Review of "Get To The Point: Summarization with Pointer-Generator Networks"

Kekoa Riggin
University of Washington
kekoar@uw.edu

Abstract

"Get To The Point: Summarization with Pointer-Generator Networks is an overall well-structured and thorough article. The limitations of extractive and abstractive summarization approaches is clearly outlined along with the problems with previous workarounds. The methods of the experiment are explained and distinguished from the attempts of previous text summarization experiments. The limitations regarding subjective measurements and domain are fairly considered as are the methodological variations of this approach from that of other applications and phenomena. The benefits of the approach in comparison with a baseline and previous results are presented with standard measurements and explained. Adapting the model to lean towards abstraction is recommended as the future of this project.

1 Problem: Extraction vs. Abstraction

State of the art text summarization technologies utilize one of two main methods: abstraction and extraction. The extraction method uses phrases from within the source text to build a shortened summary of the text while abstraction generates the summary rather than copy-pasting.

Most summarization technologies have utilized extractive techniques until recently as recurrent neural networks have made abstractive summarization more viable. However, although recurrent neural networks have increased performance on two-sentence level summarization datasets, extractive methods have been shown to outperform abstractive ones based on the ROUGE metric.

The authors explain that extractive methods often produce repetitive results and lack the ability to generate new words, which allows for more generation of short summarizations, but abstractive approaches may generate incorrect facts due to out-of-vocabulary words.

The solution tested in this experiment is a mixture of extractive and abstractive methods that utilizes a pointer-generator network to recognize words as copy words rather than trying to generate it. Using a mixture of copy and generated words and identifying them by category, this summarization approach maintains the ability to generate words while reducing the number of extra-factual summarizations due to out-of-vocabulary words.

2 Methods

Three modules belong to this experiment:

1. a baseline sequence-to-sequence model,
2. a pointer-generator model, and
3. a coverage mechanism that can be added to either of the first two models.

For each of the models, a detailed description is provided, which includes a step-by-step process of the states and the mathematical notation of the variables at each state. The state variables include the attention distribution, the weighted sum of the encoder hidden states (known as the context vector), the decoder state, the vocabulary distribution, the final distribution, the loss for timestep, and the overall loss.

Because the sequence-to-sequence model serves as the baseline for this experiment, the distinguishing features of the models are described as changes made in the pointer-generator network from the sequence-to-sequence model. The pointer-generator network includes the work from the previous model but also uses pointing to copy words from the source, meaning that a generation probability must also be processed.

The coverage mechanism is described as the feature that reduces repetition in the summaries. The coverage mechanism uses a coverage vector to penalize the pointer for attending to the same location, reducing the number of times a word or sections of the text can be copied.

Each model is described by the size of the vocabulary and the number of hidden states, which are based on the vocabulary and hidden states used by previous experiments. However, discrepancies between the models for this experiment and others are clearly outlined with further considerations regarding said discrepancies being discussed in the results.

The dataset used for the experiment is the same that is used by two previous summarization experiments: the CNN/Daily Mail dataset, a collection of online articles. The methods of obtaining the data are taken from one of the previous experiments, ensuring that the experiments can be compared objectively; however, the data for the previous experiments was anonymized while this experiment does not anonymize it. This is discussed as a possible discrepancy in the results, but they are still reported fairly and accurately.

The authors mention that portions of this approach as well as the data structures behind the modules have been used in NMT with reported success. However, there are several unique features of the modules that were designed specifically for the problem of extraction and abstraction in summarization. The differences and similarities to NMT approaches are listed in the article.

3 Results

The results from the experiment feature preliminary data and observations. The preliminary data focuses primarily on the ROUGE metrics but also makes use of training time and the number of n-grams generated by each module.

Regarding training time, the authors make it clear that the pointer-generator module was far more efficient, requiring less than half the time and iterations to complete.

Regarding the testing results, the authors reported the F scores for the ROGUE-1, ROGUE-2, and ROGUE-L measures as well as the measures for the METEOR metric. These metrics serve as the current standard for text summarization and are the same metrics used by previous experiments.

The results clearly showed increased performance in the pointer-generator module with coverage in comparison to the pointer-generator, the sequence-to-sequence and the abstractive models from previous experiments. These results are consistent between both the ROGUE and METEOR metrics.

There was one important discrepancy discussed in the Limitations section below.

The results regarding coverage showed that, regardless of the number of grams, the coverage results were far better than the modules without coverage. In fact, the coverage feature produced results with repetition numbers similar to those of the reference summaries.

The authors clearly present the results in the article and explain the causes of the improvement. The pointer-generator approach enables the summarizer to extract text from the source as well as generate words from the vocabulary and finally reduce repetition with the coverage feature. Detailed analysis of the performance of the pointer-generator is discussed in the article and is mentioned in the Limitations section below.

4 Limitations

The limitations of this experiment, as outlined by the authors, are the domain of the dataset and the ROGUE metric as a summary metric.

The CNN/Daily Mail dataset consists of news articles, which are written in a particular style for new media. According to the authors, the content style typically features important sentences at the beggining of the article. This resulted in high ROGUE and METEOR scores for the lead-3 experiment, which used only the first 3 sentences to create a summary. The authors claim that it is unlikely that the lead-3 method would produce accurate results on another dataset but that the pointer-generator method would produce similar results to what was seen in this experiment.

The authors also discuss the objectivity of the ROGUE metric. Because the ROGUE reference summaries are extracted from the text, the ROGUE score is biased towards results that contain structures and phrases that are found within the source text. Because the pointer-generator method allows for generation of word and structure, it is possible and plausible for the ROGUE score of an acceptable summary to be very low or 0.

These limitations are described in detail with examples and causes in the article and high-level solutions are provided by the authors, specifically, the need for more summarization datasets.

5 The Future of Summarization

The authors discuss the success of the experiement, considering the amount of abstraction that took place in the results. The findings show that the pointer-generator produced a much smaller number of n-grams, which indicates less abstraction. However, it seems that the machine takes a copy first, generate second approach, which allows the model to stich together summaries while maintaining gramaticality.

The authors confirm that the results of the experiment show that this mixture model is an improvement over previous approaches, but further research into creating more abstractive models is worth the effort.

6 Conclusion

This article not only presents interesting findings, but presents them in a comprehensive report. From the declaration of the research question and problem, to past experiements, methods, and results, the project has been clearly outlined and reported. Furthermore, a complete analysis of the results and the causes for improvements and discrepancies have been considered fully and fairly.

Acknowledgments

I would like to thank David Inman, the instructor of LING 473 for guiding me through the basics of Computational Linguistics as well as the faculty of the CLMS program at the University of Washington for providing the resources necessary for students like me to develop as computational linguists.

References

See, Abigail and Liu, Peter J. and Manning, Christopher D. 2017. Get To The Point: Summarization with Pointer-Generator Networks. *Association for Computational Linguistics*, July 2017:1073–1083.