ATHENA@CL-SciSumm 2019: Siamese recurrent bi-directional neural network for identifying cited text spans

Aris Fergadis[1,3], Dimitris Pappas[2,3], and Haris Papageorgiou[3]

[1] School of Electrical and Computer Engineering, National Technical University of Athens, Greece
[2] Department of Informatics, Athens University of Economics and Business, Greece
[3] Athena Research and Innovation Center, Greece

Abstract. In this paper we describe our participation to the Task1 of the CL-SciSumm 2019. The task is on automatic paper summarization in the research area of Computational Linguistics. Our approach is a two step binary sentence pair classification between the so-called citances and *candidate sentences*. Firstly, we classify sentences in the abstracts to predefined classes we call "zones". These zones capture the discourse structure of a scientific publication. We then expand these zones with additional, similar sentences which are found in the main sections of the publication body. We train a Siamese bi-directional GRU neural network with a logistic regression layer to decide if a citance alludes to a candidate sentence. The cited sentences are also assigned one or more discourse facets (i.e., categories defined in the Task) using a multi-class SVM. We ran extensive experiments in three different datasets achieving promising results.

Keywords: Candidate Cited Sentence Selection · Siamese Neural Network · Discourse Facet

1 Introduction

Researchers are confronted with a continuously increasing volume of scientific publications, facing difficulties to monitor and track [13]. The ability to create synopsis of the key-points, contribution and importance of a paper within an academic community is an important step [12]. This synopsis, can be created by using citation sentences (i.e., the citances) that reference a specific paper and can be considered as a community-created summary of a topic or a paper. Scientific summaries offer an overview of the cited paper useful to scholars, writers or literature reviewers [7, 10]. The CL-SciSumm Shared Task focuses on the scientific summarization of papers [6], organized into two tasks. For both tasks the organizers provide several Reference Papers (RPs) called "topics".

Task1A: For each citance, identify the spans of text (cited text spans) in the RP that most accurately reflect the citance. These spans are of the granularity of

a sentence fragment, a full sentence, or several consecutive sentences (no more than 5).

Task1B: For each cited text span, identify what facet of the paper it belongs to, from a predefined set of facets.

Task2 (optional): Generate a structured summary of the RP from the cited texts pans of the RP. The length of the summary should not exceed 250 words.

We participated on the tasks 1A and 1B of 2019 shared task [1] and present here our methodology.

2 Methodology

We approach Task 1A as a binary sentence pair classification problem. We create pairs of citance and a candidate sentence extracted from the Citing Papers (CP) and the RPs respectively. Word embeddings are used to select candidate sentences. A Siamese neural network process these pairs to decide whether or not the candidate sentence is a cited text span of the citance. For the Task 1B a multi-class SVM [3] model assigns a discourse facet to the cited text spans.

2.1 System Components

Word Embeddings We use word embeddings for both the candidate sentences selection and for the embedding layer of our network. Embedding vectors are trained on the ACL Corpus dump[4] using the CBOW implementation of word2vec [11] of the gensim[5] tool, with negative sampling set to 5 and 100 for dimensionality of the word vectors. All words are converted to lowercase.

Candidate Sentence Selection We select sentences from the RP as candidate sentences. The intuition is that not all sentences are equally important as cited text spans. Thus, we try to select sentences that are about *methodology*, *results* and *conclusions* and discard sentences about *background* and *related work*. This is also supported by the fact that the cited text is assigned a discourse facet. Our approach tries to eliminate sentences that potentially would be false positives.

To select the candidate sentences of the RP we split the abstract into *zones* [9]. Each sentence is classified to one of the following zones: *Background*, *Method*, *Result* and *Conclusion*. We keep only the sentences that belong to *Method*, *Result* and *Conclusion* zones (referred to as *zone sentences*). Sentences are split into words[6], punctuation and numbers are removed and each word is assigned its embedding vector. For each zone sentence and the rest of the RP sentences we calculate sentence embeddings by averaging the word embedding vectors.

[4] http://acl-arc.comp.nus.edu.sg/archives/acl-arc-160301-parscit/
[5] https://radimrehurek.com/gensim/, version 3.7.3
[6] Using the tokenization tools of the gensim module

Using the embedding vectors of the N zone sentences and all the other embedding vectors of the M RP sentences, we calculate a similarity matrix $S \in \mathbb{R}^{N \times M}$ using cosine similarity measure. To get the most similar sentences to the zone sentences we define a threshold t_s. The RP sentences $S_{i,j}$ that pass the similarity threshold t_s and the zone sentences are kept as candidate sentences. The decision of the t_s value is discussed into section 3.

Siamese Neural Network The Siamese neural network is composed of two bi-directional GRUs (biGRU) [14, 2] and a logistic regression layer, as depicted in Figure 1. Each biGRU processes one sentence at a time. For each citance and a set of candidate sentences of the RP, the left biGRU takes as input the citance and the right biGRU takes as input one of the candidate sentences. We use $w_{1:n}$ to denote a sequence of words $w_{1:n} = w_1, \dots w_n$, each with their corresponding d_{emb} dimensional word embedding $\boldsymbol{e_i} = \boldsymbol{E}_{[w_i]}$. The embedding matrix $\boldsymbol{E} \in \mathbb{R}^{|V| \times d_{emb}}$ associates words from the vocabulary V with d_{emb} dimensional dense vectors.

The left biGRU applies additive zero-centered Gaussian noise [4] to word embeddings with $\sigma = 0.05$ as a regularization layer at the training phase. The outputs $\boldsymbol{y_1^b}$ and $\boldsymbol{y_n^f}$ of the backward GRU^b and the forward GRU^f respectively are concatenated in one vector

$$
\begin{aligned}
\boldsymbol{y_1^b} &= \text{GRU}^b(\boldsymbol{e_{n:1}}) \\
\boldsymbol{y_n^f} &= \text{GRU}^f(\boldsymbol{e_{1:n}}) \\
\boldsymbol{x^l} &= [\boldsymbol{y_1^b}; \boldsymbol{y_n^f}] \\
\boldsymbol{y_1'^b} &= \text{GRU}^b(\boldsymbol{e_{n:1}}) \\
\boldsymbol{y_n'^f} &= \text{GRU}^f(\boldsymbol{e_{1:n}}) \\
\boldsymbol{x^r} &= [\boldsymbol{y_1'^b}; \boldsymbol{y_n'^f}]
\end{aligned}
$$

We use $\boldsymbol{x^l}$ to denote the output of the left input and $\boldsymbol{x^r}$ of the right input and $[\cdot;\cdot]$ to denote concatenation. The two output vectors are element wise multiplied to give a vector $\boldsymbol{x}$. A logistic regression layer (LR) with a sigmoid activation function $\sigma(\cdot)$ is used to make the final prediction $\hat{y}$. To summarize the architecture

$$
\begin{aligned}
p(y = k|w_{1:n}) = \hat{y}, k \in \{0, 1\} \\
\hat{y} = \text{LR}(\boldsymbol{x}), \text{with } \sigma(\cdot) \text{ activation} \\
\boldsymbol{x} = [\boldsymbol{x^l} \times \boldsymbol{x^r}]
\end{aligned}
$$

The described model considers one sentence at a time. In order to find if a citance references more than one sentences in the RP, we take the predictions of all the candidate sentences and keep the maximum score s_{max}. We define a threshold as $s_t = 0.98 \cdot s_{max}$. Any candidate sentence that has score s such as $s_t \leq s \leq s_{max}$ is selected as a cited sentence.

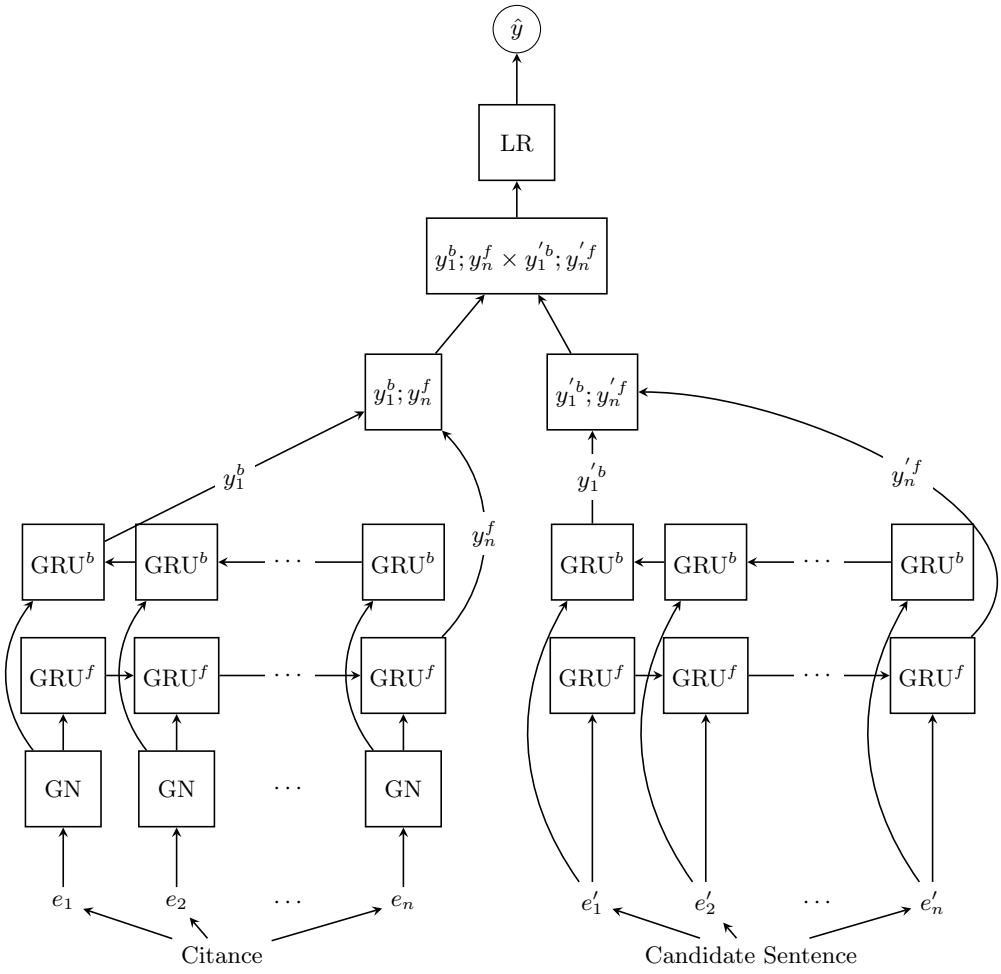

Fig. 1. Siamese bi-directional GRU network. The left input is the *Citance* and the right input a *Candidate Sentence*. The output of the biGRU networks are concatenated, element wise multiplied and a Logistic Regression (LR) layer with sigmoid activation gives the a prediction if the Citance cites the Candidate Sentence. *GN* denotes Gaussian Noise Layer.

Discourse Facet Task 1B asks "for each cited text span, identify what facet of the paper it belongs to, from a predefined set of facets". The predefined facets are *Aim*, *Hypothesis*, *Method*, *Result* and *Implication*. We approach this task as a multi-class classification problem due to the fact that some cited text spans may have up to two facets. We build a bag-of-terms representation of all n-grams with $n = 1, 2, 3$ and calculate their tf-idf values using L1-norm. Five one-vs-rest SVM classifiers were trained assigning a cited text span to each of the five facets.

3 Experiments and Results

The dataset provided was split into a training and a development set. The training set consists of a set of 40 RPs with their CPs annotated by humans and 1000 RPs and their corresponding CPs that were automatically annotated. For our experiments we only used the human part of the training set (*the TR-H set*).

As a first step for our experiments we selected candidate sentences from the RPs. By keeping only the candidate sentences we might miss cited sentences in the RP which were not selected by our method. In Table 1, *coverage* is the number of RP sentences we kept and the *hits* metric is the number of the cited sentences in our candidate list (in percentage). Our target is to get minimum coverage with maximum hits. Minimum coverage means that we have kept all the good candidates while maximum hits denotes that the cited sentences are within our candidate list.

Table 1 displays the average coverage and hits for the 40 RPs of the training set and the 10 RPs of the development set for different thresholds. Based on the results, t_s was set to 0.5. Using this threshold, we keep about 70% of the RP sentences on the training set and 60% on the development set, on average. Despite the fact that we discarded about 30% and 40% of the candidate sentences we only lose 15% and 20% of the cited text spans, respectively.

Table 1. Average of the *coverage* and the *hits* of the selected candidate sentences for the 2018 training and development set using two thresholds.

		Coverage	**Hits**
$t_s = 0.5$	Training Set Average	69.87%	84.16%
	Development Set Average	61.58%	81.15%
$t_s = 0.7$	Training Set Average	21.89%	38.50%
	Development Set Average	15.14%	33.02%

We evaluated our system in three different versions of the dataset; for each version, we used for testing the development set (Dev), the 2016 test set (2016) and the 2017 test set (2017) respectively. For training, we used the *TR-H set* provided that we have excluded all papers in the relevant testing set for obvious reasons. The results shown in Table 2 are comparable to those of the previous shared tasks [6, 5, 8].

4 Conclusions and Future Work

Scientific summarization is a challenging task as it is evident from the results of the previous shared tasks [6, 5, 8]. In our methodology we create pairs of citance and a candidate sentence extracted from the CP and the RP respectively. These pairs are classified from a Siamese neural network as positive if a citance indeed cites a sentences, otherwise as negative. The cited sentences are also assigned one

Table 2. Results on the three test sets reporting Micro and Macro average scores for Tasks 1A and 1B.

		Task 1A			Task 1B		
Test Set	**Average**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
Dev	Micro	0.137	0.090	0.108	0.950	0.114	0.203
	Macro	0.125	0.087	0.102	0.750	0.104	0.183
2016	Micro	0.077	0.055	0.064	0.941	0.076	0.140
	Macro	0.103	0.102	0.102	0.100	0.100	0.100
2017	Micro	0.136	0.094	0.112	1.000	0.135	0.238
	Macro	0.182	0.156	0.168	0.600	0.182	0.279

or more discourse facets. We applied our methods on the dataset of the 2019 CL-SciSumm shared task. The evaluation of our system indicates that the Siamese neural network performs comparable to other machine learning methods.

In future work we will investigate the impact of replacing the logistic regression layer with other similarity functions, such as cosine similarity. We also plan to select the best value for the s_t threshold via hyper-parameter tuning. Finally, we will experiment with different methods for cited sentences selection which take into account the scores of neighboring sentences.

5 Acknowledgement

We acknowledge support of this work by the Data4Impact Project which received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 770531.

References

1. Chandrasekaran, M.K., Yasunaga, M., Radev, D., Kan, M.Y.: Overview and results: Cl-scisumm shared task 2019. In: Proceedings of the 4th Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL 2019) @ SIGIR 2019, Paris, France. (2019)
2. Cho, K., Van Merriënboer, B., Bahdanau, D., Bengio, Y.: On the properties of neural machine translation: Encoder-decoder approaches. arXiv preprint arXiv:1409.1259 (2014)
3. Cortes, C., Vapnik, V.: Support-vector networks. Machine learning **20**(3), 273–297 (1995)
4. Hinton, G., Van Camp, D.: Keeping neural networks simple by minimizing the description length of the weights. In: in Proc. of the 6th Ann. ACM Conf. on Computational Learning Theory. Citeseer (1993)
5. Jaidka, K., Chandrasekaran, M.K., Jain, D., Kan, M.Y.: The cl-scisumm shared task 2017: Results and key insights. In: BIRNDL@ SIGIR (2). pp. 1–15 (2017)
6. Jaidka, K., Chandrasekaran, M.K., Rustagi, S., Kan, M.Y.: Overview of the cl-scisumm 2016 shared task. In: Proceedings of the Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL). pp. 93–102 (2016)

7. Jaidka, K., Khoo, C., Na, J.C.: Deconstructing human literature reviews–a framework for multi-document summarization. In: proceedings of the 14th European workshop on natural language generation. pp. 125–135 (2013)
8. Jaidka, K., Yasunaga, M., Chandrasekaran, M.K., Radev, D.R., Kan, M.Y.: The cl-scisumm shared task 2018: Results and key insights. In: BIRNDL@SIGIR (2018)
9. Jin, D., Szolovits, P.: Hierarchical neural networks for sequential sentence classification in medical scientific abstracts. ArXiv **abs/1808.06161** (2018)
10. Jones, K.S.: Automatic summarising: The state of the art. Inf. Process. Manage. **43**, 1449–1481 (2007)
11. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. pp. 3111–3119 (2013)
12. Nakov, P.I., Schwartz, A.S., Hearst, M.: Citances: Citation sentences for semantic analysis of bioscience text. In: Proceedings of the SIGIR. vol. 4, pp. 81–88 (2004)
13. Qazvinian, V., Radev, D.R.: Scientific paper summarization using citation summary networks. In: Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1. pp. 689–696. Association for Computational Linguistics (2008)
14. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing **45**(11), 2673–2681 (1997)