Improving ASR Output Using a Transformer-based Grammatical Error Correction Approach

Rachel Gao, Juliana Gómez-Consuegra, Erica Nakabayashi
University of California, Berkeley
{rachelgao, julianagc, ericanaka}@berkeley.edu

Abstract

In this work, we introduce a transformer-based encoder-only grammatical error correction approach for improving automatic speech recognition by utilizing both a grammatical acceptability classifier (GAC) and a grammatical error correction model (GEC). We investigate different strategies for optimizing the models and show that using raw data to train our GAC model generates better outputs than using cleaned and augmented data. We also find that the model which punctuates and propercases the input data by means of Named Entity Recognition (NER) yields better results than other GEC models, leading to reduced over-correction. Considering phonetics also improved model performance, and the performance differs between gender and emotions.

1 Introduction

Error correction from Automatic Speech Recognition (ASR) is challenging due to the diversity of speaking styles and inflections that occur when speaking. While conventional ASR transcription error correction has predominantly concentrated on neutral-toned speeches, emotions play a prevalent role in everyday discourse (Adigwe et al., 2018). Despite English being a high-resource language, with nearly two billion speakers worldwide (Gordon, 2005), most of them are non-native speakers, adding an additional layer of complexity in dealing with diverse phonetics arising from various accents. Incorrectly transcribed output from ASR not only introduces noise into downstream NLP tasks but also disrupts the reading experience. Many readers rely on ASR-generated outputs, particularly when accessing video and streaming platforms. However, if the transcribed text contains grammatical errors, the readers may face challenges comprehending the speaker's intended message based on the provided transcription.

While current ASR systems achieve high levels of accuracy, producing a perfect transcription remains elusive due to spelling errors, disfluency, and grammatical errors (Dutta et al., 2022). One commonly used ASR model is wav2vec 2.0 (Baevski et al., 2020), which uses self-supervised learning followed by fine-tuning on transcribed speech from Librispeech (voice recordings of out-of-copyright books). Despite wav2vec 2.0 achieving close to the state-of-the-art performance, its fine-tuning task outputs characters sequentially and spells words phonetically, leading to misspelled words and improper spacing. Further, it does not consider context, resulting in grammatically incorrect outputs.

Traditional text-based grammatical error correction employs an encoder-based approach (Omelianchuk et al., 2020) by using grammatically annotated texts, while error correction for ASR relies on an encoder-decoder-based seq-to-seq approach (Dutta et al., 2022) without annotated text. Both methods have inherent limitations: grammatically annotated texts are scarce, costly and hard to generate (Liao et al., 2021), while the encoder-decoder architecture required in traditional error correction for ASR is computationally expensive and time-consuming. Our contribution is an encoder-only ASR error correction approach which considers phonetics, spelling, punctuation, casing, and NER, reducing the need for grammatically annotated texts, by capitalizing on recent advancements in transformer-based Large Language Models (LLMs) that exhibit contextual awareness and some robustness towards grammar (Yin et al., 2020; Vaswani et al., 2017).

2 Background

Traditional methods for ASR error correction have focused on speech-to-text error correction within the ASR system. Common strategies include utilizing N-best hypotheses with beam search (Zhu et al., 2021) or treating error correction as a machine translation task, leveraging seq-to-seq models like BART (Dutta et al., 2022). This second

approach necessitates an encoder-decoder architecture to convert speech input into text output, incurring in potentially high training costs, as well as making it unsuitable for downstream tasks where the autotranscription has already been generated but requires correction.

In contrast, text-based grammatical error correction has favored an encoder-only architecture by training models on essays written by English language learners, annotated with different types of grammatical errors. The CoNLL-2014 shared task on grammatical error correction (Ng et al., 2014) and the BEA-2019 shared task on grammatical error correction (Bryant et al., 2019) focused on this approach. Some common strategies include the N-gram-based method (Bryant and Briscoe, 2018), with recent advancements veering towards a language-model-based approach (Alikaniotis and Raheja, 2019) that circumvents the need for grammatically annotated data.

3 Methods

Our encoder-only ASR error correction approach (appendix figure 1) consisted of a GAC model (Warstadt et al., 2019) and a GEC model (Bryant and Briscoe, 2018; Alikaniotis and Raheja, 2019) . The GAC model output the grammatical probability of a sentence. Sentences were then either passed into the GEC, or not, based on a grammatical threshold (table 2). The GEC model corrected the errors in the given sentence, with the original sentence being replaced by the corrected sentence if the grammatical probability increased by a certain threshold (improvement threshold, appendix table 2) as a result of grammatical error correction.

3.1 Grammatical acceptability classifier

Inspired by (Yin et al., 2020), we selected BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) as the LLMs for our grammatical acceptability fine-tuning task with the CoLA dataset (Warstadt et al., 2019). We experimented with BERT-base-uncased, BERT-large-cased, RoBERTa-base, and RoBERTa-large. For training the GAC model, we experimented with using the raw CoLA dataset, and with the cleaned CoLA dataset, where we capitalized and inserted missing punctuation and applied data augmentation using the Python package nlpaug (Ma, 2019) to create more ungrammatical sentences to balance the dataset. During our experimentation, we found that using the pooling average of last hidden states on RoBERTa-large yielded better results than any of the other specifications, consistent with findings from (Mosbach et al., 2020), so our results will focus on that model.

We experimented with unfreezing different numbers of layers (0,3,6,12,24) and using different batch sizes (4,8), while holding the following hyperparameters constant: max token length (512), output state (pooling average of the last hidden state), training time (10 epochs), size of the fully-connected layer (1024) and dropout rate (0.1). We used the TensorFlow Polynomial Decay learning rate schedule with an initial learning rate of 1e-5 that decayed over 5336 steps to a final learning rate of 1e-10 with a power of 1.0. We also experimented with changing the grammatical threshold from the default 0.5 to 0.75.

3.2 Grammatical error correction

We experimented with four GEC models: SimpleGEC, PretrainedGEC, PhoneticGEC and RawGEC (appendix figure 2). The latter three models were designed based on weaknesses exposed by the SimpleGEC model (table 5). All models used the same dataset, but the data cleaning technique for the RawGEC differed from the other three models. For all models, we combined data from the EmoV_DB (Adigwe et al., 2018) and the Arctic (Kominek and Black, 2004) datasets, in order to have access to a wide range of speaking styles from both genders and emotions. Due to computational restrictions, the data was down-sampled by deleting all repeated instances of an utterance (sentence), ending up with a train set of 906 sentences and a test set of 227 sentences. The speech data was autotranscribed by using wav2vec 2.0, and cleaned by removing duplicates and by removing instances where the autotranscription did not align with the label due to errors in the original dataset.

For all models other than RawGEC, the autotranscription was cleaned by using the punctuation insertion model trained by (Alam et al., 2020) (appendix table 2, reduced_clean, appendix figure 4) to restore the punctuation and casing. The cleaned autotranscription was passed to GEC for error correction only if the grammatical probability (from GAC) was below a certain threshold (grammatical threshold). For RawGEC, we cleaned the autotranscription (appendix table 2, reduced_raw) by

Error Type	Input Sentence	SimpleGEC Suggestion
Punctuation	Instead, he joined her. And they ate like two hungry children.	Instead, he joined to her anthey i do argree children.
NER	He waded into the edge of the water	Smith waded into the edge of the water
Phonetics	He waded into the edge of the water	He wated in the edge of the water

Table 1: Error handling approaches of our SimpleGEC model. Input Sentence = original sentence fed into the model. SimpleGEC Suggestion = sentence suggested by the model.

capitalizing the first letter, adding a period to the end of the sentence, proper casing the word 'I', and proper casing for nouns based on NER, using Spacy (Honnibal and Montani, 2017) before passing it to the model. The cleaned autotranscription was passed to RawGEC for error correction regardless of its initial grammatical probability (appendix figure 3).

For our baseline GEC model (SimpleGEC), we used an algorithm similar to that of (Bryant and Briscoe, 2018) and (Alikaniotis and Raheja, 2019), consisting of the following steps:

1. Iteratively mask each word in the sentence.
2. Use TFRoBERTaforMaskedLM with k-beam = 3 to predict the masked word.
3. Use GAC model to calculate the grammatical probability of the predicted sentences (where the masked word is replaced with the predicted word).
4. Replace the original sentence with predicted sentence if the grammatical probability has increased by the improvement threshold (0.25 or 0.1).
5. Repeat steps 1-4 until the final predicted sentence reaches the pre-defined grammatical threshold (0.5, 0.75 or 0.9), or until three iterations have occurred with no improvement.

Compared to SimpleGEC, all future models (PretrainedGEC, PhoneticGEC, and RawGEC) use the pre-trained MaskedLM where we pre-trained TFRoBERTaforMaskedLM with our dataset. For pre-training, we selected grammatically acceptable sentences from the CoLA dataset, and merged them with our EmoV_DB + Arctic dataset, removing all duplicates, and replicating each sentence five times in order increase the size of our dataset for dynamic masking. We experimented with RoBERTa-base and RoBERTa-large for pre-training TFRoBERTaforMaskedLM on our dataset, with dynamic masking, similar to how RoBERTa was initially trained (Liu et al., 2019). We used the same hyperparameters as our GAC model and we applied early stopping. As with our GAC model, we found that RoBERTa-large outperformed others, so we will focus on this model only.

For both the PhoneticGEC and the RawGEC, the number of k-beams was increased from 3 to 20. Additionally, step 2 of the SimpleGEC algorithm was modified in the PhoneticGEC, as follows: a phonetics dictionary was generated for all words from the NLTK words dictionary (Bird et al., 2009), with Soundex and Metaphone phonetics. A phonetic confusion set was then created by querying the phonetics dictionary based on the phonetics of the masked word. The masked word was then replaced with words that occurred in both the MaskedLM k-beam output and the phonetics confusion set to create candidate sentences. For RawGEC, however, the phonetics dictionary was eliminated, and the phonetics of the MaskedLM k-beam output was generated at run-time, and only words that matched the phonetics of the masked word are retained. We also experimented with fuzzy phonetics match based on Levenshtein distance of the phonetics between the masked word and the predicted words. However, when we increased the Levenshtein distance, the model tended to overcorrect, so we discarded the fuzzy phonetic and focused on only accepting exact matches between the masked word and the predicted words.

3.3 Effects of gender and emotion

To evaluate the correlation between grammatical error correction and emotions and gender, we ran a subset of sentences from our original training set through our best-performing RawGEC model. The gender data subset was created by randomly selecting 95 utterances per gender with the same emotion (neutral) from the original training set. The emotion data subset was created by randomly selecting 83 utterances per emotion (amused, dis-

gusted, sleepy, neutral, angry) with the same gender (female).

3.4 Evaluation metrics

We evaluated the results for all our models using WER (Chen et al., 1998), BLEU (Papineni et al., 2002), GLEU (Mutton et al., 2007), and BERTScore (Zhang et al., 2019) (precision, recall and F1). We did so by evaluating transcription and label as sentence pairs, and then obtaining the average score for all sentences. To avoid inflating error rates due to punctuation and casing, when calculating evaluation metrics, we cast all auto transcriptions to lower case and added a period to all labels which did not have any punctuation at the end of the sentence. We were also interested in how these results compared to those when we removed punctuation and lower-cased all labels, autotranscriptions and corrected transcriptions (appendix, tables 3 and 4).

4 Results and Discussion

4.1 Grammatical acceptability classifier

In the raw CoLA dataset, 68% of the sentences are grammatically correct, this majority class serves as our baseline. Our goal was to not only beat the baseline but to also surpass the MCC of 0.6780 reported by (Liu et al., 2019).

During our experimentation, we found that the model performed best (table 2, model e) with the original uncleaned unbalanced CoLA dataset. It is possible that our initial aim of cleaning and balancing the dataset to enhance performance might have inadvertently altered the grammatical correctness of sentences during the cleaning and augmentation process, hindering model performance as a result. Increasing the grammatical acceptability threshold to 0.75 improved performance and reduced the percentage of false negatives, which is why we decided to experiment with thresholds of 0.5, 0.75 and 0.9 in our grammatical error correction models.

We ran inference on the CoLA dev set (both in and out of domain), by setting a grammatical acceptability threshold of 0.5 and found that in our best performing model (model e, table 2) precision was comparable for both the inputs labelled as grammatically incorrect and grammatically correct (table 3), but recall and F1-score were much higher in the inputs labelled as grammatically correct. This suggests that our model was better at identifying grammatically correct sentences, which may be due to class imbalance in the training data.

Model	Unfrozen layers	Batch size	Validation accuracy	MCC
Baseline	N/A	N/A	68.00%	unknown
RoBERTa paper	24	16	unknown	0.6780
a	0	8	70.47%	0.1733
b	3	8	83.51%	0.5978
c	6	8	83.80%	0.6052
d	12	8	84.56%	0.6249
e	**24**	**4**	**87.06%**	**0.6884**
f	12	4	84.28%	0.6177

Table 2: Best scoring GAC models trained on the CoLA dataset, and using RoBERTa large. Baseline = majority class (grammatically acceptable). (Liu et al., 2019).

Class	precision	recall	f1-score	support
0	0.86	0.69	0.77	324
1	0.87	0.95	0.91	719

Table 3: Evaluation metrics per class, for GAC models. Class 0: grammatically incorrect. Class 1: grammatically correct.

By utilizing the CoLA major and minor grammatical annotations on the falsely classified sentences, we found that the most common types of grammatical error, for sentences that are grammatically unacceptable but predicted to be acceptable (false positives) were related to syntactic constructions, questions and auxiliary verbs, while the most common types of grammatical error for sentences that are grammatically acceptable but predicted to be unacceptable (false negatives) were related to auxiliary verbs, alternate arguments and syntactic constructions (appendix table 1).

4.2 Grammatical error correction

The best scoring RawGEC model (tables 5 and 6) outperformed all other GEC models (table 4), including the RawGEC model where the autotranscription, labels and GEC transcriptions were lower-cased and punctuation was removed. In the latter dataset, the autotranscribed sentences obtained better evaluation metrics than the candidate sentences produced by any of our models. This was true under most metrics, excepting BERTScore prediction and BERTScoreF1, where the PhoneticGEC (0.9 grammatical threshold, 0.1 improvement threshold) outperformed all other models. This improved performance of the PhoneticGEC

Metric	Autotranscription	SimpleGEC	PretrainedGEC	PhoneticGEC
WER	0.3760	0.1892	0.1877	0.1843
BLEU	0.4833	0.6578	0.6607	0.6663
GLEU	0.5406	0.7133	0.7153	0.7190
Precision	0.9573	0.9682	0.9687	0.9690
Recall	0.9625	0.9701	0.9705	0.9719
F1	0.9598	0.9691	0.9695	0.9704

Table 4: Evaluation metrics for models where only sentences with levels of grammatical acceptability lower than a grammatical threshold of 0.75 were passed through the model, and whose improvement threshold was 0.25. Values from autotranscription are the scores for the reduced clean dataset. Precision, recall and F1 values were calculated with the BERTScore.

Metric	Autotranscription	Punctuated RawGEC: Train	Punctuated RawGEC: Inference
WER	0.3918	0.1563	0.1636
BLEU	0.4713	0.7039	0.6813
GLEU	0.5318	0.7518	0.7423
Precision	0.9580	0.9727	0.9731
Recall	0.9631	0.9742	0.9746
F1	0.9605	0.9734	0.9738

Table 5: Evaluation metrics for RawGEC model under the best scoring improvement threshold of 0.25. Values from autotranscription represent the scores for the reduced raw dataset. Precision, recall and F1 values were calculated with the BERTScore.

under these thresholds may be due to the reduced subset of candidate words produced by the intersection of MaskedLM and the phonetic dictionary confusion sets.

Differences between the models were more prominent with WER, BLEU, and GLEU, but less so with BERTScore metrics (precision, recall, and F1), given that all models performed relatively well on all three BERTScore metrics. The standard deviation for all models in BERTScore precision, recall and F1 was 0.0049, 0.0045, and 0.0046, respectively. Given that these three metrics are the best in terms of evaluating context, this suggests that they are the most reliable metrics in terms of grammatical errors, since BERTScore computes cosine similarity between contextual embeddings. This is to be expected, as the RoBERTa-large embeddings yielded the best results when evaluating BERTScores in English (Zhang et al., 2019).

The reasons for the differences in model performance when evaluating them with the other metrics may be due to limitations of the metrics. WER ignores the importance of word order, which is particularly relevant in grammar. Moreover, certain words exhibit multiple acceptable spellings (e.g., "doughnut" vs. "donut"), both of which are correct and convey identical meanings. BLEU does not take intelligibility or grammatical correctness into account (Amin and Ragha, 2021), therefore a high BLEU score does not guarantee grammatically correctness in the candidate sentence. It also has some limitations when evaluating single sentences, as it is designed to evaluate a full corpus (Papineni et al., 2002). We evaluated GLEU as well to counteract this limitation, and given that the results for all models were similar when evaluating BLEU and GLEU, the limitations of BLEU being an evaluation metric focused on a whole corpus does not seem to be an issue in this task.

4.3 Effects of gender and emotion

We observed differences in error correction by gender, using our top-performing RawGEC model. The model exhibited more accurate corrections in terms of insertions, deletions and substitutions for utterances spoken by female actors compared to those by their male counterparts (see WER, BLEU and GLEU in appendix table 5). However, when analyzing BERTScore metrics (appendix table 5), there was a marginal difference, favouring male

All High	
Clean Label	Jeanne was turning the bow shoreward.
SimpleGEC	**Jean** was turning the bow shoreward.
RawGEC	**Jean** was turning the bow shoreward.
High RawGEC	
Clean Label	Darkness hid him from Jeanne.
SimpleGEC	**i** hid him from **jean**.
RawGEC	**Darkness** hid him from **Jean**.
All low	
Clean Label	Thus he turned the tenets and jargon of psychology back on me.
SimpleGEC	**Thus** he turned the **tenits** and **jargon** of psychology back on me.
RawGEC	**Thus** he turned the **tenits** and **Jargon** of psychology back on me.
Only high RawGEC	
Clean Label	The Resident Commissioner is away in **Australia**.
SimpleGEC	The **resident** is away in **australia**.
RawGEC	The **resident commissioners** away in **Australia**.

Table 6: Candidate output sentences from all GEC models during training, with improvement threshold of 0.25 and grammatical threshold of 0.75 (except for RawGEC, where the grammatical threshold was 0). All high = high F1 improvement compared to autotranscription, for all GEC models. High RawGEC = high F1 improvement compared to autotranscription, for RawGEC only. All low = low F1 improvement compared to autotranscription, for all GEC models. All low, higher RawGEC = low F1 improvement compared to autotranscription, higher improvement for RawGEC.

utterances. This discrepancy may stem from biases related to differences in pitch levels between females and males, but a comprehensive analysis is essential for drawing meaningful conclusions from this outcome. We also noted differences in performance between different emotions (appendix table 6), with the model achieving its peak performance for utterances conveyed with a "neutral" emotion and the least favorable results for those spoken with an "amused" emotion. While most ASR systems are trained with predominantly "neutral" toned voices, it is crucial to acknowledge the expression of different emotions in daily speech. Further research is required to assess the robustness of both ASR and our error correction model in handling various emotions.

5 Conclusion

In our work, we utilized a grammatical acceptability classifier (GAC) and a grammatical error correction model (GEC) with an encoder-only approach, reducing the need for annotated grammatical text, to correct for errors from automatic speech recognition. We found that the RawGEC model outperformed other models in terms of WER, BLEU and GLEU scores, but was only marginally better in terms of BERTScores, suggesting that our best performing model is better at general error correction than the other models, but as good as the others at grammatical error correction. Using our best-performing RawGEC on the reduced dataset in terms of gender and emotions showed sentences uttered by female speakers were easier to correct for insertions, deletions and substitutions, and neutral utterances were easier than other emotions, especially amusement. One of the limitations of our model is that some of the labels used in training were grammatically incorrect, giving rise to over-correction, where the autotranscription was corrected for grammar when the label was not. Another limitation is that our dataset only includes data from native speakers. Our next steps include efforts to make our model more robust to different accents, emotions, and gender, by performing adversarial training, using the Multi-speaker Corpora of the English Accents in the British Isles. (Demirsahin et al., 2020).

Acknowledgements

The authors want to thank Mark Butler and Peter Grabowski for their continued guidance and support.

References

Adaeze Adigwe, Noé Tits, Kevin El Haddad, Sarah Ostadabbas, and Thierry Dutoit. 2018. The emotional voices database: Towards controlling the emotion dimension in voice generation systems. *arXiv preprint arXiv:1806.09514*.

Tanvirul Alam, Akib Khan, and Firoj Alam. 2020. Punctuation restoration using transformer models for high- and low-resource languages. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, pages 132–142, Online. Association for Computational Linguistics.

Dimitris Alikaniotis and Vipul Raheja. 2019. The unreasonable effectiveness of transformer language models in grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 127–133, Florence, Italy. Association for Computational Linguistics.

Sujit S Amin and Lata Ragha. 2021. Text generation and enhanced evaluation of metric for machine translation. In *Data Intelligence and Cognitive Informatics: Proceedings of ICDICI 2020*, pages 1–17. Springer.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Christopher Bryant and Ted Briscoe. 2018. Language model based grammatical error correction without annotated training data. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 247–253, New Orleans, Louisiana. Association for Computational Linguistics.

Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. 2019. The BEA-2019 shared task on grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–75, Florence, Italy. Association for Computational Linguistics.

Stanley F Chen, Douglas Beeferman, and Roni Rosenfeld. 1998. Evaluation metrics for language models.

Isin Demirsahin, Oddur Kjartansson, Alexander Gutkin, and Clara Rivera. 2020. Open-source multi-speaker corpora of the English accents in the British isles. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6532–6541, Marseille, France. European Language Resources Association.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Samrat Dutta, Shreyansh Jain, Ayush Maheshwari, Souvik Pal, Ganesh Ramakrishnan, and Preethi Jyothi. 2022. Error correction in asr using sequence-to-sequence models. *arXiv preprint arXiv:2202.01157*.

Raymond G. Gordon. 2005. *Ethnologue: Languages of the world*. SIL International.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing.

John Kominek and Alan W Black. 2004. The cmu arctic speech databases. In *Fifth ISCA workshop on speech synthesis*.

Junwei Liao, Yu Shi, Ming Gong, Linjun Shou, Sefik Eskimez, Liyang Lu, Hong Qu, and Michael Zeng. 2021. Generating human readable transcript for automatic speech recognition with pre-trained language model. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7578–7582. IEEE.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Edward Ma. 2019. Nlp augmentation. https://github.com/makcedward/nlpaug.

Marius Mosbach, Anna Khokhlova, Michael A Hedderich, and Dietrich Klakow. 2020. On the interplay between fine-tuning and sentence-level probing for linguistic knowledge in pre-trained transformers. *arXiv preprint arXiv:2010.02616*.

Andrew Mutton, Mark Dras, Stephen Wan, and Robert Dale. 2007. GLEU: Automatic evaluation of sentence-level fluency. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 344–351, Prague, Czech Republic. Association for Computational Linguistics.

Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. 2014. The CoNLL-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14, Baltimore, Maryland. Association for Computational Linguistics.

Kostiantyn Omelianchuk, Vitaliy Atrasevych, Artem Chernodub, and Oleksandr Skurzhanskyi. 2020. Gector–grammatical error correction: tag, not rewrite. *arXiv preprint arXiv:2005.12592*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Fan Yin, Quanyu Long, Tao Meng, and Kai-Wei Chang. 2020. On the robustness of language encoders against grammatical errors. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3386–3403, Online. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Linchen Zhu, Wenjie Liu, Linquan Liu, and Edward Lin. 2021. Improving asr error correction using n-best hypotheses. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 83–89. IEEE.

A Appendix

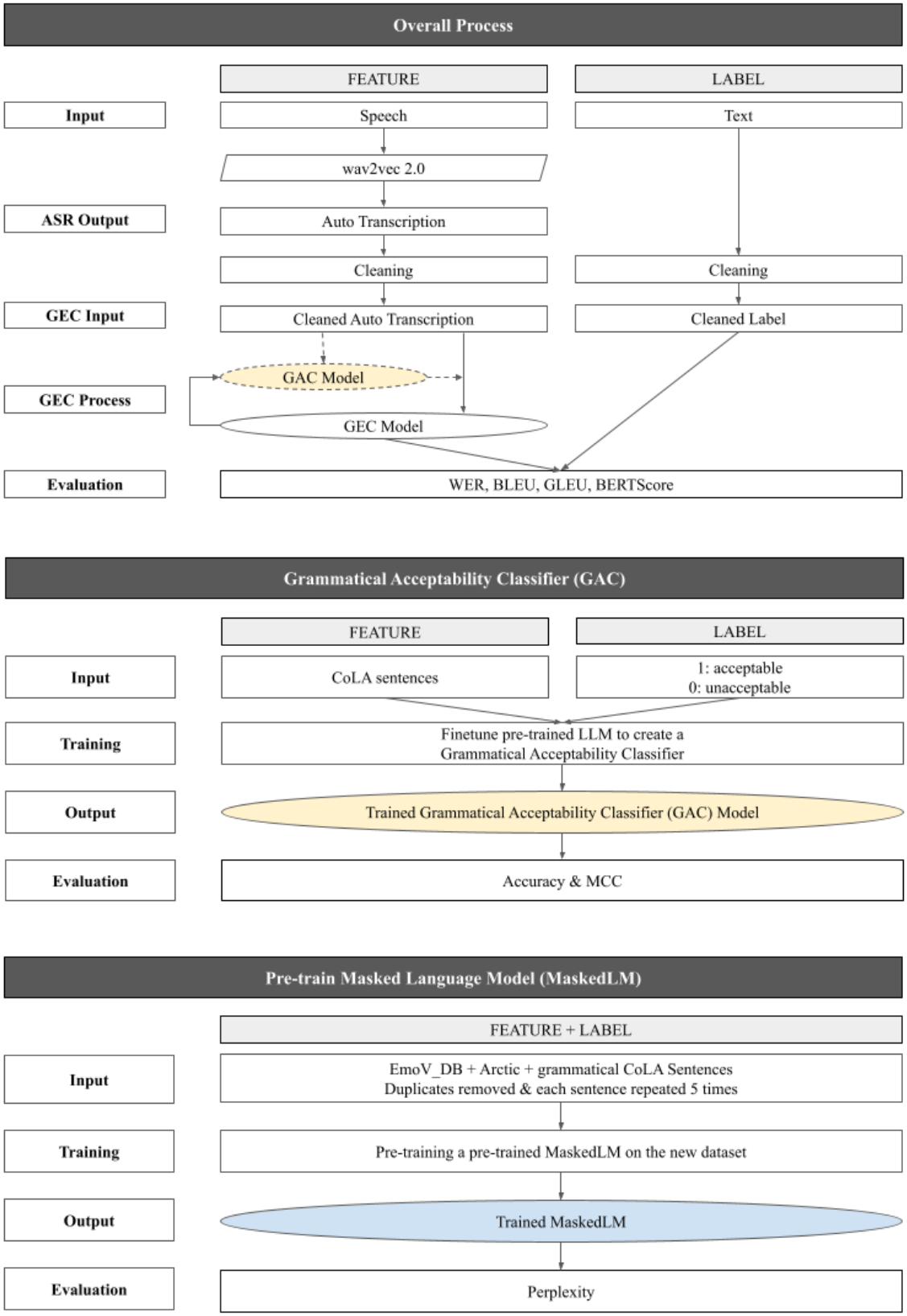
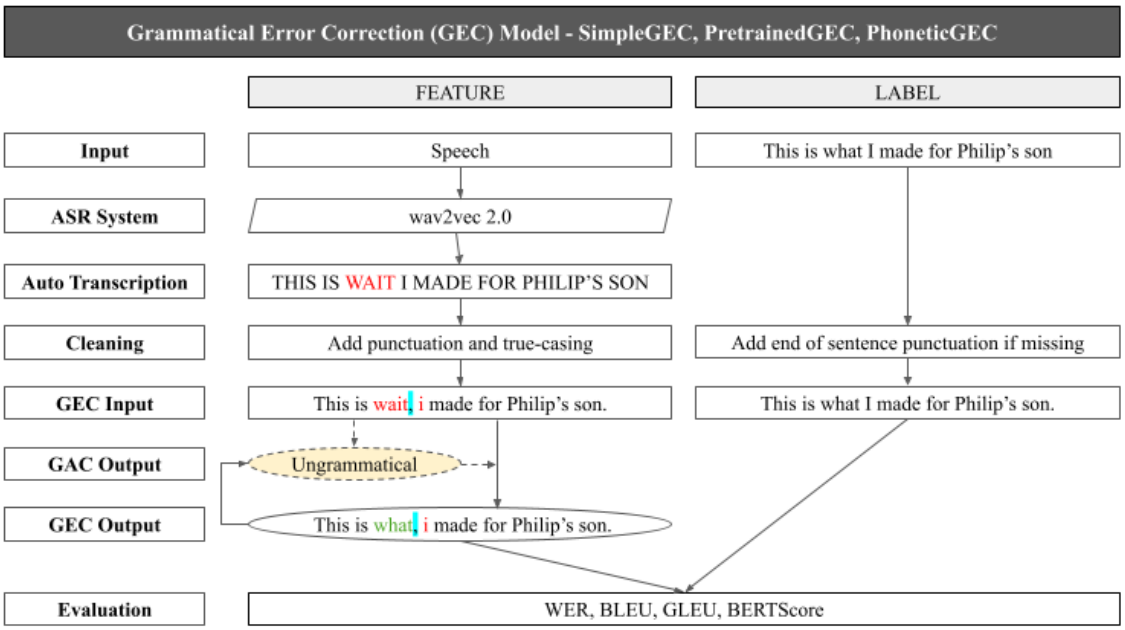


Figure 1: Overall process for improving ASR output using a transformer-based Grammatical Error Correction approach, and steps of the GAC model and MaskedLM used in our approach.



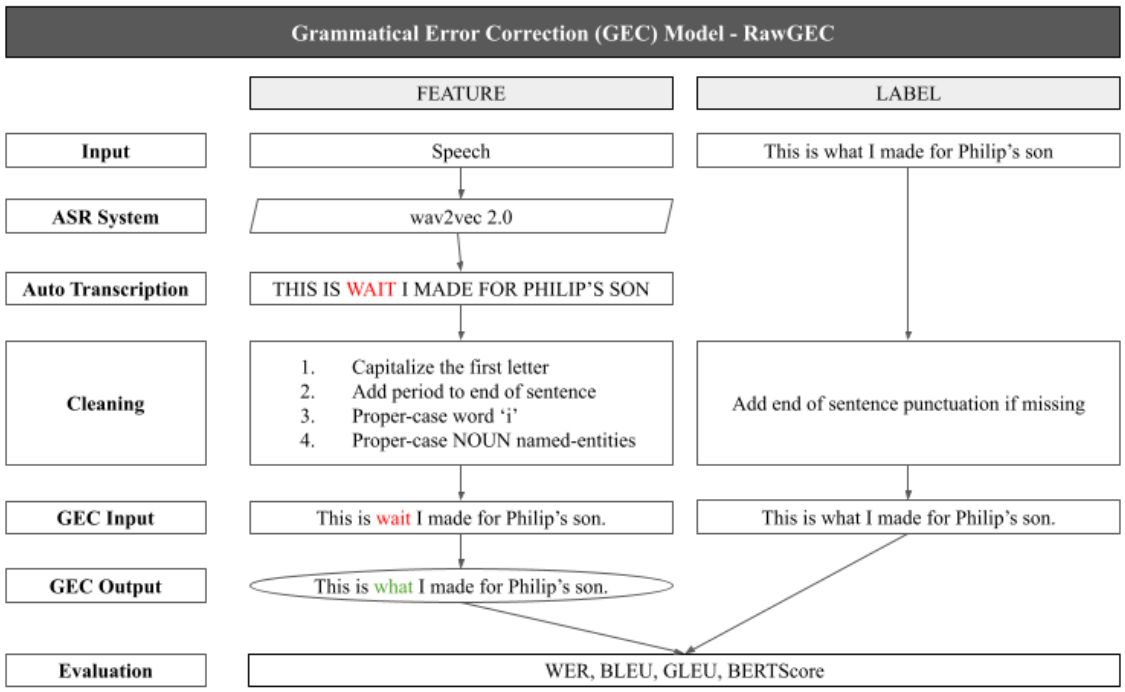


Figure 2: Steps of the different GEC models.

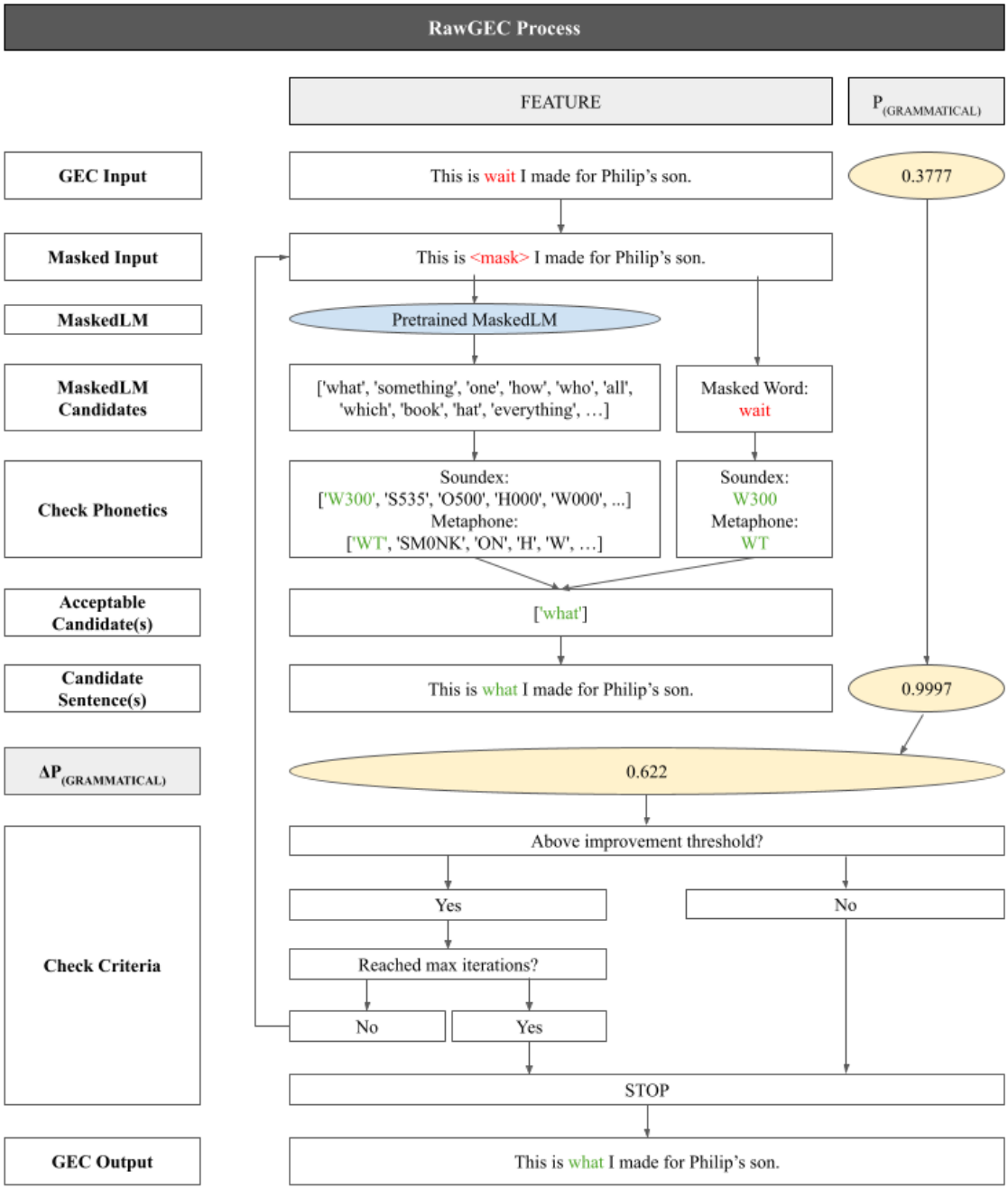


Figure 3: Steps of the best performing RawGEC.

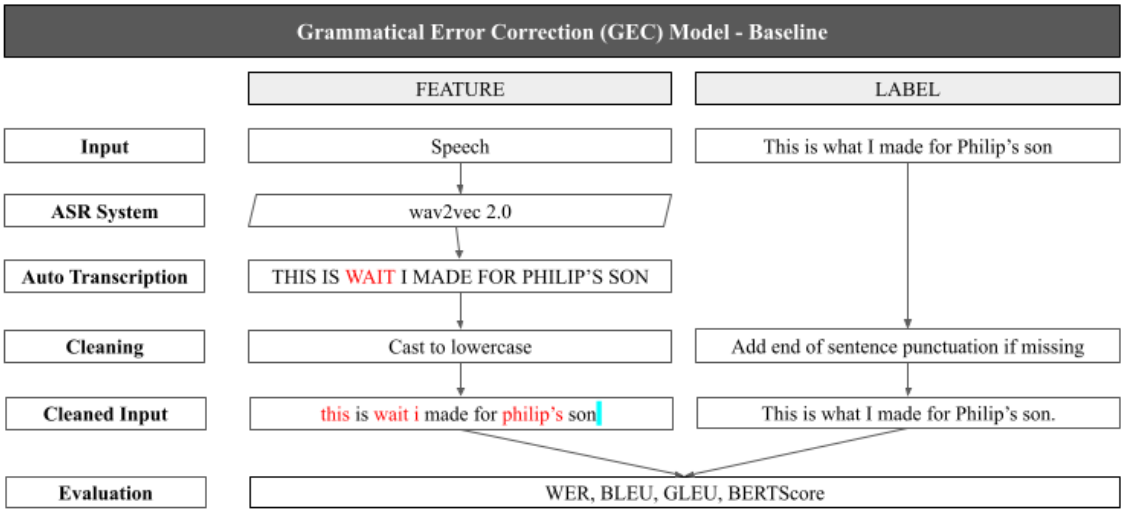


Figure 4: Cleaning steps performed for the baseline GEC models.

Type	Category	Sample sentence
FP	s-syntax	As you eat the most, you want the least.
	question	Who does John visit Sally because he likes?
	auxiliary	The more you would want, the less you would eat.
FN	auxiliary	Most people probably consider, even though the courts didn't actually find, Klaus guilty of murder.
	arg altern	The tank leaked the fluid free.
	s-syntax	Mary claimed that eating cabbage, Holly shouldn't.

Table 1: FP = false positives. i.e. sentences that are grammatically unacceptable but predicted to be acceptable. FN = false negatives. i.e. sentences that are grammatically acceptable but predicted to be unacceptable.

Model reference	Model type	Grammatical threshold	Improvement threshold	K-beams	Dataset
A	SimpleGEC	0.50	0.25	3	reduced_clean
B	SimpleGEC	0.75	0.25	3	reduced_clean
C	SimpleGEC	0.90	0.10	3	reduced_clean
D	PretrainedGEC	0.50	0.25	3	reduced_clean
E	PretrainedGEC	0.75	0.25	3	reduced_clean
F	PretrainedGEC	0.90	0.10	3	reduced_clean
G	PhoneticGEC	0.50	0.25	20	reduced_raw
H	PhoneticGEC	0.75	0.25	20	reduced_raw
J	PhoneticGEC	0.90	0.10	20	reduced_raw
K	RawGEC	0.00	0.10	20	reduced_raw
L	RawGEC	0.00	0.15	20	reduced_raw
M	RawGEC	0.00	0.25	20	reduced_raw

Table 2: Evaluation metrics for all GEC models with the reduced_train set. The clean dataset included proper casing and punctuation, whereas the raw dataset was generated by lower-casing the autotranscription generated by wav2vec 2.0 and proper-casing named entities.

Model	WER	BLEU	GLEU	PRECISION	RECALL	F1
Autotranscription	0.3760	0.4833	0.5406	0.9573	0.9625	0.95987
A	0.1837	0.6691	0.7224	0.9691	0.9716	0.9703
B	0.1892	0.6578	0.7133	0.9682	0.9701	0.9691
C	0.2061	0.6397	0.6972	0.9659	0.9672	0.9665
D	0.1833	0.6695	0.7230	0.9694	0.9718	0.9706
E	0.1877	0.6607	0.7153	0.9687	0.9705	0.9695
F	0.1998	0.6412	0.6983	0.9660	0.9674	0.9667
G	0.1749	0.6838	0.7333	0.9708	0.9742	0.9725
H	0.1843	0.6663	0.7190	0.9690	0.9719	0.9704
J	0.1753	0.6836	0.7332	0.9709	0.9743	0.9726
K	0.1594	0.6963	0.7465	0.9722	0.9738	0.9729
L	0.1578	0.7006	0.7494	0.9724	0.9740	0.9732
M	0.1563	0.7039	0.7518	0.9727	0.9742	0.9734

Table 3: Evaluation metrics for all punctuated GEC models with the reduced_train set. Precision, recall and F1 values were calculated with the BERTScore.

Model	WER	BLEU	GLEU	PRECISION	RECALL	F1
Autotranscription	0.0911	0.8195	0.8573	0.9783	0.9801	0.9791
A	0.1063	0.7873	0.8312	0.9751	0.9756	0.9753
B	0.1133	0.7713	0.8178	0.9737	0.9736	0.9736
C	0.1306	0.7362	0.7895	0.9706	0.9693	0.9699
D	0.1057	0.7897	0.8326	0.9754	0.9759	0.9756
E	0.1114	0.7749	0.8208	0.9741	0.9739	0.9740
F	0.1300	0.7399	0.7910	0.9702	0.9689	0.9695
G	0.0920	0.8176	0.8557	0.9782	0.9799	0.9790
H	0.1068	0.7857	0.8293	0.9749	0.9761	0.9755
J	0.0918	0.8179	0.8562	0.9784	0.9801	0.9792
K	0.0966	0.8047	0.8467	0.9774	0.9790	0.9782
L	0.0950	0.8096	0.8500	0.9777	0.9793	0.9785
M	0.09353	0.8132	0.8527	0.9781	0.9797	0.9788

Table 4: Evaluation metrics for all unpunctuated and lowercased GEC models with the reduced_train set. Precision, recall and F1 values were calculated with the BERTScore.

Metric	Female Autotranscription	Female GEC	Male Autotranscription	Male GEC
WER	0.3089	0.0761	0.3123	0.0858
BLEU	0.6017	0.8429	0.5869	0.8211
GLEU	0.6302	0.8690	0.6162	0.8461
Precision	0.9747	0.9879	0.9708	0.9839
Recall	0.9779	0.9898	0.9745	0.9863
F1	0.9762	0.9888	0.9726	0.9851

Table 5: Evaluation metrics for RawGEC model under the best-scoring improvement threshold of 0.25 for gender, using the gender subsets. Precision, recall and F1 values were calculated with the BERTScore.

Metric	**Autotranscription**	**Neutral**
WER	0.3292	0.1225
BLEU	0.5639	0.7830
GLEU	0.5975	0.8059
Precision	0.9703	0.9820
Recall	0.9725	0.9821
F1	0.9713	0.9820
Metric	**Autotranscription**	**Angry**
WER	0.3576	0.1533
BLEU	0.5191	0.7258
GLEU	0.5643	0.7619
Precision	0.9619	0.9750
Recall	0.9651	0.9758
F1	0.9635	0.9754
Metric	**Autotranscription**	**Disgust**
WER	0.3544	0.1470
BLEU	0.5135	0.7286
GLEU	0.5638	0.7632
Precision	0.9641	0.9774
Recall	0.9664	0.9765
F1	0.9652	0.9769
Metric	**Autotranscription**	**Sleepy**
WER	0.3594	0.1599
BLEU	0.5266	0.7309
GLEU	0.5636	0.7615
Precision	0.9588	0.9725
Recall	0.9646	0.9744
F1	0.9616	0.9734
Metric	**Autotranscription**	**Amused**
WER	0.3742	0.1830
BLEU	0.5099	0.6914
GLEU	0.5561	0.7335
Precision	0.9567	0.9682
Recall	0.9643	0.9729
F1	0.9604	0.9705

Table 6: Evaluation metrics for RawGEC model under the best-scoring improvement threshold of 0.25 for emotion, using the emotion subsets. Precision, recall and F1 values were calculated with the BERTScore.