Enhancing BLEU with Semantic Word Vectors

John Morgan

December 6, 2014

Abstract

Machine translation metrics are proposed that attack the pocity of references problem. Semantic word vectors are incorporated into the matching algorithm for the BLEU machine translation evaluation metric.

1 Introduction

1.1 THE Pocity of References Problem

Naturally occurring sentences have billions of correct translations into one foreign language[3]. By a correct translation of a source sentence we mean any sentence in the target language that has the same meaning as the source sentence. The BLEU[4] metric was designed assuming that the system output would be compared to a set of four reference translations. In practice system developers rarely have access to more than one reference translation. In this work we assume only one human generated reference is available.

The designers of BLEU were aware that input sentences could have many translations into a foreign language. They incorporated features that capture variations of references. Bleu does not require exact match for the entire sentence. It allocates credit to sub-sentence segments by considering matches of overlapping ngrams for n equals 1 through 4. Even with this partial credit assignment and assuming four references, it can hardly be claimed that this coverage is enough to be representative of billions of possible correct translations. A system can produce a perfectly adequate translation and get a very low BLEU score.

1.2 Improving BLEU

Making bleu better is a goal almost as old as BLEU itself. Several projects have been devoted to developing metrics that extend BLEU by expanding the set of references to include more sentences with equivalent meaning. Some of these projects like meteor[1][2], TERP[6][5] and HyTER have produced successful metrics. It is clear that an automatic method of generating synonyms and sentences with equivalent meaning will improve BLEU.

We list some features we would like to see in an improved BLEU metric and some features we want to keep. We want to assign credit to approximate matches. System output that expresses similar meaning to the input should receive adequate credit from the metric. We want it to adapt easily to a new language pair. By easily Adaptable to a new language pair, we mean requiring little or no human annotation. We want a metric that assigns reasonable scores to partial output. We want to use this metric as a component that measures translation quality in interactive incremental language systems. Such systems need to measure the quality of partial inputs.

1.3 Word Similarities

The ultimate goal of our project is to attack the pocity of references problem directly by using the semantic vector space to generate references equivalent to a single given reference. In this paper however, we investigate an easier solution. We use the similarity measured defined in the semantic vector space to provide scores for *n*grams that do not have exact matches in the reference. Word2vec embedds single word tokens into a vector space, so it makes sense to measure similarity between words in that space. The word2vec developers suggest that the similarity function can be extended from words to phrases. Although we do not believe that this is theoretically well-founded, we take on this suggestion and build two enhanced BLEU metrics that compute similarities between phrases. One of these metrics yields a significant boost in scores relative to BLEU. Many *n*grams that did not match exactly receive a non-zero similarity measure with reference phrases. However, an inspection of the matches indicates that a large portion of them are spurious. We conclude that this is not an appropriate use of the similarity measure to provide approximate match scores for phrases. We instead believe that more sophisticated approaches need to be taken to generate phrases with equivalent meaning to a given reference.

2 Background

2.1 Previous Work

2.1.1 BLEU

Advantages quick inexpensive language-independent correlates highly with human evaluation has little marginal cost per run Best at parameter optimization BLEU is well known to have weaknesses. Work has been done to remedy these weaknesses by incorporating llinguistic features into the evaluation metric. These enhancements make the metrics more accurate and correlate more with human judgements. Precision and Recall Bleu computes precision for ngrams matches for n between 1 and 4.

2.1.2 METEOR

Meteor is an evaluation metric that takes into consideration stem, synonym, and paraphrase matches between words inaddition to exact string matches. Enhancements: Porter Stemming. Words are determined to be synonyms if they share a synset in Wordnet. Function word discounting.

These enhancements yielded improvements over bleu, but they are language dependent. later versions have incorporated methods for adapting METEOR to new languages. Meteor extends BLEU by incorporating semantic features into the evaluation of SMT output. We similarly enhance BLEU with semantic knowledge by replacing words in the reference transcriptions with words that are close in meaning.

2.1.3 TERP

TERP builds on Translation Edit Rate (TER), which is a metric that uses the minimum edit distance to score translations. TERP inherits the two stemming and synonym enhancements from meteor. TERP takes a step up from comparing words with equivalent meaning (synonyms) to comparing phrases with equivalent meaning (paraphrases). TERP is also grounded in human judgements. The scoring function parameters, that is, the weights associated with the diffferent edit operations, are optimized to correlate with the editing operations performed by humans. TERP also produces as a byproduct an alignment between the hypothesis and reference.

2.1.4 HyTER

Dreyer and Marcu's work on HyTER solves the reference posity problem by employing human annotators to write alternative translations for phrases. The phrases are compiled into Recursive Transition Networks that encode an exponential number of reference sentences. It has been shown that HyTER is better at differentiating human from machine translation than bleu, terp, and meteor. HyTER also ranks machine translation systems performance better than these metrics, where the correct ranking is given by human judgements. HyTER also outperforms the other automatic methods as corpus size decreases from document to sentence. In summary, HyTER has clear advantages over other automatic metrics. However, HyTER has one disadvantage we want to avoid. The process of creating the meaning equivalent networks used by HyTER requires a prohibitively large amount of human labor.

We provide background on the method of semantic vector spaces, distributional representation semantics, and deep learning that we rely on to obtain word synonyms and meaning equivalent phrases and sentences. We devote a section to our method incorporating synonyms into bleu. We discuss our methods for scoring phrases and sentences in two sections. Distributional Representational Semantics We use the Word2vec tool. The modification will consist of passing the system hypothesis through word2vec to generate candidate synonyms together with smoothing factors. We will also use this method to generate

reference sentences with similar meaning to a given reference sentence. The word2vec tool is used to build a vector space word distribution model of the target language. Then a distance function defined in the vector space is used to generate a neighborhood of words that are close to the words in the reference transcriptions. Word2vec only has one language-dependent requirement:a large monolingual corpus of text. It does not require a parser, a stemmer, or a part of speech tagger. The representations of words are learned using the distributed Skip-gram or Continuous Bag-of-Words (CBOW) models recently proposed by Mikolov. These models learn word representations using a neural network architecture that predicts the neighbors of a word. We embedd a large text corpus in the target language into a vector space using either the cbow or skipgram neural network. This embedding gives us two methods that we can use for improving bleu scores. We can measure the distance between word vectors. We can generate a list of the closest word vectors to a given word vector. We will explain how we use these methods in the next section.

3 Methods

We propose four methods that we group into 2 major classes: segment methods and hierarchical methods. We subdivide the segment methods into word-level methods and phrase-level methods. There are two features of semantic vector space embeddings that we use to design our methods. The vector space embeddings come with an associated similarity measure. The methods that use this feature we will call similarity methods. The vector space embeddings can also be used to generate a list of vectors in a neighborhood of a given vector. The vectors in the neighborhood can be ranked by their distance to the given vector. We use this feature to generate extra references for a given reference. Methods that use this feature we will call generration methods.

Our simplest method only differs from BLEU in the precision computation for 1grams. When a candidate 1gram does not match any 1gram in the reference, we compute the similarities with each of the remaining 1grams in the reference. Instead of adding 0 to our match count, we add the maximum of these similarities. We then remove one instance of the 1gram that yielded the maximum similarity. This step of removing the argmax 1gram is done to mitigate the overgeneration problem. In the discussion section we consider a problem that this step causes and for which we do not yet have a solution.

We consider the 1gram case separately, because it has a stronger theoretical basis. The similarities are given to us by word2vec. The vector space embedding performed by word2vecd operates on 1grams. Similarities for higher $n$grams are computed and we exploit this later in our phrase-level method, but we see this as having weaker theoretical foundations.

3.1 Data

We test the metrics on the WMT13 corpus. This corpus consists of 3000 segment pairs. Each pair consists of a system output and one reference. The system output has 55786 tokens The reference has 56089 tokens.

4 Results

metric	1-grams	2-grams	3-grams	4-grams
BLEU SLEU VLEU	56089	53089	50105	47138
XLEU	153916	151194	149844	149160

Table 1: Total $n$grams.

	n-gram matches				precisions			
1	2	3	4	1	2	3	4	
BLEU	27867	13406	7070	3853	0.50	0.25	0.14	0.08
SLEU	27939.7 13406.0	7070.0	3853.0	0.50	0.25	0.14	0.08	
VLEU	27938.0	13489.3	7151.3	3912.4	0.50	0.25	0.14	0.08
XLEU	47364.6	41186.7	30672.9	22107.1	0.31	0.27	0.20	0.15

Table 2: Matches and precisions.

BLEU	0.193981132795
SLEU	0.194107449678
VLEU	0.195708957845
XLEU	0.210213517967

Table 3: Scores.

5 Discussion

Bleu is a precision metric. Consider the following candidate/reference pair: the the the the the the the the cat on the mat We call this the the overgeneration problem (the two the tokens are not a typo) We handle this problem by removing a the in the reference each time it is matched . This takes care of the the overgeneration problem in the standard bleu metric. The the overgeneration problem remains in our method. Consider the candidate reference pair: The price for the weapon I learned there I found out the price of the weapon only there The word "for" does not appear in the reference. Our method obtains similarity scores between "for" and each word in the reference that has not been mathced yet. Unfortunately, the word with highest similarity to "for"

is “there. “there” is chosen as the match for “for” and is removed from the referenced word list. When we later try to find a match for “there” an exact match is not made. In this case the standard BLEU metric assigns a higher score to the candidate than our method.

References

[1] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *ACL*. Association for Computational Linguistics, 2005.

[2] Michael Denkowski and Alon Lavie. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*, 2014.

[3] Markus Dreyer and Daniel Marcu. Hyter: Meaning-equivalent semantics for translation evaluation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 162–171, Montréal, Canada, June 2012. Association for Computational Linguistics.

[4] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In *ACL*, 2002.

[5] Matt Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz. TERp: A system description. In *Proceedings of the First NIST Metrics for Machine Translation Challenge (MetricsMATR)*, October 2008.

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