Implementation of Metrics for Automatic Evaluation of MT

Mariana Camarneiro, Matilde Pires, Rui Monteiro
R20170744, R20170783, R20170796

1 Introduction

During this project we participated in a simulated version of the WMT Metrics Shared Task and created a metric that correlates with human assessments of quality. The corpus is composed by the following language pairs: Russian to English (*ru-en*), German to English (*de-en*), Czech to English (*cs-en*), Chinese to English (*zh-en*), English to Chinese (*en-zh*) and English to Finish (*en-fi*).

Our main goal was to use existing Machine Translation (MT) metrics and to create our own models, using Python, in order to predict scores that correlate well with the existing quality assessments in the corpus, according to *Pearson's* and *Kendall's Tau* correlation coefficients.

In Section 2 we will describe our approaches in terms of preprocessing, used MT metrics, and other Machine Learning models used with different input features. Section 3 presents the results obtained in the whole process and a comparison between their performance. Section 4 presents the Conclusion.

2 Method/Approach

2.1 Preprocessing

Several decisions were taken to reach a logical preprocessing baseline. After testing, we established it would not make sense to remove the stop words, as it would remove the entire content in case the sentence was only composed by stop words and it would harm the results if it happened in the test set. Also, removing stop words might classify a bad translation as good, since it would not consider the stop words that might have a wrong translation.

Furthermore, we lowercase all the words since capitalization is not relevant when assessing a translation. The same thinking was applied to punctuation - even if the punctuation was not correctly placed, in most cases it does not affect the translation quality.

Considering this, we defined two preprocessing approaches, where both take all this into account, but the second uses *stemming* as well, with the English stemmer for the 4 corpora with English translation and a Finnish stemmer for the English to Finnish one. Therefore, "Preprocessing 1" consists on the lowercasing of sentences and removal of punctuation, and "Preprocessing 2" is similar to the previous one, but it also includes *stemming*.

Lastly, the English to Chinese corpus required special attention, but it is equivalent to the first approach of preprocessing. It consisted in the punctuation removal for the previously explained reasons and in the use of the *Jieba* package, used for Chinese text segmentation. This is needed because Chinese is written in particular characters, and it has unique aspects. With this package, we were able to apply the traditional splits between tokens, which is required by some MT metrics. Also, the tokenization was done in an accurate way, because *Jieba* takes into consideration the meaning of each character in each sentence [1].

2.2 MT Metrics

We started by implementing **simple and non-trainable metrics**. The baselines we chose to compare each pair of reference and translation were: distance metrics to measure sentence similarity (*Euclidean Distance, Manhattan Distance, Cosine Similarity and Jaccard Similarity*); Bi-Lingual Evaluation Understudy (*BLEU*); Recall-Oriented Understudy for Gisting Evaluation (*ROUGE*), more specifically *ROUGE-1* and *ROUGE-L*; Translation Error Rate (*TER*); Character n-gram F-score (*chrF*). We also implemented more **advanced pre-existent metrics**: Metric for Evaluation of Translation with Explicit ORdering (*METEOR*); Google-BLEU (*GLEU*).

2.3 Machine Learning Models

Aside from the MT metrics used as described above, another approach of building end-to-end models was also taken. Firstly, we built a Bag of Words model with the reference and translation for all the corpora with the translation to English (*cs-en, ru-en, zh-en* and *de-en*), for the Finnish translations and another for the Chinese translations. The BoWs were done with *CountVectorizer* and had 10 000 features, ignored features with a document frequency higher than 0,8 and an n-gram range of (1, 3). After performing the train/development split - 0.7/0.3, respectively, and with the shuffling option - for each of the three approaches, the BoWs were given to a Neural Network (*MLP Regressor* with two hidden-layers, each having 10 neurons, and L2 Regularization) and a *Deep Learning NN*, to predict the z-score. The DL NN had *ReLU* activation functions, the *RMSprop* optimizer and *MSE loss* function.

Secondly, we used *LASER* word embeddings (of source, reference and translation) for each language-pair and, similarly to the previous approach, created the train/development split for all six partitions, with the same split of 0.7/0.3. Afterwards, these embeddings were given to the same regression models (MLP and DL NN), also to predict the z-score.

Lastly, we used some of the MT metrics obtained previously that performed better to feed an ensemble model (*Gradient Boosting Regressor*), also with the goal of predicting the z-score and with a 0.7/0.3 train/development split.

3 Results and Discussion

In this section the results will be discussed, and the following tables will be analysed below.

	cs-en		ru-en		zh-en		de-en		en-fi		en-zh		AVG	
	Pear	Kend	Pear	Kend	Pear	Kend	Pear	Kend	Pear	Kend	Pear	Kend	Pear	Kend
Euclidean dist	-0,232	-0,187	-0,183	-0,173	-0,231	-0,172	-0,171	-0,139	-0,369	-0,235	-0,031	-0,037	**-0,207**	**-0,173**
Manhattan dist	-0,211	-0,19	-0,187	-0,176	-0,236	-0,179	-0,167	-0,146	-0,338	-0,233	-0,036	-0,038	**-0,199**	**-0,178**
Cosine sim	0,389	0,256	0,299	0,196	0,266	0,17	0,279	0,189	0,524	0,344	0,023	-0,001	**0,289**	**0,193**
Jaccard sim	0,252	0,167	0,195	0,136	0,19	0,123	0,174	0,115	0,373	0,266	0,031	0,005	**0,193**	**0,1295**
BLEU	0,453	0,305	0,356	0,243	0,339	0,224	0,341	0,233	0,614	0,406	0,431	0,305	**0,394**	**0,274**
ROUGE-1	0,427	0,287	0,332	0,228	0,318	0,21	0,313	0,216	0,532	0,35	0,023	-0,001	**0,325**	**0,222**
ROUGE-L	0,437	0,295	0,344	0,239	0,331	0,221	0,322	0,225	0,521	0,342	0,022	-0,001	**0,338**	**0,232**
chrF	0,403	0,27	0,321	0,219	0,293	0,191	0,315	0,215	0,566	0,369	0,417	0,3	**0,362**	**0,245**
TER	-0,439	-0,316	-0,335	-0,247	-0,264	-0,235	-0,304	-0,224	-0,456	-0,35	-0,074	-0,07	**-0,3195**	**-0,241**
METEOR	0,449	0,304	0,346	0,24	0,34	0,228	0,319	0,224	0,511	0,344	0,022	-0,001	**0,343**	**0,234**
GLEU	0,457	0,306	0,358	0,243	0,342	0,225	0,343	0,233	0,612	0,406	0,464	0,325	**0,408**	**0,275**
AVG	**0,403**	**0,27**	**0,321**	**0,219**	**0,293**	**0,191**	**0,313**	**0,215**	**0,521**	**0,344**	**0,023**	**-0,001**		

Table 1 - Correlation scores between each metric and the z-score (for Preprocessing 1)

From the table above it is possible to notice that the corpus that achieved better average results on the quality of the translations is the English to Finnish one, being also the one where the highest correlation with the z-score was achieved with *BLEU*: 0.614 for *Pearson's* and 0.406 for *Kendall's Tau*. This was also the language pair that achieved the highest average correlation.

The metric that most consistently achieved better results was *GLEU*, thus being the one with highest correlations for the language pairs, on average. *BLEU*, *METEOR*, *TER* and *chrF* also performed well, therefore, these were the ones we used for the ensemble model presented further below.

Another result worth mentioning is that there is not a significant difference in the performance between the two types of preprocessing (see results for Preprocessing 2 in Appendix A, on the Appendices section). Since this similarity was constant for all language pairs, this reveals that, for this example, **adding *stemming* to the corpus did not add any value for the quality of the evaluation metric**. For this reason, on the following models we used only the simplest preprocessing approach, which is also the one that registered slightly higher correlations.

		To English								To Finnish		To Chinese		AVG	
		cs-en		ru-en		zh-en		de-en		en-fi		en-zh			
		Pear	Kend	Pear	Kend	Pear	Kend	Pear	Kend	Pear	Kend	Pear	Kend	Pear	Kend
Train	*NN w/ BoW*	0,874	0,664	0,874	0,664	0,874	0,664	0,874	0,664	0,979	0,907	0,72	0,518	**0,87**	**0,66**
	DL w/ BoW	0,882	0,71	0,882	0,71	0,882	0,71	0,882	0,71	0,969	0,872	0,673	0,496	**0,88**	**0,71**
	NN w/ Embed	0,776	0,548	0,653	0,436	0,606	0,418	0,951	0,788	0,785	0,585	0,7	0,498	**0,745**	**0,546**
	DL w/ Embed	0,878	0,667	0,653	0,436	0,693	0,481	0,704	0,474	0,91	0,738	0,783	0,576	**0,770**	**0,562**
Dev	*NN w/ BoW*	0,266	0,187	0,266	0,187	0,266	0,187	0,266	0,187	0,342	0,221	0,34	0,224	**0,316**	**0,211**
	DL w/ BoW	0,287	0,202	0,287	0,202	0,287	0,202	0,287	0,202	0,385	0,244	0,353	0,233	**0,342**	**0,226**
	NN w/ Embed	0,484	0,321	0,274	0,193	0,35	0,239	0,31	0,22	0,392	0,226	0,439	0,285	**0,375**	**0,247**
	DL w/ Embed	0,509	0,348	0,274	0,193	0,34	0,236	0,342	0,238	0,422	0,288	0,452	0,298	**0,390**	**0,267**

Table 2 - Correlation scores between each model's predictions and the z-score (for Train and Development)
Note: the cells in light grey are the same because they are all relative to the same model that was trained and validated with all language pairs with English translation.

As for the ML and DL models implemented, the results obtained were quite high, especially for the models with word embeddings, for each language pair, which, in the development set, were able to yield higher average correlations than most of the metrics used previously (*Pearson* of 0,375 for MLP and of 0,39 for DL NN). Here, unlike before, the best result was seen on the Czech to English language pair with the Deep Learning NN that resulted on a *Pearson* correlation of 0,509 and a *Kendall's Tau* correlation of 0,348 in the development set.

Despite the good results, the metrics proved to have better correlations for different languages, which is what led us to believe that combining them in an Ensemble model could be a good approach. This way we joined several that were good in general and for each language, to make sure that the model would receive as input complementary metrics, so as to lead to even better predictions.

The best model used to predict the z-score in test set was proved to be the ensemble (*Gradient Boosting Regressor*), presenting a *Pearson* correlation of 0,419 and 0,404, and *Kendall's Tau* of 0,277 and 0,273, for the training and development set respectively, and no evidence of overfitting.

When applying the Ensemble, one approach was to test it separately for each language pair. As expected, the results obtained were better than when using all the pairs, since the test was separated in each language pair, and thus it would be more adapted. Nevertheless, the final decision was to use all the language pairs in one single model, enhancing the ability of generalization, and thus leading to better overall results, when tested against any language.

It is important to note that there were a few null values received in the test set on the reference column, for the *ru-en*, *zh-en* and *en-zh* corpora. All of these were replaced by "*na"*, so as not to remove those rows and still avoid any error resulting from running specific metrics, for instance, *TER* which does not run with an empty reference.

4 Conclusion

During this project, we carried out an NLP task: automatic evaluation of MT. To do this in a proper manner, we started by testing different types of preprocessing related to *lowercasing*, removal of *punctuation*, removal of *stop words*, or the use of *stemming*. During this stage it was essential to deal with the Chinese language separately, due to its uniqueness if compared with the others.

After this, we implemented a number of well-kwon MT metrics like *BLEU, ROUGE* and *METEOR*, to name a few, and also ML and DL models, in order to predict the z-scores on all corpora. These models included bag of words, word embeddings and the results of other MT metrics as initial features.

The model that yielded the most satisfactory results in terms of correlation to the existing quality assessments was the *Gradient Boosting Regressor*, providing for the development set a *Pearson's* correlation 0,404, and *Kendall* of 0,273. Thus, this was the one we used to predict the scores for the test set.

References

[1] Sun Junyi (2021). *jieba: Chinese text segmentation*. Available at: <https://github.com/fxsjy/jieba> [Accessed 20 May 2021].

A. Appendices

	cs-en		ru-en		zh-en		de-en		en-fi		AVG	
	Pear	Kend	Pear	Kend	Pear	Kend	Pear	Kend	Pear	Kend	Pear	Kend
Euclidean dist	-0,243	-0,194	-0,191	-0,18	-0,241	-0,177	-0,179	-0,145	-0,398	-0,256	**-0,241**	**-0,18**
Manhattan dist	-0,225	-0,199	-0,197	-0,183	-0,249	-0,186	-0,176	-0,152	-0,372	-0,254	**-0,225**	**-0,186**
Cosine sim	0,402	0,263	0,309	0,203	0,276	0,176	0,288	0,195	0,539	0,349	**0,309**	**0,203**
Jaccard sim	0,251	0,165	0,196	0,137	0,188	0,121	0,171	0,112	0,366	0,262	**0,196**	**0,137**
BLEU	0,449	0,301	0,353	0,241	0,339	0,224	0,335	0,231	0,6	0,392	**0,353**	**0,241**
ROUGE-1	0,438	0,295	0,341	0,233	0,326	0,214	0,322	0,222	0,548	0,356	**0,341**	**0,233**
ROUGE-L	0,446	0,3	0,349	0,243	0,336	0,223	0,328	0,23	0,536	0,345	**0,349**	**0,243**
chrF	0,403	0,268	0,32	0,22	0,297	0,194	0,311	0,214	0,547	0,357	**0,32**	**0,22**
TER	-0,446	-0,321	-0,342	-0,251	-0,266	-0,238	-0,308	-0,228	-0,467	-0,356	**-0,342**	**-0,251**
METEOR	0,449	0,305	0,348	0,241	0,338	0,226	0,323	0,225	0,527	0,35	**0,348**	**0,241**
GLEU	0,453	0,302	0,355	0,241	0,342	0,225	0,338	0,231	0,597	0,392	**0,355**	**0,241**
AVG	**0,403**	**0,268**	**0,32**	**0,22**	**0,297**	**0,194**	**0,311**	**0,214**	**0,536**	**0,349**		

Appendix A - Correlation scores between each metric and the z-score (for Preprocessing 2)