UGENT-LT3 SCATE System for Machine Translation Quality Estimation

Arda Tezcan **Veronique Hoste** **Bart Desmet** **Lieve Macken**
Department of Translation, Interpreting and Communication
Ghent University
Groot-Brittanniëlaan 45, 9000 Ghent, Belgium
{arda.tezcan, veronique.hoste, bart.desmet,
lieve.macken}@ugent.be

Abstract

This paper describes the submission of the UGENT-LT3 SCATE system to the WMT15 Shared Task on Quality Estimation (QE), viz. English-Spanish word and sentence-level QE. We conceived QE as a supervised Machine Learning (ML) problem and designed additional features and combined these with the baseline feature set to estimate quality. The sentence-level QE system re-uses the word level predictions of the word-level QE system. We experimented with different learning methods and observe improvements over the baseline system for word-level QE with the use of the new features and by combining learning methods into ensembles. For sentence-level QE we show that using a single feature based on word-level predictions can perform better than the baseline system and using this in combination with additional features led to further improvements in performance.

1 Introduction

Machine Translation (MT) Quality Estimation (QE) is the task of providing a quality indicator for unseen automatically translated sentences without relying on reference translations (Gandrabur & Foster, 2003; Blatz et al., 2004). Predicting the quality of MT output has many applications in computer-aided translation workflows that utilize MT, including error analysis (Ueffing and Ney 2007), filtering translations for human post-editing (Specia et al., 2009) and comparing the quality of different MT systems (Rosti et al. 2007).

The most common approach is to treat the QE problem as a supervised Machine Learning (ML) task, using standard regression or classification algorithms. A considerable amount of related work on both word and sentence-level QE is described in the WMT shared tasks of previous years (Bojar et al., 2014; Bojar et al., 2013).

The WMT 2015 QE shared task proposes three evaluation tasks: (1) scoring and ranking sentences according to predicted post-editing effort given a source sentence and its translation; (2) predicting the individual words that require post-editing; and (3) predicting the quality at document level. In this paper, we describe the UGENT-LT3 SCATE submissions to task 1 (sentence-level QE) and task 2 (word-level QE).

Sentence-level and word-level QE are related tasks. Sentence-level QE assigns a global score to an automatically translated sentence whereas word-level QE is more fine-grained and tries to detect the problematic word sequences. Therefore we first developed a word-level QE system and incorporate the word-level predictions as additional features in the sentence-level QE system. The usefulness of including word-level predictions in sentence-level QE has already been demonstrated by de Souza et al. (2014)

For both tasks, we extracted additional features and combine these with the baseline feature set to estimate quality. The new features try to capture either *accuracy* or *fluency* errors, where *accuracy* is concerned with how much of the meaning expressed in the source is also expressed in the target text, and *fluency* is concerned with to what extent the translation is well-formed, regardless of sentence meaning. This distinction is well known in quality assessment schemes for MT (White, 1995; Secară, 2005; Lommel et al., 2014). Some of the additional features are based on ideas that were explored in previous work on QE, such as; context features for the target word and of POS tags, (Xiong et al., 2010), alignment context features (Bach et al., 2011) and adequacy and fluency indicators (Specia et al., 2013).

Proceedings of the Tenth Workshop on Statistical Machine Translation, pages 353–360,
Lisboa, Portugal, 17-18 September 2015.

The rest of this paper is organized as follows. Section 2 and Section 3 give an overview of the shared task on word-level QE and sentence-level QE respectively and describe also the features we extracted, the learning methods and the additional language resources we used and the experiments we conducted. Finally, in Section 4, we discuss the results we obtained and the observations we made.

2 Word-level Quality Estimation

The word-level QE task is conceived as a binary classification task. The goal is to label translation errors at word level by marking words either as "GOOD" or "BAD". The WMT2015 QE task focuses on the F1 score for the "BAD" class as the main evaluation metric. For the word-level QE task, the organizers provided a data set of English-Spanish sentence pairs generated by a statistical MT system, which consists of a training set of 11,271 sentences, a development set of 1,000 sentences and a test set of 1,817 sentences. All the target sentences of the training and development data sets contain binary reference labels for each word, which were automatically derived by aligning the MT output and the post-edited translations using TERCOM (Snover et al., 2006). The distribution of the binary labels in the training and development sets is provided in Table 1.

	# Words	*% GOOD*	*% BAD*
training set	257548	80.85	19.15
dev. set	23207	80.82	19.18

Table 1: Distribution of the binary labels on the training and development set for word-level QE

2.1 Language Resources and Features

In our experiments, in addition to the provided 25 baseline features which were described in the WMT14 QE shared task (Bojar et al., 2014), we added 55 features to characterize each target word of the MT output. The new features were extracted from the provided training data and additional language resources we gathered. The new features try to model the two main MT error categories: *accuracy* and *fluency*. For *fluency*, we extracted surface-level features as well as more abstract PoS-based features and Named Entity (NE) information. For *accuracy*, we used bilingual information. In the following subsections, we describe the additional language resources and list out the additional features we used in the WMT 2015 word-level QE task. Necessary preprocessing operations are applied on the target sentences (depending on the feature type) prior to feature extraction.

2.1.1 Additional Resources

Since most of the new features rely on statistical information, we used two additional data resources. As monolingual data resource, we used a corpus of more than 13 million Spanish sentences collected from the News Crawl Corpus[1] (years 2007-2013) to build two types of language models: one based on surface forms and one based on PoS codes. The following preprocessing steps have been applied on the data before building the language models: normalizing punctuation and numbers, tokenization, named entity recognition using the Stanford NER tool (Finkel et al., 2005), lowercasing, and PoS-tagging using FreeLing (Padró and Stanilovsky, 2012). The surface form LM has been built using KenLM (Heafield 2011). For the PoS LM, we used IRSTLM with Witten-Bell smoothing (Federico et al., 2008) as the modified Kneser-Ney smoothing, which is used by KENLM, is not well defined when there are no singletons (Chen and Goodman 1999), which leads to modeling issues in the PoS corpus.

As bilingual data, we selected 6 million sentence pairs from OPUS (Tiedemann 2012) from various domains and used the Moses toolkit (Koehn et al. 2006) to obtain word and phrase alignments. Even though there are more bilingual sentences available, to avoid a bias to one specific domain, a similar number of sentences of different domains were selected. The following preprocessing steps have been applied on the data prior to training: normalization on punctuation and numbers, tokenization, NER (only the Spanish side) and lowercasing. The phrase table has been pruned to exclude alignments with a *direct alignment probability* $P(t|s) < 0.01$, where s denotes *source* and t denotes *target text*.

The resulting language models and phrase tables were stored in databases and indexed to speed up lookup.

2.1.2 Fluency Features

The fluency features try to capture whether the Spanish MT translations adhere to the norms of the Spanish language. Most of the fluency features are derived from the two language models

[1] http://www.statmt.org/wmt13/translation-task.html

described in section 2.1.1 and use the context around the focus word (w_i). To ensure computational feasibility, we limited the language models to 3-gram sequences. However, for each w_i, for which we extract a contextual feature, we generate three 3-gram features depending on the position of w_i using a sliding window approach:

- $w_{i-2}\, w_{i-1}\, w_i$
- $w_{i-1}\, w_i\, w_{i+1}$
- $w_i\, w_{i+1}\, w_{i+2}$

This sliding window approach (sw) is used for extracting all context features. In the table below, these features are indicated with "*sw*" together with the total number of features extracted by this approach.

The following fluency features were used:

- The LM score of $\boldsymbol{w_i}$ (one feature);
- (sw) The LM scores of the 3-gram context of $\boldsymbol{w_i}$ (three features);
- (sw) Binary features indicating whether a 3-gram context exists in the 3-gram database (three features);
- Separate features of the PoS codes of w_{i-1}, w_i, w_{i+1} (three features);
- Separate features of the simplified PoS codes (only main category) of w_{i-1}, w_i, w_{i+1} (three features);
- (sw) The PoS sequences of the 3-gram context of $\boldsymbol{w_i}$ (three features);
- (sw) The simplified PoS sequences of the 3-gram context of $\boldsymbol{w_i}$ (three features);
- The PoS LM score of PoS tag of $\boldsymbol{w_i}$ (one feature);
- (sw) The PoS LM scores of the 3-gram PoS context of $\boldsymbol{w_i}$ (three features);
- (sw) Binary features indicating whether a 3-gram PoS context exists in the 3-gram PoS database (three features);
- (sw) The Log-Likelihood Ratio (LLR)[2] of the 3-gram PoS context of $\boldsymbol{w_i}$ (three features);
- (sw) Binary features indicating whether the LLR of the 3-gram PoS context of the focus word is above the critical value 3.84 (95th percentile; significant al the level of $p < 0.05$) (three features);
- Binary features indicating whether $\boldsymbol{w_i}$ is the first word or the last word in a sentence (two features);
- Binary features indicating whether $\boldsymbol{w_{i-1}, w_i}$ or $\boldsymbol{w_{i+1}}$ is a NE (three features);
- (sw) NE annotation of the 3-gram context of $\boldsymbol{w_i}$ (three features).

[2] LLR compares frequencies weighted over two different corpora (in our case the Spanish MT output and the Spanish News Crawl Corpus) and assigns high LLR values to sequences in the Spanish MT output having much lower or higher frequencies than expected.

2.1.3 Accuracy Features

The accuracy features try to capture errors that can only be identified when comparing source and target sentences: wrong translations, additions and deletions. Some accuracy features are derived from the phrase table described in section 2.1.1. Other accuracy features make use of the alignment features that were given in de baseline feature set. The following accuracy features were used:

- (sw) Phrase table alignment scores of any possible alignment of words in the source sentence with words in the target sentence, containing $\boldsymbol{w_i}$, using *direct translation probability* (six features are defined for n-grams of size 1-3);
- (sw) Same phrase table as above with the additional condition that the source alignment for each $\boldsymbol{w_i}$ (which is provided as a baseline feature) is included in the alignments found (six features are defined similarly);
- Binary feature indicating whether $\boldsymbol{w_i}$ is identical to its source alignment, the alignment given as in the baseline features (one feature);
- Binary features indicating whether $\boldsymbol{w_i}$ and its source alignment are either both content words or both function words, based on the PoS codes of $\boldsymbol{w_i}$ and its source alignment, given as in the baseline features (two features).

2.2 Learning Methods

We use Conditional Random Fields (CRFs) (Lafferty et al., 2001) and Memory-Based Learning (MBL) (Daelemans and Van den Bosch, 2005) as ML methods for word-level QE. CRFs take an input sequence *X* with its associated features, and try to infer a hidden sequence *Y*, containing the class labels. They are as such comparable to Hidden Markov Models (HMMs) and Maximum Entropy Markov Models (MEMMs). However, CRFs, unlike HMMs, do not assume that all features are independent, and they can take future observations into account using a

forward-backward algorithm, unlike MEMMs, thus avoiding two fundamental limitations of those models (Lafferty et al, 2001). We used the CRF++ toolkit, version 0.58 (Lafferty et al., 2001). In MBL, on the other hand, a so-called lazy learner, which stores all training instances in memory and at classification time, a new test instance X is compared to all instances Y in the memory. The similarity between the instances is computed using a distance metric $\Delta(X, Y)$. The extrapolation is done by assigning the most frequent category within the found set of most similar example(s) (the k-nearest neighbors) as the category of the new test example. We used TiMBL, version 6.4.2 (Daelemans et al., 2010) in our experiments. In addition, we used Gallop (Desmet et al, 2013), a genetic algorithm (GA) toolbox for optimizing the classifiers on two levels: feature selection and hyper-parameter optimization.

2.3 Experiments

We carried out experiments with the two ML methods and three different feature sets, namely the baseline features (b), the new features (n) we described in Section 2.2 and a merged feature set (m), which contain all features from the first two groups. We trained CRF models with basic unigram (uni) and bigram (bi) templates and the default settings for the regularization algorithm and the hyper-parameters. While unigram templates use each feature as it is, bigram templates automatically create additional features, combining the features for $\boldsymbol{w_{i-1}}$ and $\boldsymbol{w_i}$. TiMBL learning is performed with explicitly defined numerical features. For a first round of experiments, both learners were applied relying on their default parameter settings. Figure 1 summarizes the classification results of the first round of experiments, where evaluation metrics are defined as follows:

$$Accuracy = \frac{\mathrm{tp} + \mathrm{tn}}{\mathrm{tp} + \mathrm{tn} + \mathrm{fp} + \mathrm{fn}}$$

$$F1\ "BAD" = \frac{2 \cdot P_{BAD} \cdot R_{BAD}}{P_{BAD} + R_{BAD}}$$

where tp, tn, fp, fn denote *true positives*, *true negatives*, *false positives* and *false negatives* respectively, and P_{BAD} and R_{BAD} denote *precision* and *recall* for the "BAD" class. Figure 1 shows that merging the baseline features with the newly designed features improves the classification performance on the "BAD" class for both learning methods (systems "CRF m-uni", "CRF m-bi" and "TiMBL m"). For this experiment, the unigram CRF systems generally have a better performance than the bigram systems.

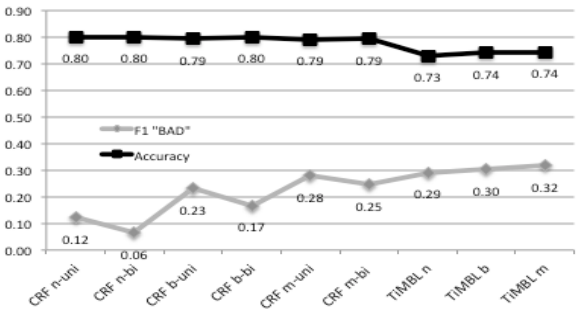

Figure 1: Classification performance of different feature groups and learning methods.

In order to gain more insight into which features are most informative for our task, we performed feature selection using a GA-based search. Given that it is by no means certain that the default parameters, in both learners, are also the optimal parameter settings for our classification task, we performed joint feature selection and parameter optimization. For this purpose, we used Gallop with 3-fold cross-validation, population size of 100 and a maximum of 50 generations.

Due to time limitations, we used a reduced training data set of 60,000 feature vectors for the Gallop experiments. Unfortunately, we were not able to improve the results of "*TiMBL m"* by using only the features or the hyper-parameters that are selected by Gallop. Some of the features that were consistently selected by Gallop in the 5-best scoring feature sequences, are the following: LM scores of 3-gram context, binary features indicating whether the 3-gram context appears in the LM or POS-LM, binary feature indicating whether the target word is identical to the source alignment, binary feature indicating whether the target word and the corresponding source alignment are both content or function words.

Based on the hypothesis that both learners use a different learning strategy and might thus make different types of errors, we performed a final experiment with classifier ensembles, using two simple methods. While the first method uses the TiMBL word-level predictions as an additional feature in CRF (hybrid-1), the second method combines the labels of the best CRF and TiMBL systems ("*CRF m-uni"* and "*TiMBL m"*) by voting for the "BAD" label if (1) any of the systems labels the target word as "BAD" (hybrid-2A) or (2) both systems label the target word as "BAD" (hybrid-2B). The classification performance of

the ensemble systems, together with the best TiMBL system, are provided in Table 2.

	Accuracy	*F1 "BAD"*
TiMBL-m	0,74	0.317
Hybrid 1	0,79	0.292
Hybrid 2A	**0,81**	0.161
Hybrid 2B	0,73	**0.375**

Table 2: Classification performance of the best TiMBL system, in comparison with the ensemble systems on the development set.

Based on all the results, we selected the following systems for the submission of this year's shared task on word-level QE:

- *SCATE-HYBRID*: Hybrid 2B
- *SCATE-MBL*: TiMBL-m

These two systems obtained comparable scores (F1 "BAD") on the test set of 0.367 and 0.305 respectively.

3 Sentence-level Quality Estimation

The sentence-level QE task aims at predicting Human mediated Translation Edit Rate (HTER) (Snover et al., 2006) between the raw MT output and its manually post-edited version. In addition to *scoring* the sentences for quality, a *ranking* variant of this task is defined as ranking all MT sentences, for all source sentences, from best to worst.

3.1 Features and Language Resources

In our experiments, in addition to 17 baseline features that were provided together with the data sets, we designed 17 additional features. In this section, we briefly list out the additional features we used in WMT 2015 sentence-level QE task. We used the same additional language resources as in the word-level QE task to extract additional features. As mentioned before, we include the word level predictions as features for sentence-level QE. The following additional features were used:

- The percentage of predicted "BAD" tokens in the target sentence (p_{BAD}).
- The percentage of PoS n-grams in the target sentence that appear in the PoS n-gram database more than once (p_{POS}). Five features are extracted for n-grams of size 2-6.
- The percentage of n-grams in the target sentence that appear in the n-gram database at least once (p_{tok}). Four features are extracted for n-grams of size 2-5.
- The percentage of n-grams in the target sentence that appear in the phrase table, being aligned to n-grams from the corresponding source sentence with *direct alignment probability (EN-to-ES)* $P(t|s) > 0.01$ (p_{pt}). Seven features are extracted for n-grams of size 1-7.

3.2 Learning Methods

We use LibSVM (Chang and Lin 2011) to train a regression model using Support Vector Machines (SVMs) with a Radial Basis Function (RBF) kernel.

3.3 Experiments

In a first set of experiments we compare the performance of a system using the baseline features with three systems using only a single feature (p_{BAD}), that is the percentage of predicted "BAD" tokens in the target sentence. We extract this feature from three different word-level QE systems "*TiMBL m"*, "*CRF m-uni"* and "*HYBRID_2B"*. The performance of these sentence-level QE systems are measured with Mean Squared Error (MSE), Squared Correlation Coefficient (r^2) and Mean Average Error (MAE), which are defined as follows:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\mathrm{f(x_i)} - \mathrm{y_i})^2$$

$$\mathrm{r^2} = \frac{(n\sum_{i=1}^{n} f(x_i)y_i - \sum_{i=1}^{n} f(x_i)\sum_{i=1}^{n} y_i)^2}{(n\sum_{i=1}^{n} f(x_i)^2 - (\sum_{i=1}^{n} f(x_i))^2)\,(n\sum_{i=1}^{n} y_i^2 - (\sum_{i=1}^{n} y_i)^2)}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\mathrm{f(x_i)} - \mathrm{y_i}|$$

where $f(x_1), \ldots, f(x_n)$ are the decision values predicted by LibSVM and $y_1, \ldots, y_n$ are the true values. We train the systems with default values for hyper-parameters and perform evaluation on the development set provided for the sentence-level QE task. Figure 3 summarizes the performance of baseline features in comparison with P_{BAD}, which is obtained from different word-level QE systems. In addition to the systems above, we build a final system, which uses the given reference labels to extract P_{BAD} (P_{BAD} - ReferenceLabels). The purpose of building and evaluating this system is to show an upper boundary for the performance of P_{BAD}, as a single feature.

	MSE	r^2	*MAE*
baseline	0,039	0,03	0,147
P_{BAD} – Timbl m	0,038	0,06	0,145
P_{BAD} - CRF m-uni	0,037	0,08	0,145
P_{BAD} - HYBRID 2B	0,036	0,10	0,144
P_{BAD} - ReferenceLabels	**0,005**	**0,89**	**0,055**

Table 3: Sentence-level QE performance of SVM systems using baseline features vs. p_{BAD} extracted from three different systems.

As a second set of experiments we enrich the baseline feature set by combining it with the additional features that are described in Section 3.1. For the feature P_{BAD} we use the best output, coming from the system "HYBRID_2B". Table 4 shows the impact of the different feature sets on the overall performance.

	MSE	r^2	*MAE*
basel.	0,037	0,03	0,147
p_{BAD}	0,036	0,07	0,147
basel.+p_{POS}	0,037	0,04	0,147
basel.+p_{POS}+p_{pt}	0,036	0,06	0,145
basel.+p_{POS}+p_{pt}+p_{tok}	0,036	0,07	0,143
basel.+p_{POS}+p_{pt}+p_{tok}+p_{BAD}	0,035	0,10	0,142

Table 4: Performance of the SVM systems on sentence-level QE, using different feature sets

Based on the results, we selected the following two systems for the submission of this year's shared task on sentence-level QE:

- *SCATE-SVM-single*: SVM trained with the single feature p_{BAD}
- *SCATE-SVM*: SVM trained with baseline and new features (base.+p_{POS}+p_{pt}+p_{tok}+p_{BAD})

	MSE	r^2	*MAE*
p_{BAD}	0.035	0.07	0,146
basel.+pos+pt+tok+p_{BAD}	**0.034**	**0.10**	**0,142**

Table 5: Performance of the submitted sentence-level QE systems on development set, compared with the baseline system.

We apply grid search to optimize the γ, ε and *C* parameters using 5-fold cross validation prior to building SVM models to use for our submissions. We perform *sentence ranking* based on the predicted HTER scores for both systems. Table 5 gives an overview of the performance of the two optimized systems we submit on the development set. On the test set, the performance (MAE) of both of these systems was 0.14, based on the official results.

4 Results and Discussion

For the word-level QE task, we extracted additional features based on *accuracy* and *fluency* of translations, for labeling words for quality as a ML classification problem. The results showed that the additional features, as a whole, were found to be relevant for the two different learning methods. We obtained better results using both MBL and CRF when we used the additional features in combination with the baseline feature set. We also observe that MBL performs better than CRF when looking at the F1 scores on the "BAD" class for this task, even though it performs worse when overall classification accuracy is considered. One possible explanation for MBL obtaining a better performance could be the use of similarity-based reasoning as a smoothing method for estimating low-frequency events, considering the heterogeneous nature of the "BAD" class for this specific task and the suitability of MBL for handling exceptions (Daelemans and Van den Bosch, 2005).

Finally, a simple combination of the two classifiers into an ensemble system provides a better system for classifying the "BAD" class, which encourages us to carry out more experiments with ensemble systems for the word-level QE task.

For sentence-level QE, we trained regression models using additional features we extracted, in combination with the baseline feature set. We see in Table 4 that a single feature, which is based only on the predicted word labels, can lead to a sentence-level QE system with better performance than a system built with 17 baseline features. For demonstrating the potential of this single feature further, we built a system based on the given correct word labels, which defines a high upper bound for quality estimations, as expected. As a result we show that a word-level QE system that is accurate "enough" can lead to successful sentence-level QE. In the future, we would like to investigate more closely the relationship between word-level and sentence-level QE and examine the portability of the developed systems to English-Dutch.

Acknowledgements

This research has been carried out in the framework of the SCATE[3] project funded by the Flemish government agency IWT.

[3] http://www.ccl.kuleuven.be/scate

References

Nguyen Bach, Fei Huang, and Yaser Al-Onaizan. 2011. "Goodness: A method for measuring machine translation confidence." In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pp. 211-219. Association for Computational Linguistics,

John Blatz, Erin Fitzgerald, George Forster, Simona Gandrabur, Cyril Goutte, Alberto Kulesza, AlexSanchis, and Nicola Ueffing. 2003. "Confidence Estimation for Machine Translation." In *Summer Workshop, Center for Language and Speech Processing*, 853–56. Genève.

Ondrej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia. 2014. "Findings of the 2014 Workshop on Statistical Machine Translation." *WMT* Ninth Work: 12–58.

Ondrej Bojar, Christian Buck, Chris Callison-Burch, Christian Federmann, Barry Haddow, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2013. Findings of the 2013 Workshop on Statistical Machine Translation. In *Eighth Workshop on Statistical Machine Translation*, pp. 1–44, WMT, Sofia, Bulgaria.

Chih-Chung Chang and Chih-Jen Lin. 2011. "LIBSVM: A library for support vector machines." *ACM Transactions on Intelligent Systems and Technology (TIST)* 2.3 (2011): 27.

Stanley Chen and Joshua Goodman. 1998. An empirical study of smoothing techniques for language modeling. *Technical Report TR-10-98*, Harvard University, August

Lluís Padró Cirera and Evgeny Stanilovsky. 2012. "FreeLing 3.0: Towards Wider Multilinguality." *International Conference on Language Resources and Evaluation*, 2473–79.

Walter Daelemans, Jakub Zavrel, Ko van der Sloot (2010). *TiMBL: Tilburg Memory Based Learner, version 6.3*. reference guide. Technical Report 10-01, ILK.

Walter Daelemans and Antal van den Bosch. 2005. *Memory-Based Language Processing*. Cambridge University Press.

Bart Desmet, Veronique Hoste, David Verstraeten, Jan Verhasselt. 2013. Gallop Documentation. Tech. Rep. LT3 13-03

Marcello Federico, Nicola Bertoldi and Mauro Cettolo. 2008. "IRSTLM: An Open Source Toolkit for Handling Large Scale Language Models." In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 1618–21.

Jenny Rose Finkel, Trond Grenager and Christopher Manning. 1997. "Incorporating Non-Local Information into Information Extraction Systems by Gibbs Sampling." In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, 363–70.

Simona Gandrabur and George Foster. 2003. "Confidence Estimation for Text Prediction." In *Proc.~Conf.~on Natural Language Learning (CoNLL)*, 95–102. Edmonton.

Kenneth Heafield. 2011. "KenLM : Faster and Smaller Language Model Queries." In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, 187–97.

Philipp Koehn, Wade Shen, Marcello Federico, Nicola Bertoldi, Chris Callison-Burch, Brooke Cowan, Chris Dyer, et al. 2006. "Open Source Toolkit for Statistical Machine Translation." In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, 177–80.

John Lafferty, Andrew Mccallum and Fernando Pereira. 2001. "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In Proceedings of 18th." *International Conference on Machine Learning* pages: 282–89.

Arle Richard Lommel, Aljoscha Burchardt, Maja Popovic, Kim Harris, Eleftherios Avramidis, Hans Uszkoreit. 2014. Using a new analytic measure for the annotation and analysis of MT errors on real data. *Proceedings of the 17th Annual Conference of the European Association for Machine Translation, Pages 165-172, Dubrovnik, Croatia, European Association for Machine Translation, Croatian Language Technologies Society.*

Antti-Veikko I Rosti, Bing Xiang, Spyros Matsoukas, Richard Schwartz, Necip Fazil Ayan and Bonnie J Dorr. 2007. "Combining Output from Multiple Machine Translation Systems." In

Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference, 228–35.

Alina Secară. 2005. "Translation Evaluation - a State of the Art Survey." In *Proceedings of the eCoLoRe/MeLLANGE Workshop, Leeds*, 39–44.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla and John Makhoul. 2006. "A Study of Translation Edit Rate with Targeted Human Annotation." In *Proceedings of Association for Machine Translation in the Americas*, 223–31.

José GC De Souza, Jesús González-Rubio, Christian Buck, Marco Turchi, and Matteo Negri. 2014. "FBK-UPV-UEdin Participation in the WMT14 Quality Estimation Shared-Task." In *Acl 2014*, 322–28.

Lucia Specia, Marco Turchi, Nicola Cancedda, Marc Dymetman and Nello Cristianini. 2009. "Estimating the Sentence-Level Quality of Machine Translation Systems." In *13th Conference of the European Association for Machine Translation*, 28–37.

Lucia Specia, Kashif Shah, José GC De Souza, and Trevor Cohn. 2013. "QuEst-A translation quality estimation framework." In *ACL (Conference System Demonstrations)*, pp. 79-84.

Jörg Tiedemann. 2012. "Parallel Data, Tools and Interfaces in OPUS." In *Lrec*, 2214–18.

Nicola Ueffing and Hermann Ney. 2007. "Word-Level Confidence Estimation for Machine Translation." *Computational Linguistics* 33 (1). MIT Press: 9–40.

S. John White. 1995. "Approaches to Black Box MT Evaluation." In *MT Summit V Proceedings*, 10.

Deyi Xiong, Min Zhang, and Haizhou Li. 2010. "Error detection for statistical machine translation using linguistic features." In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 604-611. Association for Computational Linguistics,