CovRelex-SE: Adding Semantic Information for Relation Search via Sequence Embedding

Dinh-Truong Do[1] **Chau Nguyen**[1] **Vu Tran**[2] **Ken Satoh**[3]
Yuji Matsumoto[4] **Minh Le Nguyen**[1]

[1]Japan Advanced Institute of Science and Technology
{truongdo,chau.nguyen,nguyenml}@jaist.ac.jp
[2]The Institute of Statistical Mathematics, Japan
vutran@ism.ac.jp
[3]National Institute of Informatics, Japan
ksatoh@nii.ac.jp
[4]RIKEN Center for Advanced Intelligence Project (AIP), Japan
yuji.matsumoto@riken.jp

Abstract

In recent years, COVID-19 has impacted all aspects of human life. As a result, numerous publications relating to this disease have been issued. Due to the massive volume of publications, some retrieval systems have been developed to provide researchers with useful information. In these systems, lexical searching methods are widely used, which raises many issues related to acronyms, synonyms, and rare keywords. In this paper, we present a hybrid relation retrieval system, CovRelex-SE, based on embeddings to provide high-quality search results. Our system can be accessed through the following URL: https://www.jaist.ac.jp/is/labs/nguyen-lab/systems/covrelex-se/.

Keywords: COVID-19, relation search, biomedical domain, relation extraction, entity recognition, semantic search.

1 Introduction

Scientific information related to the coronavirus disease has received a lot of attention in recent years. The number of COVID-19-relevant publications is increasing daily. In the record of CORD-19 dataset (Wang et al., 2020a), there are more than 900K papers introduced by March 31^{st}, 2022. The huge number of documents demonstrates the importance of retrieval systems for providing researchers with informative knowledge.

A relation is an object that consists of three components ($arg1$, rel, $arg2$), where $arg1$, and $arg2$ are noun phrases that may contain biomedical entities and rel is an expression describing the relation between $arg1$ and $arg2$. A query is made up of partial information on a relation, which includes keywords regarding these components. Ideally, a relation retrieval system should return all relevant relations with the corresponding papers, which can be used to answer two different types of questions: single-hop and multi-hop. Regarding single-hop questions, such as *"COVID-19 disables which things?"*, we can input the query *("COVID-19", "disable", any-arg2)*, and then extract the answer by using the returned results of *arg2*. On the other hand, we can combine two queries: *("COVID-19", "cause", DISEASE)*, and *(CHEMICAL, "treat", DISEASE)* to answer the multi-hop question *"What are CHEMICAL that can treat some DISEASES caused by COVID-19?"*. By that, the answer can be extracted by using the returned results at the position of *CHEMICAL*.

In this paper, we propose CovRelex-SE, a hybrid retrieval system to search the relations that effectively tackles the issues raised by the lexical approach in the CovRelex system (Tran et al., 2021). Instead of searching relations using lexical methods, CovRelex-SE ranks their scores by utilizing the combination of lexical scores (based on the Elasticsearch[1] engine) and semantic scores (based on CORD19-BERT embeddings). In summary, our contributions in this paper are as follows: **(I)** A novel approach to ranking COVID-19-relevant relations, which combines the effectiveness of lexical approach and vector representation approach; **(II)** A new pre-trained language model,

[1]https://www.elastic.co/

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics
System Demonstrations, pages 35–42

CORD19-BERT, which is pre-trained from scratch using the CORD-19 dataset; **(III)** A web-based relation search system, CovRelex-SE, which provides two search functions: Single-Relation Search and Graph Search; that aims to answer two type of questions: single-hop and multi-hop; **(IV)** An experimental evaluation, which shows the superior performance of CovRelex-SE system using the CORD-19 dataset by March 31^{st}, 2022.

2 Related Work

Due to the COVID-19 outbreak, it is vital to collect crucial information from a huge number of COVID-19-related publications. Zhang et al. (2020) created Covidex, a search engine that allows users to query the COVID-19 Open Research Dataset and access inside information. Esteva et al. (2020) introduced Co-Search, a semantic search engine composed of a retriever and a ranker that was built to handle complex queries throughout the COVID-19 papers. Additionally, Wang et al. (2020b) created the EvidenceMiner web-based solution. Given a query as a natural language statement, EvidenceMiner retrieves textual evidence at the sentence level from the CORD-19 corpus for life sciences. More recently, Raza et al. (2022) present an Information Retrieval System that uses latent information to select relevant works related to specific concepts. Otegi et al. (2022) develop a Question Answering system that receives a set of questions asked by experts about the disease COVID-19 and SARS-CoV-2 virus, and provides a ranked list of expert-level answers to each question.

Conceptually, the most similar to our work, CovRelex (Tran et al., 2021), is a retrieval system for scientific publications that target entities and relations via relation extraction from COVID-19 scientific papers. However, there is still a lack of systems that automatically extract the diverse relations through papers and obtain the results using semantic information, especially given the rapid publication of COVID-19 papers. This issue motivates us to create the CovRelex-SE system.

3 Method

3.1 Overview

Figure 1 illustrates our proposed system, CovRelex-SE. From the raw text of document abstracts, we extract relations and recognize biomedical entities inside the extracted relations. For each relation, $arg1$, $arg2$, and rel are converted into vectors by using CORD19-BERT. Three Faiss (Johnson et al., 2019) indices are then trained using all of the embedding vectors. At the query time, the user input a query, which will be converted to embedding vectors. Following that, the Faiss indices will be looked up for the most similar relations according to the query and return semantic scores. The Elasticsearch engine will also look up the query and utilize the BM25 algorithm (Robertson et al., 1995) to calculate lexical scores. The system then combines the lexical scores and semantic scores as final scores for relations. Finally, CovRelex-SE returns the top-ranked triplets after filtering query entities using the Elasticsearch engine.

3.2 Relation Extraction & Entity Recognition

In this paper, to extract the relations in the documents as many as possible, we use a variety of relation extraction methods. As each method has its own characteristics, we can obtain more unique relations when combining all of them. The following are brief descriptions of the methods.

- **ReVerb** (Fader et al., 2011) tackles the issues of incoherent and uninformative relation extractions by introducing syntactic and lexical constraints on binary verb-based relations.
- **OLLIE** (Schmitz et al., 2012) overcomes the limitation of prior methods, which extract only relations mediated via verbs. OLLIE broadens the syntactic scope by identifying relations mediated by nouns, adjectives, etc.
- **ClausIE** (Del Corro and Gemulla, 2013) is a clause-based approach to open information extraction. It separates the detection of clauses and clause types from the actual generation of propositions.
- **Relink** (Tran and Nguyen, 2021) is a method inherited partly from ReVerb. It extracts relations from connected phrases, unlike ClauseIE which extracts clause types.
- **OpenIE** (Angeli et al., 2015) breaks a long sentence into short, coherent clauses, and then finds the maximally simple relations.

After extracting relations, we use the SpaCy[2] models provided by the SciSpacy (Neumann et al., 2019) library to recognize the biomedical entities.

[2] https://spacy.io/

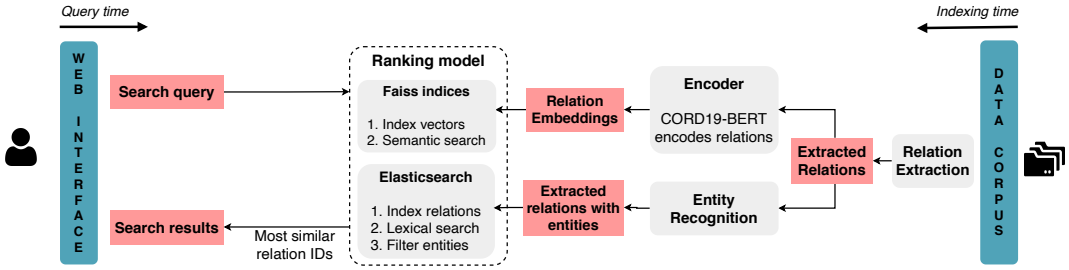


Figure 1: Overview of the CovRelex-SE system.

Since each model is trained on a different annotated corpus (Li et al., 2016; Bada et al., 2012; Kim et al., 2004; Pyysalo et al., 2015), it can recognize a different set of biomedical entities. Table 1 shows SciSpacy models that were utilized.

Table 1: SciSpacy models used in our system.

Models	Training corpus
en_ner_craft_md	CRAFT corpus (for cell types, chemicals, proteins, genes) (Bada et al., 2012)
en_ner_jnlpba_md	JNLPBA corpus (for cell lines, cell types, DNAs, RNAs, proteins) (Collier and Kim, 2004)
en_ner_bc5cdr_md	BC5CDR corpus (for chemicals and diseases) (Li et al., 2016)
en_ner_bionlp13cg_md	BioNLP13CG (for cancer genetics) (Pyysalo et al., 2015)

3.3 Embedding Extraction & Faiss Index

In recent years, domain-specific pre-trained models have led to effective results on many natural language processing tasks (Chalkidis et al., 2020; Lee et al., 2020). Generally, there are two common ways to pre-train a domain-specific language model: from scratch or continual over a general language model such as BERT (Devlin et al., 2018). However, Gu et al. (2020) show that if we have a large enough training data, pre-training from scratch would be better. The particular reasons for this circumstance are as follows: (**I**) The ability to develop a new vocabulary for the specific domain, (**II**) The fact that general documents basically differ from documents of this domain, increasing the likelihood of negative transfers that reduce the overall performance.

Based on the above points, we pre-train a new language model, CORD19-BERT, from scratch using the data of CORD-19 corpus. Figure 2 illustrates the relative coverages of the vocabularies of three models CORD19-BERT, PubMedBERT (Gu et al., 2020) and BERT-base (Devlin et al., 2018). There is a considerable variation in the three vocabularies. Especially, there are some common COVID-19 related words that do not exist in the vocabularies of BERT-base and PubMedBERT, such as *covid, unvaccinated, etc*. In this step, we use the masked language model task to pre-train CORD19-BERT. Following BERT, we mask 15% of tokens, and the model needs to predict the masked tokens in the sentence. We share our pre-trained CORD19-BERT model via Huggingface[3].

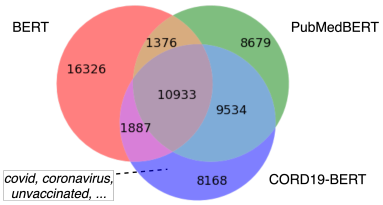


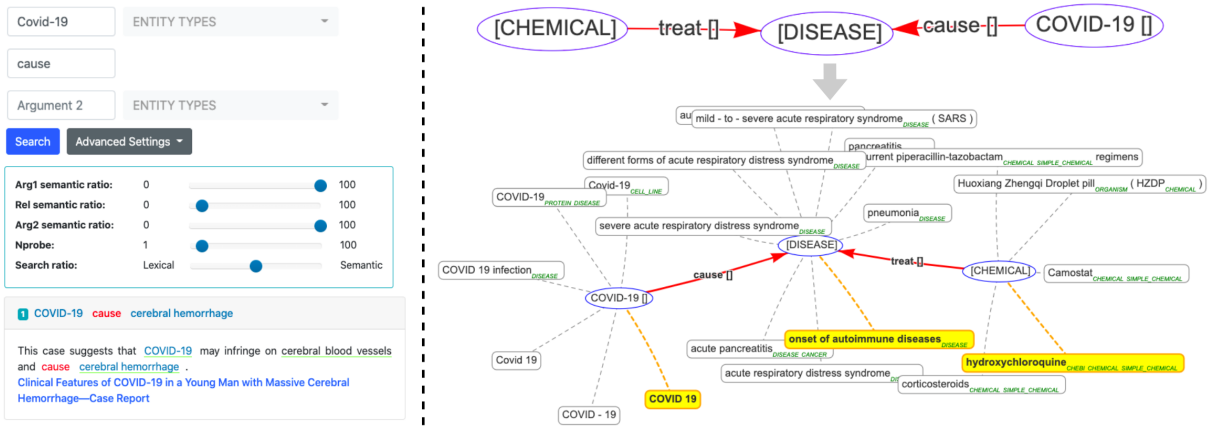
Figure 2: The relative coverages of three vocabularies of BERT, PubMedBERT, and CORD19-BERT. When the vocabulary size of each model is 30,522 tokens.

After pre-training the model, we use CORD19-BERT to extract the embeddings of relations. One issue with data processing is the excessive number of embedding vectors. Therefore, we used the Faiss index (Johnson et al., 2019) to resolve this problem. Faiss is a method for searching and grouping dense vectors in an efficient manner. More details about the Faiss settings used in this paper are shown in the experimental section.

3.4 Relation Scoring

The score of a relation ($arg1$, rel, $arg2$) is calculated based on semantic and lexical scores. The semantic score is determined using the embeddings from CORD19-BERT, whereas the lexical score is computed using the Elasticsearch engine. Specifically, let (*s_arg1*, *s_rel*, *s_arg2*) be the semantic

[3] https://huggingface.co/CovRelex-SE/CORD19-BERT



(a) Single-Relation Search (b) Graph Search

Figure 3: Examples of Single-Relation Search and Graph Search.

scores of (*ar1*, *rel*, *arg2*) calculated by Faiss indices based on squared Euclidean (L2) distance. Following that, the semantic score of a relation is calculated based on formula 1 with hyperparameters α, β, and γ. These parameters' values can be controlled by the users.

$$score_{se} = \alpha * s\_arg1 + \beta * s\_rel + \gamma * s\_arg2 \quad (1)$$

The Elasticsearch engine, which is based on the BM25 scoring algorithm, is used to compute the lexical score. After obtaining the semantic and lexical scores, the final score is calculated by combining these two scores using the formula 2. Additionally, there is a user-specified hyperparameter with θ, depending on whether the user wants to search exact match or by semantic similarity.

$$score_{final} = \theta * score_{se} + (1 - \theta) * score_{lex} \quad (2)$$

3.5 Retrieval System

The retrieval system provides two different types of searching scenarios: **Single-Relation Search** and **Graph Search**. While **Single-Relation Search** provides a simple way to discover a specific relation, **Graph Search** aims to answer complex questions from users.

3.5.1 Single-Relation Search

In **Single-relation Search**, a query consists of partial information of a relation which can contain keywords about $arg1$, $arg2$, and rel, and the sliders which determine the values of the hyperparameters in formulas 1 and 2. The retrieved results are relevant relations along with their corresponding papers. An example of a single-relation query is illustrated in Fig. 3a. The query relation is (*"Covid-19", "cause", ""*). The results are highest score relations, for instance, (*"COVID-19", "cause", "cerebral hemorrhage"*).

3.5.2 Graph Search

In addition to single relation searching, we provide a multi-relation search tool called **Graph Search**. The input graph is a directed graph where each edge indicates a relation, and the label of the edge is determined by the value of *rel*. Each edge contains *arg1* and *arg2* as its source and target. The retrieved result is a graph that matches the query graph. The main purpose of **Graph Search** is to find out the answer for complex questions that are challenging to answer with single search searching. For example, the question *"What CHEMICAL can treat some DISEASE caused by COVID-19?"* can be represented as a graph with three nodes and two edges defining two relations *("COVID-19", "cause", DISEASE)*, and *(CHEMICAL, "treat", DISEASE)*. This allows us to perform the question on the system as a graph query, which will be answered by the retrieved results. Additionally, **Graph Search** is also a visualization of retrieved relations that makes users easier to understand the results. Figure 3b shows the outcome of the above query. One of the results is the graph with two relations *("COVID-19", "cause", "onset of autoimmune diseases")*, and *("hydroxychloroquine", "treat", "autoimmune diseases")*.

4 Experimental Results

4.1 Corpus

The CovRelex-SE system makes use of a snapshot of CORD-19 at March 31^{st}, 2022. The dataset is

a resource of over 900,000 scholarly articles about COVID-19 and related coronaviruses. Relation extraction and embedding extraction were performed on the abstracts of the papers.

4.2 Relation Extraction & Entity Recognition

As illustrated in Table 2, we extracted **107.4** million relations, **82.2** million of which were unique. On average, there are **160** unique relations extracted from a single document abstract. Among the methods, OpenIE generates the most results.

Table 2: Statistics of extracted relations.

Method	Non-uniq/corpus	Uniq/corpus	Uniq/abstract
ReVerb	5.5M	4.5M	8
OLLIE	11.0M	8.9M	17
ClausIE	20.9M	16.9M	32
Relink	12.7M	10.0M	20
OpenIE	57.3M	45.8M	90
Overall	107.4M	82.2M	160

As shown in Table 3, four entity recognition models have identified **15.1** million distinct entities from the corpus. An average of **24** recognized entities are present for each abstract of CORD-19. Among the models, **en_ner_jnlpba_md** generates the most results. The top 3 common recognized entities are AMINO_ACID, CANCER, and CELL.

Table 3: Statistics of recognized entities.

Model	/corpus	/abstract
en_ner_craft_md	3.1M	5
en_ner_jnlpba_md	6.6M	11
en_ner_bc5cdr_md	3.4M	5
en_ner_bionlp13cg_md	2.0M	3
Total	15.1M	24

4.3 Embedding Extraction & Faiss Index

To pre-train CORD19-BERT, we extract **52.8** million sentences from the CORD-19 corpus using both abstract and full-text of documents. We then pre-train the model following the BERT-base settings (110M parameters) (Devlin et al., 2018). In the initialization step, we use a peak learning rate **5e-05** and train for **4.7** million steps, Adam optimizer with epsilon **1e-08**, and batch size of **32** sequences with **512** tokens. Training took **99** hours on one NVIDIA A100 GPU. After that, we use the pre-trained model to perform embedding extractions. Each component of a relation is converted to a 768-dim vector using this model.

Using the Faiss package, we divide the searching space of embedding vectors into **100** clusters. When searching a query, the users can easily alter the value of parameter **nprobe**, which affects how many adjacent clusters are used to search. Table 4 shows the required time to search a query *("covid", "cause", DISEASE)* based on different values of **nprobe**. The increase of the **nprobe** implies longer search time. In general, there are three main factors that affected the search time of a query including the number of non-empty components in the query, the number of components with entity types, and the value of **nprobe**.

Table 4: Statistics of search time for different hyperparameter values of **nprobe**.

Hyperameter	Search time
Nprobe=1	5.92 s
Nprobe=10	6.86 s
Nprobe=50	19.46 s
Nprobe=100	23.19 s

4.4 Evaluation Settings and Results

To demonstrate the effectiveness of two search functions of our system, we conduct an evaluation task. The queries are created by using the content of sample articles in the corpus. There are 50 single-relation search queries and 30 graph search queries were created. Two evaluators work together to evaluate the returned results. Specifically, the evaluation process contains three phrases as follows:

- **Phase 1: How to use system.** Two evaluators carefully read the manual[4] of our system.
- **Phase 2: Evaluating.** Two evaluators separately determine whether the returned result of systems are correct or not. A correct result contains at least one relation that can be entailed from its corresponding paragraph and answer the query. After that, if any answers weren't identical, they adjudicated with each other.
- **Phase 3: Combining answers.** We collect the answers from two evaluators. Only answers that are accepted by both evaluators are counted as correct ones. In addition, we used Cohen's kappa coefficient (McHugh, 2012)

[4] https://www.jaist.ac.jp/is/labs/nguyen-lab/systems/covrelex-se/docs/

Table 5: Evaluation results on systems. **Correct I&II**: evaluated as correct results (can be entailed the expected answer from top-5 returned relations) by both the evaluators. **Kappa**: Cohen's kappa coefficient.

Function	Method	Correct I & II	Kappa
Single Relation Search	CovRelex	28 (56%)	0.78
	CovRelex with semantic	41 (82%)	0.85
	CovRelex-SE	**42 (84%)**	0.83
Graph Search	CovRelex	15 (50%)	0.72
	CovRelex-SE	**22 (73.3%)**	0.70

to estimate the agreement between the two evaluators.

As a baseline, we perform the queries on CovRelex using the system's default settings. Moreover, we add the semantic search component to the baseline and refer to this setting as CovRelex with semantic. For CovRelex with semantic and CovRelex-SE, we set the values of $(\alpha, \beta, \gamma, \theta)$ in section 3.4 to *(1.0, 0.1, 1.0, 0.5)*.

Table 5 shows the evaluation results. For single-relation search, we can see that using the semantic improves the system by 26% over using simply the lexical method. In addition, after employing the latest data corpus, the accuracy of our system enhances to 84%. For graph search, our system performs better than CovRelex by 23.3%. Moreover, Cohen's kappa coefficients of the methods are greater than or equal to 0.7, which is considered a good agreement (Fleiss et al., 2013).

4.5 Result Discussion

We observe that the proposed system is able to make more effective use of semantic information than the baseline CovRelex system. Specifically, instead of lexical matching only, the CovRelex-SE system also searches with the meaning of keywords. For example, there is a query *("", "shield", "lung")* that describes the question *"What thing shields the lungs?"*. Figure. 4 presents top-1 retrieved relations based on each system for this query. We can see that the CovRelex system can not return any results. On the other hand, the CovRelex-SE system knows the close meaning between *"shield"* and *"protect"* in this context and returns the relation *("ARBs", "protect", "lung")*.

There are some cases that the CovRelex-SE system with default settings fails to retrieve the correct relations in top-5 results, for example *("lung radiological image", "screen", "covid")*. In general,

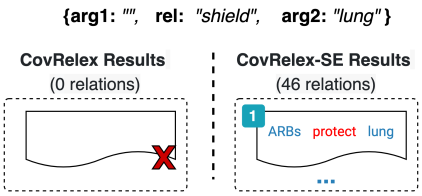


Figure 4: An example results of systems.

we can further improve the accuracy of the system by changing the values of hyperparameters such as **nprobe**. However, there is a trade-off between computation time and accuracy.

5 Threats to Validity

There are two main threats to validity in this study, which are described as follows.

5.1 Threat of Evaluation Settings

In this study, we evaluate the performance of the systems in Table 5 using the default settings. As a result, their configuration values might not be optimal for the systems. To reduce the threat, we run several queries through the systems, manually changing the value of each setting and selecting the one with the most relevant and consistent results. Also, we intend to use an evaluation task to determine the best settings for each system.

5.2 Threat of Extracting Relations

This threat mainly lies in the extracted relations that are used for ranking. The threat may come from the relation extraction methods that do not capture all available relations or extract the incorrect ones. To minimize the threat of extracting false positive relations, we carefully investigate the relation extraction methods. Also, we plan to use additional relation extraction methods to capture all possible relations in the documents.

6 Conclusions

In this paper, we present CovRelex-SE, a novel COVID-19 retrieval system for ranking relations in the CORD-19 corpus. The score of a relation is calculated based on semantic and lexical scores. The semantic score is determined using the embeddings from CORD19-BERT, whereas the lexical score is computed using the Elasticsearch engine. In order to evaluate the effectiveness of CovRelex-SE, we conducted an evaluation task. The experimental results show that our system outperforms the

CovRelex system in both single-relation search and graph search.

References

Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D. Manning. 2015. Leveraging linguistic structure for open domain information extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 344–354, Beijing, China. Association for Computational Linguistics.

Michael Bada, Miriam Eckert, Donald Evans, Kristin Garcia, Krista Shipley, Dmitry Sitnikov, William A Baumgartner, K Bretonnel Cohen, Karin Verspoor, Judith A Blake, et al. 2012. Concept annotation in the craft corpus. *BMC bioinformatics*, 13(1):1–20.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. Legal-bert: The muppets straight out of law school. *arXiv preprint arXiv:2010.02559*.

Nigel Collier and Jin-Dong Kim. 2004. Introduction to the bio-entity recognition task at jnlpba. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP)*, pages 73–78.

Luciano Del Corro and Rainer Gemulla. 2013. Clausie: clause-based open information extraction. In *Proceedings of the 22nd international conference on World Wide Web*, pages 355–366.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Andre Esteva, Anuprit Kale, Romain Paulus, Kazuma Hashimoto, Wenpeng Yin, Dragomir Radev, and Richard Socher. 2020. Co-search: Covid-19 information retrieval with semantic search, question answering, and abstractive summarization. *arXiv preprint arXiv:2006.09595*.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, pages 1535–1545.

Joseph L Fleiss, Bruce Levin, and Myunghee Cho Paik. 2013. *Statistical methods for rates and proportions*. john wiley & sons.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2020. Domain-specific language model pretraining for biomedical natural language processing.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547.

Jin-Dong Kim, Tomoko Ohta, Yoshimasa Tsuruoka, Yuka Tateisi, and Nigel Collier. 2004. Introduction to the bio-entity recognition task at jnlpba. In *Proceedings of the international joint workshop on natural language processing in biomedicine and its applications*, pages 70–75. Citeseer.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. 2016. Biocreative v cdr task corpus: a resource for chemical disease relation extraction. *Database*, 2016.

Mary L McHugh. 2012. Interrater reliability: the kappa statistic. *Biochemia medica*, 22(3):276–282.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. 2019. Scispacy: fast and robust models for biomedical natural language processing. *arXiv preprint arXiv:1902.07669*.

Arantxa Otegi, Iñaki San Vicente, Xabier Saralegi, Anselmo Peñas, Borja Lozano, and Eneko Agirre. 2022. Information retrieval and question answering: A case study on covid-19 scientific literature. *Knowledge-Based Systems*, 240:108072.

Sampo Pyysalo, Tomoko Ohta, Rafal Rak, Andrew Rowley, Hong-Woo Chun, Sung-Jae Jung, Sung-Pil Choi, Jun'ichi Tsujii, and Sophia Ananiadou. 2015. Overview of the cancer genetics and pathway curation tasks of bionlp shared task 2013. *BMC bioinformatics*, 16(10):1–19.

Shaina Raza, Brian Schwartz, and Laura C Rosella. 2022. Coquad: a covid-19 question answering dataset system, facilitating research, benchmarking, and practice. *BMC bioinformatics*, 23(1):1–28.

Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. 1995. Okapi at trec-3. *Nist Special Publication Sp*, 109:109.

Michael Schmitz, Stephen Soderland, Robert Bart, Oren Etzioni, et al. 2012. Open language learning for information extraction. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 523–534.

Vu Tran, Van-Hien Tran, Phuong Nguyen, Chau Nguyen, Ken Satoh, Yuji Matsumoto, and Minh Nguyen. 2021. Covrelex: A covid-19 retrieval system with relation extraction. In *Proceedings of the*

16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, pages 24–31.

Xuan-Chien Tran and Le-Minh Nguyen. 2021. Relink: Open information extraction by linking phrases and its applications. In *International Conference on Distributed Computing and Internet Technology*, pages 44–62. Springer.

Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Darrin Eide, Kathryn Funk, Rodney Kinney, Ziyang Liu, William Merrill, et al. 2020a. Cord-19: The covid-19 open research dataset. *ArXiv*.

Xuan Wang, Yingjun Guan, Weili Liu, Aabhas Chauhan, Enyi Jiang, Qi Li, David Liem, Dibakar Sigdel, John Caufield, Peipei Ping, et al. 2020b. Evidenceminer: Textual evidence discovery for life sciences. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 56–62.

Edwin Zhang, Nikhil Gupta, Raphael Tang, Xiao Han, Ronak Pradeep, Kuang Lu, Yue Zhang, Rodrigo Nogueira, Kyunghyun Cho, Hui Fang, et al. 2020. Covidex: Neural ranking models and keyword search infrastructure for the covid-19 open research dataset. *arXiv preprint arXiv:2007.07846*.

A Appendix

A.1 Detailed settings of Faiss

One issue with data processing is the excessive number of embedding vectors; 82 million relations correspond to 246 million 758-dim vectors. Then calculating the query's embedding and iterating over all corpus relations' embeddings is impractical. Therefore, we used Faiss to resolve this issue. The detailed settings of Faiss[5] are shown in Table. 6.

Table 6: Detailed settings of Faiss.

Setting	Value
Version	1.7.2
Faiss Index	IndexIVFFlat
Faiss Quantizer	IndexFlatL2
Faiss **nlist**	100
Faiss **nprobe**	10 (default)

[5]See Faiss wiki page for the meaning of each setting: https://github.com/facebookresearch/faiss/wiki/

A.2 Examples appearing in CORD19-BERT Vocabulary

Table 7 shows some examples of subwords that exist in the CORD19-BERT vocabulary but not in BERT and PubMedBERT. From these examples, it can be seen that the embedding spaces of BERT and PubMedBERT are not capable of describing the important concepts of COVID-19-related documents directly. This can affect the model's performance on representing the semantic information of phrases. We provide subwords with explanations of them from Oxford Online Learner's Dictionary[6]. Most of these terms are related to the coronavirus.

Table 7: Examples appearing in CORD19-BERT vocabulary, not in BERT and PubMedBERT vocabularies.

Token	Explanation
covid	A disease caused by a coronavirus, especially Covid-19.
coronavirus	A type of virus that can cause pneumonia and other diseases in humans and animals.
respirator	A piece of equipment that makes it possible for somebody to breathe over a long period when they are unable to do so naturally.
quarantine	A period of time when an animal or a person that has or may have a disease is kept away from others in order to prevent the disease from spreading.
vaccinate	To give a person or an animal a vaccine, especially by injecting it, in order to protect them against a disease.
disinformation	False information that is given deliberately.
distancing	To become less involved or connected with somebody/something.
facemask	Something that you wear over part or all of your face, in order to protect it or to prevent the spread of disease.
lockdown	An official order to control the movement of people or vehicles because of a dangerous situation.
##infection	Wordpiece in words containing "infection" (e.g. reinfection, coinfection)

[6]https://www.oxfordlearnersdictionaries.com/