Predicting Translation Performance with Referential Translation Machines

Ergun Biçici
orcid.org/0000-0002-2293-2031
bicici.github.com

Abstract

Referential translation machines achieve top performance in both bilingual and monolingual settings without accessing any task or domain specific information or resource. RTMs achieve the 3rd system results for German to English sentence-level prediction of translation quality and the 2nd system results according to root mean squared error. In addition to the new features about substring distances, punctuation tokens, character n-grams, and alignment crossings, and additional learning models, we average prediction scores from different models using weights based on their training performance for improved results.

1 Introduction

Quality estimation task (QET) in WMT17 (Bojar et al., 2017) (QET17) is about prediction of the quality of machine translation output at the sentence- (Task 1), word- (Task 2), and phrase-level (Task 3) in IT and pharmaceutical domains without using reference translations. Prediction of translation performance can help in estimating the effort required for correcting the translations during post-editing by human translators if needed. RTMs are capable to model different domains and tasks while achieving top performance in both monolingual (Biçici and Way, 2015) and bilingual settings (Biçici, 2016b). We develop RTM models for all of the three subtasks of QET17, which include English to German (en-de), and German to English (de-en) translation directions. Task 1 is about predicting HTER (human-targeted translation edit rate) scores (Snover et al., 2006), Task 2 is about binary classification of word-level quality, and Task 3 is about binary classification of phrase-level quality.

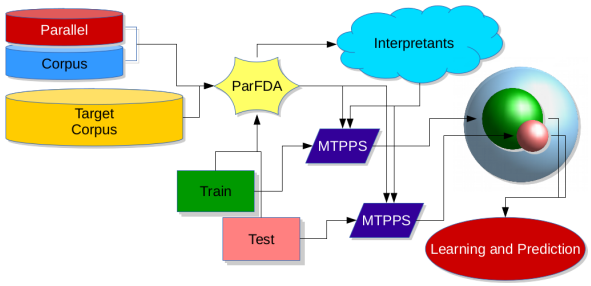

Figure 1: RTM depiction: ParFDA selects interpretants close to the training and test data using parallel corpus in bilingual settings and monolingual corpus in the target language or just the monolingual target corpus in monolingual settings; an MTPPS use interpretants and training data to generate training features and another use interpretants and test data to generate test features in the same feature space; learning and prediction takes place taking these features as input.

2 Referential Translation Machines

Referential translation machine (RTM) models are predict data translation between the instances in the training set and the test set. RTMs use interpretants, data close to the task instances, to derive features measuring the closeness of the test sentences to the training data, the difficulty of translating them, and to identify translation acts between any two data sets for building prediction models. RTMs are applicable in different domains and tasks and in both monolingual and bilingual settings. Figure 1 depicts RTMs and explains the model building process. RTMs use ParFDA (Biçici, 2016a) for instance selection and machine translation performance prediction system (MTPPS) (Biçici and Way, 2015) for generat-

Task		Model	DeltaAvg	r_P	r_S	RMSE	MAE	RAE	MAER	MRAER	Rank
Task 1	en-de	MIX 4	8.64	0.4544	0.4768	0.1707	0.1296	0.8483	0.7594	0.7962	9
		PLS GBR	8.22	0.4302	0.4518	0.1727	0.1311	0.8586	0.7769	0.8099	10
	de-en	MIX 4	8.94	0.6004	0.5704	0.1566	0.1085	0.7034	0.7201	0.6921	4
		TREE	9.18	0.5845	0.5729	0.158	0.1186	0.7685	0.9013	0.7627	5

Table 1: Task 1 test results of the top 2 individual RTM models. RTM becomes the 2nd system according to RMSE and 3rd system in de-en and 6th system in en-de. r_P is Pearson's correlation and r_S is Spearman's correlation.

			RTM Interpretants	
Task	Train	Test	Training	LM
Task 1, 2, 3 (en-de)	24000	2000	1.1M	17.6M
Task 1, 2, 3 (de-en)	26000	2000	1.1M	17.6M

Table 2: Number of instances used as interpretants by the RTM models.

ing features where the total number of features becomes 514, increasing depending on the order of n-grams used and we used up to 5-grams for translation features and 7-grams for language model (LM) at QET17.

We use ridge regression (RR), k-nearest neighors (KNN), support vector regression (SVR), AdaBoost (Freund and Schapire, 1997), and extremely randomized trees (TREE) (Geurts et al., 2006) as learning models in combination with feature selection (FS) (Guyon et al., 2002) and partial least squares (PLS) (Wold et al., 1984). We use `scikit-learn` [1] for most of these models. The following parameters are optimized: λ for RR, k for KNN, γ, C, and ϵ for SVR, minimum number of samples for leaf nodes and for splitting an internal node for TREE, the number of features for FS, and the number of dimensions for PLS. For AdaBoost, we do not optimize but use exponential loss and 500 estimators like we use also with the TREE model. We use grid search for SVR. Evaluation metrics we use are Pearson's correlation (r), mean absolute error (MAE), relative absolute error (RAE), MAER (mean absolute error relative), and MRAER (mean relative absolute error relative) (Biçici and Way, 2015). DeltaAvg (Callison-Burch et al., 2012) calculates the average quality difference between the top $n-1$ quartiles and the overall quality for the test set. Official evaluation metrics include r, MAE, and DeltaAvg.

We improved RTM models (Biçici, 2016b) with additional features:

- normalized Levenshtein distance between the source sentence and its translation and their longest common prefix, suffix, and substring (Tian et al., 2017) normalized by the minimum length of the compared sentences.
- number of tokens about punctuation in the source sentence and the translation (Kozlova et al., 2016) and the cosine between them.
- modified CHRF_3 (Popović, 2015) to compute character n-grams split by word boundary space with $n \in [3,7]$ whereas the F_1 (Biçici, 2011) we already use compute with word n-grams up to $n=5$.
- proportion of alignments that cross ($\nleftrightarrow$) the link (Sagemo and Stymne, 2016) of any other alignments:

$$\sqrt{\frac{0.5 \times |a \nleftrightarrow A|}{|A|}} \quad (1)$$

- word alignment correspondence features (Sagemo and Stymne, 2016).
- additional learning models including KNN, AdaBoost, and gradient boosting regressor (GBR) (Tian et al., 2017; Hastie et al., 2009).

We also use prediction averaging (Biçici, 2017), where the performance on the training set is used to obtain weighted average of the top k predictions, $\hat{y}$ with evaluation metrics indexed by $j \in J$:

$$\begin{aligned} \hat{\boldsymbol{y}}_{\mu_k} &= \frac{1}{k}\sum_{i=1}^{k}\hat{\boldsymbol{y}}_i && \text{MEAN} \\ \hat{\boldsymbol{y}}_{j,w_k^j} &= \frac{1}{\sum_{i=1}^{k}\frac{1}{w_{j,i}}}\sum_{i=1}^{k}\frac{1}{w_{j,i}}\hat{\boldsymbol{y}}_i \\ \hat{\boldsymbol{y}}_k &= \frac{1}{|J|}\sum_{j\in J}\hat{\boldsymbol{y}}_{j,w_k^j} && \text{MIX} \end{aligned} \quad (2)$$

MAER is used to select the predictions and weights are inverted to decrease error.

We use Global Linear Models (GLM) (Collins, 2002) with dynamic learning (GLMd) (Biçici,

[1] `http://scikit-learn.org/`

Model			splits	% error	weights
2017 word	en-de	GLMd	4	0.0773	$[0.5, 2]$
		GLMd	5	0.0668	
	de-en	GLMd	4	0.0468	
		GLMd	5	0.0469	
2017 phrase	en-de	GLMd	4	0.0068	
		GLMd	5	0.0059	
	de-en	GLMd	4	0.0129	
		GLMd	5	0.0125	
2016 word	en-de	GLMd	4	0.0688	
		GLMd	5	0.0757	
2016 phrase	en-de	GLMd	4	0.0051	
		GLMd	5	0.0051	

Table 3: RTM Task 2 training results where GLMd parallelized over 4 splits is referred as GLMd s4 and GLMd with 5 splits as GLMd s5.

Model			F_1 BAD	F_1 OK	wF_1
Word	en-de	GLMd s4	0.318	0.8844	0.2813
		GLMd s5	0.36	0.8778	0.3158
	de-en	GLMd s4	0.3363	0.9386	0.3157
		GLMd s5	0.3381	0.9395	0.3176
Phrase	en-de	GLMd s4	0.4043	0.8079	0.3283
		GLMd s5	0.4114	0.8079	0.3323
	de-en	GLMd s4	0.2472	0.9073	0.2242
		GLMd s5	0.3598	0.8884	0.3197

Table 4: RTM Task 2 results on the test set after the challenge. wF_1 is average weighted F_1 score.

2016b) for word- and phrase-level translation performance prediction. GLMd uses weights in a range $[a, b]$ to update the learning rate dynamically according to the error rate.

3 Results

Table 2 lists the number of sentences in the training and test sets for each task and the number of instances used as interpretants in the RTM models (M for million). We tokenize and truecase all of the corpora using Moses's (Koehn et al., 2007) processing tools. [2] LMs are built using KENLM (Heafield et al., 2013).

3.1 QET 2017 Results

The results on the Task 1 test set are listed in Table 1. [3] For Task 2 and Task 3, we list the results we obtain after the challenge for coherent presentation on the training sets in Table 3 and on the test set in Table 4. The results we obtained in the challenge are similar. Ranks for Task 1 are out of 14 submissions and 9 systems. Top RTM models that competed in Task 1 were MIX 4, which combines top 4 predictions, PLS GBR, and TREE. RTM becomes the 2nd system according to RMSE and 3rd system in de-en and 6th system in en-de.

3.2 Recomputing QET 2016 Results

QET17 also compares results on QET16 test sets. QET16 test set domain was different than the domain of QET17, overlapping on the IT domain. We use the RTM models built for QET17 to obtain results on the QET16 test sets, which is categorized as transductive transfer learning. [4] Transfer learning attempt to re-use and transfer knowledge from models developed in different domains or for different tasks such as using models developed for handwritten digit recognition for handwritten character recognition (Guyon et al., 2012). The results are in Table 5 for Task 1, which does not show improvement, and in Table 7, which show improvements with RTM models built for QET17.

3.3 Comparison with Previous Results

We compare the difficulty of tasks according to MRAER levels achieved. In Table 6, we list the RTM test results when predicting sentence-level HTER in 2013–2017. Compared with QET16, we observe improvements in MRAER and both MAE and RAE are improved when QET17 is compared with others.

4 Conclusion

Referential translation machines achieve top performance in automatic, accurate, and language independent prediction of translation performance and achieve to become the 2nd system according to RMSE when predicting the translation performance from German to English. RTMs pioneer a language independent approach for predicting translation performance and remove the need to access any task or domain specific information or resource.

[2] https://github.com/moses-smt/mosesdecoder/tree/master/scripts

[3] We calculate r_S using scipy.stats.

[4] www.youtube.com/watch?v=9ChVn3xVNDI; we use the RTM models for the same task in different domains.

	Model	DeltaAvg	r	MAE	RMSE	RAE	MAER	MRAER
2017	ST TREE	5.14	0.2052	0.1456	0.1875	0.9634	0.8844	0.8666
	PLS GBR	3.71	0.1875	0.1474	0.1914	0.9755	0.8706	0.8966
2016	SVR	6.38	0.3581	0.1359	0.1806	0.8992	0.7509	0.8567
	FS SVR	6.66	0.3764	0.1346	0.1781	0.8905	0.7537	0.8388

Table 5: QET16 Task 1 results are not improved with QET17 Task 1 RTM models.

Task	Translation	Model	r	MAE	RAE	MAER	MRAER
QET17 Task 1 HTER	en-de	MIX 4	0.4544	0.1296	0.8483	0.7594	0.7962
	de-en	MIX 4	0.6004	0.1085	0.7034	0.7201	0.6921
QET16 Task 1 HTER	en-de	FS SVR	0.3764	0.1346	0.8905	0.7537	0.8388
QET15 Task 1 HTER	en-es	FS+PLS SVR	0.349	0.1335	0.903	0.8284	0.8353
QET14 Task 1.2 HTER	en-es	SVR	0.5499	0.134	0.8532	0.7727	0.8758
QET13 Task 1.1 HTER	en-es	PLS-SVR	0.5596	0.1326	0.8849	2.3738	1.6428

Table 6: Test performance of the top RTM results when predicting sentence-level HTER in 2013–2017.

		Model	wF_1	F_1 OK	F_1 BAD
2017	Word	GLMd s4	0.2857	0.8775	0.3256
		GLMd s5	0.3053	0.8653	0.3528
	Phrase	GLMd s4	0.3421	**0.8192**	0.4176
		GLMd s5	0.3504	**0.817**	0.4289
2016	Word	GLMd s4	0.2725	**0.8884**	0.3068
		GLMd s5	0.3081	0.8820	0.3494
	Phrase	GLMd s4	0.3070	**0.8145**	0.3770
		GLMd s5	0.3274	**0.8016**	0.4084

Table 7: QET16 Task 2 and Task 2p results show improvement.

References

Ergun Biçici. 2011. *The Regression Model of Machine Translation*. Ph.D. thesis, Koç University. Supervisor: Deniz Yuret.

Ergun Biçici. 2016a. ParFDA for instance selection for statistical machine translation. In *Proc. of the First Conference on Statistical Machine Translation (WMT16)*. Association for Computational Linguistics, Berlin, Germany. http://aclanthology.info/papers/parfda-for-instance-selection-for-statistical-machine-translation.

Ergun Biçici. 2016b. Referential translation machines for predicting translation performance. In *Proc. of the First Conference on Statistical Machine Translation (WMT16)*. Association for Computational Linguistics, Berlin, Germany. http://aclanthology.info/papers/referential-translation-machines-for-predicting-translation-performance.

Ergun Biçici. 2017. RTM at SemEval-2017 task 1: Referential translation machines for predicting semantic similarity. In *Proc. of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Association for Computational Linguistics, Vancouver, Canada, pages 194–198. http://nlp.arizona.edu/SemEval-2017/pdf/SemEval030.pdf.

Ergun Biçici and Andy Way. 2015. Referential translation machines for predicting semantic similarity. *Language Resources and Evaluation* pages 1–27. https://doi.org/10.1007/s10579-015-9322-7.

Ondrej Bojar, Christian Buck, Rajen Chatterjee, Christian Federmann, Barry Haddow, Matthias Huck, Jimeno Antonio Yepes, Julia Kreutzer, Varvara Logacheva, Aurelie Neveol, Mariana Neves, Philipp Koehn, Christof Monz, Matteo Negri, Matt Post, Stefan Riezler, Artem Sokolov, Lucia Specia, Karin Verspoor, and Marco Turchi. 2017. Proc. of the second conference on Machine Translation. In *Proc. of the Second Conference on Machine Translation*. Association for Computational Linguistics, Copenhagen, Denmark.

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 *Proc. of the Seventh Workshop on Statistical Machine Translation*. Montréal, Canada, pages 10–51.

Michael Collins. 2002. Discriminative training methods for hidden markov models: theory and experiments with perceptron algorithms. In *Proc. of the ACL-02 conference on Empirical methods in natural language processing - Volume 10*. Stroudsburg, PA, USA, EMNLP '02, pages 1–8. https://doi.org/10.3115/1118693.1118694.

Yoav Freund and Robert E Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences* 55(1):119–139. https://doi.org/10.1006/jcss.1997.1504.

Pierre Geurts, Damien Ernst, and Louis Wehenkel. 2006. Extremely randomized trees. *Machine Learning* 63(1):3–42.

Isabelle Guyon, Gideon Dror, Vincent Lemaire, Graham W. Taylor, and Daniel L. Silver, editors. 2012. *Unsupervised and Transfer Learning - Workshop held at ICML 2011, Bellevue, Washington, USA, July 2, 2011*, volume 27 of *JMLR Proceedings*. JMLR.org. http://clopinet.com/ul.

Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. 2002. Gene selection for cancer classification using support vector machines. *Machine Learning* 46(1-3):389–422. https://doi.org/10.1023/A:1012487302797.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer-Verlag, 2nd edition.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified Kneser-Ney language model estimation. In *Proc. of the 51st Annual Meeting of the Association for Computational Linguistics*. Sofia, Bulgaria, pages 690–696.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proc. of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proc. of the Demo and Poster Sessions*. Association for Computational Linguistics, pages 177–180. aclweb.org/anthology/P07-2045.

Anna Kozlova, Mariya Shmatova, and Anton Frolov. 2016. Ysda participation in the wmt'16 quality estimation shared task. In *Proceedings of the First Conference on Machine Translation*. Association for Computational Linguistics, Berlin, Germany, pages 793–799. http://www.aclweb.org/anthology/W/W16/W16-2385.

Maja Popović. 2015. chrf: character n-gram f-score for automatic mt evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Lisbon, Portugal, pages 392–395. http://aclweb.org/anthology/W15-3049.

Oscar Sagemo and Sara Stymne. 2016. The uu submission to the machine translation quality estimation task. In *Proceedings of the First Conference on Machine Translation*. Association for Computational Linguistics, Berlin, Germany, pages 825–830. http://www.aclweb.org/anthology/W/W16/W16-2390.

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

Junfeng Tian, Zhiheng Zhou, Man Lan, and Yuanbin Wu. 2017. Ecnu at semeval-2017 task 1: Leverage kernel-based traditional nlp features and neural networks to build a universal model for multilingual and cross-lingual semantic textual similarity. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Association for Computational Linguistics, Vancouver, Canada, pages 182–188. http://www.aclweb.org/anthology/S17-2028.

S. Wold, A. Ruhe, H. Wold, and W. J. III Dunn. 1984. The collinearity problem in linear regression. the partial least squares (pls) approach to generalized inverses. *SIAM Journal on Scientific and Statistical Computing* 5:735–743.