BLEU2VEC: the Painfully Familiar Metric on Continuous Vector Space Steroids

Andre Tättar and **Mark Fishel**
Institute of Computer Science
University of Tartu, Estonia
{andre.tattar, fishel}@ut.ee

Abstract

In this participation in the WMT'2017 metrics shared task we implement a fuzzy match score for n-gram precisions in the BLEU metric. To do this we learn n-gram embeddings; we describe two ways of extending the WORD2VEC approach to do so. Evaluation results show that the introduced score beats the original BLEU metric on system and segment level.

1 The Painfully Familiar Metric

The BLEU metric (Papineni et al., 2002) has deeply rooted in the machine translation community and is used in virtually every paper on machine translation methods. Despite the well-known criticism (Callison-Burch et al., 2006) and a decade of collective efforts to come up with a better translation quality metric (from Callison-Burch et al., 2007 to Bojar et al., 2016) it still appeals with its ease of implementation, language independence and competitive agreement rate with human judgments, with the only viable alternative on all three accounts being the recently introduced CHRF (Popovic, 2015).

The original version of BLEU is harsh on single sentences: one of the factors of the score is a geometric mean of n-gram precisions between the translation hypothesis and reference(s) and as a result sentences without 4-gram matches get a score of 0, even if there are good unigram, bigram and possibly trigram matches. There have been several attempts to "soften" this approach by using arithmetic mean instead (NIST, Doddington, 2002), allowing for partial matches using lemmatization and synonyms (METEOR, Banerjee and Lavie, 2005) and directly implementing fuzzy matches between n-grams (LEBLEU, Virpioja and Grönroos, 2015).

Our work is most closely related to LEBLEU, where BLEU is augmented with fuzzy matches based on the character-level Levenshtein distance. Here we use independently learned word and n-gram embeddings instead.

2 The Continuous Vector Space Steroids

Together with neural networks came the necessity to map sparse discrete values (like natural language words) into dense continuous vector representations. This is done explicitly e.g. with WORD2VEC (Mikolov et al., 2013), as well as learned as part of the whole learning process in neural networks-based language models (Mikolov et al., 2010) and translation approaches (Bahdanau et al., 2015). The approach of learning embeddings has since been extended for example to items in a relational database (Barkan and Koenigstein, 2016), sentences and documents (Le and Mikolov, 2014) and even users (Amir et al., 2017).

The core part of this work consists of n-gram embeddings, the aim of which is to find similarities between short phrases like "research paper" and "scientific article", or "do not like" and "hate". We propose two solutions, both reducing the problem to the original WORD2VEC ; the first one only handles n-grams of the same length while the second one is more general. These are described in the following sections.

2.1 Separate N-gram Embeddings

Our first approach is learning separate embedding models for unigrams, bigrams and trigrams. While unigram embeddings are handled by the baseline WORD2VEC method, in this approach we group the n-gram tokens into a single entry, ignoring the overlapping parts, for example:

Uni-grams: this is a test .

Bi-grams: this_is is_a a_test test_.

Tri-grams: this_is_a is_a_test a_test_.

and then compute embeddings for the new tokens with the baseline approach.

Since the number of different n-grams is much higher than single tokens, we filter out bi-grams that occur less than 30 times and tri-grams that occur less than 50 times.

2.2 Joint N-gram Embeddings

Our first method can only learn similarities between n-grams of the same lengths. While it is enough for this submission's metric, it also runs the danger of learning overlapping n-grams, as these are generated next to each other. We therefore define a more general solution.

By modifying the process of extracting input-output training pairs from text sentences we can achieve direct inclusion of both the words and the n-grams, with each of them being treated a separate lexical entry. See Figure 1 for an example of skip-gram training:

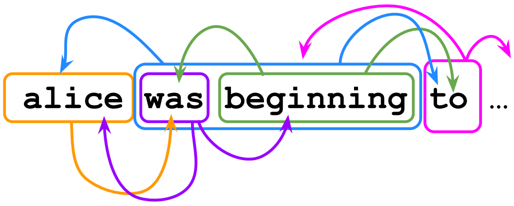

Figure 1: Example of skip-gram training for words and n-grams. Boxes show the input entries and arrows point to output entries; context window width of 1 is used for a simpler figure's sake. We follow (Yu and Dredze, 2015) and predict single words on the output side while feeding words and n-grams on the input side.

In addition to frequency filtering we also sample the n-grams randomly, sometimes including or excluding them from training. To increase the chances of more rare n-grams being included we define the sampling probability based on smoothed reverse frequency:

$$p = \exp(-\beta \log(f)) = \frac{1}{f^{\beta}},$$

where f is the n-gram absolute frequency, p is the sampling probability and β is a small weight. For example with $\beta = \frac{1}{8}$ the sampling likelihood of an tri-gram with minimum frequency (50) is 0.613, while a high frequency like 10000 will have the probability of 0.316. Using this dynamic probability is equivalent to down-sampling the more frequent n-grams, leaving more exposure to the entries with lower frequency.

Finally, by sampling only n-grams that do not overlap we reduce the problem to the original word-level WORD2VEC by randomly re-deciding which n-grams to join into a single lexical entry at each epoch. This also means that n-grams are present as both the input and output entries.

In the next section we apply the learned n-gram embeddings to compute a soft-constraint translation metric score.

3 BLEU2VEC

The original BLEU metric defines a hard constraint: a word or n-gram from the hypothesis is considered either precise or not. Our modification is defined as follows:

- a hypothesis translation word or n-gram present in the reference translation is considered precise (weight 1)
- all other words and n-grams in the hypothesis are aligned to same-length n-grams in the reference by greedily selecting the most similar pair first. Similarity is computed via the cosine of the embeddings, and is used as the pair's weight
- overlaps are not allowed: once a pair is aligned it is removed from the search space for the next n-grams

The rationale behind this simple modification is that partially correct words will be hopefully considered similar by the embedding model, while completely wrong words will only find alignments with lower similarity.

4 Evaluation

In order to evaluate the metric we trained word and n-gram embeddings using the monolingual

Metric	fi-en	de-en	cs-en	ru-en	Average
BLEU	0.929	0.865	0.958	0.851	0.901
BLEU2VEC_SEP	0.953	0.867	0.970	0.857	0.912
BLEU2VEC_JOINT	0.946	0.863	0.969	0.846	0.906

Table 1: System-level correlation between human judgments from WMT'2015 and the original BLEU metric as well as our two modifications. BLEU2VEC_SEP stands for separate n-gram embedding learning and BLEU2VEC_JOINT stands for the joint learning model.

Metric	fi-en	de-en	cs-en	ru-en	Average
SENT-BLEU	0.308	0.360	0.391	0.329	0.347
BLEU2VEC_SEP	0.327	0.366	0.422	0.320	0.359
BLEU2VEC_JOINT	0.326	0.363	0.417	0.318	0.356

Table 2: Segment-level correlation between human judgments and the SENT-BLEU metric as well as our two modifications.

data from the WMT'2017 news translation shared task: we took a random 50 million sentences from the News Crawl corpora for each language (except Chinese, where we used a portion of Common Crawl).

While this year's human judgments are still being annotated at the time of final submission, we present correlation results based on WMT 2015 data for English in Table 1 for system-level correlations and Table 2 for segment-level correlations.

Results show that both our metrics perform better than the baseline on system-level evaluation. In all cases the joint n-gram embedding learning model performs slightly worse than the separate learning approach.

The same effect can be seen on segment-level evaluations, whereas for Russian-English translations the correlation of both our metrics is worse than SENT-BLEU.

5 Discussion and Conclusions

We defined BLEU2VEC, a modification of the BLEU score that uses word and n-gram embedding similarities for fuzzy matches. Compared to our expectations the metric is underwhelming, but still has higher system-level and segment-level correlations than the original BLEU metric in most evaluated cases.

The main disadvantage of the metric is that the embedding models need to be trained for it to work. On one hand, only raw text is needed for the training. On another hand, this means that the results depend on the size of the training material, as well as the text domain overlap and other similarities/dissimilarities between the training data and the evaluated translations. Evaluating how much this affects the metric remains to be done in future work.

Our future plans include evaluating the metric on other languages; one can expect a bigger difference in metric performance for morphologically complex languages, since our metric aims at reducing the sparsity effect of the original BLEU metric. Other ways of representing words with embeddings have to be experimented with, especially the ones where word and character-level representations are mixed, like Charagram (Wieting et al., 2016). It is also interesting to see, whether this metric can be used for hill-climbing and system development.

The code of our implementation is available on GitHub[1].

References

Silvio Amir, Glen Coppersmith, Paula Carvalho, Mário J. Silva, and Byron C. Wallace. 2017. Quantifying mental health from social media with neural user embeddings. *CoRR* abs/1705.00335. http://arxiv.org/abs/1705.00335.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. *Proceedings of the International Conference on Learning Representations (ICLR)* .

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for mt evaluation with improved correlation with human judgments. In *Pro-*

[1] https://github.com/TartuNLP/bleu2vec

ceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization. volume 29, pages 65–72.

Oren Barkan and Noam Koenigstein. 2016. Item2vec: Neural item embedding for collaborative filtering. *CoRR* abs/1603.04259. http://arxiv.org/abs/1603.04259.

Ondřej Bojar, Yvette Graham, Amir Kamran, and Miloš Stanojević. 2016. Results of the wmt16 metrics shared task. In *Proceedings of the First Conference on Machine Translation*. Berlin, Germany, pages 199–231.

Chris Callison-Burch, Cameron Fordyce, Philipp Koehn, Christof Monz, and Josh Schroeder. 2007. (Meta-)evaluation of Machine Translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*. Prague, Czech Republic, pages 136–158.

Chris Callison-Burch, Miles Osborne, and Philipp Koehn. 2006. Re-evaluation the role of bleu in machine translation research. In *EACL*. volume 6, pages 249–256.

George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram cooccurrence statistics. In *Proceedings of the second international conference on Human Language Technology Research*. pages 138–145.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*. pages 1188–1196.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR* abs/1301.3781. http://arxiv.org/abs/1301.3781.

Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Interspeech*. volume 2, page 3.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*. pages 311–318.

Maja Popovic. 2015. chrf: character n-gram f-score for automatic mt evaluation. In *Proceedings of the 10th Workshop on Statistical Machine Translation (WMT-15)*. pages 392–395.

Sami Virpioja and Stig-Arne Grönroos. 2015. Lebleu: N-gram-based translation evaluation score for morphologically complex languages. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*. Lisbon, Portugal, pages 411–416.

John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2016. Charagram: Embedding words and sentences via character n-grams. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas, pages 1504–1515.

Mo Yu and Mark Dredze. 2015. Learning composition models for phrase embeddings. *Transactions of the Association for Computational Linguistics* 3:227–242.