BERTScore Paper Report

Abraham Paul Elenjical

February 18, 2024

1 Introduction

BERTScore[1] is a language generation evaluation metric introduced in 2020, which improved upon the previous methods of automatic evaluation of the similarity of generated text to the gold standard, similarity with the reference sentence annotated by human annotators.

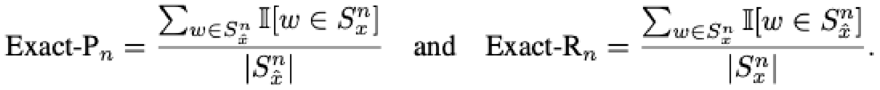
2 Paper Summary

2.1 Prior Metrics of Evaluation

The paper begins by introducing the prior metrics of evalution, their approaches to evaluations, and their shortcomings.

2.1.1 n-GRAM MATCHING APPROACHES

N-gram matching model, as the name suggests, primarily use the number of common n-grams as an evaluation metric of similarity. The exact scores match precision and recall scores are calculated using the formula: Specific metrics have their slight variations to this metric -



$$\text{Exact-P}_n = \frac{\sum_{w \in S^n_{\hat{x}}} \mathbb{I}[w \in S^n_x]}{|S^n_{\hat{x}}|} \quad \text{and} \quad \text{Exact-R}_n = \frac{\sum_{w \in S^n_x} \mathbb{I}[w \in S^n_{\hat{x}}]}{|S^n_x|}.$$

Figure 1: Precison and Recall Scores.

- BLEU - one of the most widely used metrics in machine translation, uses n-gram matching along with 3 major improvements to the algorithm.
  - It allows each reference n-gram to be matched only once.
  - The number of exact matches between the reference sentence and the candidate sentence is divided by the total number of n-grams in the candidate.
  - It averages the scores for different n-gram lengths geometrically.
- METEOR allows backing off from exact unigram matching to matching synonyms, word stems, etc. (would match leaves if no match is found for the unigram ”leaf”).
- Other popular metrics using n-Gram Matching include BLEU, SENT-BLEU and NIST.

2.1.2 EDIT-DISTANCE-BASED METRICS

Some methods use Word Edit Distance or Word Error Rate as a factor in calculating their scores. For example, TER computes the score by normalizing the edit distance by the number of words in the reference sentence.

2.1.3 EMBEDDING-BASED METRICS

Word Embeddings is a technique quantifies individual words as real-valued vectors. Methods such as YISI-1 and YISI-1 make use of word embeddings and shallow semantic parses to compute the similarity. A point to note is that most of the previous evaluation metrics used Static Word Embeddings for their calculations.

2.1.4 LEARNED METRICS

2.2 BERTScore

The paper then goes on to introduce BERTScore and its approach to the task of automatic evaluation.

- In BERTScore, each token in the candidate and reference sentences is converted to contextual embedding form. Contextual embeddings differ from Static Word Embeddings as it generates different values for the vector based on the context the token is encountered in. BERT is used for the tokenization and a Transformer Model is used for the calculation of embeddings.

- After the tokenization and embeddings calculation, the pairwise similarity of the embeddings is measured using cosine similarity, boosted by the use of pre-normalized vectors to reduce calculation.

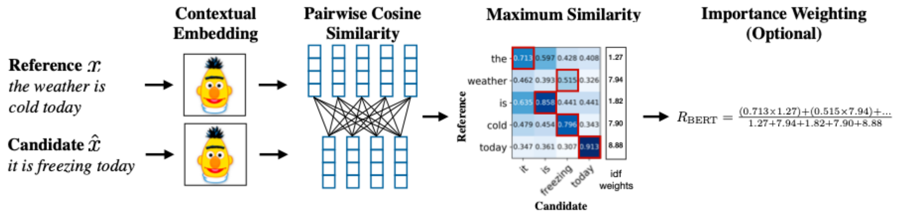

Contextual Embedding
Pairwise Cosine Similarity
Maximum Similarity
Importance Weighting (Optional)
Reference x
the weather is cold today
Candidate $\hat{x}$
it is freezing today
Reference
the 0.713 0.597 0.428 0.408 1.27
weather 0.462 0.393 0.515 0.326 7.94
is 0.635 0.858 0.441 0.441 1.82
cold 0.479 0.454 0.796 0.343 7.90
today 0.347 0.361 0.307 0.913 8.88
it
is
freezing
today
idf weights
Candidate
$R_{\mathrm{BERT}} = \frac{(0.713\times1.27)+(0.515\times7.94)+...}{1.27+7.94+1.82+7.90+8.88}$

Figure 1: Illustration of the computation of the recall metric R_{BERT}. Given the reference x and candidate $\hat{x}$, we compute BERT embeddings and pairwise cosine similarity. We highlight the greedy matching in red, and include the optional idf importance weighting.

Figure 2: BERT embeddings and Calculation.

- The BERTScore is then calculated by using greedy matching to match each candidate token embedding efficiently with the ref. token it is most similar to. Precision, Recall and F1 Scores are calculated.

$$R_{\mathrm{BERT}} = \frac{1}{|x|} \sum_{x_i \in x} \max_{\hat{x}_j \in \hat{x}} \mathbf{x}_i^\top \hat{\mathbf{x}}_j \;, \quad P_{\mathrm{BERT}} = \frac{1}{|\hat{x}|} \sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} \mathbf{x}_i^\top \hat{\mathbf{x}}_j \;, \quad F_{\mathrm{BERT}} = 2\frac{P_{\mathrm{BERT}} \cdot R_{\mathrm{BERT}}}{P_{\mathrm{BERT}} + R_{\mathrm{BERT}}} \;.$$

Figure 3: BERTScore

- Another additional feature used is the inverse-document frequency, since it has been observed that less frequently occurring words generally have more value when considering the semantic meaning of a sentence/text. So the recall of a sentence would be calculated as:

$$R_{\mathrm{BERT}} = \frac{\sum_{x_i \in x} \mathrm{idf}(x_i) \max_{\hat{x}_j \in \hat{x}} \mathbf{x}_i^\top \hat{\mathbf{x}}_j}{\sum_{x_i \in x} \mathrm{idf}(x_i)} \;.$$

Figure 4: Recall with idf calculation.

- Since the final range of scores is not evenly spread out between -1 and 1 (cosine similarity), the baseline is rescaled by shifting the baseline to an empirical lower bound calculated using large datasets like Common Crawl.

$$\hat{R}_{\mathrm{BERT}} = \frac{R_{\mathrm{BERT}} - b}{1 - b} \quad .$$

Figure 5: Baseline Calculation

2.3 Evaluation

The two main tasks on which the evaluation is conducted are Machine Translation and Image Captioning.
Evaluates twelve pre-trained contextual embedding models, including variants of BERT, RoBERTa, XLNet, and XLM.

2.3.1 Machine Translation

The main evaluation corpus is the **WMT18 metric evaluation dataset** which contains predictions of 149 translation systems across 14 language pairs, gold references, and two types of human judgment scores. They use absolute Pearson correlation and Kendall rank correlation to evaluate metric quality

2.3.2 Image Captioning

The main evaluation corpus is the human judgments of twelve submission entries **COCO 2015 Captioning Challenge**. They use Pearson correlation with two system-level metrics: the percentage of captions that are evaluated as better or equal to human captions (M1) and the percentage of captions that are indistinguishable from human captions (M2).

2.4 Results

2.4.1 Machine Translation

In system-level correlation to human judgements, correlations on hybrid systems, and model selection performance, BERTScore is noted as consistently performing above average. Although RUSE shows somewhat similar performances on English datasets, they argue that since it is a supervised, human-trained model, it does not work without additional training.
The authors admit that applying importance weighting using idf at times provides small benefit, but in other cases **does not help**.
It is also noted that F1Score works reliably as a good measure across all settings.

2.4.2 Image Captioning

"BERTSCORE outperforms all task-agnostic baselines by large margins."[1]
In contrast to Machine Translation, in Image Captioning, idf played a significant role in improving the results, indicating the importance given to content words by the annotators.

3 Opinions

3.1 Major Strengths

- **Contextual Word Embeddings** - The usage of contextual rather than static embeddings evidently played a significant role in the improvement of scores across all tasks. It allows the model to understand nuances and polysemy of words.

- **In-Depth Analysis of the Performance of Models** - The paper uses several different evaluation metrics such as system level correlation, correlations on hybrid systems and model selection performance to compare the different evaluation metrics, giving a more in-depth analysis of their performances.

- **Novelty** - BERTScore was one of the first automatic metrics to leverage pre-trained language models like BERT for evaluating text generation quality.

3.2 Major Weaknesses

- **Reliance on Human annotated gold standards** - Although not an immediately solvable issues, the evident reliance on human annotated reference texts definitely restricts the sizes of the datasets that can be used for evaluation of the metrics.

3.3 Scope for Improvements

- **Understanding of the usage of idf for improvement in evaluation** - The authors themselves admit that there is scope for improvement in understanding the factors which dictate the impact of idf as an additional feature in the calculation of scores.

- **Moving away from task-specific evaluation** - It is possible to potentially generalize to other text generation tasks like summarization.

- **Incorporating the use of other Linguistic Features** - Like the authors have used idf to improve accuracy of evaluation, we can explore the usage of other linguistics features and models such as Dependency Parse Trees, POS Tagging, or Chunking to improve model performance.

References

[1] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert, 2020.