BioCite: A Deep Learning-based Citation Linkage Framework for Biomedical Research Articles

Sudipta Singha Roy
University of Western Ontario
ssinghar@uwo.ca

Robert E. Mercer
University of Western Ontario
mercer@csd.uwo.ca

Abstract

Research papers reflect scientific advances. Citations are widely used in research publications to support the new findings and show their benefits, while also regulating the information flow to make the contents clearer for the audience. A citation in a research article refers to the information's source, but not the specific text span from that source article. In biomedical research articles, this task is challenging as the same chemical or biological component can be represented in multiple ways in different papers from various domains. This paper suggests a mechanism for linking citing sentences in a publication with cited sentences in referenced sources. The framework presented here pairs the citing sentence with all of the sentences in the reference text, and then tries to retrieve the semantically equivalent pairs. These semantically related sentences from the reference paper are chosen as the cited statements. This effort involves designing a citation linkage framework utilizing sequential and tree-structured siamese deep learning models. This paper also provides a method to create an automatically generated corpus for such a task.

1 Introduction

Research articles from different domains use varying writing styles and formats. They serve different purposes as well. A research publication may discuss current research trends, a novel discovery, or alternative approaches to solving a problem in a given domain. While writing a research article, the author mentions prior research that was either significant in resolving the same topic or impacted the author's views mentioned in the current research paper. This referencing another document in a research piece is referred to as a *citation* (Houngbo, 2017). This way, citations establish connections between distinct research literature as well as alleviating authors' writing burden by preventing them from having to write the same thing mentioned in another research article again. Simultaneously, it assists readers in acquiring prior knowledge about a subject that may be necessary to comprehend the ideas contained in the ongoing research work.

The idea of citation indexing was first introduced in 1964 where indexes contain the references in a research document. Citation-based bibliometrics are utilized to evaluate the significance of a research work (Garfield, 1972). In response to the growing popularity of citation indexing, a more critical analysis of citing was later suggested. Garzone and Mercer (Garzone and Mercer, 2000) devised a mechanism for determining the objective of a reference in biochemistry and physics research publications. Moreover, citations help to keep track of the logical argumentation across various research articles (Mercer, 2016). Prominent applications of citation incorporate maintaining the trail of scientific research argumentation across different research articles (Palau and Moens, 2009) and summarization of these documents (Radev et al., 2000).

In scientific research publications, a citation refers to the source article from which the cited notion is drawn. However, in experimental biomedical research articles, a citing sentence usually only relates to a small text span of the cited document's contents. This small span of text can be from the method section, result analysis section or any other section of the reference document (Singha Roy et al., 2020). The above-mentioned applications would substantially benefit if such a text span could be extracted from the original document. It would also free up the readers from having to read the full document to locate the cited piece of text.

The citation linkage task is more complicated for biomedical research papers as the same chemical or biological component has various representation formats and the use of these variations is very common in such research articles. For example, the chemical compound carbon dioxide can be represented as CO_2 as well as $O{=}C{=}O$, whereas in some

Proceedings of the BioNLP 2022 workshop, Dublin, Ireland, pages 241–251
May 26, 2022. ©2022 Association for Computational Linguistics

articles the writers write the whole name in plain text (*carbon dioxide*). Similarly, there are multiple representations to indicate the same reactions between various genes, chemicals, and drugs. On top of that, the only human annotated corpus available for the citation linkage task in the biomedical domain is from (Houngbo and Mercer, 2017) which comes with 3857 sentence pairs which are highly imbalanced with only 2% positive samples and 98% negative samples. The size and imbalanced nature of this corpus makes it difficult to train deep learning models on this dataset. To overcome this, we propose an automatically generated corpus for this task containing 74,568 sentence pairs.

This paper has two objectives: first, introducing an automatically generated corpus for the citation linkage task for biomedical research papers and second, providing a framework for this task to retrieve the cited text span from the reference paper given the citing sentence by means of measuring the semantic similarities between the citing sentence and candidate cited sentences from the referenced paper. The cited text span can be a single sentence, part of a sentence or even one or more paragraphs (Houngbo and Mercer, 2017). However, for this task this text span is restricted to a single sentence like Li et al. (2017). Considering the first objective, we introduce an automatically generated corpus containing 74,568 sentence pairs and also an approach to annotate data automatically without any human effort. The quality of the data annotation is evaluated by annotating a portion of the dataset by human experts and then measuring Cohen's κ among the human annotators' decisions and the automatically generated annotation labels. Sentence pairs from this dataset are used only for training the models for the citation linkage task. And for the second aspect, we have investigated multiple sequential and tree-structured neural networks and presented one ensemble architecture, which we call BioCite, that computes the semantic similarity between the citing statement and all of the sentences in the referred document. The performance of the model is tested against the expert annotated dataset from Houngbo and Mercer (2017) which contains citing sentences that refer to methods statements in the cited documents. The outline for the paper is: Section 2 gives a brief description of the citation linkage task and Section 3 mentions and discusses a few prominent works for the citation linkage task. Section 4 discusses the automatically generated corpus creation and the framework design. The performance of the models are reported and analyzed in Section 5. The parameters of the models are also described in this section. The paper ends with a brief summary and possible future directions of this research.

2 Citation Linkage

Citations construct semantic bridges between citing and cited manuscripts. To support the findings, claims and hypotheses, authors cite several resources while preparing manuscripts. They also try to address the results and findings of the other research works. It is also important to mention others' works, in order to demonstrate the authors' significance and progress with their current work.

A citation in any research paper focuses on some specific sections of the referenced article acknowledged as the *citation context*. This citation context often focuses on a specific idea or issue in the referenced manuscript (Houngbo, 2017). The intent of a using citation is to provide the readers with the apposite background information for a better understanding of the concepts introduced in the citing paper. The citation context can reveal information about a cited publication's hypotheses, findings, methodologies, etc. In order to improve the performance or make the method compatible with the domain for which it is intended to be used, an author may adapt or modify the method described in the citing paper to the extent necessary. Aside from that, the author may undertake experiments based on the idea presented in a cited paper to confirm or refute the idea presented in that work. References to the hypotheses and methodologies that were employed in the referenced paper aid the readers to grasp the concepts presented in the current work.

However, citations only provide the source of information which is being referred. The current citation indexing approach does not provide a way to indicate which text span from the cited research manuscript is actually being touched on. It provides no method other than going through the whole referenced article for the reader if he or she wants to grasp the idea properly. On the other hand, research articles that include detailed information on the study's discoveries, as well as relevant background information, are more appealing to readers. This necessity has influenced the work we are presenting in this paper.

The author can cite a paper by paraphrasing the statements from the cited paper. He or she can also elaborate some statements from the cited paper. For example in the citing statement, “DNA samples are frequently harmed by exposure to excessively acidic environment”, Wang et al. (2009) explains that "pH4" is an “excessively acidic environment” when citing “DNA is fairly stable in mildly acidic solutions, although the beta glycosidic link in the purine bases is hydrolyzed at around pH4.” (Bonin et al., 2003). Sometimes these citations are the interpretations of the cited statements, e.g., the citing sentence “Different PCR buffer systems and/or Taq polymerases may produce variable results in real time PCR.” (Huijsmans et al., 2010) is nothing but an interpretation of the cited sentence “There is a significant disparity between the outcomes obtained using the various DNA polymerase-buffer solutions.” (Wolffs et al., 2004). As these examples demonstrate, precise mapping between words and sentences is required to establish a connection between the citing and cited sentences.

This paper provides a citation linkage framework for biomedical research articles along with an automatically generated corpus comprising 74,568 sentence pairs. The framework at first generates sentence pairs with the citing sentence and all the sentences from the referenced paper. Then, the model measures the semantic similarity scores between the sentences in each pair. Based on these similarity scores, it retrieves the actual cited sentences from the referenced manuscript. We have formulated this semantic similarity measurement task as a binary classification task where each sentence pair is predicted with either label 1 or label 0. Sentence pairs predicted with label 1 are selected as the cited sentences given the particular citing sentence.

3 Related Work

The study of citations in scientific research has led to a lot of work. Citation analysis attempts to identify which section (i.e., abstract of the paper, introduction of the problem statement, description of methods, analysis of result, etc.) of the referenced article this sentence refers to (Garfield, 1972; Garzone and Mercer, 2000). However, this form of study cannot pinpoint the citation span.

Another type of work is to determine the citation span. PolyU (Cao et al., 2016) applied RankSVM over chunks of sentences to predict the cited text span. Baruah and Kolla (2018) computed cosine similarity of word embeddings for the citation linkage task. Yeh et al. (2019) applied majority voting to six machine learning classifiers over the lexical, knowledge-based, corpus-based, syntactic and surface features for this task.

The CL-SciSumm Shared Task tries to solve three aspects: find the cited text span given the citation sentence (“citance”), identifying the discourse facet of the cited sentence and summarise the referred article using only the text spans that are quoted many times in the referenced document. However, the later two sub-tasks are out of the scope of this work. Ma et al. (2017) applied different classifiers and voting mechanism over similarity, rule and position-based features to determine the similarity between the citing and cited statements for CL-SciSumm-17. The citation linkage between citing and cited sentence pairs was determined by Li et al. (2017) utilizing inverse document frequency and Jaccard similarity. In their following works, they computed the sentence vectors by concatenating 200 dimensional word vectors (Li et al., 2018) and then applying a convolutional neural network (CNN) over that concatenated vector representation (Li et al., 2019). In both cases, the cited text span is determined by measuring the cosine similarities between the citing and candidate cited statements. Other works, such as AbuRa’ed et al. (2017) have also worked with the CL-SciSumm corpus.

Recently, BERT-based models have been deployed for the citation linkage task and are being used in many experiments. Gidiotis et al. (2020) fine-tuned BERT to determine the referred cited sentences from the cited document. Zerva et al. (2019) applied a CNN over SciBERT-based features (Beltagy et al., 2019) to determine which text span in the cited article is actually being referred. They concatenated the features from the BERT-based model for feature generation. Umapathy et al. (2020) utilized key-phrase similarity using the Rapid Automated Keyword Extraction Algorithm (Rose et al., 2010) and a BERT-based architecture for cited text span identification.

However, only a few citation linkage works are found for biomedical research papers. Citation linkage for biomedical research articles is more challenging due to various representations of the same component. One notable work for this domain is from 2017, where Houngbo and Mercer (2017) used traditional machine learning approach over their own small expert-annotated corpus. And so far, this is the only human annotated corpus for the citation linkage task in the biomedical domain.

4 BioCite: Description of the Framework

The development of the framework involves two major steps: creating a balanced automatically generated training corpus of reasonable size and building a framework for determining the referred statements from the cited document for a particular citing statement.

4.1 Corpus Creation

The only expert-annotated corpus for the biomedical domain to serve the purpose of our work is from Houngbo and Mercer (2017) which comes with only 3857 sentence pairs. For training, the major problem with this dataset is the class imbalance: only 81 positive pairs which is only 2% of the corpus. Eventually, training any model with this corpus would make it biased towards negative outcome. At the same time, manually annotating enough data from biomedical and biochemical research articles for this task is time consuming. So, we have created an automatically generated corpus of 74,568 sentence pairs spanning three biomedical sub-domains: biochemistry, cell biology and chemical biology. We are calling this corpus automatically generated as no human annotation has been used for generating these sentence pairs. For the validation and testing of the models, we have used the validation and testing sets from the Houngbo and Mercer (2017) corpus (800 samples with 20 positive ones for validation and 3057 samples containing 61 positives for test set). The sentence pairs in the training set are annotated with 0 (not semantically similar) or 1 (semantically similar) to make it compatible with the validation and test set.

We collected 28,310 research documents from BioMed Central spanning multiple biomedical sub-domains. From these documents, 138 are randomly chosen from the above-mentioned three sub-domains and then corresponding citing statements from 2736 papers (manually collected) citing these 138 articles are extracted manually. The citing statements are then paired-up with all of the sentences from the corresponding cited documents, ending-up with 522,398 pairs.

Sentences of each pair are fed individually to the Sent2Vec (Pagliardini et al., 2018) model, which is trained over all of the research documents we accumulated, and the cosine similarity between the paired sentences is measured. Pairs with cosine similarity value greater than a cutoff value 0.57 (selected after testing against the validation set) are labelled 1, 0 otherwise. We experimented with different cut-off values and plotted the results on AUC and ROC curves while testing on the validation set from the expert annotated corpus (Houngbo and Mercer, 2017). From there, we chose the cut-off value for which the best validation accuracy was found. From there As there are many fewer positive samples than negative ones, for each citing statement, negative samples are randomly chosen for each citing sentence to balance the classes. In this automatically generated corpus, for each citing sentence, an equal number of positive and negative samples are preserved. The overall process of this corpus creation is illustrated in Figure 1.

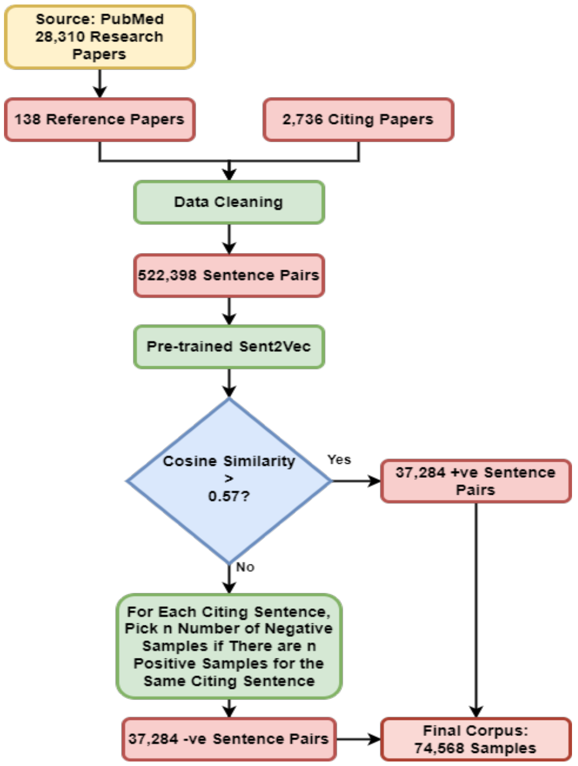
Source: PubMed 28,310 Research Papers
138 Reference Papers
2,736 Citing Papers
Data Cleaning
522,398 Sentence Pairs
Pre-trained Sent2Vec
Cosine Similarity > 0.57?
Yes
No
37,284 +ve Sentence Pairs
For Each Citing Sentence, Pick n Number of Negative Samples if There are n Positive Samples for the Same Citing Sentence
37,284 -ve Sentence Pairs
Final Corpus: 74,568 Samples

Figure 1: automatically generated corpus build-up: Sentence pair creation and annotation.

4.2 Semantic Similarity Measurement Module

The aim of building this citation linkage framework is to link the citing sentence to the referenced text span in the referenced biomedical research article. To solve this challenge, we have used a variety of supervised deep learning-based models to estimate the semantic similarity between the citing and cited text span where the text span is limited to a single sentence. The predictions of these models are set to binary class labels: 0 and 1. Here 1 indicates that the candidate cited and the particular citing statement are semantically similar and it can be interpreted as the candidate cited sentence is truly being referenced by the citing sentence and if the prediction value is 0, it represents the candidate cited sentence is not being referred.

The base of the sequential and tree-structured neural network models is InferSent (Conneau et al., 2017): a siamese architecture. This is a supervised sentence representation model which is able to work with sentence pairs and has been used in many cases for semantic relatedness measurement tasks (Ahmed et al., 2019; Reimers and Gurevych, 2019). The overview of the training process of InferSent for the semantic similarity measurement task is portrayed in Figure 2. In InferSent two identical encoder neural network topologies are used with identical parameter settings. The citing sentence (S_{citing}) and the cited sentence (S_{cited}) are encoded by them in parallel. This is followed by generating a feature map that concatenates concatenation, absolute point-wise difference, and point-wise multiplication. This feature map is then loaded into the dense and *softmax* layers in sequence to predict the binary class label. As the encoder models, four sequential and four tree-structured neural networks are used. The functioning principles of these models are first outlined, and then the ensembles of them are discussed. The best encoder model for the BioCite framework is chosen in the end based on the performances of the investigated models.

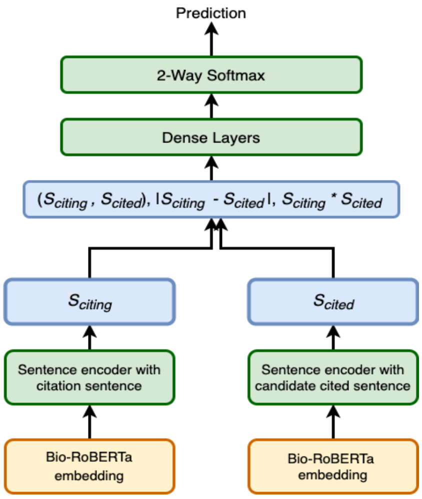
Prediction
2-Way Softmax
Dense Layers
$(S_{citing}, S_{cited}), |S_{citing} - S_{cited}|, S_{citing} * S_{cited}$
S_{citing}
S_{cited}
Sentence encoder with citation sentence
Sentence encoder with candidate cited sentence
Bio-RoBERTa embedding
Bio-RoBERTa embedding

Figure 2: InferSent training for the citation linkage task.

4.3 Sequential Encoders

As the encoder for the InferSent model, four sequential models are applied. The first one is the Bi-LSTM with a following max-pooling layer. The second encoder model applies inner attention (Liu et al., 2016) over the Bi-LSTM output features for producing the sentence representations. The third encoder model utilizes the hierarchical attention (Yang et al., 2016) in place of inner attention over the Bi-LSTM. This attention mechanism was introduced for document classification where at the first layer it attends on the words for generating sentence representation and in the second layer it attends over the sentences for paragraph or document representation. As our work is limited to single sentences, we have used only the first layer of this attention mechanism. This approach is designed in such a way that it can focus on four different parts of the sentence. Thus it generates four sentence representations, which are concatenated to form the sentence vector. The last sequential encoder we investigated is the hierarchical CNN with four layers of convolution operations, each followed by one max-pooling operation. These four feature maps are concatenated in the end to generate the sentence representation vector.

4.4 Tree-Structured Encoders

Sequential neural networks provide reasonable sentence representations. However, they can't preserve structural information and miss semantic compositionality. Tree-structured neural networks, on the other hand, can preserve both semantic and syntactic properties of the text by working with the parse tree. For the tree-structured neural network models we investigated the dependency and constituency tree-transformers with both multi-head and multi-branch attention mechanisms over child nodes' representations (Ahmed et al., 2019). For completeness, we provide details of these tree-transformers that are developed therein.

A constituency tree contains words at leaf nodes only, whereas a dependency tree has a word at each node. So, while traversing a dependency tree, it is required to consider both the child and corresponding parent nodes whereas for constituency tree, only after traversing every sub-tree the non-terminal intermediate nodes can be calculated. So, in both cases, the children nodes are considered. This approach (Ahmed et al., 2019) uses self attention mechanism for attending the child nodes. This attention mechanism uses three matrices: *key*, *value* and *query* like the transformer model (Vaswani et al., 2017) (Equ. 1).

$$\alpha = \texttt{softmax}(\frac{query\ key^T}{\sqrt{d_k}})value \tag{1}$$

Here d_k is the dimension of the *key*, *value* and *query* matrices. For this experiment the dimension of all these matrices are kept the same. n copies of these matrices are generated for n branches of the multi-branch attention mechanism. Here, n is the number of branches to be used. Then scaled dot product is used as in Equ. 2:

$$\beta_i = \alpha_{i\in[1,n]}(query_i\ \omega_i^q, key_i\ \omega_i^k, value_i\ \omega_i^v) \tag{2}$$

where ω_i^q, ω_i^k, ω_i^v are the hyper-parameter weight matrices for *query*, *key*, and *value*, respectively.

Following this scaled dot product operation, a residual connection is employed over these tensors β. A layer-wise batch normalization is used in the following step which is multiplied with a scaling factor τ (Equ. 3). Over every $\tilde{\beta}$, position-wise CNN (PCNN) is then employed (Equ. 4). By applying weighted summation then, the attention encoded semantic sub-spaces' representation are generated (Equ. 5). Here $\gamma \in \mathcal{R}^n$ is a hyper-parameter. In the end, another residual connection is established with `BranchAttn` which is then fed to a non-linearity function tanh and an element-wise summation function `EWS` is done to produce the parent node representation (Equ. 6) (Ahmed et al., 2019).

$$\tilde{\beta}_i = \texttt{LayerNorm}(\beta_i\omega_i^b + \beta_i) \times \tau_i \tag{3}$$

$$\texttt{PCNN}(x) = \texttt{Conv}(\textit{Relu}(\texttt{Conv}(x) + b_1)) + b_2 \tag{4}$$

$$\texttt{BranchAttn} = \sum_{i=1}^{n} \gamma_i \texttt{PCNN}(\tilde{\beta}_i) \tag{5}$$

$$\texttt{ParentNodeRep} = \texttt{EWS}(\tanh((\tilde{\chi} + \chi)\omega + b)) \tag{6}$$

For multi-head attentions, attention matrices *key*, *value* and *query* are projected h times (Vaswani et al., 2017) and it is calculated as follows:

$$\begin{aligned}&\texttt{MultiHead}(query, key, value)\\ &\quad = \texttt{Concat}(\texttt{head}_1, ..., \texttt{head}_h)W^O\end{aligned} \tag{7}$$

where, for each head,

$$\texttt{head}_i = \alpha(queryW_i^q, keyW_i^k, valueW_i^v) \tag{8}$$

All of the Ws are the hyper-parameter matrices which get updated during training.

4.5 Ensemble Architectures

After investigating the sequential and tree-structured neural network models, we experimented with two ensemble models. The first ensemble architecture utilizes all the models investigated here. After all the models are trained separately, each sentence pair is fed to all the models in parallel. Each model individually predicts the semantic similarity score and in the end, the final similarity value is selected by applying a winner-takes-all approach (Roy et al., 2018) over all the predictions. In the second approach we used only the tree-transformer models. The dependency tree-transformer is able to preserve the word level dependency between different part of the sentence, whereas the constituency tree-transformer can preserve phrase-level information. To benefit from both of these models, we concatenated the feature representations generated from both of the tree-transformers and used it as the vector representation of the sentence. This sentence vector is then fed to a multi-layer perceptron for the similarity score prediction.

5 Experimental Setup and Result Analysis

In this section, the experimental setup and the results of the models investigated for the citation linkage task are discussed. As the human annotated test data is highly imbalanced, apart from F-1 score, Matthews Correlation Coefficient (MCC) and Balanced accuracy (BAcc) are also used to assess the performance of the models.

5.1 Experimental Setup

Sent2Vec was trained with various parameter settings. The cutoff value and the best model are chosen based on the MCC and BAcc over the validation set. The best hyper-parameter settings for Sent2Vec are: 500d sentence embedding, window size 20, learning rate 0.2, negative sampling loss function and sampling threshold 0.0001. For the four sequential models: hierarchical CNN (hCNN), Bi-LSTM with max pooling, hierarchical and inner attentions over Bi-LSTM; the learning rates (LR) were initialized to 0.1. With a drop in validation accuracy, the LR is multiplied by 0.2. The batch size and LR threshold are set to 50 and 0.0001, respectively. For training, stochastic gradient descent is used as the optimizer. For hCNN, 4 layers of convolution are used followed by max-pooling.

Table 1: Statistics of the annotations by the experts and the automatically generated corpus for the 1500 samples

	Annotator Group 1	Annotator Group 2	The Automatically Generated Corpus
Positive samples (in total)	731	709	750
Negative Samples (in total)	769	791	750

Table 2: Analysis of the agreements among the expert annotators and the automatically generated corpus

	Between Annotator Groups 1 and 2	Between Annotator Group 1 and the Automatically Generated Corpus	Between Annotator Group 2 and the Automatically Generated Corpus
Agreed Positive Samples	706	715	701
Agreed Negative Samples	765	750	750
Cohen's κ	0.96	0.95	0.93

Four context vectors are used for both hierarchical and inner attention mechanisms to focus on 4 distinct parts which are concatenated for final sentence representations. For all of the tree-structured transformer models, 6 parallel heads are used with 50d query, value and key matrices where 6 position-wise convolution layers are used for multi-branch attention. Two layers of CNN (first layer: 341 1d kernel and no dropout, second layer: 300 1d kernels, 0.1 dropout) are used in the PCNN layer as the composition function which is the same as Ahmed et al. (2019). For parameter tuning, Adagrad (Duchi et al., 2011) with LR 0.0002 is used in all cases.

5.2 Performance Analysis

We first evaluate the quality of the automatically generated corpus. For analyzing the quality of the data annotation, we randomly picked 750 positive and 750 negative samples (labelled as such in the automatically generated corpus) from the 74,568 citing and candidate cited sentence pairs. These 1500 sentence pairs were provided to two groups of expert annotators. Each group consisted of three people and each person annotated 500 samples. So, each 500 sample chunk was annotated by two individuals, one from each group. Each reviewer also mentioned their confidence level for each sample annotation. We then used Cohen's κ (Cohen, 1960) to compute inter-annotator reliability between the human annotators and the automatically generated corpus. The overall statistics are shown in Table 1. The first group identified 731 positive and 769 negative samples in the 1500 sentence pairs, and the second group identified 709 positive and 791 negative samples. Table 2 shows the annotator groups' decisions agreed for 706 positive and 765 negative samples. The reliability factor κ found here is 0.96. While comparing the annotation provided by the automatically generated corpus against the first and second annotator groups, we see that the annotation decisions match for 715 and 701 positive samples between the automatically generated corpus and groups 1 and 2, respectively. For negative samples, the agreed decisions are 750 samples in both cases. The κ values are 0.95 (between first annotator group and the automatically generated corpus) and 0.93 (between second annotator group and the automatically generated corpus). These values indicate that the automatically generated corpus annotations match the experts' annotations quite well. When interpreting these high κ values, it is important to recall that the data given to the annotators were balanced (50/50 split of positive and negative samples). From Table 2 it is clear that the human annotators have high agreement for both of their positive and negative choices.

Next we provide the citation linkage task outcomes. To compare the performance of the model against the previous models, we evaluated the model with the gold standard human annotated data from Houngbo and Mercer (2017) because the previous models were tested against this gold standard corpus. This corpus focusses on citations of methods used in the citing and cited articles. Houngbo (2017) suggests that in most cases the citation refers to single sentences in the cited articles. As an example, the citing statement "Recently, Chauhan et al. employed SVM to predict the ATP binding residues in ATP binding proteins using amino acid

Table 3: Performance analysis of different architectures for the citation linkage task for biomedical research articles. Models tagged with † are the investigated ones in this work. Here, CT: constituency tree, DT: dependency tree, MB: multi-branch attention, MH: multi-head attention, TP: true-positive, FP: false-positive, TN: true-negative, FN: false-negative.

	Model	TP	FP	TN	FN	F1	MCC	BAcc (in %)
Previous Works	Houngbo	34	995	2001	27	0.06	0.07	61.27
	Li	39	779	2217	22	0.09	0.12	68.97
Sequential Models	Hierarchical CNN †	45	580	2416	16	0.13	0.19	77.21
	Bi-LSTM + Max-pooling †	54	361	2635	7	0.23	0.31	88.24
	Inner attentive Bi-LSTM †	55	372	2624	6	0.23	0.31	88.87
	Hierarchical Attentive Bi-LSTM †	56	355	2641	5	0.24	0.33	89.98
Tree Structured	DT-Transformer (MH) †	57	301	2695	4	0.27	0.36	91.70
	DT-Transformer (MB) †	58	287	2709	3	0.29	0.38	92.75
	CT-Transformer (MH) †	57	315	2681	4	0.26	0.35	91.46
	CT-Transformer (MB) †	57	309	2687	4	0.27	0.36	91.56
Ensemble	Winner-takes-all ensemble †	59	253	2743	2	0.32	0.41	94.14
	BioCite †	**60**	**240**	**2756**	**1**	**0.33**	**0.42**	**95.17**

sequences and their evolutionary profiles" (Firoz et al., 2011) indicates the cited sentence "Our SVM module predicts a score for each residue in protein (in range of -1.0 to 1.0), we define a threshold to discriminate ATP interacting and non-interacting residues" (Chauhan et al., 2009). Another approach for such a task could have been ranking the candidate sentences as was one of the methods done by Houngbo (2017). However, for the final classification step we used `softmax`, which gives a probability to every possible outcome, so this approach could easily be modified to be a ranking approach.

Table 3 reflects multiple performance metrics found for the models used here along with the results from a few prominent works. Among the sequential models, Bi-LSTM with the hierarchical attention mechanism fed with Bio-RoBERTa embeddings performs the best based on the MCC and BAcc (0.33 and 89.98% accordingly). However, it can correctly extract only 56 out of 61 positive samples. The inner attentive Bi-LSTM and simple Bi-LSTM followed by a max-pooling layer captures 54 and 55 positive samples correctly with the same MCC (0.31) and F1 score (0.23). However, the inner attentive Bi-LSTM model earns a slightly higher BAcc (88.87%) as it predicts more negative samples correctly.

The tree-structured models outperform all of the sequential models to extract the cited statements from the referenced documents. The reason for this is the constituency tree-transformer is able to capture phrase level information and the dependency tree-transformer is able to preserve word level dependencies. In biomedical articles, biological components' chemical names may comprise multiple words. The constituency tree-transformer has the capability to work better with such phrase level text. And in a lot of cases, the citing statements are complex in nature. The dependency tree-transformer deals with such cases well. Another important thing to notice here is that tree-transformers with multi-branch attention perform better than the tree-transformers with multi-head attention as multi-branch attention applies multiple heads in each branch and is thus able to obtain more information about each sentence (Fan et al., 2020). Here, both the constituency and dependency tree-transformers with multi-head attention mechanism predict 57 positive samples correctly. Multi-branch attentive dependency tree-transformer predicts 58 positive samples correctly. Constituency tree-transformer with multi-branch attention predicts 57 positive samples correctly. However, it predicts 6 more negative samples correctly attaining a 0.10 percentage point improvement in BAcc.

The two ensemble architectures investigated here improve the performance of the citation linkage task for biomedical research articles. The first approach ensembles all of the investigated individual models with the winner takes all selection process. This approach considers all the outcomes from different models and the outcome with the highest probability is chosen as the final prediction. It successfully predicts 59 positive samples out of 61 with 94.14% BAcc, 0.41 MCC and 0.32 F1 score which are higher compared to any of the standalone models. The second ensemble architecture considers only dependency and constituency tree-transformers with multi-branch attention. There are two reasons behind choosing only these two models for ensemble in this case: firstly, the major intention was to investigate how the model performs if we combine both the word dependency and phrase level information, and secondly, these two models showed better performance among all individual models. This ensemble architecture extracts 60 true positive cited statements given the citing statements for the citation linkage task. It also achieves 95.17% BAcc, 0.42 MCC and 0.33 F1 score. As the best performance is attained by this last ensemble architecture, for the BioCite citation linkage framework, we choose this approach for extracting cited statements from the referenced biomedical research article given the citing statement from the citing paper. Is the computationally more expensive ensemble model justified for predicting only a few more true-positives? We notice that the increase in true-positives is approximately 2%. This increase, especially in a larger corpus, would seem to justify the extra computational cost. However, it should also be noted that the false-positives have decreased by almost 20%. The applications noted in the introduction will benefit substantially by such a decrease in false-positives. This decrease in false-positives further justifies the extra computational cost of the ensemble model.

Now, there remains one more question to be discussed. Which one is actually improving the performance, the automatically generated corpus or the model? From Table 3, it is clear that, the performance of BioCite is better than the other models. To check the effectiveness of the proposed automatically generated corpus, we trained all the models over the human annotated small corpus (Houngbo and Mercer, 2017). In this experiment we found all the investigated models' accuracies were very high (around 98%). However, the BAcc, MCC and the F1 scores were very poor as the models are strongly biased towards the negative outcome. This gives evidence of the effectiveness of training models over our proposed automatically generated corpus. Furthermore, analyzing the outcomes and going through the predictions of the sentence pairs, we found that this model can successfully predict cited sentence given the citing statement when chemical components and reactions are presented in different ways. For example: the cited sentence "DNA is fairly stable in mildly acidic solutions, although the beta glycosidic link in the purine bases is hydrolyzed at around pH4." (Bonin et al., 2003) is predicted successfully for the citing sentence "DNA samples are frequently harmed by exposure to excessively acidic environment.", Wang et al. (2009). It indicates that this model has the ability to resolve "pH4" as an "excessively acidic environment" and "hydrolyzed" with "harmed".

6 Conclusion

Biomedical literature is complex in nature due to having complex biological and chemical component names. Our framework, BioCite, performs well when dealing with the human annotated test set containing research articles accumulated from the biomedical domain and outperforms the previous prominent works. However, there are still a few avenues to investigate. The text span used here is a single sentence. In future, it can be expanded to the paragraph level which would capture the contextual information as well. Graph-based neural networks which perform well when working with paragraphs (Zhang et al., 2020) could be used. Moreover, BERT-based models can be explored as well.

References

Ahmed AbuRa'ed, Luis Chiruzzo, and Horacio Saggion. 2017. What sentence are you referring to and why? Identifying cited sentences in scientific literature. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 9–17.

Mahtab Ahmed, Muhammad Rifayat Samee, and Robert E Mercer. 2019. You only need attention to traverse trees. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 316–322.

Gaurav Baruah and Maheedhar Kolla. 2018. Klick labs at cl-scisumm 2018. In *BIRNDL@ SIGIR*.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*.

Serena Bonin, F Petrera, B Niccolini, and Giorgio Stanta. 2003. PCR analysis in archival postmortem tissues. *Molecular Pathology*, 56(3):184–186.

Ziqiang Cao, Wenjie Li, and Dapeng Wu. 2016. Polyu at cl-scisumm 2016. In *Proceedings of the joint workshop on bibliometric-enhanced information retrieval and natural language processing for digital libraries (BIRNDL)*, pages 132–138.

Jagat S Chauhan, Nitish K Mishra, and Gajendra PS Raghava. 2009. Identification of atp binding residues of a protein from its primary sequence. *BMC bioinformatics*, 10(1):1–9.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. *arXiv preprint arXiv:1705.02364*.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7).

Yang Fan, Shufang Xie, Yingce Xia, Lijun Wu, Tao Qin, Xiang-Yang Li, and Tie-Yan Liu. 2020. Multi-branch attentive transformer. *arXiv preprint arXiv:2006.10270*.

Ahmad Firoz, Adeel Malik, Karl H Joplin, Zulfiqar Ahmad, Vivekanand Jha, and Shandar Ahmad. 2011. Residue propensities, discrimination and binding site prediction of adenine and guanine phosphates. *BMC biochemistry*, 12(1):1–12.

Eugene Garfield. 1972. Citation analysis as a tool in journal evaluation. *Science*, 178(4060):471–479.

Mark Garzone and Robert E Mercer. 2000. Towards an automated citation classifier. In *Conference of the Canadian Society for Computational Studies of Intelligence*, pages 337–346. Springer.

Alexios Gidiotis, Stefanos Stefanidis, and Grigorios Tsoumakas. 2020. Auth@ clscisumm 20, laysumm 20, longsumm 20. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 251–260.

Hospice Houngbo and Robert E Mercer. 2017. Investigating citation linkage with machine learning. In *Canadian Conference on Artificial Intelligence*, pages 78–83. Springer.

Kokou Hospice Houngbo. 2017. *Investigating Citation Linkage Between Research Articles*. Ph.D. thesis, The University of Western Ontario.

Cornelis JJ Huijsmans, Jan Damen, Johannes C van der Linden, Paul HM Savelkoul, and Mirjam HA Hermans. 2010. Comparative analysis of four methods to extract DNA from paraffin-embedded tissues: Effect on downstream molecular applications. *BMC Research Notes*, 3(1):239.

Lei Li, Liyuan Mao, Yazhao Zhang, Junqi Chi, Taiwen Huang, Xiaoyue Cong, and Heng Peng. 2018. Computational linguistics literature and citations oriented citation linkage, classification and summarization. *International Journal on Digital Libraries*, 19(2-3):173–190.

Lei Li, Yazhao Zhang, Liyuan Mao, Junqi Chi, Moye Chen, and Zuying Huang. 2017. CIST@CLSciSumm-17: Multiple features based citation linkage, classification and summarization. In *BIRNDL@ SIGIR (2)*, pages 43–54.

Lei Li, Yingqi Zhu, Yang Xie, Zuying Huang, Wei Liu, Xingyuan Li, and Yinan Liu. 2019. CIST@CLSciSumm-19: Automatic scientific paper summarization with citances and facets. *BIRNDL2019*.

Yang Liu, Chengjie Sun, Lei Lin, and Xiaolong Wang. 2016. Learning natural language inference using bidirectional LSTM model and inner-attention. *arXiv preprint arXiv:1605.09090*.

Shutian Ma, Jin Xu, Jie Wang, and Chengzhi Zhang. 2017. Njust @ clscisumm-17 shutian. In *Proceedings of the First Workshop on Scholarly Document Processing*.

Robert Mercer. 2016. Locating and extracting key components of argumentation from scholarly scientific writing. *Dagstuhl Reports*, 6(4):3–15.

Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018. Unsupervised learning of sentence embeddings using compositional n-gram features. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 528–540.

Raquel Mochales Palau and Marie-Francine Moens. 2009. Argumentation mining: The detection, classification and structure of arguments in text. In *Proceedings of the 12th International Conference on Artificial Intelligence and Law*, pages 98–107. ACM.

Dragomir R. Radev, Hongyan Jing, and Malgorzata Budzikowska. 2000. Centroid-based summarization of multiple documents: Sentence extraction, utility-based evaluation, and user studies. In *NAACL-ANLP 2000 Workshop: Automatic Summarization*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Stuart Rose, Dave Engel, Nick Cramer, and Wendy Cowley. 2010. Automatic keyword extraction from individual documents. *Text mining: applications and theory*, 1:1–20.

Sudipta Singha Roy, Sk Imran Hossain, MAH Akhand, and Kazuyuki Murase. 2018. A robust system for noisy image classification combining denoising autoencoder and convolutional neural network. *International Journal of Advanced Computer Science and Applications*, 9(1):224–235.

Sudipta Singha Roy, Robert E Mercer, and Felipe Urra. 2020. Investigating citation linkage as a sentence similaritymeasurement task using deep learning. In *33th Canadian Conference on Artificial Intelligence*.

Anjana Umapathy, Karthik Radhakrishnan, Kinjal Jain, and Rahul Singh. 2020. Citeqa@ clscisumm 2020. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 297–302.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Yuker Wang, Victoria EH Carlton, George Karlin-Neumann, Ronald Sapolsky, Li Zhang, Martin Moorhead, Zhigang C Wang, Andrea L Richardson, Robert Warren, Axel Walther, et al. 2009. High quality copy number and genotype data from FFPE samples using molecular inversion probe (MIP) microarrays. *BMC Medical Genomics*, 2(1):8.

Petra Wolffs, Halfdan Grage, Oskar Hagberg, and Peter Rådström. 2004. Impact of DNA polymerases and their buffer systems on quantitative real-time PCR. *Journal of Clinical Microbiology*, 42(1):408–411.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489.

Jen-Yuan Yeh, Tien-Yu Hsu, Cheng-Jung Tsai, Pei-Cheng Cheng, and Jung-Yi Lin. 2019. On identifying cited texts for citances and classifying their discourse facets by classification techniques. *Journal of Information Science & Engineering*, 35(1).

Chrysoula Zerva, Minh-Quoc Nghiem, Nhung TH Nguyen, and Sophia Ananiadou. 2019. Nactemuom@ cl-scisumm 2019. In *BIRNDL@ SIGIR*, pages 167–180.

Yufeng Zhang, Xueli Yu, Zeyu Cui, Shu Wu, Zhongzhen Wen, and Liang Wang. 2020. Every document owns its structure: Inductive text classification via graph neural networks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 334–339.