Non-Linear Models for Confidence Estimation

Yong Zhuang*
Zhejiang University
866 Yuhangtang Road
Hangzhou, China
yong.zhuang22@gmail.com

Guillaume Wisniewski and **François Yvon**
Univ. Paris Sud and LIMSI–CNRS
rue John von Neumann
91403 Orsay CEDEX, France
{firstname.lastname}@limsi.fr

Abstract

This paper describes our work with the data distributed for the WMT'12 Confidence Estimation shared task. Our contribution is twofold: *i)* we first present an analysis of the data which highlights the difficulty of the task and motivates our approach; *ii)* we show that using non-linear models, namely random forests, with a simple and limited feature set, succeeds in modeling the complex decisions required to assess translation quality and achieves results that are on a par with the second best results of the shared task.

1 Introduction

Confidence estimation is the task of predicting the quality of a system prediction without knowledge of the expected output. It is an important step in many Natural Language Processing applications (Gandrabur et al., 2006). In Machine Translation (MT), this task has recently gained interest (Blatz et al., 2004; Specia et al., 2010b; Soricut and Echihabi, 2010; Bach et al., 2011). Indeed, professional translators are more and more requested to post-edit the outputs of a MT system rather than to produce a translation from scratch. Knowing in advance the segments they should focus on would be very helpful (Specia et al., 2010a). Confidence estimation is also of great interest for developers of MT system, as it provides them with a way to analyze the systems output and to better understand the main causes of errors.

Even if several studies have tackled the problem of confidence estimation in machine translation, until now, very few datasets were publicly available and comparing the proposed methods was difficult, if not impossible. To address this issue, WMT'12 organizers proposed a shared task aiming at predict the quality of a translation and provided the associated datasets, baselines and metrics.

This paper describes our work with the data of the WMT'12 Confidence Estimation shared task. Our contribution is twofold: *i)* we first present an analysis of the provided data that will stress the difficulty of the task and motivate the choice of our approach; *ii)* we show how using non-linear models, namely random forests, with a simple and limited features set succeed in modeling the complex decisions require to assess translation quality and achieve the second best results of the shared task.

The rest of this paper is organized as follows: Section 2 summarizes our analysis of the data; in Section 3, we describe our learning method; our main results are finally reported in Section 4.

2 Data Analysis

In this section, we quickly analyze the data distributed in the context of the WMT'12 Confidence Estimation Shared Task in order to evaluate the difficulty of the task and to find out what predictors shall be used. We will first describe the datasets, then the features usually considered in confidence estimation tasks and finally summarize our analyses.

2.1 Datasets

The datasets used in our experiments were released for the WMT'12 Quality Estimation Task. All the data provided in this shared task are based on the test set of WMT'09 and WMT'10 translation tasks.

The training set is made of $1,832$ English sentences and their Spanish translations as computed by a standard Moses system. Each sentence pair is accompanied by an estimate of its translation quality. This score is the average of ordinal grades assigned by three human evaluators. The human grades are in the range 1 to 5, the latter standing for a very good translation that hardly requires post-editing, while the former stands for a bad translation that does

*This work was conducted during an internship at LIMSI–CNRS

Proceedings of the 7th Workshop on Statistical Machine Translation, pages 157–162,
Montréal, Canada, June 7-8, 2012.

not deserve to be edited, meaning that the machine output useless and that translation should better be produced from scratch. The test contains 422 sentence pairs, the quality of which has to be predicted.

The training set also contains additional material, namely two references (the reference originally given by WMT and a human post-edited one), which will allow us to better interpret our results. No references were provided for the test set.

2.2 Features

Several works have studied the problem of confidence estimation (Blatz et al., 2004; Specia et al., 2010b) or related problems such as predicting readability (Kanungo and Orr, 2009) or developing automated essay scoring systems (Burstein et al., 1998). They all use the same basic features:

IBM 1 score measures the quality of the "association" of the source and the target sentence using bag-of-word translation models;

Language model score accounts for the "fluency", "grammaticality" and "plausibility" of a target sentence;

Simple surface features like the sentence length, the number of out-of-vocabulary words or words that are not aligned. These features are used to account for the difficulty of the translation task.

More elaborated features, derived, for instance, from parse trees or dependencies analysis have also been used in past studies. However they are far more expensive to compute and rely on the existence of external resources, which may be problematic for some languages. That is why we only considered a restricted number of basic features in this work[1]. Another reason for considering such a small set of features is the relatively small size of the training set: in our preliminary experiments, considering more features, especially lexicalized features that would be of great interest for failure analysis, always resulted in overfitting.

2.3 Data Analysis

The distribution of the human scores on the training set is displayed in Figure 1. Surprisingly enough, the baseline translation system used to generate the data seems to be pretty good: 73% of the sentences have a score higher than 3 on a 1 to 5 scale. It also appears that most scores are very close: more than half of them are located around the mean. As a consequence, it seems that distinguishing between them will require to model subtle nuances.

[1]The complete list of features is given in Appendix A.

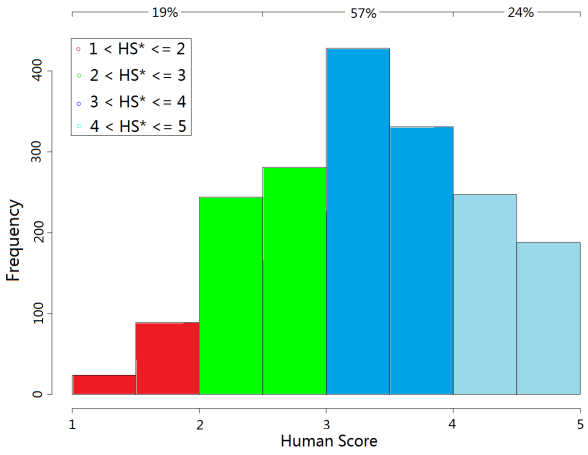

Figure 1: Distribution of the human scores on the train set. (HS* stands for Human Scores)

Figure 2 plots the distribution of quality scores as a function of the Spanish-to-English IBM 1 score and of the probability of the target sentence. These two scores were computed with the same models that were used to train the MT systems that have generated the training data. It appears that even if the examples are clustered by their quality, these clusters overlap and the frontiers between them are fuzzy and complex. Similar observations were made for others features.

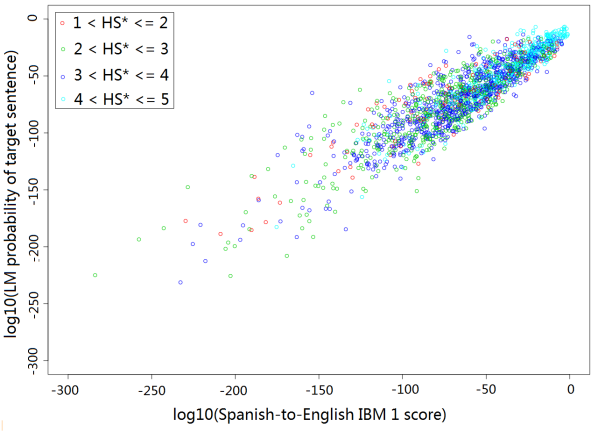

Figure 2: Quality scores as a function of the Spanish-to-English IBM 1 score and of the probability of the target sentence (HS* stands for Human Scores)

These observations prove that a predictor of the translation quality has to capture complex interaction patterns in the training data. Standard results from machine learning show that such structures can be described either by a linear model using a large number of features or by a non-linear model using a

(potentially) smaller set of features. As only a small number of training examples is available, we decided to focus on non-linear models in this work.

3 Inferring quality scores

Predicting the quality scores can naturally be cast as a standard regression task, as the reference scores used in the evaluation are numerical (real) values. Regression is the approach adopted in most works on confidence estimation for MT (Albrecht and Hwa, 2007; Specia et al., 2010b). A simpler way to tackle the problem would be to recast it as binary classification task aiming at distinguishing "good" translations from "bad" ones (Blatz et al., 2004; Quirk, 2004). It is also possible, as shown by (Soricut and Echihabi, 2010), to use ranking approaches. However, because the shared task is evaluated by comparing the actual value of the predictions with the human scores, using these last two frameworks is not possible.

In our experiments, following the observations reported in the previous section, we use two well-known non-linear regression methods: polynomial regression and random forests. We also consider linear regression as a baseline. We will now quickly describe these three methods.

Linear regression (Hastie et al., 2003) is a simple model in which the prediction is defined by a linear combination of the feature vector $\mathbf{x}$: $\hat{y} = \beta_0 + x^\top \boldsymbol{\beta}$, where β_0 and $\boldsymbol{\beta}$ are the parameters to estimate. These parameters are usually learned by minimizing the sum of squared deviations on the training set, which is an easy optimization problem with a close-form solution.

Polynomial regression (Hastie et al., 2003) is a straightforward generalization of linear regression in which the relationship between the features and the label is modeled as a n-th order polynomial. By carefully extending the feature vector, the model can be reduced to a linear regression model and trained in the same way.

Random forest regressor (Breiman, 2001) is an ensemble method that learns many regression trees and predicts an aggregation of their result. In contrast with standard decision tree, in which each node is split using the best split among all features, in a random forest the split is chosen randomly. In spite of this simple and counter-intuitive learning strategy, random forests have proven to be very good "out-of-the-box" learners and have achieved state-of-the-art performance in many tasks, demonstrating both their robustness to overfitting and their ability to take into account complex interactions between features.

In our experiments, we use the implementation provided by scikit-learn (Pedregosa et al., 2011). Hyper-parameters of the random forest (the number of trees and the stopping criterion) were chosen by 10-fold cross-validation.

4 Experimental Setting

4.1 Features

In all our experiments, we considered a simple description of the translation hypotheses relying on 31 features. The complete list of features is given in Appendix A. All these features have already been used in works related to ours and are simple features that can be easily computed using only a limited number of external resources.

A key finding in our preliminary experiments is the need to re-scale the features by dividing their value by the length of the corresponding sentence (e.g. the language model score of a source sentence will be divided by its length of the source sentence, and the one of a target sentence will be done by its length of the target sentence). This rescaling makes features that depend on the sentence length (like the LM score) comparable and results in a large improvement of the performance of the associated feature.

4.2 Metrics

The two metrics used to evaluate prediction performance are the standard metrics for regression: *Mean Absolute Error* (MAE) and *Root Mean Squared Error* (RMSE) defined by:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i|$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}$$

where n is the number of examples, y_i and $\hat{y}_i$ the true label and predicted label of the i^{th} example. MAE can be understood as the averaged error made in predicting the quality of a translation. As it is easy to interpret, we will use it to analyze our results. RMSE scores are reported to facilitate comparison with other submissions to the shared task.

All the reported scores have been computed using the tools provided by the Quality Estimation task organizers[2].

[2] `https://github.com/lspecia/QualityEstimation`

4.3 Results

Table 1 details the results achieved by the different methods introduced in the previous section. All of them achieve similar performances: their MAE is between 0.64 and 0.66, which is a pretty good result as the best reported MAE in the shared task is 0.61. Our best model is the second-best when submissions are ranked according to their MAE.

Even if their results are very close (significance of the score differences will be investigated in the following subsection), all non-linear models outperform a simple linear regression, which corroborates the observations made in Section 2.

For the polynomial regression, we tried different polynomial orders in order to achieve an optimal setting. Even if this method achieves the best results when the model is selected on the *test* set, it is not usable in practice: when we tried to select the polynomial degree by cross-validation, the regressors systematically overfitted due to the reduction of the number of examples. That is why random forests, which do not suffer from overfitting and can learn good predictor even when features outnumber examples, is our method of choice.

4.4 Interpretation

To get a better understanding of the task difficulty and to make interpretation of the error rate easier, we train another regressor using an "oracle" feature: the hTER score. It is clear that this feature can only be computed on the training set and that considering it does not make much sense in a "real-life" scenario. However, this feature is supposed to be highly relevant to the quality prediction task and should therefore result in a "large" reduction of the error rates. Quantifying what "large" means in this context will allow us to analyze the results presented in Table 1.

Training a random forest with this additional feature on $1,400$ examples of the train set chosen randomly reduces the MAE evaluated on the 432 remaining examples by 0.10 and the RMSE by 0.12. This small reduction stresses how difficult the task is. Comparatively, the 0.02 reduction achieved by replacing a linear model with a non-linear model should therefore be considered noteworthy. Further investigations are required to find out whether the difficulty of the task results from the way human scores are collected (low inter-annotators agreement, bias in the gathering of the collection, ...) or from the impossibility to solve the task using only surface features.

Another important question in the analysis of our results concerns the usability of our approach: an error of 0.6 seems large on a 1 to 5 scale and may question the interest of our approach. To allow a fine-grained analysis, we report the correlation between the predicted score and the human score (Figure 3) and the distribution of the absolute error (Figure 4). These figures show that the actual error is often quite small: for more than 45% of the examples, the error is smaller than 0.5 and for 23% it is smaller than 0.2. Figure 3 also shows that the correlation between our predictions and the true labels is "substantial" according to the established guidelines of (Landis and Koch, 1977) (the Pearson correlation coefficient is greater than 0.6). The difference between the mean of the two distributions is however quite large. Centering the predictions on the mean of the true label may improves the MAE. This observation also suggests that we should try to design evaluation metrics that do not rely on the actual predicted values.

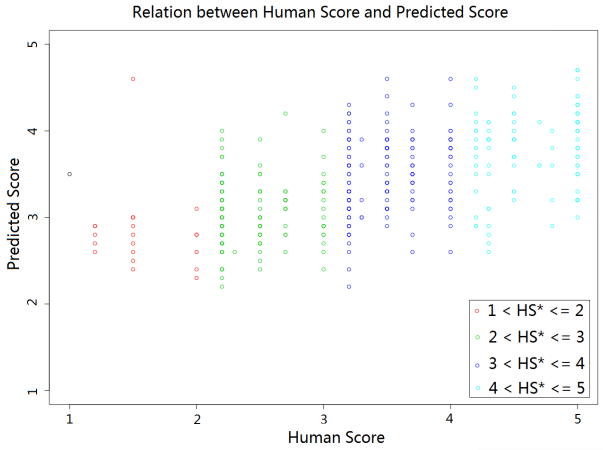

Figure 3: Correlation between our predictions and the true label (HS* stands for Human Scores)

5 Conclusion

In this work, we have presented, a simple, yet efficient, method to predict the quality of a translation. Using simple features and a non-linear model, our approach has achieved results close to the best submission to the Confidence Estimation shared task, which supports the results of our analysis of the data. In our future work, we aim at considering more features, avoiding overfitting thanks to features selection methods.

Even if a fine-grained analysis of our results shows the interest and usefulness of our approach, more remains to be done to develop reliable confidence estimation methods. Our results also highlight the need to continue gathering high-quality resources to train and investigate confidence estimation systems: even when considering only very few features, our systems

Methods	parameters	Train MAE	Train RMSE	Test MAE	Test RMSE
linear regression	—	0.58	0.71	0.66	0.82
polynomial regression	n=2	0.55	0.68	0.64	0.79
	n=3	0.54	0.67	0.64	0.79
	n=4	0.54	0.67	0.65	0.85
random forest	cross-validated	0.39	0.46	0.64	0.80

Table 1: Prediction performance achieved by different regressors

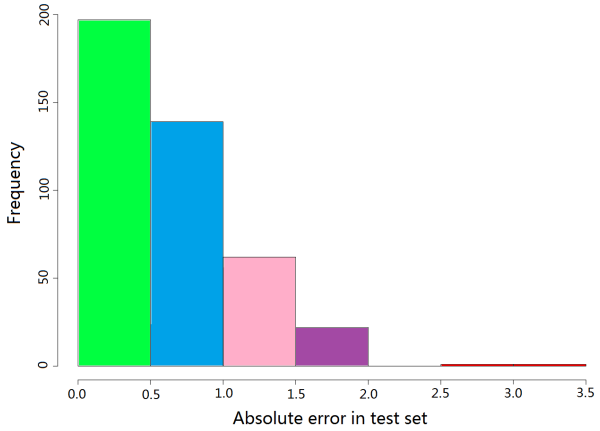

Figure 4: Distribution of the absolute error ($|y_i - \hat{y}_i|$) of our predictions

were prone to overfitting. Developing more elaborated systems will therefore only be possible if more training resource is available.

Acknowledgment

The authors would like to thank Nicolas Usunier for helpful discussions about ranking and regression using random forest. This work was partially funded by the French National Research Agency under project ANR-CONTINT-TRACE.

References

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

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*, HLT '11, pages 211–219, Stroudsburg, PA, USA. Association for Computational Linguistics.

John Blatz, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence estimation for machine translation. In *Proceedings of the 20th international conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA. Association for Computational Linguistics.

Leo Breiman. 2001. Random forests. *Mach. Learn.*, 45(1):5–32, October.

Jill Burstein, Karen Kukich, Susanne Wolff, Chi Lu, Martin Chodorow, Lisa Braden-Harder, and Mary Dee Harris. 1998. Automated scoring using a hybrid feature identification technique. In *Proceedings of the 17th international conference on Computational linguistics - Volume 1*, COLING '98, pages 206–210, Stroudsburg, PA, USA. Association for Computational Linguistics.

Simona Gandrabur, George Foster, and Guy Lapalme. 2006. Confidence estimation for nlp applications. *ACM Trans. Speech Lang. Process.*, 3(3):1–29, October.

T. Hastie, R. Tibshirani, and J. H. Friedman. 2003. *The Elements of Statistical Learning*. Springer, July.

Tapas Kanungo and David Orr. 2009. Predicting the readability of short web summaries. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, pages 202–211, New York, NY, USA. ACM.

R. J. Landis and G. G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python . *Journal of Machine Learning Research*, 12:2825–2830.

Chris Quirk. 2004. Training a sentence-level machine translation confidence metric. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, pages 825–828.

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, Nicola Cancedda, and Marc Dymetman. 2010a. A dataset for assessing machine translation evaluation metrics. In *7th Conference on International Language Resources and Evaluation (LREC-2010)*, pages 3375–3378, Valletta, Malta.

Lucia Specia, Dhwaj Raj, and Marco Turchi. 2010b. Machine translation evaluation versus quality estimation. *Machine Translation*, 24(1):39–50, March.

A Features List

Here is the whole list of the 31 features we used in our experiments († has been used in the baseline of the shared task organizer):

- † Number of tokens in the source sentence
- † Number of tokens in the target sentence
- † Average token length in source sentence
- English-Spanish IBM 1 scores
- Spanish-English IBM 1 scores
- English-Spanish IBM 1 scores divided by the length of source sentence
- English-Spanish IBM 1 scores divided by the length of target sentence
- Spanish-English IBM 1 scores divided by the length of source sentence
- Spanish-English IBM 1 scores divided by the length of target sentence
- Number of out-of-vocabulary in source sentence
- Number of out-of-vocabulary in target sentence
- Out-of-vocabulary rates in source sentence
- Out-of-vocabulary rates in target sentence
- $\log_{10}$(LM probability of source sentence)
- $\log_{10}$(LM probability of target sentence)
- $\log_{10}$(LM probability of source sentence) divided by the length of source sentence
- $\log_{10}$(LM probability of target sentence) divided by the length of target sentence
- Ratio of functions words in source sentence
- Ratio of functions words in target sentence
- † Number of occurrences of the target word within the target hypothesis (averaged for all words in the hypothesis - type/token ratio)
- † Average number of translations per source word in the sentence (as given by IBM 1 table thresholded so that prob$(t|s) > 0.2$)
- † Average number of translations per source word in the sentence (as given by IBM 1 table thresholded so that prob$(t|s) > 0.01$) weighted by the inverse frequency of each word in the source corpus
- † Percentage of unigrams in quartile 1 of frequency (lower frequency words) in a corpus of the source language (SMT training corpus)
- † Percentage of unigrams in quartile 4 of frequency (higher frequency words) in a corpus of the source sentence
- † Percentage of bigrams in quartile 1 of frequency of source words in a corpus of the source language
- † Percentage of bigrams in quartile 4 of frequency of source words in a corpus of the source language
- † Percentage of trigrams in quartile 1 of frequency of source words in a corpus of the source language
- † Percentage of trigrams in quartile 4 of frequency of source words in a corpus of the source language
- † Percentage of unigrams in the source sentence seen in a corpus (SMT training corpus)
- † Number of punctuation marks in the source sentence
- † Number of punctuation marks in the target sentence