2024

LANGUAGE MODELS FOR RARE DISEASE INFORMATION EXTRACTION: EMPIRICAL INSIGHTS AND MODEL COMPARISONS

Shashank Gupta
University of Kentucky, gupta.shashank1@outlook.com
Author ORCID Identifier:
https://orcid.org/0009-0003-8496-5271
Digital Object Identifier: https://doi.org/10.13023/etd.2024.77

STUDENT AGREEMENT:

I represent that my thesis or dissertation and abstract are my original work. Proper attribution has been given to all outside sources. I understand that I am solely responsible for obtaining any needed copyright permissions. I have obtained needed written permission statement(s) from the owner(s) of each third-party copyrighted matter to be included in my work, allowing electronic distribution (if such use is not permitted by the fair use doctrine) which will be submitted to UKnowledge as Additional File.

I hereby grant to The University of Kentucky and its agents the irrevocable, non-exclusive, and royalty-free license to archive and make accessible my work in whole or in part in all forms of media, now or hereafter known. I agree that the document mentioned above may be made available immediately for worldwide access unless an embargo applies.

I retain all other ownership rights to the copyright of my work. I also retain the right to use in future works (such as articles or books) all or part of my work. I understand that I am free to register the copyright to my work.

REVIEW, APPROVAL AND ACCEPTANCE

The document mentioned above has been reviewed and accepted by the student's advisor, on behalf of the advisory committee, and by the Director of Graduate Studies (DGS), on behalf of the program; we verify that this is the final, approved version of the student's thesis including all changes required by the advisory committee. The undersigned agree to abide by the statements above.

Shashank Gupta, Student

Dr. Ramakanth Kavuluru, Major Professor

Dr. Simone Silvestri, Director of Graduate Studies

LANGUAGE MODELS FOR RARE DISEASE INFORMATION EXTRACTION: EMPIRICAL INSIGHTS AND MODEL COMPARISONS

THESIS

A thesis submitted in partial fulfillment of the
requirements for the degree of Master of Science in the
College of Engineering
at the University of Kentucky

By

Shashank Gupta

Lexington, Kentucky

Director: Dr. Ramakanth Kavuluru

Lexington, Kentucky

2024

https://orcid.org/0009-0003-8496-5271

ABSTRACT OF THESIS

LANGUAGE MODELS FOR RARE DISEASE INFORMATION EXTRACTION: EMPIRICAL INSIGHTS AND MODEL COMPARISONS

End-to-end relation extraction (E2ERE) is a crucial task in natural language processing (NLP) that involves identifying and classifying semantic relationships between entities in text. This thesis compares three paradigms for end-to-end relation extraction (E2ERE) in biomedicine, focusing on rare diseases with discontinuous and nested entities. We evaluate Named Entity Recognition (NER) to Relation Extraction (RE) pipelines, sequence to sequence models, and generative pre-trained transformer (GPT) models using the RareDis information extraction dataset. Our findings indicate that pipeline models are the most effective, followed closely by sequence-to-sequence models. GPT models, despite having eight times as many parameters, perform worse than sequence-to-sequence models and significantly lag pipeline models. Our results also hold for a second E2ERE dataset for chemical-protein interactions. Our study is the first to conduct E2ERE for the RareDis dataset, and all dataset and code used in our experiments are publicly available at https://github.com/shashank140195/Raredis.

KEYWORDS: Natural Language Processing, End-to-end Relation Extraction, Large Language Models, GPT, Rare Diseases, Sequence-to-sequence model

Shashank Gupta

[04/24/2024]

Date

LANGUAGE MODELS FOR RARE DISEASE INFORMATION EXTRACTION: EMPIRICAL INSIGHTS AND MODEL COMPARISONS

By
Shashank Gupta

Dr. Ramakanth Kavuluru
Director of Thesis

Dr. Simone Silvestri
Director of Graduate Studies

[04/24/2024]
Date

DEDICATION

To my beloved family, whose unwavering support has been my anchor in the deepest waters. To my parents, whose love and sacrifices are the foundation upon which I stand. To my siblings, whose belief in me lights the darkest paths. This achievement is a tribute to your endless faith, a testament to the strength we hold together.

ACKNOWLEDGMENTS

I am immensely grateful to a few people whose support and guidance have been invaluable throughout the course of this project.

First and foremost, I extend my deepest gratitude to my advisor, Dr. Ramakanth Kavuluru, whose expertise and insightful feedback have been instrumental in shaping this work. Their patience and encouragement have been a constant source of inspiration.

My heartfelt thanks go to my committee members, Dr. Cody Bumgardner and Dr. A.B. Siddique, for their valuable suggestions and constructive criticism, which have significantly contributed to the refinement of this work.

Special thanks to my colleagues and friends in the Department of Computer Science and NLP lab of Bioinformatics for their camaraderie and support. Their presence has made this journey both enjoyable and memorable.

I am deeply grateful to my family, who have provided unwavering support and encouragement throughout my academic pursuits. Their love and belief in me have been the pillars of my strength.

Lastly, I am thankful to the NIH for providing the financial support necessary to complete this project.

This accomplishment would not have been possible without the collective support and encouragement of all these individuals.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1. INTRODUCTION

The last few years have seen a significant increase in biology and medical studies and due to this, there is now a huge amount of health-related data available on the internet for people to use and study. As biomedical data is rapidly growing, the challenge is to come up with methods that help us make sense of this information. Turning this information from unstructured text into a structured form by hand is a time-consuming and hard task. As a solution, one way of extracting information is through data mining, which is the process of finding new patterns and relationships in large sets of data that we didn't know about before. Another way to get information from biomedical data is by using Artificial Intelligence (AI). AI is a term that encompasses the use of computers to mimic intelligent behavior with minimal human intervention which, in the subsequent years, became the foundation of Natural language processing (NLP). NLP emerged in the 1950s which deals with the interaction between computers and humans through natural language. It involves the development of algorithms and models that enable people to interact with computers using their everyday language. This makes it easier for people to interact with technology resulting in an overwhelming amount of textual data generated every day. NLP enables the automated processing and understanding of this vast amount of unstructured text, allowing for the extraction of valuable insights and information.

NLP has two main subcategories or tasks: **Natural language Understanding** (NLU) and **Natural language generation** (NLG) (Reiter and Dale 2000). The NLU module transforms the natural language into a structured format (formal representation) to derive meaning from the input. An NLG module may/may not utilize this structured format

to produce the response for the desired task (Tseng, et al. 2020). NLU and NLG are two independent tasks that can work individually or can be combined. Tasks such as text classification and sentiment analysis are subcategories of NLU in comparison to recent sequence-to-sequence tasks such as machine translation and text summarization which fall under NLG (Nadkarni 2011) (Dreisbach 2019) (Baby 2017) (Liddy 2001).

When utilized in the biomedical field, NLP is revolutionizing the healthcare. The use of NLP in medicine spans various areas encompassing research, direct patient care, diagnostics, clinical coding, and patient-facing interfaces. For example, NLP has shown success in diagnostic settings, where it has been used to classify radiology reports to determine the appropriate clinical response, thereby reducing the need for human input (Kolanu 2020) (Swartz 2017). NLP can be combined with computer vision tasks as demonstrated by (Gupta, Jiang and Imran 2023) in their work of creating medical reports from chest-Xray images. **Hence, we can leverage NLP for various medical problems and in this thesis, the focus is on applying NLP techniques for rare diseases**.

The National Institutes of Health (NIH) estimates that around 7,000 rare diseases impact between 25 and 30 million Americans, which translates to approximately 1 out of every 10 Americans. ((NORD 2019) Around 95% of the known rare diseases currently lack any treatment options ((NORD 2019). Due to the scarcity of these diseases, it can be a challenge to diagnose and treat—nearly 95% of rare diseases have no known cure, and the number of drugs available for treating these conditions are limited to 100 (Klimova, et al. 2017). The average diagnostic delay is around seven years (Global Genes n.d.). Many rare diseases are genetic in nature and are caused by mutations in a single gene. However, because there are thousands of rare diseases, each with unique symptoms and genetic

causes, developing effective treatments can be a significant challenge. Developing a structured compendium of information about rare diseases has the potential to help expedite search, discovery, and hypothesis generation for these conditions.

Manually building and updating a knowledge base from existing publications on rare diseases is difficult for biomedical researchers. Simple indexing and keyword searching are not enough for complex searches. Therefore, automatic methods that can understand human languages and find important information are becoming crucial for managing biomedical information. This task involves identifying terms like diseases, drugs, genes etc., and extracting information about what is said or predicted about these terms, including relationships and inferences among biomedical substances and other hidden information.

BioNLP, or biomedical natural language processing, stands as a specialized subfield of NLP that has taken on the task of extracting information from the rapidly expanding corpus of biomedical text. The impetus behind BioNLP is clear: a typical hospital generates 50 petabytes of data every year and with medical data doubling every 73 days, according to World Economic forum 2024 traditional manual methods for data analysis are no longer viable (Forum 2024). This huge amount of data includes clinical notes, lab tests, medical images, sensor readings, genomics, and operational and financial information. However, 97% of all the data produced by hospitals globally each year is not used (Forum 2024). Researchers in BioNLP are developing and applying sophisticated algorithms to parse and understand the language of life sciences. These tools are becoming essential in extracting meaningful patterns from datasets that are too vast for human analysis. The practical applications of BioNLP are extensive. It enhances drug discovery

processes by identifying potential drug targets and adverse drug reactions. In precision medicine, BioNLP supports the tailoring of treatments based on a patient's genetic profile by extracting relevant information from their genomic data. Additionally, in the context of public health, it enables the monitoring and analysis of disease outbreaks by sifting through vast amounts of epidemiological data.

Despite these advances, BioNLP still faces considerable challenges. The specialized language of biomedicine, with its ever-growing lexicon, nuances, and context-dependent meanings, presents an ongoing challenge for NLP systems. As it continues to evolve, the focus is also shifting toward improving the interpretability and trustworthiness of NLP models. This is particularly critical in medicine, where decisions based on algorithmic recommendations can have profound implications on patient care and outcomes. Moreover, the ethical use of patient data for NLP training necessitates rigorous privacy safeguards and transparent data governance policies.

In sum, BioNLP has become an essential catalyst in the processing and utilization of biomedical information, driving innovations in healthcare and advancing our understanding of complex biological systems. It stands not only as a testament to the power of artificial intelligence but also as a beacon for its future potential in transforming the landscape of biomedicine.

As mentioned earlier, there are many tasks in the field of biomedical NLP that are being extensively being worked on. In this thesis, we focus on a task called **relation extraction (RE)** for rare diseases. CHAPTER 2 presents an exposition on RE literature including efforts well known in BioNLP community. CHAPTER 3 introduces the rare disease RE dataset we use in this thesis. CHAPTER 4 covers all the methods we developed

and used for rare disease RE. In CHAPTER 5, we discuss training objectives and RE evaluation metrics. After elaborating on training configurations in CHAPTER 6, we present and discuss our main results in CHAPTER 7 including a preliminary error analysis. We hint at potential future directions in CHAPTER 8 to further improve upon our main results from this thesis.

CHAPTER 2. END-TO-END RELATION EXTRACTION

In the digital age, the exponential growth of textual data in the form of news articles, publications, blogs, and reports has underscored the necessity of developing techniques for automatic information extraction. Information extraction techniques are pivotal for enhancing access to and management of the knowledge hidden within extensive text corpora. The information that users seek to extract from documents typically falls into two categories: **named entities** and **relations**.

A **named entity (NE)** is usually a word or phrase that denotes a specific real-world existence, such as a person, organization, location, or a disease. **Named Entity Recognition (NER)** is the task dedicated to identifying all span/mentions of a particular entity type in each set of documents. This process serves as a foundational step in various natural language processing (NLP) applications, including information retrieval, question answering, and knowledge base construction.

On the other hand, a **relation** typically signifies a well-defined connection/link between two or more entities. A relation is typically expressed as a triple that has a subject entity and an object entity connected via a predicate (or relation type) as in the example (subject: atorvastatin, predicate: treats, object: hyperlipidemia). The task of **RE** involves identifying mentions of such relations of interest within each sentence of the given documents. This task is vital for constructing semantic networks and for populating relational databases from unstructured text sources.

2.1 End-to-end Relation Extraction

Many RE efforts in the past assume that the entity spans are already provided as part of the input and hence addressed an easier problem of relation classification (RC) (Zeng, et al. 2014) (Zhou, et al. 2016) (Kavuluru, Rios and Tran 2017). However, a more realistic setting is the ability to extract both entity spans and associated relations from the raw text where entities are not provided. The integration of NER and RE tasks leads to the concept of **End-to-End Relation Extraction (E2ERE)**. E2ERE aims to simultaneously identify named entities and their relations within a text, thereby streamlining the information extraction process. With the recent deluge of deep neural networks (or deep learning methods), the NLP community has been focusing more on E2ERE efforts (Miwa and Bansal 2016) (Zhang, Zhang and Fu 2017) (Pawar, Bhattacharyya and Palshikar 2017) (Tran and Kavuluru 2018). Efforts have also been expanded from single sentence E2ERE to a more complex setting of extractions at the document level, involving cross-sentence relations, where entities expressed in different sentences are to be linked (Peng, et al. 2017) (Yao, et al. 2019).

2.2 Biomedical Relation Extraction

Cohen and Hunter (K. a. Cohen 2008) highlight the exponential growth of biomedical literature, particularly in PubMed/MEDLINE publications. Manually converting this vast amount of information into a structured format is also extremely difficult due to the sheer volume of publications. Automatic RE can be a valuable tool in

addressing this issue. It has the potential to reduce the time researchers spend reviewing literature. Additionally, it can cover a significantly larger number of scientific articles than what is typically reviewed, making the process more efficient.

Disease and treatment mechanisms are often driven at the biological level by protein-protein and chemical-protein interactions while clinical relations such as drug-disease treatment relations and disease-symptom causative relations are helpful in providing care. Most new relational information is first discussed in textual narratives (e.g., scientific literature, clinical notes, or social media posts), and extracting and storing it as triples enable effective search systems (Dietze and Schroeder 2009), high-level reasoning, hypothesis generation, and knowledge discovery applications (Henry and McInnes 2017). According to Craven and Kumlien (Craven and Kumlien 1999) and Xu and Wang (Wang 2014), biomedical literature and clinical narratives contain a wealth of interactions between entities mentioned in the text. These interactions can be valuable for various applications, including bio-molecular information extraction, pharmacogenomics, and identifying drug-drug interactions (DDIs), as noted by Luo et al. (Luo, Uzuner and Szolovits 2017). This rapid expansion presents a promising research opportunity for applying information and data mining techniques. As such, NER and RE have become standard tasks in biomedical natural language processing (BioNLP) (Kilicoglu, et al. 2020).

In the biomedical domain, RE is challenging because there is a shortage of labeled data, and the costs of annotation are high, requiring the expertise of domain specialists (Amin, et al. 2022). Additional intricacies arise when named entities are discontinuous or when their spans overlap (Li, et al. 2021). For example, consider the string "**accumulation**

of fats (lipids) called GM 2 gangliosides," where entity span "**accumulation of GM 2 gangliosides**" is discontinuous with a gap involving outside words. In the example phrase "**central pain syndrome**" both the full three-word string and the middle word "**pain**" can constitute two different entities, where the latter entity is fully nested in the longer 3-word entity.

2.3 A History of methods for Relation extraction

RE has been around since the 1980s and has developed over time. The Message Understanding Conference (MUC) was held seven times from 1987 to 1998, specifically in the years 1991 (MUC3) (MUC3 '91: Proceedings of the 3rd conference on Message understanding 1991), 1992 (MUC4) (MUC4 '92: Proceedings of the 4th conference on Message understanding 1992), 1993 (MUC5) (MUC5 '93: Proceedings of the 5th conference on Message understanding 1993), 1995 (MUC6) (MUC6 '95: Proceedings of the 6th conference on Message understanding 1995), and 1998 (MUC7) (Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29 - May 1, 1998 1998). These conferences played a big role in advancing Relation Extraction technology. The task of Named Entity Recognition was introduced in the sixth Message Understanding Conference (MUC-6) in 1995.

In recent years, many methods have been suggested to extract relationships from biomedical text. These methods range from simple rule-based pattern approaches to more complex ones that use deep neural networks. These approaches vary in several ways, such as how they use NLP techniques to analyze the input text and the methods they use to learn

extraction rules. The techniques used in these systems can be divided into various groups: **co-occurrence, pattern or rule-based, machine learning (ML)-based, Neural Network based** and last **Large Language models-based** approaches. In the following sections, we will discuss the most common characteristics of these methods.

The evolution of relation extraction is rooted in rule-based systems, wherein early efforts were guided by manually crafted rules derived from linguistic and domain-specific insights (Riloff and others 1993) (Appelt, et al. 1993) (Brin 1998) (Muslea and others 1999) (Ciravegna 2001) (Shaalan and Raza 2008) (Agichtein and Gravano 2000) (Jayram, et al. 2006) (Shen, et al. 2007)). These methods are also known as hand-built pattern methods. In these methods, a set of extraction patterns is defined for a predetermined set of relations. These extraction patterns are then matched with the text. If a pattern matches, then a relation corresponding to that pattern is identified in the text. These systems aimed to identify and classify semantic relations within the literature. One major drawback of these methods is that when moving to a new domain, you need to redefine a new set of target relations and either create new extraction patterns or provide new hand-labeled training examples. This reliance on human involvement is a significant disadvantage of these approaches.

As researchers experimented with different techniques to extract meaningful information from text, parallel to the rule-based, they came up with co-occurrence method, which is the easiest way to find a relationship between two things that appear in the same sentence, abstract, or document. This method assumes that if two things are often mentioned together, they probably have some connection. However, since two things can be mentioned together without any real connection, most systems use a scoring method

based on how often they appear together to ignore those connections that just happen by chance. The rarer the connection is, the more the system thinks the two things are related. Co-occurrence methods usually find a lot of connections but might not be very accurate since biomedical texts often have complex sentences with many things mentioned, but only a few of those things are connected. In co-occurrence-based methods, the strength of the relationship between two entities is typically measured by some statistical metric, such as pointwise mutual information (PMI) or the Jaccard index. These metrics quantify how much more frequently the entities appear together than would be expected by chance. As the field developed, the strictness of pure rule-based systems was replaced by more flexible approaches, paving the way for the use of more advanced techniques. Even though rule-based and co-occurrence methods have been mostly overshadowed by the advanced computational methods, the early systems were essential. They provided important steps toward the sophisticated, AI/ML driven relation extraction systems we use today.

The next significant step in the evolution of RE methods was the incorporation of machine learning techniques. This shift represented a move away from the rigid, rule-based systems towards more flexible and scalable approaches. Machine learning allowed for the automatic identification of patterns in annotated corpora, making it possible to generalize from specific examples and thus, handle the growing volume and complexity of biomedical literature. One of the early applications of ML in biomedical RE was demonstrated by Cohen and Hersh in 2005, who explored the use of various machine learning algorithms, including Support Vector Machines (SVMs) and Decision Trees, for text mining tasks in biomedicine. Their work underscored the potential of ML to automate the extraction process and improve the efficiency and accuracy of information retrieval

(A. a. Cohen 2005). The K-means clustering approach proposed by Chen et al. (2005) utilizes a feature selection mechanism based on frequency and entropy. This mechanism helps in identifying important context words for different types of relations. Despite the advances brought about by ML, challenges remained, particularly in the form of data scarcity. The reliance on annotated datasets for supervised learning posed a bottleneck, as the manual annotation of biomedical texts was time-consuming and expensive. This led researchers to explore unsupervised and semi-supervised learning approaches, which could leverage unannotated data to overcome the limitations of purely supervised methods.

NLP methods for RE further expanded by leveraging syntactic phenomena of biomedical texts. This progress met the demand for more advanced methods to deal with the linguistic complexity of biomedical texts. It enabled the extraction of more detailed and contextually relevant information by exploiting the grammatical structure (syntactic parse) of sentences, facilitating the identification of relationships between entities. One notable example of NLP integration is the work of Krallinger et al. (2008), who highlighted the importance of combining text mining and information extraction techniques for linking genes to literature. Their approach leveraged NLP to improve the extraction of gene-disease associations, demonstrating the potential of NLP to enhance the accuracy and depth of biomedical RE (Krallinger, Valencia and Hirschman 2008). The use of NLP also allowed for the development of more sophisticated feature extraction methods, which were crucial for machine learning models. By encoding linguistic and semantic information as features, these models could better capture the context and meaning of relationships in the text. Furthermore, the integration of NLP facilitated the move towards more advanced machine learning and deep learning models. Techniques such as word embeddings, which

represent words in a continuous vector space, enabled the capturing of semantic similarities between terms, further improving the performance of RE systems. Overall, the addition of NLP techniques was a key step in the development of biomedical RE. It improved the field's ability to extract meaningful information from complex biomedical texts and set the stage for the creation of more advanced computational methods.

The arrival of deep learning marked a new era in biomedical RE. It used advanced neural network architectures to understand complex patterns and relationships in the data. This change led to more precise and detailed extraction of relationships from biomedical texts, setting the stage for more advanced and automated analysis in the field. Deep neural network architectures, such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and Gated Recurrent Units (GRUs), have been instrumental in advancing biomedical RE. These architectures are capable of handling complex data patterns and dependencies, making them highly effective for extracting meaningful relationships from biomedical texts.

Convolutional Neural Networks (CNNs), traditionally used in image processing, also found application in RE. CNNs were particularly adept at identifying local patterns and were used to extract features from sentences. (Zeng, et al. 2014) illustrated the potential of CNNs in relation classification, highlighting their ability to capture semantic and syntactic features from text without the need for explicit feature engineering. In this approach, the RE task is viewed as a multi-class classification problem. The model is designed to assign a specific relation class to a sentence that contains a pair of mentioned entities. CNNs are great at picking up patterns that are close together, but they're not so good at understanding patterns that are far apart or in a sequence over time. To address

this limitation, Recurrent Neural Networks were among the first deep learning models to be applied to RE. Their ability to process sequential data made them particularly well-suited for handling textual information. RNNs could capture dependencies between words in a sentence, allowing for a better understanding of the context in which biomedical entities and their relations appeared. However, standard RNNs were plagued by issues such as the vanishing gradient problem, which made it challenging to capture long-range dependencies in text. This limitation was addressed by the introduction of Long Short-Term Memory (LSTM) networks, a variant of RNNs. LSTMs incorporated memory cells that allowed them to maintain information over long sequences, making them more effective in modeling the complex sentence structures often found in biomedical literature. The work of (Zhou, et al. 2016) demonstrated the efficacy of bidirectional LSTM networks in biomedical RE, showcasing their ability to capture both past and future context in text. The deep learning phase moved away from the traditional methods that were based on specific features. Instead, it focused on automatically learning patterns directly from the data. This shift allowed for the development of more robust and scalable RE systems, capable of handling the growing complexity and volume of biomedical literature. The advancements in deep learning, particularly the use of LSTM, RNN, and CNN architectures, set the stage for further innovations in the field, leading to more accurate and efficient extraction of biomedical relations.

With the publication of the 'Attention is All You Need' paper (Vaswani, et al. 2017) which introduced the transformer architecture, a significant milestone was marked in the evolution of RE. These models, with their ability to capture long-range dependencies and contextual nuances, set new benchmarks in the field. BERT (Bidirectional Encoder

Representations from Transformers), introduced by (Devlin, et al. 2019), was a transformative model that leveraged the transformer architecture to pre-train deep bidirectional representations from unlabeled text. This pre-training, followed by fine-tuning on specific tasks, allowed BERT to achieve state-of-the-art performance across various NLP tasks, including RE. The success of BERT paved the way for its biomedical variant, BioBERT, developed by (Lee, et al. 2020). BioBERT was pre-trained on a large corpus of biomedical texts, which enabled it to capture domain-specific language and semantics. This specialization made BioBERT highly effective in biomedical RE, outperforming previous models and setting new standards in the field. Following the success of BERT and BioBERT, other transformer-based models emerged, each contributing to the advancement of biomedical RE. For example, XLNet (Yang, et al. 2019), an extension of BERT, introduced permutation-based training to capture bidirectional context more effectively. In the biomedical domain, models like BlueBERT (Peng, Yan and Lu 2019), a variant of BERT pre-trained on both biomedical and general-domain corpora, further enhanced performance in RE tasks by leveraging a broader range of linguistic knowledge. For example, models like ClinicalBERT (Huang, Altosaar and Ranganath 2019) and BioELECTRA (Kanakarajan, Kundumani and Sankarasubbu 2021) have been tailored for clinical and biomedical texts, respectively. These models are trained on specific subsets of biomedical literature, such as clinical notes or research articles, to capture the unique language used in different subdomains of biomedicine. The development of large language models (LLMs) like GPT-3 (Brown, et al. 2020) and its biomedical adaptations further pushed the boundaries of what was possible in biomedical RE. These models, with their immense scale and generative capabilities, brought new

levels of understanding and contextual awareness to the task of extracting relations from biomedical texts. The era of transformer-based models and LLMs in biomedical RE represents a significant leap forward in the field's ability to process and interpret complex biomedical literature. These models have not only improved the accuracy and efficiency of RE but also opened new possibilities for exploring intricate relationships in biomedical data.

Building on the advancements brought by biomedical Large Language Models (LLMs) such as PMC-Llama (Wu, et al. 2023), BioGPT (Luo, et al. 2022) and BioMedLM (Bolton, et al. 2024), the field of natural language processing witnessed a groundbreaking development with the introduction of ChatGPT (GPT 3.5). This model, a variant of the Generative Pre-trained Transformer (GPT) architecture by Open AI, has gathered significant attention for its flexibility and capability to perform a wide array of tasks, including biomedical relation extraction. It has become a benchmark for comparing the performance of various models in tasks including biomedical relation extraction. Its ability to comprehend and generate contextually relevant responses has set a high standard in the field.

2.4 The E2ERE Task

Given a document, the output of E2ERE is typically a set of asserted triples (m_1, r, m_2) in it where $m_1$is the subject entity span, m_2 is the object entity span, and r is a predicate or relation type from a predetermined fixed set of relations *R*. Typically, there is no requirement to identify entity spans that do not participate in a relation. However, in

addition to the entity spans, the associated entity type is also expected to be predicted. Additionally, predetermined rules may be available with regards to the types for entities allowed to take the role of a subject or an object for each predicate in *R*. When such rules are not available, they can be inferred from training data. The key point is that the input provided to E2ERE is raw text without any pre-spotted entities.

CHAPTER 3. DATASET

3.1 RareDis dataset

As discussed in Chapter 1, building a compendium of rare disease information necessitates developing NLP models for RE that can scour through biomedical literature. To this end, Maritinez-deMiguel et al. (Martínez-deMiguel, et al. 2022) created an annotated corpus for rare disease-related information extraction. This resource is based on the database of articles about rare diseases maintained by the National Organization for Rare Disorders (https://rarediseases.org/rare-diseases/). The dataset contains six entity types and six relation types, and the annotation process is described in detail by the authors (Martínez-deMiguel, et al. 2022).

3.2 Entity and Relation types

The six entity types in RareDis are: **disease**, **rare disease**, **symptom**, **sign**, **anaphor**, and **rare skin disease** with frequencies shown in the Table 3.1. There are six relation types (with counts shown in the Table 3.2): **produces** (relation between any disease entity and a sign/symptom produced by that entity), **increase_risk_of** (relation between a disease entity and another disease entity where the subject disease increases the likelihood of suffering from the object disease), **is_a** (relation

between a given disease and its classification as a more general disease), **is_acron** (relation between an acronym and its full or expanded form), **is_synon** (relation between two different names designating the same disease) and **anaphora** (relation of an anaphor entity with its antecedent entity). Here an anaphor entity refers to pronouns or pronominal constructs (e.g., 'it" or "this disease") that point to a named entity that is already mentioned in the preceding context (the "antecedent" of the anaphora relation). An example is shown in Figure 3.1.

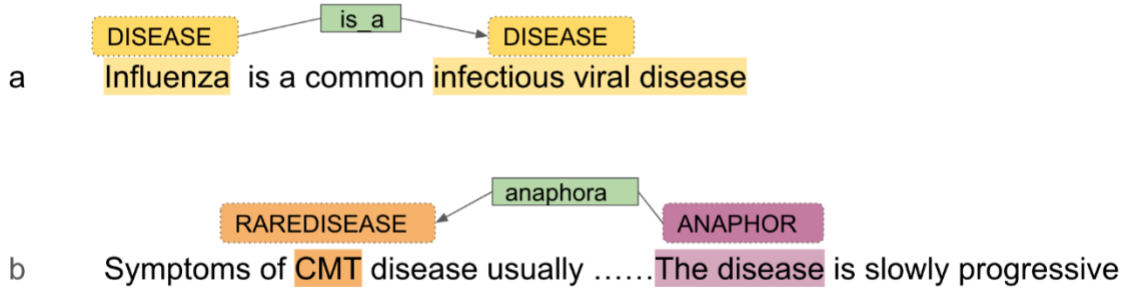

Figure 3.1 Examples of is_a and anaphora relations in the RareDis dataset.

The dataset contains discontinuous and overlapping/nested entities as discussed with examples in Section 1; Table 3.3 throws light on the relative frequency of these situations where "flat" corresponds to continuous entities. While in both tables in this section we show training, development, and test set counts, the original dataset consisted of only training and development datasets where the authors claim to withhold the test set for a future shared task, which has not happened yet. We split up their training dataset into training and development for our experiments and their development split became our test split.

Table 3.1 Statistics of entity types in the RareDis corpus

TYPE	TRAINING	DEV	TEST
SIGN	2945	798	528
RARE DISEASE	2533	624	480
DISEASE	1369	278	230
ANAPHOR	913	195	151
SKIN RARE DISEASE	393	58	45
SYMPTOM	275	44	24

Table 3.2 Statistics of relation types in the RareDis corpus.

TYPE	TRAINING	DEV	TEST
PRODUCES	3256	850	556
ANAPHORA	918	195	151
IS_A	544	149	88
INCREASE_RISK_OF	161	8	22
IS_ACRON	142	44	34
IS_SYNON	66	14	16

Table 3.3 Counts of entity types in the corpus.

ENTITY TYPE	TRAINING	DEV	TEST
FLAT	7103	1666	1212
DISCONTINUOUS	528	136	103
OVERLAPPED	720	166	112
NESTED	77	29	31

3.3 Modifications to the original dataset

While exploring the dataset, we observed some annotation issues that we confirmed with the creators of the RareDis dataset through email communication. Next, we describe what they are and how we fixed them at a high level in this section. We created a custom train, validate, test split of the full dataset after fixing the following errors and made it available as a Google Drive link on our GitHub page for this work.

3.3.1 Relation argument error

Figure 3.2 shows an example of how the annotations are provided for each instance. For this example, we see the entities (T1, . . . , T9) listed first along with types, character-based offsets, and lexical spans. Next, relations between entities are listed (R1, . . . , R5) along with the relation type and the arguments (subject and object). Although there are only nine entities, we see for anaphora relation R5, the second argument is T90 with a trailing 0 after 9. This happened several times — arguments in relations referring to entity IDs that are not present in the preceding entity list. This almost always happened with a trailing extra zero. We safely removed that zero and it fixed all these errors, which accounted for 9% of the total number of relations. In the example in Figure 2, the anaphora relation R5 was referring to the bigram “This disorder”.

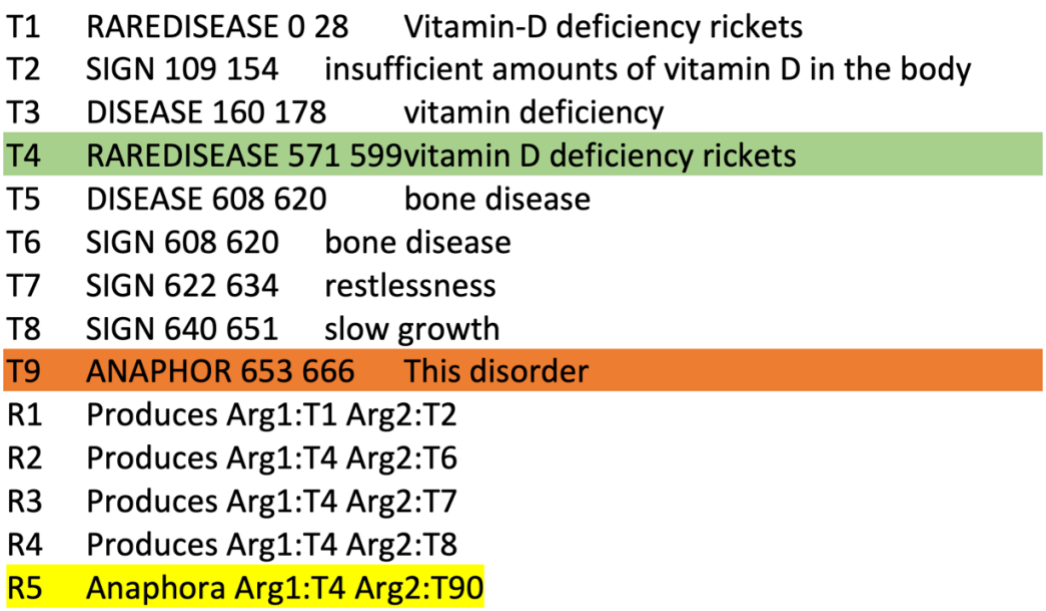

Figure 3.2 An example of the argument error due to an extra trailing zero in entity IDs.

3.3.2 Span mismatch Error

There were a few occasions (less than 1% of the full dataset) where the character offsets for entities captured an extra character than needed or missed the last character of a word. We used simple rules to remove the extra character or add the missing character. For example, in the sentence “Balantidiasis is a rare infectious disease caused by the single-celled (protozoan) parasite Balantidium coli,” the bold phrase was annotated as [T24, DISEASE,1272 1289, infectious diseas] with a missing trailing character ‘e’.

3.3.3 Offset order error.

For some discontinuous entities where more than one span is part of the full entity, the order used for the spans was not left to right and we simply reordered them as such.

CHAPTER 4. E2ERE METHODS

4.1 Pipeline: The PURE Approach

PURE by Zhong and Chen (Zhong and Chen 2021) is a span-based model that has two different models for NER and RE parts of the E2ERE system. It improved upon prior joint modeling approaches even though it separately trains NER and RE models. The main argument by Zhong and Chen, the authors of PURE, is that NER and RE need different representations of tokens because they need different types of signals to make the predictions; and combining the signals can hurt the performance of both.

One weakness of PURE is that it does not handle discontinuous entities in its NER component while it easily handles flat and nested entities. So, we needed to adapt the PURE approach to the RareDis setting. Since PURE is pipeline-based, we could simply use a different NER model for identifying discontinuous entities and retain the PURE model to spot flat and nested entities. Hence, we use a specialized model that was exclusively developed for handling discontinuous entities called SODNER (Li, et al. 2021), which is also a span-based NER model that models discontinuous NER task as a classification problem to predict whether entity fragments with gaps ought to be linked to form a new entity. To do this, SODNER uses dependency parses of the input document to guide a graph convolutional neural (GCN) network (Guo, Zhang and Lu 2019) (Zhang, Qi and Manning 2018) that obtains enhanced contextual embeddings to link disparate

fragments and form discontinuous entities. Figure 4.1 shows the schematic of the pipeline we use. It starts on the left with the SODNER model identifying discontinuous entities.

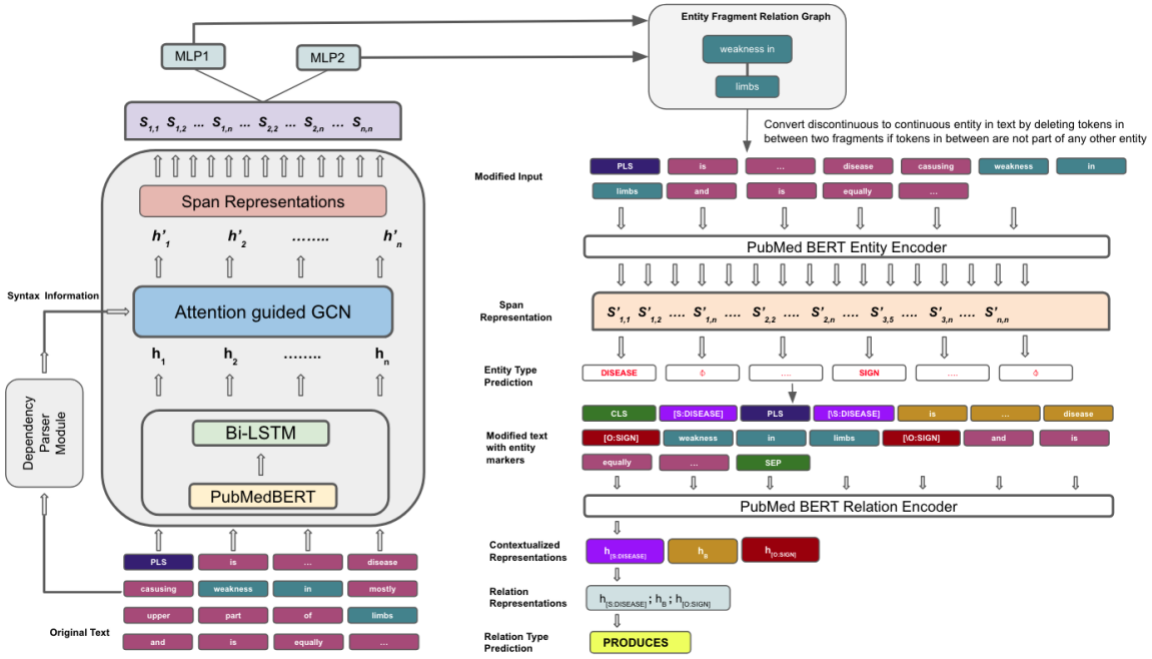

Figure 4.1 Pipeline approach using SODNER and PURE models for end-to-end relation extraction.

Even if SODNER successfully identifies discontinuous entities, PURE's relation extraction model cannot handle them. The PURE relation model puts exactly one start and one end entity marker token around each candidate subject (or object) entity span. This modified input is passed through the contextual language model (such as PubMedBERT (Gu, et al. 2021)) and the marker token embeddings are used to predict the relation type. This is reflected by the purple [S:Disease] and [\S:Disease] tokens on the right side of Figure 4.1. But SODNER outputs multiple fragments for discontinuous entities. Rather than change the PURE relation model architecture, we use the discontinuous entity fragments and straightforward rules to convert the input sentence to a modified one where the discontinuous entities are rendered in a continuous format. For instance, consider the

input, “weakness in the muscles of the arms and legs,” which contains two entities: one flat entity, “**weakness in the muscles of the arms** and legs” and one discontinuous entity, “**weakness in the muscles of the** arms and **legs**.” Both entities have the gold entity type **Sign**. Our modified new input will read as: “**weakness in the muscles of the arms** and **weakness in the muscles of the legs**”. This transformed sentence is now input through the PURE NER model and then through the PURE relation model (For more details, see appendix).

Neither the PURE NER model nor SODNER can handle cases where the same span has more than one entity type (e.g., a span being both a disease and a sign). This is a special case of overlapped entities where the overlap is exact, leading to the same span having two types. Since most relations involving such spans only use one of the entity types, this has not caused major issues in RE evaluation.

4.2 Sequence-to-Sequence: The Seq2Rel

The Seq2Rel model (Giorgi, Bader and Wang 2022) model uses an encoder-decoder framework to process the input document and output relations akin to machine translation where the source language sentence is ingested into the encoder and the target language sentence is output by the decoder one token at a time. Here the target sequence is essentially a list of relations. Unlike the machine translation setting where the target is a natural language sequence where an order is inherent, relations do not have any order among them. Hence, during training an order is imposed on the relations in a document. Special tokens are also used to represent entity types. For example, the relation R2 in

Figure 3.2 indicates: (**Rare disease** "Vitamin D Deficiency Rickets", produces, **sign** "bone disease"), where the entity types are in bold. This will be linearized in Seq2Rel as: Vitamin D Deficiency Rickets *@RareDisease@* bone disease *@Sign@ @PRODUCES@*, where **@ENTITY-TYPE@** and **@RELATION-TYPE@** are special tokens indicating entity and relation types, respectively. The **@ENTITY-TYPE@** tokens are preceded by the actual entity spans in the input. If an input does not contain any relations, a special **@NOREL@** is coded as the output. The order imposed during training is simply the order in which the entities occur in the document. This is reflected in Figure 3.2 where relations involving entities that occur earlier in the document are annotated before relations that involve entities that occur later. This left-to-right order is followed until all relations are output followed by a special end of sequence token **@END@** signaling that all relations have been output. Besides this linearization schema, a "copy mechanism" (Zeng, et al. 2018) is applied to the decoder, restricting it to generate tokens only from the observed input sequence, unlike the full vocabulary of the target language in machine translation. This mechanism enables the decoder to output spans of the input text that correspond to entities, as well as special tokens representing relation labels that connect these entities. The Seq2Rel model uses a PubMedBERT model as the encoder and a long short-term memory (LSTM) network as the decoder. The schematic workflow of seq2rel is shown in Figure 4.2.

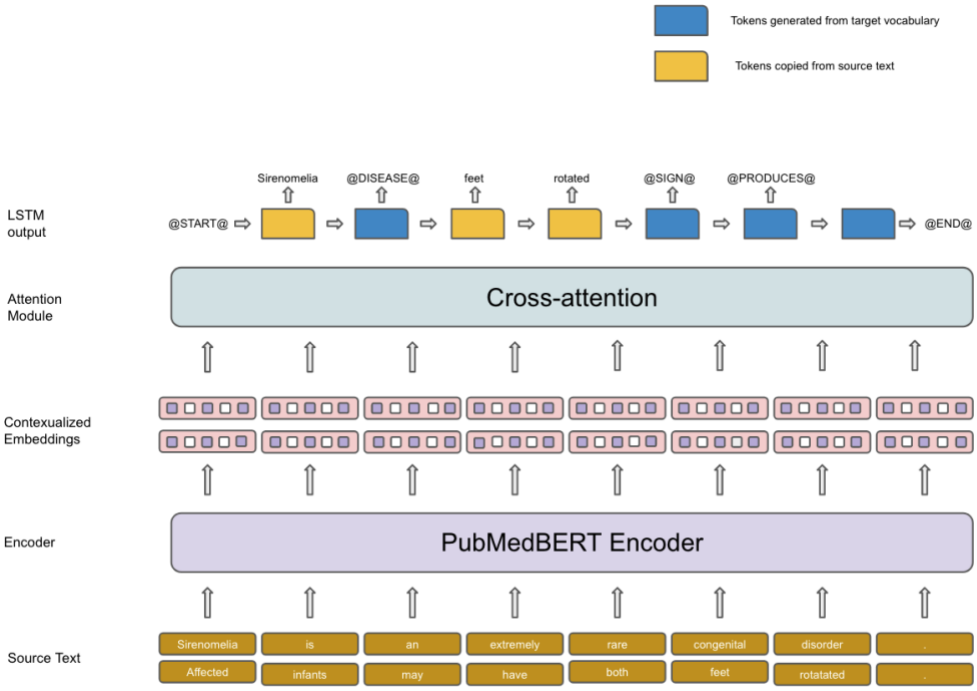

Figure 4.2 Seq2rel input/output flow for Raredis Corpus

4.3 Pretrained Language Models – T5

. T5, developed by Google Research (Raffel, et al. 2020), challenges the conventional task-specific architectures by converting every NLP problem into a text-to-text input-output format. A key aspect of T5 is its baseline pre-training objective. For this, a large free text dataset known as the "Colossal Clean Crawled Corpus" also called C4 (TensorFlow n.d.) was created and random spans of text are masked with the model tasked to predict these spans. Unlike masked language modeling in BERT models, each masked spans are replaced with only one sentinel token given a unique ID. This approach helps the model learn a broad understanding of language and context. This baseline model is further trained on a suite a suite of NLP tasks (e.g., sentiment analysis, word sense

disambiguation, and sentence similarity) in the text-to-text format. Another significant feature of T5 is its scalability, with versions ranging from small (60 million) to extremely large (11 billion), allowing it to be tailored to specific computational constraints and performance requirements.

Flan-T5 (Chung, et al. 2022) is an extension of T5 that is instruction fine-tuned on 1800 tasks. During this phase, the model is finetuned on a diverse range of tasks but with instructions provided in natural language. This training method enables Flan-T5 to understand and execute tasks based on straightforward instructions, making it more flexible and applicable to a wide range of real-world scenarios without requiring extensive task-specific data. It is fine-tuned both with and without exemplars (i.e., zero-shot and few-shot) and with and without chain-of-thought (Wei, et al. 2022), enabling generalization across a range of evaluation scenarios. Please note that unlike Seq2Rel architecture, the outputs for T5 model's variants are expected to following natural sentence structures, which are discussed in the next section as they are common to both T5 and GPT models.

4.4 Domain-Specific Pretrained Models – BioMedLM & BioGPT

Generative pre-trained transformers (GPTs) have captured the fascination of the public and researchers alike, especially since the introduction of ChatGPT in December 2022. However, the in-context learning and few-shot capabilities have already surfaced in June 2020, when Open AI released GPT-3. Building on the decoder component of the transformer architecture with the main objective of autoregressive left to right next token

prediction task, they have excelled at text generation tasks (e.g., summarization). However, there is a growing interest in assessing their capabilities for language understanding tasks including relation extraction. BioGPT and BioMedLM have been pre-trained from scratch on biomedical abstracts from PubMed and full text articles from PubMed Central (from the corresponding subset from Pile (Gao, et al. 2020)) based on the GPT-2 model (Radford, et al. 2019). In this effort, we focus on BioMedLM, a 2.7B parameter model, comprised of 32 layers, a hidden size of 2560, and 20 attention heads. BioMedLM is an order of magnitude larger than BioGPT and nearly twice as large as $\text{BioGPT}_{\text{large}}$. It has been shown to be superior to BioGPT models (including in our experiments for this paper where BioGPT underperforms by 10-15% in F-score) and to our knowledge is the largest public GPT-style model for biomedicine. Hence, we only show BioMedLM results in this manuscript for the sake of clarify and simplicity.

Unlike Seq2Rel whose sequence generation capabilities are highly constrained to terms observed in the input, BioMedLM, T5 and BioGPT are purely generative, and supervised finetuning involves using appropriate prompts and output templates. Technically, we could simply use the linearization schemas introduced for Seq2Rel. However, these generative models generate natural language statements and not unnatural-looking templates. So, our initial experiments using a Seq2Rel style output schemas have failed. So, we considered two types of schemas here:

- **rel-is template:** This output template is the same as that used by the original BioGPT paper for E2ERE: "The relation between **subject-span** and **object-span** is **relationType.noun**," where **relationType.noun** is the noun form of the

predicate. With this template, as an example, the output for the gold relation (Wilm's tumor, is_a, kidney cancer) is: "The relationship between Wilm's tumor and kidney cancer is hyponym". We can see here that we converted "is a" predicate to a noun representation "hyponym" in the template and a similar strategy was followed for all predicates.

- **natural-lang**: We came up with different natural language templates tailored to each relation type in RareDis. They are fully specified in Table 4.1, each with a representative example.

Table 4.1 Natural language templates used to encode RareDis relations as BioMedLM outputs.

RELATION TYPE	**NATURAL LANGUAGE OUTPUT TEMPLATE (AN EXAMPLE FOR THE TEMPLATE)**
PRODUCES	ent$_1$Span is a ent$_1$Type that produces ent$_2$Span, as a ent$_2$Type (Asherman's syndrome is a rare disease that produces abdominal pain, as a symptom)
ANAPHORA	The term ent$_2$Span is an anaphor that refers back to the entity of the ent$_1$Type ent$_1$Span. (The term "it" is an anaphor that refers back to the entity of the disease encephalitis)
IS_SYNON	The ent$_1$Type ent$_1$Span and the ent$_2$Type ent$_2$Span are synonyms. (The disease diastrophic dysplasia and the rare disease disastrophic dwarfism are synonyms)
IS_ACRON	The acronym ent$_1$Span stands for ent$_2$Span, a ent$_2$Type. (The acronym LQTS stands for long QT syndrome, a rare disease)
INCREASES_RISK_OF	The presence of the ent$_1$Type ent$_1$Span increases the risk of developing the ent$_2$Type of ent$_2$Span. (The presence of the disease neutropenia increases the risk of developing the disease infections)
IS_A	The ent$_1$Type ent$_1$Span is a type of ent$_2$Span, a ent$_2$Type. (The rare skin disease Bowen disease is a type of skin disorder, a disease)

CHAPTER 5. TRAINING OBJECTIVES AND EVALUATION METRICS

5.1 Evaluation Metrics

In this work, we evaluate models using famous F1 score.

- **True Positives (TP):** True positives are instances in which a model correctly predicts the positive class. In other words, they are the cases where the model identifies an instance as positive, and it is indeed positive.

- **False Positives (FP):** False positives are instances in which a model incorrectly predicts the positive class. In other words, they are the cases where the model identifies an instance as positive, but it is negative.

- **False Negative (FN):** False negatives are instances in which a model incorrectly predicts the negative class. In other words, they are the cases where the model fails to identify an instance as positive, even though it is positive.

With the concepts of true positives (TP), false positives (FP), and false negatives (FN) established, we can now introduce two important metrics for calculating F1 score for evaluating the performance of our models: **precision** and **recall**.

- **Precision**: It measures the accuracy of the positive predictions made by the model. It is the ratio of true positives to the total number of instances predicted as positive (the sum of true positives and false positives). Precision can be expressed mathematically as:

$$Precision = \frac{TP}{TP + FP}$$

- **Recall**: Recall, also known as sensitivity, measures the model's ability to correctly identify all actual positive instances. It is the ratio of true positives to the total number of actual positive instances (the sum of true positives and false negatives). Recall can be expressed as:

$$Recall = \frac{TP}{TP + FN}$$

The F1 score is a specific case of the more general Fβ score, where β is a parameter that determines the relative importance of precision and recall. The F1 score is the harmonic mean of precision and recall when β is set to 1, giving equal weight to both metrics. The formula for the Fβ score is:

$$F_\beta = (1 + \beta^2) \times \frac{Precision \times Recall}{(\beta^2 \times Precision) + Recall}$$

When β = 1, this formula simplifies to the F1 score:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

5.2 Training Objective

5.2.1 Cross Entropy

For the pipeline model, the training objective is the well-known **cross entropy** function for both NER and RE components. Here we use multi-class cross entropy loss function as our goal is to predict the right class among several classes. The general formula for cross entropy loss function is as below:

$$L(\hat{y}, y) = -\sum_{c=1}^{C} y_c \log(\hat{y}_c)$$

Here $\boldsymbol{y_c}$ is the **true label** for class c, and $\boldsymbol{\hat{y}_c}$ is the **predicted probability** for class c. The logarithmic function penalizes the model more when it is confident and wrong, assigning a high probability to the incorrect class.

5.2.2 Auto Regressive Language model loss

The term "auto" implies self, and "regressive" indicates using past data. Therefore, an autoregressive model progresses by using its previous predictions as part of the input for the next prediction. For example, given an incomplete sentence with words $x_1, x_2 \ldots x_{n-1}$ an autoregressive model aims to predict the next word x_n.The model looks at the sequence $x_1, x_2 \ldots x_{n-1}$ and predicts the probability of each possible word in the

vocabulary that could come next. If the true next word is x_n the model ideally should assign a high probability to x_n.

The loss function for this prediction, the autoregressive loss, specifically penalizes the model if it assigns a low probability to the true next word. Mathematically, for each step t in our sequence, we calculate the negative log probability of the true word x_t , given the words that came before it:

$$L = -\sum_{t=1}^{T} \log P(x_t \mid x_1, x_2 \ldots x_{t-1})$$

In this formula, $P(x_t \mid x_1, x_2 \ldots x_{t-1})$ represents the probability the model assigns to the actual word x_t at position t, conditioned on the true preceding sequence of words. By summing up these negative log probabilities (Cross entropy per target) across the sequence and minimizing this sum during training, we are effectively guiding the model to give the highest probability to the actual next word in the sequence. The more details in Chapter 9.7 of Jurafsky and Martin (Jurafsky and Martin 2023).

Also, for evaluation, we note that RareDis annotations are at the span level and hence the same exact relation connecting the same entities can occur multiple times if it is discussed several times in the document. However, generative models do not keep track of the number of times a relation occurs as they are generative and do not operate on spans; but the pipeline models output all connections as they operate at the span level. To ensure fair evaluation, if the same relation occurs multiple times within an instance, it is collapsed into a single occurrence. This is natural and harmless because there is no loss of information if duplicate relations are ignored. Since Seq2Rel, BioMedLM and T5 produce

sequences, we use regular expressions on top of the output templates and schemas to produce the triples we need. For a relation to be counted as correctly predicted, the subject and object entity types, their spans, and the relation type all need to exactly match the ground truth relation.

CHAPTER 6. TRAINING

6.1 Experiment Settings and hyperparameters.

Experiments for the pipeline approach were performed on University of Kentucky's in-house LCC cluster of 32GB GPU. All experiments for Seq2Rel were performed on Google Colab Pro+ using an Nvidia A100-sxm4-40gb GPU with access to high RAM. In Seq2Rel, we use AllenNLP, an open-source NLP library developed by the Allen Institute for Artificial Intelligence (AI2). Fairseq, a sequence modeling toolkit, is used for training custom models for text generation tasks for BioGPT on Google Colab Pro. We used Lambda Labs to fine-tune BioMedLM and T5 on a single H100 80GB GPU. Next, we describe model configurations and hyperparameters. Our settings for learning rate, number of epochs, and other hyperparameters are determined based on experiments on the validation dataset.

- **Pipeline (SODNER+PURE):** We used a batch size of 8, a learning rate of 1e-3, and 100 epochs to train the SODNER model for discontinuous entities with a $\text{PubMedBERT}_{\text{base}}$ encoder. For the PURE NER model, we used $\text{PubMedBERT}_{\text{base}}$ and trained for 100 epochs, with a learning rate of 1e-4 and a batch size of 8. We also experimented with $\text{PubMedBERT}_{\text{large}}$ with the same settings. For the PURE relation model, we used both $\text{PubMedBERT}_{\text{base}}$ and $\text{PubMedBERT}_{\text{large}}$ as encoders

with a learning rate of 1e-5 and trained for 25 epochs with the training batch size of 8.

- **Seq2Rel**: Training was conducted for 150 epochs, with a learning rate of 2e-5 for the encoder (PubMedBERTbase or PubMedBERTlarge) and 1.21e-4 for the decoder (LSTM) with a batch size of 2 and a beam size of 3 (for the decoder).

Since we require an exact match for a prediction to be correct, we appended explicit natural language instructions to the input for the generative models, directing models to generate tokens from the input text: "**From the given abstract, find all the entities and relations among them. Do not generate any token outside the abstract."**

- **BioMedLM:** Despite supervised fine-tuning, it is not uncommon for GPT models to output strings that were not part of the input. We observed that nearly 3%-7% of entities output by BioMedLM did not exactly match ground truth spans. We used a batch size of 1 with gradient_accumulation_steps of 16, a learning rate of 1e-5, and 30 epochs for BioMedLM.

- **T5:** Using the same output templates used for BioMedLM, we trained T5 3B, Flan-T5-Large (770M), and Flan- T5-XL (3B). For T5-3B, we used a batch size of 1 with gradient_accumulation_steps set to 16, lr = 3e-4, 100 epochs, and generation beam size of 4. For Flan-T5, we used a batch size of 2 with gradient_accumulation_steps set to 16, and the rest of the hyperparameters same as T5-3B. For Flan-T5-XL, we used a batch size of 1 with

gradient_accumulation_steps set to 16, lr = 3e-4, 100 epochs, and generation beam size of 4 with DeepSpeed for CPU offloading of the parameters.

6.2 Post-Processing

We needed some post-processing tricks to handle the idiosyncrasies of the three different models. As we discussed earlier in Section 4.1, for the pipeline models, since discontinuous entities are not handled natively by the PURE relation model, we had to transform the inputs to render the discontinuous entities in a flat fashion before passing them on to the PURE model (More details can be found in appendix). For the Seq2Rel model, due to the WordPiece tokenization in BERT models, the output sometimes contains extra spaces around hyphens and brackets. To align such output strings with the input text, as a post-processing step, we removed these additional spaces, specifically around hyphens, curved brackets, and forward slashes. For the rel-is template, T5 and its variant were predicting synonym relation with the string “synonyms”; so, as a part of the post-processing, we replaced with “synonym.”

CHAPTER 7. RESULTS AND ERROR ANALYSIS

The main results of the comparison using different models are presented in Table 7.1. For BioMedLM and T5 models, the ‘copyInstruct’ column in the table indicates the additional input prompt discussed earlier in this section where models are directed to only generate tokens observed in the input. We observe that the SODNER+PURE pipeline (with PubMedBERTbase encoder) produces the best F1-score of 52.2, which is 5 points more than the best-performing Seq2Rel model with the PubMedBERTlarge encoder (47.15 F1), 5.2 points better than the best-performing model from T5 family (Flan-T5-large), and 13 points more than best performing BioMedLM model (38.9 F1). The pipeline’s performance does not increase when using the PubMedBERTlarge model. For Seq2Rel, using PubMedBERTlarge outperforms a model with PubMedBERTbase (44.53 F1) by 2.5 points, with an increase in both precision and recall. Potentially, the increased model capacity of PubMedBERTlarge enables it to capture more complex and subtle relationships between medical terms and concepts. However, it is not clear why similar gains were not observed with PubMedBERTlarge in the pipeline.

Table 7.1 Performances of different models under different settings on the RareDis dataset.

METHOD	CONFIGURATION	COPY-INSTRUCT	SCORE		
			P	**R**	**F1**
SODNER+PURE	$PubMedBERT_{base}$	NA	55.99	**48.89**	**52.20**
	$PubMedBERT_{large}$	NA	**56.20**	48.52	52.08
SEQ2REL	$PubMedBERT_{base}$	NA	47.60	40.90	44.53
	$PubMedBERT_{large}$	NA	51.46	43.51	47.15
FLAN-T5-LARGE	rel-is	yes	46.52	46.58	46.55
	rel-is	no	48.63	45.54	47.04
	natural-lang	yes	43.83	42.82	43.32
	natural-lang	no	40.07	40.17	40.12
T5-3B	rel-is	yes	41.13	39.36	40.22
	rel-is	no	45.72	41.50	43.51
	natural-lang	yes	44.25	40.71	42.40
	natural-lang	no	37.80	41.21	39.43
FLAN-T5-XL	rel-is	yes	45.00	40.82	42.82
	rel-is	no	44.16	38.10	40.91
	natural-lang	yes	44.68	42.87	43.76
	natural-lang	no	42.05	40.87	41.45
BIOMEDLM	rel-is	yes	40.19	29.68	34.14
	rel-is	no	42.14	36.1	38.89
	natural-lang	yes	38.64	32.81	35.49
	natural-lang	no	44.22	33.76	38.29

The best performance for BioMedLM is an F1 score of 38.89 using the rel-is template for prompting the model when copy instructions were not provided. When copy instructions are not provided, rel-is does slightly better (<1% F1) and when copy instructions are not provided, natural-lang does better job (1.35 of points gain) So looks like there is no advantage to using copy instructions. (However, when using the smaller BioGPT models, the natural language prompting seemed to perform slightly better than the rel-is template.) Note that, BioMedLM's best performance is still ≈ 6 points lower than then Seq2Rel's best score and 11 points lower than the pipeline score. Note that BioMedLM is over eight times larger than our best-performing pipeline model (considering it has three encoders based on the encoder PubMedBERTbase, which has 110M parameters). However, its low performance compared to the pipeline is not surprising because GPT models are autoregressive and do not benefit from language understanding arising from the bidirectional masked language modeling objective used in BERT models. Although the original BioMedLM [27] effort did not perform RE, it reports SOTA scores on biomedical Q&A tasks. The smaller BioGPT models were shown to do better than BERT models for E2ERE too. Hence, we repurposed them for this RE task and as the largest publicly available GPT-based model, BioMedLM outperformed BioGPT models [26] by 10–15% in F1 score and we do not see these as worthy of reporting in this manuscript.

The best-performing model from the T5 family is Flan-T5-large with an F1 score of 47 using the rel-is template for prompting the model when copy instructions were not provided, which is the same configuration that worked best for BioMedLM. It is surprising

to see that even though Flan-T5-Large (780M) is much smaller than T5-3B and Flan-T5-XL (3B), it outperforms the other two in every setting, except Flan-T5-XL with the natural-lang template. On comparing the same size T5 models (T5-3B and Flan-T5-XL), Flan-T5-XL performs better in most settings. We believe much larger models (GPT-3, GPT-3.5, GPT-4) ought to be used to fully leverage the power of generative LMs. Furthermore, some recent results also show that using GPT-style models to generate additional training examples to augment the training data may be a more effective way of using them, rather than fine-tuning them for RE tasks.

We also wanted to examine scores per relation type in our models to see if there are any predicates for which we are underperforming more than expected. From Table 7.2, we notice that recall is less than 5% for increases_risk_of relation type. This is quite awful but not surprising given the prevalence of such relations is very small in the dataset (from Table 1). But what is very unusual is the F1 of the 'produces' relation being less than 50, when it constitutes over 60% of all relations in the dataset (from Table 3.2). Upon deeper investigation, we found that generally longer object entities lead to NER errors. We checked this more concretely by examining the errors (for 'produces') and found out that we missed 43% of the object spans for the best-performing pipeline method. Thus, a large portion of performance loss is simply due to the model not being able to predict the object entity span correctly; especially for long object entities, even missing a single token can lead to RE errors.

Table 7.2 F1 Scores for each relation type of best-performing models in the group.

RELATION	SODNER + PURE	SEQ2REL	BIOMEDLM	FLAN-T5-LARGE
ANAPHORA	**70.11**	61.08	57.38	63.24
IS_A	**58.75**	55.00	48.40	58.38
IS_ACRON	**63.33**	45.65	51.61	53.33
PRODUCES	**47.51**	44.00	34.87	43.24
IS_SYNON	**30.00**	22.23	0.00	0.00
INCREASES_RISK_OF	8.33	**10.52**	0.00	0.00

Thus, the overall performance pattern observed for the RareDis dataset is **Pipeline > Seq2Rel > Flan-T5-Large > Flan-T5-XL > T5-3B > BioMedLM**. We wanted to verify this with at least one other dataset. Considering our prior experiences with the chemical-protein interaction extraction task (Ai and Kavuluru 2023), we repeated our E2ERE experiments using the BioCreative Shared Task VI dataset and the results showed the same performance pattern with pipeline leading to a 69 F1 score, followed by Seq2Rel with 49, and BioMedLM with 37 points.

While finetuning BioMedLM and T5 models with the same output scheme as the Seq2Rel approach did not work in our experiments, this could be due to our dataset being small. This may not be an issue for ample training data. Adding a BiLSTM on top of an encoder-only model is a common trick being pursued these days, which is the case for the SODNER model in this project. However, using a bigger encoder-only model might help

us do away with the BiLSTM and help improve the efficiency of the overall architecture. Additional experiments are needed to verify this hypothesis. The next-token loss for the BioMedLM model may potentially be dominated by the constant non-varying words in the output templates selected. This can be potentially avoided by formulating a different loss that exclusively deals with the subject and object entity spans output by the model. Furthermore, directly operating on the hidden representation of the output tokens, before decoding them into words through the language modeling head of the decoder-only architecture, could lead to more interesting and direct ways to extract entity pairs participating in a relation. These aspects need further investigation.

Next, we focus on the detailed error analysis. Before we proceed, we note that many RE errors appear to arise from NER errors. This can lead to a snowball effect of errors in the RE phase. Consider a single entity participating in n gold relations. If it is predicted incorrectly as a partial match, it may potentially lead to 2n relation errors because it can give rise to n false positives (FPs) (because the relation is predicted with the wrong span) and n false negatives (FNs) (because the gold relation with the right span is missed). Thus, even a small proportion of NER errors can lead to a high loss in RE performance. In this section, we discuss a few error categories that we observed commonly across models.

- **Partial matches**: When multi-word entities are involved, the relation error is often due to the model predicting a partial match (a substring or superstring of a gold span) and this was frequent in our effort. Consider the snippet “Kienbock disease changes may produce pain...The range of motion may become restricted”. Here Kienbock disease is the subject of a produce’s relation with the gold object span:

"the range of motion may become restricted". However, the Seq2Rel model predicted "range of motion restricted" as the object span, leading to both an FP and FN. But common sense tells us that the model prediction is also correct (and potentially even better) because it removed the unnecessary "may become" substring. In a different example, when the relation involved the gold span "neurological disorder," the model predicted a superstring "progressive neurological disorder" from the full context: "Subacute sclerosing panencephalitis (SSPE) is a progressive neurological disorder."

- **Entity type mismatch**: Because our evaluation is strict, predicting the entity spans and relation type correctly, but missing a single entity type can invalidate the whole relation leading to both an FP and an FN. The models are often confused between closely related entity types. Rare disease and skin rare disease were often confused along with the pair sign and symptom.

- **Issues with discontinuous entities**: Discontinuous entities are particularly tricky and have led to several errors, even if the prediction is not incorrect, because the model was unable to split an entity conjunction into constituent entities. Consider the snippet: "affected infants may exhibit abnormally long, thin fingers and toes and/or deformed (dysplastic) or absent nails at birth." Instead of generating relations with the two gold entities "abnormally long, thin fingers" and "abnormally long, thin toes", the model simply created one relation with "long, thin fingers and toes."

- **BioMedLM generations not in the input**: In several cases we noticed spans that were not in the input but were nevertheless closely linked with the gold entity span's meaning. For example, for the gold span "muscle twitching", BioMedLM predicted "muscle weakness". It also tried to form meaningful noun phrases that capture the meaning of longer gold spans. For instance, for the gold span "ability to speak impaired", it predicted "difficulty in speaking". For the gold span, "progressive weakness of the muscles of the legs" it outputs "paralysis of the legs". All these lead to both FPs and FNs, unfortunately.
- **Errors due to potential annotation issues**: In document-level RE settings, it is not uncommon for annotators to miss certain relations. But when these are predicted by a model, they would be considered FPs. Consider the context: "The symptoms of infectious arthritis depend upon which agent has caused the infection, but symptoms often include fever, chills, general weakness, and headaches." Our model predicted that "infectious arthritis" produces "fever". However, the gold predictions for this did not have this and instead had the relation "the infection" (anaphor) produces "fever". While the gold relation is correct, we believe what our model extracted is more meaningful. However, since we missed the anaphor-involved relation, it led to an FN and an FP.

CHAPTER 8. CONCLUSION AND FUTURE WORK

In this work, we explored four state of the art representative models for E2ERE from three competing paradigms: pipelines (SODNER + PURE), sequence-to-sequence models (Seq2Rel, T5), and generative LMs (BioMedLM). Our evaluations used a complex dataset (RareDis) involving discontinuous, nested, and overlapping entities. Even with the advances in Seq2Seq models and generative transformers, a custom-built pipeline still seems to be the best option based on our experiments in this work. The performance gap between Seq2Rel and the pipeline is not as high as that between BioMedLM and pipeline. As such there could be other datasets where Seq2Rel matches the pipeline methods especially for simpler NER scenarios without discontinuous entities. We still would not want readers to conclude that more advanced models are not suitable for this task and not to take away from the few-shot abilities of GPT models. Also, the generative aspects of GPT models may not be suitable for the type of strict evaluation imposed here where an exact match with gold spans is required. In the future, this may be mitigated by using vector similarity or edit-distance metrics to map such phrases to the closest matches of the input. Using inference-only proprietary large models such as GPT-4 (Bubeck, et al. 2023) to generate paraphrases for training instances to create larger augmented training datasets could also be helpful. However, in the end, a small ≈ 200M parameter pipeline model that can run on consumer desktops may be preferable for several use-cases even in the current era of excitement over generative transformers.

The observations pointed in the previous section lead to several future directions for our team:

- Using larger generative LMs such as GPT-3 (with 10s or 100s of billion parameters) or similarly sized open source GPT-like models like Llama (H. a.-A. Touvron 2023), Llama2 (Touvron, et al. 2023), OPT (Zhang, et al. 2022), Gemini (Team, et al. 2023), Falcon (Almazrouei, et al. 2023), PaLM (Chowdhery, et al. 2023), for E2ERE.
- Comparing performances of the models from this effort on multiple datasets, especially on those in BioNLP benchmarks such as BLURB and BigBio.
- Mitigating issues with GPT models generating entity spans not in the input by using vector similarity/edit-distance metrics to map such phrases to closest matches of the input.
- Comparing different language models with fine-tuned, zero shot, and few-shot experiments.
- Using inference-only proprietary large models such as GPT-4 to generate paraphrases for training instances to create larger augmented training datasets.

APPENDIX

Here we discuss the approaches for the conversion of Discontinuous entities to continuous entities.

8.1 Statistics of Discontinuous Entity

We remove discontinuous entities with more than 2 fragments. We define overlapping discontinuous entities and non-overlapping discontinuous entities in the document as below:

- **Non-overlapping discontinuous entities**: Discontinuous entities where no other entity is overlapping with either of the two fragments of discontinuous entities and no other entity is present in between the two fragments.
- **Overlapping discontinuous entities**: Discontinuous entities where at least one of the other entities in the document is overlapping with at least one of two fragments or is present in between the two fragments.

There are a total of 173 non-overlapping and 310 overlapping discontinuous entities (2 fragments) for the training dataset. The frequency is shown in Table 8.1.

Table 8.1 Statistics of Discontinuous Entity types with different fragments length

DISC. FRAGMENTS	TRAINING	DEV
2 FRAGMENTS	483	116
3 FRAGMENTS	40	16
4 FRAGMENTS	4	3
5 FRAGMENTS	1	0
SUM	528	135

8.2 Patterns of overlapping discontinuous entities

- **Pattern 1**: The second fragment of the discontinuous entity is overlapping with another continuous entity.

- **Pattern 2**: The second fragment of the discontinuous entity is overlapping with the second fragment of another discontinuous entity and the first fragment does not overlap.

- **Pattern 3**: The first fragment of the discontinuous entity is overlapping with another continuous entity.

- **Pattern 4**: The first fragment of the discontinuous entity is overlapping with the first fragment of another discontinuous entity and the second fragment does not overlap.

- **Pattern 5**: The first fragment of the discontinuous entity is overlapping with the first fragment of another discontinuous entity and the second fragment of the

discontinuous entity is overlapping with the second fragment of another discontinuous entity.

Also, we have

$$frag1\_cur[1] > frag1\_another[1]$$

$$frag1\_cur[0] > frag1\_another[0]$$

where,

frag1_cur[1] is the index of the right end of the first fragment of the current discontinuous entity, frag2_cur[0] is the index of the left end of the second fragment of the current discontinuous entity, frag1_another[1] is the index of the right end of the first fragment of another discontinuous entity: frag1_another[1] and frag2_another[0] is the index of the left end of the second fragment of another discontinuous entity.

- **Patten 6**: All other patterns.

8.3 Rules for converting discontinuous entities into continuous entities.

- **Rule 0**: Remove all tokens between two fragments of the discontinuous entity.
- **Rule 1**: Copy the second fragment and put it right after the first fragment.
- **Rule 2**: Copy the first fragment and put it just before the second fragment.

Apply Rule 0 for non-overlapping discontinuous entities. Apply Rule 1 for overlapping discontinuous entities with patterns 1, 2, and 5. Apply Rule 2 for overlapping discontinuous entities with patterns 3, 4, and 6.

8.4 Examples of Application of rules

- Apply **Rule 0** for non-overlapping discontinuous entities. See Figure 8.1.

Before Rule: " **accumulation of** fats (lipids) called **GM 2 gangliosides** in the neurons and other tissues ..."

After Rule: "...**accumulation of GM 2 gangliosides** in the neurons and other tissues ..."'

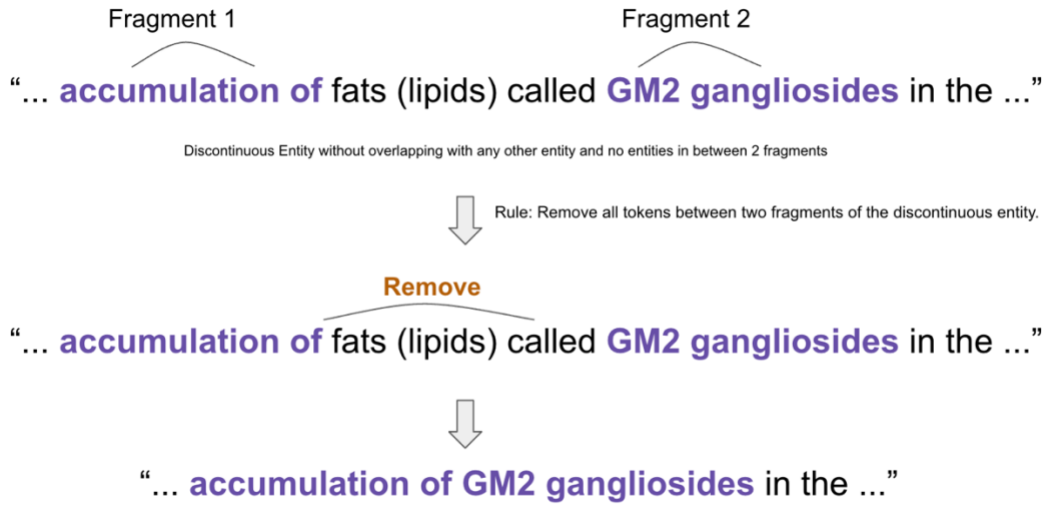

Figure 8.1 Example to convert discontinuous to continuous applying rule 0.

- Apply **Rule 1** for overlapping discontinuous entities with Pattern 1. See Figure 8.2.

Before rule: "...**crackles** or rales **in the infected lung** ..." ← **current entity**

"...crackles or **rales in the infected lung** ..." ← **another entity**

After rule: "...**crackles in the infected lung** or **rales in the infected lung** ..."

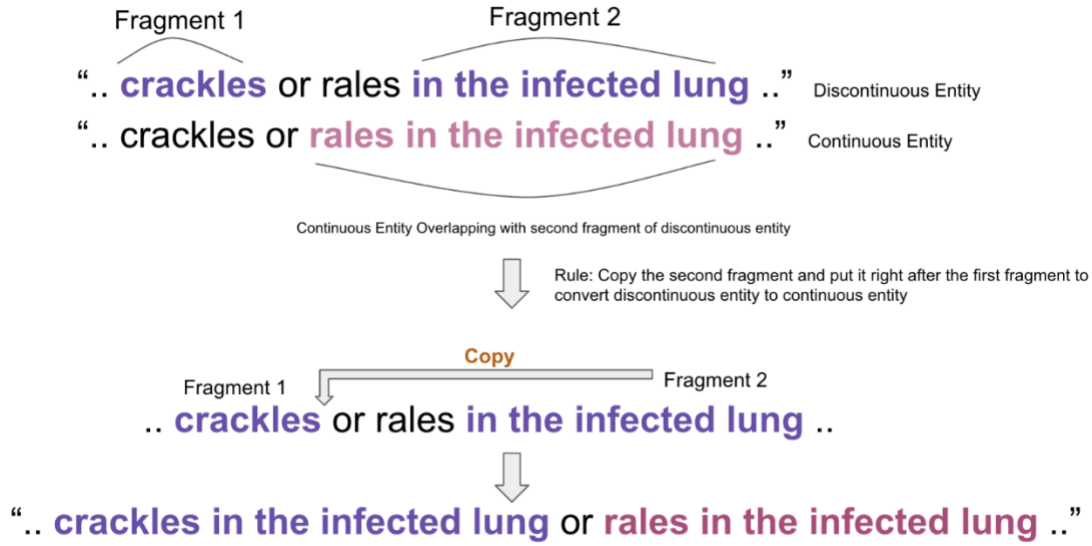

Figure 8.2 Example to convert discontinuous to continuous applying rule 1.

- Apply **Rule 2** for overlapping discontinuous entities with Pattern 3: See Figure 8.3.

Before rule: "...**weakness in the muscles of the** arms and **legs** ..." ← **current entity**

"...**weakness in the muscles of the arms** and legs ..." ← **another entity**

After rule: "...**weakness in the muscles of the arms** and **weakness in the muscles of the legs** ..."

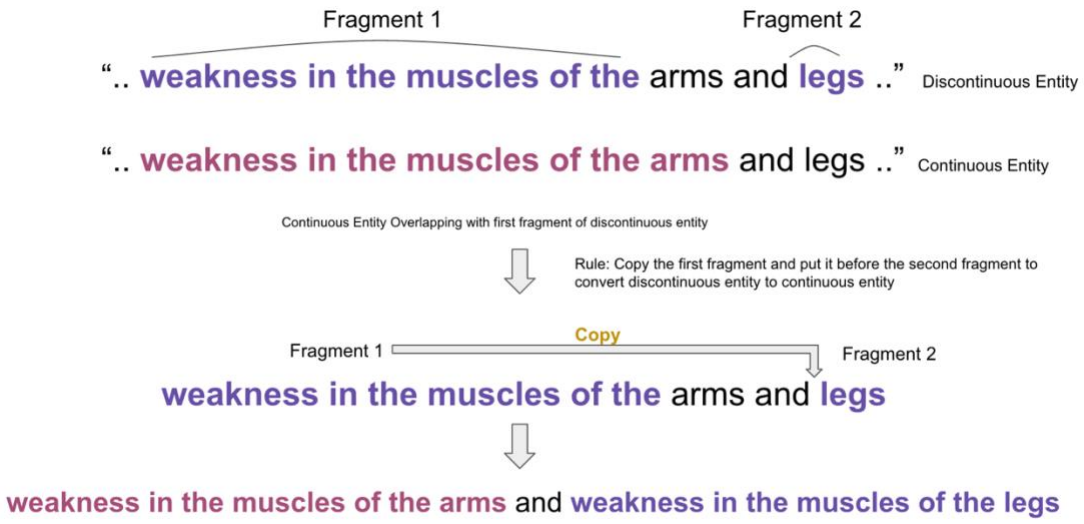

Figure 8.3 Example to convert discontinuous to continuous applying rule 2.

8.5 Mentions with double entity types.

There are some mentions with two labels. For example, the same mention can be labeled as both **Sign** and **Disease** entity types. In the training dataset, there are 355 mentions which have two labels.

Let's say some mention has two labels: entity type 1, and entity type 2. Entity type 1 contributes to N_1 relations and entity type 2 contributes to N_2 relations. There are 4 scenarios for (N_1, N_2) as below:

- **(0, 0).** This means none of the labels is contributing to any of the relations. There are 9 such mentions. We randomly remove an entity type for each mention.

- **(0, 1) or (1, 0).** This means only 1 label is contributing to the relation. There are 309 mentions. We remove the entity type which does not contribute to any relation.

- **(1, 1).** Both labels are contributing to one relation. There are 21 mentions. We randomly remove an entity type and corresponding relation for each mention.

- **(1, N) or (N, 1) where N is larger than 1**. There are 16 mentions. We remove the entity type which contributes to only one relation and the corresponding relation.

In the development dataset, there are 83 mentions which have double entity types.

- (0, 0). There are 2 mentions.
- (0, 1) or (1, 0). There are 72 mentions.

- (1, 1). There are 7 mentions.
- (1, N) or (N, 1) where N is larger than 1. There are 2 mentions.

BIBLIOGRAPHY

(NORD, National Organization for Rare Disorders. 2019. *Rare Disease Database Frequently Asked Questions.* Accessed 10 14, 2023. https://rarediseases.org/wp-content/uploads/2019/01/RDD-FAQ-2019.pdf.

Agichtein, Eugene, and Luis Gravano. 2000. "Snowball: Extracting relations from large plain-text collections." *Proceedings of the fifth ACM conference on Digital libraries.* 85–94.

Ai, Xuguang, and Ramakanth Kavuluru. 2023. "End-to-End Models for Chemical–Protein Interaction Extraction: Better Tokenization and Span-Based Pipeline Strategies." *2023 IEEE 11th International Conference on Healthcare Informatics (ICHI).* 610–618.

Almazrouei, Ebtesam, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, et al. 2023. "The falcon series of open language models." *arXiv preprint arXiv:2311.16867.*

Amin, Saadullah, Pasquale Minervini, David Chang, Pontus Stenetorp, and Günter Neumann. 2022. "Meddistant19: towards an accurate benchmark for broad-coverage biomedical relation extraction." *arXiv preprint arXiv:2204.04779.*

Appelt, Douglas E., Jerry R. Hobbs, John Bear, David Israel, and Mabry Tyson. 1993. "FASTUS: A finite-state processor for information extraction from real-world text." *IJCAI.* 1172–1178.

Baby, Cyril and Khan, Faizan and J.N., Swathi. 2017. "Home automation using IoT and a chatbot using natural language processing." 1-6.

Bolton, Elliot, Abhinav Venigalla, Michihiro Yasunaga, David Hall, Betty Xiong, Tony Lee, Roxana Daneshjou, et al. 2024. "BioMedLM: A 2.7 B Parameter Language Model Trained On Biomedical Text." *arXiv preprint arXiv:2403.18421.*

Brin, Sergey. 1998. "Extracting patterns and relations from the world wide web." *International workshop on the world wide web and databases.* 172–183.

Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, et al. 2020. "Language models are few-shot learners." *Advances in neural information processing systems* 33: 1877–1901.

Bubeck, Sébastien, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, et al. 2023. "Sparks of artificial general intelligence: Early experiments with gpt-4." *arXiv preprint arXiv:2303.12712.*

Chowdhery, Aakanksha, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, et al. 2023. "Palm: Scaling language modeling with pathways." *Journal of Machine Learning Research* 24: 1–113.

Chung, Hyung Won, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, et al. 2022. "Scaling instruction-finetuned language models." *arXiv preprint arXiv:2210.11416.*

Ciravegna, Dr Fabio. 2001. "Adaptive information extraction from text by rule induction and generalisation."

Cohen, Aaron and Hersh, William. 2005. "A Survey of Current Work in Biomedical Text Mining." *Briefings in bioinformatics* 6: 57-71.

Cohen, Kevin and Hunter, Lawrence. 2008. "Getting Started in Text Mining." *PLoS computational biology* 4: e20.

Craven, Mark, and Johan Kumlien. 1999. "Constructing Biological Knowledge Bases by Extracting Information from Text Sources." *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology.* AAAI Press. 77–86.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." Edited by Jill Burstein, Christy Doran and Thamar Solorio. *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).* Minneapolis, Minnesota: Association for Computational Linguistics. 4171–4186. doi:10.18653/v1/N19-1423.

Dietze, Heiko, and Michael Schroeder. 2009. "GoWeb: a semantic search engine for the life science web." *BMC bioinformatics* (Springer) 10: 1–13.

Dreisbach, Caitlin and Koleck, Theresa and Bourne, Philip and Bakken, Suzanne. 2019. "A systematic review of natural language processing and text mining of symptoms from electronic patient-authored text data." *International Journal of Medical Informatics* 125.

Forum, World Economic. 2024. *How to harness health data to improve patient outcomes.* January. Accessed 03 03, 2024. https://www.weforum.org/agenda/2024/01/how-to-harness-health-data-to-improve-patient-outcomes-wef24/.

Gao, Leo, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, et al. 2020. "The pile: An 800gb dataset of diverse text for language modeling." *arXiv preprint arXiv:2101.00027.*

Giorgi, John, Gary D. Bader, and Bo Wang. 2022. "A sequence-to-sequence approach for document-level relation extraction." *arXiv preprint arXiv:2204.01098.*

Global Genes. n.d. "Facts." *Facts.*

Gu, Yu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. "Domain-specific language model pretraining for biomedical natural language processing." *ACM Transactions on Computing for Healthcare (HEALTH)* (ACM New York, NY) 3: 1–23.

Guo, Zhijiang, Yan Zhang, and Wei Lu. 2019. "Attention guided graph convolutional networks for relation extraction." *arXiv preprint arXiv:1906.07510.*

Gupta, Shashank, Yuhang Jiang, and Abdullah Al Zubaer Imran. 2023. "Transforming Radiology Workflows: Pretraining for Automated Chest X-ray Report Generation." *Medical Imaging with Deep Learning, short paper track.* https://openreview.net/forum?id=YEMH26an2bM.

Henry, Sam, and Bridget T. McInnes. 2017. "Literature based discovery: models, methods, and trends." *Journal of biomedical informatics* (Elsevier) 74: 20–32.

Huang, Kexin, Jaan Altosaar, and Rajesh Ranganath. 2019. "Clinicalbert: Modeling clinical notes and predicting hospital readmission." *arXiv preprint arXiv:1904.05342.*

Jayram, Thathachar S., Rajasekar Krishnamurthy, Sriram Raghavan, Shivakumar Vaithyanathan, and Huaiyu Zhu. 2006. "Avatar information extraction system." *IEEE Data Eng. Bull.* 29: 40–48.

Jurafsky, Daniel, and James H. Martin. 2023. *Speech and Language Processing (3rd Edition).*

Kanakarajan, Kamal Raj, Bhuvana Kundumani, and Malaikannan Sankarasubbu. 2021. "BioELECTRA:Pretrained Biomedical text Encoder using Discriminators." 143-154. doi:10.18653/v1/2021.bionlp-1.16.

Kavuluru, Ramakanth, Anthony Rios, and Tung Tran. 2017. "Extracting drug-drug interactions with word and character-level recurrent neural networks." *2017 IEEE International Conference on Healthcare Informatics (ICHI).* 5–12.

Kilicoglu, Halil, Graciela Rosemblat, Marcelo Fiszman, and Dongwook Shin. 2020. "Broad-coverage biomedical relation extraction with SemRep." *BMC bioinformatics* (Springer) 21: 1–28.

Klimova, Blanka, Michael Storek, Martin Valis, and Kamil Kuca. 2017. "Global view on rare diseases: a mini review." *Current medicinal chemistry* (Bentham Science Publishers) 24: 3153–3158.

Kolanu, Nithin and Brown, A and Beech, Amanda and Center, Jacqueline and White, Christopher. 2020. "OR29-02 Natural Language Processing of Radiology Reports Improves Identification of Patients with Fracture." *Journal of the Endocrine Society* 4.

Krallinger, Martin, Alfonso Valencia, and Lynette Hirschman. 2008. "Linking genes to literature: Text mining, information extraction, and retrieval applications for biology." *Genome Biol.* 9 Suppl 2: S8. doi:10.1186/gb-2008-9-s2-s8.

Lee, Jinhyuk, 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* (Oxford University Press) 36: 1234–1240.

Li, Fei, ZhiChao Lin, Meishan Zhang, and Donghong Ji. 2021. "A Span-Based Model for Joint Overlapped and Discontinuous Named Entity Recognition." *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers).* 4814–4828.

Liddy, E. D. 2001. "Natural Language Processing." In *Encyclopedia of Library and Information Science*. NY: Marcel Decker, Inc.

Luo, Renqian, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. 2022. "BioGPT: generative pre-trained transformer for biomedical text generation and mining." *Briefings in bioinformatics* (Oxford University Press) 23: bbac409.

Luo, Yuan, Özlem Uzuner, and Peter Szolovits. 2017. "Bridging semantics and syntax with graph algorithms—state-of-the-art of extracting biomedical relations." *Briefings in Bioinformatics* 18: 722-722. doi:10.1093/bib/bbx048.

Martínez-deMiguel, Claudia, Isabel Segura-Bedmar, Esteban Chacón-Solano, and Sara Guerrero-Aspizua. 2022. "The RareDis corpus: a corpus annotated with rare diseases, their signs and symptoms." *Journal of Biomedical Informatics* (Elsevier) 125: 103961.

Miwa, Makoto, and Mohit Bansal. 2016. "End-to-End Relation Extraction using LSTMs on Sequences and Tree Structures." *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* 1105–1116.

1991. "MUC3 '91: Proceedings of the 3rd conference on Message understanding." USA: Association for Computational Linguistics.

1992. "MUC4 '92: Proceedings of the 4th conference on Message understanding." USA: Association for Computational Linguistics.

1993. "MUC5 '93: Proceedings of the 5th conference on Message understanding." USA: Association for Computational Linguistics.

1995. "MUC6 '95: Proceedings of the 6th conference on Message understanding." USA: Association for Computational Linguistics.

Muslea, Ion, and others. 1999. "Extraction patterns for information extraction tasks: A survey." *The AAAI-99 workshop on machine learning for information extraction.*

Nadkarni, Prakash and Ohno-Machado, Lucila and Chapman, Wendy. 2011. "Natural language processing: An introduction." *Journal of the American Medical Informatics Association : JAMIA* 18: 544-51.

National Organization for Rare Disorders (NORD). 2019. "Rare Disease Database Frequently Asked Questions." *Rare Disease Database Frequently Asked Questions.*

Pawar, Sachin, Pushpak Bhattacharyya, and Girish Palshikar. 2017. "End-to-end relation extraction using neural networks and Markov logic networks." *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers.* 818–827.

Peng, Nanyun, Hoifung Poon, Chris Quirk, Kristina Toutanova, and Wen-tau Yih. 2017. "Cross-sentence n-ary relation extraction with graph lstms." *Transactions of the Association for Computational Linguistics* (MIT Press) 5: 101–115.

Peng, Yifan, Shankai Yan, and Zhiyong Lu. 2019. "Transfer learning in biomedical natural language processing: an evaluation of BERT and ELMo on ten benchmarking datasets." *arXiv preprint arXiv:1906.05474.*

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. *Language models are unsupervised multitask learners.* https://insightcivic.s3.us-east-1.amazonaws.com/language-models.pdf.

Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. "Exploring the limits of transfer learning with a unified text-to-text transformer." *Journal of machine learning research* 21: 1–67.

Reiter, Ehud, and Robert Dale. 2000. *Building natural language generation systems.* USA: Cambridge University Press.

Riloff, Ellen, and others. 1993. "Automatically constructing a dictionary for information extraction tasks." *AAAI.* 2–1.

1998. "Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29 - May 1, 1998." https://aclanthology.org/M98-1000.

Shaalan, Khaled, and Hafsa Raza. 2008. "Arabic named entity recognition from diverse text types." *Advances in Natural Language Processing: 6th International Conference, GoTAL 2008 Gothenburg, Sweden, August 25-27, 2008 Proceedings.* 440–451.

Shen, Warren, AnHai Doan, Jeffrey F. Naughton, and Raghu Ramakrishnan. 2007. "Declarative Information Extraction Using Datalog with Embedded Extraction Predicates." *VLDB.* 1033–1044.

Swartz, Jordan and Koziatek, Christian and Theobald, Jason and Smith, Silas and Iturrate, Eduardo. 2017. "Creation of a Simple Natural Language Processing Tool to Support an Imaging Utilization Quality Dashboard." *International Journal of Medical Informatics* 101.

Team, Gemini, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, et al. 2023. "Gemini: a family of highly capable multimodal models." *arXiv preprint arXiv:2312.11805.*

TensorFlow. n.d. "C4: Colossal Cleaned Common Crawl." *C4: Colossal Cleaned Common Crawl.*

Touvron, Hugo and Lavril, Thibaut and Izacard, Gautier and Martinet, Xavier and Lachaux, Marie-Anne and Lacroix, Timothe and Rozire, Baptiste and Goyal, Naman and Hambro, Eric and Azhar, Faisal and others. 2023. "Llama: Open and efficient foundation language models." *arXiv preprint arXiv:2302.13971.*

Touvron, Hugo, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, et al. 2023. "Llama 2: Open foundation and fine-tuned chat models." *arXiv preprint arXiv:2307.09288.*

Tran, Tung, and Ramakanth Kavuluru. 2018. "An end-to-end deep learning architecture for extracting protein–protein interactions affected by genetic mutations." *Database* (Oxford University Press) 1-13.

Tseng, Bo-Hsiang, Jianpeng Cheng, Yimai Fang, and David Vandyke. 2020. "A Generative Model for Joint Natural Language Understanding and Generation." Edited by Dan Jurafsky, Joyce Chai, Natalie Schluter and Joel Tetreault. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.* Online: Association for Computational Linguistics. 1795–1807. doi:10.18653/v1/2020.acl-main.163.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. "Attention is all you need." *Advances in neural information processing systems* 30.

Wang, Rong Xu and QuanQiu. 2014. "Automatic construction of a large-scale and accurate drug-side-effect association knowledge base from biomedical literature." *Journal of Biomedical Informatics* 51: 191-199.

Wei, Jason, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V. Le, Denny Zhou, and others. 2022. "Chain-of-thought prompting elicits reasoning in large language models." *Advances in neural information processing systems* 35: 24824–24837.

World Economic Forum. 2024. "How to harness health data to improve patient outcomes." *How to harness health data to improve patient outcomes.* January.

Wu, Chaoyi, Xiaoman Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. 2023. "Pmc-llama: Further finetuning llama on medical papers." *arXiv preprint arXiv:2304.14454.*

Xu, Rong, and QuanQiu Wang. 2014. "Automatic construction of a large-scale and accurate drug-side-effect association knowledge base from biomedical literature." *Journal of Biomedical Informatics* 51: 191-199. doi:https://doi.org/10.1016/j.jbi.2014.05.013.

Yang, Zhilin, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R. Salakhutdinov, and Quoc V. Le. 2019. "Xlnet: Generalized autoregressive pretraining for language understanding." *Advances in neural information processing systems* 32.

Yao, Yuan, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. "DocRED: A Large-Scale Document-Level Relation Extraction Dataset." *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics.* 764–777.

Zeng, Daojian, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. "Relation Classification via Convolutional Deep Neural Network." Edited by Junichi Tsujii and Jan Hajic. *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers.* Dublin, Ireland: Dublin City University and Association for Computational Linguistics. 2335–2344. https://aclanthology.org/C14-1220.

Zeng, Xiangrong, Daojian Zeng, Shizhu He, Kang Liu, and Jun Zhao. 2018. "Extracting Relational Facts by an End-to-End Neural Model with Copy Mechanism." Edited by Iryna Gurevych and Yusuke Miyao. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* Melbourne, Australia: Association for Computational Linguistics. 506–514. doi:10.18653/v1/P18-1047.

Zhang, Meishan, Yue Zhang, and Guohong Fu. 2017. "End-to-End Neural Relation Extraction with Global Optimization." *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing.* 1730–1740.

Zhang, Susan, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, et al. 2022. "Opt: Open pre-trained transformer language models." *arXiv preprint arXiv:2205.01068.*

Zhang, Yuhao, Peng Qi, and Christopher D. Manning. 2018. "Graph Convolution over Pruned Dependency Trees Improves Relation Extraction." Edited by Ellen Riloff, David Chiang, Julia Hockenmaier and Jun'ichi Tsujii. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing.* Brussels, Belgium: Association for Computational Linguistics. 2205–2215. doi:10.18653/v1/D18-1244.

Zhong, Zexuan, and Danqi Chen. 2021. "A Frustratingly Easy Approach for Entity and Relation Extraction." *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.* 50–61.

Zhou, Peng, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. "Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification." Edited by Katrin Erk and Noah A. Smith. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers).* Berlin, Germany: Association for Computational Linguistics. 207–212. doi:10.18653/v1/P16-2034.

VITA

Name: Shashank Gupta

Education

Bachelor of Technology in Information technology, June 2013 - May 2017
Guru Gobind Singh Indraprastha University, India

Experience

- Senior AI Engineer (NLP), Magna International, Michigan Jan 2024 - Current
- Software Development Analyst, Accenture, India Oct 2017 - Dec 2020

Awards

- Best Project award by CS faculty for developing “An Ethereum-based Product Identification System for Anti-counterfeits”.

Publications

- Gupta, Shashank, and Bhaskar Kapoor. "Firebase In App Development." International Research Journal of Engineering and Technology 2, no. 12 (2016).

- Gupta, Shashank, Yuhang Jiang, and Abdullah Al Zubaer Imran. 2023. "Transforming Radiology Workflows: Pretraining for Automated Chest X-ray Report Generation." In *Medical Imaging with Deep Learning, Short Paper Track*. https://openreview.net/forum?id=YEMH26an2bM

- Gupta, Shashank, Xuguang Ai, and Ramakanth Kavuluru. "Comparison of pipeline, sequence-to-sequence, and GPT models for end-to-end relation extraction: experiments with the rare disease use-case." arXiv preprint arXiv:2311.13729 (2023).

- Gupta, Shashank. "An Ethereum-based Product Identification System for Anti-counterfeits." arXiv preprint arXiv:2308.04006 (2023).

- Gupta, Shashank. "Top K Relevant Passage Retrieval for Biomedical Question Answering." arXiv preprint arXiv:2308.04028 (2023).