Literature Review: Progress Meeting 1

1 Corpora Generation for Grammatical Error Correction (Jared et al.)

The Wikipedia revision source is primarily used in this study due to its broad coverage of topics. The first strategy of data generation from the Wikipedia source involves utilizing the edits themselves. These edits are valuable because they provide a more accurate distribution of natural grammatical errors made by humans. However, since this source contains topics that are highly popular, the results can be skewed in favor of those pages. To address this issue, the authors discard pages exceeding 64 MB. Additionally, to prevent the remaining large pages from skewing the dataset towards their topics through their numerous revisions, the authors downsample consecutive revisions from individual pages. Specifically, they select only $\log_{1.5}(n)$ pairs for a page with a total of n revisions, reducing the total amount of data 20-fold. Texts are aligned using a minimum-edit rule (unspecified), and random cuts are introduced to generate source-target pairs. Spelling errors are introduced probabilistically in the source sequences at a rate of 0.003 per character, selecting randomly among deletion, insertion, replacement, or transposition of adjacent characters.

Round-trip translation is also used to create a corrupted dataset, using several bridge languages: French (Fr), German (De), Japanese (Ja), and Russian (Ru). These languages were chosen due to their high resource availability and relative dissimilarity. Spelling errors are introduced probabilistically (insertion, deletion, and transposition, each with 0.005/3). Additional corruption is introduced stochastically using common errors identified in Wikipedia **(Jennifer et al.)**.

The round-trip translations are relatively clean but only represent a subset of real-world errors. Conversely, Wikipedia data offers broad coverage of real-world grammatical errors but is polluted by significant noise.

Iterative Decoding: Using incremental edits significantly improves performance over single-shot decoding for models trained on the Wikipedia revision data (a highly noisy corpus). However, models trained on the relatively clean round-trip translation data do not show this improvement.

The authors pretrain their model on the Wikipedia revision dataset (both methods) and then fine-tune the model using the Lang-8 dataset, a parallel corpus for grammatical error correction (GEC) that contains high noise. Finetuning on Lang-8 demonstrates a significant benefit for Wikipedia-derived data (Tables 6 and 7). The Wikipedia revision dataset includes corrections outside the scope of GEC. Thus, finetuning on noisy data (where targets may still contain errors) helps the model make more conservative edits within the scope of GEC.

The model trained on all types of synthetic data performs best, as validated on the JFLEG set. Additionally, an ensemble of models outperforms any single model trained on a single corpus.

2 GenERRate: Generating Errors for Use in Grammatical Error Detection (Jennifer et al.)

The paper introduces the error-infliction tool *GenERRate*, which probabilistically generates parts-of-speech (POS) insertion, deletion, substitution, etc., and examines its limitations and scenarios where it performs well. The heuristics used in this paper assist in POS (morphological) error generation.

The error generation setup follows a manual analysis of a corpus of errors, compiling an error analysis file with 89 types of errors. The most frequent errors involve changes in noun or verb number or article deletion.

GenERRate applies this error analysis to 440,930 British National Corpus (BNC) sentences, producing an identical-sized set of synthetic examples (termed "new-ungram-BNC").

Results from this method show a positive impact on accuracy and recall, suggesting that analyzing a small dataset from the test domain can help generate more effective training data. This method is useful when a small-to-medium-sized learner corpus is available but insufficient for training, development, and test set division.

This strategy is similar to that used by **Jared et al.**, where Wikipedia revisions were analyzed to introduce errors into identical source-target pairs.

The paper also uses another corpus for student examinations. The synthetic data underperforms in this context, likely because the original data contains spelling issues not present in the augmented data. This finding encourages the modeling of spelling mistakes in error generation, as suggested by **Jared et al.**.

Using a mix of natural and synthetic data can mitigate much of the performance loss. The combination of both types of ungrammatical training data recovers significant performance degradation, indicating that artificial data can effectively augment naturally occurring training sets.

The limitations of GenERRate include handling covert errors (errors falling outside conventional GEC, such as sentence structure improvements). In contrast, **Jared et al.** implicitly handle covert errors and better generalize by using a large corpus of Wikipedia edits and enriching it with spelling issues.

Complex Errors: Another limitation is dealing with complex errors that need more than one GEC transformation. This situation is similar to the iterative decoding method in **Jared et al.**, where errors often involve multiple grammatical adjustments (e.g., "She is one of reason I became interested in English" $\rightarrow$ "She is one of the reasons I became interested in English").

3 Generating Inflectional Errors for Grammatical Error Correction in Hindi

Most current approaches for grammatical error correction (GEC) emphasize statistical and deep learning methods over rule-based methods. These methods treat GEC as a translation task, converting ungrammatical sentences to their correct forms (Brockett et al., 2006).

Unlike the stochastic approaches used by **Jared et al.** and **Jennifer et al.**, this paper generates a parallel corpus of synthetic errors by inserting errors into grammatically correct sentences using a rule-based process, focusing specifically on inflectional errors.

Hindi is identified as a fusional language that expresses grammatical features like case, gender, number, and tense through morphological changes. The approach discussed here is potentially extensible to other Indic languages, such as Urdu.

This paper also utilizes Wikipedia revision edits, similar to **Jared et al.**, and compares transformers with other pretrained models like MLConv and CopyAug. Results indicate that simpler transformer models are significantly outperformed by these models, encouraging the exploration of the latest state-of-the-art models.

While this paper only addresses a subset of errors within conventional GEC, it acknowledges that more complex errors (discussed by **Jennifer et al.** and **Jared et al.**) can be valuable for identifying natural Hindi spelling errors. This can help circumvent dataset limitations previously encountered by Etoori et al.

4 A Tagged Corpus and a Tagger for Urdu

This study provides a corpus containing 5 million sentences, 95 million tokens, and a vocabulary of 500,000 words for Urdu.

5 A Low-Resource Approach to the Grammatical Error Correction of Ukrainian

This is a paper about a shared GEC task for ukranian. The first approach is based on finetuning a large multilingual language model (mT5) in two stages: first, on synthetic data, and then on gold data. The second approach trains a (smaller) seq2seq Transformer model pre-trained on synthetic data and finetuned on gold data. The two-stage approach is beneficial and provides a large boost compared to an mtT5 model finetuned on gold data only. Usage of full-sentence translations but only extract backtranslation pairs which are then used for introducing errors in native data. (The bridge language technique mentioned earlier). Usage of two baseline data generation techniques for low-resource settings: (1) spell-based transformations and (2) part-of-speech (POS)-based transformations. Both of the methods rely on the idea of using confusion sets that specify for each target word occurring in a native corpus a list of highly confusable words. Spell-based Transformations method involves creating errors based on spelling mistakes, these sets include words that are highly confusable based on their edit distance (the number of changes needed to transform one word into another). For this purpose, an open-source spellchecker "Aspell" , along with a Ukrainian dictionary was used. Part-of-Speech (POS)-based Transformations method involves creating errors based on the part of speech of words. These sets specify for each target word a list of highly confusable words based on their part of speech. Usage of the neural machine translation systems of **Tiedemann and Thottingal (2020)** to translation from Ukrainian into English and back.