WING-NUS at CL-SciSumm 2017: Learning from Syntactic and Semantic Similarity for Citation Contextualization

Animesh Prasad

School of Computing, National University of Singapore, Singapore
a0123877@u.nus.edu
http://wing.comp.nus.edu.sg/

Abstract. We present here system report for our model submitted for shared task on Computational Linguistic Scientific-document Summarization (CL-SciSumm) 2017. We hypothesize that search and retrieval based techniques are sub-optimal for learning complex relation likes provenance. State-of-the-art information retrieval techniques using term frequency - inverted document frequency (TF-IDF) to capture surface level closeness along with different textual similarity features for semantic closeness are insufficient to capture implied and entailed provenance with less surface-level similarity. In our comparative studies, we find that the provenance is relative i.e. something makes a better provenance than other based on certain linguistic cue or key information being more prominently conveyed, and hence we model the problem as pairwise-ranking and not simple ranking or classification. To capture above points we propose a joint scoring approach weighting surface level closeness and learned semantic relation. We use TF-IDF and Longest Common Subsequence (LCS) for the syntactic score and pairwise neural network ranking model to calculate semantic relatedness score. For citation-provenance facet classification, we retrofit the same neural network architecture on identified provenance, with the removed pairwise ranking component.

Keywords: deep learning, ranking, provenance, facet, citation contextualization

1 Introduction

With the overwhelming amount of scientific studies published every minute, it has become very difficult to keep track of the recent advancements. Keeping this vision in mind the BiomedSumm followed by CL-SciSumm task was proposed in 2014 [1]. The focus of these tasks is to create a summary of a scientific document (reference paper) by taking into account the documents citing it (citing papers) as well. This aims at gathering a comprehensive summary around the document which includes the limitations, strengths and overall view of the scientific community towards the piece of work.

To locate such references to the documents it's important to identify the cross-document discourse. One such structure used to capture such discourse is

citation context i.e. the piece of the text in the reference paper (RP) cited by the citing paper (CP). The text in the CP is called the citance and the corresponding text in RP is called the provenance. Provenance helps in citation understanding and together with the citance can be used to understand important aspects like function, emotion, polarity etc. In the task scenario, the annotations use the statements from the provenance to create the summaries making it the performance bottleneck. Hence, in the CL-SciSumm 2017, the provenance identification acts at the main task.

The overall task structure is as follows:

- Task 1a: Identifying the provenance of the citance
- Task 1b: Classifying the identified provenance in on of the 6 facets
- Task 2: Using the identified provenance make a summary of the RP (bonus task)

We present our approach for the citation contextualization (task 1a and task 1b) which has been most fundamental and difficult challenge of the complete pipeline.

2 Related Work

In prior attempts, researchers have come up with features like TF-IDF, LCS, Jaccard Similarity etc. computed over the citance and the candidates and run simple linear regression models for getting the score. Selecting the best candidate hence result into the provenance identified from human-engineered feature without considering any comparative suitability with respect to other candidates. The closest approach to ranking was first used for this problem in form of linear optimization constraints using a averaged word embedding representation [5].

3 Method

Here, we discuss statistic of the data for the task.

- CL-SciSumm 2017 training set compromises of 30 training documents (20 and 10 documents from CL-SciSumm 2016 training and test sets respectively [2])
- For each citation, the number of positive samples (provenance) is much sparser than negative samples (non-provenance) being in the order of 1 to 5 out of odd 250 lines. On training data, more than 50% of the citations have only 1 line citance and around 85% have less than 3 line as citance. This makes selecting fewer lines a better strategy.
- The training data for facet is also highly skewed. Out of the facets Hypothesis, Aim, Implication, Results and Method; Method citation makes up more than 50% of the citations which makes selecting Method always a good naïve approach.

3.1 Provenance Detection

Task 1a can be modeled in many possible ways including standard search and retrieval, sequential labeling, classification or ranking. Usually, a high fraction of citation (mainly for facets like Methods) shows high surface level similarity with the original citance and hence using retrieval techniques to capture high syntactic and semantic similarities give satisfactory results. However, this does not cover the cases where there is a less semantic similarity and the provenance are usually implied or entailed. Retrieval based techniques using TF-IDF like features are bound to fail in such cases. This calls for more powerful and general purpose model which has the ability to incorporate both the semantic similarity and learn the higher order relations between the texts.

To incorporate both such components in the model we propose a weighted scoring model as:

$$\text{score} = \alpha_1(\text{surface level closeness score}) + \alpha_2(\text{learned semantic relation score}) \tag{1}$$

To calculate surface level closeness a lot of features and scoring schemes has been proposed in prior runs of the CL-SciSumm [2]. All global statistics based features (like TF-IDF) are calculated treating each line in the RP as a document. A generic framework for incorporating such scoring is:

$$\begin{aligned}\text{surface level closeness score} = \beta_1(\text{TF-IDF score}) + \beta_2(\text{LCS score}) + \\ \beta_3(\text{Jaccard similarity}) + ...\end{aligned} \tag{2}$$

Since the evaluation criteria use the ROUGE-SU4 and exact match for evaluation we use TF-IDF and Longest Common Subsequence (LCS) score to calculate the surface level similarity. We note that this is not the exhaustive combination of features as others features have shown to add to the performance, but we use only basic features to show the validity of our hypothesis and applicability of our method. Further, we assume that other similarity based features could be captured by the semantic relation model. For the semantic relation score, we explore deep learning based models using word embeddings as the features. We experiment with the options of classification and ranking.

Classification Versus Ranking. The classification model as shown in Fig. 3.1 uses RP text and CP text to form training tuples. First, vocabulary indexed text converted to word embedding passes through a Convolutional Neural Network (CNN) or Long Short Term Memory (LSTM). The CNN subpart of the network comprises convolution layer followed by a max pooling layer. The merge layer then does an element-wise multiplication of the activations thus learned for both the text. This models representation level similarity between the two texts. Finally, this representation passes through a feed forward layer which classifies it as either 1 or 0 depending on the label of the RP text. Some practical aspects of training are:

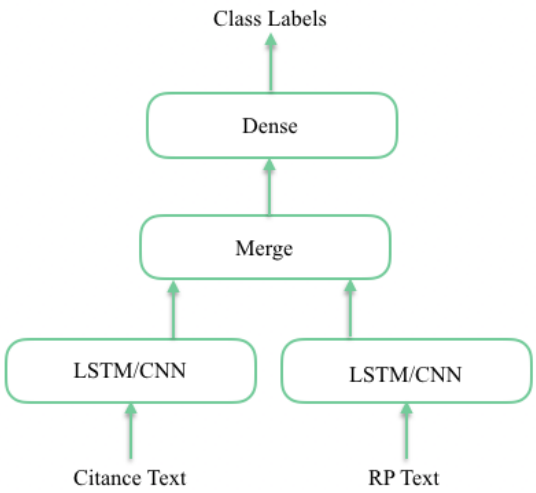

Fig. 1. The classification model

- We avoid overwhelming negative samples while constructing the snippet of RP texts for training. We form samples per line rather than all possible combination of consecutive 1 to 5 lines. This helps in keeping the number of negative samples in reasonable yet varied.
- Another way to keep the ratio of negative samples in check is to down-sample the RP texts. It can be done by random selection or more sophisticated way of filtering text e.g. selecting text only if the TF-IDF is greater than a certain threshold. In our experiments, this way does not give better performance as compared to just using all lines.

Another approach deploying similar architecture but with ranking ability is by incorporating modification as shown in 3.1. The training samples for this model are created by making tuples of two RP texts and one citance text. One of the RP text is always the correct provenance while other is sampled by one of the techniques discussed previously. The class label then predicts out of the two texts which one is better provenance for the citance text. The model predicts 0 or 1 depending on the RR text 1 or RR text 2 is the better provenance. During testing the system returns the RP text which wins the maximum number of pairwise comparison. The benefits of this model as compared with the classification model are:

- It solves the problem of skew class distribution by forming an exactly equal number of tuples of positive and negative samples.
- It adds the ability to learn comparative features from the representations which make one text provenance as compared to other texts.

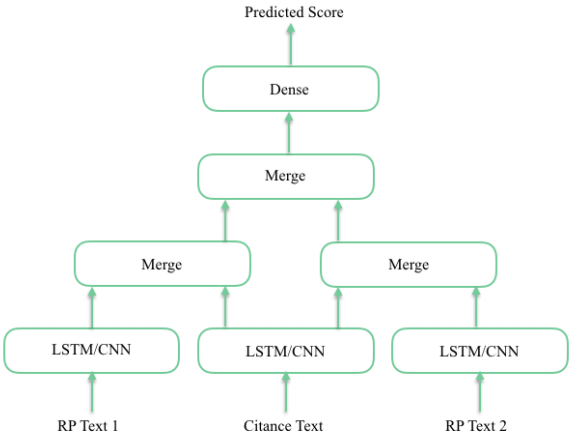

Fig. 2. The ranking model

3.2 Facet Identification

For facet identification, we reuse the classification model however its trained on true provenance and citance samples and the output labels are one of the 5 classes. For balancing the class we use class weights as the log_{1000} of the inverse of the frequency.

4 Results

Now we discuss the results for the different experimental setups we tried. All these experiments are done on the test set of CL-SciSumm 2016 while training on the train set for CL-SciSumm 2016, with no overlap among the sets. The final submitted system uses all the documents for training. The parameters α and β are selected equally to sum to 1. A more sophisticated way of selecting these values is to use development set to learn the coefficients jointly during training, which can be explored in later works. All the neural networks are trained in Keras with small learning rate for 2 iterations using Adagrad optimizer. The input is padded to form a sequence of maximum length 100. GloVe word embedding of 300 dimensions is used and the size of LSTM/CNN experimented is 64. For facet identification, since multiple classes are allowed we pick all classes which are within a certain δ set as 0.05 probability score from the highest score.

The results from Table 1 shows that the classification model does not learn a lot compared with a model which predicts all the RP tests as non-provenance. However, the ranking model does significantly better possibly because of better class distribution and better modeling power of the ranking model compared to classification model as discussed.

The results from Table 2 shows that the proposed model for task 1a and task 1b does not give better results as compared to already proposed traditional

Table 1. Result of classification versus ranking models

	P	R	F_1	F_1 All False
Classification	0.69	0.70	0.69	0.67
Ranking	0.75	0.73	0.75	0.50

Table 2. Results on CL-SciSumm 2016 test set

	P	R	F_1
Task 1a	0.09	0.05	0.07
Task 1b	0.06	0.03	0.04

syntactic similarity based features [3]. Similar trend is observed for CL-SciSumm 2017 blind[1] test set as reported in Table 3. Particularly for task 1b, the results are not even close to simple features based classifiers, even though when CNN work extremely good in sentence classification[4].

Table 3. Results on CL-SciSumm 2017 blind test set

	Evaluation	P	R	F_1
Task 1a	Micro	0.064	0.048	0.055
Task 1a	Macro	0.068	0.056	0.062
Task 1a	ROUGE2	0.078	0.124	0.084
Task 1b	Micro	0.064	0.048	0.055
Task 1b	Macro	0.058	0.017	0.026

5 Discussion

Few observations and conclusions evident from the experiments and results are:

- A lot of documents have large OCR errors. Simple feature-based models are more robust as compared to word embedding based neural models due to a large number of OOV words cause of OCR errors. Hence, passing the data through robust input processing pipeline drives the current state of the art. This would not particularly help neural architecture because its difficult to retain semantics or even find embedding after removing stop words, stemming and other such filters.
- There is a high amount of noise (subjectivity) associated with the annotation. For almost all the deep learning classifier or ranking models training does not result in significant decrease in cross-entropy.

[1] We hereby declare that despite being part of one of the organizing institutions, we did not have access to any additional data, information or help.

– For facet classification, class Method gets selected mostly. Even for task 1b results from Table 2 gets beaten by a simple model always predicting Methods giving an F_1 of 0.23. This may again be because of too many logical sub-classes being annotated together as Method.

References

1. Jaidka, K., Chandrasekaran, M.K., Elizalde, B.F., Jha, R., Jones, C., Kan, M.Y., Khanna, A., Molla-Aliod, D., Radev, D.R., Ronzano, F. and Saggion, H.: The computational linguistics summarization pilot task. In Proceedings of Text Ananlysis Conference, Gaithersburg, USA (2014)
2. Jaidka, K., Chandrasekaran, M.K., Rustagi, S. and Kan, M.Y.: Insights from CL-SciSumm 2016: The Faceted Scientific Document Summarization Shared Task. International Journal on Digital Libraries, 1–9 (2017)
3. Jaidka, K., Chandrasekaran, M.K., Jain, D. and Kan, M.Y.: Overview of the CL-SciSumm 2017 Shared Task. In BIRNDL@ SIGIR, Tokyo, Japan (2017)
4. Kim, Y.: Convolutional Neural Networks for Sentence Classification. arXiv preprint arXiv:1408.5882 (2014)
5. Nomoto, T.: NEAL: A Neurally Enhanced Approach to Linking Citation and Reference. In BIRNDL@ JCDL, 168–174 (2016)