Deep Learning-based Text Mining for Technology Monitoring in the Automotive Domain

Background – Technology Monitoring in Automotive Production

The automotive industry is undergoing a transformative phase with the integration of advanced technologies and the rise of intelligent manufacturing systems. To remain competitive in this dynamic landscape, automotive production requires effective utilization of technology monitoring as a part of technology intelligence, which encompasses the acquisition, analysis, and application of relevant technological information. By harnessing NLP techniques, automotive manufacturers can extract valuable insights from vast amounts of unstructured textual data available in the form of patents, research papers, publicly funded projects, and industry news. The goal of the Text2Tech research project is to develop methods for automated extraction of technologies and its relations to other entities from unstructured text sources. We formalize this task as a combinati on of Named Entity Recognition (NER, Yadav et al., 2018) and Relation Extraction (RE, Bach et al., 2020). Both NER and RE are fundamental, well-researched tasks in Natural Language Processing, however, their application to novel domains such as automotive manufacturing is often hindered by the lack of training and evaluation data. Prior research has shown the promising performance of Large Language Models (LLM) in such low-resource scenarios, e.g. for approaches based on few-shot learning (Fritzler et al., 2019) and instruction-tuning (Wang et al., 2023). In this study, we compare prompting and fine-tuning Large Language Models on end-to-end NER and RE.

Methods

Relation Extraction (RE) is a natural language processing task that involves identifying and extracting semantic relationships or interactions between entities mentioned in a document. Traditionally RE has been approached in two steps. First, Named Entity Recognition (NER) extracts entities from a given document. Second, Relation Classification (RC) evaluates whether given entities share a relationship. Recent approaches combine these two steps as an end-to-end task using generative Large Language Models (LLMs).

We evaluate the end-to-end approach in our low-resource scenario. For NER, we define the following entities: Technological System, Material, Method, Technical Field, and Organization. The goal of RE is to recognize the specific types of relationships between these entities. Possible relationships are, for example, "develops", "part of", or „uses". We implement and compare two models - a prompt-based approach and a fine-tuning approach on semi-automatically labelled data.

Prompt-based NER & RE:

As a baseline approach, we prompt Large Language Models (LLMs) to perform both NER and RE in a single, end-to-end fashion. We construct zero-shot prompts that carefully describe the two tasks and augment the task instruction with strategies such as adding few-shot examples, including chain-of-thought instructions, asking the model to explain its output, and asking the model to provide JSON-formatted output. We used these prompts to evaluate several LLMs, namely GPT-3.5 & 4.5, BARD, BART, Llama-2 & 3, and a smaller model, Roberta-Base, as a baseline.

Fine-tuned NER & RE:

Fine-tuning LLMs promises to outperform prompt-based approaches given the domain specificity of the data. We start by querying a general LLM (GPT-3.5) to perform NER and RE similar to the prompt-based approach described above. The resulting entities and their relations will be used to fine-tune another LLM. The results will then be manually reviewed, adjusted, and used as new input to further fine-tune the LLM in an iterative process. We benchmark two models against each other by measuring their ability to learn our defined relations. The first model is BART-large pre-trained for RE using the REBEL (Cabot et al., 2021) approach. The second is the much bigger, however not pre-trained for RE model Mistral 7B Instruct v0.2. The final goal of the fine-tuning process is to obtain a single model which 1) performs NER and subsequently RE in one pass, 2) outperforms prompt-based approaches with commercial LLMs, and 3) is on par with human annotators.

Experiments & Results

Prompt-based NER & RE:

For NER, GPT3.5 outperformed the other models and achieved a micro F1 of 0.421 with few-shot examples, and a micro F1 of 0.458 with additional chain-of-thought prompting. For prompt-based RE, we are currently in the process of constructing a test dataset. Results will be available at the time of the conference.

Fine-tuned NER & RE:

We compared REBEL's ability to learn the GPT 3.5 labeled data with Mistral's ability to learn the same task. The training, validation, and test dataset consists of 650, 150, and 200 documents each. The greater the ability of the model to learn the task the more useful the model is for an iterative process of improving the model with manual reviewed data iteratively.

REBEL is pre-trained on Wikipedia abstracts to extract entities and 200 different relation types in one forward pass. We fine-tune the model with our training data for 12 epochs. The best F1 score on the test data is reached after 6 epochs. Afterwards, we observe overfitting. A data efficiency evaluation indicates that more training data won't further improve the quality. On the contrary, training runs with 20% and 40% of the training data result in comparable results. At best, our fine-tuned REBEL achieves a F1 score of 0.496 for NER and a F1 score of 0.033 for RE compared to the GPT 3.5 generated gold labels.

With the given set of parameters, Mistral underperforms REBEL with a F1 score of 0.334 for NER but outperforms REBEL with a F1 score of 0.082 for RE. These results are still work in progress but will be available at presentation time. We fine-tuned Mistral using Low-Rank Adaptation of Large Language Models (LoRa) in 4-bit quantized (QLoRA).

Manual labeling:

F1 scores of each approach were comparable low to other NER and RE tasks. To put the results into context, we labeled 20 texts manually using five annotators. The inter annotator agreement was low with a Krippendorff's Alpha of 0,0129. The pairwise F1 score between two annotators ranged from 24.31 to 51.74 for the NER task averaging at 39.52. The pairwise F1 score for the RE task ranged from 1.12 to 14.34 averaging at 7.93. The level of disagreement shows the complexity of the task.

Discussion & Conclusion

In addition to the tasks of NER and RE, we are currently preparing a dataset for the task of Entity Linking. Entity Linking is an important step to resolve ambiguous entity mentions and to map lexicographical variants of the same concept to a normalized reference (DeCao et al., 2020).

References

Bach, N. & Badaskar, S. (2007). “A Review of Relation Extraction”. Online.

De Cao, Nicola, Gautier Izacard, Sebastian Riedel and Fabio Petroni. “Autoregressive Entity Retrieval.” ArXiv abs/2010.00904 (2020)

Fritzler, A. & Logacheva, V. & Kretov, M. (2019). “Few-shot classification in named entity recognition task.” *In Proc. ACM/SIGAPP Symposium on Applied Computing (SAC '19)*. ACM, pages 993–1000.

Kumar, A. & Starly, B. (2021). “FabNER”: information extraction from manufacturing process science domain literature using named entity recognition.” Journal of Intelligent Manufacturing 33: 2393 - 2407.

Cabot, Pere-Lluís Huguet and Roberto Navigli. “REBEL: Relation Extraction By End-to-end Language generation.” Conference on Empirical Methods in Natural Language Processing (2021).

Wang, Y., Ivison, H., Dasigi, P., Hessel, J., Khot, T., Chandu, K. R., Wadden, D., MacMillan, K., Smith, N. A., Beltagy, I. & Hajishirzi, H. (2023). “How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources.” *CoRR* abs/2306.04751.

Yadav, V. & Bethard, S. (2018). “A Survey on Recent Advances in Named Entity Recognition from Deep Learning models.” *In Proc. COLING,* pages 2145–2158, ACL.