Research plan for Pre-trained language model approach to generate natural sentences from the knowledge graph

Jiachen Li
1068299

1. Motivation

Natural language generation (NLG) was a sub-problem of the natural language processing (NLP) problem family. NLG aimed to generate human-readable natural language from unstructured data such as tables, speeches, images, etc. The NLG field of research has existed for a long time because it plays an essential role in areas such as Artificial Intelligence or question-answering systems. But it has recently gained significantly more attention due to the popularity of the rage of Metaverse [1], which is expected to have a compound annual growth rate of 43.20% in the market value between 2022 to 2030, and eventually reach a spectacular 39.25 billion USD market value in 2030 [2]. The NLG technology will be the foundation of the next-generation human-computer interactive question-answering system to provide sufficient immersion for the Metaverse.

In this research, the knowledge graph will be the type of unstructured data we generate natural sentences from. It was constructed from a network of Resource Description Framework (RDF) triple in format <subject, predicate, object>. For example, a triple would be: <Li, residence, Beijing>, and in the context of NLG, the generated sentence could be: “Li lives in Beijing”.

The state-of-art solution of NLG from the knowledge graph problem is GTR-LSTM, proposed by Trisedya et al. [3]. The GTR-LSTM took advantage of the neural network encoder-decoder-based transformer [4] approach that outperformed previous non-neural approaches in the performance and can be generalised to any domain while requiring less human-labelled training data.

Following up on the GTR-LSTM model, more research was committed to improving its performance further. On the input side of the model, Trisedya et al. first flipped the problem

to propose a neural-based model for relation extraction for knowledge graph enrichment to expand the training pool [5]. Then, they studied entity alignment to match the same real-world entity from different knowledge bases to merge the network and expand the knowledge bases [6]. Finally, working with Zhang et al. proposed a novel benchmark framework to evaluate various entity alignment strategies [7]. On the output side, Trisedya et al. integrate content planning in the encoder side of the transformer attention model for more fluent natural language with correct entity order [8]–[10]. All the above strategies to improve the performance of GTR-LSTM were considered as part of the pipeline along with the GTR-LSTM model itself to solve the NLG from knowledge graph problems.

Another trend in the NLP field in recent years was pre-trained models, described by Qiu et al. [11] as the emergence of pre-trained models "has brought natural language processing to a new era." The models were pre-trained on the large corpus to learn the universal language representations used in downstream tasks. Benefiting from the extensive training corpus, the models were advantageous in transferability to different types of downstream jobs, generalisation to avoid overfitting and reducing the training time cost.

As mentioned, Trisedya et al. have built a pipeline with multiple stages for solving NLG from knowledge graph problems with numerous research publications. The stage in each paper was further divided into fundamental NLP problems and solved using various basic NLP techniques such as N-gram, lexicon lookup, etc. However, the pre-trained models did not appear to be considered to solve any of the fundamental problems, even though the pre-trained models have been proven to be outperformed traditional approaches in various primary NLP tasks [11, Sec. 7]. The research was motivated by this fact to study if pre-trained models can improve the GTR-LSTM-based pipeline.

2. Research Question

Can the pre-trained language models be used to improve the current solution to natural sentence generation from the knowledge graph problem?

As discussed in the previous section, the problem of natural language generation from the knowledge graph was complicated but can be recursively broken into more minor problems. In this research, we identified two tasks at the atomic level that potentially can be better

solved using a pre-trained language model and aimed to evaluate the effect of the replacement in various aspects.

3. Method

We will use the Bidirectional Encoder Representations from Transformers (BERT) proposed by Devlin et al. from Google [12] as the pre-trained language model in this research. The model was designed based on a multi-layer bidirectional Transformer encoder and trained on unlabeled text from BooksCorpus (800M words) [13, pp. 19–27] and English Wikipedia (2,500M words), which are two extremely large corpus[12, p. 5]. The BERT preliminary learnt bidirectional text representation could be applied to any downstream NLP task by simply adding an output layer to the existing neural architecture [14, p. 2].

3.1 Scenario 1: RDF triple pre-processor entity mapper

Besides the encoder-decoder module, the GTR-LSTM framework proposed by Trisedya et al. [3, Sec. 3.1] also consists of two pre-processors for input RDF triple and output text. The RDF triple pre-processor can map entities in a triple to its type so that the output sentence pattern can learn based on entity type to better handle unseen test cases. In Li's example mentioned before, Li will be mapped to “PERSON” and Beijing to “CITY”. Therefore, Li can be replaced with any unseen person’s name while preserving the sentence structure. The solution in past literature was based on the DBpedia lookup API through the internet [3, Sec. 3.6]. In the NLP field, the task could be classified as a named entity recognition (NER) task and can be solved using BERT. The plan was to replace DBpedia lookup API with the NER-BERT model proposed by Liu et al. [15], which was based on BERT but fine-tuned for NER problems and was claimed to outperform BERT and other baseline traditional approaches significantly, especially when pre-training data in a particular domain were limited [15, Sec. 8]. In the context of the knowledge graph, the abundant non-repeated Proper Noun only made a little appearance in the training set, which has been precisely addressed by NER-BERT.

3.2 Scenario 2: entity alignment entity identifier

The entity alignment stage in the pipeline was to identify entities with different text representations representing the same real-world entity and merge them to expand our knowledge of the entity. This can happen within a knowledge graph or between different knowledge graphs developed by different origination under different standards. In Li’s

example, besides triple <Li, residence, Beijing>, we have another triple <Peking, Capital, China> in another knowledge graph. A successful entity alignment will realise that Peking and Beijing are the same entity in the real world and answer the question, “Which country does Li live in?” with “Li lives in China.”

Trisedya et al. [6, Sec. 4.3] proposed a solution using Joint Learning of Structure Embedding and Attribute Character Embedding to identify the same entities. The Structure Embedding was learned from the relation of the entities; for example, from triples recording the longitude and latitude of Beijing and Peking, the model may realise they are the same city as they locate in the same place. The Attribute Character Embedding was based on the entity name itself, which works better with numbers having different reserve decimals or misspelled words. The idea was also referred to as syntactic and semantic analysis in the NLP field. We found that BERT was exceptionally well at capturing syntactic and semantic knowledge [16, Sec. 3], [17, Sec. 5] that its contextualised embeddings could address the word sense disambiguation [18, Sec. 6], which means that word vectors are clustered based on the real-world entity its representing, so Beijing and Peking will be close in embedding space. Hence, we propose to use cosine similarity between entity embedding vectors pre-trained from the BERT model to decide if two entities are identical for merging.

4. Analysis

In general, the paper will use the original models proposed in previous literature as the baseline to examine if our proposed model provides any improvement. The primary metric will be the model's performance, which is the accuracy or how well the model solves the problem. The performance will be evaluated using the identical benchmark proposed in the corresponding paper to compare results from our model with claimed results. Our proposed model may fail to improve the accuracy over the original model or, even worse, dramatically reduce the accuracy. Therefore, We would also evaluate our model from various dimensions besides accuracy. We would obtain the source code provided by previous literature to evaluate and compare with our model in other metrics, such as running time to evaluate the proposed model thoroughly, so the audience of this research can make compromises when choosing which model to use in various use cases. The detailed scenario-based evaluation will be discussed in the following sub-sections.

4.1 Scenario 1 evaluation

We will use the original GTR-LSTM model proposed by Trisedya et al. [3, Sec. 4] as the baseline, which claimed to have scores of 40.1, 34.6, 50.6 in metrics of BLEU [19], METEOR [20], and TER [21] correspondingly using the GKB dataset. The benchmarks will be recreated to validate the claim scores while recording other metrics, such as training time, prediction time, memory usage and internet usage because the solution relied on API calls. The proposed model will be run and evaluated in the same environment for comparison. The accuracy of our model is expected to increase by a small amount but also has a considerable chance of dropping accuracy as the API is usually reliable and updated. Also, running BERT will likely consume more memory. However, we expect the improvement from significantly reduced training time and prediction time as the BERT is pre-trained and will save the internet response time consumed by API calls.

3.2 Scenario 2 evaluation

The baseline model proposed by Trisedya et al. [6, Sec. 5] was evaluated using four real-world datasets: DBpedia (DBP) [22], LinkedGeoData [23], Geonames (GEO) [24], and YAGO [25]. The performance was then measured using the proportion of correctly aligned entities ranked in the top 1 or 10 predictions and the mean of the rank of the correct prediction. This identical benchmark will be applied to our model and Trisedya's model for accuracy comparison. However, as pointed out by Zhang et al. [7, Sec. 7.1], the existing datasets being used have significant limitations, namely bijection, lack of name variety and small scale. Therefore, the benchmark DWY-NB proposed by Zhang et al. Field [7, Sec. 7.3] will also be applied to both models for a more comprehensive evaluation. We expect our BERT model to perform better and run faster because it was pre-trained on a significantly larger corpus.

3.2 Limitation and future work

The scope of this research is limited to using the basic BERT trained on natural sentences corpus without being fine-tuned for unstructured text. Therefore the selected two tasks serve more as general auxiliaries in the pipeline as Trisedya et al. have already fine-tuned solutions of essential tasks in the pipeline specifically for the NLG from a knowledge graph setting; for example, the encoder of the GTR-LSTM model, not only has been optimised for triple structure but also has been integrated with content-planning to obtain the relation and order of

the entities [9]. Therefore, in future work, we will fine-turn the BERT model for specific settings to solve more essential tasks in the pipeline to stimulate better performance.

5. Contribution

Even though the pipeline of solving natural language generation from the knowledge graph proposed by Trisedya et al. has outperformed previous approaches dramatically, the contribution of this research is expected to improve the performance further by utilising the state-of-art pre-trained BERT model. Ideally, the proposed model would serve as an update to improve and replace the GTR-LSTM-based pipeline's component. However, realistically, even if our model failed to improve the performance, a minor improvement in other metrics without compromising performance too much can still contribute as an alternative choice for certain application scenarios under limited constrain. The faster training prediction time can help the model run in real time on portable VR devices. Avoiding internet-based API lookup calls makes the model work in settings where internet access is unavailable or too expensive. Combining two improvements makes the model easier to scale up for cloud computing.

6. Reference

[1] S. Mystakidis, "Metaverse," *Encyclopedia*, vol. 2, no. 1, Art. no. 1, Mar. 2022, doi: 10.3390/encyclopedia2010031.

[2] "Metaverse Market Size by Technology, Component, End-User, Regions, Global Industry Analysis, Share, Growth, Trends, and Forecast 2022 to 2030 | The Brainy Insights." https://www.thebrainyinsights.com/report/metaverse-market-12815 (accessed Sep. 04, 2022).

[3] B. D. Trisedya, J. Qi, R. Zhang, and W. Wang, "GTR-LSTM: A Triple Encoder for Sentence Generation from RDF Data," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia, 2018, pp. 1627–1637. doi: 10.18653/v1/P18-1151.

[4] A. Vaswani *et al.*, "Attention is All you Need," in *Advances in Neural Information Processing Systems*, 2017, vol. 30. Accessed: Sep. 04, 2022. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[5] B. D. Trisedya, G. Weikum, J. Qi, and R. Zhang, "Neural Relation Extraction for Knowledge Base Enrichment," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, 2019, pp. 229–240. doi: 10.18653/v1/P19-1023.

[6] B. D. Trisedya, J. Qi, and R. Zhang, "Entity Alignment between Knowledge Graphs Using Attribute Embeddings," *Proc. AAAI Conf. Artif. Intell.*, vol. 33, no. 01, Art. no. 01, Jul. 2019, doi: 10.1609/aaai.v33i01.3301297.

[7] R. Zhang, B. D. Trisedy, M. Li, Y. Jiang, and J. Qi, "A Benchmark and Comprehensive Survey on Knowledge Graph Entity Alignment via Representation Learning." arXiv, May 05, 2022. doi: 10.48550/arXiv.2103.15059.

[8] B. Trisedya, J. Qi, and R. Zhang, "Sentence Generation for Entity Description with Content-Plan Attention," *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 05, Art. no. 05, Apr. 2020, doi: 10.1609/aaai.v34i05.6439.

[9] B. D. Trisedya, J. Qi, W. Wang, and R. Zhang, "GCP: Graph Encoder with Content-Planning for Sentence Generation from Knowledge Base," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2021, doi: 10.1109/TPAMI.2021.3118703.

[10] B. D. Trisedya, X. Wang, J. Qi, R. Zhang, and Q. Cui, "Grouped-Attention for Content-Selection and Content-Plan Generation," in *Findings of the Association for Computational Linguistics: EMNLP 2021*, Punta Cana, Dominican Republic, 2021, pp. 1935–1944. doi: 10.18653/v1/2021.findings-emnlp.166.

[11] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang, "Pre-trained Models for Natural Language Processing: A Survey," *Sci. China Technol. Sci.*, vol. 63, no. 10, pp. 1872–1897, Oct. 2020, doi: 10.1007/s11431-020-1647-3.

[12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." arXiv, May 24, 2019. doi: 10.48550/arXiv.1810.04805.

[13] Y. Zhu *et al.*, "Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books," presented at the Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 19–27. Accessed: Nov. 03, 2022. [Online]. Available: https://www.cv-foundation.org/openaccess/content_iccv_2015/html/Zhu_Aligning_Books_and_ICCV_2015_paper.html

[14] M. V. Koroteev, "BERT: A Review of Applications in Natural Language Processing and Understanding." arXiv, Mar. 22, 2021. doi: 10.48550/arXiv.2103.11943.

[15] Z. Liu, F. Jiang, Y. Hu, C. Shi, and P. Fung, “NER-BERT: A Pre-trained Model for Low-Resource Entity Tagging.” arXiv, Dec. 01, 2021. doi: 10.48550/arXiv.2112.00405.
[16] A. Rogers, O. Kovaleva, and A. Rumshisky, “A Primer in BERTology: What We Know About How BERT Works,” *Trans. Assoc. Comput. Linguist.*, vol. 8, pp. 842–866, 2020, doi: 10.1162/tacl_a_00349.
[17] I. Tenney, D. Das, and E. Pavlick, “BERT Rediscovers the Classical NLP Pipeline.” arXiv, Aug. 09, 2019. doi: 10.48550/arXiv.1905.05950.
[18] G. Wiedemann, S. Remus, A. Chawla, and C. Biemann, “Does BERT Make Any Sense? Interpretable Word Sense Disambiguation with Contextualized Embeddings.” arXiv, Oct. 01, 2019. Accessed: Nov. 05, 2022. [Online]. Available: http://arxiv.org/abs/1909.10430
[19] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “Bleu: a Method for Automatic Evaluation of Machine Translation,” in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, Pennsylvania, USA, 2002, pp. 311–318. doi: 10.3115/1073083.1073135.
[20] M. Denkowski and A. Lavie, “Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems,” in *Proceedings of the Sixth Workshop on Statistical Machine Translation*, Edinburgh, Scotland, 2011, pp. 85–91. Accessed: Sep. 25, 2022. [Online]. Available: https://aclanthology.org/W11-2107
[21] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, “A Study of Translation Edit Rate with Targeted Human Annotation,” in *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, Cambridge, Massachusetts, USA, Aug. 2006, pp. 223–231. Accessed: Sep. 25, 2022. [Online]. Available: https://aclanthology.org/2006.amta-papers.25
[22] J. Lehmann *et al.*, “DBpedia – A large-scale, multilingual knowledge base extracted from Wikipedia,” *Semantic Web*, vol. 6, no. 2, pp. 167–195, 2015, doi: 10.3233/SW-140134.
[23] C. Stadler, J. Lehmann, K. Höffner, and S. Auer, “LinkedGeoData: A core for a web of spatial open data,” *Semantic Web*, vol. 3, no. 4, pp. 333–354, 2012, doi: 10.3233/SW-2011-0052.
[24] “GeoNames Ontology - Geo Semantic Web.” http://www.geonames.org/ontology/documentation.html (accessed Nov. 05, 2022).
[25] J. Hoffart, F. M. Suchanek, K. Berberich, and G. Weikum, “YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia,” *Artif. Intell.*, vol. 194, pp. 28–61, Jan. 2013, doi: 10.1016/j.artint.2012.06.001.