

A Context-based Framework for Modeling the Role and Function of On-line Resource Citations in Scientific Literature

He Zhao¹, Zhunchen Luo², Chong Feng^{1†}, Anqing Zheng¹, Xiaopeng Liu²

¹Beijing Institute of Technology, School of Computer Science, Beijing, China

²Information Research Center of Military Science, PLA Academy of Military Science, Beijing, China

zhaohe1995@outlook.com zhunchenluo@gmail.com

fengchong@bit.edu.cn aqzheng1995@outlook.com liuxiaopeng1975@163.com

Abstract

We introduce a new task of *modeling the role and function for on-line resource citations in scientific literature*. By categorizing the on-line resources and analyzing the purpose of resource citations in scientific texts, it can greatly help resource search and recommendation systems to better understand and manage the scientific resources. For this novel task, we are the first to create an annotation scheme, which models the different granularity of information from a hierarchical perspective. And we construct a dataset SciRes, which includes 3,088 manually annotated resource contexts. In this paper, we propose a possible solution by using a multi-task framework to build the scientific resource classifier (SciResCLF) for jointly recognizing the role and function types. Then we use the classification results to help a scientific resource recommendation (SciResREC) task. Experiments show that our model achieves the best results on both the classification task and the recommendation task. The SciRes dataset¹ will be released for future research.

1 Introduction

In this paper, we introduce a new task of *modeling the role and function for on-line resource citations in scientific literature*. With the number of scientific publications growing dramatically, numerous on-line resources are mentioned, released and used within the scientific literature. Tracing and modeling these resources such as software, tools and datasets can greatly help researchers by developing scientific resource searching and recommendation systems or constructing scientific resource knowledge graphs. Google has launched a new search engine² in 2018 to help scientists find the

[†]Corresponding author.

¹<https://github.com/zhaohe1995/SciRes>

²<https://toolbox.google.com/datasetsearch>

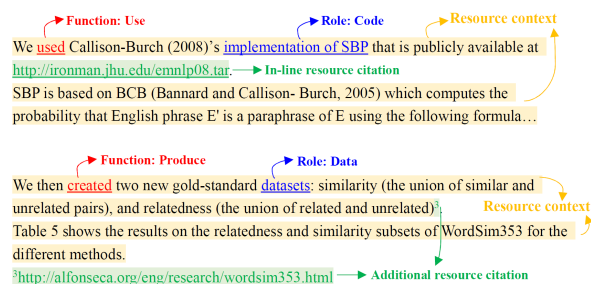


Figure 1: Examples for the two types of resource citations in scientific literature. We note the arguments (which are mostly the key verbs before the citation) for identifying the resource function and the arguments (which are mostly the target nominals before the citation) for identifying the resource role.

datasets they need, whereas the retrieved datasets can only be matched with their official names. Other current academic search engines such as Google Scholar can be only used for detecting the relevant papers where a certain resource is mentioned. However, the limitation actually exists is that the user can not learn more fine-grained information such as what role the resource plays in its paper and why the resource is cited in its contexts. To improve present works, we propose a context-based framework to model both the *resource role* and the *resource function* for on-line resource citations in scientific literature.

As is shown in Figure 1, we give examples for the two types of resource citations considering their appearances in the original publications. Through observing more than three hundred scientific publications from different domains of computer science, we find that most resource citations can be divided into two types according to the locations of their hyperlinks: the **in-line resource citations** in bodytexts and the **additional resource citations** in footnotes. We define a **resource citation** as a hyperlink mentioned in the

scientific paper text, which links to a specific on-line resource. A **resource context** is a word sequence surrounding the resource citation. As Figure 1 shows, we set the context window size to 5 in our work. So that the context includes the sentence where the hyperlink is mentioned, the two sentences to the left and the two to the right. The **resource role** is the class of a resource indicating what role the resource plays in its context (e.g. *Code* and *Data* in Figure 1). The **resource function** is the specific purpose performed by the resource with respect to the current paper’s work, indicating why the author has cited this resource here (e.g. *Use* and *Produce* in Figure 1). It’s necessary to note that one resource may have different functions in a same paper due to the different contexts.

The information of resource role and function is very important for building applications to assist scientific research. To help develop more powerful resource search systems, identifying the role can enrich the resource repository and identifying the function is crucial for understanding more complicated queries and providing more accurate results. Moreover, to help researchers quickly be acquainted with the scientific resources and easily find applicable resources for their work, the scientific resource recommendation system will have great application prospects in the future. Previous works mostly make the efforts on the task of *citation function classification* (Teufel et al., 2006; Jurgens et al., 2018) and the task of *context-based citation recommendation* (Tang and Zhang, 2009; He et al., 2011; Huang et al., 2015). Different from previous works, our work specially focuses on the on-line resources in scientific text, which are not as well studied as the paper citations. To the best of our knowledge, we are the first to model the on-line resource citations on such a fine-grained level in scientific full text.

In this paper, we first propose a new annotation scheme, which models both the general role types and the fine-grained role types from a two-hierarchy perspective, and models the function types by analyzing the purpose of citing the resources. Based on the scheme, we construct a dataset SciRes by manually annotating more than three thousand resource contexts. To better classify each on-line resource citation and benefit from the associated information between roles and functions, we apply a multi-task framework

SciResCLF, which jointly identifies the resource role types and the resource function types based on the word sequences of resource contexts. Experiments show that our model outperforms all the baselines on the classification task. We further address a context-based resource recommendation task and develop a framework SciResREC, which predicts the resource hyperlinks only given the masked resource contexts. Using the classification results, our model achieves good performance by the help of the role and function information in the recommendation task.

In summary, we make the following contributions. We introduce a novel task of modeling fine-grained information, especially the resource role and the resource function of scientific on-line resource citations. For this task, we propose a new annotation scheme and create a dataset for resource classification. We develop a multi-task learning model which can jointly identify the general role types, fine-grained role types and functions for the resource citations. Based on our united classifier we solve a resource recommendation task and analyze the evolution and maturity of the scientific resources.

2 Related Work

Context-based Citation Analysis

Context-based citation analysis addresses a citation’s value by interpreting each citation based on its context at both the syntactic and semantic levels. One research focus is the *citation function*, which is defined as the authors reason for citing a given paper. Specific schemes for classifying the function of a citation have been developed (Teufel et al., 2006; Liakata et al., 2012; Jurgens et al., 2018). Jurgens et al. (2018) did their efforts to measure the evolution of a scientific field by framing the citation function. In the past years, many annotation schemes and approaches for identifying arguments in various domains have been developed. Following these previous schemes, we develop our annotation scheme especially for modeling the role and function for scientific resource, which will be detailed in Section 3. Other previous studies focus on the *context-based citation recommendation* (Tang and Zhang, 2009; He et al., 2010, 2011; Huang et al., 2012, 2015), which aims to recommend a short list of papers that need to be cited within the given context. A citation context is defined as a sequence

	#Bodytext	#Footnote	Total	Annotation
ARC	3,718	15,043	18,761	1,026
NeurIPS	686	896	1,582	1,056
PUBMED	31,852	510	32,362	1,006
Total	36,256	16,449	52,705	3,088

Table 1: A description for the collection of 52,705 data sample in the scientific resource corpus.

of words that appear around a particular citation. Based on this definition, we develop the concepts of *resource citation* and *resource context* in this paper.

Resource Discovery for Scientific Text

There are also some relevant research which focus on detecting resources in biomedical literature. Some approaches to this end are reported in (Duck et al., 2012, 2014, 2016; Yamamoto and Takagi, 2007; de la Calle et al., 2009). Yamamoto and Takagi (2007) crawled abstracts from MEDLINE and full text from BioMed to extract the URLs of resources using regular expressions and heuristic rules. The BIRI addressed by de la Calle et al. (2009) utilized handcrafted regular expressions expressed as transition networks for resource naming, functionality detection and resource classification to generate resource inventories. Duck et al. (2012) proposed the BioNERDS, which is a NER system for detecting database and software names in scientific literature using extraction rules generated by examining 30 bioinformatics articles. However, there has been no existing framework for modeling the resource role and function at such a fine-grained level in general domain scientific text.

3 Dataset

There are some current scientific literature datasets (e.g. Semantic Scholar Open Research Corpus³ and Arxiv Dataset⁴) contain the structured metadata such as title, author and abstract. While as is illustrated in Figure 1, most on-line resource citations are placed in the full texts of the papers. We need to locate the resource citations and extract the resource contexts first to perform further analysis. To the best of our knowledge, due to the difficulties in collecting large scale of scientific full texts from the PDF publications, there is no ready-to-use dataset for our task. Hence we

³<http://api.semanticscholar.org/corpus/>

⁴<https://www.kaggle.com/neelshah18/arxivdataset>

collect a large scientific resource context corpus of 52,705 data samples and construct a manually annotated dataset (called **SciRes**) which includes annotations for the resource role types and the resource function types. SciRes attempts to interpret the role of resource citations from a hierarchical perspective, with both the general role types and more fine-grained role types. And SciRes aims at understanding the author’s intention of citing a resource by the function types. We hope the SciRes can facilitate future research for context-based resource analyzing in scientific literature.

3.1 Data Processing

We used the paper full texts from three different sources: the ACL Anthology Reference Corpus (ARC)⁵, a corpus of scientific publications about computational linguistics; the NeurIPS Proceedings (NeurIPS)⁶, a corpus of conference proceedings about neural information; and the PubMed⁷, an archive of biomedical and life sciences journal literature. We collected 21,411 papers of ARC up to 2015, 7,147 papers of NeurIPS from 1988 to 2017, and randomly downloaded 11,043 publications from PubMed. For each paper in the corpus, we downloaded the PDF and used Omnipage⁸ to perform OCR in translating the files to characters with coordinates and properties. So that we can detect the footnotes and find their anchors in the bodytext by developing some regular expression filters. Then we applied a conditional random field-based parsing tool, ParsCit⁹ to get the structural raw text. To build the SciRes dataset we first extracted all the hyperlinks as resource citations in a scientific paper from both the bodytexts and the footnotes. Three PhD students from the above three fields were asked to read 200 randomly selected resource contexts to investigate the context window size. It was found that 95% samples can be determined the resource role and function types by getting across the 5 sentences around the citations. So we set the window size to 5 and extracted the sentences along with each hyperlink as the resource context. Finally, we construct a collection of 52,705 data samples, with a detailed description shown in Table 1.

⁵<http://acl-arc.comp.nus.edu.sg/>

⁶<http://papers.nips.cc/>

⁷<https://www.ncbi.nlm.nih.gov/pubmed>

⁸<https://www.nuance.com>

⁹<https://github.com/knmnyn/ParsCit>

1st-category	2nd-category	#ARC	#NeurIPS	#PUBMED	1st-Total	1st-%	2nd-Total	2nd-%
Material	Data	333	452	219	1,004	31.0	1,004	31.0
	Tool	340	106	171	617	19.0		
Method	Code	74	256	91	421	13.0	1,057	38.7
	Algorithm	79	118	19	216	6.7		
Supplement	Document	81	97	132	310	9.6		
	Website	146	66	144	256	7.9		
	Paper	3	3	242	248	7.6	984	30.3
	License	16	1	97	114	3.5		
	Media	4	49	3	56	1.7		

Table 2: The 3 1-st category role types and 9 2-nd category role types along with their statistics in **SciRes** from three scientific literature sources.

3.2 Annotation Scheme

Many annotation schemes for the citation functions have been created over the past years (Teufel et al., 2006; Liakata et al., 2012; Jurgens et al., 2018). Based on these previous works and some recent annotation schemes for ScienceIE (Luan et al., 2017; Augenstein et al., 2017; Zhao et al., 2019), we address our scheme:

3 general (1st-category) Role types: Material, Method, Supplement.

9 fine-grained (2nd-category) Role types: Data, Tool, Code, Algorithm, Document, Website, Paper, License, Media.

6 Function types: Use, Produce, Introduce, Extend, Compare, Other.

The hierarchical relationships between 1st-category role types and 2nd-category role types are shown in Table 2. More detailed definitions and examples for each type can be found in the Appendix. Annotations were performed by a group of 3 PhD students, of which one majors in NLP, one majors in deep learning and another majors in bioinformatics. We randomly selected 1,100 data samples from each scientific literature source. Since too short text might not cover sufficient information for identifying the target role and function types, we filtered out the samples of which the context sequence had less than 10 words. Each resource citation together with its context was assigned at least one label for the general role types, one label for the fine-grained role types and a unique label for the function types. Fleiss’s Kappa (κ) is 0.79 for the 1st-category resource role, 0.54 for the 2nd-category resource role and 0.65 for the resource function, indicating a relatively high agreement between annotators considering the number of class and the difficulties of the fine-grained classification task. The decision is made by a majority rule when a conflict occurs. Finally we obtained the manually an-

	#ARC	#NeurIPS	#PUBMED	Total	%
Use	560	578	347	1,485	48.1
Produce	125	308	201	634	20.5
Introduce	254	89	281	624	20.2
Extend	53	30	39	122	4.0
Other	17	1	106	124	4.0
Compare	18	50	30	98	3.2

Table 3: The 6 resource function types along with their statistics in **SciRes** from three scientific literature sources.

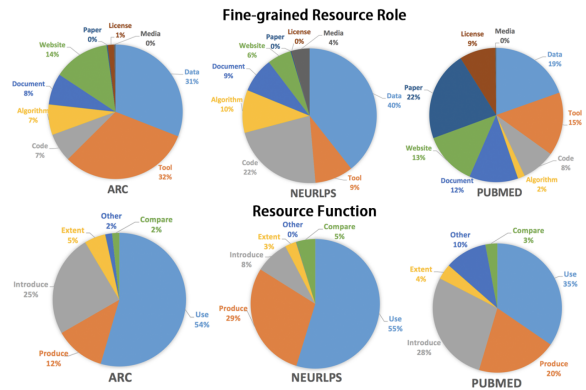


Figure 2: The distributions of resource role and function types in different data sources.

notated **SciRes** dataset of 3,088 data samples. All the function and role types along with their statistics are shown in Table 2 and Table 3.

For the more fine-grained role types, the distribution is very skewed. The majority of 2-nd category roles are Data (31.0%) and then come the Tool (19.0%) and the Code (13.0%). The majority of functions are Use (48.1%) while the Compare (3.2%) and Extend (4.0%) are much fewer. The existence of a large imbalance between the types bring challenges for our classification task. There are also some findings when comparing the distributions of roles and functions between different data sources, as shown in Figure 2. For the fine-grained resource roles, the ARC dataset has relatively more Tools reflecting that there are more re-

search about NLP applications in the ARC. While the more theoretical NeurIPS has least Tools but most Algorithm and Data citations. Instead of the packaged software, the papers from NeurIPS prefer the implementations in codebases. Furthermore, due to the difference of article formats and writing styles in different domains, the papers in the field of bioinformatics from the PubMed tend to link papers by in-line hyperlinks in bodytexts while the papers from the ARC and the NeurIPS tend to cite papers in reference lists. For the resource functions, we can see that the papers from the NeurIPS tend to produce or release more new resources while the papers from the ARC and the PubMed tend to introduce more resources as background to support their works. We assume the difference in the resource function is because that the research from the NeurIPS are more theory-based and often put forward new methodologies, whereas the research from the ARC and the PubMed are more comprehensive and contain more application-oriented works which tend to review many related resources.

4 Classification Model

To identify the resource role types and the resource function types for the resource citations, we apply a multi-task learning framework, called **SciResCLF** to jointly classify the 1-st category roles, the 2-nd category roles and the functions by sharing the resource context representations as classification features. Figure 3 shows an overall architecture of our SciResCLF framework.

Based on our SciRes dataset, many challenges make the classification task not easy. First, it is important to parse, encode and model the semantic information in the resource citation contexts, which are relatively short texts having no more than 5 sentences. Second, as the examples shown in Figure 1, by observing the data samples we find in most cases some key nominals or verbs located nearby the citations can imply the role and function types of the resources (e.g. the nearest verb before a resource citation such as "use", "apply" or "adopt" often indicates the function of **Use**). For this reason, the relative positions of the words in the sequence is very significant information to be considered in our task. Furthermore, due to the limited labeled data most neural-based models with a large number of parameters can not be trained well. Therefore, effective methods for solv-

ing these particular challenges need to be developed for this new scientific resource classification task.

From the SciRes dataset we can find that there is a strong correlation between the role type and the function type for a resource citation in its context. To better incorporate the associated information, for our classification task we adopt a multi-task setup, which has been proven effective in the ScienceIE problems (Augenstein et al., 2017; Luan et al., 2018). Recently, the pre-trained language models, such as ELMo (Peters et al., 2018), OpenAI GPT (Radford, 2018), and BERT (Devlin et al., 2018), have shown their effectiveness to alleviate the effort of feature engineering. Especially, BERT has achieved excellent results in text classification problems such as sentiment analysis (Sun et al., 2019; Xu et al., 2019) and document classification (Adhikari et al., 2019). Based on previous works, our framework also takes advantage of the pre-trained BERT model to learn contextualized representations for the resource contexts, as shown in Figure 3. To adapt BERT to our classification tasks, following Devlin et al. (2018) we first get the hidden state corresponding to the $[CLS]$ input token. Then the hidden representation h is passed into three separate softmax layers: $\mathbf{L}_{1st.role} = softmax(W_1h + b_1)$, $\mathbf{L}_{2nd.role} = softmax(W_2h + b_2)$, $\mathbf{L}_{func} = softmax(W_3h + b_3)$, where $W_1, W_2, W_3 \in \mathbb{R}^{r_h}$, $b_1, b_2, b_3 \in \mathbb{R}$ and r_h is the size of the hidden dimension. So that given the word sequence input of a resource context, the total loss \mathbf{L}_{CLF} is defined as a weighted sum of the cross entropy loss of the three multinomial classification tasks:

$$\mathbf{L}_{CLF} = \alpha \mathbf{L}_{1st.role} + \beta \mathbf{L}_{2nd.role} + \gamma \mathbf{L}_{func} \quad (1)$$

5 Recommendation Model

To show a practical application scenario of modeling the role and function for scientific resource citations, we further address a new resource recommendation task. The problem takes the scientific text sentences about an indexed resource as a query and aims to predict the hyperlink of a possible on-line resource to fill the citation blank of a resource context. We first give the formalized definition for the resource recommendation task. The input is the word sequence of a resource context $C = \{w_{n-l}, \dots, w_{n-1}, w_n, w_{n+1}, \dots, w_{n+r}\}$, in which w_n is a placeholder "[URL]" of the resource hyperlink, l is the length of the sequence

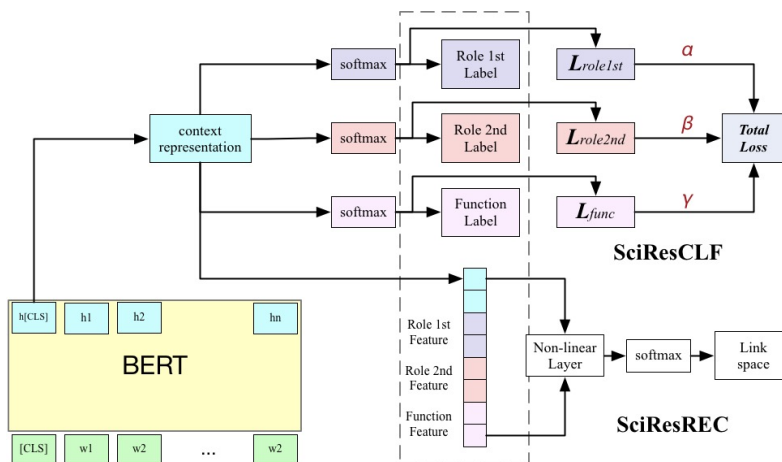


Figure 3: The overall architecture of the SciResCLF and the SciResREC.

left to the hyperlink and r is the length of the sequence right to the hyperlink. To eliminate the information of the original resource citation, we mask the resource names mentioned in the resource context. So that the resource hyperlink can only be determined by the other words related to the target resource from the context query. The output is a ranking list, which contains the top N possible predictions from the resource hyperlink space.

For this task, we develop a framework, called **SciResREC** to predict the resource hyperlinks by learning information of the resource contexts and benefiting from the classification results of the SciResCLF. Figure 3 also shows a high-level overview of the SciResREC framework. For each input resource context sequence, we first apply the SciResCLF to respectively get the output results of the 1-st category role classifier, the 2-nd category role classifier and the function classifier. To incorporate the role and function information, the three classification labels are used as features for predicting the resource hyperlinks. In SciResREC, we use the same pre-trained BERT encoder to learn context representations. Then the hidden output $h_{[CLS]}$ is concatenated with the three label representations, which is passed to a non-linear layer with the ReLU activation function. Hence the recommendation task is transformed into a multi-class classification problem which maps the context feature vectors into the resource hyperlink space.

6 Experiments

6.1 Data and Metrics

To test the performance of SciResCLF, we do experiments on our SciRes dataset, which is split into 3 parts: 80% for training, 10% for testing and 10% for developing. To overcome the class imbalance, we utilize an up-sampling strategy by simply replicating the minority class samples. We respectively deploy the up-sampling strategy for three classification tasks to get the best model for each task. The strategy is only deployed on the training set while not on the testing and developing sets. Finally we get 2,988 data samples for training the 1st-category role classifier, 7,236 for the 2nd-category role classifier and 7,404 for the function classifier. The testing and developing sets are shared by the three tasks, both with 334 samples. The size of the dataset is relatively small to perform most neural-based methods, but we will show that our model can achieve good performance with even limited labeled data. For evaluation, we report the micro-F1 score and the macro-F1 score across the role and function types.

To build the training set for SciResREC, we first collect all the resource contexts and the citation hyperlinks from the ARC dataset (including articles up to December 2015). We select the top 100 most frequent resources and add their hyperlinks into the hyperlink space. For testing, we extract resource contexts from the publications of ACL2016, ACL2017, EMNLP2016 and EMNLP2017 and select the ones of which the hyperlink exists in the space. For each context, either the in-line resource citation or the additional resource citation is replaced by a “[URL]”

Methods	1st-category Roles		2nd-category Roles		Functions	
	F1-micro	F1-macro	F1-micro	F1-macro	F1-micro	F1-macro
AvgEmbed+LR	0.733	0.732	0.453	0.462	0.534	0.475
AvgEmbed+SVM	0.691	0.690	0.429	0.447	0.550	0.466
FastText	0.694	0.692	0.489	0.472	0.638	0.471
CNN	0.649	0.647	0.450	0.313	0.618	0.498
RCNN	0.697	0.697	0.430	0.407	0.621	0.517
LSTM	0.607	0.605	0.413	0.393	0.508	0.399
LSTM+AT	0.706	0.705	0.502	0.483	0.715	0.551
LSTM+AT <i>multi-task</i>	0.735	0.734	0.532	0.540	0.754	0.577
SciResCLF <i>single-task</i>	0.772	0.772	0.672	0.685	0.795	0.640
+ <i>joint 1st role</i>	-	-	0.666	0.667	0.823	0.668
+ <i>joint 2nd role</i>	0.759	0.760	-	-	0.805	0.675
+ <i>joint function</i>	0.778	0.778	0.673	0.671	-	-
SciResCLF <i>multi-task</i>	0.784	0.783	0.706	0.709	0.817	0.715

Table 4: Comparison results of SciResCLF for the three classification tasks.

placeholder. And the words of resource name are masked with a “[MASK]” token. Finally we get a training set of 2,910 samples and a testing set of 235 samples. To evaluate the predicted ranking list, we report the Precision@Top3 and the MAP metrics.

6.2 Baselines and Setups

For the classification task, we compare the SciResCLF with widely used context classification approaches on the SciRes: **Average Embedding + LR/SVM**, two machine learning algorithms, logistic regression (LR) and SVM, using the average of word embeddings as input; **FastText**, an implementation for FastText of [Joulin et al. \(2017\)](#); **CNN**, an implementation for TextCNN of [Kim \(2014\)](#); **RCNN**, an implementation for a recurrent convolutional neural network of [Lai et al. \(2015\)](#); **LSTM**, a 3-layer structure with input word embeddings, a bidirectional LSTM layer and a softmax output layer; **LSTM+AT**, using the attention mechanism to LSTM; and **LSTM+AT *multi-task***, using LSTM+AT to jointly learn the three tasks.

For the recommendation task, we compare our SciResREC with the **Random Forest (RF) classifier**, which is robust to overfitting even with large numbers of features. For the RF classifier, we use two types of features: **BoW+TFIDF**, the 20,000 most frequent words from the training set are selected and the TFIDF of each word is used as features; **N-grams+TFIDF**, the TFIDF of the most frequent 20,000 N-grams (up to 5-grams).

Our BERT encoder is based on Google’s reference implementation¹⁰ (TensorFlow 1.12.0). For

¹⁰<https://github.com/google-research/bert>

1st Roles	F1	2nd Roles	F1	Functions	F1
Material	0.787	Paper	0.902	Use	0.870
Method	0.793	License	0.870	Produce	0.861
Supplement	0.769	Media	0.846	Introduce	0.824
		Data	0.765	Other	0.716
		Code	0.742	Compare	0.667
		Tool	0.680	Extend	0.353
		Document	0.592		
		Algorithm	0.539		
		Website	0.448		

Table 5: Results (F1-score) on each category predicted by the best model for each of the three tasks.

training, we begin with the BERT-Base model (uncased, 12-layer, 768-hidden, 12-heads, 110M parameters) and then fine-tune the model on our training sets for the three classification tasks. The maximum word sequence length is 128; the learning rate is set to 5e-5 and all other defaults settings are used. For the recommendation model, the feature embedding dimension of role labels and function labels is 32, which are initialized at random and the hidden size is set to 64. For all the baselines, we use the word embeddings pre-trained on our large-scale scientific corpus, which is a full-text collection of 40 thousand research papers. And we stop the training when we found the best result in the developing set.

6.3 Results and Discussions

Classification Results

Our model achieves the best results on all the three classification tasks, as shown in Table 4. And the F1-score for each role or function type is shown in Table 5. From the results we can see that there is still a large gap with other general context-based classification tasks, which provides considerable potential for advancement in the future research. For more in-depth performance

Methods	Precision@Top3	MAP
RF (<i>BoW+TFIDF</i>)	0.438	0.275
RF (<i>N-grams+TFIDF</i>)	0.449	0.306
SciResREC	0.489	0.597
- <i>Function feature</i>	0.471	0.569
- <i>Role 2nd feature</i>	0.420	0.539
- <i>Role 1st feature</i>	0.399	0.497

Table 6: The Precision@Top3 and the MAP results for the ranking list predicted by SciResREC.

analysis, we note that most neural-based models are inferior to the traditional machine learning methods, which is perhaps due to the limited labeled datasets. While our model benefiting from the BERT encoder, which is pre-trained on vast amounts of text and fine-tuned on our task-specific, show an effective solution to the data limitation. We also see that the LSTM performs better when the attention layer introduced. It indicates the attention mechanism, which is sufficient used in the transformer structure of BERT, is significant to learn which words of a sequence are more important for determining the labels in our classification tasks. And the position embeddings in BERT is also effective to learn the relative positions between the resource citation and other words. By comparing the results between different joint models, we can get some interesting findings about the relationships between the resource roles and the resource functions. First, the general role and the fine-grained role can not benefit each other, which is perhaps because the high-level constraints can reduce the inter-class errors but meanwhile introduce more intra-class errors. When introducing the function information, the results for the general role will improve while the fine-grained role will slightly drop off. Moreover, jointly learning the two-level role information can observably enhance the results for function classification. And considering the complex interaction among the three tasks, the multi-task model achieves the best performance.

The resulting function classifier is sufficient to be applied to the entire ARC dataset. Nonetheless, errors remain. Consider the following example:

*To give their network a better initialization, they learn word embeddings ... They **released** their 50-dimensional word embeddings (vocabulary size 130K) under the name SENNA. <CITE>*,

which is notably a review of others’ work and the correct function is **Introduce**. While our model mistakes it to **Produce** caused by a higher

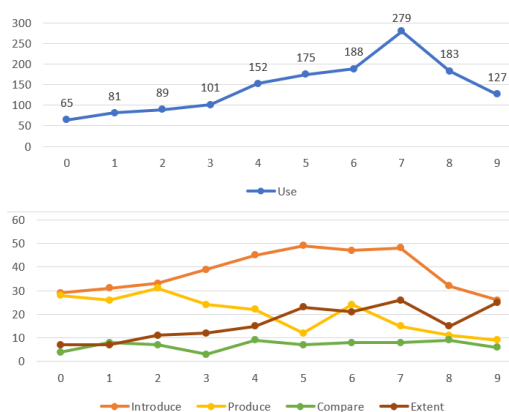


Figure 4: The revolution of resource functions in ten years starting from the first appearance.

attention score of the word “released”. This case inspires us that the dependency parsing can also be considered in the future research.

Recommendation Results

As Table 6 shows, our SciResREC framework outperforms the two baselines. An ablation test suggests that each feature component of our model contributes to the final performance, which indicates the information of role and function are helpful for understanding the scientific resources. And we can observe that the feature of 2-nd category role label has the largest impact on performance indicating that capturing fine-grained role types is important for recognizing specific resources.

Resource Evolution Analysis

To study what the resource function can tell us about the scientific resource development, we apply our function classifier trained on SciRes to the large ARC dataset. We select the top 100 most frequent resources and filter out the ones which have been existing less than 10 years. Finally we obtain a set of 33 resources and use our trained classifier to identify the function of each resource citation in its context. A statistical result is shown in Figure 4. The horizontal axis represents the number of years after the resource first appeared in the scientific corpus. From the figure we can see that a resource will drive to its maturity stage and be widely used in 4-5 years after it first exists. And in 8-9 years it will gradually be out of date and replaced by other new technologies. Moreover, citing the resources as background and making extensions based on the resources progressively increase along with time, which is consistent with the general expect.

7 Conclusion and Future Work

We introduce a novel task of *modeling the role and function for on-line resource citations in scientific literature*. For this task we first create an annotation scheme, collect a large-scale corpus and construct a manually labeled dataset. And we propose a multi-task model to jointly classify the role and function types based on the resource contexts. By incorporating the associated information, our SciResCLF framework effectively improves the performance across all tasks. Moreover, we propose a resource recommendation task and develop the SciResREC framework using the predicted labels as features. Our frameworks can contribute to build more powerful search and recommendation systems for scientific on-line resources. For future work, we will try using pre-trained BERT for scientific domain such as the SciBERT (Beltagy et al., 2019) and explore using our frameworks to help more tasks such as the evaluation, prediction and knowledge graph construction for scientific resources.

8 Acknowledgements

This work was supported by the National Key R&D Program of China (No. 2017YFB1002101), the Joint Advanced Research Foundation of China Electronics Technology Group Corporation (CETC) (No.6141B08010102), and National Natural Science Foundation of China (No. U1636203, No. 61602490).

References

- Ashutosh Adhikari, Achyudh Ram, Raphael Tang, and Jimmy Lin. 2019. Docbert: Bert for document classification. *CoRR*, abs/1904.08398.
- Isabelle Augenstein, Mrinal Das, Sebastian Riedel, Lakshmi Vikraman, and Andrew McCallum. 2017. Semeval 2017 task 10: Scienceie - extracting keyphrases and relations from scientific publications. In *SemEval@ACL*.
- Iz Beltagy, Arman Cohan, and Kyle Lo. 2019. Scibert: Pretrained contextualized embeddings for scientific text. *ArXiv*, abs/1903.10676.
- Guillermo de la Calle, Miguel García-Remesal, Stefano Chiesa, Diana de la Iglesia, and Victor Maojo. 2009. Biri: a new approach for automatically discovering and indexing available public bioinformatics resources from the literature. *BMC Bioinformatics*, 10:320 – 320.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.
- Geraint Duck, Goran Nenadic, Andy Brass, David L. Robertson, and Robert Stevens. 2012. bionerds: exploring bioinformatics database and software use through literature mining. In *BMC Bioinformatics*.
- Geraint Duck, Goran Nenadic, Andy Brass, David L. Robertson, and Robert Stevens. 2014. Extracting patterns of database and software usage from the bioinformatics literature. In *Bioinformatics*.
- Geraint Duck, Goran Nenadic, Michele Filannino, Andy Brass, David L. Robertson, and Robert Stevens. 2016. A survey of bioinformatics database and software usage through mining the literature. In *PLoS one*.
- Qi He, Daniel Kifer, Jian Pei, Prasenjit Mitra, and C. Lee Giles. 2011. Citation recommendation without author supervision. In *WSDM*.
- Qi He, Jian Pei, Daniel Kifer, Prasenjit Mitra, and C. Lee Giles. 2010. Context-aware citation recommendation. In *WWW*.
- Wenyi Huang, Saurabh Kataria, Cornelia Caragea, Prasenjit Mitra, C. Lee Giles, and Lior Rokach. 2012. Recommending citations: translating papers into references. In *CIKM*.
- Wenyi Huang, Zhaohui Wu, Liang Chen, Prasenjit Mitra, and C. Lee Giles. 2015. A neural probabilistic model for context based citation recommendation. In *AAAI*.
- Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *EACL*.
- David Jurgens, Srijan Kumar, Raine Hoover, Daniel A. McFarland, and Daniel Jurafsky. 2018. Measuring the evolution of a scientific field through citation frames. *TACL*, 6:391–406.
- Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *EMNLP*.
- Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In *AAAI*.
- Maria Liakata, Shyamasree Saha, Simon Dobnik, Colin R. Batchelor, and Dietrich Rebholz-Schuhmann. 2012. Automatic recognition of conceptualization zones in scientific articles and two life science applications. In *Bioinformatics*.
- Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *EMNLP*.

- Yi Luan, Mari Ostendorf, and Hannaneh Hajishirzi. 2017. Scientific information extraction with semi-supervised neural tagging. In *EMNLP*.
- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations.
- Alec Radford. 2018. Improving language understanding by generative pre-training.
- Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence. *CoRR*, abs/1903.09588.
- Jie Tang and Jing Zhang. 2009. A discriminative approach to topic-based citation recommendation. In *PAKDD*.
- Simone Teufel, Advaith Siddharthan, and Tidhar Dan. 2006. Automatic classification of citation function. In *Proc. 2006 Conference on Empirical Methods in Natural Language Processing, Sydney, Australia*, pages 103–110.
- Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. 2019. Bert post-training for review reading comprehension and aspect-based sentiment analysis. *CoRR*, abs/1904.02232.
- Yasunori Yamamoto and Toshihisa Takagi. 2007. Orefil: an online resource finder for life sciences. *BMC Bioinformatics*, 8:287 – 287.
- He Zhao, Zhunchen Luo, Chong Feng, and Yuming Ye. 2019. A context-based framework for resource citation classification in scientific literatures. In *SIGIR*.