E0 270 Machine Learning

Course Project

Automatic Evaluation Metrics for Multi-Sentence Text

By
Eshwar S R
Karan Jeswani
Rahul John Roy

Mentored by Annervaz K M

Aim and Problem Statement

- Human evaluation of model generated text is accurate but very time consuming, hence automatic evaluation metrics are developed which can perform this task, and if their correlation with human evaluation is good enough, we can use it to quickly evaluate such natural language generation models, thus accelerating research in the field.
- Objective of our project is to review currently used metrics, like BLEU[1], Word Mover Similarity[2], Sentence Mover Similarity[3], and BERTScore[4] and evaluate their correlation with human level judgements.
- We calculate scores provided by the metrics stated above, on the Automated Student Assessment Prize (ASAP) AES[7] and SAS[8] dataset, which is an Essay Evaluation task. We have also tested the metrics on CNN/DailyMail[9] dataset.
- We also use the WMT18 dataset[10] to compare results with BERTScore paper.

BLEU (Bilingual Evaluation Understudy)

- It works by calculating n-gram co-occurrence statistics between reference and candidate sentence, where each word in the candidate text is clipped by the maximum number of the same word occurring in any reference.
- 1-gram matches lead to fulfilling of the adequacy criteria and the longer n-gram matches account for fluency.
- A brevity penalty is introduced so that very small candidate sentences don’t get high scores if all words match.
- Benefits of using BLEU is that it is very fast and shows good correlation for machine translation task (it is also the most widely used metric for this task).
- Recall is not taken into account because for a list of references, recalling all words would lead to inaccurate scores. Instead, brevity penalty is included in BLEU.

BLEU (contd.)

The final BLEU score is given by:

$$BLEU = BP * \exp\left(\sum_{n=1}^{N} \omega_n log(p_n)\right)$$

where,
p_n = modified n-gram precisions up to length N;
ω_n = positive weights summing up to 1

In the Baseline BLEU, N = 4 and $\omega_n = 1/N$.

BP = exp(1 - r/c)

where

r = Best match reference length
c = Candidate length

Earth Mover's Distance (EMD)

- Also known as Wasserstein metric
- Calculates distance between 2 probability distributions.
- Was first introduced to measure difference between images.
- Intuitively, EMD[3] can be explained better with sand and holes analogy. Consider one distribution as sand (heap size corresponding to weights) at different points. And the other one with holes (depth corresponding to weights) at different points. EMD is the minimum work done in moving sand from one distribution to holes in another distribution.
- Recently, this is being used in GANs.

EMD (contd.)

- More formally, P = {(p1, wp1),(p2, wp2), ...,(pm, wpm)}, where pi is the cluster representative and wpi is the weight of the cluster.
- Similarly another signature Q = {(q1, wq1),(q2, wq2), ...,(qm, wqm)} has n clusters.
- Let D = [di,j] be the ground distance between clusters pi and qj .
- We want to find a flow F = [fi,j] the flow between pi and qj , that minimizes the overall cost. subjected to the constraints:

$$f_{i,j} \geq 0, 1 \leq i \leq m, 1 \leq j \leq n$$

$$\sum_{j=1}^{n} f_{i,j} \leq w_{pi}, 1 \leq i \leq m$$

$$\sum_{i=1}^{m} f_{i,j} \leq w_{qj}, 1 \leq j \leq n$$

$$\sum_{i=1}^{m}\sum_{j=1}^{n} f_{i,j} = \min\left\{ \sum_{i=1}^{m} w_{pi}, \quad \sum_{j=1}^{n} w_{qj} \right\}$$

$$EMD(P,Q) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{i,j} d_{i,j}}{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{i,j}}$$

Word movers similarity (WMS)

- First time EMD was used in NLP for measuring text similarity.
- The words are represented as points in d dimensional space (d is the dimension of embedding vectors).
- For calculating the WMD (i.e EMD in this context), the distance between the points can be euclidean distance and the weights of each point is its term frequency given by count(word) / total number of words.
- In the proposed paper, the authors used word2vec for computing the embeddings.
- This metrics outperformed the other n-gram metrics like BLEU.

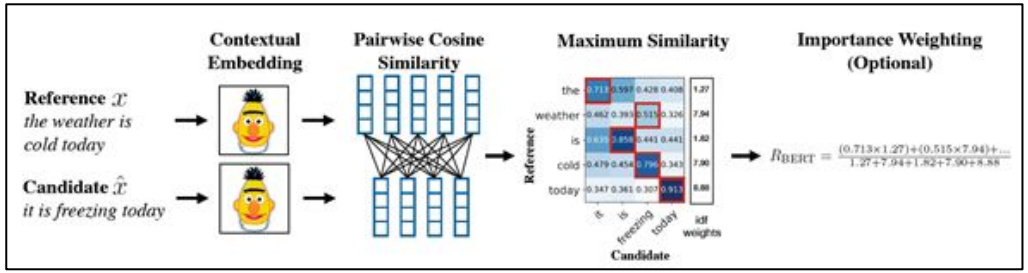
Sentence Mover Similarity (SMS)

SMS is very similar to WMD. The authors proposed 2 metrics.
1. Sentence Mover Similarity
- The embeddings were found for each sentence instead of individual words.
- The sentence embeddings can be computed as the mean of the embeddings of the constituent words.
- The weights for each sentence is number of words in sentence divided by total number of words in the document.

2. Sentence + Word Movers Similarity
- The embeddings from both sentences and words are used in computing the scores.

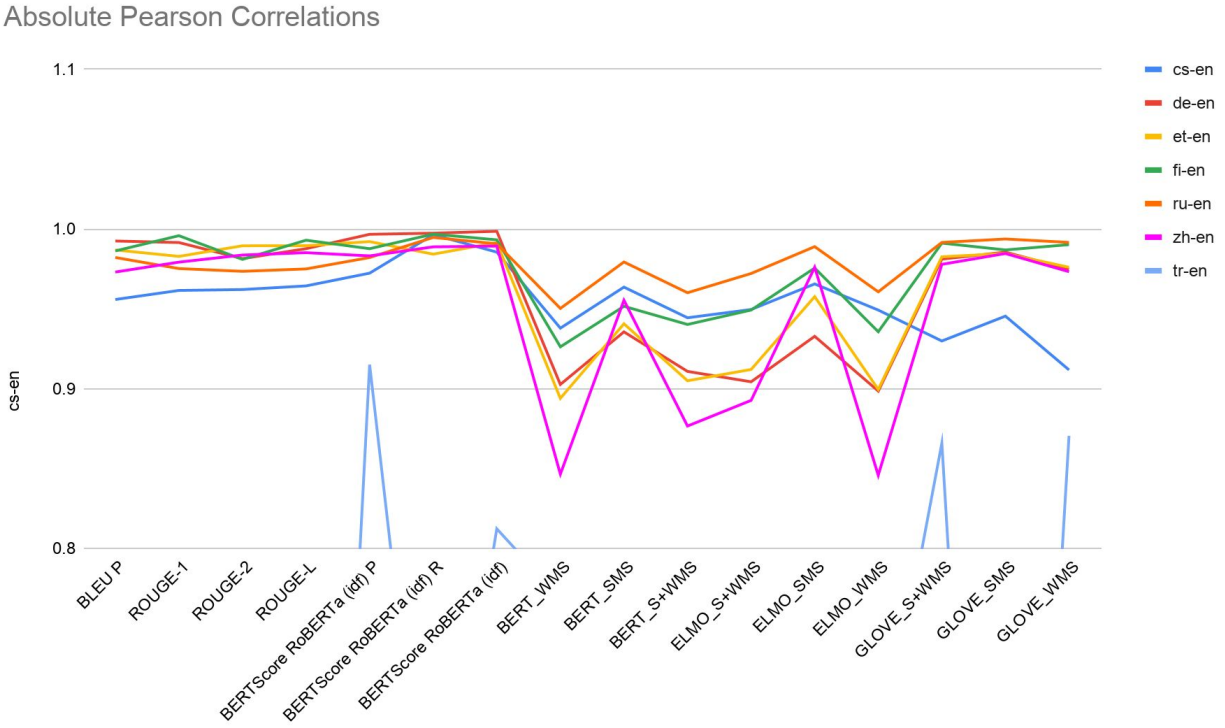
Authors of the paper used both glove (type based) and elmo (context based) embeddings.
- They showed these metrics' gain over ROUGE-L is consistent across word embedding types
- There is no significant difference between type-based and contextual embeddings.
- The metric significantly improves scores' correlation with human judgments, both on automatically generated and human-authored texts.

BERTScore



- Calculates cosine similarity between word embeddings, and considers the closest word from reference for a given candidate. We get a score for the sentence by taking a weighted average of all such word scores, weighting it with its importance in the reference corpus.
- The benefit of this score is that it gets contextualized embeddings for words from pretrained models, trained on text summarization tasks.
- It captures **paraphrases** of candidate sentences.
- It gives an importance weight to each word while taking a weighted average, because more frequently occurring words have less importance.
- It takes more compute, because we have to pass through a pre-trained model.

Comparison of scoring metrics on WMT18 system w.r.t human scores with Absolute Pearson Correlation



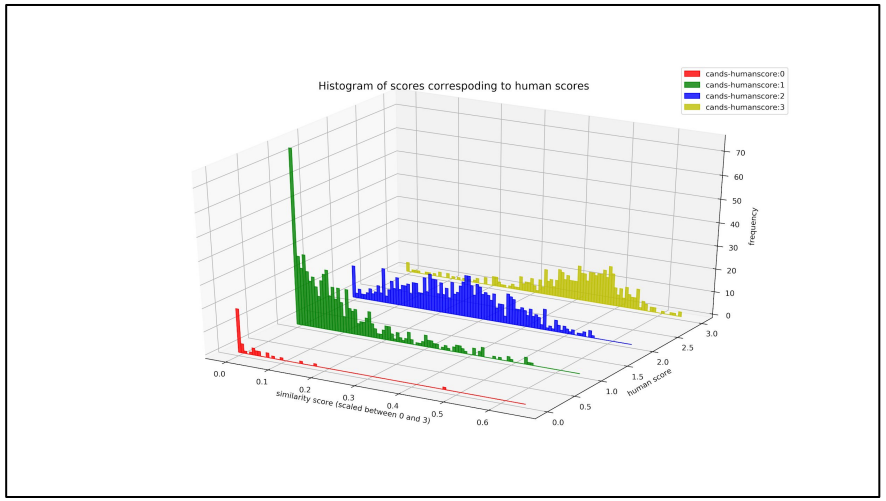
Novel Experiments

- Used BERT, RoBERTa-Large embeddings for WMS, SMS, S+WMS.
- RoBERTa - Large performed third best only beaten by BLEU and S+WMS (ELMo) in AES dataset.
- Used Sentence-BERT embeddings in SMS.
- Sentence-BERT gave the best result in SAS dataset.

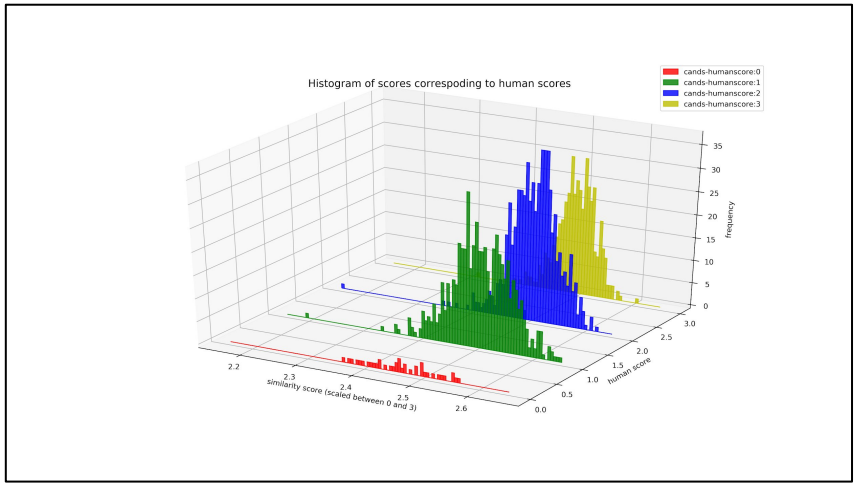
Implementation: https://github.com/EshwarSR/AutomaticEvaluationMetrics

Spearman Rank Correlations			
Score Metric	**CNN**	**AES**	**SAS**
BLEU	0.037	**0.615**	0.050
ROUGE-L	0.102	0.337	0.027
WMS (GloVe)	0.180	0.429	0.082
SMS (GloVe)	0.257	0.449	0.116
S+WMS (GloVe)	0.214	0.488	0.090
WMS (ELMo)	0.160	0.44	0.015
SMS (ELMo)	0.253	0.438	0.063
S+WMS (ELMo)	0.203	0.483	0.031
WMS (BERT)	0.169	0.368	0.034
SMS (BERT)	0.229	0.412	0.070
S+WMS (BERT)	0.199	0.414	0.048
WMS (RoBERTa Large)	0.163	0.304	0.088
SMS (RoBERTa Large)	0.244	0.463	0.081
S+WMS (RoBERTa Large)	0.200	0.460	0.103
BERTScore (P)	0.178	0.135	0.054
BERTScore (R)	**0.263**	0.536	0.113
BERTScore (F1)	0.255	0.147	0.088
SMS (Sentence-BERT)	0.115	0.297	**0.124**

Graphs showing score distribution for different human scores (AES Dataset)



BLEU Precision



BERTScore Precision

Conclusion

- A review of various evaluation metrics were carried out, and the performance of different embeddings for encoding words and sentences were verified. Embeddings from various pre-trained transformer models like BERT, Sentence-BERT and RoBERTa-Large were also used in the experiments. It was seen that these did not provide much improvement when compared to existing embeddings like GloVe.
- We reviewed metrics based on contextual embeddings and type embeddings. We observe that there is no significant difference between them.
- We see reasonably high correlations for metrics when compared to human scores. But, we cannot automate the process of evaluation of essays because we can see that the best correlated scores also show a variation in the scores assigned for the same human score.
- We observe that sentence based embeddings when incorporated give better correlations with human scores than word embeddings only, most of the time. Hence sentence based embeddings are in general more useful.
- We also use fundamentally different methods of finding similarity i.e. cosine similarity and earth mover distance. We use the same pre-trained model embedding for both and find cosine similarity is more correlated to human scores in all three datasets.

References (Papers)

1. Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. Bleu: A method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, ACL '02, pp. 311–318, USA, 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://doi.org/10.3115/1073083.1073135.
2. Kusner, M. J., Sun, Y., Kolkin, N. I., and Weinberger, K. Q. From word embeddings to document distances. In Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37,ICML'15, pp. 957–966. JMLR.org, 2015.
3. Clark, E., Celikyilmaz, A., and Smith, N. A. Sentence mover's similarity: Automatic evaluation for multi sentence texts. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 2748–2760, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/ v1/P19-1264. URL https://www.aclweb.org/anthology/P19-1264.
4. Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. Bertscore: Evaluating text generation with BERT. CoRR, abs/1904.09675, 2019. URL http://arxiv.org/abs/1904.09675.
5. Lin, C.-Y.ROUGE: A package for automatic evaluation of summaries. In Text Summarization Branches Out, pp. 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. https://www.aclweb.org/anthology/W04-1013.
6. Reimers, N. and Gurevych, I. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 3982–3992, Hong Kong,China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410. https://bit.ly/2UgkBmA.

References (Datasets and code)

7. ASAP AES dataset - https://www.kaggle.com/c/asap-aes.
8. ASAP SAS dataset - https://www.kaggle.com/c/asap-sas.
9. CNN/DailyMail dataset - https://bit.ly/price-of-debiasing.
10. WMT-18 - http://www.statmt.org/wmt18/index.html.
11. ROUGE Implementation - https://github.com/pltrdy/rouge/.
12. WMD Implementation - https://github.com/src-d/wmd-relax.
13. Sentence-BERT - https://github.com/UKPLab/sentence-transformers.
14. BERTScore Implementation - https://github.com/Tiiiger/bert_score/.

Thank you