

Attention based Sentence Extraction from Scientific Articles using Pseudo-Labeled data

XXX

YYY

abc@zzz.com

XXX

YYY

abc@zzz.com

XXX

YYY

abc@zzz.com

ABSTRACT

In this work, we present a weakly supervised sentence extraction technique for identifying important sentences in scientific papers that are worthy of inclusion in the abstract. We propose a method for determining the focus of a given paper using topic models and use it to create document-level context. We also propose an attention based sentence encoding model that jointly learns to identify important content as well as the cue phrases that are indicative of summary worthy sentences. We use a collection of articles publicly available through ACL anthology for our experiments. Our system achieves a performance that is better, in terms of several ROUGE metrics, as compared to several state of art extractive techniques. It also generate more coherent summaries and preserves the overall structure of the document.

KEYWORDS

Summarization, Attention, LSTM, Topic model

ACM Reference Format:

XXX, XXX, and XXX. 2018. Attention based Sentence Extraction from Scientific Articles using Pseudo-Labeled data. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 INTRODUCTION

Automatic text summarization has been a major focus area for researchers since a few decades now. However, due to small amount of training data and even fewer number of gold standard summaries, experiments in automatic summarization have largely been dependent on the news wire corpora created under the *document understanding conference (DUC)* and *text analysis conference (TAC)*. With the increasing use of deep learning techniques for various NLP tasks, large volumes of training data are more important than ever before, but few of them are publicly available[15],[4]. In this work we draw attention to one such publicly available corpus of scientific articles published in ACL anthology. This is perhaps one of the largest corpus of documents with corresponding manually written abstract. The only other corpora of similar size come from news wire data. But the summaries in that case are largely limited to highlights or article headings as opposed to an actual abstractive

summary. All the previous experiments using scientific articles were limited to not more than a few hundred articles[1].

In this work we propose a novel sentence extraction technique as a first step towards automatically generating the abstracts of scientific papers. We demonstrate that a very efficient sentence extractor can be created using this data, with weakly supervised training and without any manual labelling. The main contributions are two fold, firstly we propose a simple context encoder which can capture the overall theme of the document and generate a context embedding. Second, we propose an attention model that uses sequence encoder based sentence embeddings along with this context embedding to assign importance to different words in a given sentence. This module jointly learns to capture the informative content as well as the cue phrase information that make a sentence summary worthy. As we show in the results, our model is able to identify and leverage cue phrases like " and " to decide the *summary worthiness* of the sentence. Using this information, we are able to maintain the overall structure of document when creating the abstract, which is not possible using the existing extractive techniques. Contrary to most of the existing techniques, our approach is not dependent on manually tagged data or any linguistic resources.

The most notable attempt was by [17]. They solve this problem by leveraging rhetorical status of sentences to generate the abstract. The idea is to select sentences in such a manner that the abstract highlights new contribution of the paper and also relates it to the existing work. The authors identify *Rhetorical zones* to which a sentence can belong like, aim of the experiment, statements that describe structure of the article, comparison with other works, etc. Such features are then used to decide the importance of the sentence as well as to maintain the overall structure of the final extract. The work by [12] focuses on generating impact based summaries for scientific articles. Sentences are ranked based on the impact they have produced on other works in the same or related domains. *Document sentences* that best match the content of the *citing sentence* are identified using language models. They used a dataset of around 1300 papers published by ACM SIGIR. The work described in [1] clusters articles based on their citation graph and then use lexrank to rank sentences within each cluster. The work proposed in [6] focuses on identifying the sections of original paper to which a given abstract sentence is related. Our model implicitly tries to learn similar information. In the results we show that the attention model learns to identify phrases which are indicative of the section information and such sentences are usually selected in the summary. Another related work to the proposed approach is by [2]. It focuses on generating headlines of news articles using sentence and document encoders. The authors use sentence and word level labelling to identify important phrases in the document and then generate an extract based on that technique.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Conference'17, July 2017, Washington, DC, USA

© 2018 Copyright held by the owner/author(s).

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.

<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

2 PROPOSED ARCHITECTURE

Our proposed model consists of four main blocks: A lstm based sentence encoder, topic modelling based context encoder, attention module and a binary classifier. Overall the aim is to determine the probability $p(y|s, d)$ where $p(y)$ is the probability that sentence s in a document d is *summary worthy*. We represent s as a embedding vector of fixed dimensions, using a sentence encoder. Next we represent each document by the topic extracted using lda, and use those topics to create a context embedding. The attention model then uses the sentence and context embeddings to learn to assign weights to different parts of the sentence. Finally the classifier uses output of attention module and the original context embeddings to decide whether or not a sentence is *summary worthy*. Below we describe the individual blocks.

2.1 Sentence Encoder

Each sentence S is represented as a sequence of N vectors $[x_1, \dots, x_N]$ where x_i is the i^{th} word represented by its word embedding vector. The initial word embeddings were created by training a word2vec[14] model on the entire ACL corpus, and were updated during the training. The word embedding matrix E is of size $V \times D$, where V is the vocabulary size and D is the word embedding size. Next we use a LSTM based sequence encoder with a hidden size of U for creating the sentence embeddings using these word embeddings. LSTM based sentence encoders are now considered a standard technique for creating sentence embeddings. We limit the maximum possible length of a sequence to L .

2.2 Context Encoder

Even for humans, knowledge about the overall scope of an article is pivotal when selecting important information that has to be included in the abstract. There have been attempts to generate a document encoding, by using an additional LSTM based sequence encoder that takes input a sequence of sentence embeddings created by the sentence encoder defined above[2] and gives a single vector or the *document embedding*. However, such an approach requires large amount of training data, of the order hundreds of thousands of article, and takes much longer to train. As an alternative we propose a simpler approach, that efficiently captures the overall scope of the document and can be efficiently trained using a few thousand documents. It is noteworthy that here our aim is not to capture the document structure explicitly, but to capture the overall theme of the document.

Our context encoder follows a two step approach. In the first step we encode each article in terms of representative concepts present in them. We extracted 500 abstract topics from the overall corpus using *Latent Dirichlet Allocation* based topic modelling. Topic vectors for each document can be represented as a matrix $T \in \mathbb{R}^{M \times M}$, $T = [t_1, \dots, t_M]$, where t_i is the one-hot encoded vector of size $1 \times M$ for topic i , and M is the pre-decided number of topics. We separately initialized a topic embedding matrix $F \in \mathbb{R}^{M \times C}$, where M is the total number of topics and C is the context embedding size. We randomly initialize F and it is jointly updated with the overall model. $J \in \mathbb{R}^{C \times M}$ represents the topic embeddings. We then perform a weighted average of the topic embeddings using their probabilities(p_i). This additional step helps in reducing the

sparsity of LDA representation as well as to leverage latent similarity between different topics, and at the same time assigning an importance score to each of the topics. $c \in \mathbb{R}^{C \times 1}$ represents the final weighted context embeddings.

$$J = F^T T \quad (1)$$

$$c = \sum_i p_i J_i \quad (2)$$

2.3 Attention module

This module plays a key role in the overall architecture. In past few years, attention networks have become a popular choice for several NLP tasks. Several approaches have been proposed for using attention for document summarization[2],[15]. We propose a simple attention architecture that takes into account the document context and sentence embedding for generating attention weights over the sentence words. We argue that besides identifying informative content in the document, such an attention model would help in automatically identifying words or phrases, which can act as a cue for deciding whether or not that sentence is summary worthy. The attention weights($[w_1, \dots, w_L]$) are computed as shown in equation 3, where $Z \in \mathbb{R}^{(S+C) \times L}$ and $w \in \mathbb{R}^{L \times 1}$. The attention module learns weights w as a function of the sentence embedding(local context) as well as the context embedding (global context). L is the maximum allowed length of any input sentence. Sentences shorter than this are padded to make them of the same length. $Y \in \mathbb{R}^{L \times S}$ denotes the intermediate steps of LSTM output at each of the L time stamps. $Y = [y_1, \dots, y_L]$ where y_i represents intermediate output at a particular time stamp i .

$$w = Z(s, c) \quad (3)$$

$$a = w^T Y \quad (4)$$

2.4 Classifier

The classifier consists of two layered feed forward network. We used a hidden layer with weights $H \in \mathbb{R}^{(A+C) \times Q}$ followed by a output layer $O \in \mathbb{R}^{Q \times 1}$ and a sigmoid activation function (σ).

$$h = H[a, c] \quad (5)$$

$$o = \sigma(Oh) \quad (6)$$

The entire architecture is shown in the Figure 1 below.

3 EXPERIMENTAL SETUP

We use a subset of the ACL Anthology corpus which is a collection of scientific articles broadly from computational linguistics and related domains. These articles are openly available in the acl anthology website¹ in pdf formats. We used the publicly available Science Parse library² for extracting section wise information from the pdf documents. Only the articles published in or after the year 2000 were included. Further, the articles that were not parsed properly were discarded. Finally, 27,801 articles were used in this experiment, which were divided into train (23000), validation(2000)

¹<http://aclweb.org/anthology/>

²<https://github.com/allenai/science-parse>

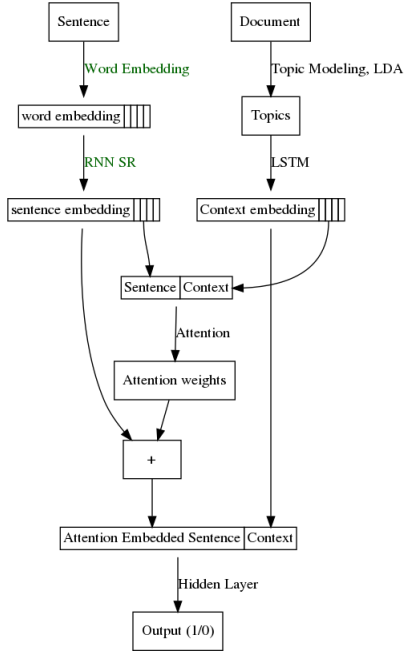


Figure 1: Attention based sentence selector

and test sets(2801). The ids of these articles can be found here³. We did not perform any preprocessing or manual labelling.

For each sentence in the document we assign a pseudo-label of 1(*important*) or 0(*not important*), based on their cosine similarity with the sentences in abstract. For each sentence in the abstract we select the best matching sentence from the document if the cosine similarity is above 0.75 and assign it a label 1. All other sentences are assigned a label 0. Compared to a summary of a newswire cluster, abstracts of scientific articles are much more precise with a higher compression ratio. The average size of ACL articles is 200 sentences or about 3600 words, while the average summary size is 125 words. This results in a heavily skewed training data, with more than 95% sentences being labelled as *not important*. To mitigate this bias, we filter out sentences with tf-idf scores lower than 0.05. Further we randomly sample the *not important* sentences to bring down the positive-negative ratio to 1:4. We then use a weighted loss function, explained below, to assign a higher loss to false negatives as compared to false positives. Next we train the attention based model described in previous section with the sentences as inputs and the pseudo labels as target. The implementation details and choice of parameters are described below.

3.1 Implementation Details

We used the pytorch library⁴ for our experiments. For training we use Adam optimizer to minimize the weighted binary cross-entropy loss. We use weighted binary cross entropy to partially mitigate the class imbalance issue mentioned previously. We use a weight of 0.2 for negative samples and 0.8 for positive samples. For adam

³urlunavailabletomaintainanonymity

⁴<http://pytorch.org/>

optimizer we use the most common setting with a learning rate of 0.001, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. Training was performed with shuffled minibatches of size 500 and a dropout of 0.2 was used for all layers. All the random initializations used xavier normal distribution. We used $D = 100$ (word embedding size), $C = 10$ (context embedding size) and $M = 500$ (number of topics). We used a single LSTM layer with 200 hidden states and $Q = 100$ (classifier hidden layer size). We plan to make the source code publicly available.

4 RESULTS

We evaluate our model on a held out set of 2801 documents. ROUGE metrics[9] are used to compare the system generated extracts with the original abstracts of the papers. As explained previously summarizing scientific documents is a precision oriented task, and hence We report ROUGE-N precision (N=1,2,3,4). We compare our results with five widely accepted extractive summarization systems besides two state of art techniques. We specifically choose the topic signature[10] and latent semantic analysis[16] based approaches due to their ability to identify overall context and latent topics in a document. Besides these, we also compare our results with the popular graph based approaches, lexrank[3] and textrank[13] and a simple frequency based approach. We also compare the results with Submodular optimization based technique[11] and Integer linear programming based summarization[5], which are considered to be state of art techniques for sentence extraction[7].

In order to make the results reproducible, we follow the guidelines suggested in [7] and use a fixed set of parameters when computing ROUGE scores⁵. Since the abstract size varies across documents in the evaluation set, we use the average abstract length of 125 words, when computing the ROUGE scores. All other parameters are same as those mentioned in [7]. The results are shown in table 1 below.

Table 1: Results (ROUGE-N Precision)

| Summarizer | R-1 | R-2 | R-3 | R-4 |
|------------|--------------|--------------|--------------------------|--------------------------|
| Topicsum | 0.266 | 0.055 | 0.020 | 0.012 |
| LSA | 0.302 | 0.065 | 0.027 | 0.018 |
| LexRank | 0.354 | 0.087 | 0.037 | 0.020 |
| TextRank | 0.305 | 0.074 | 0.030 | 0.018 |
| FreqSum | 0.331 | 0.088 | 0.034 | 0.018 |
| Submodular | 0.360 | 0.087 | 0.036 | 0.022 |
| ILP | | | | |
| Neural | 0.344 | 0.090 | 0.042[†] | 0.027[†] |

Figures in bold indicate the best performing system

[†] indicates significant difference with $\alpha = 0.05$

As evident from table 1, the proposed approaches outperforms all the existing systems on most ROUGE metrics. The only exception is Rouge-1 measure, where Submodular performs the best. We observe that R-3 and R-4 better reflect the systems ability to retain structural information in the abstract. A summary with good R-1 has more informative words, but misses out on the structural information. A summary with higher R-3 or R-4 usually prefer sentences with

⁵ROUGE-1.5.5 with the parameters: -n 4 -m -a -l 125 -x -c 95 -r 1000 -f A -p 0.5 -t 0

clear cue phrases like 'results are significantly higher compared to' or 'in this paper we propose'. This is closer to the way a human would decide whether or not to include the information. A good R-1 does not necessitate that.

Below, we include a summary generated by our system along with the original abstract of the paper. The intensity of highlight shows the attention weights assigned by our model to a particular word. Darker the shade, higher the attention. In general, we observe that the proposed attention model efficiently identifies content words and cue phrase, both of which are important when selecting a sentence. For example, consider the first sentence of system generated summary: "In this paper we propose a statistical model for measure word generation for English-to-Chinese SMT systems, in which contextual knowledge from both source and target sentences is involved.". Our model identifies the phrases "In this paper we propose" (cue phrase) and "statistical model for measure word" (part of title) as important.

Document ID: P08-1011

Original Abstract: Measure words in Chinese are used to indicate the count of nouns. Conventional statistical machine translation (SMT) systems do not perform well on measure word generation due to data sparseness and the potential long distance dependency between measure words and their corresponding head words. In this paper, we propose a statistical model to generate appropriate measure words of nouns for an English-to-Chinese SMT system. We model the probability of measure word generation by utilizing lexical and syntactic knowledge from both source and target sentences. Our model works as a post-processing procedure over output of statistical machine translation systems, and can work with any SMT system. Experimental results show our method can achieve high precision and recall in measure word generation.

System generated summary:

In this paper we propose a statistical model for measure word generation for English-to-Chinese SMT systems, in which contextual knowledge from both source and target sentences is involved. To overcome the disadvantage of measure word generation in a general SMT system, this paper proposes a dedicated statistical model to generate measure words for English-to-Chinese translation. Experimental results show our method can significantly improve the quality of measure word generation. We also compared our method with a wellknown rule-based machine translation system - SYSTRAN3. Most existing rule-based English-to-Chinese MT systems have a dedicated module handling measure word generation.

Sample abstract and system generated summary

It is also interesting to note that the proposed model efficiently captures overall structure of the document, it starts with proposed work, then some details about experiment and system comparison. Barring the last sentence it is quite precise and coherent in terms of content. Although it is not always possible to have sentences in the original documents that can directly be included in the abstract,

but the results of current experiment are quite encouraging and can serve as a very good first step towards abstract generation.

5 CONCLUSIONS

In this work we proposed a weakly-supervised approach for generating extracts from scientific articles. We use topic models to create a context embedding that defines scope of the article and then use an attention based sequence encoder to generate sentence encoding. We then use pseudo labelled data to train a classifier that predicts whether or not a given sentence is *summary worthy*. When evaluating on ACL anthology corpus, we were able to outperform the existing baseline and state of art techniques on ROUGE-2,3 and 4 metrics, while achieving a comparable performance on ROUGE-1. Moreover we also demonstrate that our approach well preserves the overall structure of original document resulting in a final summary that is quite coherent. We envision this as a first step towards automatically creating abstracts of scientific articles and the results can be further used by generative techniques.

REFERENCES

- [1] Amjad Abu-Jbara and Dragomir Radev. 2011. Coherent citation-based summarization of scientific papers. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 500–509.
- [2] Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. *arXiv preprint arXiv:1603.07252* (2016).
- [3] Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research* 22 (2004), 457–479.
- [4] Katja Filippova, Enrique Alfonseca, Carlos A Colmenares, Lukasz Kaiser, and Oriol Vinyals. 2015. Sentence compression by deletion with lstms. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. 360–368.
- [5] Dan Gillick and Benoit Favre. 2009. A scalable global model for summarization. In *Proceedings of the Workshop on Integer Linear Programming for Natural Language Processing*. Association for Computational Linguistics, 10–18.
- [6] Kenji Hirohata, Naoaki Okazaki, Sophia Ananiadou, and Mitsuru Ishizuka. 2008. Identifying sections in scientific abstracts using conditional random fields. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I*.
- [7] Kai Hong, John M Conroy, Benoit Favre, Alex Kulesza, Hui Lin, and Ani Nenkova. [n. d.]. A Repository of State of the Art and Competitive Baseline Summaries for Generic News Summarization.
- [8] Julian Kupiec, Jan Pedersen, and Francine Chen. 1995. A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 68–73.
- [9] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out* (2004).
- [10] Chin-Yew Lin and Eduard Hovy. 2000. The automated acquisition of topic signatures for text summarization. In *Proceedings of the 18th conference on Computational linguistics-Volume 1*. Association for Computational Linguistics, 495–501.
- [11] Hui Lin and Jeff Bilmes. 2012. Learning mixtures of submodular shells with application to document summarization. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 479–490.
- [12] Qiaozhu Mei and ChengXiang Zhai. 2008. Generating impact-based summaries for scientific literature. *Proceedings of ACL-08: HLT* (2008), 816–824.
- [13] Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing Order into Texts. In *Proceedings of Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Barcelona, Spain, 404–411.
- [14] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).
- [15] Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685* (2015).
- [16] Josef Steinberger. 2004. Using latent semantic analysis in text summarization and summary evaluation. In *Proceedings of ISIMA'04*. 93–100.
- [17] Simone Teufel and Marc Moens. 2002. Summarizing scientific articles: experiments with relevance and rhetorical status. *Computational linguistics* 28, 4 (2002), 409–445.