GrantRel: Grant Information Extraction via Joint Entity and Relation Extraction

Junyi Bian [1 8] **, Li Huang** [1 8] **, Xiaodi Huang** [2] **, Hong Zhou** [3]**, Shanfeng Zhu** [4 5 6 7 8]
[1] School of Computer Science, Fudan University, Shanghai 200433, China
[2] School of Computing and Mathematics, Charles Sturt University
Albury, NSW 2640, Australia
[3] Atypon Systems, LLC, UK
[4] Institute of Science and Technology for Brain-Inspired Intelligence, Fudan University, China
[5] Key Laboratory of Computational Neuroscience and Brain-Inspired Intelligence
(Fudan University), Ministry of Education, Shanghai 200433, China
[6] MOE Frontiers Center for Brain Science, Fudan University, Shanghai 200433, China
[7] Zhangjiang Fudan International Innovation Center, Shanghai 200433, China
[8] Shanghai Key Lab of Intelligent Information Processing,
Fudan University, Shanghai 200433, China
{zhusf, 20110240003}@fudan.edu.cn, hzhou@atypon.com

Abstract

As part of scientific articles, grant information refers to funder names and their corresponding grant numbers. Extracting such funding information from articles is of significant importance to both academic and funding bodies. The studies on this topic face two major challenges: 1) no high-quality benchmark datasets; and 2) difficulties in extracting complex relationships between funders and grantIDs. In this paper, we present a novel pipeline framework called GrantRel, which consists of a funding sentence classifier, as well as a joint entity and relation extractor. For this purpose, we manually label two high-quality datasets called Grant-SP and Grant-RE, respectively. In addition, our relation extraction (RE) model uses both position embedding and context embedding in an adaptive-learning way. The experiment results have demonstrated that our model outperforms several state-of-the-art BERT-based RE baselines as higher as 6.5% of F1 scores against the PubMed Central (PMC) test set and 3.5% of that against the arXiv test set.

1 Introduction

As an element of scientific articles, grant information generally includes funder names, grant numbers, and their relations. Specifically, a funder name refers to an agency, organization, or program which provides financial support for the research. A grantID is a numerical string by which to distinguish one grant from another. Such grant source information should be automatically identified. The reasons for this are as follows: (a) The funding bodies need to track their funding statuses; (b) For the academic, the impact of funding agencies in the scientific literature can be measured, and agencies actively supporting specific directions can be identified; and (c) The literature management systems require the funding register information. Therefore, a systematic framework that is capable of automatically extracting grant information from papers is needed.

Extracted Grant Information

Funder Name	Grant ID
NIH	DK76131
NIH	HL49277
Juvenile Diabetes Research Foundation	-

Figure 1: In this sentence, *DK76131* and *HL49277* are two grant numbers of the agency funder *NIH*. The *Juvenile Diabetes Research Foundation* is another funder.

Generally, authors would express their acknowledgments in the papers if their research receives funding. Based on this fact, an extraction should start with selecting the funding sentences from an acknowledgment. To train such a classifier, we manually build a dataset named Grant-SP with 1402 sentences. After that, a relation extraction (RE) model is applied on funding sentences to identify grant entities and their relations. Specifically, a funder name entity and a grant number entity are viewed as a subject and an object in a relation triplet, respectively. For accurately extracting grant information via RE from scientific articles,

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 2674–2685

there are, however, two major challenges: 1) no high-quality RE benchmark datasets; and 2) difficulties in extracting complex relationships between funding organizations and grantIDs by using RE models.

The 2017 BioASQ challenge (Nentidis et al., 2017) is about building a system that extracts the funding information from a benchmark dataset on the full text of biomedical papers. From this dataset, only 107 agencies, however, are required to be identified as funder names such as NIH or CIHR. For example, the winning systems on the challenge such as GrantExtractor (Dai et al., 2018) cannot extract the grant funders beyond 107 agencies such as NASA or JSPS. For overcoming this limitation, we propose a manually-crafted dataset Grant-RE which covers nearly 2k different funder names.

There often exist the complex, many-to-many relationships between funder names and grantIDs. This fact makes it difficult to identify such complex relationships by using a RE model. In addition, the Grant-RE dataset has only two types of entities but with a higher frequency in a sentence, compared with common REs. For example, we count the number of entities with the highest number of occurrences in each sentence of CoNLL04 (Roth and Yih, 2004). The average number of such entities is 2.1, while the number is 2.8 in our Grant-RE dataset. This would be challenging to build correct relations between two entities. Further, a grantID or a funder name could even present independently (see Figure 1). To address this challenge, our GrantRel framework includes a novel joint entity and RE model. This model starts with using the powerful encoding layer of BioBERT (Lee et al., 2019), and can jointly extract funders, grantIDs, and their relations by considering grant relation features. It has been demonstrated that our RE model outperforms the state-of-the-art RE baselines in Grant-RE by a large margin.

In summary, this paper has the following contributions: (a) We propose a novel framework called GrantRel that automatically extracts grant information from academic papers. The RE model in GrantRel is designed to accurately extract both grant number, funder name, and their relation by combining the location of grant information in a sentence and its context embedding in an adaptive way. (b) By manually labelling funding sentences, we retrieved the papers from PubMed Central (PMC) and arXiv, and created a classification dataset called Grant-SP with 1402 sentences for training, as well as a grant RE dataset called Grant-RE with 3331 sentences. (c) Extensive experiments have been conducted to test the performance of the whole framework, and to compare RE models with the RE baselines in both biomedical (PMC) and universal (arXiv) domains.

To the best of our knowledge, this is the first work on reporting a benchmark dataset[1] and model for extracting general grant information by the supervised RE.

2 Related work

The prior studies have addressed the problem of grant information extraction with a limited capability by traditional machine learning methods. A naive Bayes method (Kim et al., 2009) was used to locate the grant support (GS) zone from an article text, followed by inferring GS types with a pattern matching method. As such. only fourteen GS types can be identified. Zhang et al. (2009) used a semi-supervised method to detect grant-related zones from online medical articles. Gross et al. (2016) proposed a rule-based model for extracting metadata (grant number and grant sponsor) from articles. All these methods do not establish a specific relationship between a funder and a grant number.

Recently, Dai et al. (2018) built a pipeline system for grant information extraction. They first selected funding sentences by relying on manually designed features, then extracted grantIDs by using the BiLSTM-CRF tagger, finally identified the agencies by applying a multi-class classifier to each grantID with manually designed features. However, this method is still limited, because it cannot recognize new grant agencies other than 107 designated ones. In contrast, GrantRel learns a joint model on the name recognition of funder and grantID, and extraction of their relationship. As such, it can handle new funder names very well.

Traditionally, RE is achieved through a pipeline (Zelenko et al., 2003; Chan and Roth, 2011; Zhou et al., 2005) with two phases: entity recognition and relation classification. Since the two phases may benefit from the use of correlated signals, research for joint entities and relation extraction have attracted more attention. Early work of joint approaches uses feature-based models (Yu and Lam, 2010; Miwa and Sasaki, 2014). Recently, neural network-based models (Zeng et al., 2018; Li et al.,

[1] https://github.com/Eulring/GrantRel

2019; Dai et al., 2019; Fu et al., 2019), especially the BERT-based (Devlin et al., 2019) models (Wei et al., 2020; Eberts and Ulges, 2019; Wang et al., 2020) that replace the manually constructed features with learned representation, have achieved the considerable success in completing the RE task. Following this idea, our RE model uses BioBERT (Lee et al., 2019) as an encoding core. Inspired by the CasRel (Wei et al., 2020) further, our RE model establishes a relation as a function that maps funder to grantID. Since an ordinary model cannot accurately distinguish the complex relationship between multiple funders and grantIDs, the features that can describe the interaction between entities become critical. Therefore, we use relative position embedding and localized context embedding (Eberts and Ulges, 2019), which make a significant improvement. In addition, we design a mechanism by adaptively integrating the two embeddings to obtain better performance.

3 Dataset description

Although BioASQ 5c provides a dataset of grant information extraction, it has three serious drawbacks, 1) with only 107 agency names used in the labels, many common funder names are ignored. In fact, there are nearly 57000 different funder names in a funder name database downloaded from crossref [2]; 2) normalized agency names and the corresponding grantIDs are provided without specifying their exact positions in the articles, which is inconvenient for supervised RE training; 3) the quality of annotation is limited (Dai et al., 2018). To address these issues, we therefore manually built two datasets, namely, Grant-RE and Grant-SP, for the two modules in our framework.

3.1 Dataset: Grant-RE

Grant-RE is the dataset for the RE model. We downloaded articles with the original xml format from open access subset of PMC [3]. The raw text from the acknowledgement section of an article was then parsed into readable paragraphs, and the sentences were split by using NLTK[4] tools. We manually selected the funding sentence and labelled grant information. A grant relation is represented as four integers for the intervals of a funder entity and a grantID entity.

[2]https://gitlab.com/crossref/open_funder_registry
[3]https://ftp.ncbi.nlm.nih.gov/pub/pmc/oa_bulk/
[4]https://www.nltk.org/

As given in Table 1, we present the statistics of the train/dev/test splits for the grant information extraction dataset. There are two versions of test splits. One is from PMC, which is as same as train/dev split, while another from arXiv is used for conducting evaluations of our approaches on the universal domain. To ensure quality, the GrantRE dataset was annotated by 4 well-trained annotators, with each sentence being annotated twice by different annotators. For those sentences having different annotations, we will seek advice to experts to decide their final annotations. Besides, the test data splits were repeatedly checked 3 times.

	train	dev	test	test[a]
# sentence	2104	477	500	350
# funder entity	4592	1192	1297	706
# grantID entity	4195	1084	1116	646
# grant relation	4107	1097	1179	684

Table 1: Statistics of Grant-RE. Test, train, and dev sets are from PMC. The test[a] is from arXiv papers.

3.2 Dataset: Grant-SP

Unlike Grant-RE, we sampled sentences from all sections in a paper to annotate a funding sentence classification dataset. Because the numbers of positive and negative sentences were unbalanced, we discarded most of the negative sentences in the train/dev set to accelerate the training.

The test set in Grant-SP is used not only for the classifier evaluation, but also for the whole framework evaluation. For building the test set, we strictly followed our framework pipeline: for each article, we kept all negative sentences, and tagged grant information for positive sentences. Because the classifier has a high recall, when labeling the test split, we borrow the outputs from trained models for the auxiliary reference. For a sentence that the classifier considers to be positive and the RE model can also extract information, we manually relabel it. In Table 2, we report the statistics of train/dev/test splits.

	train	dev	test
# sentence	908	282	16069
# positive	158	51	101
# articles	-	-	50

Table 2: Statistics of Grant-SP

4 Methodology

4.1 Framework

As shown in Figure 2, the left side illustrates the overall workflow of our GrantRel. Given prepossessed sentences from raw articles, the sentence classification module selects the sentences that may contain grant information. Without this step, the framework may suffer from low precision. After this, the RE module will extract grant information.

4.2 Identification of funding sentences

Our models use a pre-trained BioBERT (Lee et al., 2019) to encode context information. Suppose sentence $\mathbf{x}$ is first tokenized into byte-pair encoded (BPE) tokens (Sennrich et al., 2016) $x = \{x_1, x_2, ..., x_l\}$ with length l. BioBERT takes it as an input and outputs a length of $l+2$ embedding sequence $\boldsymbol{e} = \{\boldsymbol{e}_{CLS}, \boldsymbol{e}_0, \boldsymbol{e}_1, ..., \boldsymbol{e}_l, \boldsymbol{e}_{SEP}\}$. The additional embedding $\boldsymbol{e}_{CLS}$ captures the whole sentence context. A Logistic Regression is then used to calculate the probability:

$$p_{sent} = \sigma(W_{sent}\boldsymbol{e}_{CLS} + b_{sent}) \tag{1}$$

Here the $\sigma(\cdot)$ is the sigmoid function, and $\{W_{sent}, b_{sent}\}$ are trainable parameters.

4.3 Joint entity and RE

A grant relation consists of a funder (subject eneity s) and grantID (object entity o). Given input sentence $\mathbf{x}$ and its tokens x, we use T to represent the set of all grant relations of this sentence. The likelihood of all relations $T = \{(s, o)\}$ in this sentence can be written as:

$$\prod_{(s,o)\in T} p(s, o|x) = \prod_{s\in T}\left[p_{fd}(s|x)\prod_{o\in T|s} p_{gr}(o|s, x)\right] \tag{2}$$

In Eq.(2), the role of $p_{fd}(s|x)$ acts as a subject tagger that recognizes funder name entities in the sentence, where $s \in T$ denotes a subject appearing in T. $p_{gr}(o|s, x)$ is to identify the object with only having a relation with the specific s. $o \in T|s$ is the object in T led by subject s. Indeed, this extracting scheme allows us to extract the grantID at once for each funder name. To handle independent grantIDs, we add an additional probability item p_{id} to tag grantID. As such, the overall likelihood of grant information in x is:

$$\prod_{s\in T}\left[p_{fd}(s|x)\prod_{o\in T|s} p_{gr}(o|s, x)\right]\prod_{o\in T} p_{id}(o|x) \tag{3}$$

4.3.1 Funder name detection

The low-level tagging module aims to detect all possible funder entities from x. Similar to sentence classification, BioBERT (Lee et al., 2019) generates the tokens representation $\boldsymbol{e}$. Using the IOB tagging scheme, we predict the IOB tag y for each token. A specific operation on the i^{th} token is as follow.

$$y_i = \text{softmax}(W_{fd}\boldsymbol{e}_i + b_{fd}) \tag{4}$$

4.3.2 Grant relation detection

A funder name is either extracted at the first phase or provided by the dataset during the training. The conditional grant number tagger distinguishes the grantID that only belongs to this particular funder name from other candidates. We first use a fused BERT embedding $\boldsymbol{e}_{fd}$ to represent this funder name:

$$\boldsymbol{e}_{fd} = f_{fd}(\boldsymbol{e}, \boldsymbol{u}_{fd}) \tag{5}$$

where $\boldsymbol{u}_{fd} = [u_{fd}^{start}, u_{fd}^{end}]$ is the position boundary of a funder name entity. Since the length of the funder name can vary, function $f_{fd}(\cdot)$ is used to produce a fixed-size feature for funder names. On choosing $f_{fd}(\cdot)$, we use the average pooling of the entire entity span. For each token, the grant relation module classifies tag $\boldsymbol{z}$ as :

$$z_i = \text{softmax}(W_{gr}[\boldsymbol{e}_{fd}, \boldsymbol{e}_i, \boldsymbol{e}_{gr}] + b_{gr}) \tag{6}$$

where $\boldsymbol{e}_i$ is the encoding of token x_i, and $\boldsymbol{e}_{gr}$ is the grant relation feature explained below (Senction 4.4).

4.3.3 GrantID detection

If a funder name is undetected in the previous step, we will miss the corresponding grant numbers. In addtion, some grant numbers even occur independently for some reasons, such as a sentence segmentation error. For extracting the complete grant information, an auxiliary item $p_{id}(o|x)$ is used to tag all grantIDs . We view the detection of grantIDs as a special case of the grant relation detection by using trainable vector $\hat{\boldsymbol{e}}$ to represent all funder names.

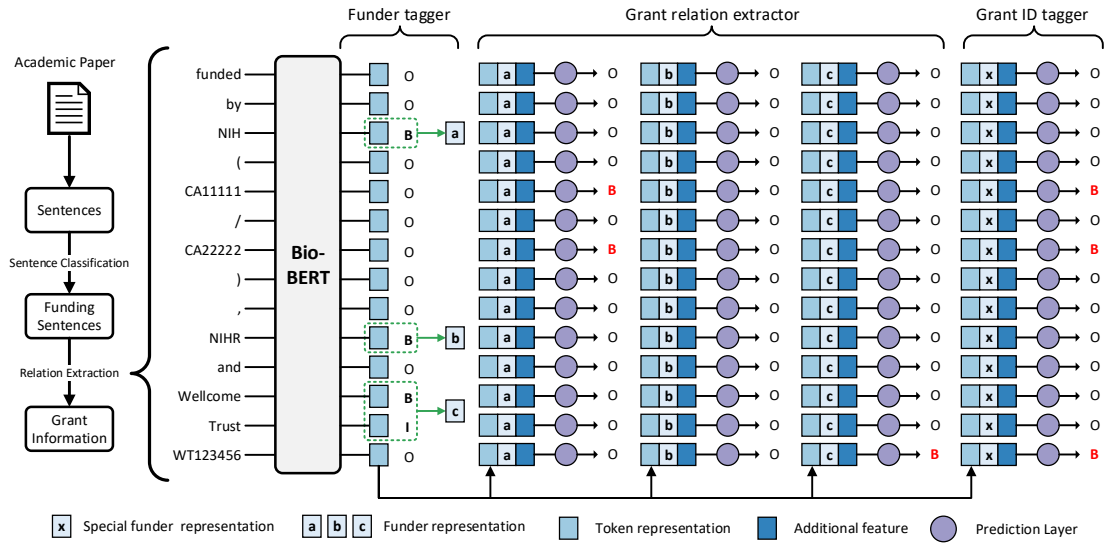

Figure 2: The left side is the grant information extraction pipeline, with the RE model in GrantRel on the right side. For convenience, words are presented without tokenization. The three funders in the sentence are detected by the funder tagger. For each funder name, its corresponding grantID is matched by predicting its label in each position. Note that an ID tagger is able to find all grant numbers at once.

This means that all grantIDs in the sentence should match this special funder name. The operation on the i^{th} token is as follows.

$$o_i = \text{softmax}(W_{gr}[\hat{e}, \boldsymbol{e}_i, \boldsymbol{e}_{gr}] + b_{gr}) \quad (7)$$

4.4 Grant relation feature

To establish the correct connection between a grantID and a funder name, we use additional features $\boldsymbol{e}_{gr}$ other than entity representation, which characterize the relation between the funder name and the i^{th} token in x in Eq 6. These features can be captured by using information such as the span of funder $\boldsymbol{u}_{fd}$ and input context $\mathbf{x}$.

4.4.1 Position embedding

First, we use the relative distance to measure the two positions:

$$d(i,j) = min(max(-k, (i-j)), k) \quad (8)$$

where the distance is clipped into a region of $[-k, k]$. The position of an extracted funder entity is an interval $\boldsymbol{u}_{fd}$. Some funder names have relative long spans, so it would be inaccurate to represent all the distances by a single number. We concatenate two relative distance embedding as our final position embedding:

$$\boldsymbol{e}_{pos} = [\text{emb}(d(u_{fd}^{start}, i)), \text{emb}(d(u_{fd}^{end}, i))] \quad (9)$$

where $\text{emb}(\cdot)$ represents a learnable embedding.

4.4.2 Context embedding

We observe that the context for the funder and target token has semantic information that is helpful for establishing relationships. Therefore, we utilize $\boldsymbol{e}$ to represent context embedding $\boldsymbol{e}_{ctx}$. For example, a sentence is: *"funded by NIH (CA123456) , and CIHR (R01 12111)"* During the grant relation phase, the subject funder name is *"NIH"*, the target token is *"12111"*, their localized context is the blue part of *"(CA123456) , and CIHR (R01"* in the sentence. The max-pooling for encoding $\boldsymbol{e}$ of the localized context is used to generate a fix-size representation $\boldsymbol{e}_{ctx}$.

4.4.3 Adaptive embedding

A combination of two embeddings of position and context can make our model more robust. Furthermore, when the context meaning is abundantly clear, we expect the proposed model can concentrate more on the context information. According to this view, we propose a mechanism that can balance two embeddings to deal with different situations in an adaptive way:

$$\boldsymbol{e}_{gr} = \alpha \cdot \boldsymbol{e}_{pos} + (1-\alpha) \cdot \boldsymbol{e}_{ctx} \quad (10)$$

where α is a scalar decided by the context embedding as:

$$\alpha = \sigma(W_{ada}\boldsymbol{e}_{ctx} + b_{ada}) \quad (11)$$

Test Set	Models	Funder Entity			GrantID Entity			Grant Realtion		
		Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
PMC	CasRel(Wei et al., 2020)	-	-	-	-	-	-	71.7	49.0	58.2
	SpERT(Eberts and Ulges, 2019)	88.6	93.0	90.7	89.8	97.9	93.6	82.6	90.3	86.3
	GrantRel-base	91.9	92.9	92.4	92.8	97.7	95.2	68.8	56.9	62.3
	GrantRel-pos	91.9	93.0	92.5	96.1	97.8	96.9	88.9	87.4	88.1
	GrantRel-ctx	91.3	92.3	91.8	95.4	97.8	96.6	90.0	89.6	89.8
	GrantRel	91.8	93.0	92.4	96.4	97.3	96.8	92.7	89.7	**91.2**
	GrantRel$_{BERT}$	91.7	92.0	91.9	95.3	97.2	96.2	91.4	89.2	90.3
arXiv	CasRel(Wei et al., 2020)	-	-	-	-	-	-	70.1	39.0	50.1
	SpERT(Eberts and Ulges, 2019)	82.5	85.9	84.1	86.5	94.3	90.3	76.6	81.8	79.1
	GrantRel-base	83.6	82.9	83.2	90.3	97.2	93.6	66.8	49.7	57.0
	GrantRel-pos	85.8	85.7	85.8	92.5	97.1	94.7	87.0	81.0	83.9
	GrantRel-ctx	85.4	85.4	85.4	93.9	96.8	95.3	85.1	85.1	85.1
	GrantRel	86.2	85.9	86.0	93.9	96.6	95.2	86.9	84.3	**85.6**
	GrantRel$_{BERT}$	86.3	83.6	84.9	90.7	96.0	93.3	83.8	80.1	82.0

Table 3: The performance of GrantRel compared with typical RE models on the test sets of PMC and arXiv.

5 Experiments

In this section, we compare the performance of the GrantRel RE model with several RE baselines on the Grant-RE dataset. The varying degree of the improvement of the RE model with different features is also examined. Finally, the overall performance of the proposed GrantRel framework is comprehensively evaluated.

5.1 Experiment settings

In Table 3, we define **GrantRel-base** as the pure RE model without considering additional features. Compared to GrantRel-base, **GrantRel-pos** makes use of the position embedding, while **GrantRel-ctx** uses context embedding. As our ultimate model, GrantRel integrates two embeddings of position and context in an adaptive way. These models both initially encode the input by using the BioBERT pretraining. In particular, **GrantRel**$_{BERT}$ uses the BERT encoding for a fair comparison with other BERT-based baselines: **CasRel** (Wei et al., 2020) the state-of-the-art model of WebNLG (Gardent et al., 2017) and NTY(Riedel et al., 2010) dataset, and **SpERT** (Eberts and Ulges, 2019) the state-of-the-art model of CoNLL2004 (Roth and Yih, 2004) dataset. In order to use the SpERT in Grant-RE, we extend the max span size from the original one of 20 to 25. This increases the training time, but covers the widest span of funders in our dataset. Other baselines settings strictly follow the optimal settings of the original paper.

We used Pytorch to implement the deep learning models. All GrantRel models were trained by using Adam (Kingma and Ba, 2015) optimizer. During the training, the number of epochs was chosen as 30, and the learning rate dropped 20% in every two epochs with an initial learning rate of 5e-5. In addition, the distance threshold k in position embedding was set to 40, together with the batch size of 10, and the dimension of context and position embedding of 768. All of our experiments were conducted on a single GTX 1080Ti GPU.

5.2 Evaluation metrics

In this work, we use f1-score (F1), precision (Prec.), and recall (Rec.) to measure the performance of our models on extracting grant relation, grant number, and funder entities. For all the evaluations, a predicted entity is correct only if both its head and tail are correct.

Grant relation evaluation: For relation evaluation, we tested only the triplets with a complete grantID and funder name in the test dataset by excluding isolated funder names or grantIDs. This also held true for the other RE tasks.

Grant information evaluation: Grant information evaluation aims to test the overall performance for our GrantRel framework. Differing from relation evaluation, the overall evaluations include isolated funder names and grant numbers.

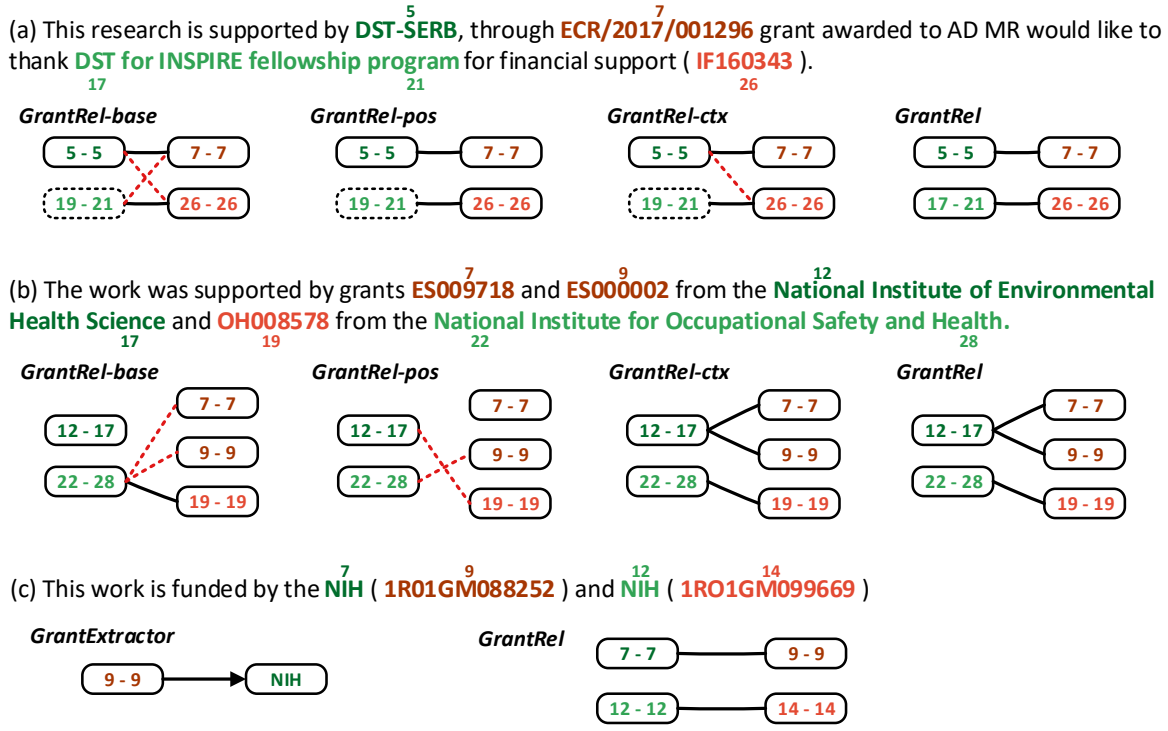

Figure 3: Outputs from different models. Each circle represents an entity, while the number in a circle represents the entity's span positions. A dotted circle indicates an output of a wrong entity span. A red dotted line indicates a wrong relationship. The arrow in case (3) means that the funder was directly inferred from GrantID. In all of these examples, GrantRel outputted correct results.

5.3 Experiment Results: Grant-RE

The experiments here focus only on the RE model, with the funding sentences provided. Main results on the Grant-RE dataset are shown in Table 3. We have four main findings. (1) GrantRel achieves the best performance on both PMC and arXiv test splits, with an increase of 3.9% and 6.5% respectively compared with other baselines. (2) Grant relation features are critically important. Without adding additional features that characterise the relationship between a funder name and a grant number, the GrantRel-base model and CasRel have a bad performance. When the position embedding was integrated (GrantRel-pos), the f1-score, however, increase significantly with 27.5% improvements. Context embedding(GrantRel-ctx) perform better than position embedding by another increase of 1.7%. SpERT using a context embedding also has considerable performance(86.3%). Further, the combination of context embedding and position embedding in GrantRel produce the best f1-score 91.2%. (3) GrantRel$_{BERT}$ perform worse than GrantRel in both test sets. Which means that BioBERT, as an encoding layer, performs better than BERT in terms of grant information extraction. The reason for this is that BERT was trained only from wiki and books, but BioBERT was trained on additional scientific papers. (4) When tested on a new domain (arXiv), the performance of all models dropped slightly. This is because most funder names in the arXiv test set are different from those in PMC.

5.4 Experiment Results: Grant-SP

Before applying relation extraction, we first identify which sentence in a given paper is grant-related by using the sentence classifier. In this experiment, two models are combined into a pipeline. If a sentence is predicted as negative by the classifier, we will exclude it from relation extraction. As we know, the best RE model from Section 5.3 is the downstream module. To verify the effect of the funding sentence classifier, we compared our GrantRel (Clf+RE) with the framework without classifier (RE), framework with key-words sentence matching (Key+RE), and framework with perfect classifier (Gold+RE), respectively. The experiment results are reported in Table 4. Since we discarded most of the negative samples in training, our funding sentence classifier had achieved a

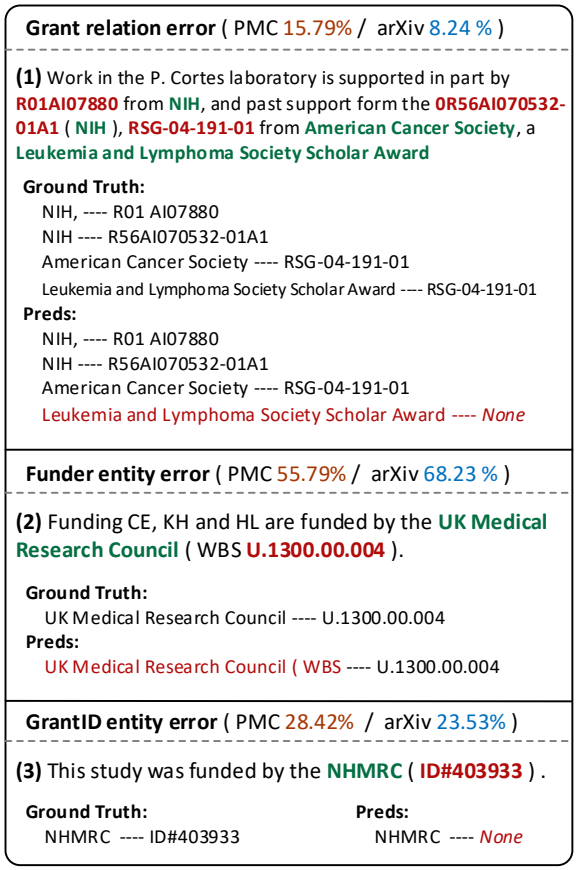

Figure 4: Example of error cases from the GrantRel RE model. There are three types of errors, each of which is statistically analyzed on PMC test set and arXiv test set.

very high recall. Compared with RE and Key+RE models, the framework with the sentence classifier achieved a significantly higher precision. Meanwhile, the sentence classifier could reduce search costs. In our experiments, the RE model could process 25 sentences per second. In contrast, our framework could process 50 sentences per second by filtering out the non-funding sentences.

5.5 Case study

We review the results from different models and select some cases for further analysis in this section.

First, we examine the results from RE models with different features in Figure 3. In case (a), only the GrantRel identified correct funder names and grant relations. The base model GrantRel-base matches each agency to all grant numbers. GrantRel-pos produced the correct relation. However, GrantRel-ctx built the wrong connection between *DST-SERB* and *ID160343*. We speculate that the context information for the entity and the ID may not work. But, the distance between the two

Pipelines	Grant Sent. Prec.	Rec.	F1	Grant Info. Prec.	Rec.	F1
RE	-	-	-	12.0	94.9	21.3
Key + RE	51.0	74.3	60.5	86.1	68.9	76.5
Clf + RE	85.6	100	**92.2**	85.7	93.3	**89.4**
Gold + RE	100	100	100	89.8	93.3	91.6

Table 4: The pipeline performance on Grant-SP. Clf+RE is the GrantRel framework; Key+RE selects the funding sentence by keywords matching; Gold+RE uses the ground truth to select funding sentence; and RE extracts grant information on each sentence.

entities is too long. As a result, only models that incorporate position information output the correct relation. In case (b), GrantRel-base still had terrible performance. For the sentences with grantIDs that are located at the front of their corresponding funders, GrantRel-pos performed poorly. Nevertheless, this case can be easily handled by considering context information as does in our framework. By analysis, we find that the base model intends to predict whether a funder is associated with numbers first. If it is, the funder will be established the relations with all found grantIDs. If not, the funder will be regarded as isolated. Context embedding can build relations in a complicated semantic situation. Position embedding is particularly helpful when context embedding is inadequate or ambiguous. In case (c), we compare our framework with GrantExtractor (Dai et al., 2018). GrantExtractor can only extract grant number *1R01GM088252* from the sentence and infer the *NIH* by this ID. However, it even misses the number *1RO1GM099669* if the char "0" is wrongly spelled as "O". It is easy for our model to identify such error-spelled grantIDs.

Second, we carry out the error analysis on wrong cases by GrantRel (see Figure 4). In case (1), grantID *RSG-04-191-01* is related not only to *American Cancer Society*, but also to *Leukemia and Lymphoma Society Scholar Award*. But the RE model treated the following entity as an independent funder. Such an example requires the model to have a deeper understanding of semantic information. Moreover, training data lacks such a kind of samples which make the RE model more difficult to extract. In case (2), GrantRel wrongly recognized the funder name, and this kind of error accounts for the majority. In case (3), GrantRel failed to find the grant number. This can be explained by the

fact that the *"ID"* mostly appears independently in training without being tagged as a number entity. Such errors can be corrected by using more fine-grained tokenization.

6 Conclusion

In this paper, we have presented a novel pipeline framework named GrantRel for automatically extracting grant information from academic articles. The framework has two components of the text classification module and the joint RE module. Moreover, we manually labelled two datasets for training and testing modules. Compared to the previous approaches, the proposed framework has achieved significant improvements in extracting any types of funder names mentioned in articles. Overall, the experiments have demonstrated that our RE model outperforms several state-of-the-art baselines of grant extraction.

7 Acknowledgments

This study has been supported by Shanghai Municipal Science and Technology Commission [2018SHZDZX01], ZJ Lab, Shanghai Center for Brain Science and Brain-Inspired Technology, 111 Project [B18015] and National Natural Science Foundation of China (No.61872094). This research is also sponsored by Atypon Systems, LLC.

8 Ethics Statement

Datasets have been collected in a manner which is consistent with the terms of use of any sources and the intellectual property. For each annotator, we compensate based on the number of annotated sentences. More details of our datasets are depicted in Section 3.

References

Yee Seng Chan and Dan Roth. 2011. Exploiting Syntactico-Semantic Structures for Relation Extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 551–560.

Dai Dai, Xinyan Xiao, Yajuan Lyu, Qiaoqiao She, Shan Dou, and Haifeng Wang. 2019. Joint Extraction of Entities and Overlapping Relations using Position-Attentive Sequence Labeling. *AAAI 2019 : Thirty-Third AAAI Conference on Artificial Intelligence*, 33(1):6300–6308.

Suyang Dai, Zihan Zhang, Wenxuan Zuo, Xiaodi Huang, and Shanfeng Zhu. 2018. Grantextractor: A winning system for extracting grant support information from biomedical literature. In *IEEE International Conference on Bioinformatics Biomedicine*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT 2019: Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 4171–4186.

Markus Eberts and Adrian Ulges. 2019. Span-Based Joint Entity and Relation Extraction with Transformer Pre-Training. In *ECAI*, pages 2006–2013.

Tsu-Jui Fu, Peng-Hsuan Li, and Wei-Yun Ma. 2019. GraphRel: Modeling Text as Relational Graphs for Joint Entity and Relation Extraction. In *ACL 2019 : The 57th Annual Meeting of the Association for Computational Linguistics*, pages 1409–1418.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. Creating Training Corpora for NLG Micro-Planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 179–188.

Mark Gross, Tammy Bilitzky, and Richard Thorne. 2016. Extracting Funder and Grant Metadata from Journal Articles. *Balisage: The Markup Conference*.

Jongwoo Kim, Daniel X. Le, and George R. Thoma. 2009. Inferring grant support types from online biomedical articles. In *2009 22nd IEEE International Symposium on Computer-Based Medical Systems*, pages 1–6.

Diederik P. Kingma and Jimmy Lei Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR 2015 : International Conference on Learning Representations 2015*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and Jiwei Li. 2019. Entity-Relation Extraction as Multi-Turn Question Answering. In *ACL 2019 : The 57th Annual Meeting of the Association for Computational Linguistics*, pages 1340–1350.

Makoto Miwa and Yutaka Sasaki. 2014. Modeling Joint Entity and Relation Extraction with Table Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1858–1869.

Anastasios Nentidis, Konstantinos Bougiatiotis, Anastasia Krithara, Georgios Paliouras, and Ioannis A. Kakadiaris. 2017. Results of the fifth edition of the BioASQ Challenge. In *BioNLP 2017*, pages 48–57.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *ECMLPKDD'10 Proceedings of the 2010th European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part III*, pages 148–163.

Dan Roth and Wen-tau Yih. 2004. A Linear Programming Formulation for Global Inference in Natural Language Tasks. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*, pages 1–8, Boston, Massachusetts, USA. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1715–1725.

Yucheng Wang, Bowen Yu, Yueyang Zhang, Tingwen Liu, Hongsong Zhu, and Limin Sun. 2020. TPLinker: Single-stage Joint Extraction of Entities and Relations Through Token Pair Linking. In *COLING*, pages 1572–1582.

Zhepei Wei, Jianlin Su, Yue Wang, Yuan Tian, and Yi Chang. 2020. A Novel Cascade Binary Tagging Framework for Relational Triple Extraction. In *ACL 2020: 58th annual meeting of the Association for Computational Linguistics*, pages 1476–1488.

Xiaofeng Yu and Wai Lam. 2010. Jointly Identifying Entities and Extracting Relations in Encyclopedia Text via A Graphical Model Approach. In *Coling 2010: Posters*, pages 1399–1407.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. 2003. Kernel methods for relation extraction. *Journal of Machine Learning Research*, 3(6):1083–1106.

Xiangrong Zeng, Daojian Zeng, Shizhu He, Kang Liu, and Jun Zhao. 2018. Extracting Relational Facts by an End-to-End Neural Model with Copy Mechanism. In *ACL 2018: 56th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 506–514.

Xiaoli Zhang, Jie Zou, Daniel X. Le, and George R. Thoma. 2009. A semi-supervised learning method to classify grant support zone in web-based medical articles. In *Proceedings of SPIE, the International Society for Optical Engineering*, volume 7247.

GuoDong Zhou, Jian Su, Jie Zhang, and Min Zhang. 2005. Exploring Various Knowledge in Relation Extraction. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 427–434.

A Tagging Standard

In the process of dataset construction, it is a challenge to set a standard for annotations, especially for determining funder entities. After reviewing lots of examples, we decided to use the following rules to determine a funder entity in our tagging.

- Apart from agencies, specific programs, awards, foundation, and fellowships are also regarded as funder names.
- If the name of a program, or fellowship, or award, etc., is associated with the corresponding agency, we will treat them as a whole funder name.
- The address or abbreviation associated with a funder name will be included as part of its funder name.
- The sub-division associated with an agency is viewed as part of the funder name.

B Performance Impact of the Funder Representation

In Table 5, we examine the performance under different funder representations e_{fd}. The following RE models all adopted a standard GrandRel structure (Using the adaptive embedding), with differing only in their representation approaches of funder names

	Grant Realtion		
Funder Representation	Prec.	Rec.	F1
Head	91.71	90.00	90.85
Head+Max	91.93	89.83	90.87
Head+Mean	91.89	90.25	91.06
Head+Tail	91.46	88.90	90.16
Max	92.56	89.58	91.04
Mean	92.65	89.75	91.18

Table 5: Results of GrantRel with different funder representations with respect to the PubMed test set.

- **Head**: The funder entity representation uses the first token representation.
- **Head+Max**: The max-pooling of the entity span representation metric concatenates with the first token representation to represent the whole entity.

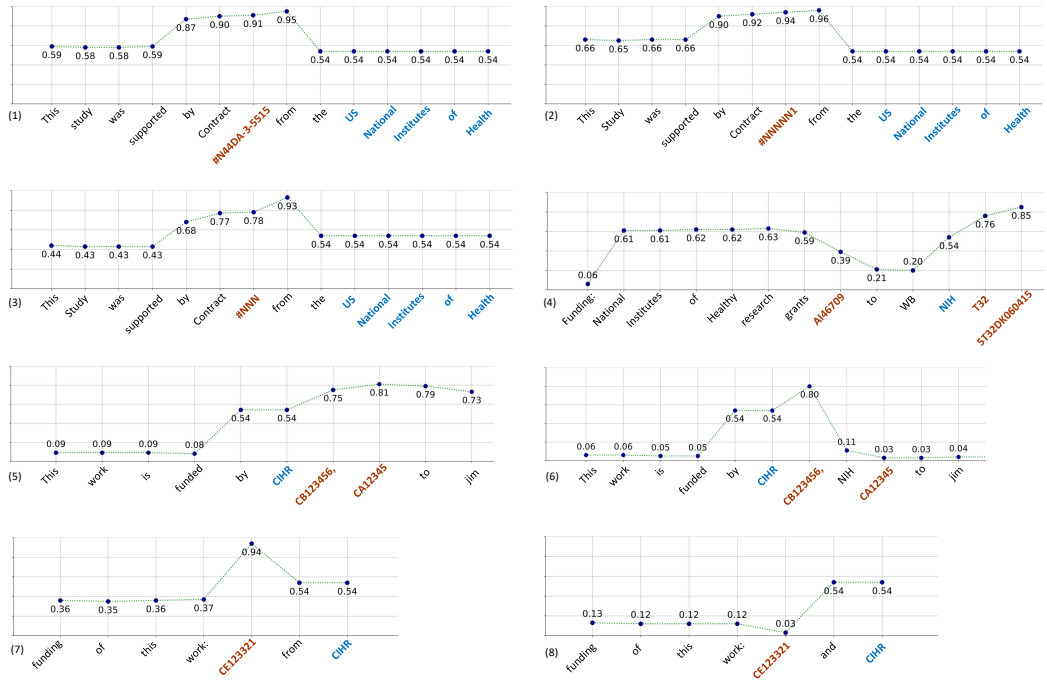

Figure 5: In each sentence, the blue-colored word is the selected funder entity, and the red-colored word is all grant numbers in the sentence. The float number on each word represents its alpha value when calculating the adaptive embedding under the blue-colored funder.

- **Head+Mean**: The average-pooling of the entity span representation metric concatenates with first token representation to represent the whole entity.
- **Head+Tail**: The first token representation concatenates the last token representation.
- **Max**: The max-pooling of the entity span representation metric.
- **Mean**: The average-pooling of the entity span representation metric.

It is observed that the average-pooling of the entity span has the best performance. Hence, we adopted this funder representation method in all our experiments.

C Performance Impact of the Adaptive Mechanism

Our adaptive embedding approach (GrantRel) were compared with the simple fuse approach (GrantRel pos+ctx), which merges both position embedding and context embedding by simply adding them. The results in Table 6 show GrantRel is slightly better.

As shown in Figure 5, we further analyze the impact of α on the embedding by using some chosen samples. For each sentence, given a funder entity being contained in this sentence, GrantRel calculated the value of α among all positions in Eq. (11).

Model name	Grant Realtion		
	Prec.	Rec.	F1
GrantRel pos+ctx	92.63	89.49	91.03
GrantRel	92.65	89.75	91.18

Table 6: Comparisons between GrantRel and GrantRel(pos+ctx) against the test set ofPubMed relation extraction .

For cases (1)-(4), we examine the impact of position embedding. As such, the outputs of GrantRel are compared with those of GrantRel(pos+ctx). In sentence (1), both GrantRel and GrantRel(pos+ctx) could recognize the grant number, but GrantRel(ctx) could not. Besides, we can see that the α value is high for grant number *"#N44DA-3-5515"*. In sentence 2, we manually built a case by replacing the GrantID with a more pseudo one. At a result, GrantRel still identified it as a grant number. But the GrantRel(pos+ctx) whose alpha value is always 0.5 did not recognize. In case (3), without Arabic chars in *"#NNN"*, GrantRel did not identify it as an ID even with a high α

value, either. We can conclude that if a token is close to the funder entity, and the alpha has a high value, the model tends to label an ID-like token into a GrantID. In case (4), GrantRel(pos-ctx) wrongly distributed "AI46706" to "NIH". In contrast, GrantRel assigned a low α value to "AI46706" according to its context of"to WB" and thus discarded this wrong relation.

In cases (5) to (8), we further explore the impact of different factors, which may influence the α value. For case (5) and case (6), the α values on grantID *"CA12345"* differ largely. But the only difference is that there is an agency of *"NIH"* in (6) between two IDs. We find that α dramatically decreases if the local context has other funder names. In cases (7) and (8), we find that some words can also reduce the α value except for funder names. Thus, the model should automatically pay more attention to context information. For example, the word "and" in (8) means that the previous grant information is parallel to the following grant information. Hence the model did not establish a connection between "CE123321" and "CIHR".