A Comparison of Neural and Unsupervised Text Summarization Techniques

Video Link: https://www.youtube.com/watch?v=rv0KJZyJg E
CS230 Spring 2019

Devin Cintron
Department of Computer Science

Abstract

There is a growing phenomenon of long-form documents available on the web. The ability to synthesize substrates of large volumes of texts into a concise, communicable format will enable users and service providers to quickly review large amounts of information in a more manageable form. In this work, we study text summarization techniques that have been advanced which effectively reduce long texts into much shorter formats. The nature of the inputs of these methods, however, is notably diverse.

In this work, we consider the performance of two summarization models developed by different techniques: the unsupervised TextRank algorithm and a neural Pointer-Generator Network. We explain the performance of these methods using the commonly used ROUGE score as well as through a survey of human preferences. Interestingly, we find that while the Pointer-Generator Network performs better as measured by ROUGE score, each model has its advantages.

1 Dataset

We use the CNN/Daily Mail data set in order to compare these methods. The data set consists of articles drawn from the two news services with accompanying summaries for each article. The average length of the articles is approximately 100 words and the average length of the summaries is approximately 60 words. In order to compare against a state-of-the-art model, we compare the results of the Pointer-Generator model taken from the highest-achieving parameters found by the authors.

This leaves us with a subset of 11,490 articles and summaries drawn from the large total CNN/Daily Mail.

2 Methods

2.1 Extractive Baseline First

As a baseline for comparison of the two other models, we first design a naive ‘summarization’ algorithm which simply takes the first 5 sentences of an article, where n is a random choice among integers such that the expected value of the length of the summary is consistent with our other extractive approach.

2.2 Extractive Approach: TextRank

In (Mihalcea and Tarau, 2004) the authors define TextRank, a graph-based extractive summarization approach adapted from Larry Page’s PageRank algorithm. A directed graph is created from a full-text in which summarative text units represent vertices and edges represent relationships between text elements. Then a ranking algorithm is applied and the best scoring elements are kept. This algorithm is similar to PageRank as scores are reflective of the number of incoming connections and the scores of the source-vertices of those incoming connections, the difference being that, in TextRank, edges weights are weighted on a basis of the strength of the relationship. The full process of TextRank is as follows:

1. Identify text units that best define the task at hand, and add them as vertices in the graph.
2. Identify relations that connect such text units, and use those relations to draw edges between vertices in the graph. Edges can be directed or undirected, weighted or non-weighted.
3. Iterate the graph-based ranking algorithm until convergence.
4. Sort vertices based on their final score. Use the relations attached to each vertex for ranking selection decisions.

One implementation uses a variation of this scoring function found in (Battiti, et al. 2016) and takes from the Gensim python library.

2.3 Abstractive Approach: Pointer Generator with Coverage

For our abstractive summarization example, we chose the pointer-generator network approach as advanced by (See et al., 2017), in particular the authors’ pointer-generator with coverage technique. Token-wise encoding is performed by a single-layer bidirectional LSTM and decoding by a unidirectional LSTM. Bidirectional attention is used to compute an attention distribution and context vector. A probability for generating a new term is calculated at each time step via a sigmoid of the weighted product of the context vector, decoded state, and decoded input, as well as a bias. Coverage is an additional component which functions to prevent any particular section of the document from receiving an unbalanced amount of attention—it is the ongoing sum at all previous time steps. The coverage value is then included in future calculations of attention to inform the past distributions. In our training, we were unable to match a performance exceeding that of the authors’ so we opt to use their test results directly when comparing against our extractive and naive approaches.

3 Results

3.1 Automatically-Produced Results

<table>
<thead>
<tr>
<th>Average ROUGE F-Score</th>
<th>Rouge-1</th>
<th>Rouge-2</th>
<th>Rouge-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>CG-Cov</td>
<td>0.4166</td>
<td>0.3039</td>
<td>0.4098</td>
</tr>
<tr>
<td>Gesture</td>
<td>0.3006</td>
<td>0.1461</td>
<td>0.2748</td>
</tr>
<tr>
<td>Nature</td>
<td>0.38916</td>
<td>0.17351</td>
<td>0.35959</td>
</tr>
</tbody>
</table>

For an automated approach to comparison, we used the standard ROUGE score. Specifically we used ROUGE-1, ROUGE-2, and ROUGE-L metrics—these metrics are calculated via overlap of n-grams, overlap of bigrams, and longest common subsequence, respectively. The pointer generator with coverage approach outperforms the TextRank approach by all measures.

Our calculation of ROUGE is performed via National School of Computer Science and Applied Mathematics of Grenoble PhD student Paul Teddy’s FME/Rouge implementation. We believe that this implementation may be somewhat inferior as we note that the scores appear to be inflated relative to other ROUGE implementation measurements—however, since our objective is comparison between the abstractive and extractive techniques, we find this to be permissible for our application.

3.2 Human-Produced Results

In order to provide a human measurement of evaluation, we surveyed 21 individuals on their preferences between the methods. Each individual was asked to read a selection of 3 articles, then to select which summary they preferred between that created by the TextRank approach and the Pointer Generator approach. The order of the summaries was randomized when read to decrease the effect of any sequential bias. Interestingly, the TextRank approach was rated as more preferred in 43 of the 66 assessments.

Conclusions

- The abstractive techniques produce better results by all ROUGE metrics
- The extractive techniques produce better results by human evaluation
- Further research is necessary to reconcile the incommensurability. Importantly, we see a disparity in the nature of summaries depending upon the application.

Future Research

Valuable future work would include an assessment of a wider range of models for summarization. Additionally, it would be valuable to perform a greater investigation into comparisons of human evaluation versus ROUGE scoring across a diverse range of summarization techniques and contexts. While research comparing ROUGE and human scoring has been performed previously, it is important to compare these methods across different summarization contexts and methods as there is a great diversity in the nature of summaries depending upon the application.

Acknowledgements of Work

All work herein was performed alone by Devin Cintron.