UAM-CSI at MultiGEC-2025: Parameter-efficient LLM Fine-tuning for Multilingual Grammatical Error Correction

Ryszard Staruch
Adam Mickiewicz University in Poznan
Center for Artificial Intelligence AMU
ryszard.staruch@amu.edu.pl

Abstract

This paper describes the solution of the UAM-CSI team to the shared task on Multilingual Grammatical Error Correction (MultiGEC-2025), which is part of the workshop on Natural Language Processing for Computer-Assisted Language Learning (NLP4CALL). The shared task covers 12 languages: Czech, English, Estonian, German, Greek, Icelandic, Italian, Latvian, Russian, Slovene, Swedish and Ukrainian. The aim of the task is to correct errors in the provided texts. Our system is a google/gemma-2-9b-it model with 2 QLoRA adapters, one for the minimal-edit track and another for the fluency-edit track. Our solution achieves the best performance on the test sets on GLEU and $F_{0.5}$ metrics for all languages and the best performance on the Scribendi Score metric except for the Greek language in the minimal-edit track.

1 Introduction

Grammatical Error Correction (GEC) is an NLP task that covers the detection and correction of all errors occurring in the given text. There are two main directions in the GEC field: the minimal-edit error correction and the fluency-edit error correction.

The first direction for English language is mostly concerned around second language learners in their learning process, which was carried out in published datasets, for example FCE (Yannakoudakis et al., 2011) and previous shared tasks: CoNLL-2014 (Ng et al., 2014) and BEA-2019 (Bryant et al., 2019). The most common measure of the effectiveness of the minimal-edit error correction systems is the $F_{0.5}$ score, which puts the higher weight for precision than recall.

The second direction for the English language focuses not only on correcting errors in texts but also on improving the fluency of the texts (Sakaguchi et al., 2016). There is only one dataset for English that was designed for the fluency-edit approach, the JFLEG dataset (Napoles et al., 2017). The primary metric for the JFLEG dataset is GLEU (Napoles et al., 2015), which is a modified version of BLEU (Papineni et al., 2002) that better fits the text correction task.

One of the main problems in GEC research is that most of the work is done only for the English language. There is ongoing research for other languages, mostly Chinese and Arabic, but there is an urgent need to address the lack of research on lesser-used languages. The biggest problem is mostly related to limited high-quality datasets, which are needed to create and evaluate GEC systems.

MultiGEC-2025 (Masciolini et al., 2025a) is the first shared task that covers many languages. It comes with the training, development and test datasets for each language. The task has two tracks: the minimal-edit track and the fluency-edit track. The novel feature of this shared task is that the texts are not divided on the sentence level, which was common practice in previous datasets. Systems are evaluated using three evaluation metrics: $F_{0.5}$, GLEU and Scribendi Score (Islam and Magnani, 2021). The Scribendi Score is a reference-free metric that uses a language model perplexity score to evaluate predictions. Using three metrics provides different perspectives on the quality of the submitted systems. It also enables the opportunity to analyze how different metrics behave across all datasets for solutions in the shared task, which will contribute to the research on the GEC evaluation.

In this paper, we describe two systems for the shared task, each for a different track. The organizers encouraged developing systems that are

Lang	Subcorpus	Learners	# Train	# Dev	# Test	# Total	# References
cs	NatWebInf	L1	3620	1291	1256	6167	2
cs	Romani	L1	3247	179	173	3599	2
cs	SecLearn	L2	2057	173	177	2407	2
cs	NatForm	L1	227	88	76	391	2
en	Write & Improve	L2	4040	506	504	5050	1
et	EIC	L2	206	26	26	258	3
et	EKIL2	L2	1202	150	151	1503	2
de	Merlin	L2	827	103	103	1033	1
el	GLCII	L2	1031	129	129	1289	1
is	IceEC	L1	140	18	18	176	1
is	IceL2EC	L2	155	19	19	193	1
it	Merlin	L2	651	81	81	813	1
lv	LaVA	L2	813	101	101	1015	1
ru	RULEC-GEC	mixed	2539	1969	1535	6043	3
sl	Solar-Eval	L1	10	50	49	109	1
sv	SweLL_gold	L2	402	50	50	502	1
uk	UA-GEC	mixed	1706	87	79	1872	4

Table 1: Overview of the subcorpora of the MultiGEC-2025 shared task with their sizes measured by the number of essays.

able to process all languages using a single model, which was done in our systems. We use the same architecture for both tracks: google/gemma-2-9b-it model (later denoted as Gemma 2) with QLoRA adapters, one for each track. The difference between systems is that the minimal-edit track system was fine-tuned only on one reference text for each dataset, whereas for the fluency-edit track, the system was fine-tuned on all reference texts. Our intuition behind this approach is that the model should produce more fluent output if it sees many ways to correct given text.

2 Related work

In recent years, there were a few research studies that covered Grammatical Error Correction for many languages. Rothe et al. (2021) describes two things that are needed to produce state-of-the-art multilingual GEC models. The first one focuses on generating synthetic datasets. The other one is to use multilingual language models that already possess the ability to use different languages. The important takeaway from this work is that larger models are needed to perform effectively on many languages.

One of the most recent works (Luhtaru et al., 2024) shows that leveraging decoder-only large language models (LLMs) as both synthetic data generators and correctors leads to state-of-the-art results for German, Estonian and Ukrainian languages.

Coyne et al. (2023) shows that instruction-tuned LLMs without task-specific fine-tuning are able to correct text better than fine-tuned models for the task when evaluating on the fluency-edit GEC dataset. If we think of the grammatical error correction as the task of making the text more probable, it could mean that the GEC task is directly related to the language modeling task. In the minimal-edit task we want to make more probable text in the parts that are clearly considered as erroneous, when for the fluency-edit task we can think more widely of making the text more probable. Then, the fine-tuning process should be mostly responsible for adjusting the way of correcting a given text, which is always subjective to the annotator.

These studies show that in order to create a promising single-model system capable of correcting text in many languages, it is necessary to use a pre-trained, large, multilingual language model that is fine-tuned to learn how to effectively correct errors in different languages.

3 Dataset overview

The dataset used in the MultiGEC-2025 shared task is a multilingual Grammatical Error Correction corpus (Masciolini et al., 2025b). It covers

Hyperparameter name	Value
learning rate	5e-5
batch size	4
gradient accumulation steps	4
warmup steps	40
lr scheduler	linear
epochs	2
optimizer	AdamW8bit
weight decay	0.01
threshold (max tokens)	300
LoRA rank	128
LoRA alpha	64

Table 2: Hyperparameter values used during fine-tuning.

12 European languages: Czech, English, Estonian, German, Greek, Icelandic, Italian, Latvian, Russian, Slovene, Swedish and Ukrainian. The dataset is divided into 17 subcorpora. The detailed statistics about the dataset can be found in Table 1.

It is worth noting that the size of the subcorpora is measured by the number of essays, whereas most existing datasets are divided and measured at the sentence level. It enables to take into consideration context of the whole text, which should be beneficial during the correction process. Czech, Estonian, Russian and Ukrainian datasets contain more than one correct reference. The texts in almost every dataset are written by either L1 or L2 learners. Only the RULEC-GEC and the UA-GEC corpora contain mixed types of text authors. This makes the task even more challenging because different types of learners make different errors.

4 System description

Due to the need to use a multilingual LLM and limited resources (a single Nvidia RTX 4090 card), we decided to go for the Gemma 2 model as it is one of the best performing multilingual models in its size. Its effectiveness could be related to the large vocabulary of 256k tokens and the fine-tuning process, which involves learning the entire probability distribution from the larger model rather than just predicting the next token in the sentence (Gemma Team et al., 2024). To be able to use a relatively large context, for which more VRAM is needed, we decided to use the 4-bit model quantization, 2 QLoRA adapters (Dettmers et al., 2024), one for each track, and the Unsloth framework (Daniel Han, 2023).

Some essays in the MultiGEC-2025 dataset are too long to load them into the model, thus the proper essay splitting algorithm is needed to fulfill two conditions:

1. Do not extend the maximum input length threshold (later denoted as **threshold**).

2. Use more than a single sentence as the input for the model, to make sure that the larger context than a single sentence is being used.

Our essay splitting algorithm is defined as follows:

1. If the number of essay tokens in both the source and target texts is below the threshold, add the text pair to the dataset. Otherwise, go to point 2.

2. Split the essay by newlines to get **paragraphs**. For each paragraph, if the number of essay tokens in both source and target texts is below the threshold append it to the dataset. Otherwise, go to point 3.

3. Split the paragraph on the sentence level using SaT model (Frohmann et al., 2024) to get **sentences**. Then, sentences are sequentially joined together until the source text or the target text created from sentences exceeds the threshold. After exceeding the threshold, the text pair is added to the dataset and the process is repeated for the remaining sentences.

The above algorithm for the development and test datasets are applied only for the source text part. The information for the paragraphs and sentences splits is saved to properly align the predictions from the model.

Both QLoRA adapters were fine-tuned using the same hyperparameters, described in Table 2. The adapters were fine-tuned only for 2 epochs, because fine-tuning for more epochs did not improve the results on all development subcorpora. Fine-tuning for a single epoch takes about 3 hours.

As mentioned in the Introduction, the only difference between adapters is that the adapter for the minimal-edit track was fine-tuned on the single, first reference from the dataset. The fluency-edit track QLoRA adapter was fine-tuned on all references. During fine-tuning, the datasets were combined and shuffled, so the adapters were fine-tuned on all languages at once.

Lang	Subcorpus	Track	P	R	$F_{0.5}$	GLEU	Scribendi
cs	NatWebInf	Minimal	69.81	63.95	68.55	69.89	0.79
		Fluency	71.05	64.28	69.58	70.04	0.79
cs	Romani	Minimal	59.94	50.13	57.68	60.07	0.92
		Fluency	59.23	50.18	57.17	60.23	0.91
cs	SecLearn	Minimal	62.58	47.23	58.76	55.81	0.98
		Fluency	62.21	46.50	58.27	55.16	0.99
cs	NatForm	Minimal	68.32	46.94	62.62	81.44	0.99
		Fluency	68.71	46.82	62.83	81.07	0.95
en	Write & Improve	Minimal	62.24	50.78	59.55	81.5	0.98
		Fluency	62.57	48.67	59.19	80.67	0.98
et	EIC	Minimal	54.39	36.23	49.44	55.76	1.0
		Fluency	56.79	38.6	51.9	57.89	1.0
et	EKIL2	Minimal	58.82	41.28	54.21	66.85	1.0
		Fluency	56.66	42.86	53.23	68.23	1.0
de	Merlin	Minimal	68.17	66.43	67.81	81.13	1.0
		Fluency	67.42	66.28	67.19	81.23	0.96
el	GLCII	Minimal	53.79	45.11	51.8	56.84	0.88
		Fluency	53.62	44.12	51.4	55.96	0.9
is	IceEC	Minimal	57.28	8.45	26.58	84.98	1.0
		Fluency	61.76	9.03	28.48	85.09	0.72
is	IceL2EC	Minimal	38.68	4.62	15.62	43.6	0.63
		Fluency	41.18	4.13	14.73	43.62	0.74
it	Merlin	Minimal	69.04	59.54	66.91	81.89	0.98
		Fluency	67.45	56.67	64.98	79.97	1.0
lv	LaVA	Minimal	80.77	78.32	80.27	84.5	1.0
		Fluency	79.76	78.54	79.51	84.65	1.0
ru	RULEC-GEC	Minimal	61.09	33.01	52.21	83.11	0.46
		Fluency	62.3	30.94	51.8	82.65	0.43
sl	Solar-Eval	Minimal	53.89	30.4	46.68	66.46	1.0
		Fluency	54.14	29.77	46.52	66.32	1.0
sv	SweLL_gold	Minimal	54.54	45.88	52.56	69.29	1.0
		Fluency	55.29	46.69	53.32	69.62	1.0
uk	UA-GEC	Minimal	74.31	54.11	69.15	79.55	0.89
		Fluency	74.65	55.02	69.68	79.82	0.8

Table 3: Results for the test sets for all MultiGEC-2025 shared task subcorpora.

5 Results

Table 3 shows our results for the test datasets for the minimal-edit track and the fluency-edit track. The systems for both tracks perform similarly across the datasets, although there are a few subcorpora with notable differences between the metric values.

For the $F_{0.5}$ score metric there are two subcorpora for which the differences are much larger compared to other datasets: the et/EIC dataset for the fluency-edit model and the it/Merlin dataset for the minimal-edit model. The et/EIC is one of the smallest datasets, so providing additional pairs for this subcorpus could be the reason for the improved results. On the other hand, for the it/Merlin dataset, adding more references for other languages might have caused worse results for other datasets, because adjusting model weights for one language could affect performance for the other languages. Although for most of the datasets the difference is much smaller.

The differences for the GLEU metric are similar to the $F_{0.5}$ score metric, which is expected since both metrics are reference-based metrics. Although, when looking at the results of the other participants[1] the results with low $F_{0.5}$ score metric have a relatively high GLEU metric value, because the unchanged text does not have a 0 value for the GLEU metric. This makes it more difficult to interpret the metric value compared to the $F_{0.5}$ score metric.

The results for the Scribendi Score metric are very high or perfect for almost all datasets, even if the $F_{0.5}$ score values are around 50%. The metric gives a discrete score of -1, 0, or 1 for each text, so minimal improvements in the text lead to the positive score, even if many errors in the text are not corrected. The metric should work better in the sentence-level GEC, because instead of a single score for the long text, there would be many scores for each sentence that could be averaged. It reveals the drawbacks of the metric and shows that there is a need for research in the reference-less GEC evaluation, especially for long texts.

6 Conclusions

This work shows that a single LLM can effectively correct text in many languages. Despite limited resources, our systems were able to achieve the highest scores for each track and for each metric across all datasets except for the Scribendi Score for the fluency-edit track for the GLCII dataset. Our essay splitting algorithm provides an efficient way to make use of longer parts of texts. The use of three metrics for the task revealed that $F_{0.5}$ still remains a useful and practical metric and that the Scribendi Score metric could be modified to better fit the long-text GEC.

The MultiGEC-2025 Shared Task makes a valuable contribution to multilingual grammatical error correction research and opens new paths for GEC researchers.

7 Limitations

Our system requires a modern graphics card to effectively run the model inference, which could be a problem for users who want to run the model on their devices. We only tested the models performance on the datasets provided in the shared task, so we do not know how effectively it corrects errors in other languages. We also did not test other language models due to the shared task deadlines. Our work does not include human evaluation or analysis of different types of errors, which could provide more insight into the performance of the system.

References

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.

Steven Coyne, Keisuke Sakaguchi, Diana Galvan-Sosa, Michael Zock, and Kentaro Inui. 2023. Analyzing the performance of gpt-3.5 and gpt-4 in grammatical error correction.

Unsloth team Daniel Han, Michael Han. 2023. Unsloth.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2024. Qlora: efficient finetuning of quantized llms. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Markus Frohmann, Igor Sterner, Ivan Vulić, Benjamin Minixhofer, and Markus Schedl. 2024. Segment any text: A universal approach for robust, efficient and adaptable sentence segmentation. In *Proceedings of*

[1] https://spraakbanken.github.io/multigec-2025/shared_task.html#results

the 2024 Conference on Empirical Methods in Natural Language Processing, pages 11908–11941, Miami, Florida, USA. Association for Computational Linguistics.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Cogan, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. 2024. Gemma 2: Improving open language models at a practical size.

Md Asadul Islam and Enrico Magnani. 2021. Is this the end of the gold standard? a straightforward reference-less grammatical error correction metric. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3009–3015, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Agnes Luhtaru, Taido Purason, Martin Vainikko, Maksym Del, and Mark Fishel. 2024. To err is human, but llamas can learn it too. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 12466–12481, Miami, Florida, USA. Association for Computational Linguistics.

Arianna Masciolini, Andrew Caines, Orphée De Clercq, Joni Kruijsbergen, Murathan Kurfalı, Ricardo Muñoz Sánchez, Elena Volodina, and Robert Östling. 2025a. The MultiGEC-2025 shared task on multilingual grammatical error correction at NLP4CALL. In *Proceedings of the 14th Workshop on Natural Language Processing for Computer Assisted Language Learning*, Tallin, Estonia. University of Tartu.

Arianna Masciolini, Andrew Caines, Orphée De Clercq, Joni Kruijsbergen, Murathan Kurfalı, Ricardo Muñoz Sánchez, Elena Volodina, Robert Östling, Kais Allkivi, Špela Arhar Holdt, Ilze Auzina, Roberts Darģis, Elena Drakonaki, Jennifer-Carmen Frey, Isidora Glišić, Pinelopi Kikilintza, Lionel Nicolas, Mariana Romanyshyn, Alexandr Rosen, Alla Rozovskaya, Kristjan Suluste, Oleksiy Syvokon, Alexandros Tantos, Despoina-Ourania Touriki, Konstantinos Tsiotskas, Eleni Tsourilla, Vassilis Varsamopoulos, Katrin Wisniewski, Aleš Žagar, and Torsten Zesch. 2025b. Towards better language representation in Natural Language Processing – a multilingual dataset for text-level Grammatical Error Correction. *International Journal of Learner Corpus Research*.

Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 2015. Ground truth for grammatical error correction metrics. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 588–593, Beijing, China. Association for Computational Linguistics.

Courtney Napoles, Keisuke Sakaguchi, and Joel Tetreault. 2017. JFLEG: A fluency corpus and benchmark for grammatical error correction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 229–234, Valencia, Spain. 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.

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.

Sascha Rothe, Jonathan Mallinson, Eric Malmi, Sebastian Krause, and Aliaksei Severyn. 2021. A simple recipe for multilingual grammatical error correction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 702–707, Online. Association for Computational Linguistics.

Keisuke Sakaguchi, Courtney Napoles, Matt Post, and Joel Tetreault. 2016. Reassessing the goals of grammatical error correction: Fluency instead of grammaticality. *Transactions of the Association for Computational Linguistics*, 4:169–182.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 180–189, Portland, Oregon, USA. Association for Computational Linguistics.

A Prompt used during fine-tuning

Both adapters were fine-tuned using the same prompt. The following prompt was used:

Correct the following text, making only minimal changes where necessary.

Text to correct:

(text to correct)

Corrected text:

(corrected text)

B Requirements needed to run the model

The model requires 8.8GB of VRAM to be loaded into the graphics card. Additional VRAM is also required for the inference, so a graphics card with 12 GB of VRAM is the minimum requirement that is needed to run the inference, although more VRAM allows the batch size to be increased and the cache to be used.