[Data Science](#)[Natural Language Processing](#)[NLP Papers Summary](#)

Day 116: NLP Papers Summary – Data-Driven Summarization Of Scientific Articles

By Ryan

25th April 2020

No Comments

Objective and Contribution

Created two multi-sentence summarisation datasets from scientific articles: the title-abstract pairs (title-gen) and abstract-body pairs (abstract-gen) and applied a wide range of extractive and abstractive models to it. The title-gen dataset consists of 5 million biomedical papers whereas the abstract-gen dataset consists of 900K papers. The analysis show that scientific papers are suitable for data-driven summarisation.



WHAT IS DATA-DRIVEN SUMMARISATION?

It is a way of saying the recent SOTA results of summarisation models rely heavily on large volume of training data.

Datasets

The two evaluation datasets are title-gen and abstract-gen. Title-gen was constructed using MEDLINE and abstract-gen was conducted using PubMed. The title-gen pairs the abstract to the title of the paper whereas the abstract-gen dataset pairs the full body (without tables and figures) to the abstract summary. The text processing pipeline is as follows:

1. Tokenisation and lowercase
2. Removal of URLs
3. Numbers are replaced by # token
4. Only include pairs with abstract length 150 – 370 tokens, title length 6 – 25 tokens and body length 700 – 10000 tokens

We also computed the Overlap score and Repeat score for each data pairs. The Overlap score measures the overlapping tokens between the summary (title or abstract) and the input text (abstract or full body). The Repeat score measures the average overlap of each sentence in a text with the remainder of the text. This is to measure the repetitive content that exists in the body text of a paper where the same concepts are repeated over and over again. Below are the summary statistics of both datasets.

Table 1: Statistics (mean and standard deviation) of the two scientific summarization datasets: *title-gen* and *abstract-gen*. Token/sentence counts are computed with NLTK.

<i>title-gen</i>	<i>Abstract</i>	<i>Title</i>
Token count	245 ± 54	15 ± 4
Sentence count	14 ± 4	1
Sent. token count	26 ± 14	-
Overlap	73% ± 18%	
Repeat	44% ± 11%	-
Size (tr/val/test)	5'000'000/6844/6935	

<i>abstract-gen</i>	<i>Body</i>	<i>Abstract</i>
Token count	4600 ± 1987	254 ± 54
Sentence count	172 ± 78	10 ± 3
Sent. token count	26 ± 17	26 ± 14
Overlap	68% ± 10%	
Repeat	74% ± 7%	44% ± 11%
Size (tr/val/test)	893'835/10'916/10'812	

Experimental Setup and Results

MODELS COMPARISON

1. *Extractive summarisation methods.* Two unsupervised baselines here: TFIDF-emb and r[^] d-rank. TFIDF-emb creates sentence representation by computing a weighted sum of its constituent word embeddings. Rwmd-rank ranks sentences by how similar the sentence is compared to all the other sentences in the document. Rwmd stands for Relaxed Word Mover's Distance, which it's the formula used to compute similarity and subsequently LexRank is used to rank the sentences.
2. *Abstractive summarisation methods.* Three baselines here: lstm, fconv, and c2c. Lstm is the common LSTM encoder-decoder model but with an attention mechanism at the word-level. Fconv is a CNN encoder-decoder on subword-level, separating words into smaller units using byte-pair encoding (BPE). Character-level models are good at dealing with rare / out-of-vocabulary (OOV) words. C2c is a character-level encoder-decoder model. It builds character representations from the input using CNN and feed it into an LSTM encoder-decoder model.

RESULTS

The evaluation metrics are ROUGE scores, METEOR score, Overlap score and Repeat score. Despite the weaknesses of ROUGE scores, they are common in summarisation. METEOR score are used for machine translation and Overlap score can measure to what extent the models just copy text directly from input text as summary. Repeat score can measure how often the summary contains repeated phrases, which it's a common problem in abstractive summarisation.



Table 2: Metric results for the *title-gen* dataset. R-1, R-2, R-L represent the ROUGE-1/2/L metrics.


Model	R-1	R-2	R-L	METEOR	Overlap	Token count
<i>oracle</i>	0.386	0.184	0.308	0.146	-	29 ± 14
<i>lead-1</i>	0.218	0.061	0.169	0.077	-	28 ± 14
<i>lexrank</i>	0.26	0.089	0.201	0.089	-	32 ± 14
<i>emb-tfidf</i>	0.252	0.081	0.193	0.082	-	35 ± 17
<i>rwmd-rank</i>	0.311	0.13	0.245	0.116	-	28 ± 13
<i>lstm</i>	0.375	0.173	0.329	0.204	78% ± 20%	12 ± 3
<i>c2c</i>	0.479	0.264	0.418	0.237	93% ± 10%	14 ± 4
<i>fconv</i>	0.463	0.277	0.412	0.27	95% ± 9%	15 ± 7

Table 3: Metric results for the *abstract-gen* dataset. R-1, R-2, R-L represent the ROUGE-1/2/L metrics.

Model	R-1	R-2	R-L	METEOR	Overlap	Repeat	Token count
<i>oracle</i>	0.558	0.266	0.316	0.214	-	42% ± 10%	327 ± 99
<i>lead-10</i>	0.385	0.111	0.18	0.138	-	20% ± 4%	312 ± 88
<i>lexrank</i>	0.45	0.163	0.213	0.157	-	52% ± 10%	404 ± 131
<i>emb-tfidf</i>	0.445	0.159	0.216	0.159	-	52% ± 10%	369 ± 117
<i>rwmd-rank</i>	0.454	0.159	0.216	0.167	-	50% ± 10%	344 ± 93
<i>fconv</i>	0.354	0.131	0.209	0.212	98% ± 2%	52% ± 28%	194 ± 15

For title-gen results (table 2), rwmd-rank is the best extractive model, however, c2c (abstractive model) outperformed all extractive models by a large margin, including the oracle. Both c2c and fconv achieved similar results with similar high overlap scores. For abstract-gen results (table 3), lead-10 was a strong baseline and only extractive models managed to outperform it. All extractive models achieved similar ROUGE scores with similar Repeat score. Abstractive models performed poorly based on ROUGE scores but outperformed all models in terms of METEOR score so it was difficult to draw up conclusion.

Qualitative evaluation is common and conducted on the generated summary. See below an example of the title-gen qualitative evaluation. The observations are as follow:

1. Large variation of sentence locations selected by extractive models on title-gen, with first sentence in the abstract being the most important
2. Many abstractive generated titles tend to be of high quality, demonstrating their ability to select important information
3. Lstm tends to generate more novel words whereas c2c and fconv tend to copy more from input text
4. The generated titles occasionally make mistakes by using incorrect words,  too generic and fail to capture the main point of the paper. This could all lead to factual

inconsistencies 

5. For abstract-gen, it appears that introduction and conclusion sections are most relevant for generating abstract. However, important content are spread across sections and sometimes the reader focuses more about the methodology and results
6. Output of fconv abstractive model is of bad quality where it lacks coherent and content flow. There is also the common problem of repeated sentence or phrases in the summary



Conclusion and Future Work

There was a mixed results where the models performed well in title generation but struggled with abstract generation. This can be explained by the high-level of difficulty in understanding long input and output sequences. A future work is a hybrid extractive-abstractive end-to-end approaches.

Source: <https://arxiv.org/pdf/1804.08875.pdf>

Ryan

Data Scientist

Previous Post

< Day 115: NLP
Papers Summary -
SCIBERT: A
Pretrained
Language Model
for Scientific Text

Next Post

Day 117: NLP
Papers Summary -
Abstract Text
Summarization: A
Low Resource
Challenge





[Data Science](#) [Natural Language Processing](#) [NLP Papers Summary](#)

Day 365: NLP Papers Summary – A Survey on Knowledge Graph Embedding

Ryan

30th December 2020





[Data Science](#) [Implementation](#) [Natural Language Processing](#)

Day 364: Ryan's PhD Journey – OpenKE-PyTorch Library Analysis + code snippets for 11 KE models

Ryan

29th December 2020

