LeBLEU: N-gram-based Translation Evaluation Score for Morphologically Complex Languages

Sami Virpioja
Lingsoft, Inc.
Helsinki, Finland
sami.virpioja@lingsoft.fi

Stig-Arne Grönroos
Aalto University
Dept. of Signal Processing and Acoustics
stig-arne.gronroos@aalto.fi

Abstract

This paper describes the LeBLEU evaluation score for machine translation, submitted to WMT15 Metrics Shared Task. LeBLEU extends the popular BLEU score to consider fuzzy matches between word n-grams. While there are several variants of BLEU that allow to non-exact matches between words either by character-based distance measures or morphological preprocessing, none of them use fuzzy comparison between longer chunks of text. The results on WMT data sets show that fuzzy n-gram matching improves correlations to human evaluation especially for highly compounding languages.

1 Introduction

The quality of machine translation has improved to the level that the translation hypotheses are useful starting points for human translators for almost any language pair. In the post-editing task, the ultimate way to evaluate the machine translation quality is to measure the editing time. Editing times are naturally related to the number and types of the edits—and thus the number of keystrokes (Frederking and Nirenburg, 1994)—the post-editor needs to get the final translation from the hypothesis. If we compare the raw translation hypothesis and its post-edited version, an appropriate edit distance measure should correlate to the edit time. However, implementing such a measure is far from trivial.

In automatic speech recognition, common evaluation measures are Word Error Rate (WER) and Letter Error Rate (LER) that are based on the Levenshtein edit distance (Levenshtein, 1966). LER is more reasonable measure than WER for morphologically complex languages, in which the same word can occur in many inflected and derived forms (Creutz et al., 2007). However, both give too high penalty for the variations in word ordering, which are frequent in translations. Even in English, there are often at least two grammatically correct orders for a complex sentence. For languages in which the grammatical roles are marked by morphology and not the word order, there may be many more options.

An edit distance measure suitable for machine translation would require move operations. However, such measures are computationally very expensive: finding the minimum edit distance with moves is NP-hard (Shapira and Storer, 2002), making it cumbersome for evaluation and unsuitable for automatic tuning of the translation models. Possible solutions include limiting the move operations or searching only for an approximate solution. For example, Translation Edit Rate (TER) by Snover et al. (2006) uses a shift operation that moves a contiguous sequence of words to another location, as well as a greedy search algorithm to find the minimum distance. Stanford Probabilistic Edit Distance Evaluation (SPEDE) by Wang and Manning (2012) applies a probabilistic push-down automaton that captures non-nested, limited distance word swapping.

A different approach to avoid the requirement of exactly same word order in the hypothesis and reference translations is to concentrate on comparing only small parts of the full texts. For example, the popular BLEU metric by Papineni et al. (2002) considers only local ordering of words. To be precise, it calculates the geometric mean precision of the n-grams of length between one and four. As high precision is easy to obtain by providing a very short hypothesis translation, hypotheses that are shorter than the reference are penalized by a brevity penalty.

BLEU, TER and many other word-based methods assume that a single word (or n-gram) is either correct or incorrect, nothing in between. This

Proceedings of the Tenth Workshop on Statistical Machine Translation, pages 411–416,
Lisboa, Portugal, 17-18 September 2015.

is problematic for inflected or derived words (e.g. "translate" and "translated" are considered two different words) as well as compound words (e.g. "salt-and-pepper" vs. "salt and pepper"). This is a minor issue for English, but it makes the evaluation unreliable for many other languages. For example, in English–German translation, producing "Arbeits Geberverband" from "employers' organization" would give no hits if the reference had the compound "Arbeitgeberverband".

A common approach to the problem of inflected word forms—as well as to the simpler issues of uppercase letters and punctuation characters—is preprocessing. For example, METEOR (Banerjee and Lavie, 2005; Denkowski and Lavie, 2011) uses a stemmer. Popović (2011) applies and combines BLEU-style scores based on part-of-speech (POS) tags as well as morphemes induced by the unsupervised method by Creutz and Lagus (2005). Also the AMBER score by Chen and Kuhn (2011) combines many BLEU variants, and in some variants, the words are heuristically segmented.

Our approach is to extend the BLEU metric to work better on morphologically complex languages without using any language-specific resources. Instead of giving one point for exactly same n-gram or zero points for any difference, we include "soft" or "fuzzy" hits for word n-grams based on letter edit distance. We call the score LeBLEU; this name can be interpreted either as "Letter-edit-BLEU" or "Levenshtein-BLEU". LeBLEU has two main parameters, n-gram length and fuzzy match threshold, that are easy to tune for different types of languages.[1]

There are at least three previous approaches that resemble LeBLEU in that they try not to overpenalize different word orderings and word forms, but do not require any preprocessing tools or resources. Denoual and Lepage (2005) simply use the standard BLEU score on the level of characters, treating word delimiters as any other characters. In order to capture long enough sequences of text, they increase the maximum n-gram length to 18. Compared to word-based BLEU, their method does not increase the correlations to human evaluation in English.

Homola et al. (2009) propose a score that is a weighted combination of two measures: an alignment score that applies letter edit distances between the word forms and a structural score that measures the differences in word order. In contrast to LeBLEU, it still strongly penalizes errors in compounding, as the alignment is word-to-word and fuzzy matches are accepted only if the LER between a pair of words is lower than 15%.

More recently, Libovický and Pecina (2014) have proposed "tolerant BLEU", a variant of BLEU that similarly to LeBLEU finds fuzzy matches between hypothesis and reference words. Instead of Levenshtein edit distance, they apply a specific affix distance measure that requires an exact match in the middle of the words. Moreover, they apply a more complex procedure, in which the words between the hypothesis and reference are first aligned using the Munkres algorithm. Then the hypothesis words are replaced by the matched reference words while applying a penalty based on the affix distance, and finally standard BLEU calculations are performed on the modified hypothesis. Similarly to the method by Homola et al. (2009), there is no matching between word n-grams of different lengths.

2 Method

LeBLEU differs from the standard BLEU (Papineni et al., 2002) in the following aspects (in the order of decreasing importance):

First, the matching of word n-grams is fuzzy: for a close match, the hits are increased according to a similarity score. The similarity score is one minus letter edit distance normalized by the length of the longer n-gram in characters. Even though we use the term "letter edit", the calculations are based on all characters, including the spaces between the words. If the similarity score is lower than the selected threshold parameter δ, the fuzzy match is ignored.

In contrast to standard BLEU, there is no need for lowercasing or even tokenization. For example, a punctuation character following a word is included in the n-gram as a part of the word. Thus, with a reasonably low threshold parameter, missing the punctuation character will result only in a relative small decrease in the score.

Second, to facilitate the matching of compound words, the hypothesis n-grams are not matched only to reference n-grams of the same order, but n-grams of any order between one and $2n$, where n is highest order of hypothesis n-gram considered.

Third, the brevity penalty is not based on the

[1] In contrast, for example the AMBER score by Chen and Kuhn (2011) includes nearly 20 weight parameters.

number of word tokens but the number of characters in the data. By this, we try to avoid giving too much penalty for mistakes in compound words. Character-based penalty is also one of the penalty variants in AMBER (Chen and Kuhn, 2011).

Fourth, when calculating mean over the different n-gram orders, arithmetic mean is taken instead of geometric mean. That the arithmetic mean is often a better choice than the geometric mean has been noted also by Song et al. (2013).

2.1 Algorithm

Our algorithm for calculating the LeBLEU score consists of four phases: First, the hypothesis n-grams and their frequencies are collected. Second, hypothesis n-grams are matched to the reference n-grams, collecting the normalized letter-edit scores. Third, the scores are summed up for each n-gram order and normalized by the total number of hypothesis n-grams. Finally, average precision over n-gram orders is calculated and multiplied by the brevity penalty.

Only the second phase differs significantly from calculating the standard BLEU score. It is also the most time-consuming part of the algorithm, so we will describe the implemented optimizations in more detail. We also discuss how further speed-up can be obtained by sampling the hypothesis n-grams in the first phase.

2.1.1 Calculating distances between n-grams

As we need to compare all hypothesis n-grams (up to n) to all reference n-grams (up to $2n$), the worst-case complexity for the number of Levenshtein calculations is $O(n^2HR)$ for hypothesis sentence of H words, reference sentence of R words and maximum n-gram order n. We use several strategies to optimize this task without changing the resulting scores.

To calculate the Levenshtein distances, we use a modified version of python-Levenshtein, a Python extension module written in C.[2] The number of function calls from Python to C is minimized by passing in two lists of strings to compare: all extracted n-grams from the hypothesis and reference. This strategy results in a large number of comparisons, making it attractive to prune comparisons that will not affect the final score due to the threshold parameter δ.

Two lower bounds for Levenshtein distance were used for pruning. The first lower bound is given by the difference in lengths of the two strings: the number of letter edits is at least the absolute difference of the lengths. The second lower bound is the bag distance (Bartolini et al., 2002), which uses the difference between character histograms calculated from the compared strings. In addition to the lower bounds, we use early stopping of the dynamic programming algorithm for Levenshtein distance, if all possible paths have grown past the pruning threshold.

For each hypothesis n-gram, the pruning threshold is initially set to δ. As we are looking only for the m-best matches (where m is the number of times the hypothesis n-gram occurred in the sentence), we can constraint the threshold whenever better matches to the reference n-grams are found. For example, if the two best matches are required, a third score that is worse than the current second-best cannot affect the score. Keeping track of the desired number of best matches can be accomplished using for example a heap data structure. However, most of the n-grams occur only once, in which case the heap degenerates into a single item. To simplify the implementation, we adjust the threshold only in this case.

2.1.2 Sampling of n-grams

Regardless of the optimizations above, the evaluation speed may get impractically slow for very long sentences. In such cases, a suitable approximation is to estimate the precision for only a subset of the hypothesis n-grams. If the sample size is limited to L n-grams, the time complexity becomes $O(LnR)$. A sensible scheme is to select n-grams evenly from the hypothesis sentence. In practice, we exclude or include n-grams starting from every kth word for a suitable value of k.[3] If the gaps are never longer than $n - 1$ words, all words in the hypothesis will influence the result. We set the maximum n-gram sample size L to 2000. If $n = 4$, this means that we use all n-grams if the number of words in the hypothesis $H \leq 500$. Some words in the hypothesis would be completely discarded only if $H > 2000$.

[2] Our fork is available from `https://github.com/Waino/python-Levenshtein`.

[3] If $L/H < 0.5$, we set $k = \lfloor H/L \rfloor$ and include every kth word. Otherwise we set $k = \lfloor H/(H - L) \rfloor$ and exclude every kth word.

3 Experiments

We study the proposed evaluation score using the data sets from the shared tasks of the Workshops on Statistical Machine Translation (WMT). The data sets contain human evaluations for different machine translation systems and system combination outputs. The translation hypotheses are ranked both in the level of segments (individual sentences) and systems. The translation hypotheses and references were used as inputs to the LeBLEU score as such: no preprocessing was performed on the texts.

3.1 Parameter tuning

We tuned the two parameters of the evaluation score on the data sets published from the WMT 2013 and 2014 shared tasks (Macháček and Bojar, 2013; Macháček and Bojar, 2014). We ran a grid search on the parameters for each language and level. We tested four values of the maximum n-gram length n (from 1 to 4) and six values of the fuzzy match threshold δ (from 0.2 to 0.8 using step size 0.1).

Our WMT 2015 submission includes two versions regarding the method parameters: "default" and "optimized". For the default submission, we selected the parameters based on the smallest rank sum over all languages, data sets (2013/2014) and levels of evaluation (system/segment). These parameters, which we set as the default parameters for our implementation, are $n = 4$ and $\delta = 0.4$.

For the optimized submission, we took the parameters with the best average correlation over WMT 2013 and 2014 data sets for each language pair and level of evaluation. The results are shown in Table 1. For the Finnish language that was not present in the 2013 and 2014 shared tasks, we took the best parameters for German, another language with complex morphology and long compound words.

		segment		system	
Source	Target	n	δ	n	δ
English	French	4	0.7	4	0.4
English	German	3	0.2	4	0.2
English	Czech	2	0.3	4	0.3
English	Russian	2	0.3	2	0.2
French	English	3	0.6	4	0.6
German	English	4	0.5	4	0.4
Czech	English	4	0.5	4	0.7
Russian	English	4	0.5	4	0.3

Table 1: Results of parameter optimization for each language pair and level of evaluation (segment or system).

3.2 Results for the WMT shared tasks

Table 2 shows the results from the WMT 2013, WMT 2014, and WMT 2015. Topline for system-level data of WMT 2013 is not included due to the use of Spearman's rank correlation instead of Pearson's product-moment correlation. Segment-level results of WMT 2013 are dominated by single submission, SIMPBLEU-RECALL by Song et al. (2013). Considering morphologically complex languages, LeBLEU would have ranked first in English–German and second in English–Czech and English–Russian. For translations to English, LeBLEU would have ranked in the top five among the 10 methods.

For WMT 2014 segment-level data, optimized LeBLEU provides the highest correlations for all language pairs from English. It also outperforms all the included methods for English–German and English–Russian system-level data. For system-level English–French, it would have ranked 5th. For system-level English–Czech, the optimized parameters yielded lower correlation than the default ones, and neither come close to the topline. Somewhat surprisingly, LeBLEU provides the top correlation for system-level German–English and third best for system-level Czech–English translations. For other system-level pairs to English, and all segment-level pairs to English, the correlations are reasonably high but quite far from the respective toplines. We can also compare LeBLEU to two related methods, standard BLEU and AMBER (Chen and Kuhn, 2011). LeBLEU outperforms both in almost all tasks already with the default parameters. The only exception is the system-level English–Czech task, in which BLEU provided a slightly higher correlation.

In the WMT 2015 evaluation, LeBLEU provides quite stable correlations across the different language pairs: Segment-level correlations are between 0.345–0.436 with default parameters and 0.347–0.438 with optimized parameters. System-level correlations are between 0.850–0.955 with default parameters and 0.842–0.984 with optimized parameters, except for English–Finnish, which gets 0.835 with the default parameters and

			WMT 2013			WMT 2014					WMT 2015		
Source	Target	Level	def.	opt.	top	def.	opt.	ref-B	ref-A	top	def.	opt.	top
English	French	segment	.231	.234	**.261**	.292	**.296**	.256	.264	.293	.345	.347	**.366**
English	Finnish	segment	–	–	–	–	–	–	–	–	.368	.368	**.380**
English	German	segment	.247	**.260**	.254	**.273**	**.273**	.191	.227	.268	.398	**.399**	.398
English	Czech	segment	.167	.168	**.192**	.342	**.349**	.290	.302	.344	.406	.410	**.446**
English	Russian	segment	.230	.233	**.245**	.446	**.449**	.381	.397	.440	.404	.404	**.439**
French	English	segment	.255	.259	**.303**	.380	.395	.378	.367	**.433**	.373	.376	**.398**
Finnish	English	segment	–	–	–	–	–	–	–	–	.383	.391	**.445**
German	English	segment	.256	.262	**.318**	.324	.320	.271	.313	**.380**	.402	.399	**.482**
Czech	English	segment	.225	.227	**.388**	.278	.282	.213	.246	**.328**	.436	.438	**.495**
Russian	English	segment	.229	.230	**.234**	.302	.309	.263	.294	**.355**	.376	.374	**.418**
English	French	system	.971	.971	–	.947	.947	.937	.928	**.960**	.933	.933	**.964**
English	Finnish	system	–	–	–	–	–	–	–	–	.835	.803	**.878**
English	German	system	.947	.919	–	.451	**.531**	.216	.241	.357	.850	.868	**.879**
English	Czech	system	.842	.857	–	.973	.964	.976	.972	**.988**	.953	.952	**.977**
English	Russian	system	.787	.870	–	.926	**.941**	.915	.926	**.941**	.896	.908	**.970**
French	English	system	.948	.956	–	.964	.964	.952	.948	**.981**	.955	.984	**.997**
Finnish	English	system	–	–	–	–	–	–	–	–	.900	.900	**.977**
German	English	system	.933	.933	–	**.963**	**.963**	.832	.910	.943	.916	.916	**.981**
Czech	English	system	.960	.946	–	.918	.988	.909	.744	**.993**	.947	.976	**.993**
Russian	English	system	.836	.855	–	.805	.799	.789	.797	**.870**	.908	.842	**.981**

Table 2: Performance of LeBLEU in recent WMT metrics shared tasks. Pearson's correlation coefficients (system-level data) and average Kendall's tau correlation coefficients (segment-level data) for LeBLEU with default parameters (def.), LeBLEU with optimized parameters (opt.), and topline method for the shared task (top). For WMT 2014 data, also two reference methods are included: BLEU (ref-B) and AMBER (ref-A).

only 0.803 with the German-optimized parameters. The choice of German-based parameters was clearly unsuccessful, and the effect of optimization for evaluation in Finnish remains to be seen. On average, optimization based on WMT 2013 and 2014 data sets improved the performance.

Compared to other methods submitted to WMT 2015, LeBLEU outperformed others in segment-level English–German translation. It also ranked second in system-level English–German and third in segment-level English–French. Moreover, even though unoptimized for the task, it ranked third in segment-level and fourth in system-level English–Finnish evaluations.

4 Conclusions

We have described the LeBLEU evaluation score for machine translation. It is an extension of the popular BLEU evaluation metric, but much more suitable for evaluating machine translation to morphologically complex languages. The extension is conceptually simple and does not require any language-specific resources. Instead, morphological variants and mistakes in compound words are accepted by using fuzzy matching between the word n-grams in the hypothesis and reference translations.

In the WMT15 shared task, LeBLEU provided high correlations to the human evaluations especially when translating from English to a morphologically more complex language. In particular, it outperformed other methods in the segment-level evaluation of English–German translation. The performance is equally good for WMT 2013 and 2014 data sets. This is remarkable especially as the method uses neither rule-based nor data-driven tools for morphological processing. As German is a highly compounding language, this indicates that the mistakes in compound words are frequently over-penalized by the current evaluation methods.

Implementation for the LeBLEU evaluation score is available from `https://github.com/Waino/LeBLEU`.

Acknowledgments

We thank Vesa Siivola for his help on the initial implementation of the method and useful comments on the manuscript.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Ilaria Bartolini, Paolo Ciaccia, and Marco Patella. 2002. String matching with metric trees using an approximate distance. In *String processing and information retrieval*, pages 271–283. Springer.

Boxing Chen and Roland Kuhn. 2011. AMBER: A modified BLEU, enhanced ranking metric. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 71–77, Edinburgh, Scotland, July. Association for Computational Linguistics.

Mathias Creutz and Krista Lagus. 2005. Unsupervised morpheme segmentation and morphology induction from text corpora using Morfessor 1.0. Technical Report A81, Publications in Computer and Information Science, Helsinki University of Technology.

Mathias Creutz, Teemu Hirsimäki, Mikko Kurimo, Antti Puurula, Janne Pylkkönen, Vesa Siivola, Matti Varjokallio, Ebru Arisoy, Murat Saraçlar, and Andreas Stolcke. 2007. Morph-based speech recognition and modeling of out-of-vocabulary words across languages. *ACM Transactions on Speech and Language Processing*, 5(1):3:1–3:29, December.

Michael Denkowski and Alon Lavie. 2011. Meteor 1.3: Automatic metric for reliable optimization and evaluation of machine translation systems. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 85–91, Edinburgh, Scotland, July. Association for Computational Linguistics.

Etienne Denoual and Yves Lepage. 2005. BLEU in characters: towards automatic MT evaluation in languages without word delimiters. In *Companion Volume to the Proceedings of the Second International Joint Conference on Natural Language Processing*, pages 81–86.

Robert Frederking and Sergei Nirenburg. 1994. Three heads are better than one. In *Proceedings of the Fourth Conference on Applied Natural Language Processing*, pages 95–100, Stuttgart, Germany, October. Association for Computational Linguistics.

Petr Homola, Vladislav Kuboň, and Pavel Pecina. 2009. A simple automatic MT evaluation metric. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 33–36, Athens, Greece, March. Association for Computational Linguistics.

Vladimir I. Levenshtein. 1966. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10(8):707–710.

Jindřich Libovický and Pavel Pecina. 2014. Tolerant BLEU: a submission to the WMT14 metrics task. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 409–413, Baltimore, Maryland, USA, June. Association for Computational Linguistics.

Matouš Macháček and Ondřej Bojar. 2013. Results of the WMT13 metrics shared task. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 45–51, Sofia, Bulgaria, August. Association for Computational Linguistics.

Matouš Macháček and Ondřej Bojar. 2014. Results of the WMT14 metrics shared task. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 293–301, Baltimore, Maryland, USA, June. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, PA, USA, July. Association for Computational Linguistics.

Maja Popović. 2011. Morphemes and POS tags for n-gram based evaluation metrics. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 104–107, Edinburgh, Scotland, July. Association for Computational Linguistics.

Dana Shapira and James A. Storer. 2002. Edit distance with move operations. In *Proceedings of the 13th Annual Symposium on Computational Pattern Matching*, pages 85–98.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of Association for Machine Translation in the Americas*, pages 223–231.

Xingyi Song, Trevor Cohn, and Lucia Specia. 2013. BLEU deconstructed: Designing a better MT evaluation metric, March. On-line: `http://staffwww.dcs.shef.ac.uk/people/X.Song/song13deconstructed.pdf`. Accessed Oct 2013.

Mengqiu Wang and Christopher Manning. 2012. SPEDE: Probabilistic edit distance metrics for MT evaluation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 76–83, Montréal, Canada, June. Association for Computational Linguistics.