A B. Tech Project Report Submitted
in Partial Fulfillment of the Requirements
for the Degree of

Bachelor of Technology

by

Aryan Chauhan Rishikesh Songra
((180101012) (180101065))

under the guidance of

Amit Chintamani Awekar

to the

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI
GUWAHATI - 781039, ASSAM

CERTIFICATE

This is to certify that the work contained in this thesis entitled "" is a bonafide work of ***Aryan Chauhan (Roll No 180101012)*** *,* ***Rishikesh Songra (Roll No 180101065)*** *carried out in the Department of Computer Science and Engineering, Indian Institute of Technology Guwahati under my supervision and that it has not been submitted elsewhere for a degree.*

Supervisor: **Dr. Amit Chintamani Awekar**

Assistant Professor,

Nov, 2022

Department of Computer Science & Engineering,

Guwahati.

Indian Institute of Technology Guwahati, Assam.

Acknowledgement

We'd like to take this opportunity to thank our supervisor, **Dr.Amit Chintamani Awekar, IIT Guwahati CSE** department for his constant support, patience, enthusiasm, and encouragement during our B.Tech project. We appreciate his valuable input, insights, and guidance throughout the assignment, which aided us in gaining a theoretical and practical understanding of the subject. We appreciate his useful feedback on each problem or difficulty we encountered with our project, and he was always willing to help.

Sincerely,

Aryan, Rishikesh

Abstract

Motivated by recent findings on the probabilistic modeling of acceptability judgments, we have proposed few language model scores ..... ,as a metric for reference-less grammar evaluation of natural language generation output at the sentence level.Using these scores we can harness a more compact language model potential. Our findings suggest that the current way of normalization of log-likelihood by the length of the sentences is not optimal. We show that ..... yields a significantly higher correlation with human judgments than all other LM scores.

Contents

List of Figures

Chapter 1

Introduction

Grammatical Error Correction as the name suggests is the process by which the detection and correction to an error in the text are done. The problem seems easy to understand but is actually tough due to the diverse vocabulary and set of rules in a language.

There are immense applications to this problem, the reason being writing is a very common means to share ideas and information. This could help the writer to speed up their work with very minimal chance of error. Moreover, there could be many individuals who are not fluent in a particular language. Therefore, these types of models make sure that language is not a barrier in communication.

1.1 Motive for New Scores

We have observed that the current research in NLP is more focused on improving the language model themselves which is crucial part but we believe that choosing a optimal score metric is also very important to harness the full potential of LM's. Current LM's score are not sufficient for every use case this we have emperically demonstrated in our work.

Specifically, we test our hypothesis that our score should be a suitable for evaluation of grammar which

- Does not rely on references(Sentences which are used as ground truth for the evaluation).
- Does not need human grammar annotations of any kind.

In particular the first aspect, i.e., our scores not needing references, makes it a promising candidate for automatic evaluation. Getting rid of human references has practical importance in a variety of settings, e.g., if references are unavailable due to a lack of resources for annotation, or if obtaining references is impracticable.

1.2 Contributions

To summerize we have made the following contributions

- We have build novel scores for evaluating the grammatical correctness of sentences which we have tested on the task of grammatical sorting. **Grammatically sorting - Sorting a list of sentences such that more grammatically correct sentences appear at the beginning of the list**

Chapter 2

Literature Survey

Numerous reference based metrics for GEC have been explorered by us, such as the F-score, precision, recall, MaxMatch, and GLUE.In NLP, well-known metrics for performance evaluation are the F-score, precision, and recall. The F-score was the most widely used metric in the initial GEC research, and it uses the harmonic mean of precision and recall as the nal performance value. However, these metrics exhibit the weakness whereby they cannot evaluate sentences that exceed the phrase level.

Moreover, the F-score cannot capture the difference between "no change" and "wrong edits" of the GEC model. To alleviate the limitations of traditional methods, Dahlmeier and Ng [9] suggested a metric known as the MaxMatch scorer, that could consider edits up to the phrase level.

However, MaxMatch requires annotations for individual errors.The limitations with these metrics are that they are referenced based metrics and they will not work without having ground truth sentence.

With the advent of deep learning, GLUE [21] and the bilingual evaluation understudy (BLEU) [22] were mainly used as the GEC metric. BLEU is a metric for MT that compares MT and human translation results. The measurement criteria are based on the n-gram. This metric can be used regardless of language and offers the advantage of rapid calculation.

As with BLEU, GLUE, whichwas proposed by Napoles, only requires human annotators to correct a sentence by rewriting the source sentence. The difference with GLEU is that it considers the source sentence and it is a performance evaluation metric that is specialized for the correction system. The majority of current research uses this metric as the official metric of GEC [1], [6][8], [13], [16], [17], [27].

In today's world with new progressions in Deep Learning based Language Models, LM's performance has greatly improved, now LM's can generate score which can evaluate sentences without the need of any reference/annotation.

Therefore LM's are perfect candidates for automating NLP based task like grammar error correction.In our experiments we have used GPT-2 variations.

Perplexity (PPL) is one of the most common metrics for evaluating language models. we should note that the metric applies specifically to classical language models (sometimes called auto regressive or causal language models) and is not well defined for masked language models like BERT.

Perplexity has as advantage over other metrics that it is a reference-less metric and does not require human based annotations, so we have used perplexity as a baseline comparison.

We have evaluated our metrics on the task of grammatical sorting. Calculating the number of inversions is the classical way of determining how much sorted an array is, by taking inspiration from this fact we have defined something similar to inversions in our work.

Chapter 3

Proposed Metrics

In this section, we first describe **MCP(Minimum Contextual Probability)**, **WSNLL(Weighted Sum of Negative log-likelihoods)** and **KPPL(Kth-perplexity)** and look at the intuition behind these metrics/scores.

We have tried two approaches to compute the contextual probabilities(defined in Section 3.1) vector

- We first remove the words which had contextual probability below a certain threshold and then for the rest of the words recalculated the contextual probabilities.
- In the second approach we have included all the words while calculating contextual probabilities.

3.1 Minimum Contextual Probability

Given a sentence X tokenized as [x0,....xn-1] , MCP is defined as the minimum of the contextual probabilites of all the tokens in the sentence.

$$MCP = min(p_\Theta(x_i|x_{<i})) \quad \forall \quad i \quad \epsilon \{1, 2..., n\}$$

where $p_\Theta(x_i|x_{i-1})$ is the contextual probabilty of the ith token conditioned on the preceding tokens $x_{<i}$.

The intuition behind this score is that the token with minimum probability will denote the most unlikely word in the sentence,this word is the most out of context word in the sentence. Hence we expect a sentence with lower minimum probability score will be more grammatically incorrect with respect to a sentence having higher minimum probability.

3.2 Weighted Sum of negative log-likelihoods

Like MCP here we take the contextual probability vector and construct the negative log-likelihood vector(NLLV) corresponding to it.Then we sort this NLLV in desecending order.Finally we take a heuristical weight vector where ith term is $\alpha^i(\alpha < 1)$ using this weight vector and sorted NLLV we output their dot product as the score.

$$WSNLL = \sum_{i=1}^{n-1} - \ log(\ p'_{\Theta}(x_i|x_{<i})\ )\ *\ \alpha^i$$

The intuition behind this score is that for grammar correctness the word having the least probability should contribute more to the score. Hence the weight are less for more more in-context words.

3.3 Kth-perplexity

This score is almost identical to perplexity except that here we divide by n^k rather than n.

$$\log(KPPL) = \sum_{i=1}^{n-1} \frac{-\ log(\ p'_{\Theta}(x_i|x_{<i})\ )}{n^k}$$

The intuition is that say a sentence of length 20 has 2 out of context words and another sentence of length 10 has only 1 out of context word. Then perplexity (or 1PPL) for both the sentences would be similar but according to intuition the first sentence of length 20 is bad when it comes to the grammatical correctness as it has two errors as compared to the sentence of length 10. Therefore, some power $k < 1$ would be a better measure when it comes to the grammatical correctness of the sentence.

Chapter 4

Observations and Experimental Results

4.1 Dataset Construction

We experimented on the CONLL-13 dataset.CONLL-13 includes a more comprehensive list of error types, including determiner, preposition, noun number, verb form, and subject-verb agreement errors. Extending into more error types introduces the possibility of correcting multiple interacting errors. Examples of such interacting errors include determiner and noun number ('that cars' → 'that car' or 'those cars') and preposition and verb form ('an interest to study' → 'an interest in studying'). The above dataset is first converted to a list of pairs. Where every pair contains a sentence and its label. The label is a boolean value denoting if the sentence is correct or not. Our dataset comprises of approximately 1400 sentences of which 260 of grammatically correct.

4.2 Optimal Parameters Selection

We first divided the sentences dataset into two parts namely validation and testing. 400 sentences for validation and 981 for testing. We have used grid search for finding the optimal parameters using the validation dataset. Then we have tested these parameters on the testing dataset.

4.3 Pre-Trained Language Model

try to do bert as well.

We have used the following pre-trained LMs from the transformers library :

1. GPT2 Small
2. GPT2 Medium
3. GPT2 Large
4. OpenAI-GPT
5. GPT-Neo
6. Bert

We have used library's default Hyper-parameters.

4.4 Baseline Metrics

We are comparing MCP, WSNLL and KPPL Metrics with Perplexity and BLEU as baseline metrics.

1. Perplexity : Our first baseline is perplexity,which is commanly used for

evaulating LM's,which corresponds to the exponentiated crossentropy:

$$\log(PPL) = \sum_{i=1}^{n-1} \frac{-\ log(\ p'_{\Theta}(x_i|x_{<i})\ )}{n}$$

2. BLEU : We wanted to compare our score with some reference based baseline, Hence as BLEU is a comman reference based metric so we have used it in our comparision.

4.5 Correlation and Evaluation Scores

We have evaluted the scores on the task of grammatical sorting as defined previously using the following metrics.

1. **TOP k-Error** : It is the number of grammatically correct sentences among the sentences having top k score,here higher score means that the sentence is more grammatically incorrect.

2. **Number Of Inversion** : An inversion is a pair (i,j) such that i¡j and ith sentence is grammatically incorrect while the jth sentence is grammatically correct.

3. **Bottom k-Error** : It is the number of grammatically incorrect sentences among the sentences having least k scores.

4.6 Results and Discussion

Results					
Models	Inversion	Top 50 error rate (%)	Top 100 error rate	Top 150 error rate	Top 200 error rate
Perplexity	100288	2	6	8.6	10.5
WSNLL	53258	0	1	0.66	0.5
MCP	85955	10	7	7.3	8
0.4th-PPL	51537	0	0	1.33	1

Results					
Models	Inversion	Top 50 error rate (%)	Top 100 error rate	Top 150 error rate	Top 200 error rate
Perplexity	100288	2	6	8.6	10.5
WSNLL	53258	0	1	0.66	0.5
MCP	85955	10	7	7.3	8
0.4th-PPL	51537	0	0	1.33	1

4.7 Discussion

As from above result we can observe

- We can observe that MCP result were comparable to perplexity and BLEU. Here Inversion count,k-errors(Top-K and Bottom-k errors) were comparable to perplexity and BLEU.We believe inversion are less because MCP handles length based normalization of sentences better than baseline metrics, but the results are not significantly better because MCP discards a lot of information as it considers only one token.

- Regarding WSNLL and KPPL we have significantly better result than baseline metrics.Here Inversion count, k-errors are significantly better.Thus KPPL and WSNLL handle normalization and contextual information very well which results in low inversions and k-errors.

- We have observed that removing bad words approach leads to degraded performance.The reason why the performance degrades after removing the bad words is that our initial hypothesis of having the rest of the part wrong after encountering a bad word is not correct, as it is evident from the NLL vector. Eg.
  - Sentence - Some people started to think if electronic products can be further operated to more advanced utilization and replace human beings for better performances.
  - NLL's - [6.1, 2.8, 6.8, 1.5, 3.0, 5.8, 12.2, 5.8, 3.0, 0.9, 9.5, 11.4, 4.7, 5.8, 3.7, 10.6, 2.6, 7.8, 3.3, 2.0, 4.2, 5.8, 6.3, 7.5].
  - Here we have kept cutoff to 14.

Chapter 5

Conclusion and Future works

5.1 Conclusion

From above experiments we can conclude the following :

- We have discovered powerful non-referential language model metric like WSNLL,kPPL.
- We empirically confirmed the effectiveness of kth-perplexity and LWSNLL, LM score which better accounts for the effects of sentence length and individual unigram probabilities, as a metric for grammatical correctness of sentences.
- The normalization of the LM score by the length of the sentence is not optimal for all task as it is evident from the above results for the task of grammatical sorting.

- These Metrics better harness the power of LM's, which reduced both model size and training time at a similar evaluation performance.

5.2 Limitations

There were some following limitations to our model

- These Metrics does not perform effectively when the sentences contains non-frequent words as the contextual probability of these words are very low.
- These Metrics only on the prefix context of the sentence and does not take into account the suffix of the sentence.

5.3 Future Work

There were some future

- We would like to make the metrics to take into account non-frequent words.
- Try to work on different languages, currently our dataset contains only English sentences.
- Currently these metrics haven't been tested on the degree of error which we plan to do in future.

References