Target-Centric Features for Translation Quality Estimation

Chris Hokamp and **Iacer Calixto** and **Joachim Wagner** and **Jian Zhang**
CNGL Centre for Global Intelligent Content
Dublin City University
School of Computing
Dublin, Ireland
{chokamp|icalixto|jwagner|zhangj}@computing.dcu.ie

Abstract

We describe the DCU-MIXED and DCU-SVR submissions to the WMT-14 Quality Estimation task 1.1, predicting sentence-level perceived post-editing effort. Feature design focuses on target-side features as we hypothesise that the source side has little effect on the quality of human translations, which are included in task 1.1 of this year's WMT Quality Estimation shared task. We experiment with features of the QuEst framework, features of our past work, and three novel feature sets. Despite these efforts, our two systems perform poorly in the competition. Follow up experiments indicate that the poor performance is due to improperly optimised parameters.

1 Introduction

Translation quality estimation tries to predict the quality of a translation given the source and target text but no reference translations. Different from previous years (Callison-Burch et al., 2012; Bojar et al., 2013), the WMT 2014 Quality Estimation shared task is MT system-independent, i. e. no glass-box features are available and translations in the training and test sets are produced by different MT systems and also by human translators.

This paper describes the CNGL@DCU team submission to task 1.1 of the WMT 2014 Quality Estimation shared task.[1] The task is to predict the **perceived** post-editing effort given a source sentence and its raw translation. Due to the inclusion of human translation in the task, we focus our efforts on target-side features as we expect that the quality of a translation produced by a human translator is much less affected by features of the source than by extrinsic factors such as time pressure and familiarity with the domain.

To build our quality estimation system, we use and extend the QuEst framework for translation quality estimation[2] (Shah et al., 2013; Specia et al., 2013). QuEst provides modules for feature extraction and machine learning. We modify both the feature extraction framework and the machine learning components to add functionality to QuEst.

The novel features we add to our systems are *(a)* a language model on a combination of stop words and POS tags, *(b)* inverse glass-box features for translating the translation, and *(c)* random indexing (Sahlgren, 2005) for measuring the semantic similarity of source and target side across languages. Furthermore, we integrated *(d)* source-side pseudo-reference features (Soricut and Echihabi, 2010) and *(e)* error grammar features (Wagner, 2012), which were used first in MT quality estimation by (Rubino et al., 2012; Rubino et al., 2013).

The remaining sections are organised as follows. Section 2 gives details on the features we use. Section 3 describes how we set up our experiments. Results are presented in Section 4 and conclusions are drawn in Section 5 together with pointers to future work.

2 Features

This section describes the features we extract from source and target sentences in order to train prediction models and to make predictions in addition to the baseline features provided for the task.

We focus on the target side as we assume that the quality of the source side has little predictive power for human translations, which are included in task 1.1.

[1] A CNGL system based on referential translation machines is submitted separately (Biçici and Way, 2014).

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

Proceedings of the Ninth Workshop on Statistical Machine Translation, pages 329–334,
Baltimore, Maryland USA, June 26–27, 2014.

2.1 QuEst Black-Box Features and Baseline Features

We use the QuEst framework to extract 47 basic black-box features from both source and target side, such as the ratio of the number of tokens, punctuation statistics, number if mismatched brackets and quotes, language model perplexity, n-gram frequency quartile statistics ($n = 1, 2, 3$), and coarse-grained POS frequency ratios. 17 of the 47 features are identical to the baseline features from the shared task website, i. e. 30 features are new. To train the language models and to extract frequency information, we use the News Commentary corpus (Bojar et al., 2013).

2.2 POS and Stop Word Language Model Features

For all languages, we extract probability and perplexity features from language models trained on POS tagged corpora. POS tagging is performed using the IMS Tree Tagger (Schmid, 1994).

We also experiment with language models built from a combination of stop words[3] and POS tags. Starting with a tokenised corpus, and its POS-tagged counterpart, we create a new representation of the corpus by replacing POS tags for stop words with the literal stop word that occurred in the original corpus, leaving non-stop word tags intact.[4] The intuition behind the approach is that the combined POS and stop word model should encode the distributional tendencies of the most common words in the language.

The log-probability and the perplexity of the target side are used as features. The development of these features was motivated by manual examination of the common error types in the training data. We noted that stop word errors (omission, mistranslation, mis-translation of idiom), are prevalent in all language pairs, indicating that features which focus on stop word usage could be useful for predicting the quality of machine translation. We implement POS and stop word language models inside the QuEst framework.

2.3 Source-Side Pseudo-Reference Features

We extract source-side pseudo-reference features (Albrecht and Hwa, 2008; Soricut and Echihabi, 2010; Rubino et al., 2012), for English to German quality prediction using a highly-tuned German to English translation system (Li et al., 2014) working in the reverse direction. The MT system translates the German target side, the quality of which is to be predicted, back into English, and we extract pseudo-reference features on the source side:

- BLEU score (Papineni et al., 2002) between back-translation and original source sentence, and
- TER score (Snover et al., 2006).

For the 5th English to German test set item, for example, the translation

(1) *Und belasse sie dort eine Woche.*

is translated back to English as

(2) *and leave it there for a week .*

and compared to the original source sentence

(3) *Leave for a week.*

producing a BLEU score of 0.077 using the Python interface to the cdec toolkit (Chahuneau et al., 2012).

2.4 Inverse Glass-Box Features for Translating the Translation

In the absence of direct glass-box features, we obtain glass-box features from translating the raw translation back to the source language using the same MT system that we use for the source-side pseudo-reference features. We extract features from the following components of the Moses decoder: distortion model, language model, lexical reordering, lexical translation probability, operational sequence model (Durrani et al., 2013), phrase translation probability, and the decoder score.

The intuition for this set of features is that back-translating an incorrect translation will give low system-internal scores, e. g. a low phrase translation score, and produce poor output with low language model scores (garbage in, garbage out).

We are not aware of any previous work using inverse glass-box features of translating the target side to another language for quality estimation.

[3] We use the stop word lists from Apache Lucene (McCandless et al., 2010).

[4] The News Commentary corpus from WMT13 was used to build these models, same as for the black-box features (Section 2.1).

2.5 Semantic Similarity Using Random Indexing

These features try to measure the semantic similarity of source and target side of a translation unit for quality estimation using random indexing (Sahlgren, 2005). We experiment with adding the similarity score of the source and target random vectors.

For each source and target pair in the English-Spanish portion of the Europarl corpus (Koehn, 2005), we initialize a sparse random vector. We then create token vectors for each source and target token by summing the vectors for all of the segments where the token occurs. To extract the similarity feature for new source and target pairs, we map them into the vector space by taking the centroid of the token vectors for the source side and the target side, and computing their cosine similarity.

2.6 Error Grammar Parsing

We obtain features from monolingual parsing with three grammars:

1. the vanilla grammar shipped with the Blipp parser (Charniak, 2000; Charniak and Johnson, 2005) induced from the Penn-Treebank (Marcus et al., 1994),
2. an error grammar induced from Penn-Treebank trees distorted according to an error model (Foster, 2007), and
3. a grammar induced from the union of the above two treebanks.

Features include the log-ratios between the probability of the best parse obtained with each grammar and structural differences measured with Parseval (Black et al., 1991) and leaf-ancestor (Sampson and Babarczy, 2003) metrics. These features have been shown to be useful for judging the grammaticality of sentences (Wagner et al., 2009; Wagner, 2012) and have been used in MT quality estimation before (Rubino et al., 2012; Rubino et al., 2013).

3 Experimental Setup

This section describes how we set up our experiments.

3.1 Cross-Validation

Decisions about parameters are made in 10-fold cross-validation on the training data provided for the task. As the datasets for task 1.1 include three to four translations for each source segment, we group segments by their source side and split the data for cross-validation between segments to ensure that a source segment does not occur in both training and test data for any of the cross-validation runs.

We implement these modifications to cross-validation and randomisation in the QuEst framework.

3.2 Training

We use the QuEst framework to train our models. Support vector regression (SVR) meta-parameters are optimised using QuEst's default settings, exploring RBF kernels with two possible values for each of the three meta-parameters C, γ and ϵ.[5]

The two final models are trained on the full training set with the meta-parameters that achieved the best average cross-validation score.

3.3 Classifier Combination

We experiment with combining logistic regression (LR) and support vector regression (SVR) by first choosing the instances where LR classification is confident and using the LR class label (1, 2, or 3) as predicted perceived post-editing effort, and falling back to SVR for all other instances.

We employ several heuristics to decide whether to use the output of LR or SVR. As the LR classifier learns a decision function for each of the three classes, we can exploit the scores of the classes to measure the confidence of the LR classifier about its decision. If the LR classifier is confident, we use its prediction directly, otherwise we use the SVR prediction.

For the cases where one of the three decision functions for the LR classifier is positive, we select the prediction directly, falling back to SVR when the classifier is not confident about any of the three classes. We implement the LR+SVR classifier combination inside the QuEst framework.

4 Results

Table 1 shows cross-validation results for the 17 baseline features, the combination of all features and target-side features only. We do not show combinations of individual feature sets and baseline features that do not improve over the base-

[5] We only discovered this limitation of the default configuration after the system submission, see Sections 4 and 5.

Features	Classifier	RMSE	MAE
Basel.17	LR+SVR	0.75	0.62
ALL	LR+SVR	0.74	0.59
ALL	LR> 0.5+SVR	0.75	0.58
Target	LR+SVR	0.75	0.59
ALL	LR> 0.5+SVR-r	0.78	**0.55**

Table 1: Cross-validation results for English to German. LR > 0.5 indicates that we require the LR decision function to be > 0.5. SVR-r rounds the output to the nearest natural number.

line. Several experiments, including those with the semantic similarity feature sets, are thus omitted. Furthermore, we only exemplify one language pair (English to German), as the other language pairs show similar patterns. The feature set *target* contains the subset of the QuEst black-box features (Section 2.1) which only examine the target side.

Our best results for English to German in the cross-validation experiments are achieved by combining a logistic regression (LR) classifier with support vector regression (SVR). Furthermore, performance on the cross-validation is slightly improved for the mean absolute error (MAE) by rounding SVR scores to the nearest integer. For the root-mean-square error (RMSE), rounding has the opposite effect.

Performing a more fine-grained grid search for the meta-parameters C, γ and ϵ after system submission, we were able to match the scores for the baseline features published on the shared task website.

4.1 Parameters for the Final Models

The final two models for system submission are trained on the full data set. We submit our best system according to MAE in cross-validation combining LR, SVR and rounding with all features (ALL) as DCU-MIXED. For our second submission, we choose SVR on its own (system DCU-SVR). For English-Spanish, we only submit DCU-SVR.

5 Conclusions and Future Work

We identified improperly optimised parameters of the SVR component as the cause, or at least as a contributing factor, for the placement of our systems below the official baseline system. Other potential factors may be an error in our experimental setup or over-fitting. Therefore, we plan to repeat the experiments with a more fine-grained grid search for optimal parameters and/or will try another machine learning toolkit.

Unfortunately, due to the above problems with our system so far, we cannot draw conclusions about the effectiveness of our novel feature sets.

A substantial gain is achieved on the MAE metric with the rounding method, indicating that the majority of prediction errors are below 0.5.[6] Future work should account for this effect. Two ideas are: *(a)* round all predictions before evaluation and *(b)* use more fine-grained gold values, e. g. the (weighted) average over multiple annotations as in the WMT 2012 quality estimation task (Callison-Burch et al., 2012).

For the error grammar method, the next step will be to adjust the error model to errors found in translations. It may be possible to do this without a time-consuming analysis of errors: Wagner (2012) suggests to use parallel data of authentic errors and corrections to build the error grammar, first parsing the corrections and then guiding the error creation procedure with the edit operations inverse to the corrections. Post-editing corpora can play this role and have recently become available (Potet et al., 2012).

Furthermore, future work should explore the inverse glass-box feature idea with arbitrary target languages for the MT system. (There is no requirement that the glass-box system translates back to the original source language).

Finally, we would like to integrate referential translation machines (Biçici, 2013; Biçici and Way, 2014) into our system as they performed well in the WMT quality estimation tasks this and last year.

Acknowledgments

This research is supported by the European Commission under the 7th Framework Programme, specifically its Marie Curie Programme 317471, and by the Science Foundation Ireland (Grant 12/CE/I2267) as part of CNGL (www.cngl.ie) at Dublin City University. We thank the anonymous reviewers and Jennifer Foster for their comments on earlier versions of this paper.

[6] The simultaneous increase on RMSE can be explained if there is a sufficient number of errors above 0.5: After squaring, these errors are still quite small, e. g. 0.36 for an error of 0.6, but after rounding, the square error becomes 1.0 or 4.0.

References

Joshua Albrecht and Rebecca Hwa. 2008. The role of pseudo references in MT evaluation. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 187–190, Columbus, Ohio, June. Association for Computational Linguistics.

Ergun Biçici and Andy Way. 2014. Referential translation machines for predicting translation quality. In *Proceedings of the Nineth Workshop on Statistical Machine Translation*, Baltimore, USA, June. Association for Computational Linguistics.

Ergun Biçici. 2013. Referential translation machines for quality estimation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 343–351, Sofia, Bulgaria, August. Association for Computational Linguistics.

Ezra Black, Steve Abney, Dan Flickinger, Claudia Gdaniec, Robert Grishman, Philip Harrison, Donald Hindle, Robert Ingria, Fred Jelinek, Judith Klavans, Mark Liberman, Mitchell Marcus, Salim Roukos, Beatrice Santorini, and Tomek Strzalkowski. 1991. A procedure for quantitatively comparing the syntactic coverage of English grammars. In E. Black, editor, *Proceedings of the HLT Workshop on Speech and Natural Language*, pages 306–311, Morristown, NJ, USA. Association for Computational Linguistics.

Ondřej 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 *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 1–44, Sofia, Bulgaria, August. Association for Computational Linguistics.

Chris Callison-Burch, 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 Seventh Workshop on Statistical Machine Translation*, pages 10–51, Montréal, Canada, June. Association for Computational Linguistics.

Victor Chahuneau, Noah A. Smith, and Chris Dyer. 2012. pycdec: A python interface to cdec. *Prague Bull. Math. Linguistics*, 98:51–62.

Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n-best-parsing and maxent discriminative reranking. In *Proceedings of the 43rd Annual Meeting of the ACL (ACL-05)*, pages 173–180, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Eugene Charniak. 2000. A maximum entropy inspired parser. In *Proceedings of the First Annual Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-00)*, pages 132–139, Seattle, WA.

Nadir Durrani, Barry Haddow, Kenneth Heafield, and Philipp Koehn. 2013. Edinburgh's machine translation systems for European language pairs. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 114–121, Sofia, Bulgaria, August. Association for Computational Linguistics.

Jennifer Foster. 2007. Treebanks gone bad: Parser evaluation and retraining using a treebank of ungrammatical sentences. *International Journal on Document Analysis and Recognition*, 10(3-4):129–145.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86.

Liangyou Li, Xiaofeng Wu, Santiago Cortés Vaíllo, Jun Xie, Jia Xu, Andy Way, and Qun Liu. 2014. The DCU-ICTCAS-Tsinghua MT system at WMT 2014 on German-English translation task. In *Proceedings of the Nineth Workshop on Statistical Machine Translation*, Baltimore, USA, June. Association for Computational Linguistics.

Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The Penn Treebank: Annotating predicate argument structure. In *Proceedings of the 1994 ARPA Speech and Natural Language Workshop*, pages 114–119.

Michael McCandless, Erik Hatcher, and Otis Gospodnetic. 2010. *Lucene in Action, Second Edition: Covers Apache Lucene 3.0*. Manning Publications Co., Greenwich, CT, USA.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL02)*, pages 311–318, Morristown, NJ, USA. Association for Computational Linguistics.

Marion Potet, Emmanuelle Esperança-Rodier, Laurent Besacier, and Hervé Blanchon. 2012. Collection of a large database of French-English SMT output corrections. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*.

Raphael Rubino, Jennifer Foster, Joachim Wagner, Johann Roturier, Rasul Samad Zadeh Kaljahi, and Fred Hollowood. 2012. Dcu-symantec submission for the wmt 2012 quality estimation task. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 138–144, Montréal, Canada, June. Association for Computational Linguistics.

Raphael Rubino, Joachim Wagner, Jennifer Foster, Johann Roturier, Rasoul Samad Zadeh Kaljahi, and Fred Hollowood. 2013. DCU-Symantec at the

WMT 2013 quality estimation shared task. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 392–397, Sofia, Bulgaria, August. Association for Computational Linguistics.

Magnus Sahlgren. 2005. An introduction to random indexing. In *Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering (TKE)*, volume 5, Copenhagen, Denmark.

Geoffrey Sampson and Anna Babarczy. 2003. A test of the leaf-ancestor metric for parse accuracy. *Natural Language Engineering*, 9(4):365–380.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*, Manchester, United Kingdom.

Kashif Shah, Eleftherios Avramidis, Ergun Biçici, and Lucia Specia. 2013. QuEst - design, implementation and extensions of a framework for machine translation quality estimation. *The Prague Bulletin of Mathematical Linguistics*, 100.

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*, pages 223–231.

Radu Soricut 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.

Lucia Specia, Kashif Shah, Jose G.C. de Souza, and Trevor Cohn. 2013. QuEst - a translation quality estimation framework. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 79–84, Sofia, Bulgaria, August. Association for Computational Linguistics.

Joachim Wagner, Jennifer Foster, and Josef van Genabith. 2009. Judging grammaticality: Experiments in sentence classification. *CALICO Journal (Special Issue of the 2008 CALICO Workshop on Automatic Analysis of Learner Language)*, 26(3):474–490.

Joachim Wagner. 2012. *Detecting grammatical errors with treebank-induced, probabilistic parsers*. Ph.D. thesis, Dublin City University, Dublin, Ireland.