Citation Worthiness of Sentences in Scientific Reports

Hamed Bonab, Hamed Zamani, Erik Learned-Miller, and James Allan
College of Information and Computer Sciences
University of Massachusetts Amherst
Amherst, MA 01003
{bonab,zamani,elm,allan}@cs.umass.edu

ABSTRACT

Does this sentence need citation? In this paper, we introduce the task of *citation worthiness* for scientific texts at a sentence-level granularity. The task is to detect whether a sentence in a scientific article needs to be cited or not. It can be incorporated into citation recommendation systems to help automate the citation process by marking sentences where needed. It may also be useful for publishers to regularize the citation process. We construct a dataset using the ACL Anthology Reference Corpus; consisting of over 1.1M "not_cite" and 85K "cite" sentences. We study the performance of a set of state-of-the-art sentence classifiers for the citation worthiness task and show the practical challenges. We also explore section-wise difficulty of the task and analyze the performance of our best model on a published article.

ACM Reference Format:
Hamed Bonab, Hamed Zamani, Erik Learned-Miller, and James Allan. 2018. Citation Worthiness of Sentences in Scientific Reports. In *SIGIR '18: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, July 8–12, 2018, Ann Arbor, MI, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3209978.3210162

1 INTRODUCTION

Finding proper citations and referring to them appropriately in scientific manuscripts is often a labor-intensive task [7]. Citation recommendation system aims to ease the process by suggesting reference candidates through a two-step interactive procedure. *First,* the user specifies the location in the manuscript where the citation is needed. *Second,* the system ranks the possible candidates from a corpus or bibliographical list. Ranking candidate references as the second step of citation recommendation has been extensively studied in the literature [4, 5, 7, 8, 17]. However, minimizing the authors' effort in the first step is relatively unexplored.

Citation sentences are those where some references to other papers are required—for validating, motivating, or other purposes. In this study, we use linguistic features to detect citation sentences in *sentence-level granularity.* To this end, we define the task of evaluating sentences for citation, in short *citation worthiness.* Indeed, given a sentence s, the citation worthiness task is to classify the sentence to either "cite" or "not_cite" class, i.e., a binary classification task. The task assumes that no sentence in the input text has signatures of citation sentences (citation placeholders "(author(s), year)", the author's name for a cited work, especial phrases like "et al.").

The task we introduce here is similar to the Teufel's *Argumentative Zoning (AZ)* task [18, 19]. Using simple features (e.g., sentence location, length, whether the sentence contains citation, linguistic features of the sentence, etc.), AZ aims to identify and classify scientific text into different pre-specified categories—e.g., background, motivation, or contrasting statements. Teufel later introduced a *citation function* task for predicting the author's reason for citing a given paper with a linguistically inspired solution [21]. Some other similar tasks are defined in the literature, e.g., citation sentiment detection [3], argumentation mining [14], rhetorical classification [9, 20], text summarization using citation sentences [11], reference scope identification [1], and citation recognition in public comments [2]. Contrary to the mentioned studies, *citation worthiness* does not use any external knowledge bases and does not depend on citation signatures.

Table 1 presents four example sentences for each binary label. The objectives for "cite" sentences are based on four main categories presented in [21] and "not_cite" ones are based on argumentative zones [19]. These examples provide an insight into the differences between "cite" and "not_cite" sentences.

2 CITATION WORTHINESS DATASET

To the best of our knowledge, there is no ready-to-use dataset for our task. We construct a *citation worthiness* dataset using the articles of ACL Anthology Reference Corpus (ARC).[1] We use the SEPIC corpus[2] [15], which includes sentence-level segmentation of 10,921 articles from ACL ARC 1.0, up to February 2007. The sentence splitter and chunker of the Apache OpenNLP 1.5 [3] in addition to the Stanford tokenizer and POS tagger, and the MaltParser tools were used.

For our experiments, we see that even though sentences are to some extent cleansed, a few further pre-processing steps are necessary. This is mostly because the text of articles are extracted from their pdf files. We take out the following text from our dataset: (i) Footnotes and conference names repeated through each page of articles, (ii) Title of sections, (iii) The words that were strung together (with missing delimiters), (iv) Mathematical formulas, and (v) URLs.

For annotating sentences with "cite" and "not_cite" labels, we use the signatures of citation sentences. These signatures include citation placeholders "(author(s), year)", the author name of a cited work in the text of sentence, and special phrases like "et al.". For

SIGIR '18, July 8–12, 2018, Ann Arbor, MI, USA

ACM ISBN 978-1-4503-5657-2/18/07…$15.00
https://doi.org/10.1145/3209978.3210162

[1] ACL ARC: https://acl-arc.comp.nus.edu.sg/
[2] SEPID corpus: http://pars.ie/lr/sepid-corpus
[3] Apache OpenNLP 1.5: https://opennlp.apache.org/docs/1.5.3/manual/opennlp.html

Table 1: Example sentences for "cite" and "not_cite" labels—the objectives are based on [19, 21]. The sentences are taken from the ACL Anthology Reference Corpus [6].

L	Example Sentence	Objective
cite	The only known work automating part of a customer service center using natural language dialogue is [ref]	POSITIVE
	There have been several classic projects in the area of natural language dialogue like trains trips project [ref]	NEUTRAL
	The underlying decoding strategies [ref] are too time consuming for our application	NEGATIVE
	We shall not discuss the method [ref] by which determines the most similar training example	WEAK
not_cite	Some sample entries are for each tuple input, the filler checks off the fields which correspond to the tuple	TECHNICAL
	Since customer service centers are used by a variety of users, we needed a user independent system	MOTIVATING
	We think that the effect of aggregation spreads from text planning to sentence realization	OPINIONS
	We are concerned with lexical knowledge for specifying rules representing the blocks of our parsing system	CONCERNS

detecting these signatures, we define different rules. It is mostly due to different standards through the dataset—e.g., "(author(s), year as [YYYY])", "author <sentence text> (year as [YYYY])", "(author(s), year as [YY])", "[<citation number>]". Whenever any of these signatures is detected, the sentence is categorized as "cite" and all of these signatures are omitted from the sentence. Otherwise, the sentence is categorized as "not_cite." We only keep the alphanumerical characters. Considering these, we constructed *citation worthiness* dataset with 85,778 sentences with the "cite" label and 1,142,275 sentences with the "not_cite" label. We made the dataset public.[4]

3 CLASSIFICATION APPROACHES

We exploit a set of state-of-the-art sentence classifiers in our experiments, ranging from neural network approaches to robust linear classifiers.

3.1 CNN-based Classifier

Figure 1 presents the architecture of the CNN-based sentence classifier we use for our classification task. The effectiveness of this CNN architecture on multiple text sentiment classification benchmarks is investigated in [10]. A sensitivity analysis on the model is presented by Zhang and Wallace [24]. We adapt the model used in [12].

The word embedding vectors are used for constructing the sentence embedding matrix with $n \times k$ dimensions (n representing the maximum number of words taken from the sentence and k denotes the embedding dimensionality). Let l_h be the width of the widest filter in the network. Concatenating the word embedding of each sentence and padding by $l_h - 1$ zero vectors with k dimensions result in a matrix of $k \times (n + l_h - 1)$ dimensions as the input to the network.

A convolutional operation with filter of $w \in \mathbb{R}^{h \times k}$ is applied on h words in a given sentence to produce ngram-based features. Then, a max-over-time pooling operation is applied over the feature map and takes the maximum value as the feature corresponding to each particular filter. This is for capturing the most important feature for each map. It also deals with variable sentence length. The model uses multiple filters to obtain multiple features from a given sentence. These features form the penultimate layer and are passed to a fully connected softmax layer whose output is a probability distribution over the labels. Dropout is used for regularization. Three scenarios are designed and experimented as below:

[4]Citation Worthiness dataset: https://ciir.cs.umass.edu/downloads/sigir18_citation/

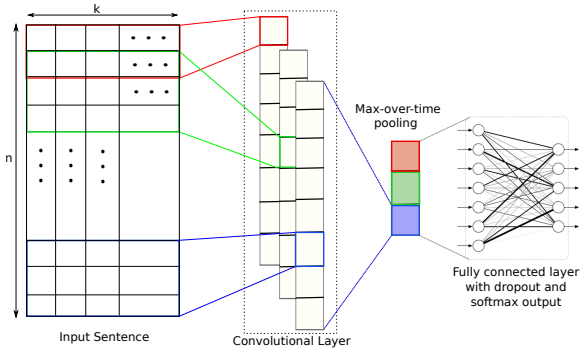

Figure 1: Architecture of the CNN-based Classifier.

- **CNN-rnd-update.** Initializing word embeddings randomly and letting the model to update these embedding vectors along with other parameters of the network.
- **CNN-w2v-static.** Initializing word embeddings using pre-trained word embeddings provided by the word2vec model [13] trained on Google News. This model does not let the model to update the embedding vectors.
- **CNN-w2v-update.** Initializing the embeddings using pre-trained word vectors, similar to CNN-w2v-static. This model allows the embedding vectors to be updated.

3.2 Linear Classifiers

Multinomial Naive Bayes (MNB) and Support Vector Machine (SVM) are used for many text classification tasks in the literature. Although the models are well-known, their feature set definition and model variation can impact the performance significantly [22]. Here, we explain the model variant and feature set we use for our task, formulated as linear classifiers, based on [22]. For a given test sentence s, with feature count vector $x^s \in \mathbb{R}^{|F|}$, the predicted label $y^s \in \{-1, 1\}$, can be modeled as following. F is the set of defined features.

$$y^s = sign(w^T x^s + b) \quad (1)$$

MNB. It is more reliable compared to other variations of the Naive Bayes (NB) model [16]. An indicator function is applied on feature count vector. Afterwards, the log-count ratio is defined as $r = \log(\frac{p/||p||_1}{q/||q||_1})$ where $p = \alpha + \sum_{i:y^i=1} x^i$ and $q = \alpha + \sum_{i:y^i=-1} x^i$. We set the smoothing parameter α to 1 in our experiments. Regarding Equation (1) modeling, $w = r$ and $b = \log(N_+/N_-)$ are defined for MNB.

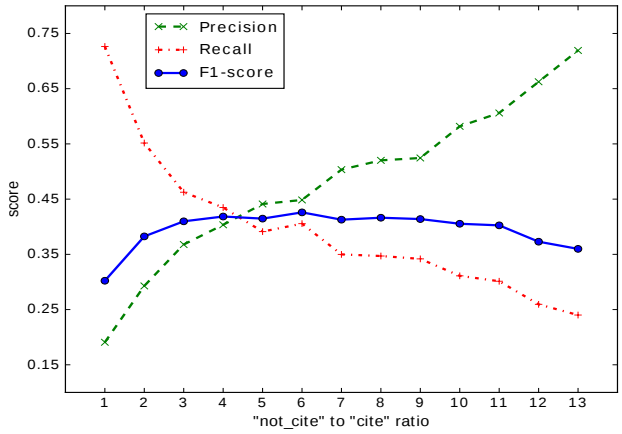

Figure 2: Down-sampling sensitivity analysis on the ratio of "not_cite" to "cite" sentences of the training set for CNN-w2v-update.

SVM. We use the L2-loss with a L2-regularization and 5 iterations for minimizing w and b. The same features as MNB are used for SVM. The regularization constant is set to 0.001.

NBSVM. It is an interpolation between MNB and SVM classifiers with parameter $\beta = 0.25$ [22].

Note that we use unigram and bigram tokenizations for our feature count vector of the linear classifiers. In addition, the parameters here are chosen based on a grid search on possible variants.

4 EXPERIMENTAL SETUP AND RESULTS

The dataset is split to 80% training, 10% validation, and 10% testing chunks. We report Precision, Recall, and F1-score for "cite" sentences and Accuracy of the prediction on the test set. The collected dataset is imbalanced, i.e., the number of "not_cite" sentences are 13 times higher than "cite" sentences. To this end, a random selection procedure is used for down-sampling the training data with different ratio of "not_cite" to "cite" sentences and the best ratio is selected using the performance on the validation data. Figure 2 presents the sensitivity of CNN-w2v-update performance with respect to the "not_cite" to "cite" ratio. The results suggest that the best ratio is 6 for CNN model as the peak point for F1-score values. It also shows the trade-off between recall and precision on different ratios. Linear classifiers tend to give the highest performance with ratio of 1.

The maximum sentence length is set to 100 in our experiments. We also use all the terms occurring in the training sentences to construct vocabulary. For the cases where a pre-trained word embedding is loaded, random initialization is used for the terms not in the pre-trained vocabulary set. The word embedding dimension is set to 300. In addition, using a grid search over the validation set, we use rectified linear unit (ReLU) as activation function, filter window sizes of [3, 4, 5] with 100 feature maps each, dropout rate of 0.5, mini-batch size of 64, and learning rate of 10^{-4}. Training process uses stochastic gradient descent over shuffled mini-batches with Adam optimizing update rule. In addition, the experimental results presented here are based on 4 epochs for static and 10 epochs for updating scenarios, selected based on early stopping on the validation set.

Table 2: Experimental Results. Note that Precision, Recall, and F1-score values are measured for "cite" class label. For each column, the highest value is marked with bold-face.

Method	Acc(%)	Precision	Recall	F1-score
CNN-rnd-update	91.89	0.4180	0.4086	0.4133
CNN-w2v-static	**93.16**	**0.5271**	0.1992	0.2892
CNN-w2v-update	92.36	0.4485	0.4056	**0.4260**
MNB	66.22	0.1468	**0.7974**	0.2480
SVM	72.57	0.1569	0.6694	0.2542
NBSVM	92.06	0.4117	0.3193	0.3597

Table 2 presents our experimental results. Comparing CNN-w2v-static with CNN-rnd-update shows that CNN classifier can learn the word embedding vectors with random initialization. Due to the imbalance nature of the test data, accuracy values are misleading [23]. We include them in the table to give a sense of the classifiers performance on "not_cite" sentences. As a general conclusion, it seems that CNN-w2v-update and NBSVM perform more reliable on recognizing citation sentences, compared to other variants in CNN-based and linear classifiers.

5 ANALYSIS

5.1 Section-wise Analysis

In order to study the difficulty of citation worthiness detection for sentences of each section, we divided our test data into 7 different sections. Table 3 presents the list of sections, the number of "cite" and "not_cite" sentences for each section, and classification performance of the CNN-w2v-update classifier. Since we are using the same validation data as the previous section, the ratio is set to 6. Abstract section has the highest accuracy and lowest F1-score. This might be due to the rare occurrences of citation sentences in this section—only 2.5% of abstract sentences are citation sentences. For related work and acknowledgments sections 31.1% and 10.1% of sentences are citation sentences; could be an explanation on the high values of precision for these sections. The Method, Evaluation, and Conclusion sections include about 8% of citation sentences in each section, where the model achieves the highest citation worthiness performance, in terms of F1-score. This analysis suggests that it might be useful to train separate section-wise classifiers to improve the overall performance of the system.

5.2 Published Paper Analysis

In order to measure the performance of the model on a sample paper, we extract the sentences of [10] as an example article. In total 90 sentences are extracted from the body of the paper. We manually labeled and removed the citation signatures from each sentence. We train CNN-w2v-update and NBSVM classifiers using the training data explained in the previous section, and test with the 90 extracted sentences. Table 4 presents the confusion matrix of predictions for both classifiers. It can be seen that true positive and false negative numbers are higher for CNN-w2v-update compared to NBSVM, showing its higher reliability.

Comparing the prediction of both methods for sentences in the paper shows that both are failing on the exact same sentences. For example, both algorithms miss-classify the following sentences. It

Table 3: Section-wise analysis of citation worthiness. CNN-w2v-update is used in this experiment. For each column, the highest value marked with bold-face.

Section	#pos	#neg	Acc(%)	Prec.	Recall	F1-score
Abstract	392	15286	**94.06**	0.1811	0.3903	0.2474
Introduction	1133	15529	92.62	0.4491	0.3742	0.4083
Related Work	200	443	70.76	**0.5508**	0.3250	0.4088
Method	**5720**	**70085**	92.37	0.4925	0.3743	0.4253
Evaluation	446	5738	92.69	0.4920	0.4148	0.4501
Conclusion	537	5702	91.62	0.5154	**0.4357**	**0.4722**
Acknowledg.	117	1044	90.27	0.5270	0.3334	0.4084

1. #pos, #neg: the number of "cite" and "not_cite" sentences, respectively,
2. 'opening' sentences are included into abstract section, 'background' and 'general terms' are included into related work, and 'discussion' are included into evaluation section.

seems that when a special phrase related to a specific concept is the reason for citation, both algorithms fail.

—"*the model architecture shown in figure NUM is a slight variant of* ***the cnn architecture of***"

—"***trec question dataset*** *task involves classifying a question into NUM question types whether the question is about person location numeric information etc*"

On the other hand, both algorithms categorized following sentences as citation sentences. Both of these sentences do not have any citation in the original paper. The author of the paper did not put citation for both of these since the related citations are mentioned in the previous sentences.

—"*we use the publicly available word2vec vectors that were trained on NUM billion words from google news*"

—"*in such dense representations semantically close words are likewise close in euclidean or cosine distance in the lower dimensional vector space*"

This analysis suggests that exploiting such a system in practice, possibly as part of a paper writing software, can ease the authors' effort. We also show that considering the sentence context is important for identification of citation worthiness. In addition, there exist "not_cite" sentences in scientific articles that actually require citation. These sentences are assigned incorrect labels by rule-based automatic labeling of sentences, as done in this paper.

6 CONCLUSION

In this study, we introduced the task of *citation worthiness* for scientific reports in sentence-level granularity. We exploited a set of state-of-the-art sentence classifiers showing the feasibility of the task and practical challenges. The section-wise difficulty of the task was studied for seven widely used sections, and the real-world applicability of the methods was examined on a published article. The results suggested that it could be interesting to train individual section-wise classifiers, and design context-aware classifiers that exploit previous sentences as the context. We also intend to incorporate citation worthiness into an end-to-end citation recommendation system. Further analysis on the reasons for citing a paper, and proper citation placement on different granularity might be also beneficial.

Acknowledgements. This work was supported in part by the Center for Intelligent Information Retrieval and in part by NSF grant #IIS-1160894. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

Table 4: Confusion Matrices for published paper analysis using the CNN-w2v-update and NBSVM classifiers.

		True Label: cite	True Label: not_cite	Total
Predicted Label (**CNN-w2v-update**)	cite	21	20	41
	not_cite	2	47	49
	Total	23	67	90

		cite	not_cite	Total
Predicted Label (**NBSVM**)	cite	20	30	50
	not_cite	3	37	40
	Total	23	67	90

REFERENCES

[1] Amjad Abu-Jbara and Dragomir Radev. 2012. Reference scope identification in citing sentences. In *NAACL*. 80–90.

[2] Jaime Arguello, Jamie Callan, and Stuart Shulman. 2008. Recognizing citations in public comments. *JITP* 5, 1 (2008), 49–71.

[3] Awais Athar. 2011. Sentiment analysis of citations using sentence structure-based features. In *ACL (student session)*. 81–87.

[4] Krisztian Balog and Heri Ramampiaro. 2013. Cumulative citation recommendation: classification vs. ranking. In *SIGIR*. 941–944.

[5] Krisztian Balog, Heri Ramampiaro, Naimdjon Takhirov, and Kjetil Nørvåg. 2013. Multi-step classification approaches to cumulative citation recommendation. In *OAIR*. 121–128.

[6] Steven Bird, Robert Dale, Bonnie J Dorr, Bryan R Gibson, Mark Thomas Joseph, Min-Yen Kan, Dongwon Lee, Brett Powley, Dragomir R Radev, Yee Fan Tan, and others. 2008. The ACL Anthology Reference Corpus: a reference dataset for bibliographic research in computational linguistics. In *LREC*.

[7] Qi He, Jian Pei, Daniel Kifer, Prasenjit Mitra, and Lee Giles. 2010. Context-aware citation recommendation. In *WWW*. 421–430.

[8] Wenyi Huang, Zhaohui Wu, Liang Chen, Prasenjit Mitra, and C Lee Giles. 2015. A Neural Probabilistic Model for Context Based Citation Recommendation.. In *AAAI*. 2404–2410.

[9] Rahul Jha, Amjad-Abu Jbara, Vahed Qazvinian, and Dragomir R Radev. 2017. NLP-driven citation analysis for scientometrics. *JNLE* 23, 1 (2017), 93–130.

[10] Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* (2014).

[11] Wendy Lehnert, Claire Cardie, and Ellen Riloff. 1990. Analyzing research papers using citation sentences. In *CogSci*. 511–18.

[12] Amit Mandelbaum and Adi Shalev. 2016. Word embeddings and their use in sentence classification tasks. *arXiv preprint arXiv:1610.08229* (2016).

[13] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*. 3111–3119.

[14] Raquel Mochales Palau and Marie-Francine Moens. 2009. Argumentation mining: the detection, classification and structure of arguments in text. In *ICAIL*. 98–107.

[15] Behrang QasemiZadeh, Paul Buitelaar, and Fergal Monaghan. 2010. Developing a dataset for technology structure mining. In *ICSC*. 32–39.

[16] Jason D Rennie, Lawrence Shih, Jaime Teevan, and David R Karger. 2003. Tackling the poor assumptions of naive bayes text classifiers. In *ICML*. 616–623.

[17] Trevor Strohman, W. Bruce Croft, and David Jensen. 2007. Recommending citations for academic papers. In *SIGIR*. 705–706.

[18] Simone Teufel. 1997. Sentence extraction as a classification task. *Intelligent Scalable Text Summarization* (1997).

[19] Simone Teufel. 1999. *Argumentative zoning: information extraction from scientific text.* Ph.D. Dissertation.

[20] Simone Teufel and Marc Moens. 1998. Sentence extraction and rhetorical classification for flexible abstracts. In *AAAI*. 89–97.

[21] Simone Teufel, Advaith Siddharthan, and Dan Tidhar. 2006. Automatic classification of citation function. In *EMNLP*. 103–110.

[22] Sida Wang and Christopher D Manning. 2012. Baselines and bigrams: Simple, good sentiment and topic classification. In *ACL*. 90–94.

[23] Hamed Zamani, Pooya Moradi, and Azadeh Shakery. 2015. Adaptive User Engagement Evaluation via Multi-task Learning. In *SIGIR*. 1011–1014.

[24] Ye Zhang and Byron Wallace. 2015. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *arXiv preprint arXiv:1510.03820* (2015).