Investigating Continuous Space Language Models for Machine Translation Quality Estimation

Kashif Shah[§], **Raymond W. M. Ng**[§], **Fethi Bougares**[†], **Lucia Specia**[§]
§Department of Computer Science, University of Sheffield, UK
{kashif.shah, wm.ng, l.specia}@sheffield.ac.uk

†LIUM, University of Le Mans, France
fethi.bougares@lium.univ-lemans.fr

Abstract

We present novel features designed with a deep neural network for Machine Translation (MT) Quality Estimation (QE). The features are learned with a Continuous Space Language Model to estimate the probabilities of the source and target segments. These new features, along with standard MT system-independent features, are benchmarked on a series of datasets with various quality labels, including post-editing effort, human translation edit rate, post-editing time and METEOR. Results show significant improvements in prediction over the baseline, as well as over systems trained on state of the art feature sets for all datasets. More notably, the addition of the newly proposed features improves over the best QE systems in WMT12 and WMT14 by a significant margin.

1 Introduction

Quality Estimation (QE) is concerned with predicting the quality of Machine Translation (MT) output without reference translations. QE is addressed with various features indicating fluency, adequacy and complexity of the translation pair. These features are used by a machine learning algorithm along with quality labels given by humans to learn models to predict the quality of unseen translations.

A variety of features play a key role in QE. A wide range of features from source segments and their translated segments, extracted with the help of external resources and tools, have been proposed. These go from simple, language-independent features, to advanced, linguistically motivated features. They include features that summarise how the MT systems generate translations, as well as features that are oblivious to the systems. The majority of the features in the literature are extracted from each sentence pair in isolation, ignoring the context of the text. QE performance usually differs depending on the language pair, the specific quality score being optimised (e.g., post-editing time vs translation adequacy) and the feature set. Features based on n-gram language models, despite their simplicity, are among those with the best performance in most QE tasks (Shah et al., 2013b). However, they may not generalise well due to the underlying discrete nature of words in n-gram modelling.

Continuous Space Language Models (CSLM), on the other hand, have shown their potential to capture long distance dependencies among words (Schwenk, 2012; Mikolov et al., 2013). The assumption of these models is that semantically or grammatically related words are mapped to similar geometric locations in a high-dimensional continuous space. The probability distribution is thus much smoother and therefore the model has a better generalisation power on unseen events. The representations are learned in a continuous space to estimate the probabilities using neural networks with single (called shallow networks) or multiple (called deep networks) hidden layers. Deep neural networks have been shown to perform better than shallow ones due to their capability to learn higher-level, abstract representations of the input (Arisoy et al., 2012). In this paper, we explore the potential of these models in context of QE for MT. We obtain more robust features with CSLM and improve the overall prediction power for translation quality.

The paper is organised as follows: In Section 2 we briefly present the related work. Section 3 describes the CSLM model training and its various settings. In Section 4 we propose the use of CSLM features for QE. In Section 5 we present our experiments along with their results.

2 Related Work

For a detailed overview of various features and algorithms for QE, we refer the reader to the

Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 1073–1078,
Lisbon, Portugal, 17-21 September 2015.

WMT12-14 shared tasks on QE (Callison-Burch et al., 2012; Bojar et al., 2013; Ling et al., 2014). Most of the research work lies on deciding which aspects of quality are more relevant for a given task and designing feature extractors for them. While simple features such as counts of tokens and language model scores can be easily extracted, feature engineering for more advanced and useful information can be quite labour-intensive.

Since their introduction in (Bengio et al., 2003), neural network language models have been successfully exploited in many speech and language processing problems, including automatic speech recognition (Schwenk and Gauvain, 2005; Schwenk, 2007) and machine translation (Schwenk, 2012).

Recently, (Banchs et al., 2015) used a Latent Semantic Indexing approach to model sentences as bag-of-words in a continuous space to measure cross language adequacy. (Tan et al., 2015) proposed to train models with deep regression for machine translation evaluation in a task to measure semantic similarity between sentences. They reported positive results on simple features; larger feature sets did not improve these results.

In this paper, we propose to estimate the probabilities of source and target segments with continuous space language models based on a deep architecture and to use these estimated probabilities as features along with standard feature sets in a supervised learning framework. To the best of our knowledge, such approach has not been studied before in the context of QE for MT. The result shows significant improvements in many prediction tasks, despite its simplicity. Monolingual data for source and target language is the only resource required to extract these features.

3 Continuous Space Language Models

A key factor for quality inference of a translated text is to determine the fluency of such a text and how well it conforms to the linguistic regularities of the target language. It involves grammatical correctness, idiomatic and stylistic word choices that can be derived by using n-gram language models. However, in high-order n-grams, the parameter space is sparse and conventional modelling is inefficient. Neural networks model the non-linear relationship between the input features and target outputs. They often outperform conventional techniques in difficult machine learning tasks. Neural network language models (CSLM) alleviate the curse of dimensionality by projecting words into a continuous space, and modelling and estimating probabilities in this space.

The architecture of a deep CSLM is illustrated in Figure 1. The inputs to a CSLM model are the $(K-1)$ left-context words $(w_{i-K+1}, \ldots, w_{i-2}, w_{i-1})$ to predict w_i. A one-hot vector encoding scheme is used to represent the input w_{i-k} with an N-dimensional vector. The output of CSLM is a vector of posterior probabilities for all words in vocabulary, $P(w_i|w_{i-1}, w_{i-2}, \ldots, w_{i-K+1})$. Due to the large output layer (vocabulary size), the complexity of a basic neural network language model is very high. Schwenk (2007) proposed efficient training strategies in order to reduce the computational complexity and speed up the training time. They process several examples at once and use a *short-list* vocabulary V with only the most frequent words.

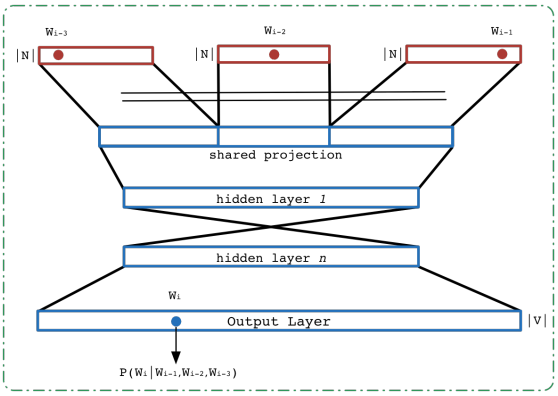

Figure 1: Deep CSLM architecture.

Following the settings mentioned in (Schwenk et al., 2014), all CSLM experiments described in this paper are performed using deep networks with four hidden layers: first layer for the projection (320 units for each context word) and three hidden layers of 1024 units with *tanh* activation. At the output layer, we use a *softmax* activation function applied to a *short-list* of the 32k most frequent words. The probabilities of the out-of-vocabulary words are obtained from a standard back-off n-gram language model. The projection of the words onto the continuous space and the training of the neural network is done by the standard back-propagation algorithm and outputs are the converged posterior probabilities. The model parameters are optimised on a development set.

4 CSLM and Quality Estimation

In the context of MT, CSLMs are generally trained on the target side of a given language pair to ex-

press the probability that the generated sentence is "correct" or "likely", without looking at the source sentence. However, QE is also concerned with how well the source segments can be translated. Therefore, we trained two models, one for each side of a given language pair. We extracted the probabilities for QE training and test sets for both source and its translation with their respective models and used them as features, along with other features, in a supervised learning setting.

Finally, we also used CSLM in a spoken language translation (SLT) task. In SLT, an automatic speech recogniser (ASR) is used to decode the source language text from audio. This creates an extra source of variability, where different ASR models and configurations give different outputs. In this paper, we use QE to exploit different ASR outputs (i.e. MT inputs) which in turn can lead to different MT outputs.

5 Experiments

We focus on experiments with sentence level QE tasks. Our English-Spanish experiments are based on the WMT QE shared task data from 2012 to 2015.[1] These tasks are diverse in nature, with different sizes and labels such as post-editing effort (PEE), post-editing time (PET) and human translation error rate (HTER). The results reported in Section 5.5 are directly comparable with the official systems submitted for each of the respective tasks. We also performed experiments on the IWSLT 2014 English-French SLT task [2] to study the applicability of our models on n-best ASR (MT inputs) comparison.

5.1 QE Datasets

In Table 1 we summarise the data and tasks for our experiments. We refer readers to the WMT and IWSLT websites for detailed descriptions of these datasets. All datasets are publicly available.

WMT12: English-Spanish news sentence translations produced by a Moses "baseline" statistical MT (SMT) system, and judged for perceived post-editing effort in 1–5 (highest-lowest), taking a weighted average of three annotators (Callison-Burch et al., 2012).

WMT13 (Task-1): English-Spanish sentence translations of news texts produced by a Moses "baseline" SMT system. These were then post-edited by a professional translator and labelled using HTER. This is a superset of the WMT12 dataset, with 500 additional sentences for test, and a different quality label (Bojar et al., 2013).

WMT14 (Task-1.1): English-Spanish news sentence translations. The dataset contains source sentences and their human translations, as well as three versions of machine translations: by an SMT system, a rule-based system system and a hybrid system. Each translation was labelled by professional translators with 1-3 (lowest-highest) scores for perceived post-editing effort.

WMT14 (Task-1.3): English-Spanish news sentence translations post-edited by a professional translator, with the post-editing time collected on a sentence-basis and used as label (in milliseconds).

WMT15 (Task-1): Large English-Spanish news dataset containing source sentences, their machine translations by an online SMT system, and the post-editions of the translation by crowdsourced translators, with HTER used as label.

IWSLT14: English-French dataset containing source language data from the 10-best (sentences) ASR system output. On the target side, the 1-best MT translation is used. The ASR system leads to different source segments, which in turn lead to different translations. METEOR (Banerjee and Lavie, 2005) is used to label these alternative translations against a reference (human) translation. Both ASR and MT outputs come from a system submission in IWSLT 2014 (Ng et al., 2014). The ASR system is a multi-pass deep neural network tandem system with feature and model adaptation and rescoring. The MT system is a phrase-based SMT system produced using Moses.

Dataset	Lang.	Train	Test	Label
WMT12	en-es	1,832	422	PEE 1-5
WMT13	en-es	2,254	500	HTER 0-1
WMT14$_{task1.1}$	en-es	3,816	600	PEE 1-3
WMT14$_{task1.3}$	en-es	650	208	PET (ms)
WMT15	en-es	11,271	1,817	HTER 0-1
IWSLT14	en-fr	8,180	11,240	MET. 0-1

Table 1: QE datasets: # sentences and labels.

5.2 CSLM Dataset

The dataset used for CSLM training consists of Europarl, News-commentary and News-crawl corpus. We used a data selection method (Moore

[1] http://www.statmt.org/wmt[12,13,14,15]/quality-estimation-task.html

[2] https://sites.google.com/site/iwsltevaluation2014/slt-track

and Lewis, 2010) to select the most relevant training data with respect to a development set. For English-Spanish, the development data is the concatenation of newstest2012 and newstest2013 of the WMT translation track. For English-French, the development set is the concatenation of the IWSLT dev2010 and eval2010. In Table 2 we show statistics on the selected monolingual data used to train back-off LM and CSLM.

Lang.	Train	Dev	LM ppl	CSLM ppl
en	4.3G	137.7k	164.63	116.58 (29.18%)
fr	464.7M	54K	99.34	64.88 (34.68%)
es	21.2M	149.4k	145.49	87.14 (40.10%)

Table 2: Training data size (number of tokens) and language models perplexity (ppl). The values in parentheses in last column shows percentage decrease in perplexity.

5.3 Feature Sets

We use the `QuEst` [3] toolkit (Specia et al., 2013; Shah et al., 2013a) to extract two feature sets for each dataset:

- **BL**: 17 features used as baseline in the WMT shared tasks on QE.
- **AF**: 80 augmented MT system-independent features[4] (superset of **BL**). For the En-Fr SLT task, we have additional 36 features (21 ASR + 15 MT-dependent features)

The resources used to extract these features (corpora, etc.) are also available as part of the WMT shared tasks on QE. The CSLM features for each of the source and target segments are extracted using the procedure described in Section 3 with the CSLM toolkit. [5]

We trained QE models with following combination of features:

- **BL + CSLM**$_{src,tgt}$: CSLM features for source and target segments, plus the baseline features.
- **AF + CSLM**$_{src,tgt}$: CSLM features for source and target segments, plus all available features.

For the WMT12 task, we performed further experiments to analyse the improvements with CSLM:

- **CSLM**$_{src}$: Source side CSLM feature only.
- **CSLM**$_{tgt}$: Target side CSLM feature only.
- **CSLM**$_{src,tgt}$: Source and target CSLM features by themselves.
- **FS(AF) + CSLM**$_{src,tgt}$: CSLM features in addition to the best performing feature set (**FS(AF)**) selected as described in (Shah et al., 2013b; Shah et al., 2015).

5.4 Learning algorithms

We use the Support Vector Machines implementation of the `scikit-learn` toolkit to perform regression (SVR) with either Radial Basis Function (RBF) or linear kernel and parameters optimised via grid search. To evaluate the prediction models we use Mean Absolute Error (MAE), its squared version – Root Mean Squared Error (RMSE), and Pearson's correlation (r) score.

Task	System	#feats	MAE	RMSE	r
WMT12	BL	17	0.6821	0.8117	0.5595
	AF	80	0.6717	0.8103	0.5645
	BL + CSLM$_{src,tgt}$	19	0.6463	0.7977	0.5805
	AF + CSLM$_{src,tgt}$	82	0.6462	0.7946	0.5825
WMT13	BL	17	0.1411	0.1812	0.4612
	AF	80	0.1399	0.1789	0.4751
	BL + CSLM$_{src,tgt}$	19	0.1401	0.1791	0.4771
	AF + CSLM$_{src,tgt}$	82	0.1371	0.1750	0.4820
WMT14 Task 1.1	BL	17	0.5241	0.6591	0.2502
	AF	80	0.4896	0.6349	0.3310
	BL + CSLM$_{src,tgt}$	19	0.4931	0.6351	0.3545
	AF + CSLM$_{src,tgt}$	82	0.4628*	0.6165*	0.3824*
WMT14 Task 1.3	BL	17	0.1798	0.2865	0.5661
	AF	80	0.1753	0.2815	0.5871
	BL + CSLM$_{src,tgt}$	19	0.1740	0.2758	0.6243
	AF + CSLM$_{src,tgt}$	82	0.1701**	0.2734	0.6201
WMT15	BL	17	0.1562	0.2036	0.1382
	AF	80	0.1541	0.1995	0.2205
	BL + CSLM$_{src,tgt}$	19	0.1501	0.1971	0.2611
	AF + CSLM$_{src,tgt}$	82	0.1471	0.1934	0.2862
IWSLT14	BL	17	0.1390	0.1791	0.5012
	AF	116	0.1361	0.1775	0.5211
	BL + CSLM$_{src,tgt}$	19	0.1358	0.1750	0.5321
	AF + CSLM$_{src,tgt}$	118	0.1337	0.1728	0.5445

Table 3: Results for datasets with various feature sets. Figures with * beat the official best systems, and with ** are second best. Results with CSLM features are significantly better than BL and AF on all tasks (paired t-test with $p \leq 0.05$).

Task	System	#feats	MAE	RMSE	r
WMT12	BL + CSLM$_{src}$	18	0.6751	0.8125	0.5626
	BL + CSLM$_{tgt}$	18	0.6694	0.8023	0.5815
	CSLM$_{src,tgt}$	2	0.6882	0.8430	0.5314
	FS(AF)	19	0.6131	0.7598	0.6296
	FS(AF) + CSLM$_{src,tgt}$	21	0.5950*	0.7442*	0.6482*

Table 4: Impact of different combinations of CSLM features on the WMT12 task. Figures with * beat the official best system. Results with CSLM features are significantly better than BL and AF on all tasks (paired t-test with $p \leq 0.05$).

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

[4] 80 features `http://www.quest.dcs.shef.ac.uk/quest_files/features_blackbox`

[5] `http://www-lium.univ-lemans.fr/cslm/`

5.5 Results

Table 3 presents the results with different feature sets for data from various shared tasks. It can be noted that CSLM features always bring significant improvements whenever added to either baseline or augmented feature set. A reduction in both error scores (MAE and RMSE) as well as an increase in Pearson's correlation with human labels can be observed on all tasks. It is also worth noticing that the CSLM features bring improvements over all tasks with different labels, evidencing that different optimisation objectives and language pairs can benefit from these features. However, the improvements are more visible when predicting post-editing effort for WMT12 and WMT14's Task 1.1. For these two tasks, we are able to achieve state-of-the-art performance by adding the two CSLM features to all available or selected feature sets.

For WMT12, we performed another set of experiments to study the effect of CSLM features by themselves and in combination. The results in Table 4 show that the target side CSLM feature bring larger improvements than its source side counterpart. We believe that it is because the target side feature directly reflects the fluency of the translation, whereas the source side feature (regarded as a translation complexity feature) only has indirect effect on quality. Interestingly, the two CSLM features alone give comparable results (slightly worse) than the BL feature set [6] despite the fact that these 17 features cover many complexity, adequacy and fluency quality aspects. CSLM features bring further improvements on pre-selected feature sets, as shown in Table 3. We also performed feature selection over the full feature set along with CSLM features, following the procedure in (Shah et al., 2013b). Interestingly, both CSLM features were selected among the top ranked features, confirming their relevance.

In order to investigate whether our CSLM features 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. Table 5 shows the official results from the shared task (Off.) (Callison-Burch et al., 2012), those from training an SVR on these features with and without CSLM features. Note that the official scores are often different from the results obtained with our SVR models because of differences in the learning algorithms. As shown in Table 5, we observed similar improvements with additional CSLM features over all of these feature sets.

System	#feats	Off.	SVR without CSLM	SVR with CSLM
SDL	15	0.61	0.6115	0.5993
UU	82	0.64	0.6513	0.6371
Loria	49	0.68	0.6978	0.6729
UEdin	56	0.68	0.6879	0.6724
TCD	43	0.68	0.6972	0.6715
WL-SH	147	0.69	0.6791	0.6678
UPC	57	0.84	0.8419	0.8310
DCU	308	0.75	0.6825	0.6812
PRHLT	497	0.70	0.6699	0.6649

Table 5: MAE score on official WMT12 feature sets using SVR with and without CSLM features.

6 Conclusions

We proposed novel features for machine translation quality estimation obtained using a deep continuous space language models. The proposed features led to significant improvements over standard feature sets for a variety of datasets, outperforming the state-of-art on two official WMT QE tasks. These results showed that different optimisation objectives and language pairs can benefit from the proposed features. The proposed features have been shown to also perform well on QE within a spoken language translation task.

Both source and target CSLM features improve prediction quality, either when used separately or in combination. They proved complementary when used together with other feature sets and produce comparable results to high performing baseline features when used alone for prediction. Finally, results comparing all official WMT12 QE feature sets showed significant improvements in the predictions when CSLM features were added to those submitted by participating teams. These findings provide evidence that the proposed features bring valuable information into prediction models, despite their simplicity and the fact that they require only monolingual data as resource, which is available in abundance for many languages.

As future work, it would be interesting to explore various distributed word representations for quality estimation and joint models that look at both the source and the target sentences simultaneously.

Acknowledgements

This work was supported by the QT21 (H2020 No. 645452), Cracker (H2020 No. 645357) and DARPA Bolt projects.

[6] We compare results in terms of MAE scores only.

References

Ebru Arisoy, Tara N. Sainath, Brian Kingsbury, and Bhuvana Ramabhadran. 2012. Deep neural network language models. In *NAACL-HLT 2012 Workshop: Will We Ever Really Replace the N-gram Model? On the Future of Language Modeling for HLT*, pages 20–28, Montreal, Canada.

Rafael E Banchs, Luis F D'Haro, and Haizhou Li. 2015. Adequacy–fluency metrics: Evaluating mt in the continuous space model framework. *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, 23(3):472–482.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, volume 29, pages 65–72.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *The Journal of Machine Learning Research*, 3:1137–1155.

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*, pages 1–44, Sofia, Bulgaria.

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

Wang Ling, Luis Marujo, Chris Dyer, Alan Black, and Isabel Trancoso. 2014. Crowdsourcing high-quality parallel data extraction from twitter. In *Ninth Workshop on Statistical Machine Translation*, WMT14, pages 426–436, Baltimore, USA.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. In *HLT-NAACL*, pages 746–751.

Robert C. Moore and William Lewis. 2010. Intelligent selection of language model training data. In *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort '10, pages 220–224, Stroudsburg, PA, USA. Association for Computational Linguistics.

Raymond W. N. Ng, Mortaza Doulaty, Rama Doddipatla, Oscar Saz, Madina Hasan, Thomas Hain, Wilker Aziz, Kashif Shaf, and Lucia Specia. 2014. The USFD spoken language translation system for IWSLT 2014. *Proc. IWSLT*, pages 86–91.

Holger Schwenk and Jean-Luc Gauvain. 2005. Training neural network language models on very large corpora. In *Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 201–208.

Holger Schwenk, Fethi Bougares, and Loic Barrault. 2014. Efficient training strategies for deep neural network language models. In *NIPS workshop on Deep Learning and Representation Learning*.

Holger Schwenk. 2007. Continuous space language models. *Computer Speech & Language*, 21(3):492–518.

Holger Schwenk. 2012. Continuous space translation models for phrase-based statistical machine translation. In *COLING (Posters)*, pages 1071–1080.

Kashif Shah, Eleftherios Avramidis, Ergun Biçicic, and Lucia Specia. 2013a. Quest - design, implementation and extensions of a framework for machine translation quality estimation. *The Prague Bulletin of Mathematical Linguistics*, 100:19–30.

Kashif Shah, Trevor Cohn, and Lucia Specia. 2013b. An investigation on the effectiveness of features for translation quality estimation. In *Machine Translation Summit*, volume 14, pages 167–174.

Kashif Shah, Trevor Cohn, and Lucia Specia. 2015. A bayesian non-linear method for feature selection in machine translation quality estimation. *Machine Translation*, 29(2):101–125.

Lucia Specia, Kashif Shah, José G. C. de Souza, and Trevor Cohn. 2013. QuEst - A translation quality estimation framework. In *51st Annual Meeting of the Association for Computational Linguistics: Demo Session*, pages 79–84, Sofia, Bulgaria.

Liling Tan, Carolina Scarton, Lucia Specia, and Josef van Genabith. 2015. Usaar-sheffield: Semantic textual similarity with deep regression and machine translation evaluation metrics. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, pages 85–89, Denver, Colorado.