Master Thesis Review

Charles University, Faculty of Mathematics and Physics

Thesis author	Peter Grajcar		
Thesis title	Data-to-Text Generation With Text-Editing Models		
Submitted	2023		
Program	Computer Science	**Specialization**	Language Technologies and Computational Linguistics

Review author	Dušan Variš	**Role**	reviewer
Position	Institute of Formal and Applied Linguistics		

Review text:

The thesis focuses on the problem of data-to-text generation using the text-editing editing models and expands the existing state-of-the-art approaches, namely the pipeline introduced by Kasner and Dušek (2022) and the FELIX text-editing model. The student proposes several improvements to the model, such as autoregressive decoding, decoding constraints and clause reordering using pointer networks. Furthermore, they propose several data augmentations to the existing WebNLG and DiscoFuse datasets and measure the their effects on the model training. The student measures the performance of the investigated methods using both the automatic metrics and a small scale manual evaluation. They achieve improvements in fluency compared the basic pipeline, however, they do not surpass their comparative baseline, the End-to-End BART model. Their manual evaluation shows that both the proposed methods and the End-to-End baseline tend to generate similar amount of hallucinations, however, the nature of the hallucinations is different.

The overall structure of the thesis is fairly organized, although, the structure of the sections describing the architecture details and the experiments would benefit from more refinement and verbosity. The thesis is fairly short (37 pages without citations) and these sections would be the major candidates for a more in-detail description. Student proposes improvements to several steps of the existing text-editing pipeline, however, understanding the individual contributions was quite difficult at first, possibly due to a lack of a more general overview of the pipeline. Even though the details can be found in the cited related work, the thesis would benefit from its own breakdown of the individual steps of the pipeline with the proposed modifications. Instead, the student describes individual steps at various parts of the thesis and often uses back-references where needed. On the other hand, I really appreciated the included examples, namely the examples of the data (e.g. Table 2.1, 3.1) and the clause extraction (Figure 3.2) which made the description in the related sections much more comprehensible. Still, regarding the clause extraction section, examples of the other two types of handling, similar to Figure 3.2 would be helpful but were missing.

In Chapter 2, the student provides a brief description of the Transformer architecture and refers the reader to the original publications for more details. I found it odd that the student avoids using the usual nomenclature such as self-attention or encoder-decoder attention when describing the network architecture. When describing the FELIX model extension, the student mentions “replacing the feed-forward layer on top of the BERT encoder”. I found this description insufficient because it is not completely clear which BERT encoder they are referring to (there are two independent BERT models in FELIX, mentioned in Section 3.3). A more detailed schematic (similar to Figure 3.1) would help clarify the modifications proposed by the student.

In Section 4.2.1, the student describes an alternative triple-to-sentence generation through the fine-tuned BART model. The details about the model fine-tuning, i.e. tuning hyper-parameters are missing. It seems that they only evaluated this generator extrinsically (by including it in the existing pipeline), however, no intrinsic evaluation of the generator itself (i.e. comparison with the rule/template-based system) is not mentioned. Later in Section 4.4, they describe the models used for the data-to-text generation. The hyper-parameters of the BART End-to-End Baseline are listed, however, reasoning behind the choice of hyper-parameters could be described in more detail. In the following experiment setups, the model parameters are not listed directly due to them being based on the varying previous work which is properly cited.

The thesis is missing more detailed statistics about the used datasets, mainly the sizes of the datasets before (WebNLG missing) and after the filtering and/or clause extraction. Only the distribution of the number of necessary edits is provided. Although some of these basic statistics are not presents, the student later (during evaluation) provides the edit label distribution for the WebNLG testset and a comparison of the distributions of connectives between the DiscoFuse and WebNLG datasets. The numbers in the DiscoFuse histogram (Figure 4.2, right) does not seem to add up - the student mentions in Section 3.2.5 that the dataset contains more than 16M examples, however, the summation of the values in the figure is roughly less than 1M. Therefore, it is not clear how were the values in these figures obtained? Similarly, more details about the WebNLG testset (section 4.4.4) should be provided. I am also interested in the analysis of the effects of a potential domain (mis)match (DiscoFuse or WebNLG training, assuming WebNLG in the NLGI evaluation) on the presented results.

In the experiment section, the student investigates different text-editing pipeline modifications in four sets of experiments. First, they compare the effects of using different training datasets on the performance of the basic pipeline. Although they describe several dataset augmentations such as filtering, oversampling or clause extraction, they only compare the original WebNLG and DiscoFuse datasets, their pretraining+fine-tuning combination and the filtered DiscoFuse. The performance of the basic pipeline using the other proposed training data is missing even though a direct side-by-side comparison of the other training data listed in Section 4.4.2 with the basic setup seems reasonable. Similarly, it would interesting to see how different training data affect the BART End-to-End baseline.

In the following experiment (Table 5.2), the student shows that the WebNLG-oversampled dataset is not as effective as the original DiscoFuse dataset in combination with the autoregressive decoder. Still, they show the improvements gained by using autoregressive decoding that are further improved by enforcing the generation of the input triples resulting in the reduction of the number of generated hallucinations. As mentioned earlier, one thing missing is evaluation of the base model with the implemented triples enforcement. While I understand the need for adjustment of the original setup, the results of this setting would shed more light on the effectiveness of this specific decoding constraint.

The experiment related to clause extraction/reordering (Table 5.3) did not show any significant improvement. I am not sure about the motivation behind the clause extraction - even though it probably results in generating more “simple” training data, more similar to DiscoFuse, the resulting examples should be less similar to the target domain (WebNLG). I would also be intereted whether the student considered some additional analysis of the provided results (listed in the questions at the end of this review).

The final experiment (Table 5.4) shows that extending triple-to-text generation by a separate, fine-tuned BART model can also be beneficial to the pipeline. The student combines the BART templates with the DiscoFuse + WebNLG clauses dataset. They remove clause reordering due to lack of improvement in the previous experiment. It is not clear, what is the motivation behind the choice of dataset (DiscoFuse + WebNLG clauses). In contrast, they could also choose the Filtered DiscoFuse with keep-triples instead, a setup that resulted in a better performance in terms of BLEURT metric and was only slightly worse in terms NLGI (3 OK example difference).

The final manual evaluation confirms the improvements gained by the proposed modifications in the terms of fluency. On the other hand, it showed that the proposed systems generate significantly more hallucinations than the basic model. Performing further analysis, the student argues that this discrepancy between the automatic and manual evaluation is caused mainly by the misuse of connectives and subjunctives. The student supports their claim by providing evidence of the difference between the distribution of connectives in the WebNLG and DiscoFuse datasets. However, the models in Table 5.5 were trained using the same dataset (DiscoFuse + WebNLG) and it is not clear why some setups are more prone to the misuse of connectives than the others. I am interested whether the student has another hypothesis that could setup potential future work.

To sum up, the experiments presented in the work were, with some marginal exceptions, reasonably motivated, fairly diverse, however in my opinion, could be better organized. The student demonstrated that the proposed modifications can lead to improvement in the terms of automatic metrics but these results were not confirmed by the manual evaluation. Personally, I would prefer a slightly more in-depth comparison of the proposed augmented training data in a more isolated setting (i.e. only with the basic pipeline) and an additional intrinsic evaluation of the BART-based templates. Furthermore, I am not sure whether the reasoning behind the explanation behind the results of the manual evaluation based on misuse of connectives is correct - I am curious about the other hypotheses behind this discrepancy. The main weakness of this thesis was mainly the presentation of the student's work. Given the short nature of the thesis, it would surely benefit from additional thought-out details, mainly in the sections describing the text-editing pipeline and the section describing the individual experiments with respect to the said pipeline, including the datasets.

The following is a list of questions related to the experiments within the thesis:

- How does the potential domain mismatch between the training (DiscoFuse) and test (WebNLG) datasets affect the presented results compared to the "in-domain" WebNLG models?
- Did you directly compare the BART-based template generation to the previous rule-based method? Is it possible to compare these methods intrinsically and how?
- There seems to be a discrepancy between the number of examples in the DiscoFuse dataset and the Values in Figure 4.2. How were the results in the figure obtained?
- Is it correct that the setup on line 2, 3 and 4 (Table 5.3) does not perform clause extraction during inference becuase the input is the concatenation of the sentencese created by templates from the set of input triples?
- What is the performance of the system trained on the "WebNLG clauses" dataset with gold/predicted ordering?
- How does ordering affect the Filtered DiscoFuse with "WebNLG clauses"?
- It is not clear, what is the motivation behind the choice of dataset (DiscoFuse + WebNLG clauses). Why did you not use the Filtered DiscoFuse with keep triples instead since it resulted in better performance in terms of BLEURT and only slightly worse performance in terms of NLGI (difference of 3 OK examples)?
- The models in Table 5.5 were trained using the same dataset (DiscoFuse + WebNLG) and it is not clear why some setups are more prone to the misuse of connectives. Do you have any other hypothesis, why the basic setup performed better in the manual evaluation?

Lastly, I have two questions regarding the theoretical part of the thesis:

- In section 2.2.4, attention use (2) - how can the autoregressive decoder mask the positions $> i$ when decoding $i - th$ position if the future positions have not yet been generated?
- In section 2.2.5, it is not clear in the description of the autoregressive top-k decoding, how it differs from the beam search decoding - how are the next-step input symbols chosen from the top-k token list?

I recommend the thesis for defense.

I suggest to not consider the thesis for the annual award.

In Prague, 22. 8. 2023

Signature: