Final Activity Report

Extending QuEst with Word-Level Quality Estimation

Lucia Specia and Gustavo Paetzold

University of Sheffield

1. Introduction

The goal of this project is to investigate the word-level features for machine translation (MT) quality estimation (QE), such that the end users can obtain the quality predictions for words in translated segments. The plan is to design and develop such features and integrate them into QuEst (http://www.quest.dcs.shef.ac.uk/) – an open source (BSD license) toolkit for MT QE developed by the University of Sheffield. It has two main modules: feature extraction and machine learning. At model learning time it extracts features, it takes examples of source and translation segments labelled for quality, extracts a number of features describing these, and builds a quality prediction model. At 'test time', given a source segment and its translation, it produces a score indicating its estimated quality.

A major shortfall in QuEst is that the features as well as prediction models are designed for sentence level only, which is too coarse-grained for certain users, such as post-editors, to identify the translation errors within the segment. One of the challenges is the complexity in designing the models at word-level, which require a careful investigation of efficient feature representations and model learning algorithms. This will require a major refactoring of the feature extractor component in QuEst, as well as integration (via wrappers to existing algorithms) of new learning models at word level. The development of such framework will allow users to get a fine-grained quality predictor for each word in the given segment. Another big advantage would be to push forward the state-of-the-art in word level QE, which has performed poorly as compared to sentence level prediction models (e.g., in WMT QE shared tasks) due to the unavailability of open source tools as a starting point. We have received a large number of requests to provide a similar functionality at word-level within QuEst.

The project started February 2015, and had the following schedule for milestones and deliverables:

- March-1-2015: Initial investigation of word-level features and learning algorithms

- May-1-2015: Refactored QuEst with feature extractors implemented; report (mid-project milestone)
- October-30-2015: Prediction models implemented
- December-31-2015: Benchmarking and paper submission
- January-31-2016: Final release of refactored QuEst source code for word-level prediction, with online-demo of prediction models at word-level.

In this report we describe all the activities developed throughout the project.

2. Activities Developed

From the beginning of February until the beginning of March, we have focused on investigating the potential of several features and learning techniques for Word-Level Quality Estimation (WQE). We have found that the features presented by (Luong, 2014), when used to train Conditional Random Fields models, would be the most viable combination. The results obtained by them in the WQE task of WMT 14 (Bojar et. al., 2014) show that their strategy is not only one of the most effective solutions for the task, but is also easily extensible to various languages, such as English, Spanish, French and German. Conditional Random Fields (CRF) also allow for not only numeral, but nominal features to be included in the new QuEst implementation, which would consequently further increase the reach of the final product. Given these observations, we have selected to include in QuEst the set of approximately 40 features introduced by (Luong, 2014), as well as a learning module which allows for such features to be used in the training of CRF models.

Once the features and learning strategies have been selected, we then began to restructure and refactor the latest version of QuEst in order for it to support the creation of WQE models. Initially, a new WQE module was introduced to QuEst. It is composed by several Java classes that both repurpose several components previously developed, and introduce new interfaces, utilities and templates that greatly facilitate the creation of new modules for QuEst in the future.

A total of 40 features for Word-Level Quality Estimation have been already included in the latest version of QuEst. They explore 8 distinct kinds of information from a given translation:

- **Target context:** These are features that explore the context of the target word we want to predict the quality for. Given a word *t* in position *i* of a given sentence *S*, we extract as features *t*, i.e. the word itself, as well as all bigrams and trigrams of which *t* is part of in *S*.

- **Alignment context:** These features explore the word alignment between source and target sentences. They require the 1-to-N alignment between source and target sentences to be provided. Given a word *t* in position *i* of a target sentence and a word *S* aligned to it in position *j* of a source sentence, the features are: the aligned word *s* itself, as well as all bigrams and trigrams combining word *t* and source words in positions *j-2*, *j-1*, *j+1* and *j+2*.

- **Lexical:** These features explore POS information on the source and target words. Given the POS tag *Pt* of word *t* in position *i* of a target sentence and the POS tag *Ps* of word *s* aligned to it in position *j* of a source sentence, we extract: the POS tags *Pt* and *Ps* themselves, and all bigrams and trigrams of target POS tags of which *Pt* is part of.

- **LM:** These features are related to the backoff behavior of a word's context with respect to an LM (Raybaud et. al., 2011). Two features are extracted: lexical backoff behavior and syntactic backoff behavior, which require LMs of the target language trained over sequences of words and POS tags. Given a word *t* in position *i* of a target sentence, the lexical backoff behavior is calculated as:

$$f(t_i) = \begin{cases} 7 & \text{if } t_{i-2}, t_{i-1}, t_i \text{ exists} \\ 6 & \text{if } t_{i-2}, t_{i-1} \text{ and } t_{i-1}, t_i \text{ exist} \\ 5 & \text{if only } t_{i-1}, t_i \text{ exists} \\ 4 & \text{if } t_{i-2}, t_{i-1} \text{ and } t_i \text{ exist} \\ 3 & \text{if } t_{i-1} \text{ and } t_i \text{ exist} \\ 2 & \text{if } t_i \text{ exists} \\ 1 & \text{if } t_i \text{ is out of the vocabulary} \end{cases}$$

The syntactic backoff behavior is calculated in an analogous fashion: instead of verifying for the existence of n-grams of words in the LM of word sequences, it verifies for the existence of n-grams of POS tags in the POS-tagged LM. The POS tags of target sentence are produced by the Stanford Parser which is integrated in QuEst.

- **Syntactic:** These features explore the syntax of a target sentence. QuEst provides one syntactic feature that proved very promising in previous work: the Null Link (Xiong et. al., 2010). It is a binary feature that receives value 1 if a given word *t* in a target sentence has at least one dependency link with another word, and 0 otherwise. The Stanford Parser is used for dependency parsing.

- **Semantic:** These features explore the polysemy of target and source words, i.e. the number of senses existing as entries in a WordNet for a given target word *t* or a source word *s*. We employ the Universal WordNet, which provides access to WordNets of various languages.

- **Pseudo-reference:** This binary feature explores the similarity between the target sentence and a translation for the source sentence produced by another MT system. The feature is 1 if the given word *t* in position *i* of a target sentence *S* is also present in a pseudo-reference translation *R*.

- **Translational:** Explore the intricacies of a translation model between the source and target languages. These features estimate the number of translations a word could have given a certain probability threshold: if the translation model suggests that the source word *s* has a translation probability to a target word *t* above the threshold, then the count is increased. This process is repeated to all words in the target language vocabulary.

The user can calculate any or all of such features for a given WQE dataset by editing the same feature configuration file used for Sentence-Level Quality Estimation in QuEst. For the WQE learning module, which allows for one to use the feature values calculated to produce WQE models, we have decided to create an interface for CRFSuite, a system developed in C++ that allows for WQE models to be trained with various distinct learning algorithms and settings. We have chosen CRFSuite over other tools due to its compatibility with both nominal and numeric features.

The learning module has been implemented in Python, and allows for the user to configure various aspects of CRFSuite, such as:

- Learning algorithm
- Parameter maximization strategy
- Cut-off threshold for occurrence frequency of a feature
- The coefficient for L1 regularization
- The coefficient for L2 regularization
- The number of limited memories
- The maximum number of iterations
- The epsilon parameter
- The threshold for the stopping criterion
- Calibration rates

To configure CRFSuite through the learning module, the user must provide a simple configuration file of which the template is included in QuEst.

3. Experiments

In order to evaluate the efficiency of QuEst's WQE module, a performance experiment was conducted. The module's performance was evaluated on the WMT14 WQE task with respect to its English-Spanish, Spanish-English, English-German and German-English datasets. The system was evaluated in three distinct sub-tasks:

- **Binary:** A Good/Bad label, where Bad indicates the need for editing the token. The evaluation metric for this task is the F-1 of instances with Bad labels.

- **Level 1:** A Good/Accuracy/Fluency label, specifying the coarser level categories of errors for each token, or Good for tokens with no error. The evaluation metric is the average F-1 of all but the Good class

- **Multi-Class:** One of 16 labels specifying the error type for the token (mistranslation, missing word, etc.). The evaluation metric is the average F-1 of all but the Good class.

All of the previously described features were calculated for each dataset. The features were then used by CRFSuite in order for WQE models to be trained. For all datasets except German-English, we use the Adaptive Regularization of Weight Vector learning algorithm of CRFSuite. For the German-English dataset, we use the Passive Aggressive learning algorithm, which tends to work better for this language pair. We compare the performance of QuEst with the ones of all systems submitted to the WQE task of WMT14. The results obtained are shown in Tables 1, 2, 3 and 4.

System	Binary	Level 1	Multiclass
QuEst	0.502	0.392	**0.227**
Baseline	**0.525**	**0.404**	0.222
LIG/BL	0.441	0.317	0.204
LIG/FS	0.444	0.317	0.204
FBK-1	0.487	0.372	0.170
FBK-2	0.426	0.385	0.230
LIMSI	0.473	-	-
RTM-1	0.350	0.299	0.268
RTM-2	0.328	0.266	0.032

Table 1 - F-1 scores for the WMT14 English-Spanish task

System	Binary	Level 1	Multiclass
QuEst	**0.386**	**0.267**	**0.161**
Baseline	0.299	0.151	0.019

RTM-1	0.269	0.219	0.087
RTM-2	0.291	0.239	0.081

Table 2 - F-1 scores for the WMT14 Spanish-English task

System	Binary	Level 1	Multiclass
QuEst	**0.507**	**0.287**	**0.161**
Baseline	0.445	0.117	0.086
RTM-1	0.452	0.211	0.150
RTM-2	0.369	0.219	0.124

Table 3 - F-1 scores for the WMT14 English-German task

System	Binary	Level 1	Multiclass
QuEst	**0.401**	**0.230**	**0.079**
Baseline	0.365	0.149	0.069
RTM-1	0.261	0.082	0.023
RTM-2	0.229	0.085	0.030

Table 4 - F-1 scores for the WMT14 German-English task

The results highlight the potential of the new WQE module, and show that it was able to outperform all participating systems in WMT14 except for the English-Spanish baseline in the Binary and Level 1 tasks.

We have also evaluated how well the WQE module of QuEst performs in the WMT15 Word-Level Quality Estimation task. The language pair of the task is English-Spanish. The training and test sets contain 11,271 and 1,817 instances, respectively. The task's results are showcased in Table 5.

System	Weighted F1 - All	F1 - Bad	F1 - Good
UAlacant/OnLine	71.47	43.12	78.07
HDCL/QUETCHPLUS	72.56	43.05	79.42
UAlacant/OnLine-SBI	69.54	41.51	76.06
SAU-KERC/CRF	77.44	39.11	86.36
SAU-KERC-SLG/CRF	77.4	38.91	86.35
SHEF2/W2V-BI-2000	65.37	38.43	71.63
SHEF2/W2V-BI-	65.27	38.40	71.52

SIM			
Word-Level QuEst	**62.07**	**38.36**	**67.58**
UGENT-LT3/SCATE	74.28	36.72	83.02
DCU-SHEF/ 2000	67.33	36.60	74.49
HDCL/QUETCH	75.26	35.27	84.56
DCU-SHEF/ 5000	75.09	34.53	84.53
UGENT-LT3/MBL	74.17	30.56	84.32
DCU/s5-RTM-GLMd	76.00	23.91	8812
DCU/s4-RTM-GLMd	75.88	22.69	88.26
Baseline	75.31	16.78	88.93

Table 5 - F-1 scores for the WMT15 English-Spanish task

It is noticeable that, even though QuEst was not able to achieve the best scores among all systems, it still offers very competitive performance, especially considering how easy it is for one to both train and test models using its tools and resources.

4. Budget

The budget planned for the project development has been spent as planned with the research intern (400 hours, at the student rate of € 15/hour = € 6,000) The remaining budget will be use used to support the participation of the research intern in the EAMT2016 conference to demonstrate the software and research that has been done with it.

5. Conclusion

In this report we have presented the progress made in the project “Extending QuEst with Word-Level Quality Estimation”. We have decided to add a new module to QuEst, which allows for the user to calculate several both numerical and nominal features for WQE. As of now, QuEst supports a total of 40 features. The WQE model learning module has not yet been implemented, but it will consist on an interface to the CRFSuite tool, which allows for one to train Conditional Random Field models. We have chosen to use Conditional

Random Fields for QuEst's WQE module because of two reasons: it has shown to be a promising strategy in the WQE task of WMT14, and it supports both numerical and nominal features.

We have also conducted a benchmarking with the current version of QuEst's WQE module over two distinct shared tasks. All 40 supported features were used, and models were trained with the Python learning module, which offers an interface to CRFSuite. The results obtained reveal that QuEst can outperform all approaches submitted to WMT14. Our participation in the Word-Level Quality Estimation task of WMT15 has also shown that word-level QuEst offers performance scores competitive to those of with much more elaborate solutions for the task.

In order to download, use and contribute to word-level QuEst, users can visit the Github page at https://github.com/ghpaetzold/questplusplus. There, one will find, along with the code, easy-to-follow tutorials on how to exploit all the tool's utilities, as well as a detailed manual on all the configuration options available.

References

Ngoc Quang Luong. Word Confidence Estimation for Statistical Machine Translation. Ph.D. Thesis. 2014.

Sylvain Raybaud, David Langlois, and Kamel Smaïli. "this sentence is wrong." detecting errors in machine-translated sentences. Machine Translation, 25(1):1–34, 2011.

Deyi Xiong, Min Zhang, and Haizhou Li. Error detection for statistical machine translation using linguistic features. In Proceedings of the 48th Association for Computational Linguistics, pages 604–611. 2010.

Bojar, Ondrej, et al. "Findings of the 2014 workshop on statistical machine translation." Proceedings of the Ninth Workshop on Statistical Machine Translation. Association for Computational Linguistics Baltimore, MD, USA, 2014.