition of the feature set X over the previous row setting

translation (fine-grained) and providing accurate system rankings comparable to human ranking (coarse-grained). In theory, the accuracyBest ($\%'$) indicator would be the most appropriate indicator for system selection. However, it does not consider the actual distance among n-best translations of the ranking. Depending on the task, the QE model indicators get worse results for accuracy ($\%$) compared to accuracyBest and vice versa. For example, in Table 1, predid, conversely, the assessments inverse the relation between indicators.

Therefore, it is clear that the best QE Models are the ones obtained under a comparative strategy with a pairwise ranking approach without the consideration a global score of quality.

Impact of Human Rankings In contrast to the automatic metrics, the concept of predicted "assessment" is confusing and must be explained properly before being discussed from the tables. We assume the hypothesis that a perfect QE model would choose a sentence ranked first by the humans. In that sense the Oracle_{Best} would be 1. But the QE model might choose translations other than the best. Hence, we take the average position within the rank throughout all the source sentences. That is, a predicted assessment of 1.69 indicates the average position of the QE predicted translations within the human rankings. An $\text{Oracle}_{Dominant}$ of 1.77 means that cted BLEU, NIST and METEOR achieve better accuracy scores compared to accuracyBest anthe translations from the best overall system were ranked in 1.77 position by humans as average.

We analyzed the impact of human rankings for training the QE Models (Table 2). We evaluated their suitability either for the system-selection task according to automatic metrics and also, as it has been described, obtain system-selection translation candidates that would perform better into a real human ranking. In that sense we found that QE Models trained with human rankings obtained better system-selected (predicted) scores than the best overall system alone ($\text{Oracle}_{Dominant}$) for all 4 prediction tasks (BLEU, NIST, METEOR and assessments).

After analyzing the results, they suggest that human assessments do help to obtain better QE models for system selection for either mimic the behavior of automatic metrics or learn the human behavior when ranking different translation candidates.

Feature Analysis The last thing we wanted to analyze was the contribution of each set of features (Base, PR_C PR_A and $LM_{WEB)}$ to performance of the QE models. In Table 3 we observed that additional PR and LM features boost considerably the correlation (Spearman and Kendall) and accuracyBest results, improving in the latter case, an accuracy from $\approx 30\%$ to a $\approx 40\%$. In that case, with a considerable improvement of bestAccuracy, it seems clear that bestAccuracy indicator is the

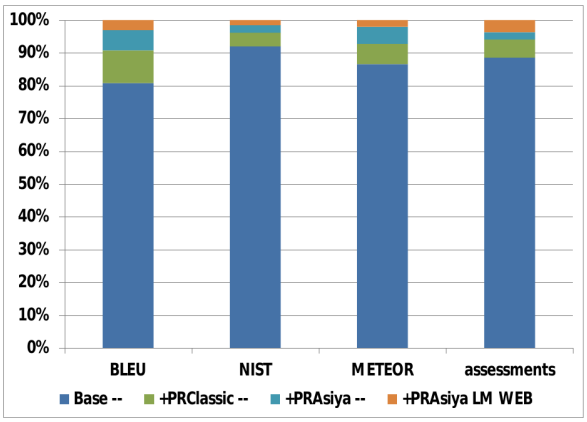

Figure 1: Plots of predicted ranking metrics according to the relative improvement of each feature set.

one that gives a better clue to determine the best QE model for the system selection task. In Figure 1 we analyzed the contribution (in percentage over total score) of each set of features to predict the final metric. We observed that, depending on the metric to predict, the baseline features contribute around 80-90% of the final score whereas the additional features can boost performance up to 20% in a cumulative way. Among the additional features, PR_C contributes to the most part of the improvement while LM_{WEB} contributes the least. PR_A also has a significant contribution over the improvement in BLEU and METEOR metrics. Finally, it is interesting to see that, under BLEU analysis, the performance of the WEB-based language model alone is better than the QE baseline features, highlighting the bias of BLEU to prioritize good n-gram matching without concerning their fluency or different length ratio, among others.

4 Related Work

The study presented in this paper roughly follows the approaches of Specia et al. (2010) and Avramidis (2012). However, the absolute results are not directly comparable as our working corpus is of a very different nature compared to theirs.

For the sake of comparison, we also trained our models with the publicly available corpus of Specia et al. (2010). In that case, we obtained accuracy levels of 75% at predicting automatic metrics and 61% when predicting the human assessments. In all the cases, accuracy values were higher than those obtained on the FAUST corpora, evincing the difficulty of the FAUST data and the effect of noise in both the source sentences and the MT output. In addition, for the system combination task over the Specia et al. (2010) corpus, we were able to obtain translation results comparable to the best individual MT system (O_D), which is unknown for the learned ranker. Note that in the corpus by Specia et al. (2010) there is a strong dominant system that makes O_D more difficult to beat. In this scenario, our QE models are able to properly identify the dominant system without having the references, making them useful for overall system comparison.

On the other hand, comparing our results to Avramidis' (2012), we obtained significantly better Kendall's Tau with respect to human assessments. We move in the range $29.67 < \tau < 33.02$, which we consider acceptable for the difficulty of the FAUST corpus. However, these results are neither directly comparable as his works covers a different language pair. We only give comparative comparisons.

Finally, it is noticeable that PR-based and ad-hoc LM features give a significant boost of performance to the QE models. To the best of our knowledge, the generalized application of pseudo-reference based features is a novel contribution of this work. Also, this is the first time that these QE models are trained with noisy data. We have demonstrated that despite this constraints we were able to perform system combination that outperforms the standalone best system, even with noisy and ambiguous data.

5 Conclusions

In this paper we have studied the problem of learning system independent quality estimation models to predict the quality of automatic translations from an online MT service. Working with real-life input text and translations implies facing some serious difficulties, derived from the usage of open-domain non-standard text, where errors, OOV words and ungrammatical sentences abound. Our study focuses on several aspects, such as the typology of the learning problem, the suitability of the learning algorithm and the best set of features to learn from. We have conducted experiments with a corpus collected from the 24/7 Reverso.net MT service, translated by 5 different MT systems, and manually annotated with adequacy

ranks. Apart from studying the correlation and accuracy of the resulting translation rankings, we have also evaluated the QE predictors in the application scenario of system combination by system selection (predict always the best translation).

Our study shows that it is possible to build reliable system-independent QE models from the FAUST real-life translation annotated corpus. The predicted rankings correlate well with the gold standard. When evaluated on the system combination task, we obtain significantly better results, across a set of evaluation measures, than random system selection and slightly better than a system-informed oracle consisting in selecting always the translation of the best overall MT system. These results and conclusions are in accordance with the state-of-the-art on standard text. Nonetheless, there is still a large room for improvement, according to the performance upper bound of the task.

We also concluded that the pairwise ranking strategy yields better QE models than an absolute quality estimation approach (i.e., regression) for the task of system selection. Moreover, human assessments help obtaining better QE models for system selection for both *i*) mimicking the behavior of automatic metrics and *ii*) learning the human behavior when ranking different translation candidates. Finally, taking a deeper look into the features defined beyond the baseline, we found that all defined add-ons (PR Classic, PR Asiya and LM WEB) were useful to boost the quality of the QE models and, therefore, improve the system-selection task.

Acknowledgements

This work has been partially funded by the Spanish *Ministerio de Economía y Competitividad*, under contracts TEC2012-38939-C03-02 and TIN2009-14675-C03, as well as from the European Regional Development Fund (ERDF/FEDER) and the European Community's FP7 (2007-2013) program under the following grant: 247762 (FAUST).

References

Albrecht, Joshua and Rebecca Hwa. 2007. Regression for sentence-level mt evaluation with pseudo references. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 296–303, Prague, Czech Republic, June. Association for Computational Linguistics.

Avramidis, Eleftherios. 2012. Comparative quality estimation: Automatic sentence-level ranking of multiple machine translation outputs. In *Proceedings of 24th International Conference on Computational Linguistics. International Conference on Computational Linguistics (COLING-12), December 8-15, Mumbai, India*, pages 115–132. The COLING 2012 Organizing Committee, 12.

Callison-Burch, Chris, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2012. Findings of the 2012 Workshop on Statistical Machine Translation. In *Proceedings of the 7th Workshop on Statistical Machine Translation*, pages 10–51, Montréal, Canada.

Denkowski, Michael and Alon Lavie. 2011. Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems. In *Proceedings of the 6th Workshop on Statistical Machine Translation*, pages 85–91, Edinburgh, Scotland.

Giménez, Jesús and Lluís Màrquez. 2010. Asiya: An Open Toolkit for Automatic Machine Translation (Meta-)Evaluation. *The Prague Bulletin of Mathematical Linguistics*, (94):77–86.

Joachims, Thorsten, 1999. *Advances in Kernel Methods – Support Vector Learning*, chapter Making large-Scale SVM Learning Practical. MIT Press.

NIST. 2002. Automatic Evaluation of Machine Translation Quality Using N-gram Co-Occurrence Statistics. Technical report, National Institute of Standards and Technology.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Pighin, Daniele, Lluís Formiga, and Lluís Màrquez. 2012. A graph-based strategy to streamline translation quality assessments. In *Proceedings of the Tenth Conference of the Association for Machine Translation in the Americas (AMTA'2012)*, San Diego, USA, October. AMTA.

Quinlan, John R. 1992. Learning with continuous classes. In *Proceedings of the 5th Australian joint Conference on Artificial Intelligence*, pages 343–348. Singapore.

Soricut, Radu and Abdessamad Echihabi. 2010. Trustrank: Inducing trust in automatic translations via ranking. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 612–621, Uppsala, Sweden, July. Association for Computational Linguistics.

Specia, Lucia, Dhwaj Raj, and Marco Turchi. 2010. Machine Translation Evaluation Versus Quality Estimation. *Machine Translation*, 24:39–50, March.