Unsupervised Partial Sentence Matching for Cited Text Identification

Kathryn Ricci* **Haw-Shiuan Chang*** **Purujit Goyal** **Andrew McCallum**
[1]CICS, University of Massachusetts, Amherst
kathryn.d.ricci@gmail.com, hschang@cs.umass.edu
purujitgoyal@umass.edu, mccallum@cs.umass.edu

Abstract

Given a citation in the body of a research paper, cited text identification aims to find the sentences in the cited paper that are most relevant to the citing sentence. The task is fundamentally one of sentence matching, where affinity is often assessed by a cosine similarity between sentence embeddings. However, (a) sentences may not be well-represented by a single embedding because they contain multiple distinct semantic aspects, and (b) good matches may not require a strong match in all aspects. To overcome these limitations, we propose a simple and efficient unsupervised method for cited text identification that adapts an asymmetric similarity measure to allow partial matches of multiple aspects in both sentences. On the CL-SciSumm dataset we find that our method outperforms a baseline symmetric approach, and, surprisingly, also outperforms all supervised and unsupervised systems submitted to past editions of CL-SciSumm Shared Task 1a.

1 Introduction

The goal of a sentence-matching task is to extract a sentence that is most relevant to the query sentence from a collection of candidate sentences. In addition to information retrieval (IR) methods, a common unsupervised approach to sentence-matching tasks is to represent the query and candidate sentences by dense vectors, each computed by averaging the (contextualized) word embeddings corresponding to all constituent words in the sentence (Milajevs et al., 2014; Arora et al., 2017).

In this way, all semantic aspects of each sentence are collapsed into a single embedding representing the entirety of its semantics. By applying a cosine similarity to each query-candidate pair of these embeddings to evaluate affinity, the implicit assumption is that the most similar pair of sentences should contain exactly the same set of semantic aspects.

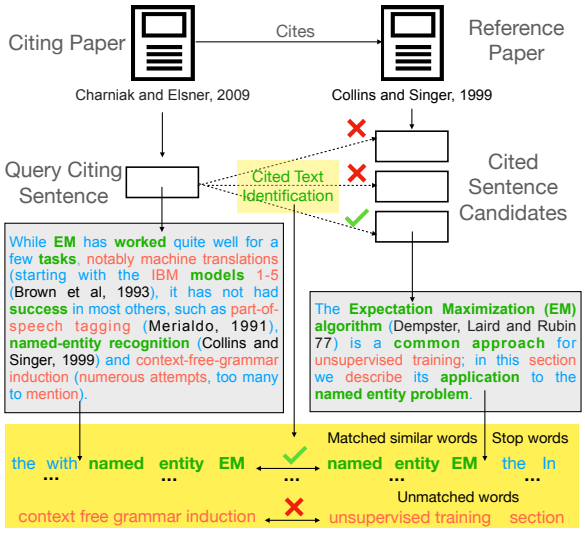

Figure 1: A sample query citing sentence and gold cited sentence from the CL-SciSumm dataset illustrating how shared semantic aspects, emphasized in the figure, may be accompanied by additional aspects in both sentences.

However, this approach is suboptimal for some applications, such as the task of identifying the "target" cited sentence(s) from a reference academic paper given a "query" citing sentence. As the example in Figure 1 shows, the target matching cited sentence may contain extra semantic aspects in addition to those that are shared, perhaps providing further details. Similarly, the query citing sentence might contain extra aspects referring to other work or to the relation of the cited information to the citing paper.

Motivated by this observation, we propose a simple and efficient unsupervised method that can accommodate extra semantic aspects in both query and candidates in the cited sentence identification task. To achieve this, our method employs an asymmetric sentence similarity measure to ignore words in the candidate that have little similarity to any query words, and we introduce a scaling function that de-emphasizes the unmatched words in the query citing sentence as well.

* Equal contribution

Proceedings of the Third Workshop on Scholarly Document Processing, pages 95–104
October 12–17, 2022.

On F1 of CL-SciSumm Shared Task 1a (Chandrasekaran et al., 2019, 2020), our method outperforms the corresponding symmetric similarity baseline, a strong unsupervised IR approach (Aumiller et al., 2020), and the best supervised approach among the past submissions between 2018 and 2020, which ensembles four BERT-based models (Chai et al., 2020).

2 Method

Figure 2 illustrates our similarity estimation method given a pair of sentences. In Section 2.1, we ignore the details in the cited sentence candidate and only consider its matched words. In Section 2.2, we softly remove the stop words because the similarity score should not consider the number of matched stop words. Finally, we reduce the influence of irrelevant words in the query citing sentence and let the similarity score be determined more by the exactly matched words in Section 2.3.

2.1 Asymmetric Sentence Similarity Measure

Kobayashi et al. (2015) perform extractive summarization by extracting the summary sentences that cover the original document best. Inspired by their work, we extract the cited sentences that cover the query citing sentence best, which means not penalizing the details or extra words in the cited sentences.

Specifically, we represent the query citing sentence as a multiset of the word embeddings. For each token in the query citing sentence, we find the most similar word in the extracted sentence candidate, and compute the asymmetric similarity score $\mathrm{sim}(S_q, S_c)$ as

$$\sum_{\boldsymbol{w}_q \in S_q} W(w_q) \max_{\boldsymbol{w}_c \in S_c} \sigma(\boldsymbol{w}_q^T \boldsymbol{w}_c), \qquad (1)$$

where $\boldsymbol{w}_q$ are the embeddings of the constituent words w in the query sentence S_q and $\boldsymbol{w}_c$ are the word embeddings from the cited sentence candidate S_c. The word embeddings are normalized by their l^2-norms so that the dot product between two word embeddings is their cosine similarity. $W(w_q)$ is the weight of word w_q and σ is a scaling function, which are detailed in Section 2.2 and Section 2.3, respectively.

We output the top K sentences S_c with the highest similarities to the query citing sentence $\mathrm{sim}(S_q, S_c)$. We find that this optimization method is better than the greedy selection for extractive summarization proposed in Kobayashi et al. (2015). See Appendix C.1 for details.

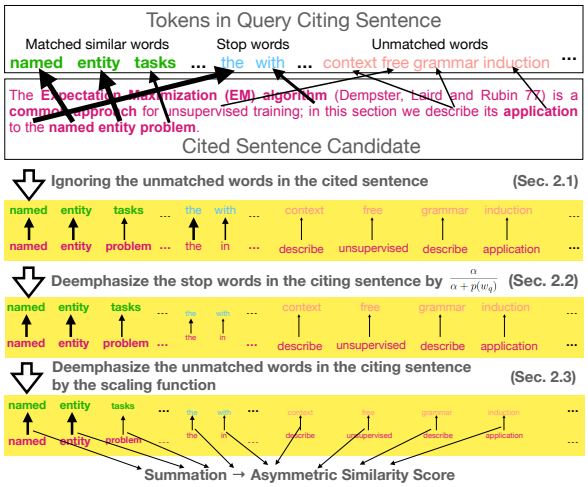

Figure 2: Illustration of our asymmetric similarity estimation. Smaller font sizes or arrows indicate smaller contribution to the output similarity score. Our method can extract the partially matched cited sentence candidates by decreasing the influence of unmatched words and stop words to highlight the matched semantic words.

2.2 Inverse Frequency Weighting

Unlike Kobayashi et al. (2015) which treats each word equally, we assign a lower weight to a common word (e.g., a stop word) in the query citing sentence because a high-frequency word is naturally more likely to be matched to irrelevant sentences in the cited paper.

Following Arora et al. (2017), we set the weight of the word w_q in Equation 1 as

$$W(w_q) = \frac{\alpha}{\alpha + p(w_q)}, \qquad (2)$$

where frequency probabilities $p(w_q)$ are computed by $\frac{f(w_q)}{N}$, $f(w_q)$ is the frequency of words, and N is total number of words in the corpus. We let $\alpha = 10^{-4}$, which is a typical value suggested by Arora et al. (2017).

2.3 Scaling Function for Word Similarities

In Figure 2, the correct sentence pair only shares a few terms, such as *named entity*, while there are several unmatched words in the query citing sentence, such as *context free*. To let the matching terms in the query contribute more to the final similarity score than the unmatched words, we set the scaling function in Equation 1 to be

$$\sigma(x) = x^d, \qquad (3)$$

where $d > 0$ is a fixed hyperparameter.

When d is large, our method effectively ignores the cosine similarities that are smaller than 1, which means it only considers the exact lexical matching words. In contrast, a small d encourages the cited sentences to contain more words that are topically related to the words in the query. We can tune d to balance the hard matching and soft matching.

3 Experiments

We evaluate our method on the CL-SciSumm dataset (Chandrasekaran et al., 2019), comparing our results to past submissions to Shared Task 1a. In the official evaluation, performance is measured based on sentence overlap F1 and ROUGE-S*[1] (Lin and Och, 2004), both micro-averaged over all sentences selected by all annotators.

We train 300-dimensional word embeddings using Word2Vec skip-gram (Mikolov et al., 2013). We use the ACL Anthology Reference Corpus Version 2 (Bird et al., 2008) as our training corpus, because the papers in CL-SciSumm are sampled from the computational linguistics domain. For each query citing sentence in the corpus, we select the top $K = 2$ sentences from our candidate ranking to submit for evaluation. All the hyperparameters are experimentally chosen to maximize average F1 scores on the training set.

3.1 Preprocessing

We use a regular expression to remove citation markers (e.g., of the form (*Author, Year*)) from the word-embedding training corpus, citing sentences, and candidate sentences. These markers do not contribute to the semantics of the sentences, yet the weights of these low-frequency markers in Equation 2 are high and the markers may erroneously match with words in our similarity computations. Our ablation study in Appendix C.2 finds that omitting this preprocessing step indeed significantly degrades performance.

Our objective function in Equation 1 encourages the selected cited sentence candidates to cover the query citing sentence. The method has a preference for selecting longer sentences because the asymmetric similarity measurement does not penalize the unmatched details in the cited sentences, and more words in each candidate tend to cover the query sentence better (Kobayashi et al., 2015).

[1]We discover that the official evaluation script outputs ROUGE-S* rather than ROUGE-SU4.

Best-Performing Model Configuration (and Tuning Range)		
Asymmetry Direction:	Candidate Covers Query	(or Reverse)
Word Similarity:	Cosine	(or Dot Product)
Optimization:	Top K	(or Greedy)
Extracted Sent. Num. K:	2	(or 1-10)
Weights of Query Words:	Arora et al. (2017)	(or Uniform)
Scaling Function Power (d):	4	(or 1-10)
Citation Markers:	Remove	(or Keep)
Truncation:	After 100 Tokens	(or 50 or None)
Casing:	Cased	(or Uncased)
Word2Vec Min. Word Count:	35	(or 50 or 100)

Table 1: Configuration of our best-performing Word2Vec-based model, **Asymm (d=4)**, on CL-SciSumm training set. The hyperparameters in parentheses are the ranges we tested.

Our scaling function alleviates the problem by emphasizing the exactly matched words. To further alleviate the issue, we truncate sentences to a chosen maximum length under the assumption that most of the relevant semantic aspects occur at the beginning of a long citing or candidate sentence.

3.2 Model and Baselines

We consider the following methods (see Appendix C.2 for more ablation baselines).

- **Asymm (d=4)**: Our proposed asymmetric method with the configuration in Table 1, the best-performing Word2Vec-based configuration on the training set. *d* refers to the power of our scaling function in Equation 3.
- **Asymm (d=1)**: Same configuration as **Asymm (d=4)** but using the trivial scaling function $\sigma(x) = x$.
- **Symm**: The symmetric method that computes a cosine similarity between average word embeddings (Milajevs et al., 2014). Our best **Symm** configuration removes stop words and does not employ the inverse frequency weighting of Arora et al. (2017), which we found to lower performance in our experiments.
- **Asymm SciBERT (d=4)**: Replacing Word2Vec in **Asymm (d=4)** with SciBERT (Beltagy et al., 2019).
- **BERT ensemble**: Best-performing submission to Shared Task 1a from 2018-2020 (Chai et al., 2020). The supervised approach creates an ensemble of four SciBERT-based models. They also set the number of output sentences $K = 2$.
- **BM25 ensemble**: An unsupervised retrieval

method proposed by Aumiller et al. (2020)[2] that considers the exact term overlap using BM25 (Robertson and Walker, 1994). The approach, which achieves the second-best F1 score on Shared Task 1a of all 2018-2020 submissions, is an ensemble of two search configurations with additional preprocessing steps to remove citation markers, as we do, and to mask math-like text.

Notice that both **BERT ensemble** and **BM25 ensemble** utilize the position information of the candidate sentence within the reference text, while all of our methods do not make any assumption on the position of extracted sentences.

3.3 Main Result

The results in Table 2 show that, according to F1, **Asymm (d=4)** outperforms **Asymm (d=1)** and **Symm**. On the test set, **Asymm (d=4)** outperforms **BERT ensemble** and **BM25 ensemble** in terms of F1, with the latter's reported F1 and ROUGE scores similar to those of **Asymm (d=1)**. This demonstrates that the unsupervised approach for cited text identification can outperform supervised approach due to the small training dataset size.

We observe that the performance of **Asymm SciBERT (d=4)** is inconsistent on training and test data. On the test set, Word2Vec significantly outperforms SciBERT. One reason might be that the keywords in ACL papers are less ambiguous compared to other text domains such as news. The result also highlights the advantages of the non-contextualized word embeddings: we can easily weight or mask individual word embeddings when matching the sentences. It is also much more efficient to train Word2Vec on a new corpus and encode a new sentence into their word embeddings.

4 Related Work

A variety of unsupervised approaches to sentence-matching tasks have been proposed. A traditional method uses an average (contextualized) word embedding as a sentence representation and computes a cosine similarity between query and candidate embeddings (Milajevs and Purver, 2014; Arora et al., 2017). Another approach solves optimal transportation to match the words between two sentences (Kusner et al., 2015). In addition, Skip-Thought (Kiros et al., 2015), BERT (Devlin et al., 2019), and SimCSE (Gao et al., 2021) encode the sentence into a single embedding to predict the nearby sentences or augmented original sentence. These methods assume that all the semantic aspects in a sentence should be matched and lack a way to emphasize the matched aspects.

Kobayashi et al. (2015) propose an asymmetric similarity measure to be used in unsupervised extractive summarization. BERTScore (Zhang et al., 2020) automatically evaluates generated text using similar asymmetric similarity scores. The coverage score from the generation to reference is its recall, and the score with the reverse direction is its precision. However, they do not use the asymmetric similarity to solve partial sentence matching tasks such as cited text identification.

There are also many supervised approaches for estimating the relevancy of two sentences. For example, the approaches built on BERT include the cross-encoder model (Devlin et al., 2019), bi-encoder model (Sentence-BERT) (Reimers and Gurevych, 2019), and the model that maximizes the coverage score from the retrieved document to the query (ColBERT) (Khattab and Zaharia, 2020). Although effective, these approaches often require a large training dataset to learn a good sentence-matching. Thus, such methods might not perform well in scientific sentence-matching tasks where annotations are very limited and expensive.

Method	Training Set			Test Set		
	Recall	F1	R-S*	Recall	F1	R-S*
Symm	15.5	13.5	12.0	18.0	12.4	9.6
Asymm (d=1)	18.0	15.6	10.2	23.1	16.0	11.3
Asymm (d=4)	18.8	16.4	11.2	**25.1**	**17.4**	12.9
Asymm SciBERT (d=4)	**19.5**	**17.0**	**12.2**	22.1	15.3	11.4
BM25 ensemble†	–	–	–	–	16.1	11.3
BERT ensemble♮♯	–	–	–	24.6	17.2	**14.7**

Table 2: Results of evaluation on the CL-SciSumm training and test sets. All scores are reported as percentages. ♮ a supervised method. † results taken from Aumiller et al. (2020). ♯ results taken from Chai et al. (2020).

5 Conclusion

We observe that many target cited sentences and query citing sentences are only partially matched, which motivates us to propose a simple asymmetric sentence similarity measurement that down-weights or masks the unmatched words, stop words, and citation markers. With only a few training labels, learning the prior weighting on contextualized word embeddings could be challenging, and

[2] Aumiller et al. (2020) also propose a two-stage re-ranking approach using a BERT re-ranker, but the second stage does not improve the result.

we suspect that this is the main reason that our simple unsupervised approach could outperform a well-tuned BERT-based supervised approach.

6 Acknowledgement

We thank Kaivankumar Shah and Anjali Ramaprasad for their preliminary exploration of this project. We also thank Gully Burns and Boris Veytsman for their constructive feedback. This work was supported in part by the Center for Data Science and the Center for Intelligent Information Retrieval, in part by the Chan Zuckerberg Initiative under the project Scientific Knowledge Base Construction, in part by the IBM Research AI through the AI Horizons Network, in part using high performance computing equipment obtained under a grant from the Collaborative R&D Fund managed by the Massachusetts Technology Collaborative, and in part by the National Science Foundation (NSF) grant numbers IIS-1922090 and IIS-1763618. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

7 Ethical and Broader Impact

There are several potential applications of our approach. For example, it could be used to accelerate the labeling process, trace the claims made by the citing sentence to verify their correctness, or serve as a baseline for future supervised cited text identification approaches.

One potential risk of our approach is that its assumptions might not be always valid and might create biases in downstream applications. For example, we assume that high-frequency words or unmatched words are less important in cited text identification tasks. This assumption could bias our method toward outputting longer sentences with more low-frequency words, which might be less comprehensible to users.

References

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A simple but tough-to-beat baseline for sentence embeddings. In *International conference on learning representations*.

Dennis Aumiller, Satya Almasian, Philip Hausner, and Michael Gertz. 2020. UniHD@CL-SciSumm 2020: Citation extraction as search. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 261–269, Online. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620.

Steven Bird, Robert Dale, Bonnie Dorr, Bryan Gibson, Mark Joseph, Min-Yen Kan, Dongwon Lee, Brett Powley, Dragomir Radev, and Yee Fan Tan. 2008. The ACL Anthology reference corpus: A reference dataset for bibliographic research in computational linguistics. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Ling Chai, Guizhen Fu, and Yuan Ni. 2020. NLP-PINGAN-TECH @ CL-SciSumm 2020. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 235–241, Online. Association for Computational Linguistics.

Muthu Kumar Chandrasekaran, Guy Feigenblat, Eduard Hovy, Abhilasha Ravichander, Michal Shmueli-Scheuer, and Anita de Waard. 2020. Overview and insights from the shared tasks at scholarly document processing 2020: Cl-scisumm, laysumm and longsumm. In *Proceedings of the first workshop on scholarly document processing*, pages 214–224.

Muthu Kumar Chandrasekaran, Michihiro Yasunaga, Dragomir Radev, Dayne Freitag, and Min-Yen Kan. 2019. Overview and results: Cl-scisumm shared task 2019. In *In Proceedings of Joint Workshop on Bibliometric-enhanced Information Retrieval and NLP for Digital Libraries (BIRNDL 2019)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910.

Gautier Izacard, Mathild Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

Kokil Jaidka, Michihiro Yasunga, Muthu Chandrasekaran, Dragomir Radev, and Min-Yen Kan.

2018. The cl-scisumm shared task 2018: Results and key insights. In *BIRNDL @ SIGIR 2018*.

Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and effective passage search via contextualized late interaction over bert.

Ryan Kiros, Yukun Zhu, Russ R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Hayato Kobayashi, Masaki Noguchi, and Taichi Yatsuka. 2015. Summarization based on embedding distributions. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1984–1989.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *International conference on machine learning*, pages 957–966. PMLR.

Chin-Yew Lin and Franz Josef Och. 2004. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 605–612.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NeurIPS*.

Dmitrijs Milajevs, Dimitri Kartsaklis, Mehrnoosh Sadrzadeh, and Matthew Purver. 2014. Evaluating neural word representations in tensor-based compositional settings. In *EMNLP*.

Dmitrijs Milajevs and Matthew Purver. 2014. Investigating the contribution of distributional semantic information for dialogue act classification. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)*, pages 40–47, Gothenburg, Sweden. Association for Computational Linguistics.

Ori Ram, Gal Shachaf, Omer Levy, Jonathan Berant, and Amir Globerson. 2022. Learning to retrieve passages without supervision. In *NAACL*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Stephen E Robertson and Steve Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR'94*, pages 232–241. Springer.

sbert.net. 2021. `sentence-transformers all-mpnet-base-v2`.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Appendix Overview

In the appendix, we list our main contributions in Appendix B, conduct more experiments and analyses in Appendix C, provide more details of CL-SciSumm Shared Task 1a in Appendix D, provide more details of computing our objective function in Appendix E, and discuss some potential future work in Appendix F.

B Main Contributions

- Inspired by Kobayashi et al. (2015), we propose a sentence-matching model that allows both query sentence and retrieved sentence to contain unmatched semantic aspects.
- We discover that some preprocessing steps such as removing the citation markers are crucial in a cited text identification task.
- Our extensive experiments on CL-SciSumm Shared Task 1a show that a simple, efficient, and unsupervised method based on Word2Vec can achieve slightly higher F1 score than the state-of-the-art supervised method that ensembles multiple BERT-based models.

C More Experimental Results

We describe our baselines for our ablation study in Appendix C.1, analyze the results of the ablation study in Appendix C.2, test different d values in our scaling function and reverse the asymmetry direction in Appendix C.3, compare the average length of extracted sentences in Appendix C.4, and report the Recall@K in Appendix C.5.

C.1 Ablation Study Setup

We start from **Asymm (d=4)**, which uses the best Word2Vec-based configuration reported in Table 1, and change one design choice or hyperparameter at a time. In addition, we test a few variants of **Asymm SciBERT**.

Kobayashi et al. (2015) theoretically show that a greedy optimization is effective for maximizing Equation 1. Hence, we also tried to greedily select the kth cited sentence S_c^k such that the selected sentence candidates up to this point $\cup_{i=1}^{k}\{S_c^i\}$ best cover the query citing sentence: $\arg\max_{S_c^k} \text{sim}(S_q, \cup_{i=1}^{k}\{S_c^i\})$. This baseline is called **Greedy Optimization**.

To confirm the effectiveness of our word weighting described in Section 2.2, we set the weights $W(w_q)$ in Equation 2 to be always 1 and call this baseline **Unif. Weights**. In addition to this, **Unif. Weights, No Stop Words** sets $W(w_q)$ as 0 if the word w_q is in our stop word list and as 1 otherwise.

Finally, to decrease the vocabulary size, we map the words to the [UNK] token if the word frequency is below a threshold. By default, the threshold is set to be 35, and we also try 50 and 100 in **Word2Vec Min. Word Count: 50 or 100**.

Method	Training Set		Test Set	
	F1	R-S*	F1	R-S*
Asymm (d=4)	16.4	11.2	**17.4**	**12.9**
Word Similarity: Dot Product	13.0	9.1	15.1	11.4
Greedy Optimization	13.8	10.0	15.3	12.0
Unif. Weights	9.5	7.1	8.9	6.9
Unif. Weights, No Stop Words	14.1	10.4	15.3	11.4
Keep Citation Markers	11.0	8.2	13.1	9.1
No Truncation	16.3	10.9	17.2	12.8
Truncate after 50 Tokens	15.9	10.9	**17.4**	12.7
Uncased	16.2	10.3	**17.4**	12.8
Word2Vec Min. Word Count: 100	15.7	11.1	16.7	12.5
Word2Vec Min. Word Count: 50	15.9	11.1	**17.4**	12.8
Asymm SciBERT (d=4)	**17.0**	12.2	15.3	11.4
Asymm SciBERT (d=1)	16.8	**12.5**	15.0	11.6
Asymm SciBERT (d=4), Unif. Weights	15.1	11.6	13.4	10.1

Table 3: Results of the ablation study. We report F1 (%) and ROUGE-S* (%) on training and test sets. See Table 1 for the configuration of **Asymm (d=4)**.

C.2 Ablation Study Results

Table 3 reports the results of our ablation study on both training and test sets. When using Word2Vec embeddings, we find that the following ablation baselines significantly degrade the performance measured by F1: (1) using a dot product instead of cosine similarity to compute word similarity (**Word Similarity: Dot Product**), (2) using greedy optimization, (3) removing the inverse frequency weighting of Arora et al. (2017) (**Unif. Weights**), and (4) omitting citation marker removal (**Keep Citation Markers**). Changing the truncation or casing configuration, or raising the minimum word count, only slightly decreases scores on the training set and results in little or no decrease in F1 on the test set.

We additionally find that the greedy sentence selection used in Kobayashi et al. (2015) is less effective than ranking sentences by their individual similarity scores when using our method for this task. We hypothesize that the effectiveness discrepancy comes from the length of the query. In the extractive summarization, the query is a long document, so we usually want the extracted next sentence to cover the aspects of query documents that are not covered by the previously extracted

sentences. In contrast, the query in cited text identification is much shorter, so the first citing sentences often can cover the important keywords of the query. As a result, the greedy method might extract the incorrect second cited sentence that does not mention these important keywords in the query citing sentences.

Furthermore, the ablation study shows that simply removing the stop words from a list (**Unif. Weights, No Stop Words**) is significantly worse than the inverse frequency weighting (**Asymm (d=4)**). This means that non-stop high-frequency words often carry less semantic information and thus, their matches should also be counted with smaller weights.

When using SciBERT embeddings, we also observe that removing inverse frequency weighting degrades performance, but the difference is smaller than the difference between **Asymm (d=4)** and **Unif. Weights**. This might highlight the difficulty of weighting the contextualized embeddings of individual words.

We note that the effect on F1 and ROUGE scores of setting $d = 1$ in the scaling function is mixed for **Asymm SciBERT**. A possible reason for this is that when using contextualized embeddings, an exact lexical match of two words does not yield a cosine similarity of 1, which makes a higher d also decrease the similarity scores between the exactly matched words from the sentence pair.

C.3 Varying the Power Hyperparameter in our Scaling Function and Reversing the Asymmetry Direction

Figure 3 plots the F1 score of **Asymm** on training and test sets against the value of the power hyperparameter d in our proposed method's scaling function. They plot the same quantity for the method that has the same configuration but reverses the standard direction of asymmetry such that the query aspects must cover the candidate aspects (**Asymm Reverse**). That is, we select the top 2 citation sentences with the highest $\text{sim}(S_c, S_q)$. **Symm**, **BERT ensemble**, and **BM25 ensemble** are also represented.

Reversing the direction of the asymmetry is an inherently challenging approach: the variability in candidate sentence length causes the system to prefer the longest candidates, as there are more terms in the summation over query words in Equation 1.

However, Figure 3 shows that, on the training set, the F1 score of **Asymm Reverse** becomes closer to that of **Asymm** as d is increased. Furthermore, on the test set, the F1 score of **Asymm Reverse** approaches that of **BM25 ensemble**, as does the score of **Asymm** after surpassing that of **BM25 ensemble** for more moderate values of d.

This observation is consistent with the intuition that as the power is increased, the mechanism of the asymmetric method approaches that of an exact word-matching method. The figure further suggests that an optimal value of d (on the training set, it is 4) might allow our method to strike a balance between a soft matching method that considers all query words and an exact matching method that considers only query words with a lexical match in the candidate, leading to improved performance over both these approaches.

Method	Selected Sentence Avg. Length
Symm	39.3
Asymm (d=1)	47.0
Asymm (d=4)	41.0
Asymm Reverse (d=1)	132.5
Asymm Reverse (d=4)	56.4

Table 4: Average sentence length of the top $K = 2$ sentences selected for all citing sentences in the training set.

C.4 Retrieved Sentence Lengths

Table 4 contains the average length of the top $K = 2$ sentences selected by each of the listed methods for the training set. An expected effect of our proposed method is to decrease the tendency of the basic asymmetric method with $d = 1$ to select longer sentences, noted in Section 3.1. From the table it is evident that adding the scaling function with $d = 4$ indeed leads to the selection of shorter sentences on average, reducing the average selected sentence length by 6 tokens to more closely approach the corresponding figure for our symmetric baseline, **Symm**.

The same effect is apparent when the standard direction of asymmetry in the similarity measure is reversed such that the query must cover the candidate (**Asymm Reverse**). In this case, we see that **Asymm Reverse (d=1)** generally selects very long candidate cited sentences, as expected due to the variability in candidate length, noted in Appendix C.3. However, increasing the power of the scaling function to $d = 4$ more than halves the average selected sentence length, likely by de-

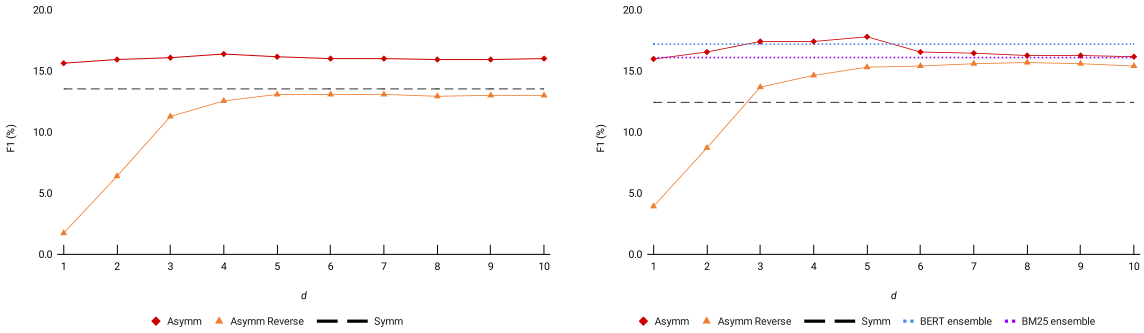

Figure 3: Training (Left) and test (Right) set F1 score of our proposed asymmetric method, **Asymm**, and the same method with the direction of asymmetry reversed (**Asymm Reverse**), as the hyperparameter d of the scaling function is varied. Scores for **Symm**, **BERT ensemble**, and **BM25 ensemble** are drawn from Table 2 for comparison where available.

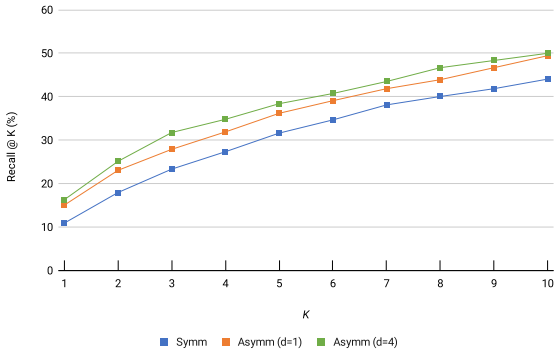

Figure 4: Recall on the test set as the number of selected sentences, K, is increased from 1 to 10.

emphasizing irrelevant words in relatively long candidates. Figure 3 show that this effect is accompanied by a drastic improvement in performance, although **Asymm Reverse** continues to be outperformed by **Asymm** for all choices of d in our experiments.

C.5 Recall@K

Figure 4 plots the test set recall performance of **Symm**, **Asymm (d=1)**, and our best-performing Word2Vec-based configuration, **Asymm (d=4)**. **Asymm (d=4)** consistently outperforms the two baselines as the number of selected sentences K is increased from 1 to 10. Within 10 predictions out of possibly 200 sentence candidates in a reference paper, the ability of our method to identify around 50% of all cited sentences selected by annotators indicates its practicality to a user who wishes to identify relevant sentences within the cited text.

	Training	Test
Num. Annotators per Citing Sentence	1	3
Num. Reference Papers	40	20
Avg. Num. Citing Sentences per Reference Paper	18.8	19.2
Num. (citing sentence,{gold cited sentences}) Pairs	753	1149

Table 5: CL-SciSumm corpus statistics.

D Experiment Details

The CL-SciSumm Shared Task was last held in 2020, and in that year the official task overview (Chandrasekaran et al., 2020) reported results for Task 1a from up to five runs from each of eight participants. Previous task offerings in 2018 (Jaidka et al. (2018); 10 participants) and 2019 (Chandrasekaran et al. (2019); 9 participants) evaluated submissions using the same blind test set, which is now public.

The dataset includes manual annotations for each citing sentence, each consisting of up to five spans from the reference paper that best reflect the citing sentence. The task statistics are reported in Table 5. We use the official evaluation script used in past editions of the Shared Task 1a to obtain our micro-averaged sentence overlap and ROUGE results.

E Method Details

Word2Vec training. The ACL Anthology Reference Corpus Version 2 (ACL ARC 2), used as our Word2Vec training corpus, contains 86M tokens. We train embeddings of dimension 300 using the Gensim library[3].

Word-frequency statistics. When our method is used with Word2Vec embeddings, the query word

[3] https://radimrehurek.com/gensim/

weights of Arora et al. (2017) are computed from word-count statistics collected from the training corpus. When our embeddings are contextualized embeddings from SciBERT, we similarly use the ACL ARC 2 corpus to compute word frequencies, but do so after WordPiece tokenization using the SciBERT tokenizer.

Stop words. We use the following lowercased stop word list: *@-@, =, <eos>, <unk>, disambiguation, etc, etc., –, @card@, ∼, -, _, @, ; &, *, <, >, (,), \, |, {, },], [, :, ;, ', '', /, ?, !, ,, ., 't, 'd, 'll, 's, 'm, 've, a, about, above, after, again, against, all, am, an, and, any, are, aren, as, at, be, because, been, before, being, below, between, both, but, by, can, cannot, could, couldn, did, didn, do, does, doesn, doing, don, down, during, each, few, for, from, further, had, hadn, has, hasn, have, haven, having, he, her, here, here, hers, herself, him, himself, his, how, how, i, if, in, into, is, isn, it, it, its, itself, let, me, more, most, mustn, my, myself, no, nor, not, of, off, on, once, only, or, other, ought, our, ours, ourselves, out, over, own, same, she, should, shouldn, so, some, such, than, that, the, their, theirs, them, themselves, then, there, these, they, this, those, through, to, too, under, until, up, very, was, wasn, we, were, weren, what, when, where, which, while, who, whom, why, with, won, would, wouldn, you, your, yours, yourself, yourselves*.

F Future Work

How to combine our approach with contextualized word embeddings more effectively is a promising research direction. For example, we can pretrain BERT on ACL papers as in Chai et al. (2020) after removing the citation markers. Furthermore, all of our experiments are done on CL-SciSumm Shared Task 1a, and we hope to also test our methods on other datasets such as SCIFACT (Wadden et al., 2020).

Recently, Gao et al. (2021) propose SimCSE, an effective unsupervised sentence similarity estimation method. Izacard et al. (2021) and Ram et al. (2022) propose unsupervised dense IR approaches. We are curious about the effectiveness of these approaches on partial sentence-matching tasks such as cited text identification. Furthermore, training Sentence-BERT (Reimers and Gurevych, 2019) on various kinds of similar sentences results in a general-purpose sentence similarity model (sbert.net, 2021). We leave the comparison with these approaches for future work.