NOVA IMS MT Metrics Shared Task

Valentyna Rusinova, Svitlana Vasylyeva
M20200591, M20200617

1 Introduction

An ability to work with text data is one of the main technological achievements: it helped many companies to transform free (unstructured) text into a structured format to identify some patterns and new insights making them able to explore hidden relationships within their data. One of the difficult tasks in NLP is Machine translation (MT) evaluation, that determines the effectiveness of the MT system in general, estimates the level of post-editing needed and sets reasonable expectations. Machine translation output can be evaluated automatically, using some ready metrics or by human judges.
The idea of this project is inspired by Metric Shared Task competition and the aim is to develop a metric that predicts the quality of a translation using the reference for six language pairs (cs-en, de-en, en-fi, en-zh, ru-en, zh-en). The metric should correlate well with the existing quality assessments (z-score and avg-score with a given number of annotators). We also evaluated quality of translation for a new text set using the developed metric.

2 Method/Approach

2.1 Corpus

We received a corpus with translations for the six language pairs mentioned above. For each language we created a separate dataframe with columns that contained the original segments, the translations and references, quality assessments (including avg-score given by human annotators, number of annotators and produced z-score).

2.2 Pre-processing

The pre-processing function *preprocessing* was created to work with a text:
A possibility to check different combinations of pre-processing steps was implemented:
- for cs-en, de-en, ru-en, zh-en pairs expanding of English language contractions was made (i.e., you've -> you have);
- lowercasing;
- removing html tags;
- removing punctuation with *regex* (*True or False* parameter in a function call);
- lemmatization (*True or False*). For all language pairs (except en-fi and en-zh) the *WordNetLemmatizer* was used and for Finnish language (en-fi) we used *Voikko* from *libvoikko* library);
- stemming (*True or False*);
- removing stop words for English, Chinese and Finnish (*True or False*);

Also *jieba* library was used for proper tokenization of Chinese words for en-zh because of the nature of Chinese language.

2.3 Automatic evaluation metrics

Four automatic evaluation metrics from *nltk* library were used to evaluate a quality of machine translation: sentence level chrF [1], sentence level GLEU (Google-BLEU) [2], METEOR score for hypothesis with multiple references [3], sentence-level BLEU score [4].
And we also used BERTScore (bert-score 0.3.9 implementation) [5].
All metrics were computed for different pre-processing combinations and a Pearson correlation was calculated for each metric and ‘z-score’ columns for each language pair.

2.4 Developed metric

One of the methods to determine a quality of machine translation is finding a sentence similarity using words embeddings. The sum of word embeddings was found to be an effective model for summarization in [6].

In this project we used Sum and Mean of Word Embeddings (SOWE and MOWE) [7] approach: for each sentence in a pair (reference/translation pair) we calculated sentence embeddings as a weighted mean of words embeddings of the words in a sentence (our weighted mean of the word embeddings takes into account frequency of a word in the sentence and produces a resulting vector). We used pre-trained GloVe word embeddings *glove.6B.300d* for cs-en, de-en, ru-en, zh-en pairs and *sgns.merge.word* for en-zh to calculate sentence embeddings. For en-fi the pre-trained multilingual BERT [8] (‘*bert-base-multilingual-cased’*) and finBERT [9] (‘*TurkuNLP/bert-base-finnish-cased-v1’*) models were used to receive sentence embeddings using hidden states of the last layer of the models because there is no GloVe model for Finnish language.

We used the cosine similarity between two sentences (two resulting vectors) as our metric to measure the similarity between reference and translation, as it is commonly used when comparing distances between embeddings.

3 Results and Discussion

3.1 Results for automatic evaluation metrics

We checked different combinations of pre-processing for nltk automatic evaluation metrics and found Pearson correlation for each nltk metric and ‘z-score’ column (Table 3.1-3.6):

Table 3.1. Without removing of punctuation and stop-words, without lemmatization and stemming (baseline scores)

	chrf	gleu	meteor	bleu
scores_cs	0.462239	0.427909	0.439981	0.468781
scores_de	0.341172	0.310139	0.308153	0.346757
scores_en_fi	0.611567	0.494636	0.491475	0.619928
scores_en_zh	0.423398	0.449157	0.453092	0.468428
scores_ru	0.361388	0.333465	0.336711	0.367557
scores_zh	0.341228	0.317931	0.326447	0.351904

Table 3.2. With removing of punctuation, without removing stop-words, without lemmatization and stemming

	chrf	gleu	meteor	bleu
scores_cs	0.460175	0.442928	0.459244	0.46626
scores_de	0.339221	0.323857	0.323407	0.343768
scores_en_fi	0.606508	0.521099	0.516653	0.616161
scores_en_zh	0.419761	0.44232	0.451714	0.475039
scores_ru	0.35807	0.341896	0.336711	0.362122
scores_zh	0.337752	0.329073	0.326447	0.347637

Table 3.3. Without removing of punctuation and stop-words, with lemmatization and without stemming

	chrf	gleu	meteor	bleu
scores_cs	0.461182	0.42928	0.44041	0.468664
scores_de	0.339801	0.311003	0.308517	0.345696
scores_en_fi	0.596299	0.525408	0.521339	0.60642
scores_en_zh	0.423398	0.449157	0.453092	0.468428
scores_ru	0.361224	0.33507	0.336711	0.366839
scores_zh	0.340611	0.318608	0.326447	0.350394

Table 3.4. Without removing of punctuation and stop-words, without lemmatization and with stemming

	chrf	gleu	meteor	bleu
scores_cs	0.458551	0.433943	0.440132	0.466033
scores_de	0.33548	0.312284	0.310658	0.342397
scores_en_fi	0.595025	0.510169	0.506616	0.609784
scores_en_zh	0.423398	0.449157	0.453092	0.468428
scores_ru	0.360056	0.338979	0.336711	0.366964
scores_zh	0.342474	0.322487	0.326447	0.353436

Table 3.5. Without removing of punctuation, with removing of stop-words, without lemmatization and stemming

	chrf	gleu	meteor	bleu
scores_cs	0.462193	0.427878	0.439901	0.468756
scores_de	0.34117	0.310134	0.308149	0.346754
scores_en_fi	0.611555	0.494621	0.49147	0.619918
scores_en_zh	0.423409	0.449227	0.453161	0.468476
scores_ru	0.361388	0.333425	0.336711	0.36755
scores_zh	0.341249	0.317942	0.326447	0.351916

Table 3.6. With all parameters set to ‘True’ except stemming: with removing of punctuation and stop-words, with lemmatization and without stemming.

	chrf	gleu	meteor	bleu	bert
scores_cs	0.458563	0.444786	0.460774	0.465282	0.567269
scores_de	0.337663	0.326861	0.325655	0.342394	0.421566
scores_en_fi	0.590391	0.560004	0.556457	0.603164	0.615133
scores_en_zh	0.419763	0.442372	0.451762	0.475075	0.541409
scores_ru	0.357554	0.345029	0.336711	0.360968	0.418593
scores_zh	0.3367	0.33273	0.326447	0.345333	0.416299

BERTScore (Table 3.6) was calculated only for last case:

- “with all parameters set to ‘True’ except stemming: with removing of punctuation and stop-words, with lemmatization and without stemming” because of the time-consuming calculations of this metric.

As could be seen for our language pairs and all nltk automatic evaluation metrics the difference between different pre-processing combinations and their influence on a resulting correlation is quite small and depends on language and a chosen metric.

3.2 Results for our cosine similarity metric

For five of our language pairs (cs-en, de-en, ru-en, zh-en, zh-en) sentence emeddings were calculated separately from en-fi pair where embeddings were received from multilingual BERT (‘*bert-base-multilingual-cased’*) and finBERT (‘*TurkuNLP/bert-base-finnish-cased-v1’*) models.
We calculated our cosine similarity metric for cs-en, de-en, ru-en, zh-en, zh-en languages for different pre-processing parameter combinations. The results of Person correlation between developed metric and ‘z-score’ column are summarized in the next tables (Table 4.1-4.3).

Table 4.1. With punctuation removed, without removing stop-words, without lemmatization and stemming (baseline scores)

	scores_cs	scores_de	scores_en_zh	scores_ru	scores_zh
Cosine_similarity (baseline)	0.321221	0.258748	0.360268	0.258712	0.229038

Table 4.2. With removing of punctuation and stop-words, without lemmatization and stemming

	scores_cs	scores_de	scores_en_zh	scores_ru	scores_zh
Cosine_similarity (w/o stopwords)	0.368188	0.290548	0.396741	0.304963	0.27196

Table 4.3. With all parameters set to ‘True’ except stemming: with removing of punctuation and stop-words, with lemmatization and without stemming

	scores_cs	scores_de	scores_en_zh	scores_ru	scores_zh
Cosine_similarity (pre-processing)	0.360295	0.287918	0.396771	0.307792	0.256307

The best result achieved (Table 4.2) was used to select a combination of parameters for pre-processing for test-set (cs-en, de-en, ru-en, zh-en, zh-en).
For Finnish language (en-fi) we used sentence embeddings from hidden states of the last layer of the models of multilingual BERT and finBERT models. Different combinations of pre-processing were tried: without/with removal of punctuation, with/ without lemmatization (*Voikko* library was used to lemmatize Finnish words), all results of Person correlation of our metric and ‘z-score’ are summarized in a next table.

Table 4.4. Comparison results of correlation using multilingual BERT and finBERT sentence embeddings

	en-fi
Multilingual BERT without pre-process	0.150627
Multilingual BERT with pre-process	0.161529
finBERT (TurkuNLP) without pre-process	0.616468
finBERT (TurkuNLP) with pre-process	0.484256

We received notably better results for our metric using finBERT model sentence embeddings without pre-processing. This combination was used to calculate cosine similarity metric for test set en-fi.
The developed cosine similarity metric is computationally cheap metric, especially given pretrained word embeddings are available.
A Python implementation could be found on [10].

4 Conclusion

The purpose of this project was to develop a metric that predicts the quality of a translation and our metric being computationally cheaper than some nltk metrics delivers quite good results that for some language pairs could be even comparable to nltk metrics results.
While working on this project we developed a more complete understanding of the machine translation evaluation method, approaches and difficulties in the field of NLP and Machine Translations.

References

[1] Maja Popovic (2015). *chrF: character n-gram F-score for automatic MT evaluation.* Humboldt University of Berlin, Germany

[2] Yonghui Wu, Mike Schuster, … & Jeffrey Dean. (2016). *Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation.*

[3] Alon Lavie and Abhaya Agarwal (2015*). Meteor: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments*. Language Technologies Institute Carnegie Mellon University Pittsburgh, USA

[4] Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. (2002). *BLEU: a Method for Automatic Evaluation of Machine Translation.* IBM T. J. Watson Research Center. Yorktown Heights, NY, USA

[5] Tianyi Zhang, Varsha Kishore, …& Yoav Artzi (2020) *BERTScore: Evaluating Text Generation with BERT.*

[6] Mikael Kageback, Olof Mogren, Nina Tahmasebi, Devdatt Dubhashi (2014). *Extractive Summarization using Continuous Vector Space Models.* Computer Science & Engineering, Chalmers University of Technology, Goteborg

[7] Lyndon White, Roberto Togneri, Wei Liu, and Mohammed Bennamoun. (2015). *How Well Sentence Embeddings Capture Meaning.* In Proceedings of the 20th Australasian Document Computing Symposium, Association for Computing Machinery, New York, USA

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.*

[9] Antti Virtanen1 Jenna Kanerva Rami Ilo2 Jouni Luoma3 Juhani Luotolahti Tapio Salakoski Filip Ginter Sampo Pyysalo (2019). *Multilingual is not enough: BERT for Finnish.* Turku NLP group, University of Turku

[10] Valentyna Rusinova, Svitlana Vasylyeva (2021). *NOVA IMS MT Metrics Shared Task.* https://github.com/svasylyeva/Text_Mining