Leveraging Domain Agnostic and Specific Knowledge for Acronym Disambiguation

Qiwei Zhong, Guanxiong Zeng, Danqing Zhu, Yang Zhang, Wangli Lin, Ben Chen, Jiayu Tang
Alibaba Group, Hangzhou, China
{yunwei.zqw, moshi.zgx, danqing.zdq, zy142206, wangli.lwl, chenben.cb, jiayu.tangjy}@alibaba-inc.com

Abstract

An obstacle to scientific document understanding is the extensive use of acronyms which are shortened forms of long technical phrases. Acronym disambiguation aims to find the correct meaning of an ambiguous acronym in a given text. Recent efforts attempted to incorporate word embeddings and deep learning architectures, and achieved significant effects in this task. In general domains, kinds of fine-grained pretrained language models have sprung up, thanks to the large-scale corpora which can usually be obtained through crowdsourcing. However, these models based on domain agnostic knowledge might achieve insufficient performance when directly applied to the scientific domain. Moreover, obtaining large-scale high-quality annotated data and representing high-level semantics in the scientific domain is challenging and expensive. In this paper, we consider both the domain agnostic and specific knowledge, and propose a Hierarchical Dual-path BERT method coined **hdBERT** to capture the general fine-grained and high-level specific representations for acronym disambiguation. First, the context-based pretrained models, RoBERTa and SciBERT, are elaborately involved in encoding these two kinds of knowledge respectively. Second, multiple layer perceptron is devised to integrate the dual-path representations simultaneously and outputs the prediction. With a widely adopted SciAD dataset contained 62,441 sentences, we investigate the effectiveness of hdBERT. The experimental results exhibit that the proposed approach outperforms state-of-the-art methods among various evaluation metrics. Specifically, its macro F1 achieves 93.73%.

Introduction

In recent years, it has witnessed the vigorous development of deep learning. Among the most successful scenarios, natural language processing (NLP) is advancing steadily. However, natural language is frequently ambiguous, so many words and phrases can be interpreted in many ways depending on the context in which they appear (Navigli 2009). Specifically, an obstacle to scientific document understanding (SDU) is the widespread use of acronyms, which are shortened forms of long technical phrases (Veyseh et al. 2020b; Beltagy, Lo, and Cohan 2019). In order to understand the document correctly, the SDU system should be able to identify acronyms and their correct meanings. The goal of acronym disambiguation (AD) is to determine the correct long form of an ambiguous acronym in a given text (Veyseh et al. 2020a). It is usually formulated as a sequence classification problem in general (Veyseh et al. 2020b). For instance, a toy sample of this task is shown in Table 1. In this example, the "*CNN*" might be an acronym for "*Convolutional Neural Network*", "*Cable News Network*" or "*Condensed Nearest Neighbor*". Given a sentence "*They use **CNN** in the proposed model.*" and a dictionary with possible expansions (i.e., long forms) of the acronym "*CNN*", the expected prediction for its correct meaning is "*Convolutional Neural Network*". Recent efforts attempted to incorporate hand crafted features (Li et al. 2018), word embeddings (Charbonnier and Wartena 2018; Ciosici, Sommer, and Assent 2019), graph structures (Prokofyev et al. 2013; Veyseh et al. 2020b), and deep learning architectures (Jin, Liu, and Lu 2019; Blevins and Zettlemoyer 2020) and achieved significant effects in this task.

In this paper, we pay more attention to the scenario of scientific acronym disambiguation. Some observations are still worthy of further investigation. Generally, large-scale training data for natural language processing tasks in general domains is often possible to obtain through crowd-sourcing, emerging a variety of domain-independent fine-grained pretrained models. However, these models based on domain agnostic knowledge might achieve insufficient performance when applied to the specific domain (Beltagy, Lo, and Cohan 2019). Furthermore, obtaining large-scale annotated data in the scientific domain is challenging and expensive (Beltagy, Lo, and Cohan 2019), which leads to the shortage of high-level semantic expression to some extent.

To remedy these challenges, we fully consider both the domain agnostic and specific knowledge, and propose a Hierarchical Dual-path BERT method coined **hdBERT** to fusion the general fine-grained and high-level specific representations for acronym disambiguation. The overall architecture is illustrated in Figure 1. We pinpoint that hdBERT is a BERT-based supervised method adopting the now ubiquitous transformer architecture (Vaswani et al. 2017). First, RoBERTa (Liu et al. 2019) and SciBERT (Beltagy, Lo, and Cohan 2019) modules are elaborately involved to distill representations from inputs consist of sentence and candidate long forms. Specifically, we utilize RoBERTa, a ro-

Input - Sentence:	They use **CNN** in the proposed model.
Input - Dictionary:	CNN: 1. Convolutional Neural Network, 2. Cable News Network, 3. Condensed Nearest Neighbor
Output:	Convolutional Neural Network

Table 1: A toy sample of acronym disambiguation.

bustly optimized method trained on general domain corpora via byte-level Byte-Pair-Encoding (Sennrich, Haddow, and Birch 2016), to capture domain agnostic and fine-grained semantic information. Moreover, SciBERT which is also a pretrained language model based on BERT (Devlin et al. 2019) is exploited to model the high-level scientific domain representation. Since it leverages unsupervised pretraining on a large multi-domain corpus of scientific publications using WordPiece (Wu et al. 2016) tokenization strategy. Second, we integrate these dual-path representations from RoBERTa and SciBERT simultaneously via multiple layer perceptron and output the prediction. The main contributions of this work are summarized as follows:

- We are the very first attempt to resolve the acronym disambiguation problem simultaneously leveraging domain agnostic and specific knowledge.
- We propose a novel hierarchical dual-path BERT method coined hdBERT to capture both general fine-grained and high-level specific representations. It is mainly implemented based on the well-known transformer architecture, which can train the overall model more effectively.
- Experiments on real-world datasets demonstrate the effectiveness of the proposed approach. It achieves competitive performance and outperforms state-of-the-art methods.

Related Work

In this section, we review the related researches on word sense disambiguation especially acronym disambiguation as well as BERT and its two representative variants.

Word Sense Disambiguation

Word sense disambiguation (WSD) is an open problem concerned with identifying which sense of a word is used in a text (Navigli 2009). It is a core and difficulty in natural language processing tasks, which affects the performance of almost all downstream tasks. The methods to solve word sense disambiguation are usually divided into two categories: knowledge-based and supervised (Wang, Wang, and Fujita 2020; Barba et al. 2020).

Knowledge-based methods usually rely on amounts of statistical information and can be easily extended to other low-resource languages (Agirre, López de Lacalle, and Soroa 2014; Scarlini, Pasini, and Navigli 2020). For example, SensEmBERT (Scarlini, Pasini, and Navigli 2020), a knowledge- and BERT-based method that combines the expressive power of language modeling with the vast amount of knowledge contained in the semantic network, produces high-quality latent semantic representations of the meanings of the word in different languages. And it can achieve competitive results attained by most of the supervised neural approaches on the WSD tasks. On the other hand, supervised methods require lots of labeled data to learn word representations (Bevilacqua and Navigli 2020; Wang, Wang, and Fujita 2020). Of course, this defect can be alleviated through semi-supervised methods (Barba et al. 2020) by jointly leveraging contextualized word embedding and the multilingual information to project some sense labels.

Furthermore, acronym disambiguation is more challenging since we need to identify the acronym first and then to understand the text to determine the correct meaning of acronyms. Recently, an effective solution is to extract acronym definitions from unstructured texts by computing the Levenshtein string edit distance between any pair of long forms (Ciosici, Sommer, and Assent 2019), which is an entirely unsupervised acronym disambiguation method. And researches also attempt to incorporate hand crafted features (Li et al. 2018), word embeddings (Charbonnier and Wartena 2018; Ciosici, Sommer, and Assent 2019), graph structures (Prokofyev et al. 2013; Veyseh et al. 2020b), and deep learning architectures (Jin, Liu, and Lu 2019; Blevins and Zettlemoyer 2020), and have achieved significant effects in this task. Specifically, a supervised method named GAD (Veyseh et al. 2020b), which utilizes the syntactic structure of sentences to extend ambiguous acronyms in sentences by combining Bidirectional Long Short-Term Memory (BiLSTM) with Graph Convolutional Networks (GCN), provides a strong baseline on acronym disambiguation tasks in the scientific domain.

BERT-based Methods

Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al. 2019) is a self-supervised learning method that trains based on a large number of corpora to express better features for word embedding. And its network architecture utilizes the multi-layer transformer structure (Vaswani et al. 2017). The feature representation of BERT could be directly adopted as word embedding features for downstream tasks. Besides, BERT provides a model for transfer learning of other tasks. It can be fine-tuned or fixed according to tasks and then treated as a feature extractor. BERT was significantly undertrained, and there have been many fine-grained improvements or specific domain variants of it (Beltagy, Lo, and Cohan 2019; Liu et al. 2019; Scarlini, Pasini, and Navigli 2020; Lee et al. 2020).

RoBERTa. RoBERTa (Liu et al. 2019) is mainly trained on general domain corpora via byte-level Byte-Pair-Encoding (Sennrich, Haddow, and Birch 2016) based on the structure of BERT and can supply more fine-grained representation. This encoding scheme can process amounts of words that are common in natural language corpora and is

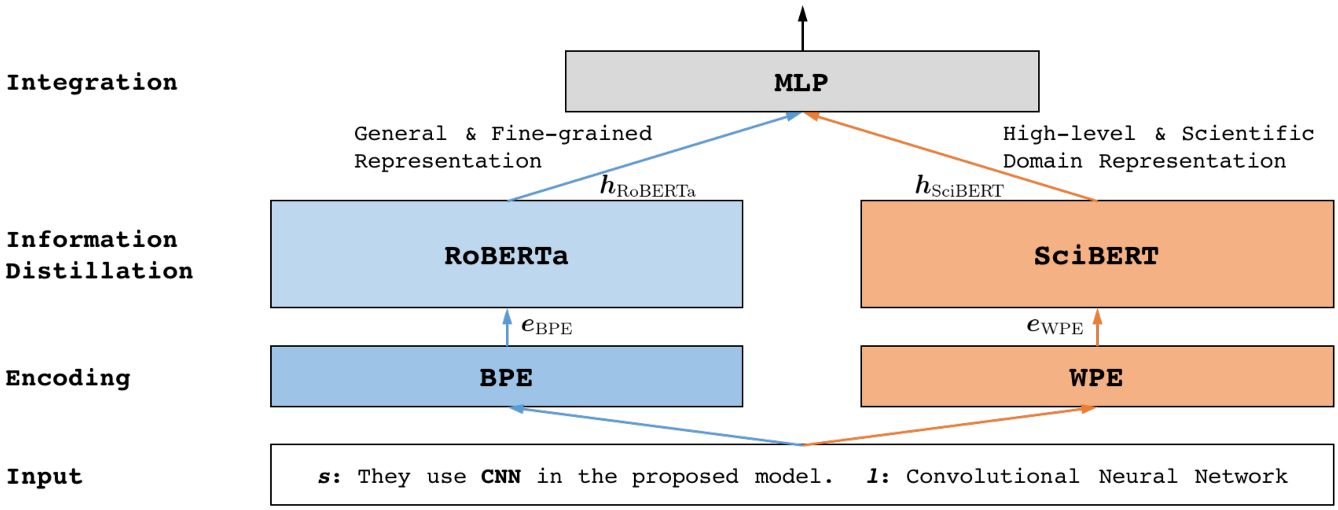
Integration
MLP
General & Fine-grained Representation
High-level & Scientific Domain Representation
h_{RoBERTa}
h_{SciBERT}
Information Distillation
RoBERTa
SciBERT
e_{BPE}
e_{WPE}
Encoding
BPE
WPE
Input
s: They use CNN in the proposed model.
l: Convolutional Neural Network

Figure 1: Illustration of the proposed hdBERT model.

more conducive to the translation of acronyms.

SciBERT. SciBERT (Beltagy, Lo, and Cohan 2019) is a specific pretrained language model for scientific domain texts. This model follows the same architecture as BERT to solve the lack of high-quality, large-scale labeled scientific data. It significantly outperforms previous BERT-based methods and achieves new state-of-the-art results on some scientific NLP tasks.

Methodology

In this section, we first introduce the problem statement of acronym disambiguation and then describe the overall architecture and details of our proposed hdBERT model.

Problem Statement

Acronym disambiguation is formulated as a sequence classification problem in general (Veyseh et al. 2020b). Formally, given an input sentence $\boldsymbol{s} = w_1, w_2, ..., w_n$ and the position of the acronym, i.e., p, the goal is to disambiguate the acronym w_p, that is, predicting the true long form $\boldsymbol{l}$ from all candidate long forms of w_p. Specifically, in this paper, we simplify it into a binary classification problem. That is, given an input sample consists of the sentence $\boldsymbol{s}$ with acronym w_p and the candidate long form $\boldsymbol{l}$, i.e., $\boldsymbol{x} = (\boldsymbol{s}; \boldsymbol{l})$, our purpose is to predict the probability of $\boldsymbol{l}$ being the right long form of w_p. We assign a label $y \in \{0, 1\}$ on each sample in training dataset to indicate whether $\boldsymbol{l}$ is a true long form of w_p in sentence $\boldsymbol{s}$ or not. In the testing phase, the long form with the highest prediction probability among the candidate long form set of a sentence would be chosen as its final result.

Overview

Figure 1 exhibits the schematic illustration of the proposed hdBERT model. As mentioned previously, we design a hierarchical integration model comprising three major components, each plays a different role in final prediction. The first two context-based components, i.e., RoBERTa (Liu et al. 2019) and SciBERT (Beltagy, Lo, and Cohan 2019) modules, distill representations of the sentence and the candidate long forms. Specifically, as a robustly optimized method trained on vast amounts of general domain corpora, we use RoBERTa to capture the general and fine-grained semantic information via byte-level Byte-Pair-Encoding (Sennrich, Haddow, and Birch 2016). Moreover, SciBERT, which leverages unsupervised pretraining on a large scientific corpus by WordPiece (Wu et al. 2016) tokenization strategy, is exploited to represent the high-level scientific domain information. Finally, a multiple layer perceptron network is devised to fusion these two kinds of representations. In the following, we present detail of each major component.

Information Distillation

General and Fine-grained Information. We involve RoBERTa to capture domain agnostic and fine-grained information of the sentence and its candidate long form. RoBERTa uses the now ubiquitous transformer architecture (Vaswani et al. 2017) via byte-level Byte-Pair-Encoding (BPE), which is a hybrid between character- and word-level representations that allow handling large vocabularies common in natural language corpora. Instead of full words, BPE relies on subwords units, which are extracted by performing statistical analysis of the training corpus. The size of the original vocabulary released with RoBERTa is about 50K, which is 20K more than BERT's.

We define the encoding of a sample $\boldsymbol{x} = (\boldsymbol{s}, \boldsymbol{l})$ after the BPE strategy as $\boldsymbol{e}_{\text{BPE}}$ and the output representation throughout the RoBERTa model as $\boldsymbol{h}_{\text{RoBERTa}}$.

$$\boldsymbol{e}_{\text{BPE}} = \textbf{BPE}(\boldsymbol{x}) \tag{1}$$

$$\boldsymbol{h}_{\text{RoBERTa}} = \textbf{RoBERTa}(\boldsymbol{e}_{\text{BPE}}) \tag{2}$$

High-level Scientific Domain Information. To handle the high-level scientific domain information, SciBERT is chosen elaborately. SciBERT follows the same architecture

Statistical Information	SciAD
number of acronyms	732
average number of long form per acronym	3.1
overlap between sentence and long forms	0.32
average sentence length	30.7
number of training	50,034
number of development	6,189
number of test	6,218

Table 2: The statistical information of original SciAD dataset. Note that the third row shows the ratio of sentences that have at least one word in common with the long forms of the acronyms appearing in the sentence.

as BERT but is instead pretrained on the scientific texts. It constructed a new WordPiece vocabulary on scientific corpus using the SentencePiece library and trained on a random sample of 1.14M papers from Semantic Scholar (Ammar et al. 2018). Its corpus consists of 18% papers from the computer science domain and 82% from the broad biomedical domain. The size of the original vocabulary released with SciBERT is about 30K, which is 20K less than RoBERTa. The resulting token overlap between SciBERT and BERT is 42%, which illustrates the significant difference in common terms between scientific and general domain texts.

We define the encoding of a sample $\boldsymbol{x} = (\boldsymbol{s}, \boldsymbol{l})$ after SciBERT's encoding strategy (noted as WPE) as $\boldsymbol{e}_{\text{WPE}}$ and the output representation throughout the SciBERT model as $\boldsymbol{h}_{\text{SciBERT}}$.

$$\boldsymbol{e}_{\text{WPE}} = \mathbf{WPE}(\boldsymbol{x}) \tag{3}$$

$$\boldsymbol{h}_{\text{SciBERT}} = \mathbf{SciBERT}(\boldsymbol{e}_{\text{WPE}}) \tag{4}$$

Integration

After modeling the two complex representations above, the obtained concatenation $\boldsymbol{h}$ is fed into multiple layer perceptron network and followed by a regression layer with sigmoid unit, as follows:

$$\boldsymbol{h} = [\boldsymbol{h}_{\text{RoBERTa}}; \boldsymbol{h}_{\text{SciBERT}}] \tag{5}$$

$$p = \text{sigmoid}(\boldsymbol{W}^{\text{T}}\mathbf{MLP}(\boldsymbol{h}) + b) \tag{6}$$

where $\boldsymbol{W}$ is the weight vector, b is the bias, and $\mathbf{MLP}(\cdot)$ represents the operation of multiple layer perceptron shown in Figure 1. Here p is the predicted probability.

Finally, our model is trained with cross entropy loss with regularization. The loss function is defined as

$$\mathcal{L}(\theta) = -\sum_{\mathcal{D}} (y \log(p) + (1-y)\log(1-p)) + \lambda \|\boldsymbol{\theta}\|_2^2 \tag{7}$$

where y is the ground truth, $\boldsymbol{\theta}$ is the parameter set of the proposed model, λ is the regularizer parameter, and $\mathcal{D}$ is the training dataset.

Experiments

In this section, we first illustrate the datasets, evaluation metrics, and implementation details, then demonstrate the experimental results and further studies.

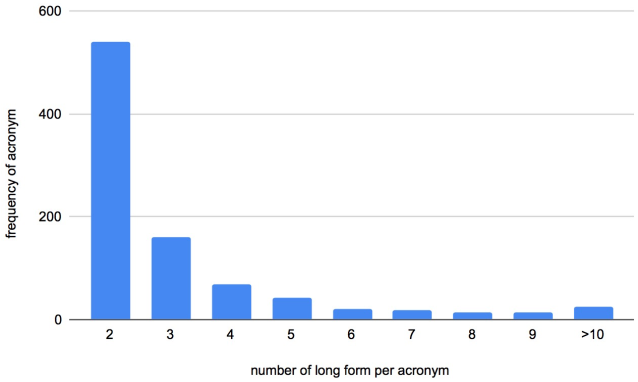

Figure 2: Distribution of acronyms based on number of long form per acronym.

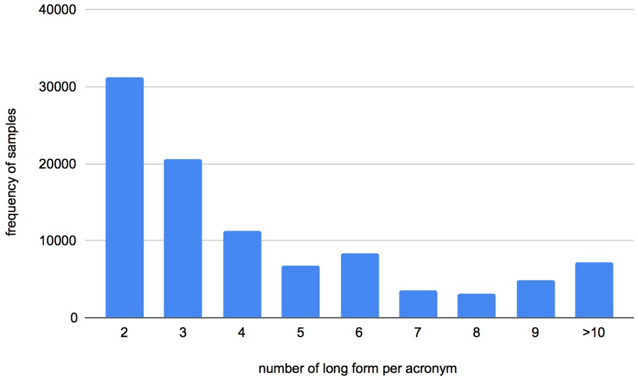

Figure 3: Distribution of samples based on number of long form per acronym.

Datasets

The SciAD [1] dataset created from 6,786 English scientific papers aims to find the correct meaning of an ambiguous acronym in a given sentence (Veyseh et al. 2020b). It contains 62,441 sentences and a dictionary of 732 ambiguous acronyms. More statistical information is shown in Table 2. Besides, a toy sample of the SciAD dataset is shown in Table 1. The input is a sentence with an ambiguous acronym and a dictionary with possible expansions (i.e., long forms) of the acronym. In this example, the ambiguous acronym "*CNN*" in the input sentence is shown in boldface and the expected prediction for its correct meaning is "*Convolutional Neural Network*". In addition, Figures 2 and 3 demonstrate more statistics of SciAD dataset (Veyseh et al. 2020b). More specifically, Figure 2 shows the distribution of the number of acronyms based on the number of long forms per acronym, and the distribution of the number of samples based on the number of long form per acronym is shown in Figure 3.

As mentioned previously, we convert the original SciAD dataset into a binary classification dataset named SciAD$_{\text{BI}}$ during modeling. For a sentence $\boldsymbol{s}$ with acronym w_p, $y = 1$

[1] We won second place in the acronym disambiguation competition. https://sites.google.com/view/sdu-aaai21/shared-task

Parameter	BERT	RoBERTa	SciBERT
pretrained model	bert-large-ucased[a]	roberta-large[b]	allenai/scibert_scivocab_uncased[c]
architecture	sequence classification	sequence classification	sequence classification
attention_probs_dropout_prob	0.1	0.1	0.1
hidden_act	gelu	gelu	gelu
hidden_dropout_prob	0.1	0.1	0.1
hidden_size	1024	1024	768
initializer_range	0.02	0.02	0.02
intermediate_size	4096	4096	3072
layer_norm_eps	1e-12	1e-05	1e-12
max_position_embeddings	512	514	512
model_type	bert	roberta	bert
num_attention_heads	16	16	12
num_hidden_layers	24	24	12
position_embedding_type	absolute	absolute	absolute
vocab_size	30522	50265	31090
learning_rate	2e-5	2e-5	2e-5
epoch	5	5	5

[a] https://huggingface.co/bert-large-uncased
[b] https://huggingface.co/roberta-large
[c] https://github.com/allenai/scibert

Table 3: Architecture and hyper parameters information. Our proposed hdBERT model ensembles RoBERTa and SciBERT via three MLP layers.

Statistical Information	SciAD$_{\text{BI}}$
number of training	352,366
number of development	28,286
number of test	28,364

Table 4: The statistical information of SciAD$_{\text{BI}}$ dataset.

if a long form $\boldsymbol{l}$ is true for w_p, while $y = 0$ for other false candidate long forms of w_p. Specifically, to alleviate the imbalance problem during training, we upsample each positive sample to equal the number of candidate long forms of its acronym. More statistics of SciAD$_{\text{BI}}$ is shown in Table 4. We finally evaluate performances on SciAD's test dataset.

Compared Methods

We compare with several state-of-the-art and representative methods including Non-deep learning methods and Deep learning methods to verify the effectiveness of our proposed method.

Non-deep learning methods.

- **MF**: most frequent which takes the long form with the highest frequency among all possible meanings of an acronym as the expanded form of the acronym.
- **ADE** (Li et al. 2018): a feature-based model that employs hand crafted features from the context of the acronyms to train a disambiguation classifier.

Deep learning methods.

- **NOA** (Charbonnier and Wartena 2018) and **UAD** (Ciosici, Sommer, and Assent 2019): language-model-based baselines that train the word embeddings using the training corpus.
- **DECBAE** (Jin, Liu, and Lu 2019) and **BEM** (Blevins and Zettlemoyer 2020): models employing deep architectures (e.g., LSTM).
- **GAD** (Veyseh et al. 2020b): supervised method which utilizes syntactic structure of sentences to extend ambiguous acronyms in sentences by combining BiLSTM with GCN.
- **BERT** (Devlin et al. 2019), **RoBERTa** (Liu et al. 2019) and **SciBERT** (Beltagy, Lo, and Cohan 2019): pretrained models use the now ubiquitous transformer architecture.

Evaluation Metrics

To evaluate the performance of different methods, three popular metrics are adopted, namely **Macro Precision**, **Macro Recall** and **Macro F1**. The definitions are as follows:

$$\textbf{Precision}_{\text{MACRO}} = \frac{\sum_{i=1}^{n} \text{Precision}_i}{n} \tag{8}$$

$$\textbf{Recall}_{\text{MACRO}} = \frac{\sum_{i=1}^{n} \text{Recall}_i}{n} \tag{9}$$

$$\textbf{F1}_{\text{MACRO}} = \frac{2 \times \text{Precision}_{\text{MACRO}} \times \text{Recall}_{\text{MACRO}}}{\text{Precision}_{\text{MACRO}} + \text{Recall}_{\text{MACRO}}} \tag{10}$$

where n is the number of total classes, Precision_i and Recall_i represent the precision and recall of class i respectively. The higher $\text{Precision}_{\text{MACRO}}$, $\text{Recall}_{\text{MACRO}}$ and F1_{MACRO} indicate the higher performance of approaches.

Methodology	**Macro Precision**(%)	**Macro Recall**(%)	**Macro F1**(%)
MF	89.03	42.20	57.26
ADE (Li et al. 2018)	86.74	43.25	57.72
NOA (Charbonnier and Wartena 2018)	78.14	35.06	48.40
UAD (Ciosici, Sommer, and Assent 2019)	89.01	70.08	78.37
BEM (Blevins and Zettlemoyer 2020)	86.75	35.94	50.82
DECBAE (Jin, Liu, and Lu 2019)	88.67	74.32	80.86
GAD (Veyseh et al. 2020b)	89.27	76.66	81.90
Human Performance (Veyseh et al. 2020b)	97.82	94.45	96.10
MF	89.00	46.36	60.97
BERT (Devlin et al. 2019)	95.26	86.92	90.90
RoBERTa (Liu et al. 2019)	95.96	88.36	92.00
SciBERT (Beltagy, Lo, and Cohan 2019)	96.36	89.77	92.95
hdBERT (ours)	**96.94**	**90.73**	**93.73**

Table 5: Performance of models in acronym disambiguation.

Implementation Details

For models ADE, NOA, UAD, DECBAE, BEM, and GAD, please refer to Veyseh et al. for more implementation information. We implement the proposed model based on Pytorch (Paszke et al. 2019) and Transformers (Wolf et al. 2020). For models BERT, RoBERTa, and SciBERT, we finetune them on dataset based on their popular pretrained models. The implementation details of these models are shown in Table 3. Moreover, the information distillation components of our model are the same as model RoBERTa and SciBERT respectively. And we simply adopt three MLP layers for integration simultaneously. As mentioned previously, in the testing phase, the long form with the highest prediction probability in the candidate long form set of a sentence would be chosen as its final result. In addition, we use two V100 GPUs with 12 cores to complete all these experiments.

Performance Comparison

Table 5 demonstrates the main results of all compared methods [2] on the dataset. The major findings from the experimental results can be summarized as follows:

First, GAD achieves a better result than methods such as ADE, NOA, UAD, BEM, and DECBAE, showing the importance of syntactic structure for the acronym disambiguation task. But it still far worse than pretraining-based models like BERT and RoBERTa. Second, between the two general-domain models, RoBERTa gets better performance than BERT, indicating the advantage of more fine-grained encoding. Moreover, SciBERT is more advanced than the domain agnostic methods, i.e., BERT and RoBERTa, with about 2.26% and 1.03% increased macro F1 respectively, showing the importance of the scientific domain pretraining for this task. Furthermore, we can clearly observe that our hdBERT model outperforms all the baselines by a large margin. Its macro F1, with the reported value of 93.73%, is about 1.88% and 0.84% higher than state-of-the-art RoBERTa and SciBERT respectively. And its loss curve falls faster and converges lower than the two pretrained methods on the development dataset, as shown in Figure 4. These observations demonstrate that it is effective to model both fine-grained domain agnostic and high-level domain specific knowledge simultaneously.

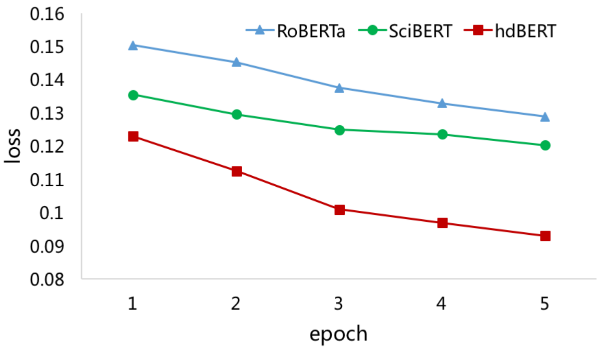

Figure 4: Loss curve on development dataset.

However, despite the significant improvements among these approaches, performances of all models are still not as effective as humans on the dataset, especially on macro recall and macro F1, thus providing many further research opportunities for this scenario.

Case Study

We further focus on studying both success and failure cases of pretraining-based models to provide more insight into acronym disambiguation. Specifically, for success case of our model in which RoBERTa and SciBERT fail, e.g., "*Each SP within an **SM** shares an instruction unit, dedicated to the management of the instruction flow of the threads.*" (DEV-6156), the true long form of "*SM*" is "*Streaming Multiprocessors*". While both RoBERTa and SciBERT output "*Shared Memory*", which may often appear in deep learning publications. It might benefit from the additional integration modeling of two different information from RoBERTa and SciBERT. However, all the three models fail in this exam-

[2] We assume that both Veyseh et al. and this task have the same distribution of dataset due to the randomly dividing by the same ratio, making all these methods comparable.

Sentence	Conflicted Annotation
Just like **RF**, QRF is a set of binary regression trees.	TR-43200: Regression Forest TR-49535: Regression Function
Extensions of the **SBM** regarding the type of graph are reviewed in Section.	TR-17276: Sequential Monte Carlo TR-47761: Stochastic Block Model
The obfuscated term is the term for which the **MACS** score is the lowest.	TR-15480: Mean Average Conceptual Similarity TR-27970: Minimum Average Conceptual Similarity

Table 6: Examples of noise data of SciAD dataset.

ple: "*In the first stage, we train the SPM, and extract the **FL** and FR.*" (DEV-4604) with the wrong prediction "*Federated Learning*" for "*FL*". The true long form of "*FL*" is "*Fixated Locations*". We guess that all models pay too much attention to "*Federated Learning*", a hot phrase nowadays, and ignore the subtle information among the sentence and its different candidate long forms. It also indicates the necessity of more advanced models for this task.

Further Discussion

As mentioned previously and shown in Table 5, all the current models are still less effective than humans in this scenario. There are still many samples that all models fail in. Some further research opportunities on this dataset are discussed in this section. First, as shown in Table 6, there are some noise data, i.e., conflicted annotation, in the SciAD dataset. For example, the acronym "*RF*" in boldface in sentence "*Just like **RF**, QRF is a set of binary regression trees.*" gets two different long form "*Regression Forest*" (TR-43200) and "*Regression Function*" (TR-49535) respectively. It will be some negative impacts on modeling to some extent. Furthermore, to a certain extent, samples constructed from the same sentence with different long forms are independent during our training stage. It might lose more subtle information among them. Therefore, recent methods such as self-training (Peng et al. 2019; Chi et al. 2020), adversarial learning (Goodfellow, Shlens, and Szegedy 2015; Miyato, Dai, and Goodfellow 2017; Zhu et al. 2021), and contrastive learning (Hadsell, Chopra, and LeCun 2006) are worth studying to further improve the performance.

Conclusions

An obstacle to scientific document understanding is the widespread use of acronyms which are shortened forms of long technical phrases. Acronym disambiguation aims to find the correct meaning of an ambiguous acronym in a given text. However, it is challenging and expensive to obtain large-scale high-quality annotated data in the scientific domain. In this paper, we present a hierarchical dual-path BERT method coined hdBERT for acronym disambiguation to resolve the special challenges in this scenario. The method is equipped with pretrained models RoBERTa and SciBERT and integrates their dual-path representations simultaneously to leveraging domain agnostic and specific knowledge. Experiments on real-world datasets demonstrate the effectiveness of the proposed approach. It achieves competitive performance and outperforms state-of-the-art methods among various evaluation metrics. Moreover, there are still many research opportunities in this task, approaches such as self-training, adversarial learning, and contrastive learning are worth studying to further improve the performance.

Acknowledgments

We thank the organizers of acronym identification and disambiguation competitions and the reviewers for their valuable comments and suggestions.

References

Agirre, E.; López de Lacalle, O.; and Soroa, A. 2014. Random Walks for Knowledge-based Word Sense Disambiguation. *Computational Linguistics* 40(1): 57–84.

Ammar, W.; Groeneveld, D.; Bhagavatula, C.; Beltagy, I.; Crawford, M.; Downey, D.; Dunkelberger, J.; Elgohary, A.; Feldman, S.; Ha, V.; et al. 2018. Construction of the Literature Graph in Semantic Scholar. In *NAACL*, 84–91.

Barba, E.; Procopio, L.; Campolungo, N.; Pasini, T.; and Navigli, R. 2020. MuLaN: Multilingual Label Propagation for Word Sense Disambiguation. In *IJCAI*, 3837–3844.

Beltagy, I.; Lo, K.; and Cohan, A. 2019. SciBERT: A Pretrained Language Model for Scientific Text. In *EMNLP*, 3606–3611.

Bevilacqua, M.; and Navigli, R. 2020. Breaking Through the 80% Glass Ceiling: Raising the State of the Art in Word Sense Disambiguation by Incorporating Knowledge Graph Information. In *ACL*, 2854–2864.

Blevins, T.; and Zettlemoyer, L. 2020. Moving Down the Long Tail of Word Sense Disambiguation with Gloss-Informed Biencoders. *arXiv preprint arXiv:2005.02590* .

Charbonnier, J.; and Wartena, C. 2018. Using Word Embeddings for Unsupervised Acronym Disambiguation. In *COLING*, 2610–2619.

Chi, J.; Zeng, G.; Zhong, Q.; Liang, T.; Feng, J.; Ao, X.; and Tang, J. 2020. Learning to Undersampling for Class Imbalanced Credit Risk Forecasting. In *ICDM*.

Ciosici, M.; Sommer, T.; and Assent, I. 2019. Unsupervised Abbreviation Disambiguation Contextual Disambiguation using Word Embeddings. *arXiv preprint arXiv:1904.00929* .

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*, 4171–4186.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2015. Explaining and Harnessing Adversarial Examples. In *ICLR*.

Hadsell, R.; Chopra, S.; and LeCun, Y. 2006. Dimensionality Reduction by Learning an Invariant Mapping. In *CVPR*, volume 2, 1735–1742.

Jin, Q.; Liu, J.; and Lu, X. 2019. Deep Contextualized Biomedical Abbreviation Expansion. In *BioNLP Workshop*, 88–96.

Lee, J.; Yoon, W.; Kim, S.; Kim, D.; Kim, S.; So, C. H.; and Kang, J. 2020. BioBERT: A Pre-trained Biomedical Language Representation Model for Biomedical Text Mining. *Bioinformatics* 36(4): 1234–1240.

Li, Y.; Zhao, B.; Fuxman, A.; and Tao, F. 2018. Guess Me if You Can: Acronym Disambiguation for Enterprises. In *ACL*, 1308–1317.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692* .

Miyato, T.; Dai, A. M.; and Goodfellow, I. 2017. Adversarial Training Methods for Semi-supervised Text Classification. In *ICLR*.

Navigli, R. 2009. Word Sense Disambiguation: A Survey. *ACM computing surveys (CSUR)* 41(2): 1–69.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An Imperative Style, High-performance Deep Learning Library. In *NIPS*, 8026–8037.

Peng, M.; Zhang, Q.; Xing, X.; Gui, T.; Huang, X.; Jiang, Y.-G.; Ding, K.; and Chen, Z. 2019. Trainable Undersampling for Class-imbalance Learning. In *AAAI*, volume 33, 4707–4714.

Prokofyev, R.; Demartini, G.; Boyarsky, A.; Ruchayskiy, O.; and Cudré-Mauroux, P. 2013. Ontology-based Word Sense Disambiguation for Scientific Literature. In *ECIR*, 594–605.

Scarlini, B.; Pasini, T.; and Navigli, R. 2020. SensEmBERT: Context-Enhanced Sense Embeddings for Multilingual Word Sense Disambiguation. In *AAAI*, 8758–8765.

Sennrich, R.; Haddow, B.; and Birch, A. 2016. Neural Machine Translation of Rare Words with Subword Units. In *ACL*, 1715–1725.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is All you Need. In *NIPS*, 5998–6008.

Veyseh, A. P. B.; Dernoncourt, F.; Nguyen, T. H.; Chang, W.; and Celi, L. A. 2020a. Acronym Identification and Disambiguation shared tasks for Scientific Document Understanding. In *AAAI Workshop on Scientific Document Understanding*.

Veyseh, A. P. B.; Dernoncourt, F.; Tran, Q. H.; and Nguyen, T. H. 2020b. What Does This Acronym Mean? Introducing a New Dataset for Acronym Identification and Disambiguation. In *COLING*, 3285–3301.

Wang, Y.; Wang, M.; and Fujita, H. 2020. Word Sense Disambiguation: A Comprehensive Knowledge Exploitation Framework. *Knowledge-Based Systems* 190: 105030.

Wolf, T.; Chaumond, J.; Debut, L.; Sanh, V.; Delangue, C.; Moi, A.; Cistac, P.; Funtowicz, M.; Davison, J.; Shleifer, S.; et al. 2020. Transformers: State-of-the-art Natural Language Processing. In *EMNLP*, 38–45.

Wu, Y.; Schuster, M.; Chen, Z.; Le, Q. V.; Norouzi, M.; Macherey, W.; Krikun, M.; Cao, Y.; Gao, Q.; Macherey, K.; et al. 2016. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. *arXiv preprint arXiv:1609.08144* .

Zhu, D.; Lin, W.; Zhang, Y.; Zhong, Q.; Zeng, G.; Wu, W.; and Tang, J. 2021. AT-BERT: Adversarial Training BERT for Acronym Identification. In *AAAI Workshop on Scientific Document Understanding*.