# A distantly supervised Grammatical Error Detection/Correction system for Swedish

**Murathan Kurfalı***
Department of Psychology†
Stockholm University
murathan.kurfali@su.se

**Robert Östling***
Department of Linguistics
Stockholm University
robert@ling.su.se

## Abstract

This paper presents our submission to the first Shared Task on Multilingual Grammatical Error Detection (MultiGED-2023). Our method utilizes a transformer-based sequence-to-sequence model, which was trained on a synthetic dataset consisting of 3.2 billion words. We adopt a distantly supervised approach, with the training process relying exclusively on the distribution of language learners' errors extracted from the annotated corpus used to construct the training data. In the Swedish track, our model ranks fourth out of seven submissions in terms of the target $F_{0.5}$ metric, while achieving the highest precision. These results suggest that our model is conservative yet remarkably precise in its predictions.

## 1 Introduction

In today's interconnected world, learning a language is not optional for the majority of people. With digital platforms now the primary medium for individuals to express their thoughts and ideas, written communication has taken precedence over verbal communication, many people often find themselves producing text in a language that is not their first language. Consequently, natural language processing (NLP) systems that can assist non-native speakers in producing grammatically correct text are now more essential than ever. Grammatical error detection (GED) and grammatical error correction (GEC) are two well-established tasks that are designed to improve the writing skills of language users by identifying their errors as well as offering possible suggestions to correct them (Ng et al., 2014; Bryant et al., 2019; Ranalli and Yamashita, 2022).

This paper presents a system description of our submission to the first Shared task on Multilingual Grammatical Error Detection, MultiGED-2023 (Volodina et al., 2023). Our approach relies on training a transformer-based sequence-to-sequence model on a synthetic dataset, building upon previous work (e.g. Grundkiewicz et al., 2019; Nyberg, 2022). The distantly supervised training process requires manually error-annotated corpus exclusively to extract the distribution of language learners' errors which is mimicked in the synthetic data creation process. Hence, the employed pipeline aims to capture the characteristics of errors made by language learners while sidestepping the problem of sparsity by eliminating the need for direct supervision or large labeled datasets.

Our submission is confined to Swedish as the developed model is intended as a baseline for our ongoing work on Swedish grammatical error correction using large language models (Östling and Kurfalı, 2022). According to the official results, our model[1] is very accurate with a high precision score, indicating that it has a low false positive rate; yet, it cannot recognize various error types, as suggested by the low recall scores. The rest of the paper discusses previous work on Swedish (Section 2), presents the system in detail (Section 3), analyzes the results and implications (Section 4), and concludes with suggestions for future research directions (Section 5).

## 2 Related Work

Following our focus on Swedish, we restrict this section to research on Swedish grammatical error correction. Granska (Domeij et al., 2000) is one of the earliest Swedish grammar-checking systems, using part-of-speech tagging, morphological features, and error rules to identify grammat-

*The authors contributed equally to this work

†Work carried out while at the Department of Linguistics.



[1]https://github.com/MurathanKurfali/swedish-gec

Murathan Kurfalı and Robert Östling. A distantly supervised Grammatical Error Detection/Correction system for Swedish. *Proceedings of the 12th Workshop on Natural Language Processing for Computer Assisted Language Learning (NLP4CALL 2023)*. Linköping Electronic Conference Proceedings 197: 35–39.

| Method | Original Sentence | Corrupted Sentence |
|---|---|---|
| 1. Rearrange words | Jag älskar att läsa läroböcker. | Jag **läroböcker** att **älskar** läsa. |
| 2. Insert spurious words or phrases | Jag älskar att läsa läroböcker. | Jag älskar att **plötsligt** läsa läroböcker. |
| 3. Replace words or phrases | Jag älskar att läsa läroböcker. | Jag älskar att **skriva** läroböcker. |
| 4. Change inflections, split compounds | Jag älskar att läsa läroböcker. | Jag **älskade** att läsa **läro bok**. |
| 5. Letter substitutions | Jag älskar att läsa läroböcker. | Jag **älskat** att **läda** läroböcker. |
| 6. Change capitalization | Jag älskar att läsa läroböcker. | **jag** älskar **ATT** läsa **LÄROBÖCKER**. |

Table 1: Illustration of corruption methods applied to a simple sentence, "I love reading textbooks." Note that the table is not exhaustive and showcases only one of the several possible ways a sentence can be corrupted by a specific strategy, and not necessarily the most probable way. For simplicity, the illustration does not show errors added on top of each other, as done in the real data.

ical issues. More recent studies have explored methods to correct errors in learner texts, such as using word embeddings to obtain correction candidates (Pilán and Volodina, 2018) and a tool developed by (Getman, 2021) that detects erroneous words and sequences, suggesting corrections based on sub-word language models and morphological features.

Nyberg (2022) is the most notable, if not the only, example of integrating neural approaches into Swedish GEC, which also serves as the basis for our approach. Nyberg (2022) conducts GEC using two different but related methods: one employing a Transformer model for a neural machine translation approach, and the other utilizing a Swedish version of the pre-trained language model BERT to estimate the likelihood of potential corrections. These methods have demonstrated promising results in correcting different error types, with the first approach excelling at handling syntactical and punctuation errors, while the latter outperforms in addressing lexical and morphological errors.

## 3 System Overview

In the following section, we provide a detailed description of our submission. Our system is primarily a grammatical error correction model which is trained on a synthetic dataset consisting of original sentences and their artificially corrupted versions. The rest of the section details our training data generation procedure, model architecture, and the post-processing step to arrive at the locations of the identified errors.

### 3.1 Training data

We generally follow the approach of (Nyberg, 2022) in generating artificial data by corrupting text, but use more extensive corruption heuristics.

Data is collected from the collection of Språkbanken[2], and consists of a number of mixed-domain corpora of modern Swedish. This includes blog texts, news, and fiction. Since all data is processed sentence by sentence, we use sentence-scrambled data which we deduplicate after merging all the subcorpora. The final amount of data is 3.2 billion words. Empirical distributions for error types is derived from the DaLAJ (Volodina et al., 2021) dataset of linguistic acceptability in Swedish.

Corruption of sentences is performed as a pipeline, where each of the following procedures is applied in order:

1. *Rearrange words*. With probability 0.1, the word at position $i$ is moved to a position sampled from $\mathcal{N}(i, 1.5)$ and rounded to the nearest integer. Words are not moved across punctuation marks.

2. Insert spurious words or phrases. For each sentence position $i$, with probability 0.025 an n-gram (possibly a unigram) is inserted at this position. The n-gram to be inserted is sampled from the DaLAJ distribution.

3. *Replace words or phrases*. For each sen-

[2]https://spraakbanken.gu.se/ – specifically we used the following corpora, which constitutes Språkbanken's collection of modern Swedish corpora at the time of download: *sweachum, sweacsam, romi, romii, rom99, storsuc, bloggmix1998, bloggmix1999, bloggmix2000, bloggmix2001, bloggmix2002, bloggmix2003, bloggmix2004, bloggmix2005, bloggmix2006, bloggmix2007, bloggmix2008, bloggmix2009, bloggmix2010, bloggmix2011, bloggmix2012, bloggmix2013, bloggmix2014, bloggmix2015, bloggmix2016, bloggmix2017, bloggmixodat, gp1994, gp2001, gp2002, gp2003, gp2004, gp2005, gp2006, gp2007, gp2008, gp2009, gp2010, gp2011, gp2012, gp2013, gp2d, press65, press76, press95, press96, press97, press98, webbnyheter2001, webbnyheter2002, webbnyheter2003, webbnyheter2004, webbnyheter2005, webbnyheter2006, webbnyheter2007, webbnyheter2008, webbnyheter2009, webbnyheter2010, webbnyheter2011, webbnyheter2012, webbnyheter2013, attasidor, dn1987, ordat, fof, snp7879, suc3, wikipedia-sv, talbanken*

tence position $i$, sample a replacement n-gram from the empirical replacement distribution in DaLAJ. Word deletion may also be performed at this stage, by replacing by a shorter n-gram. In most cases, this leads to no change.

4. *Change inflections and split compounds*. With probability 0.1, pick a random new inflection of the word (assuming it can be inflected – otherwise do nothing). With probability 0.25, split compounds by inserting spaces. The compound analysis is performed using the morphological lexicon of SALDO (Borin et al., 2013).

5. *Letter substitutions*. For each letter in the sentence, sample it using the empirical letter replacement distribution from DaLAJ. In most cases this results in no change. A temperature parameter of $t = 1.5$ is used when sampling.

6. *Change capitalization*. With probability 0.2, turn the whole sentence into lowercase. With probability 0.01, turn the whole sentence into upper-case. With probability 0.025, perform the following: for each individual *word* in the sentence, turn it to uppercase with probability 0.1.

We note that the DaLAJ dataset is derived from the SweLL corpus (Volodina et al., 2019), and the statistics used to estimate the sampling distributions for text corruption may overlap to some extent with the source of the shared task test set. It is unfortunately difficult to quantify exactly how large the overlap is, since both datasets (DaLAJ and the SweLL-derived MultiGED test set) have been created independently from the SweLL corpus using different types of processing that makes it challenging to map sentences between the two resources. We hope that future work will be able to remedy this problem by ensuring that fully disjoint sets of data are used to estimate the corruption model parameters and evaluate the final grammatical error detection system.

### 3.2 Model Architecture

We model grammatical error correction as a translation problem where the input sentence with errors is treated as the source language and the corrected sentence as the target language. Our model is based on the transformer architecture (Vaswani et al., 2017), which has become the default choice for many natural language processing tasks due to its self-attention mechanism which is highly effective in capturing long-range dependencies in sequences.

| Team Name | P | R | F0.5 |
|---|---|---|---|
| EliCoDe | 81.80 | **66.34** | **78.16** |
| DSL-MIM-HUS | 74.85 | 44.92 | 66.05 |
| Brainstorm Thinkers | 73.81 | 39.94 | 63.11 |
| Our system | **82.41** | 27.18 | 58.60 |
| VLP-char | 26.40 | 55.00 | 29.46 |
| NTNU-TRH | 80.12 | 5.09 | 20.31 |

Table 2: Official results for the Swedish language.

We implement our model with the OpenNMT-py library (Klein et al., 2017), following the suggested base configuration. The model is trained for 100,000 training steps, with a validation step interval of 10,000 and an initial warm-up phase of 8,000 steps. Both the encoder and decoder are of the transformer type, with 6 layers, a hidden size of 512, and 8 attention heads. We learn a sentencepiece vocabulary (Kudo and Richardson, 2018) of 32,000 sub-word units to tokenize the sentences.

**Training configuration** We trained our model using mini-batches containing 400 sentence pairs, distributed across four GPUs, and accumulated gradients for 4 iterations. This resulted in an effective mini-batch size of 6,400 sentence pairs. The training was carried out on A100 GPUs, taking approximately 16 hours in total to complete.

### 3.3 Post-processing: Correction to Detection

As mentioned earlier, despite the shared task's focus on grammatical error detection, our model is originally trained as a grammatical error correction model which we developed as a baseline in our ongoing work (Östling and Kurfalı, 2022). Therefore, the output of our model is in the form of corrected sentences rather than detected errors. To convert the corrected sentences into detected errors, we perform post-processing on the model's output.

We use the difflib library[3] to compare the original sentences with the corrected sentences and identify the differences between them. Given the goal of the shared task is to identify incorrect

[3]https://docs.python.org/3/library/difflib.html

| | P | R | F0.5 |
|---|---|---|---|
| Training set | 78.72 | 26.63 | 56.59 |
| Development set | 81.52 | 26.73 | 57.82 |
| Test set | 82.41 | 27.18 | 58.60 |

Table 3: Additional results on the training and development set. The last line refers to the official results on the test set.

words, we disregard all additions made by our model and focus on the changes performed on the original sentences. Specifically, any words that are not copied unchanged from the original sentence to the corrected sentence are marked as errors that needed correction.

## 4 Results and Discussion

In this section, we present the results of the shared task on grammatical error detection for the Swedish language. The performance of our system is compared to other participating teams in terms of precision (P), recall (R), and F0.5 score, which is the harmonic mean between precision and recall, with a higher emphasis on precision. Table 2 provides an overview of the performance metrics for each team.

As shown in Table 2, our system achieved the highest precision of 82.41% among all participants. This indicates that our model's predictions for grammatical errors were highly accurate. However, our recall score of 27.18% demonstrates that our model failed to identify a significant proportion of the actual errors in the dataset. This trade-off between precision and recall resulted in an F0.5 score of 58.60%, which places our system in the fourth position among the six participating teams.

In addition to the official results on the test, we present additional results on the shared task's training and development sets in Table 3 as none of these sets are utilized during the model training. We observe that the results are stable across the sets and our model exhibits the same conservative behavior.

Lastly, it is worth noting that the task of grammatical error correction is significantly more challenging than the task of grammatical error detection. While error detection is essentially a binary classification problem at the token level, error correction requires identifying the specific type and location of the error as well as suggesting a suitable correction. Consequently, our pipeline is counter-intuitive in the sense that we are using a more sparse task (error correction) to tackle a simpler one (error detection). Therefore, we would like to emphasize that the results are unlikely to reflect the full potential of such a transformers-based model for grammatical error detection. It's highly probable that the model could perform much better if trained specifically to predict whether an individual token requires correction or not.

## 5 Conclusion

In this paper, we described our submission to the first Shared task on Multilingual Grammatical Error Detection (MultiGED-2023) for the Swedish language. Our approach relied on a transformer-based sequence-to-sequence model trained on a synthetic dataset, using a distantly supervised training process. Our system achieved the highest precision score among the participating teams, indicating that our model's predictions for grammatical errors are highly accurate. However, our low recall score indicated that our model was not able to detect all errors in the dataset, possibly a limitation of the training process.

## 6 Future work

While our current proposal focuses exclusively on Swedish, the proposed pipeline can be readily adapted to other languages with an error-annotated corpus and a large monolingual corpus. Additionally, an interesting direction for further research would be to explore the effectiveness of following the error distribution derived from the error-annotated corpus through an ablation study.

## Acknowledgments

The computations were enabled by resources provided by the Swedish National Infrastructure for Computing (SNIC) at C3SE partially funded by the Swedish Research Council through grant agreement no. 2018-05973.

This work was funded in part by the Swedish Research council through grant agreement no. 2019-04129.

## References

Lars Borin, Markus Forsberg, and Lennart Lönngren. 2013. Saldo: a touch of yin to wordnet's yang. *Language resources and evaluation*, 47:1191–1211.

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.

Rickard Domeij, Ola Knutsson, Johan Carlberger, and Viggo Kann. 2000. Granska–an efficient hybrid system for swedish grammar checking. In *Proceedings of the 12th Nordic Conference of Computational Linguistics (NODALIDA 1999)*, pages 49–56.

Yaroslav Getman. 2021. Automated writing support for swedish learners. In *Swedish Language Technology Conference and NLP4CALL*, pages 21–26.

Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Kenneth Heafield. 2019. Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 252–263.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M Rush. 2017. Opennmt: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71.

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.

Martina Nyberg. 2022. Grammatical error correction for learners of swedish as a second language. Master's thesis, Uppsala University, Department of Linguistics and Philology.

Ildikó Pilán and Elena Volodina. 2018. Exploring word embeddings and phonological similarity for the unsupervised correction of language learner errors. In *Proceedings of the Second Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 119–128.

Jim Ranalli and Taichi Yamashita. 2022. Automated written corrective feedback: Error-correction performance and timing of delivery. *Language Learning & Technology*, 26(1):n1.

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.

Elena Volodina, Christopher Bryant, Andrew Caines, Orphée De Clercq, Jennifer-Carmen Frey, Elizaveta Ershova, Alexandr Rosen, and Olga Vinogradova. 2023. MultiGED-2023 shared task at NLP4CALL: Multilingual Grammatical Error Detection. In *Proceedings of the 12th workshop on Natural Language Processing for Computer-Assisted Language Learning (NLP4CALL)*, Tórshavn, Faroe Islands.

Elena Volodina, Lena Granstedt, Arild Matsson, Beáta Megyesi, Ildikó Pilán, Julia Prentice, Dan Rosén, Lisa Rudebeck, Carl-Johan Schenström, Gunlög Sundberg, et al. 2019. The swell language learner corpus: From design to annotation. *Northern European Journal of Language Technology (NEJLT)*, 6:67–104.

Elena Volodina, Yousuf Ali Mohammed, and Julia Klezl. 2021. DaLAJ – a dataset for linguistic acceptability judgments for Swedish. In *Proceedings of the 10th Workshop on NLP for Computer Assisted Language Learning*, pages 28–37, Online. LiU Electronic Press.

Robert Östling and Murathan Kurfalı. 2022. Really good grammatical error correction, and how to evaluate it. In *the ninth Swedish Language Technology Conference (SLTC2022)*.