University *of Ljubljana*
Faculty *of Computer and Information Science*

Automatic generation of Slovenian traffic news for RTV Slovenija

Matej Brodnik, Žan Mencigar, Primož Mihelak

Abstract

With the recent developments of artificial intelligence and large language models (LLMs), many simple tasks such as generating news from raw data can be streamlined and automatized. We will attempt to implement a robust solution that will enable automatic generation of traffic news in Slovene, given key traffic related events as input. We will employ prompt engineering and fine-tune the algorithm using the GaMS 9B LLM to generate short traffic news automatically. We will then analyze and present the results and discuss the algorithm's performance.

Keywords

GaMS 9B, LLM, Prompt engineering, Text generation, Traffic news

Advisors: Slavko Žitnik

Introduction

Traffic news are an important part of RTV Slovenija's radio program, where presenters read out traffic updates every 30 minutes. Currently, students manually check the promet.si portal, select relevant traffic events, and write short reports based on internal guidelines. This process takes time and could be made more efficient.

The goal of this project is to automate the generation of Slovene traffic news using GaMS 9B large language model (LLM) [1]. GaMS is based on Google's Gemma 2 family and was originally trained designed as 1B model, but was extended for 2B, 9B and 27B parameters. It was specifically trained for Slovene language, using a corpora of primarily Slovene, but also Croatian, Serbian and Bosnian documents consisting of 28 billion tokens [2].

The idea is to first explore how well we can solve the task using prompt engineering, and then improve the system by fine-tuning an LLM with provided data. We will evaluate the model using criteria such as how well it identifies important events, uses correct road names, follows text-length limits, and respects the style of RTV Slovenija's reports, while also measuring performance improvements between different methods, such as prompt engineering and fine-tuning. The goal is to create a system that can automatically generate traffic news similar to what is currently done manually.

Data

The traffic news data was scraped from promet.si over years 2022-2024 and is saved in Excel format, making up over 170k rows of data in total. Each row contains a timestamp of acquisition and 3 separate segments (A, B, C) that contain data based on their importance, with A containing urgent news, B containing more general but still important news and C containing less significant announcements that may remain unchanged for several months. This data is also split across 10 thematic categories (ie. accidents, weather, obstacles, ...). We also have access to existing RTV Slo bulletins from rtvslo.si containing human-written traffic news in regular 30 minute intervals in the same time period as the scraped data, serving as a reference to what the final output should look like.

Through manual review of the data, we also concluded that the timestamps in the data table are written in the UTC+0 time zone, as there often appears to be a 1+ hour gap between extracted data and reference reports. Since the reference reports use the UTC+1 time zone, we shift the timestamps accordingly when parsing data.

Finally, we are also given instructions detailing the news structure, road naming conventions, lecturer's notes, example formulations and general guidelines regarding the formatting of the news.

Related work

Since the emergence of LLMs, many similar projects have been conceived. Yuan et al. [3] describes the use of their

LLM named HEAL, which was purpose-built for medical conversations and was measured on automated scribing.

A related approach to automated report generation described by Guan et al. [4], is the SRAG (Summary Report Auto-Generation) framework which uses large language models to transform query data into high-quality summary reports through hierarchical corpus retrieval across general fields.

Similarly, Su et al. [5] introduce a model named TableGPT2, which is pre-trained and fine-tuned. Its extensive training enables it to excel in table-centric tasks while maintaining strong general language and coding abilities.

Zhao et al. [6] investigate the table-to-text generation capabilities of different large language models in different real-world information seeking scenarios.

Lastly, Tian et al. [7] propose ReSpark, which is a method based on LLMs to generate new reports using previous similar-topic data reports as references.

Methods

Data preparation

We collect input data by reading the timestamp of a reference report and extracting the corresponding data from the source data table. The input data is aggregated from all records that occur 45 minutes before the timestamp, and an additional 30 minutes before that recursively if no records appear in the initial time frame. We try to eliminate most duplicate occurrences to remove redundancy. In the end, we also annotate the data according to their thematic category.

Prompt engineering

We first approached our task by using GaMS 9B as is, relying only on a prompt to provide the necessary context for the task. We began by constructing the prompt from the guidelines the writers use. At first, we tried using the whole document, but realized that this overloads the LLM with information that is not always relevant for concrete cases. We reduced the instructions to a basic description of the problem and used a basic few-shot prompting strategy, focusing on finding quality examples of inputs with their corresponding output. The number of examples used is crucial – too few and the model lacks context of our domain, too many and the model might overfit or stray from key instructions. We tested with up to four unique examples, but through testing found that two examples perform best.

The processed input data is provided at the very end of the prompt, following the same structure as the examples with clear separation of input and output. The final prompt can be seen in the appendix.

Fine-tunning

We began by converting a set of .rtf files to .txt format for easier processing. For each converted file, we extracted structured information from Excel sheets for the full year of 2024. We ignored the entries where the input or the expected output was empty. This resulted in approximately 7,000 rows of data, each consisting of an input data and the corresponding expected report. We split the dataset into 90% for training and 10% for validation.

For training, we used the GaMS-9B-Instruct model.Each data sample was tokenized using the model's tokenizer, with both input and output sequences concatenated and truncated to a maximum length of 512 tokens. The model was fine-tuned using Low-Rank Adaptation.

Training was performed over 5 epochs with a batch size of 2 per device, using mixed precision to further optimize performance. Throughout training, we observed a steady decrease in loss, starting at 1.0557 and dropping to around 0.304 by the end of the 5th epoch. The final training loss averaged approximately 0.41, indicating successful convergence.

Because we trained the model solely on the available data, we found that using only the raw input data for prompting without additional instructions led to better results, as extra instructions tended to confuse the model.

Results

The evaluation using ROUGE metrics compares the performance of the prompt-engineered (Table 1) and fine-tuned models (Table 2), each tested on 30 generated outputs. The fine-tuned model consistently outperformed the prompt-engineered version in most metrics. ROUGE-1 F1-score increased notably, indicating improved identification of relevant unigrams. Similarly, ROUGE-L and ROUGE-Lsum scores were higher, suggesting better sentence structure and overall summary coherence. However, ROUGE-2 scores remained almost the same between the two models, indicating that both approaches struggled similarly with generating fluent bigrams and natural-sounding word combinations. These results show that fine-tuning enhanced informativeness and structure, though improvements in fluency were limited.

Metric	Precision	Recall	F1-Score
ROUGE1	0.3487	0.4091	0.3555
ROUGE2	0.2196	0.2558	0.2220
ROUGEL	0.2915	0.3408	0.2955
ROUGELsum	0.3245	0.3806	0.3303

Table 1. Results of ROUGE metrics over 30 instances using the base model with prompt engineering.

Metric	Precision	Recall	F1-Score
ROUGE1	0.4180	0.4503	0.4207
ROUGE2	0.2194	0.2378	0.2211
ROUGEL	0.3206	0.3444	0.3221
ROUGELsum	0.3854	0.4145	0.3875

Table 2. Results of ROUGE metrics over 30 instances using the fine-tuned model.

The ROUGE metrics have many limitations, as they lack any semantic understanding and fail to recognize differently

phrased outputs with similar meaning as correct. To better understand the practical differences between the two approaches, we manually evaluated 60 generated outputs, 30 from prompt engineering and 30 from the fine-tuned model. Each output was rated across five criteria, using a 1–5 scale:

- Accuracy: does the output correctly present facts in relation to ground truth, without hallucinations?
- Content coverage: how much of ground truth is covered in the output?
- Conciseness: how efficient is the output in communicating its message?
- Coherence: how well is the output structured, does it logically connect sentences and make sense as a whole?
- Readability: how easy is the text to read and understand, is it grammatically correct?

Metric	Prompt engineering	Fine-tuning
Accuracy	1.47	1.23
Content coverage	1.67	1.23
Conciseness	4.07	4.33
Coherence	3.53	4.43
Readability	3.83	4.57

Table 3. Results of manual evaluation over 30 instances using both approaches, showing average scores (1-5).

As shown in Table 3, the fine-tuned model generally performed better in staying on topic, consistently producing outputs focused on road traffic reports. In contrast, the prompt-based method occasionally generated content unrelated to the task.

However, the fine-tuned model did not outperform the prompt-based approach in all areas. It scored lower on Accuracy and Content Coverage, likely due to limitations or inconsistencies in the training data. Still, it showed notable improvements in Conciseness, Coherence, and Readability, making the outputs easier to understand and more structured.

These findings suggest that while the model can produce syntactically correct and well-structured traffic reports, its performance is influenced by the quality and consistency of the training data, which contains noise and inconsistencies.

Discussion

There are still many ways to improve upon the current solution: prompt engineering can be improved by trying through different prompt structures and instructions and finding optimal sets of parameters. The model can also be fine-tuned using more data and using altered parameters. The data itself could be further streamlined through detecting and removing less relevant data that is unlikely to make it to the traffic report. Unfortunately, there are many instances where an unavoidable discrepancy between input data and ground truth reports occurs. This means that no matter what optimizations we employ, it is simply impossible to output information that is not mentioned in the input. For this, we would require additional inputs.

Another way to obtain better results is by testing our solution using other LLMs. Despite GaMS's training on Slovene data, there are other LLMs that have stronger reasoning skills and may be able to follow the logic of our prompts more accurately. We could then compare the performances between general purpose multilingual LLMs and focused LLMs designed for a specific language.

References

[1] Huggingface. Gams 9b, 2025.

[2] Domen Vreš, Martin Božič, Aljaž Potočnik, Tomaž Martinčič, and Marko Robnik Šikonja. Generative model for less-resourced language with 1 billion parameters. *Zenodo*, 2024.

[3] Dong Yuan, Eti Rastogi, Gautam Naik, Sree Prasanna Rajagopal, Sagar Goyal, Fen Zhao, Bharath Chintagunta, and Jeffrey Ward. A continued pretrained llm approach for automatic medical note generation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 565–571, 2024.

[4] Xinxin Guan, Zekai Ye, Bin Cao, and Jing Fan. Summary report auto-generation based on hierarchical corpus using large language model. *SSRN*, 2023. Available at SSRN: https://ssrn.com/abstract=5095836 or http://dx.doi.org/10.2139/ssrn.5095836.

[5] Aofeng Su, Aowen Wang, Chao Ye, Chen Zhou, Ga Zhang, Gang Chen, Guangcheng Zhu, Haobo Wang, Haokai Xu, Hao Chen, et al. Tablegpt2: A large multimodal model with tabular data integration. *arXiv preprint arXiv:2411.02059*, 2024.

[6] Yilun Zhao, Haowei Zhang, Shengyun Si, Linyong Nan, Xiangru Tang, and Arman Cohan. Investigating table-to-text generation capabilities of llms in real-world information seeking scenarios. *arXiv preprint arXiv:2305.14987*, 2023.

[7] Yuan Tian, Chuhan Zhang, Xiaotong Wang, Sitong Pan, Weiwei Cui, Haidong Zhang, Dazhen Deng, and Yingcai Wu. Respark: Leveraging previous data reports as references to generate new reports with llms. *arXiv preprint arXiv:2502.02329*, 2025.

Appendix

Final prompt

```
# NAVODILA
Sestavi prometno poročilo o stanju na slovenskih cestah, ki naj vsebuje vse
pomembne informacije podane v vhodnih podatkih.
Poročilo mora:
- povzeti podatke iz vhoda
- biti napisano samo v naravni slovenščini

Obnovljeni podatki naj imajo naslednjo obliko:
Cesta in smer + razlog + posledica in odsek
ali
Razlog + cesta in smer + posledica in odsek

# Primeri vhoda (podatkov) in njegovega izhoda (poročila)
# Vhod:
Nesreče: Cesta Novo mesto – Šentjernej je zaprta pri odcepu za Dolenje Mokro Polje.
Ovire: Zaradi vozila v okvari je zaprt en pas na cesti Črnuče – Ljubljana,
proti krožišču Tomačevo.
Zastoji: Na hrvaški strani mejnih prehodov Dragonja in Sečovlje proti Sloveniji.
Splošno: Do 22. ure velja omejitev prometa tovornih vozil, katerih največja
dovoljena masa presega 7,5 t.
# Izhod:
Cesta Novo mesto – Šentjernej je zaradi prometne nesreče zaprta pri odcepu za
Dolenje Mokro Polje.
Na cesti Črnuče – Ljubljana je proti krožišču Tomačevo zaradi pokvarjenega vozila
zaprt en pas.
Pred mejnima prehodoma Dragonja in Sečovlje je zastoj proti Sloveniji.
Do 22-ih velja omejitev prometa tovornih vozil, katerih največja dovoljena masa
presega 7 ton in pol.

# Vhod:
Opozorila: Cesta Rateče – Planica bo zaprta danes do 18. ure
Zastoji: Na hrvaški strani mejnih prehodov Dragonja in Sečovlje proti Sloveniji.
Ovire: Zaprt počasni pas na primorski avtocesti zaradi okvare vozila pri priključku
Kastelec proti Ljubljani.
Dela: Delovne zapore na cesti  Ljubljana – Zagorje: – pri Beričevem bo zaprta do
24. ure.
# Izhod:
Cesta Rateče – Planica bo zaprta še do 18-ih.
Pred mejnima prehodoma Dragonja in Sečovlje je zastoj proti Sloveniji.
Na primorski avtocesti je zaradi okvare vozila zaprt počasni pas pri priključku
Kastelec proti Ljubljani.
Zaradi del so krajši zastoji na južni ljubljanski obvoznici med priključkoma
Rudnik in Center proti Kozarjam.

# NALOGA
Zdaj v izhodu sestavi prometno poročilo. Vsebovati mora naslednje vhodne podatke:
# Vhod:
(input data)
# Izhod:
```