NIST Metrics MATR System Description

SEPIA: Surface Span Extension to Syntactic Dependency Precision-based MT Evaluation

Nizar Habash and Ahmed Elkholy
Center for Computational Learning Systems
Columbia University
{habash,akholy}@ccls.columbia.edu

1 Introduction

We present a new Machine Translation (MT) evaluation metric, SEPIA. SEPIA falls within the class of syntactically-aware evaluation metrics, which have been getting a lot of attention recently (Liu and Gildea, 2005; Owczarzak et al., 2007; Giménez and Màrquez, 2007). Specifically, SEPIA uses dependency representation but extends it to include surface span as a factor in the evaluation score. The dependency surface span is the surface distance between two words that are in a direct relationship in a dependency tree. The basic idea behind SEPIA is that long-distance dependencies should receive a greater weight in MT evaluation metrics than short-distance dependencies. This is because we suspect that having more long-distance matches indicates a higher degree of grammaticality. In the rest of this document we describe the SEPIA metric and its variants, and the publicly available SEPIA package.

2 SEPIA

SEPIA evaluates a translation hypothesis segment (sentence) by computing a score based on a brevity-penalty-adjusted mean of multiple modified precision-based sub-scores. SEPIA uses two types of sub-scores: surface n-gram precision sub-scores (similar to BLEU (Papineni et al., 2002)) and span-extended structural bigram precision sub-scores. We next discuss the latter type of sub-scores which are unique to SEPIA.

2.1 Span-Extended Structural Bigram Precision

A structural bigram (SB) is defined as a head word chain of size 2 (heads) in a dependency representation of the hypothesis/reference sentence. For example, in Figure 1, the edges linking the words *Among-crises*, *mentioned-Among* and *mentioned-dispute* represent SBs. An SB can simply be the parent-child word pair or it can include additional information such as the relation of child to parent (e.g., *Among-obj-crises*), the part-of-speech (POS) of both child and parent (e.g., *Among/IN-crises/NNS*), the relative order of the two (e.g., *Among-<-crises* or *mentioned->-Among*), or any combination of the above (e.g., *Among/IN-<-obj-crises/NNS*).

We define the surface span (SS) to be the absolute surface distance between parent and child in an SB. For the SBs *Among-crises* and *mentioned-Among*, the SS values are 5 and 12, respectively. Overall, in the tree in Figure 1, there are six SBs with SS of 1, two SBs with SS of 2, three SBs with SS of 3 and one SB each for SS values 4, 5, 10 and 12.

For each unique SS value, n, associated with any SB in the hypothesis tree, we define SS_n as the count of all the SBs that have an SS value of n. We also define $SSclip_n$ as the count of all the hypothesis SBs (with SS value of n) that match reference SBs. However, if the number of matching hypothesis SBs exceeds the maximum seen in any reference tree, we use a partial count equal to (maximum # of reference SBs / # of hypothesis SBs) in computing $SSclip_n$. This is our variant of *clipping*, used by other precision-based metrics (Papineni et al., 2002) to minimize gaming. Finally, we define the set $SPANS$ to contain all the unique SS values seen in the hypothesis tree.

Next, we describe two span-extended SB precision sub-scores, which vary in how they use the SS of an SB: SN_x and SPN.

First, the sub-score SN_x is computed as follows:

$$SN_x = \frac{\sum\limits_{n \in SPANS} SSclip_n \times n^x}{\sum\limits_{n \in SPANS} SS_n \times n^x}$$

SN_x is basically the span-weighted precision of hypothesis SBs matching reference SBs. The weighing is controlled through the power term x. The default value of x is 0, which assigns all SBs equal weight regardless of the SS value. A power term of 1 effectively multiplies the count of an SB by its SS value. A multiplier of 2 multiplies the count by the square of the SS value (and so on). This allows the user to give a bigger weight to the longer-distance matching spans.

Second, the sub-score SPN is computed as follows:

$$SPN = \frac{1}{|SPANS|} \sum_{n \in SPANS} \frac{SSclip_n}{SS_n}$$

SPN is basically the average of all SS-value-specific precision calculations. This scoring approach normalizes the frequency of SS values. This effectively gives more weight to the long-distance SBs because of the Zipfian distribution of SSs: shorter spans appear more frequently than longer spans.

Although the two scoring methods are different, they both give more weight to long-distance dependencies than to short-distance dependencies.

2.2 Sub-Score Combination

The segment-level SEPIA score is computed by taking the mean of any subset of the sub-scores described above, including both surface n-gram and SB sub-scores. Note that using the surface n-gram sub-scores alone is comparable to using BLEU. The score is further adjusted by multiplying it with a brevity penalty factor. The brevity penalty factor equals $(1 + min(0, 1-(ShortestRefLength/HypLength)))$, where $ShortestRefLength$ is the length of the shortest reference sentence and $HypLength$ is the length of the hypothesis sentence. Document-level scores are computed as a segment-length-weighted (in words) average of segment scores. Similarly, system-level scores are computed as a document-length-weighted (in segments) average of document scores.

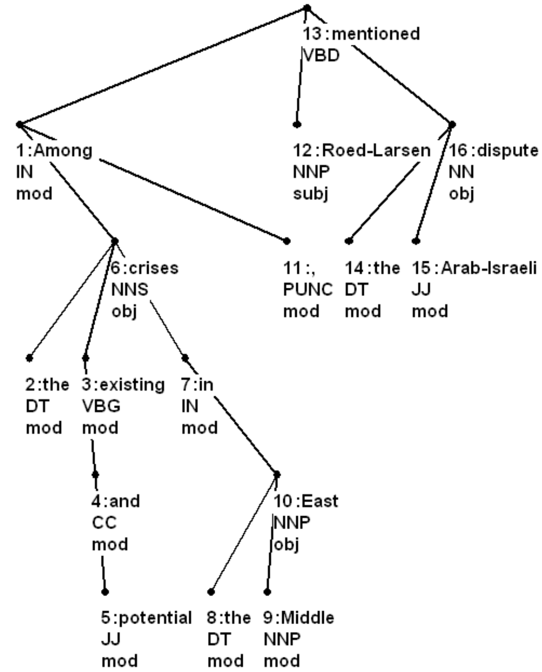

Figure 1: A dependency tree analysis for the sentence *Among the existing and potential crises in the Middle East, Roed-Larsen mentioned the Arab-Israeli dispute.*

3 SEPIA Package

SEPIA's main script is implemented in Perl as an extension to NIST's MTEval-v11b.pl script. SEPIA uses the MICA dependency parser (Nasr and Rambow, 2006), which is included in this package with its authors' permission. The SEPIA script expects a *mode* argument that allows users to specify different combinations of sub-scores: surface n-grams of size 1 through 4, SN_x (with different x values) and SPN. In addition, the basic *word-word* SB definition can be modified to include any combination of the following: POS (xP), relation/label (xR) and relative order (xO). Other parameters control whether a brevity penalty is applied or not, and whether the harmonic mean is used to combine sub-scores instead of the arithmetic mean. The SEPIA package is available to researchers as open source. Please contact the authors to acquire a copy of it.

4 Future Plans

In the future we plan to extend SEPIA in different directions. First, we would like to extend its linguistic features to include semantic role labels and Word-

Net synset expansions. Secondly, we would also like to allow parametrizable weighing of different subscores in score combination. Finally, we would like to extend SEPIA to evaluate MT into languages other than English.

Acknowledgments

This work was partially funded under the DARPA GALE program (contract W0853748) through IBM. We would like to thank Owen Rambow and Srinivas Bangalore for making MICA available and for helping us integrate it in SEPIA. We also would like to thank Bonnie Dorr and Matt Snover, who made some of their evaluation code available to us. Finally, we would like to thank Ryan Roth and Warren Churchill for helpful discussions.

References

Jesús Giménez and Lluís Màrquez. 2007. Linguistic features for automatic evaluation of heterogeneous MT systems. In *Proceedings of the Association for Computational Linguistics (ACL) Second Workshop on Statistical Machine Translation*, Prague, Czech Republic.

Ding Liu and Daniel Gildea. 2005. Syntactic features for evaluation of machine translation. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, Ann Arbor, Michigan.

Alexis Nasr and Owen Rambow. 2006. Parsing with Lexicalized Probabilistic Recursive Transition Networks. In *Finite-State Methods and Natural Language Processing*. Springer Verlag Lecture Notes in Commputer Science.

Karolina Owczarzak, Josef van Genabith, and Andy Way. 2007. Labelled dependencies in machine translation evaluation. In *Proceedings of the ACL Second Workshop on Statistical Machine Translation*, Prague, Czech Republic.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of ACL-02, Philadelphia, PA.*