Towards an Extensible Pipeline for Biomedical Knowledge Graph Construction from Scientific Papers

Salah Edine Ech-chorfi, Elmoukhtar Zemmouri

***Abstract*—In these modern times, the world has access to vast data in all aspects of life. These data are stored in different formats, from structured databases to unstructured text, images, and audio. However, with the rise in the amount of data available, the challenge of data processing emerges to extract useful knowledge effectively. Many computer science practitioners in the machine learning community have focused on Information Retrieval from data. This work focuses on knowledge graph extraction from natural language text, specifically biomedical text, for its high impact on humankind. We introduce an end-to-end pipeline that takes in raw text and outputs a Knowledge Graph summarizing the entities within and the relations between them. Using state-of-the-art methods for Named Entity Recognition and Relation Extraction, the pipeline relies heavily on supervised learning in the steps where it provides better results than the alternatives (e.g., unsupervised learning, distant supervision). The proposed process is applied to natural language text from the biomedical domain through scientific papers and preprints. In the scope of this paper, the pipeline is trained to extract four classes of entities and five types of relations. However, its application can be extended to other types of entities and relations depending on the use case.**

The pipeline is publically available on the Github repository https://github.com/echchorfisalahedine/KG_Extraction_Pipeline.

***Index Terms*—Knowledge Graph, Named Entity Recognition, Relation Extraction, Biomedical Data**

I. Introduction

TEXT mining is a field of machine learning that consists of extracting knowledge from natural language text. Its applications in various industries include information retrieval, sentiment analysis, topic detection, text summarization, and Knowledge Graph (KG) extraction. Text mining relies on multiple Natural Language Processing (NLP) tasks to transform the raw text into machine-readable data. In this paper, we aim to build a knowledge graph from biomedical text. A knowledge graph is a structured representation of data in the form of a network of real-world entities, also called nodes, interconnected with edges that define the relations among them. Automating knowledge graph construction from text relies on a series of fundamental NLP tasks required to extract the necessary elements of the KG: entities and relations. In the following, we summarize the three essential tasks that are the core of any KG construction model :

Named Entity Recognition (NER): Named Entity Recognition is an NLP task that identifies and categorizes the words or series of words representing real-world entities from text. In the biomedical field, these entities can vary among biomedical concepts such as genes, diseases, chemicals, tests, or procedures. For example, given the sentence “Symptoms of active Tuberculosis disease in the lungs usually begin gradually and worsen over a few weeks”, the NER method should identify and mark the term Tuberculosis as a disease. Over the years, NER models have been developed rapidly. The first systems are dictionary-based models that rely on string-matching algorithms that check whether the words in the text are present in a dictionary containing the targeted vocabulary. MetaMap [1] is a state-of-the-art example of a dictionary-based method introduced in 2001 that maps entities to their equivalent concepts in the Unified Medical Language System (UMLS) Metathesaurus [2] and can identify these entities in a given text. The next type of model is rule-based systems. The rule-based models perform NER by extracting the entities that satisfy a set of predefined patterns and context-based rules. Proper [3] (1998) and Text Detective [4] (2005) are examples of state-of-the-art rule-based NER models. Machine Learning-based models treat NER as a classification or sequence labeling problem. Different systems were introduced, relying on ML models such as support vector machines (SVMs), hidden Markov models (HMMs), and CRFs and leveraging several word features such as Part-of-Speech (POS) tags, prefixes, and suffixes. These models were followed by Deep Learning-based approaches that leverage word embeddings along with context information, POS tags, and position to predict the entities' tags. These models benefit from pre-trained word embeddings, the ability to use pre-trained models, and the ability to be paired with ML-based models. Long short-term memory (LSTM), GRAM-CNN, and BERT-based models have recently gained considerable popularity for their improved results.

Entity Linking (EL): In natural language text, a concept may be referred to by different words or sometimes abbreviated. For the human reader, it is relatively easy to determine whether different words refer to the same entity, which is not the case for machines. In the following example, “The most common symptoms of coronavirus disease are fever, chills, and sore throat. COVID-19 also has some uncommon symptoms like nausea and diarrhea.” the machine should recognize coronavirus disease and COVID-19 as one single entity just like a human would. Entity Linking is a crucial

Manuscript received August 20, 2024; revised January 12, 2025.

S. E. Ech-chorfi is a PhD student in the Department of Mathematics and Computer Science, Ecole Nationale Supérieure d'Arts et Métiers, Moulay Ismail University, Meknes, 50000 Morocco e-mail: sa.echchorfi@edu.umi.ac.ma.

E. Zemmouri is a professor of computer science in the Department of Mathematics and Computer Science, Ecole Nationale Supérieure d'Arts et Métiers, Moulay Ismail University, Meknes, 50000 Morocco e-mail: e.zemmouri@umi.ac.ma.

part of text mining as a continuation of NER. It primarily aims to solve the ambiguities related to the fact that different words or expressions can refer to real-world entities. EL consists of mapping the extracted entities to their corresponding concepts in a unified Knowledge Base so that the entities that share the same meaning get mapped to the same concept. EL methods exploit several KBs such as DBpedia [5], Freebase [6], Wikidata [7] in the general domain, Mesh [8], and UMLS [2] among others in the biomedical domain. The model mentioned above, MetaMap [1], performs EL, along with numerous state-of-the-art systems introduced in recent years and leveraged the advancement in neural networks such as Yamada et al. (2021) [9], and De Cao et al. (2021) [10].

Relation Extraction (RE): Relationship Extraction (RE) is a subtask of information retrieval that generally follows NER. Its purpose is to identify the relations among the previously extracted entities to provide a context for their presence and meaning for their interaction in the text. Due to its importance, RE has been of interest to the ML community for many decades, resulting in significant advancements in RE approaches over the years. The first RE models are rule-based methods, which rely on a manually determined set of patterns corresponding to a set of relations. The text is verified against these patterns, and when a span of text matches one of the patterns, the model concludes that it contains the relation corresponding to that pattern. Nebhi (2013) [11] is an example of rule-based RE models. Machine Learning-based RE methods quickly followed next and varied among different categories. Supervised learning approaches deal with the problem as a classification problem and require important amounts of data for training, such as Zhou et al. (2005) [12]. Unsupervised learning methods, like WEBRE (2012) [13], use unlabeled data and rely on clustering the pair of entities based on their context into clusters. Each cluster is assigned a semantic relation representing the relation between all entity pairs of that cluster. Semi-supervised RE methods leverage the advantages of both supervised and unsupervised models by using a small amount of labeled data to learn the extraction pattern and then extracting similar relations from the rest of the unlabeled data. Distant supervision approaches require matching the text to an existing KB of entities and relations among them. If an entity pair in a sentence matches a pair in the KB, then the sentence is associated with the relation between the pair in the KG. Mausam et al. (2012) [14] and Zeng et al. (2014) [15] are state-of-the-art examples of semi-supervised and distantly supervised methods, respectively. The advancement in Deep Learning allowed RE to benefit from new tools such as Convolutional Neural Networks (CNN) (Li et al. (2018) [16]), Long Short-term Memory (LSTM) (Zhou et al. (2015) [17]), and transformers ((Han et al. (2021) [18]) and gain state-of-the-art results.

These three tasks will be the core of our proposed biomedical Knowledge Graph construction pipeline. We will investigate the best models for each step to get good performance for the overall KG extraction process. The pipeline we propose processes raw text, especially from scientific papers, and provides a set of triples (Subject, Predicate, Object) as an output visualized as a KG following different steps of NLP techniques and relies on different state-of-the-art trainable approaches using supervised learning. The process can be extended beyond the scope of biomedical data. We begin the pipeline by preprocessing and cleaning the input text. We perform Named Entity Recognition on the forwarded text, followed by Entity Linking and Relation Extraction. Finally, we clean the predicted triples and generate a KG formalized using the Resource Description Framework (RDF) standard format.

In the following, we will review the literature to explore several approaches for KG construction from the text. Secondly, we introduce an end-to-end pipeline to extract information from natural language text in the form of KG featuring different types of entities and the relations among them. The pipeline is applied to a dataset of papers and preprints from the biomedical domain, as it is a rich field with an important amount of data stored in text format, such as scientific papers, clinical notes, lab results, and diagnosis reports. Lastly, we evaluate the overall performance of our pipeline along with the evaluation of its main components individually. The evaluation results are compared to those of the works mentioned above.

II. Related work

Knowledge extraction from natural language text has gained much interest in recent years. Many approaches have been published containing processes to transform an input unstructured text into KGs. In this section, we review the literature to explore the various adequate NLP tools and assess the advancement of this task and the results achieved. We provide an overview of some of the works applied to different biomedical data such as Electronic Medical Records (EMRs), clinical notes, and scientific papers, with a summary of their tasks, datasets, and results as depicted in table I.

Linfeng et al. [19] introduced a pipeline to transform EMRs into exploitable KG. It focuses on extracting relations between diseases and a set of different entities. The process relies on eight steps briefly described as follows :

1) Data preparation: The data is collected and prepared from the private big data platform of the Southwest Hospital in China, which contains 16,217,270 visits of 3,767,198 patients and consists of chief complaints, illness history, lab exams, and drug prescriptions.
2) Named Entity Recognition: This step is performed by a hybrid vocabulary-based bidirectional maximum matching method, BiLSTM-CRF model, and pattern recognizer. The model extracts nine types of entities in total, which are *disease*, *gender*, *age range*, *symptom*, *exam*, *lab exam*, *lab item*, *medicine*, and *surgery*.
3) Relation Extraction: The approach for this task is to establish nine types of relations between the diseases and the entities present in a single patient's visit. The relations are constructed from the nine types of entities as follows: *disease-related-disease*, *disease-related-gender*, *disease-related-agerange*, *disease-related-medicine*, *disease-related-symptom*, *disease-related-exam*, *disease-related-labexam*, *disease-related-labitem*, and *disease-related-surgery*.
4) Property calculation: This step computes a set of properties for each relation: the probability of the object being related to the subject, the specificity that reflects the object's significance to the subject, and the reliability

to curate the extracted relations. These properties are assigned to the relation as a novel quadruplet (Subject, Predicate, Object, Properties) instead of the standard SPO form (Subject, Predicate, Object).

5) Graph cleaning: In this step, entities and relations with several occurrences inferior to a chosen threshold of 10 are removed to clean noisy triples.
6) Related-entity Ranking: This step introduces a new score function to rate the relation between the subject and the object based on the calculated properties, which is the multiplication of the probability, the specificity, and the reliability of the relations between a given subject and its related objects. The function is adapted to the relation *disease-related-labexam* to account for abnormal exam results.
7) Graph Embedding: PrTransH model [20] is trained to learn graph embeddings of all diseases and relations that serve in disease clustering using the DBSCAN algorithm [21].

In another work, Harnoune et al. [22] tackle the challenge of extracting knowledge graphs from clinical notes. It introduces a pipeline for extracting entities and relations in text spans from patient drug prescriptions. The pipeline consists of 5 steps as follows:

1) Preprocessing: The first step of the pipeline consists of tokenizing the input text followed by word embedding using a BERT model [23]. The model returns text embedding that is fed to the next step.
2) Named Entity Recognition: This step applies BERT for NER paired with a CRF layer to link the extracted entities to their respective classes. Many variants of BERT were tested namely: BioBERT [24], BioClinicalBERT [25], BioDischargeSummury [25], BioRoberta [26] to extract several types of entities such as *Patient*, *Drug*, *Strength*, *Posology*, *ADE* (side effect of the drug), and *Reason*. The input text is segmented into spans containing as many paragraphs as the maximum number of tokens in BERT (512) can fit. In this context, BioClinicalBERT proved more efficient than the other variants.
3) Coreference Resolution: In this step, NeuralCoref resolves coreference in text using a pre-trained statistical model integrated into spaCy's NLP pipeline.
4) Relation Extraction: In this work, the RE task is tackled as a binary classification problem to predict if a relation is present between each pair of entities in the input span using the BERT variant BioClinicalBERT. The approach maps entities in pairs based on their classes and predicts whether the relation exists for each.
5) Graph Construction: A KG is constructed using the extracted entities and several properties and is linked based on the predicted relations. Five types of entities form the nodes (Patient, Drug, Posology, ADE, Reason), and the other types are represented by their attributes (e.g., Strength, Dosage, Duration). The edges represent the entities' relations in entity-type-1 and entity-type-2 (e.g. Patient-Posology). The graph is built on a neo4j database.
6) Graph Analysis: The work provides a series of analysis operations that can be carried out on the KG, such as the drug most used by the patients, the set of prescriptions that treat the same symptom (reason), and the most important reason for taking the drug in the graph.

Wise et al. [27] focused on extracting knowledge graphs from the scientific papers from the COVID-19 Open Research Dataset (CORD-19) [28]. The work aims to build a KG named COVID-19 Knowledge Graph (CKG) containing information about the scientific articles and the biomedical entities stored inside. The resulting KG is leveraged to perform Information Retrieval and article recommendation. However, in this section, we will focus on the pipeline and the various NLP tasks leading to the construction of the KG. The pipeline contains the following steps:

1) Entities Extraction: The KG is constructed from 5 types of entities and determined attributes. Paper entities that represent the scientific article, Author entities, and Institution entities that represent the affiliation for the authors and their attributes are retrieved from CORD-19 [28] metadata. Concept entities are extracted from the article's abstract and body using the Amazon Web Service for NER Comprehend Medical (CM) Detect Entities V2 and sorted into categories such as Anatomy, Test Treatment Procedure, Medical Conditions, and Medication. The KG also contains topic entities deduced by Z-LDA [29] and the help of medical professionals.
2) Relations Extraction: The relations of the KG are determined by the interactions of the different types of entities and are explained as follows: authored-by (Paper – Author), affiliated-with (Author – Institution), associated-concept (Paper – Concept), associated-Topic (Paper – Topic), cites (Paper – Paper).
3) KG Curation: This work discards the extracted biomedical entities with a confidence score of less than 50% to clean the resulting KG. It also performs lemmatization to normalize the entities' names and run a distribution of the entities to filter out those with an occurrence of 0.0001% and pass those with an occurrence of 50% or more to manual qualitative assessment. The authors' entities are also normalized to avoid redundancy and improve citation liking based on the authors' names.

Gajendran et al. [30] propose an end-to-end pipeline to extract biomedical knowledge from abstracts of the CORD-19 papers in the form of the top diseases, proteins, and chemicals related to COVID-19. The pipeline relies on six steps :

1) Preprocessing: The CORD-19 abstracts are collected and prepared for the next step.
2) Feature Extraction: In this step, a BERT-BiLSTM-CRF model is finetuned to extract 3 types of entities (*Disease*, *Protein*, *Chemical*) using NCBI-Disease [31], JNLPBA [32], and CHEMDNER [33] datasets respectively.
3) Named Entity Recognition: The CORD-19 abstracts are passed as input to 3 distinct NER models to extract the diseases', proteins', and chemicals' entities, respectively. This model consists of a finetuned SciBERT [34] model, which yields a representation vector of 768 in size, to the BiLSTM [35] layer that produces a vector with a length of the number of entities' types in the

training dataset. The output is fed to a CRF layer that extracts the best possible tag sequence for the input sentence.

4) Relation Extraction: This work focuses on extracting two types of relations between the entities. Two independent SciBERT models are used to predict whether or not a relation exists between entities in a single sentence. The first model is trained on BC5CDR [36] for the Chemical-Disease relation, and the second model is trained on CHEMPROT [37] for the Chemical-Protein relation.
5) Graph Construction: A Neo4j graph database stores the resulting entities and relations from the NER and RE steps. The nodes represent the entities by their names and types, and the two relations are represented by directed edges linking the head-to-tail entities. To clean the resulting KG from noise, the triples whose head or tail entities have a lower occurrence count than five are discarded.
6) Representation Learning Module: A TransD model [38] is used to detect the top 25 entities related to COVID-19. For this purpose, the model is trained to learn the embeddings of all entities and relations. The embeddings of the entities whose edge count is less than five are discarded. The remaining obtained embeddings are compared to the model's embedding of COVID-19 using cosine similarity to determine the top 25 entities related to each entity type to COVID-19.

Lamy et al. [39] is another state-of-the-art work exploiting EMRs to extract valuable knowledge for healthcare practitioners. In this work, EMRs are processed through a pipeline to provide structured clinical data that are suitable for querying and analysis operations. The records are gathered and sorted by specialties such as oncology, rheumatology, and gastroenterology. In summary, the pipeline follows four steps:

1) Data preprocessing: The initial EMRs are cleaned up by resolving all abbreviations and acronyms and correcting orthographic errors.
2) Data translation: Since major development in NLP techniques has focused on English text, the text is translated from Portuguese to English using the Google Translate API, preserving the data's original meaning and slightly impacting the pipeline's performance.
3) Named Entity Recognition: The processed English text is passed to cTAKES [40], a state-of-the-art NLP tool that extracts medical terms using a series of operations. Firstly, the Sentence Boundary Detector segments the text into sentences. These sentences are split into words using a tokenizer, which are normalized after removing prefixes and suffixes. The POS tagger assigns each word its respective tag (e.g., noun, verb). This Shallow Parser then links these words into higher logical units and noun groups such as respiratory tract infections. The final component is the Named Entity Recognizer, which identifies entities based on the SNOMED-CT dictionary containing over 300.000 clinical terms. The pipeline can extract multiple entity types such as diseases, medications, symptoms, signs, anatomical regions, and clinical procedures.
4) Storage and querying: The extracted entities are stored in an XMI (XML Metadata Interchange) file and a structured database. With the help of SQL queries, The work can extract valuable knowledge from the data and find relations between entities from different types, correlations, and patterns in the EMRs.

Table I below summarizes the works discussed above.

From the analysis of several related work methods, we notice that the approaches providing end-to-end pipelines for KG extraction usually focus on a specific type of data or specific types of entities and relations. In our work, we aim to provide a pipeline that takes in any form of biomedical natural language text and can be trained and scaled to extract any type of entities and relations. Our work also focuses on returning pure biomedical content, free from any literature or general information like the notions of authors, papers, and organizations.

III. Proposed Method

In this work, we propose a pipeline that takes a biomedical raw text as input and transforms it into a KG, using a series of text mining tasks and employing several state-of-the-art NLP tools. Our goal is to provide a trainable end-to-end pipeline for any case of KG extraction from biomedical text. This approach mainly contains supervised learning tools to offer a high degree of trainability for different scenarios, which means that the pipeline's output relies on the nature of the data used in the training phases of the different steps. The pipeline can be customized to extract different entities and relations by training some components on a set of entity and relation types. In our case, the work is applied to biomedical text in the form of scientific papers and preprints from the CORD-19 dataset [28]. For that, we use several biomedical datasets to train the different models.

Figure 1 depicts the overall architecture of the proposed pipeline. This pipeline is composed of 6 steps in total, summarized as follows:

- Text preprocessing: In this step, we prepare our input data by performing a series of NLP tasks from any data that may interfere with the pipeline's performance.
- Named Entity Recognition: We aim to extract the biomedical entities present in the text. In our case, we leverage several publicly available benchmark datasets to extract four types of entities: *Disease*, *Drug*, *Gene or gene product*, and *Cell*.
- Entity Linking: In this step, we map the extracted entities to their corresponding standard terminology in the UMLS [2] set of vocabularies.
- Relation Extraction: We perform RE to extract the relations among entities in a sentence from the input text.
- Graph cleaning: We conduct cleaning operations to cure the output from noise and wrong predictions.
- KG construction: Finally, we group the filtered entities and relations into a refined KG and visualize it for the end user.

A. Prepossessing

The first step requires cleaning and structuring the input text. We use fragments of the topic modeling pipeline [41] proposed in our previous work. Specifically, we apply the

TABLE I
APPROACHES FOR KNOWLEDGE GRAPH EXTRACTION FROM TEXT: SUMMARY OF TASKS, DATASETS, AND EVALUATION RESULTS.

Reference	Tasks	Datasets	Evaluation Metrics		Results
Linfeng et al. (2020) [19]	NER Related-entity Ranking RE KG Extraction	Chief complaints, Illness history Lab exams Drug prescriptions	NER	P	0.9727
				R	0.9689
				F1	0.9708
			Related-entity Ranking[1]	NDGC	0.85/1.00
Harnoune et al. (2021) [22]	NER RE KG Extraction	Care-III (MIMIC-III)	NER	F1	0.907
			RE	F1	0.88
Wise et al. (2020) [27]	NER KG Extraction	CORD-19			
Gajendran et al. (2023) [30]	NER RE KG Extraction KG Embedding	CORD-19 NCBI-Disease JNLPBA CHEMDNER BC5CDR CHEMPROT	NER	P	0.8849
				R	0.8902
				F1	0.8876
			RE	P	0.74
				R	0.73
				F1	0.73
Lamy et al. (2023) [39]	NER	Electronic Medical Records	NER	P	0.75
				R	0.61
				F1	0.67

same preprocessing operations to clean the input text before passing it to the next step. In this case, we perform tokenization, lemmatization, and Language detection operations using the open-source library spaCy for various NLP tasks. Next, we remove a custom list of stopwords. This custom list contains the standard Natural Language Toolkit (NLTK) stopwords in addition to a set of common words, mainly found in scientific publications (e.g., “i.e.”, “fig”, “al.”). These words do not carry a specific meaning but form an essential part of scientific text. Finally, we remove punctuation except for the full-stop symbol (.) to keep the notion of sentences and distinguish between sentences later on. For our application on CORD-19, we also integrate the part responsible for the papers’ collection from [41] to feed the papers to our pipeline. The body texts of each paper run through the same preprocessing operations and are grouped in a single string, forming a set of sentences to be forwarded to the next step.

B. Named Entity Recognition

1) Process: In this step, we aim to extract the medical entities present in the processed text by training a supervised learning model to recognize four types of entities: *Disease*, *Drug*, *Gene or gene product*, and *Cell*. In various literature reviews, benchmarks are carried out to compare NER models based on their performance and properties. Transfer learning-based approaches have proven to be effective in the case of training the model for a task on a limited amount of domain-specific data. This method consists of pre-training the model on a high-resource unlabeled dataset and then fine-tuning it on a small domain-specific dataset to perform a specific task. For NER, Agrawal et al. [42] provided a review of several neural network-based methods and their applications on different benchmark datasets (GENIA [32], GermEval 2014, JNLPBA [32]). We focus on GENIA and JNLPBA for their similarity to our training data. In Table II, we provide a comparison of the F1-scores of pre-trained BERT models from the Google AI SciBERT [34] and BioBERT [24], in addition to a CRF model and Bi-LSTM-CRF as cited in [42]. Generally, pre-trained BERT models performed better than CRF and Bi-LSTM-CRF on both datasets.

TABLE II
COMPARISON OF F1 SCORES OF SEVERAL STATE-OF-THE-ART NER MODELS ON 2 DATASETS [42]

Models	F1-score (GENIA)	F1-score (JNLPBA)
SciBERT	74.07	80.68
BioBERT	74.38	80.48
CRF model	65.15	74.23
Bi-LSTM-CRF	70.19	77.56

In another work, Lee et al. [24] performed a benchmark on different versions of the variant BioBERT [24], pre-trained on PubMed, PCM, and combinations of both. The models were trained on several benchmark datasets to extract disease, drug, chemical, gene, protein, and species entities. BioBERT v1.1, pre-trained on 1.1M PubMed abstracts (4.5B words), showed an overall better performance than the other versions of the variant, a pre-trained BERT model on a general domain corpus, and state-of-the-art results from previous works.

We adopt BioBERT [24] v1.1 and fine-tune it to perform NER on biomedical text by training multiple model instances on different datasets for each entity type. Our approach at this point consists of extracting entities on a sentence level by applying the four independent models to preserve the model’s performance to recognize a single type of entity. We use the weighted Cross Entropy Loss function to counter the effect of class imbalance and allow the models to focus on the classes representing medical entities. The extracted entities from each model are then combined into a list representing each sentence’s heterogeneous entities. Finally, these lists are grouped into a global list containing the entities from the whole corpus.

BioBERT can be fine-tuned to extract all types of entities depending on the training datasets, reflecting the customiza-

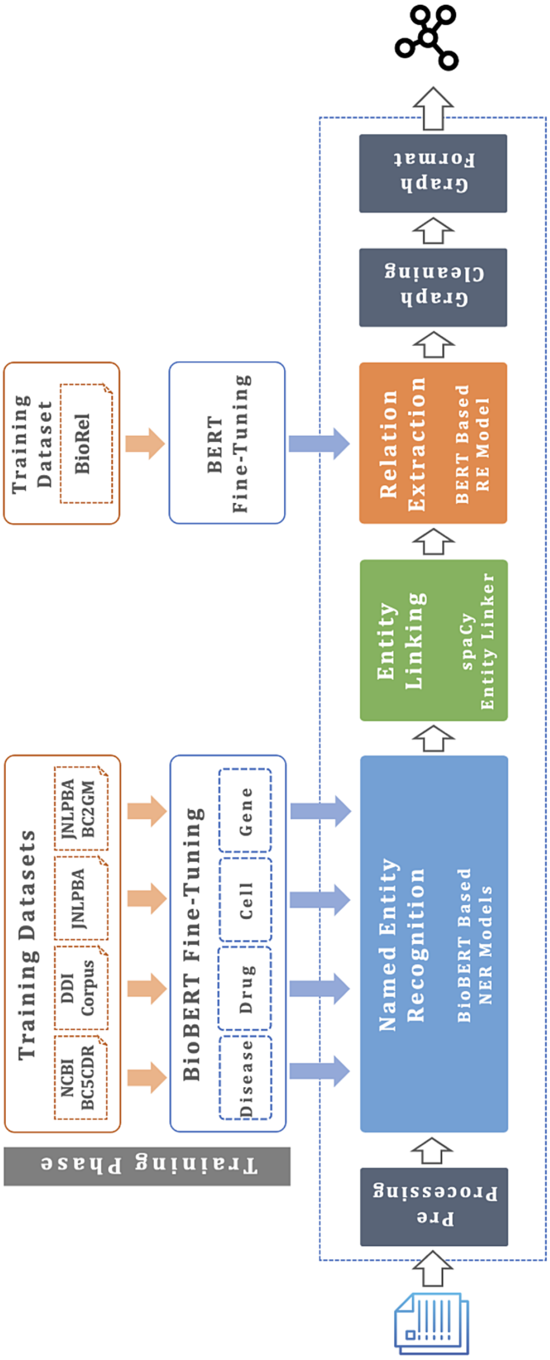

Fig. 1. Overall architecture of the proposed pipeline for Knowledge Graph construction from biomedical text: We feed the input text to the preprocessing component. Then we pass the processed text to our independently trained BioBERT models to extract 4 entities (Disease, Drug, Cell, Gene). The extracted entities are passed to the EL phase where they are linked to their respective classes in the UMLS Metathesaurus. We apply RE on each sentence in the text and its respective entities through a trained BERT model to extract the relations between those enities. Next, we clean the extracted triples and format the final triples into an RDF/XML file.

tion aspect in our pipeline. This variant of BERT's use is based on our application of the pipeline in the biomedical context. We create an Entity class to accommodate all the properties we need. The class contains the extracted name, a list of start and end indexes representing the entity's position, and the entity's predicted class (B-Disease, B-Drug, e.g.). Some extracted entities are composed of several consecutive words. We group these words into a single meaningful entity. The new entity's name is the concatenation of all the entities' names, and its position is a list containing the starting index of the first and the ending index of the last one.

2) Datasets: To fine-tune our BioBERT models for NER, we train four instances of the model on different datasets, each corresponding to a type of mined entity. We use various datasets from the BioNLP workshop for NER and other sources. We manually customize the data for our work to a unified format (Entity-name; IOB-tag). The IOB tags for all entities are O, representing the absence of the entity, B-Entity for the beginning of the entity, and I-Entity for the inside. For each entity type, we use the following datasets:

Disease: We combine the training and test sets of NBCI-Disease [31] and the training set of BC5CDR [36] for diseases into a more extensive training corpus for the model.

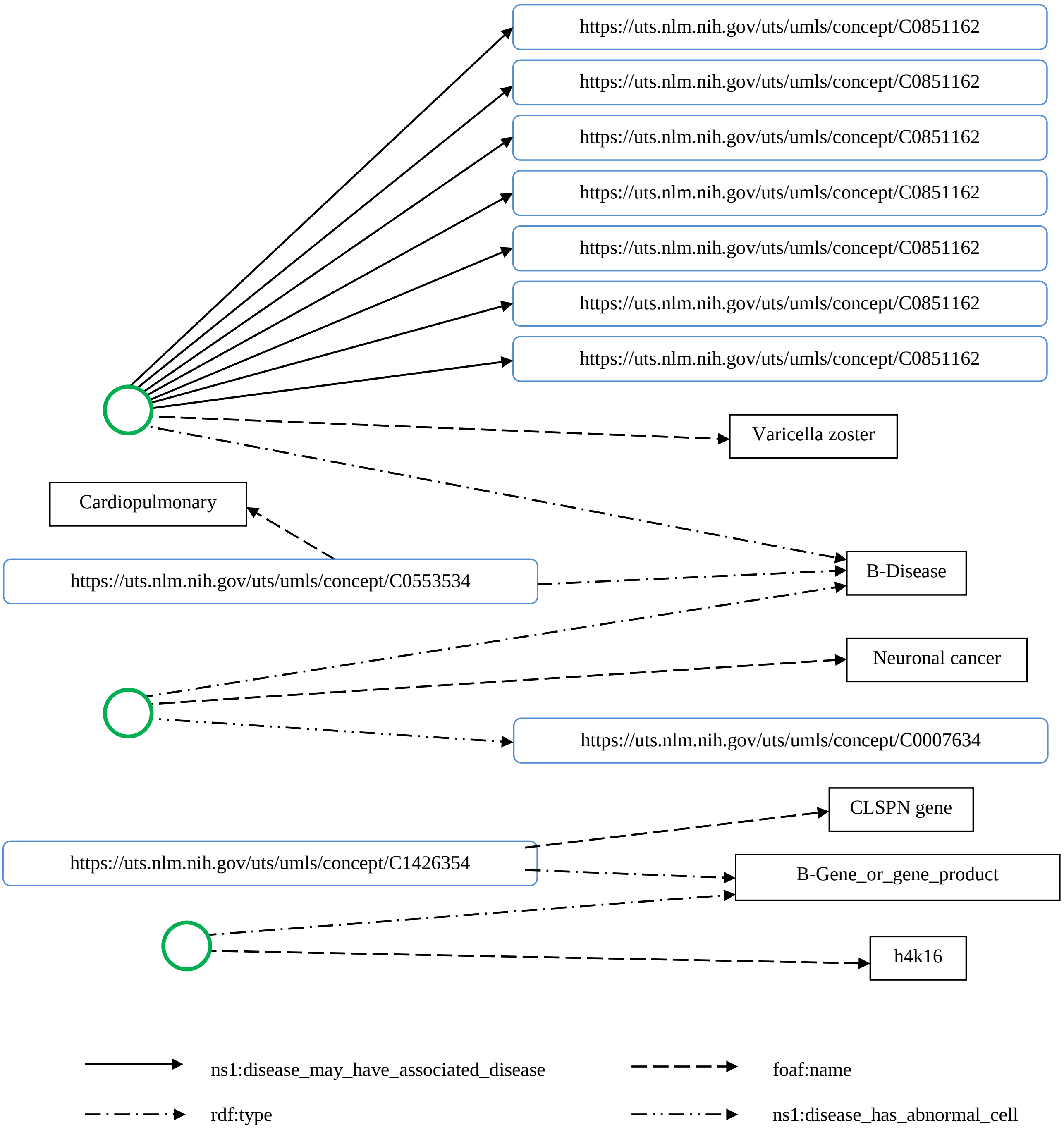
https://uts.nlm.nih.gov/uts/umls/concept/C0851162
https://uts.nlm.nih.gov/uts/umls/concept/C0851162
https://uts.nlm.nih.gov/uts/umls/concept/C0851162
https://uts.nlm.nih.gov/uts/umls/concept/C0851162
https://uts.nlm.nih.gov/uts/umls/concept/C0851162
https://uts.nlm.nih.gov/uts/umls/concept/C0851162
https://uts.nlm.nih.gov/uts/umls/concept/C0851162
Varicella zoster
Cardiopulmonary
https://uts.nlm.nih.gov/uts/umls/concept/C0553534
B-Disease
Neuronal cancer
https://uts.nlm.nih.gov/uts/umls/concept/C0007634
CLSPN gene
https://uts.nlm.nih.gov/uts/umls/concept/C1426354
B-Gene_or_gene_product
h4k16
ns1:disease_may_have_associated_disease
foaf:name
rdf:type
ns1:disease_has_abnormal_cell

Fig. 2. Visualization of a subgraph of the RDF graph extracted from 10 CORD-19 papers. The subgraph contains examples of Disease entities (Cardiopulmonary, varicella zoster) and other examples of B-Gene_or_gene_product entities (CLSPN gene, h4k16) and relations (ns1:disease_has_abnormal_cell, ns1:disease_may_have_associated_disease).

We preserve the test set of BC5CDR [36] to test and evaluate the model.

Genes or gene product: For this entity, we form our training set from the training and test sets of JNLPBA for genes [32] and the training set of BC2GM [43], and our test set from the test set of BC2GM [43].

Cell: We use the JNLPBA dataset [32] for cells with the standard distribution of the train and test sets.

Drug: DDICorpus [44] is a set of sentences with tagged drug entities and the relations between them in the form of XML documents. We retrieved the full text of the sentences and the tagged entities and transformed them into a unified format.

C. Entity Linking

Next, we feed the extracted entities into the entity linking step, applying scispaCy EntityLinker to each entity. The EnitityLinker is a spaCy pipeline component that relies on a Thinc model with a linear output layer and helps clear the disambiguation in the extracted entities by linking them to a knowledge base. In our case, we choose the entity linking component from the *en_core_sci_sm* scpCy pipeline, and we link our entities to UMLS approximately 3M concepts. As a premature cleaning step, we discarded entities that could not be linked, followed by predictions with a confidence score lower than a threshold of 70%. We also discard the extracted entities linked to a concept with a nonmedical type in UMLS, such as T092 (Organization) and T064 (Governmental or Regulatory Activity). We enrich the Entity class with two more properties, the CUI and the UMLS [2] label of the entity after performing EL.

D. Relation Extraction

1) Process: In this step, we aim to extract the relations among all entities in the corpus. We first predict the relations between every pair of entities at the sentence level to accomplish that. We use the framework OpenNRE [45], which contains models to perform RE. It allows leveraging CNN and BERT models in our use case, sentence-level RE. Results have shown that BERT outperforms CNN on all evaluation datasets. We treat this task as a multi-classification problem. We train a BERT model to predict the relation, from a set of relations, between all pairs of entities present in a single sentence. Next, we infer the model on each sentence from the corpus by feeding it the full text of the sentence and the positions of the head and tail entities. The model returns a prediction for each pair from the set of relations.

2) Datasets: Before applying our BERT model on the input text, we train the model on a personalized dataset derived from the training set of BioRel [46], a dataset containing 534277 sentences and 125 relations extracted using distant supervision from a comprehensive set of datasets such as SemEval-2010 Task 8, ACE 2003-2004, NYT, BC5CDR, BB3, SeeDev, GE4, i2b2 2010. Each sentence contains the text of the sentence, the relation, and the pair of head and tail entities, along with other properties irrelevant to our use case. Considering our application case on CORD-19 [28] papers, we limit the set of relations to a set containing five relations in total: *may_treat, disease_has_associated_gene*, *disease_may_have_associated_disease*, *disease_has_abnormal_cell*, and *chemical_or_drug_affects_gene_product*.

The choice of the relations is consistent with the specified entities as they represent the interactions that may occur among these entities. We construct our RE training set from 100 sentences corresponding to each relation and our RE testing set from 50 sentences for each relation.

E. Graph Cleaning

At this level, we possess a set of triples formed by the extracted entities from the NER and EL steps and the relations representing the interaction between each pair of these entities. Before constructing the final KG, we clean the predicted data as a continuance of the cleaning operation performed at the EL step. We discard the identified relations with a confidence score less than the 70% threshold to further clean the results from weak predictions. We also discard some relations based on medical logic. Each relation has a defined type for its head entity and a defined type for its tail entity, and any prediction that does not respect its respective pair of entity types is dropped (e.g., may_treat is a relation that can only be established between a Drug entity and a Disease entity, a prediction stating that a Drug entity may treat a Gene_or_gene_product entity is not taken into account).

F. Knowledge Graph Formalization

For the final step of the pipeline, we group the final set of triples into a Resource Description Format (RDF) graph, following the RDF standard format. The nodes are divided into two categories. The first category represents the entities linked successfully to their corresponding UMLS [2] concept, which are identified by the URI of the concept on UMLS. The second category represents the rest of the entities that were not linked to UMLS and are identified by a local node ID on the graph. This format allows our pipeline to be integrated into and paired with larger systems as a KGE component to perform other machine learning tasks such as graph completion and link prediction. Our output can also be exploited for ontology development and update, as the RDF/XML format is one of the standard representations of ontologies. Fig. 2 shows a visualization of a subgraph extracted from the output KG.

IV. Results and discussion

In this section, we evaluate the performance of our pipeline using two approaches. Firstly, we tackle the evaluation of individual trainable components in the NER and RE step using three of the metrics for classification models (precision, recall, F1 score) and compare these results to other state-of-the-art works mentioned in section II. Secondly, we tried a new approach to evaluate the performance of the whole pipeline. We feed the pipeline a biomedical text that we already know the entities and relations within. We compare the output to the ground truth data of the text and provide an overview of the quality of that output.

A. Named Entity Recognition

For this part, we train and test 4 separate BioBERT [24] models on four different datasets, each one corresponding to a single entity type. With the training epochs set to 3 for all models, the disease entities' model is trained on 26422 sentences and tested on 3941 sentences. The second model focused on drug entity extraction, and it was trained on 5288 sentences and tested on 597 sentences. A model was trained and tested on 7438 and 1695 sentences to recognize gene entities, respectively. The last model destined to extract cell entities was trained on 3032 sentences and trained on 1905 sentences. The evaluation results are extracted from the testing phase and are shown in Table III alongside other results from the state-of-the-art works mentioned above. We evaluate our model using micro precision, recall, and F1-score to account for the difference in class density in medical natural language text. The results show that our approach to tackling the NER problem is well positioned among the other solutions used by state-of-the-art methods for the same purpose.

TABLE III
COMPARISON BETWEEN THE RESULTS OF OUR APPROACH ON NER AND THE RESULTS OF OTHER RELATED WORK METHODS.

	Metrics			
Model		Precision	Recall	F1-score
Our approach (BioBERT)	Disease	0.9788	0.9776	0.97813
	Drug	0.9929	0.9929	0.9929
	Gene	0.9769	0.9765	0.9766
	Cell	0.9914	0.9915	0.9914
	Average	**0.985**	**0.9846**	**0.9847**
Gajendran et al. [30] (BERT-BiLSTM-CRF)	Disease	0.8849	0.8902	0.8876
	Chemical	0.9088	0.9225	0.9156
	Protein	0.7125	0.815	0.7606
	Average	0.8354	0.8759	0.8546
Linfeng et al. [19] (vocabulary-based bidirectional maximum matching + BiLSTM-CRF + pattern recognizer)	Disease Gender Age range Symptom Exam Lab exam Lab item Medicine Surgery	0.9727	0.9689	0.9708
Harnoune et al. [22] (BioClinicalBERT)	Patient Drug Posology ADE Reason			0.907
Lamy et al. [39] (cTAKES)	Clinical procedures Diseases Medications Symptoms Signs Anatomical regions	0.75	0.61	0.67

B. Relation extraction

We train the model on the training dataset derived from BioRel [46] over 20 epochs to prepare the BERT model for inference. The training set contains 500 sentences divided equally among the five relations. Then, we evaluate the model's performance on the constructed t est set containing 213 sentences. Our model's precision, recall, and F1-score are shown in Table IV, compared to other state-of-the-art works implementations of RE solutions for which the same evaluation metrics are available. Our approach has exceeded other state-of-the-art methods in the RE step of the KG extraction process.

TABLE IV
COMPARISON BETWEEN THE RESULTS OF OUR APPROACH ON RE AND THE RESULTS OF OTHER RELATED WORK METHODS.

	Metrics		
Model	Precision	Recall	F1-score
Our approach (BERT)	**0.9385**	**0.8442**	**0.8888**
Gajendran et al. [30] (SciBERT)	0.74	0.73	0.73
Harnoune et al. [22] (BioBERT)			**0.88**

C. Overall performance

Furthermore, after evaluating the pipeline components individually, we proceeded to analyze the performance of the pipeline as a whole. For this purpose, we manually select all the sentences (328 sentences) from the test set of BioREL [46] that contain the five relations. These sentences have the same structure as the ones in the training set mentioned in section III-D. We construct a natural language text by concatenating the text of all these sentences into one paragraph, and we keep their corresponding relations and pair of head and tail entities as golden truth data to compare it to the pipeline output at the end. After obtaining the final KG from our pipeline, we matched the extracted entities and the predicted relations on a sentence level. We analyzed the cases of the correct and wrong predictions as shown in Fig. 3. Our method correctly predicted 240 out of 328 sentences with a rate of 75%. Subsequently, it failed to do the same for the remaining 25% of the data. Upon further analysis, we noticed that 13% of the sentences were not predicted correctly because of a problem in the recognition of the entities phase, which means that the NER model did not extract the proper entities (words) from the sentences. 5% of the sentences had wrong predictions due to entity classification problems, as the NER model failed to predict the correct type for the extracted entities. Lastly, the pipeline failed to provide correct predictions for 7% of the sentences because of a RE problem linked to wrong predictions by the RE model.

V. APPLICATIONS AND CHALLENGES

A KGE pipeline, such as the one provided in this work, is an essential tool with many applications for both biomedical professionals and machine learning practitioners. It can

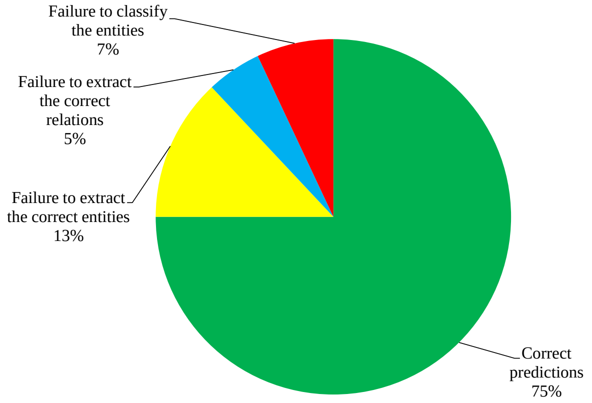

Fig. 3. Distribution of the predicted triples corresponding to the evaluation data

extract the knowledge stored in natural language text and transform it into a machine-readable and easy-to-browse format. It can also process large amounts of text in a limited time and combine the result into an interconnected output, relieving the end user of the heavy task of manually reading the whole text. Other systems, such as QA, recommendation, and knowledge discovery systems, can also exploit the output of this pipeline. Particularly in the biomedical field, it can be applied in different contexts to achieve specific results. For example, the KG can help drug discovery, where users can predict new links between drugs, diseases, gene mutations, and genetic markers. Decision support systems also use KGs as a knowledge base to assist clinicians in diagnosing complex cases by surfacing relevant symptoms, conditions, and treatment suggestions. KGE pipelines have a significant potential in ontology development, maintenance, and update. It can help ontology developers by accelerating text processing and entity and relation extraction.

The process of KGE still faces many challenges, some of which we faced during the development of this pipeline. Firstly, processing natural language text will always pose the challenge of ambiguity, polysemy, long-range dependencies, and context complexity. Other challenges can be traced back to the pipeline's components. For example, a small margin of error in the early steps of the process can be propagated to the following steps, resulting in a significant gap from the desired results. The integration of these steps also greatly impacts the overall performance and needs attention when choosing the adequate tools that work well together and the correct data to pass from one component to the next. On another note, employing supervised learning tools brings another challenge: securing important amounts of clean and correct training data, which may only sometimes be available in some fields. Applying KGE in the biomedical domain faces the same challenges as scientific and clinical text, which is more complex and ambiguous, and significant amounts of data are unavailable due to privacy and ethical concerns.

VI. Conclusion

In this work, we address the problem of knowledge extraction from natural language text. Specifically, we focus on Knowledge Graph extraction from biomedical data. For this purpose, we review some fundamental text mining concepts required for KG construction. Then, we review some state-of-the-art works that tackled the same problem by providing full pipelines to process biomedical text. Lastly, we introduce an end-to-end pipeline to process and transform natural language text into a KG.

Our approach provides a level of customization through its trainability on different data depending on the application use case. In our case, this process was applied to four biomedical concepts (Disease, Drug, Gene, Cell) and their possible interactions, influencing the choice of certain aspects of the pipeline's components. The pipeline achieved F1 scores ranging from 0.89 to 0.96 across all classes in NER and 0.88 in RE. These results are not only positioned well compared to other state-of-the-art works but also improvable with more ablation studies.

Our future work will proceed in 2 directions: Firstly, we will focus on the performance of the pipeline by analyzing its weak points and enhancing each component accordingly. Secondly, our work so far returns a KG that has various benefits to biomedical domain practitioners. However, this is not the end goal in the text mining field. For that, the output KG is set to be paired with question-answering, recommendation, decision support, and querying tools to provide more value to the end user.

References

[1] A. R. Aronson, "Effective mapping of biomedical text to the umls metathesaurus: The metamap program," *Proceedings. AMIA Symposium*, pp. 17–21, 2001.

[2] Bodenreider O., "The unified medical language system (umls): integrating biomedical terminology," *Nucleic acids research, 32(Database issue), D267–D270*, 2004.

[3] K. Fukuda, A. Tamura, T. Tsunoda, and T. Takagi, "Toward information extraction: identifying protein names from biological papers," *Pac Symp Biocomput*, pp. 707–18, 1998.

[4] Tamames J., "Text detective: a rule-based system for gene annotation in biomedical texts." *BMC Bioinformatics*, vol. 6 (Suppl 1), 2005.

[5] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "Dbpedia: A nucleus for a web of open data," in *The Semantic Web*, e. a. Aberer K., Ed., vol. 4825. Heidelberg: Springer Berlin Heidelberg, 2007, pp. 722–735.

[6] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: A collaboratively created graph database for structuring human knowledge," in *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '08. New York, NY, USA: Association for Computing Machinery, 2008, p. 1247–1250.

[7] D. Vrandečić and M. Krötzsch, "Wikidata: A free collaborative knowledgebase," *Commun. ACM*, vol. 57, no. 10, p. 78–85, sep 2014.

[8] C. E. Lipscomb, "Medical subject headings (mesh)," *Bull Med Libr Assoc.*, 2000, 88(3): 265–266.

[9] I. Yamada, K. Washio, H. Shindo, and Y. Matsumoto, "Global entity disambiguation with BERT," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, M. Carpuat, M.-C. de Marneffe, and I. V. Meza Ruiz, Eds. Seattle, United States: Association for Computational Linguistics, jul 2022, pp. 3264–3271.

[10] N. D. Cao, G. Izacard, S. Riedel, and F. Petroni, "Autoregressive entity retrieval," 2021.

[11] K. Nebhi, "A rule-based relation extraction system using dbpedia and syntactic parsing," in *Proceedings of the 2013th International Conference on NLP & DBpedia - Volume 1064*, ser. NLP-DBPEDIA'13. Aachen, DEU: CEUR-WS.org, 2013, p. 74–79.

[12] G. Zhou, J. Su, J. Zhang, and M. Zhang, "Exploring various knowledge in relation extraction," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, K. Knight, H. T. Ng, and K. Oflazer, Eds. Ann Arbor, Michigan: Association for Computational Linguistics, Jun. 2005, pp. 427–434.

[13] B. Min, S. Shi, R. Grishman, and C.-Y. Lin, "Towards large-scale unsupervised relation extraction from the web," *Int. J. Semant. Web Inf. Syst.*, vol. 8, no. 3, p. 1–23, jul 2012.

[14] Mausam, M. Schmitz, S. Soderland, R. Bart, and O. Etzioni, "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*, J. Tsujii,

J. Henderson, and M. Paşca, Eds. Jeju Island, Korea: Association for Computational Linguistics, jul 2012, pp. 523–534. [Online]. Available: https://aclanthology.org/D12-1048

[15] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, "Relation classification via convolutional deep neural network," in *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, J. Tsujii and J. Hajic, Eds. Dublin, Ireland: Dublin City University and Association for Computational Linguistics, Aug. 2014, pp. 2335–2344. [Online]. Available: https://aclanthology.org/C14-1220

[16] Y. Li, Z. Zhong, and N. Jing, "Multi-path convolutional neural network for distant supervised relation extraction," in *Proceedings of the 2nd International Conference on Computer Science and Application Engineering*, ser. CSAE '18. New York, NY, USA: Association for Computing Machinery, 2018.

[17] S. Zhang, D. Zheng, X. Hu, and M. Yang, "Bidirectional long short-term memory networks for relation classification," in *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation*, H. Zhao, Ed., Shanghai, China, Oct. 2015, pp. 73–78. [Online]. Available: https://aclanthology.org/Y15-1009

[18] Q. Wei, Z. Ji, Y. Si, J. Du, J. Wang, F. Tiryaki, S. T.-I. Wu, C. Tao, K. Roberts, and H. Xu, "Relation extraction from clinical narratives using pre-trained language models," *AMIA ... Annual Symposium proceedings. AMIA Symposium*, vol. 2019, pp. 1236–1245, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:216030150

[19] L. Li, P. Wang, J. Yan, Y. Wang, S. Li, J. Jiang, Z. Sun, B. Tang, T.-H. Chang, S. Wang, and Y. Liu, "Real-world data medical knowledge graph: construction and applications," *Artificial Intelligence in Medicine*, vol. 103, p. 101817, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0933365719309546

[20] L. Li, P. Wang, Y. Wang, J. Jiang, B. Tang, J. Yan, S. Wang, and Y. Liu, "Prtransh: Embedding probabilistic medical knowledge from real world emr data," *ArXiv*, vol. abs/1909.00672, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:202122500

[21] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, ser. KDD'96, 1996, p. 226–231.

[22] A. Harnoune, M. Rhanoui, M. Mikram, S. Yousfi, Z. Elkaimbillah, and B. El Asri, "Bert based clinical knowledge extraction for biomedical knowledge graph construction and analysis," *Computer Methods and Programs in Biomedicine Update*, vol. 1, p. 100042, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666990021000410

[23] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran, and T. Solorio, Eds. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. [Online]. Available: https://aclanthology.org/N19-1423

[24] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "BioBERT: a pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, 09 2019. [Online]. Available: https://doi.org/10.1093/bioinformatics/btz682

[25] E. Alsentzer, J. Murphy, W. Boag, W.-H. Weng, D. Jindi, T. Naumann, and M. McDermott, "Publicly available clinical BERT embeddings," in *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, A. Rumshisky, K. Roberts, S. Bethard, and T. Naumann, Eds. Minneapolis, Minnesota, USA: Association for Computational Linguistics, Jun. 2019, pp. 72–78. [Online]. Available: https://aclanthology.org/W19-1909

[26] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," 2019.

[27] C. Wise, V. N. Ioannidis, M. C. Rebollar, X. Song, G. D. Price, N. Kulkarni, R. Brand, P. Bhatia, and G. Karypis, "Covid-19 knowledge graph: Accelerating information retrieval and discovery for scientific literature," *ArXiv*, vol. abs/2007.12731, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:220793196

[28] L. L. Wang, K. Lo, Y. Chandrasekhar, R. Reas, J. Yang, D. Burdick, D. Eide, K. Funk, Y. Katsis, R. M. Kinney, Y. Li, Z. Liu, W. Merrill, P. Mooney, D. A. Murdick, D. Rishi, J. Sheehan, Z. Shen, B. Stilson, A. D. Wade, K. Wang, N. X. R. Wang, C. Wilhelm, B. Xie, D. M. Raymond, D. S. Weld, O. Etzioni, and S. Kohlmeier, "CORD-19: The COVID-19 open research dataset," in *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, K. Verspoor, K. B. Cohen, M. Dredze, E. Ferrara, J. May, R. Munro, C. Paris, and B. Wallace, Eds. Online: Association for Computational Linguistics, Jul. 2020. [Online]. Available: https://aclanthology.org/2020.nlpcovid19-acl.1

[29] D. Andrzejewski and X. Zhu, "Latent dirichlet allocation with topic-in-set knowledge," *CiteSeer X (The Pennsylvania State University)*, 01 2009.

[30] S. Gajendran, D. Manjula, V. Sugumaran, and R. Hema, "Extraction of knowledge graph of covid-19 through mining of unstructured biomedical corpora," *Comput Biol Chem*, vol. 102, pp. 107 808–107 808, 02 2023.

[31] R. I. Doğan, R. Leaman, and Z. Lu, "NCBI disease corpus: a resource for disease name recognition and concept normalization," *J Biomed Inform*, vol. 47, pp. 1–10, Jan. 2014.

[32] J.-D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii, "GENIA corpus–semantically annotated corpus for bio-textmining," *Bioinformatics*, vol. 19 Suppl 1, pp. i180–2, 2003.

[33] Martin Krallinger et al., "The chemdner corpus of chemicals and drugs and its annotation principles," *Journal of Cheminformatics*, vol. 7, pp. S2–S2, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:27996

[34] I. Beltagy, K. Lo, and A. Cohan, "SciBERT: A pretrained language model for scientific text," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, K. Inui, J. Jiang, V. Ng, and X. Wan, Eds. Hong Kong, China: Association for Computational Linguistics, nov 2019, pp. 3615–3620. [Online]. Available: https://aclanthology.org/D19-1371

[35] F. Li, M. Zhang, G. Fu, and D.-H. Ji, "A neural joint model for entity and relation extraction from biomedical text," *BMC Bioinformatics*, vol. 18, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:2003493

[36] J. Li, Y. Sun, R. J. Johnson, D. Sciaky, C.-H. Wei, R. Leaman, A. P. Davis, C. J. Mattingly, T. C. Wiegers, and Z. Lu, "Biocreative v cdr task corpus: a resource for chemical disease relation extraction," *Database: The Journal of Biological Databases and Curation*, 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:88817

[37] M. Krallinger, O. Rabal, S. A. Akhondi, M. P. Pérez, J. Santamaría, G. P. Rodríguez, G. Tsatsaronis, A. Intxaurrondo, J. A. B. López, U. K. Nandal, E. M. van Buel, A. Chandrasekhar, M. Rodenburg, A. Lægreid, M. A. Doornenbal, J. Oyarzábal, A. Lourenço, and A. Valencia, "Overview of the biocreative vi chemical-protein interaction track," 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:13690520

[38] G. Ji, S. He, L. Xu, K. Liu, and J. Zhao, "Knowledge graph embedding via dynamic mapping matrix," 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)*, C. Zong and M. Strube, Eds. Beijing, China: Association for Computational Linguistics, jul 2015, pp. 687–696. [Online]. Available: https://aclanthology.org/P15-1067

[39] M. Lamy, R. Pereira, J. C. Ferreira, F. Melo, and I. Velez, "Extracting clinical knowledge from electronic medical records," *IAENG International Journal of Computer Science*, vol. 45, no. 3, p. 488–493, 2018.

[40] G. Savova, J. Masanz, P. Ogren, J. Zheng, S. Sohn, and K. Kipper-Schuler, "Mayo clinical text analysis and knowledge extraction system (ctakes): Architecture, component evaluation and applications," *Journal of the American Medical Informatics Association : JAMIA*, vol. 17, pp. 507–13, 09 2010.

[41] S. E. Ech-chorfi and E. Zemmouri, "Mining the cord-19: Review of previous work and design of topic modeling pipeline," in *Artificial Intelligence and Industrial Applications*, T. Masrour, H. Ramchoun, T. Hajji, and M. Hosni, Eds. Cham: Springer Nature Switzerland, 2023, pp. 411–426.

[42] A. Agrawal, S. Tripathi, M. Vardhan, V. Sihag, G. Choudhary, and N. Dragoni, "Bert-based transfer-learning approach for nested named-entity recognition using joint labeling," *Applied Sciences*, vol. 12, no. 3, 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/3/976

[43] Larry L. Smith et al., "Overview of biocreative ii gene mention recognition," *Genome Biology*, vol. 9, pp. S2–S2, 2008. [Online]. Available: https://api.semanticscholar.org/CorpusID:215780186

[44] M. Herrero-Zazo, I. Segura-Bedmar, P. Martínez, and T. Declerck, "The ddi corpus: An annotated corpus with pharmacological substances and drug–drug interactions," *Journal of Biomedical Informatics*, vol. 46, no. 5, pp. 914–920, 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1532046413001123

[45] X. Han, T. Gao, Y. Yao, D. Ye, Z. Liu, and M. Sun, "OpenNRE: An open and extensible toolkit for neural relation extraction," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, S. Padó and R. Huang, Eds. Hong Kong,

China: Association for Computational Linguistics, nov 2019, pp. 169–174. [Online]. Available: https://aclanthology.org/D19-3029

[46] R. Xing, J. Luo, and T. Song, “Biorel: A large-scale dataset for biomedical relation extraction,” in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2019, pp. 1801–1808.