

# **AutoRD: An Automatic and End-to-End System for Rare Disease Knowledge Graph Construction Based on Ontology-enhanced Large Language Models**

Lang Cao, Jimeng Sun, Adam Cross

Submitted to: Journal of Medical Internet Research  
on: May 22, 2024

**Disclaimer:** © The authors. All rights reserved. This is a privileged document currently under peer-review/community review. Authors have provided JMIR Publications with an exclusive license to publish this preprint on its website for review purposes only. While the final peer-reviewed paper may be licensed under a CC BY license on publication, at this stage authors and publisher expressly prohibit redistribution of this draft paper other than for review purposes.

Table of Contents

Original Manuscript..... 11

Supplementary Files..... 30

    Figures ..... 31

        Figure 1..... 32

        Figure 2..... 33

        Figure 3..... 34

        Figure 4..... 35

        Figure 5..... 36

        Figure 6..... 37

# AutoRD: An Automatic and End-to-End System for Rare Disease Knowledge Graph Construction Based on Ontology-enhanced Large Language Models

Lang Cao<sup>1</sup> MS; Jimeng Sun<sup>1</sup> PhD; Adam Cross<sup>2</sup> MD

<sup>1</sup>University of Illinois Urbana-Champaign Urbana US

<sup>2</sup>University of Illinois College of Medicine Peoria Peoria US

## Corresponding Author:

Adam Cross MD

University of Illinois College of Medicine Peoria

1 Illini Drive

Peoria

US

## Abstract

### Background: Background and significance

Several studies have employed machine learning methods to support and enhance the medical management and process of rare diseases. Sanjak et al [18] introduced an innovative method for clustering over 3,000 rare diseases using node embeddings within a knowledge graph. This approach facilitates a deeper understanding of the relationships between different diseases and opens possibilities for drug repurposing. Alsentzer et al [19] developed Shepherd, a deep learning model designed for diagnosing rare diseases. This model effectively leverages clinical and genetic patient data along with existing medical knowledge to uncover new disease-gene associations. This work exemplifies the potential of AI in medical diagnostics, even in scenarios with limited labeled data. Rashid et al. [20] explored a unique approach in rare disease research through the National Mesothelioma Virtual Bank. They utilized REDCap and a web portal query tool to integrate and manage clinical data from multiple institutions. This method demonstrates the power of combining data management tools and web technologies to enhance research and collaboration in the field of rare diseases.

The advent of LLMs has led to their increasing application in the medical field. Datta et al [21] developed AutoCriteria, an LLM-based information extraction system that has shown high accuracy and generalizability in extracting detailed eligibility criteria from clinical trial documents for various diseases. This represents a scalable solution for clinical trial applications. In the context of rare diseases, there have been specific research efforts utilizing LLMs. Shyr et al [22] explored the performance of ChatGPT in extracting rare disease phenotypes from unstructured text, using zero- and few-shot learning techniques. This study demonstrated potential in certain scenarios, particularly with tailored prompts and minimal data. Oniani et al [23] proposed Models-Vote Prompting (MVP), an approach that improves LLM performance in few-shot learning scenarios by aggregating outputs from multiple LLMs through majority voting.

However, these studies on LLM applications for rare diseases are still preliminary. Both focused on evaluating the basic capabilities of LLMs in identifying rare diseases or used simple prompt ensembles to slightly enhance LLM performance. They did not explore the task of extracting relationships between rare diseases and related phenotypes. Additionally, these studies primarily explore basic applications of LLMs and do not extend to a comprehensive LLM-based system. Building on these foundational works, our research continues to delve into the use of LLMs for rare disease applications. Unlike prior efforts, we propose an integrated and useful system aimed at extracting rare disease information from unstructured text. The elaborate methods incorporated into this system significantly enhance extraction accuracy compared to the use of pure LLMs, marking a substantial advancement in this field.

**Objective:** Rare diseases, also known as orphan diseases, are relatively uncommon in isolation and sometimes receive less individual attention in medical research due to their low prevalence. [1] The likelihood of an individual being affected by a rare disease is relatively low. However, when considering the global population, many individuals are impacted. In the United States, rare diseases affect approximately 30 million people; [2] globally, the number rises to between 300 and 400 million. [3]

Furthermore, the rare disease patient population, distributed across 5,000 to 10,000 distinct diseases, [4] suffers from a significant lack of medical knowledge due to the rarity of a given illness. Consequently, patients often face prolonged and costly diagnostic processes and intensive treatments, with many of these diseases lacking approved therapies. [5,6] This situation underscores the substantial burden placed on both patients and healthcare systems. [7]

Online resources, including open-source databases, offer valuable references for medical professionals, contributing to the development of a comprehensive rare disease knowledge system. Examples of such databases include the Unified Medical Language System (UMLS) [8] and the Orphanet Human Phenotype Ontology (HPO). [9,10] Specifically, Orphanet's database provides detailed information linking rare diseases, genes, and phenotypes, which greatly aids in the identification and diagnosis of rare diseases, among other related processes. However, these databases require considerable human effort for curation and maintenance. Therefore, there is an urgent need to develop methods that can support the process of establishing and enhancing rare disease medical knowledge systems.

Natural Language Processing (NLP) techniques are instrumental in automatically processing unstructured text to extract structured and clinically relevant information. This technique is especially beneficial for information extraction and knowledge discovery in the medical field. Recently, Large Language Models (LLMs) have demonstrated exceptional proficiency in language understanding and generation, garnering significant attention in the open NLP domain. [11,12] Their ease of use allows humans to complete a wide range of complex tasks in everyday life. Moreover, the extensive knowledge stored within their parameters equips them to excel in domain-specific applications, such as medicine and healthcare. [13]

Current research is beginning to evaluate the capabilities of the most powerful LLMs, such as ChatGPT and GPT-4, across various medical applications. These applications include licensing examinations, [14] question answering, [15] and medical education. [16] Notably, several studies have demonstrated that LLMs are effective few-shot medical Named Entity Recognition (NER) extractors, exhibiting superior few-shot learning capabilities compared to other NLP methods. [17] In the context of rare diseases, where resources are often limited, LLMs emerge as valuable tools for extracting information about these conditions, showcasing their utility in enhancing medical knowledge systems.

In this paper we introduce Automated Rare Disease Mining (AutoRD) as an efficient tool for extracting information about rare diseases and constructing corresponding knowledge graphs. The system processes unstructured medical text as input and outputs extraction results and a knowledge graph. It is comprised of several key stages: data preprocessing, entity extraction, relation extraction, entity alignment, and knowledge graph construction. Among these, entity and relation extraction are the most critical parts. AutoRD is an LLM-based system built upon GPT-4. [12] We employ prompts as instructions to guide the LLMs through the entity and relation extraction processes. The model leverages its strong zero-shot capabilities to identify and extract entities and to analyze relationships between them. Although LLMs are pre-trained with extensive knowledge, they sometimes lack precise medical information. To address this, we enhance the LLM's medical knowledge using rare disease and phenotype ontologies. This is achieved by designing sophisticated prompts that incorporate relevant knowledge. We conducted experiments to evaluate the system and identified the advantages and limitations of AutoRD. In summary, our contributions can be summarized as follows:

1. We propose AutoRD, an automated end-to-end system which efficiently extracts rare disease information from text and builds knowledge graphs. This is a useful and practical system which can help medical professionals discover information about rare diseases.
2. We use ontology-enhanced LLMs in the module of rare disease entity extraction and relation extraction. This approach harnesses the few-shot learning capabilities of LLMs and integrates medical knowledge from ontologies, resulting in an improved performance beyond what LLMs achieve alone.
3. We conduct experiments and provide extensive analysis to demonstrate the effectiveness of AutoRD.

#### **Methods: Data**

To improve the medical understanding of LLMs, we incorporated three medical ontologies into AutoRD: Orphanet Rare Disease Ontology (ORDO)[10], HPO-ORDO Ontology Module (HOOM) [10] , and Mondo Disease Ontology (Mondo) [24] .

For assessing the entity and relation extraction capabilities of AutoRD, the RareDis-v1 dataset [25] was employed. Prior to use, this dataset underwent several reprocessing steps including manual review and revision of annotation errors followed by reshuffling. We have named this new dataset RareDis2023.

#### **AutoRD Framework**

We present AutoRD, an innovative system designed to automatically extract rare disease information from medical texts and create a knowledge graph. The AutoRD framework is illustrated in (Figure 1) and consists of a pipeline structure that includes

data preprocessing, entity extraction, relation extraction, entity calibration, and knowledge graph construction. The extraction steps, which include entity and relation extraction, are the core components of the system. In these steps, we utilize large language models, along with medical ontologies, to effectively extract information from the texts.

Figure 1. The AutoRD framework processes medical texts as input data and outputs entities related to rare diseases and rare disease triples, which are the results of the extraction process. Subsequently, it constructs a knowledge graph based on these extraction results. During the entity and relation extraction steps, ontology-enhanced large language models are utilized to enhance performance.

### Task Definition

Given a medical text  $T$ , AutoRD is designed to first extract entities  $E = \{E1, E2, ..., En\}$  and relations  $R = \{R1, R2, ..., R\}$ , and then output a knowledge graph  $KG$  based on  $E$  and  $R$ .

The entity types listed in (Table 1) include 'rare\_disease', 'disease', 'symptom\_and\_sign', and 'anaphor'. We group 'symptom' and 'sign' together because they both represent phenotypic abnormalities that may suggest a disease or medical condition. Distinguishing between them is not crucial in the context of rare disease research.

Table 1. Entity type in the entity extraction task. Definitions and examples of all entity types: 'rare\_disease', 'disease', 'symptom\_and\_sign', and 'anaphor'. The definitions and examples are based on the original RareDis dataset definitions. [25]

### Entity Type Definition Example

**rare\_disease** Diseases which affect a small number of people compared to the general population. A disease is often considered to be rare when it affects less than 1 in 2000 individuals. [26]

acquired aplastic anemia, Fryns syndrome, giant cell myocarditis

**disease** An abnormal condition of a part, organ, or system of an organism resulting from various causes such as infection, inflammation, environmental factors, or genetic defect, and characterized by a patterned group of signs and/or symptoms.

cancer, Alzheimer, cardiovascular disease

**symptom\_and\_sign** Signs and symptoms are abnormalities that may suggest a disease. A symptom is a physical or mental problem that a person experiences that may indicate a disease or condition; it is a subjective finding reported by the patient. In contrast, a sign is an observable or otherwise discoverable feature that is considered abnormal. fatigue, dyspnea, pain inflammation, rash, abnormal heart rate, hypothermia

**anaphor** Pronouns, words, or nominal phrases that refer to a rare disease (which is the antecedent of the anaphor) This disease, These diseases

Relation types are displayed in (Table 2), which include 'produces', 'increases\_risk\_of', 'is\_a', 'is\_acron', 'is\_synon', and 'anaphora.' Each relation type represents a specific kind of relationship between a subject and an object, both of which can be any medical term entity.

Table 2. Relation types in the entity extraction task. Definitions of all relation types: 'produces', 'increases\_risk\_of', 'is\_a', 'is\_acron', 'is\_synon', 'anaphora'. The definitions are based on those in the original RareDis dataset. [25]

### Relation Type Definition

**produces** Relation between any disease and a sign or a symptom produced by that disease.

**increases\_risk\_of** Relation between a disease and a disorder, in which the presence of the disease increases the likelihood of the presence of the disorder.

**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.

**anaphora** Relation between an antecedent and an anaphor entity. The antecedent must be a rare disease.

### Data Preprocessing

Before the system performs entity and relation extraction, we first preprocess the data due to the token limit of LLMs. In our system, we use GPT-4, which has a token limit of 8,000. Our maximum length for prompts is approximately 1,000 tokens, including the length of both input and output in the prompt slot. We divide lengthy input documents into segments containing

fewer than 2,000 tokens to adhere to the token limit. To minimize entity relations across segments, we recognize that relations typically occur within a single natural paragraph. Therefore, we segment documents at natural paragraph boundaries, ensuring each segment contains fewer than 2,000 tokens. For paragraphs that have been segmented, we re-extract relations from the segmented middle portion to identify new relations.

We process medical knowledge data from ontology files downloaded from official websites. ORDO encompasses rare diseases that have been discovered up to the present day. From this ontology, we extract the names and definitions of all rare diseases. Mondo offers a unified medical terminology covering various medical concepts, from which we extract all disease, symptom, and sign concepts along with their definitions. Additionally, HOOM is an ontology that annotates the relationships between clinical entities and phenotypic abnormalities and reports their frequencies of occurrence. We extract information from HOOM as triples, consisting of (Rare Disease, Frequency, Phenotype). After preprocessing the ontology files, we can easily integrate medical knowledge from these ontologies into LLMs to enhance their knowledge base.

Data in the RareDis dataset also requires preprocessing for evaluation. The input texts in RareDis are all shorter than 512 tokens and consist of single paragraphs from medical literature, which contains a total of 1,040 data elements (texts and labels). We have corrected some errors in the annotations of the original dataset. To evaluate performance and compare it with the fine-tuning baseline, we divided the dataset into training, validation, and test sets in a 6:2:2 ratio, resulting in 624, 208, and 208 entries, respectively. The training set is used for training fine-tuning models and selecting some exemplars for LLMs, while the validation set is utilized to select the best fine-tuning models during training. In alignment with our task definition, we have merged 'Symptom' and 'Sign' from the original dataset into one entity type, 'symptom\_and\_sign'. We have named the newly processed dataset RareDis2023.

#### Entity Extraction

After preprocessing, AutoRD subsequently carries out entity extraction. We drew inspiration from the concept of chain-of-thought (CoT) [27] to structure the entity extraction process. CoT proposes that tackling complex problems step by step can enhance the performance of LLMs. Similarly, we divided entity extraction into three sub-steps. In each step, an LLM completes a specific, smaller task. This division of the task allows us to integrate external medical knowledge more effectively from ontologies into the LLMs during this process. The three steps are: extract medical terms, extract more terms, and extract entities.

The first step, extract medical terms, extracts basic medical terms from the text. We only employ a string-match algorithm with negation detection in this process. A dictionary is built from the medical ontology Mondo. We use Mondo here because it encompasses nearly all standard medical terms. For each text, we use a string-match algorithm to search any medical terms in ontologies and save candidate medical terms as temporary results. For negation detection, we initially make a list of negation keywords manually, followed by the creation of a regular expression template. This template is then used to identify these keywords and extract the complete terms together. In this ontology, many terms include a free-text definition useful for model comprehension, so we make use of this information in subsequent steps.

The next step, extracting more terms, utilizes LLMs. The prompt template can be found in the left side of (Figure 2). We input the original text and medical terms extracted from the previous step into the LLMs. The LLMs then output additional medical terms. These include terms that are medically relevant but did not directly match an ontology term, including lemmatizations. This process leverages the strong language comprehension capabilities of LLMs for more flexible term extraction. In this step, LLMs also identify anaphors. In the prompt, we first outline the specifics of the current task and provide clear definitions of the entity types. Additionally, we include guidelines for the LLMs on identifying entities which are difficult to recognize. This part is significant and can be continuously improved by medical experts based on the performance of the LLMs and the results they produce. In many cases, LLMs may have misunderstandings in this task, so we need to use prompts to adjust for and correct their interpretations. Finally, we define the output format of the LLMs to be easily parsed, such as in JSON format.

Figure 2. Content of all prompt templates in AutoRD. This figure presents the simplified content of all prompts to provide a clear framework of the prompt structure. The black text represents the original text of the instructions. Grey text indicates a summary of each part of the instructions. Blue text highlights the prompt slots, where external information and inputs can be inserted.

The final step, entity extraction, also involves the use of LLMs. The instruction prompt template can be found in the central part of (Figure 2). The input for this step includes the medical terms and anaphors extracted during the previous step, while the output is comprised of all extracted entities with their appropriate categorizations. The framework of the prompt is formatted like earlier steps; however, additional external information is incorporated into the prompt slots, including medical terms, exemplars, anaphors, and rare disease knowledge. Here, 'rare disease knowledge' refers to terms that can be definitively classified as rare diseases, achieved by matching candidate entities with terms in ORDO.

Furthermore, we utilize the concept of in-context learning (ICL). [28] ICL employs exemplars to enhance the performance of LLMs. Each exemplar is a gold input-output pair, demonstrating the correct method of processing input and generating output for LLMs. This approach is beneficial for guiding the output format of LLMs and providing them with reference material and knowledge to inform their responses. Exemplars can be randomly selected from the training set. After completing these three steps, we can extract entities from text using LLMs.

#### Relation Extraction

In our methodology, relation extraction is conducted after entity extraction. All identified entities are fed into LLMs, which then output the extracted relations. The prompt template used for instructing the LLMs is depicted in the central part of (Figure 2). The underlying logic of this process is akin to that of entity extraction. In the prompt, we initially provide an overview of the current task and establish clear definitions for both entity and relation types. We also include additional considerations for the LLMs to consider during relation extraction. Finally, we define the output format for the LLMs, which is structured to be easily parsed in JSON format. The prompt also contains examples of relation extraction.

For the extraction of rare disease knowledge, we utilize HOOM, an ontology which consists of rare disease-phenotype triples. This ontology provides information on symptoms and signs associated with rare diseases. We employ rare diseases as keys to construct a dictionary, enabling the identification of related triples through string matching. This external medical knowledge aids the LLMs in acquiring information about existing relationships between rare diseases and certain phenotypes.

#### Entity Calibration

Our goal is to construct a knowledge graph based on the extraction results. We consider that many entities might not have defined relationships with other entities. Moreover, after analyzing the extraction results, we observed that entities without any relationships are more likely to be irrelevant or falsely ascribed as medical entities within the context. For instance, the system identifies the term 'disorder' during the entity extraction phase. However, in the relation extraction, the system fails to detect any 'anaphora' or other relations, indicating that it is merely a generic term and can be disregarded in this context. In other instances, some false symptoms and signs are also effectively eliminated. Therefore, we introduce entity calibration as an additional step after relation extraction. The prompt template for this task can be seen on the right side of (Figure 2). In this step, we provide all results obtained from the previous steps and use the LLMs to reanalyze the relationships, filtering out unrelated entities. By combining the results from both entity and relation extraction phases, we obtain the comprehensive outcome of the entire extraction process.

#### Knowledge Graph Construction

After extracting entities and relations, we postprocess the data to prepare for knowledge graph construction. This includes aligning entities, which involves merging identical nodes in the knowledge graph. For each triple, we assess whether the subjects or objects are the same. We begin by converting the names to lowercase and then determining if they match. Additionally, we transform all anaphoric relations to their original names.

After postprocessing, we can easily construct the knowledge graph based on these triples. Specifically, we utilize Neo4j [29] for this purpose. Neo4j is a highly flexible and scalable graph database, designed to store and process complex networks of data. It enables efficient querying and management of interconnected information. Using the API of Neo4j, we add the rare disease triples into the graph database one by one. As a result, we can visualize our rare disease knowledge graph within the Neo4j platform.

#### Evaluation

For the entity and relation extraction component, we quantitatively evaluate the performance of AutoRD using the processed RareDis2023 dataset.

Regarding our method, AutoRD, we specifically selected 'gpt-4-0613', a version of GPT-4 from OpenAI, for the LLMs. We set the LLM's temperature to 0 to ensure the most stable output. For each prediction with exemplars, we randomly choose 5 exemplars from the training set. The performance of AutoRD is evaluated exclusively on the test set, and detailed prompt templates are available in the source code.

To analyze the improvement our method brings compared to using only LLMs, we evaluate the performance of the base LLM. We use the same LLM, 'gpt-4-0613', and maintain all other settings identical to AutoRD. The detailed prompt template can be found in the source code.

For our fine-tuning model baseline, we selected BioClinicalBERT [30]. Entity recognition is performed through token

classification based on BIO labels, and relationships are identified by concatenating the embeddings of two entities, followed by a linear classification. This model is trained on the training set, optimized according to the validation set, and finally evaluated on the test set. Detailed experimental settings are available in the source code.

In terms of evaluation metrics, we use Precision, Recall, and F1 metrics in a named entity recognition setting. For entity and relation extraction, we measure entity F1 and relation F1 respectively. The final overall results are represented by the overall F1 score, calculated as the mean of entity F1 and relation F1. We consider replicated entities in our extraction measurements, which are instances of the same entity occurring at different positions within the text. If the name of an extracted entity is correct, we regard it as true. The evaluation of relation extraction is not limited to correctly identified entities and for all true entities.

In the test set, the entity instances include the follow number of each type: 463 'disease', 1054 'rare\_disease', 1255 'symptoms\_and\_signs,' and 334 'anaphor'. Numbers of each type of relation include 1261 'produces', 62 'increases\_risk\_of', 188 'is\_a', 55 'is\_acron', 22 'is\_synon', and 331 'anaphora'.

For the knowledge graph construction, we conduct qualitative experiments and provide examples of the knowledge graph results. In this case, we only use the extraction results from the test dataset of RareDis2023.

### Results: Main Experimental Results

The primary experimental results are presented in (Table 3), which includes comparisons of entity and relation extraction performance among BioClinicalBERT (a fine-tuning model), Base GPT-4 (a base LLM), and AutoRD (our method). Overall, AutoRD achieves an overall F1 score of 47.3%. The system demonstrates superior performance over these two baselines, with an improvement of 0.8% in overall F1 score compared to the fine-tuning model and a 14.4% improvement compared to the base LLM. Recall is deemed more important than precision in this context because human effort can be used to validate extracted results. The primary goal is to extract all gold entities initially. In terms of recall, our overall recall improved by 18.4% compared to Base GPT-4 and by 6.6% compared to the fine-tuning models. For each extraction objective, AutoRD achieves an overall entity extraction F1 score of 56.1% ('rare\_disease': 83.5%, 'disease': 35.8%, 'symptom\_and\_sign': 46.1%, 'anaphor': 67.5%) and an overall relation extraction F1 score of 38.6% ('produces': 34.7%, 'increases\_risk\_of': 12.4%, 'is\_a': 37.4%, 'is\_acronym': 44.1%, 'is\_synonym': 16.3%, 'anaphora': 57.5%).

Table 3. The main experimental results of entity and relation extraction. Our methods surpass both the fine-tuning model (BioClinical-BERT) and the base LLM (Base GPT-4) in terms of overall F1 score (%).

**Method:** Entity F1 ? Relation F1 ? Overall F1 ?

AutoRD (Ours) 56.1 - 38.6 - 47.3 -

AutoRD w/o Knowledge 53.8 -2.3 36.1 -2.5 45.0 -2.3

AutoRD w/o Exemplars 52.9 -3.2 34.9 -3.7 43.9 -3.4

AutoRD w/o Notice 44.7 -11.4 33.7 -4.9 39.2 -8.1

We conduct an ablation study to analyze the contribution of various components within AutoRD to the overall system. The results are presented in (Table 4), which clearly shows that each key component contributes to the improvement of AutoRD. Note that 'Knowledge' represents the external knowledge sourced from medical ontologies, while 'Notice' refers to the reminders for the LLMs. This study suggests that AutoRD can effectively utilize knowledge from both medical ontologies and exemplars. The 'Notice' component brings a significant improvement of 8.1% in overall F1. The current notices for LLMs in AutoRD have been carefully fine-tuned, demonstrating their effectiveness in adjusting and correcting for the LLMs' interpretations for extraction tasks.

### Error Analysis

We perform error analysis for each entity and relation type. To illustrate the distribution of extraction results, we use two confusion matrices: one for entity extraction results and another for relation extraction results. The results are shown in (Figure 3) and (Figure 4), respectively. The term 'Error' in the 'Predicted' axis refers to entities that have been incorrectly extracted, whereas 'Error' in the 'True' axis denotes real entities that were not extracted. We exclude replicated entities to simplify the computation of the confusion matrices.

In the entity extraction confusion matrix, there is significant confusion between the categories of 'disease' and 'rare\_disease', possibly due to overlapping textual features. The 'symptom\_and\_sign' category exhibits high classification accuracy but is also prone to being misclassified as 'Error', suggesting the need for more distinctive features or additional contextual information in the dataset. The 'Anaphor' category was accurately classified with fewer errors, indicating that the system effectively captures its linguistic features. However, many predicted entities are incorrectly extracted and are labeled as 'Error', indicating that LLMs tend to extract more information with a low precision.



Figure 3. The confusion matrix of the entity extraction task results in RareDis2023.

The confusion matrix for the relation extraction indicates varying degrees of performance across different categories. The 'produces' relation is often identified correctly but also often misclassified as 'Error', indicating recognition issues. 'Increases\_risk\_of' is more frequently an 'Error' than correct, demonstrating the recognition difficulty of this relation. 'Is\_a' has moderate success but high error rates as well. 'Is\_acron' and 'is\_synon' rarely hit true positives and mostly fall into 'Error', possibly due to acronym variability and synonym recognition failure. 'Anaphora' resolution is relatively accurate but also misclassified, hinting at context comprehension challenges. The 'Error' category's high rate of both true and false positives is primarily affected by the false results from entity extraction step before it.

Figure 4. The confusion matrix of the relation extraction task results in RareDis2023.

### Qualitative Results

Qualitative results are showcased in (Figure 5), which depicts all the extracted results from the RareDis2023 dataset. Our qualitative results have been validated by medical experts and have shown satisfactory outcomes. This visualization provides a global perspective, highlighting the relationships among various rare diseases and their associated signs and symptoms in a concise knowledge graph.

Figure 5. The example of constructed knowledge graph from RareDis2023. The result is a clear and well-structured knowledge graph.

Specific extraction results of the knowledge graph are depicted in (Figure 6). This figure offers visualizations from a local perspective, illustrating an ideal structure of the knowledge graph. In this structure, rare diseases are positioned at the center of radial formations, with connections extending to entities like symptoms and signs. For example, in Figure 6, the rare disease 'Turcot syndrome' is associated with 'abdominal pain', 'bleeding', 'fatigue', etc.

Figure 6. An example provides a detailed view of a specific local section of the constructed knowledge graph.

Additionally, we experimented with training specialized medical LLMs and compared their performance. Specifically, we utilized Camel-Platypus2-70B [31], a healthcare-tailored model that is an extension of Llama-2 [32], through continuous training. Our experiments revealed that, without specific training, this type of model struggles to execute complex tasks such as joint entity and relation extraction. It appears that the inherent medical knowledge is not readily applicable in these scenarios.

### Conclusions: Principal Results

Our experimentation demonstrates the effectiveness of our proposed system, AutoRD. It significantly improves upon the base LLM and even outperforms fine-tuning models without requiring any training. Within several designs, the incorporation of medical ontologies has notably enhanced the LLMs by addressing gaps in medical knowledge. Furthermore, the results achieved in knowledge graph construction by our system are commendable. We highlight the advantage of LLMs in low-resource scenarios like rare disease extraction, showcasing their vast potential. Our meticulously designed system, AutoRD, substantiates this claim. The emergence of LLMs is generating unparalleled opportunities in the phenotyping of rare diseases. These models facilitate the automatic identification and extraction of concepts related to these diseases. Our prompts are easily adjustable due to their clear structure, allowing for simple modifications. Additionally, medical knowledge derived from external sources can be updated at any time within the AutoRD system.

### Limitations

Nevertheless, there is considerable potential for further improvement with respect to AutoRD. For instance, integrating advanced text processing tools and specialized medical tools into our system could amplify its capabilities. In the future, we can deploy more powerful medical LLMs as base models to enhance medical understanding. Moreover, medical experts can contribute more tailored prompts to improve LLMs' performance.

Our work has potential limitations and avenues for extension. For example, we have only evaluated AutoRD on a single dataset,

so the results may not fully reflect the system's performance across the entire spectrum of rare diseases or in other long-text scenarios. Additionally, the prompts we designed are intuitive, but there is still room for continuous tuning and experimentation of different prompts. We acknowledge that AutoRD may not be the optimal LLM application for this task, yet it significantly improves upon the baseline performance of LLMs. This work aims at demonstrating the potential of LLM applications in the healthcare field.

### Conclusions

We developed AutoRD as an automatic and end-to-end system specifically designed for extracting information about rare diseases from text and building knowledge graphs. This system incorporates various innovative designs, including ontology-enhanced LLMs, to augment its medical knowledge base. The effectiveness of AutoRD has been validated through experimental evaluations, highlighting the valuable applications of LLMs in healthcare. In the future, we intend to explore how to broaden the application of LLMs to other healthcare processes in low-resource environments, including the identification of potential rare diseases in patients. Furthermore, ongoing research is focused on improving the performance of LLMs with respect to informing and augmenting medical decision-making strategies.

(JMIR Preprints 22/05/2024:60665)

DOI: <https://doi.org/10.2196/preprints.60665>

### Preprint Settings

1) Would you like to publish your submitted manuscript as preprint?

✓ **Please make my preprint PDF available to anyone at any time (recommended).**

Please make my preprint PDF available only to logged-in users; I understand that my title and abstract will remain visible to all users.

Only make the preprint title and abstract visible.

No, I do not wish to publish my submitted manuscript as a preprint.

2) If accepted for publication in a JMIR journal, would you like the PDF to be visible to the public?

✓ **Yes, please make my accepted manuscript PDF available to anyone at any time (Recommended).**

Yes, but please make my accepted manuscript PDF available only to logged-in users; I understand that the title and abstract will remain visible to all users.

Yes, but only make the title and abstract visible (see Important note, above). I understand that if I later pay to participate in a JMIR journal, my full manuscript will be made available to all users.

## Original Manuscript

# AutoRD: An Automatic and End-to-End System for Rare Disease Knowledge Graph Construction Based on Ontology-enhanced Large Language Models

## Abstract

**Background:** Rare diseases affect millions worldwide but often face limited research focus due to their low prevalence. This results in prolonged diagnoses and a lack of approved therapies. Recent advancements in Large Language Models (LLMs) have shown promise in automating the extraction of medical information, offering potential to improve rare disease diagnosis and management.

**Objective:** Our objective is to create an end-to-end system called AutoRD, which automates extracting information from medical text about rare diseases. We conducted various experiments to evaluate the performance of AutoRD and highlighted its strengths and limitations in this paper.

**Methods:** AutoRD is a pipeline system involving data preprocessing, entity extraction, relation extraction, entity calibration, and knowledge graph construction. We implement this using large language models and medical knowledge graphs developed from open-source medical ontologies. We quantitatively evaluate our system on entity extraction, relation extraction, and the performance of knowledge graph construction.

**Results:** AutoRD achieves an average of entity F1 score and relation F1 score of 47.3%, a 14.4% improvement compared to the base LLM. In detail, AutoRD achieves an overall entity extraction F1 score of 56.1% ('rare\_disease': 83.5%, 'disease': 35.8%, 'symptom\_and\_sign': 46.1%, 'anaphor': 67.5%) and an overall relation extraction F1 score of 38.6% ('produces': 34.7%, 'increases\_risk\_of': 12.4%, 'is\_a': 37.4%, 'is\_acronym': 44.1%, 'is\_synonym': 16.3%, 'anaphora': 57.5%). The qualitative experiment also demonstrates that the constructed knowledge graph's performance is respectable.

**Conclusions:** AutoRD is an automated end-to-end system for extracting rare disease information from text to build knowledge graphs. It uses ontology-enhanced LLMs for a robust medical knowledge base. The superior performance of AutoRD is validated by experimental evaluations, demonstrating the potential of LLMs in healthcare.

**Trial Registration:** N/A

**Keywords:** Rare Disease; Large Language Models; Knowledge Graph; Text Mining; Natural Language Processing.

## Introduction

### Objectives

Rare diseases, also known as orphan diseases, are relatively uncommon in isolation and sometimes receive less individual attention in medical research due to their low prevalence. [1] The likelihood of an individual being affected by a rare disease is relatively low. However, when considering the global population, many individuals are impacted. In the United States, rare diseases affect approximately 30 million people; [2] globally, the number rises to between 300 and 400 million. [Error: Reference source not found] Furthermore, the rare disease patient population, distributed across 5,000 to 10,000 distinct diseases, [4] suffers from a significant lack of medical knowledge due to the rarity of a given illness. Consequently, patients often face prolonged and costly diagnostic processes and intensive treatments, with many of these diseases lacking approved therapies. [5,6] This situation underscores the substantial burden placed on both patients and healthcare systems. [7]

Online resources, including open-source databases, offer valuable references for medical professionals, contributing to the development of a comprehensive rare disease knowledge system. Examples of such databases include the Unified Medical Language System (UMLS) [8] and the Orphanet Human Phenotype Ontology (HPO). [9,10] Specifically, Orphanet's database provides detailed information linking rare diseases, genes, and phenotypes, which greatly aids in the identification and diagnosis of rare diseases, among other related processes. However, these databases require considerable human effort for curation and maintenance. Therefore, there is an urgent need to develop methods that can support the process of establishing and enhancing rare disease medical knowledge systems.

Natural Language Processing (NLP) techniques are instrumental in automatically processing unstructured text to extract structured and clinically relevant information. This technique is especially beneficial for information extraction and knowledge discovery in the medical field. Recently, Large Language Models (LLMs) have demonstrated exceptional proficiency in language understanding and generation, garnering significant attention in the open NLP domain. [11,12] Their ease of use allows humans to complete a wide range of complex tasks in everyday life. Moreover, the extensive knowledge stored within their parameters equips them to excel in domain-specific applications, such as medicine and healthcare. [13]

Current research is beginning to evaluate the capabilities of the most powerful LLMs, such as ChatGPT and GPT-4, across various medical applications. These applications include licensing examinations, [14] question answering, [15] and medical education. [16] Notably, several studies have demonstrated that LLMs are effective few-shot medical Named Entity Recognition (NER) extractors, exhibiting superior few-shot learning capabilities compared to other NLP methods. [17] In the context of rare diseases, where resources are often limited, LLMs emerge as valuable tools for extracting information about these conditions, showcasing their utility in enhancing medical knowledge systems.

In this paper we introduce Automated Rare Disease Mining (AutoRD) as an efficient tool for extracting information about rare diseases and constructing corresponding knowledge graphs. The system processes unstructured medical text as input and outputs extraction results and a knowledge graph. It is comprised of several key stages: data preprocessing, entity extraction, relation extraction, entity alignment, and knowledge graph construction. Among these, entity and relation extraction are the most critical parts. AutoRD is an LLM-based system built upon GPT-4. [12] We employ prompts as instructions to guide the LLMs through the entity and relation extraction processes. The model leverages its strong zero-shot capabilities to identify and extract entities and to analyze relationships between them. Although LLMs are pre-trained with extensive knowledge, they sometimes lack precise medical information. To address this, we enhance the LLM's medical knowledge using rare disease and phenotype ontologies. This is achieved by designing sophisticated prompts that incorporate relevant knowledge. We conducted experiments to evaluate the system and identified the advantages and limitations of AutoRD. In summary, our contributions can be summarized as follows:

1. We propose AutoRD, an automated end-to-end system which efficiently extracts rare disease information from text and builds knowledge graphs. This is a useful and practical system which can help medical professionals discover information about rare diseases.
2. We use ontology-enhanced LLMs in the module of rare disease entity extraction and relation extraction. This approach harnesses the few-shot learning capabilities of LLMs and integrates medical knowledge from ontologies, resulting in an improved performance beyond what

LLMs achieve alone.

3. We conduct experiments and provide extensive analysis to demonstrate the effectiveness of AutoRD.

## Background and significance

Several studies have employed machine learning methods to support and enhance the medical management and process of rare diseases. Sanjak et al [18] introduced an innovative method for clustering over 3,000 rare diseases using node embeddings within a knowledge graph. This approach facilitates a deeper understanding of the relationships between different diseases and opens possibilities for drug repurposing. Alsentzer et al [19] developed Shepherd, a deep learning model designed for diagnosing rare diseases. This model effectively leverages clinical and genetic patient data along with existing medical knowledge to uncover new disease-gene associations. This work exemplifies the potential of AI in medical diagnostics, even in scenarios with limited labeled data. Rashid et al. [20] explored a unique approach in rare disease research through the National Mesothelioma Virtual Bank. They utilized REDCap and a web portal query tool to integrate and manage clinical data from multiple institutions. This method demonstrates the power of combining data management tools and web technologies to enhance research and collaboration in the field of rare diseases.

The advent of LLMs has led to their increasing application in the medical field. Datta et al [21] developed AutoCriteria, an LLM-based information extraction system that has shown high accuracy and generalizability in extracting detailed eligibility criteria from clinical trial documents for various diseases. This represents a scalable solution for clinical trial applications. In the context of rare diseases, there have been specific research efforts utilizing LLMs. Shyr et al [22] explored the performance of ChatGPT in extracting rare disease phenotypes from unstructured text, using zero- and few-shot learning techniques. This study demonstrated potential in certain scenarios, particularly with tailored prompts and minimal data. Oniani et al [23] proposed Models-Vote Prompting (MVP), an approach that improves LLM performance in few-shot learning scenarios by aggregating outputs from multiple LLMs through majority voting.

However, these studies on LLM applications for rare diseases are still preliminary. Both focused on evaluating the basic capabilities of LLMs in identifying rare diseases or used simple prompt ensembles to slightly enhance LLM performance. They did not explore the task of extracting relationships between rare diseases and related phenotypes. Additionally, these studies primarily explore basic applications of LLMs and do not extend to a comprehensive LLM-based system. Building on these foundational works, our research continues to delve into the use of LLMs for rare disease applications. Unlike prior efforts, we propose an integrated and useful system aimed at extracting rare disease information from unstructured text. The elaborate methods incorporated into this system significantly enhance extraction accuracy compared to the use of pure LLMs, marking a substantial advancement in this field.

## Methods

### Data

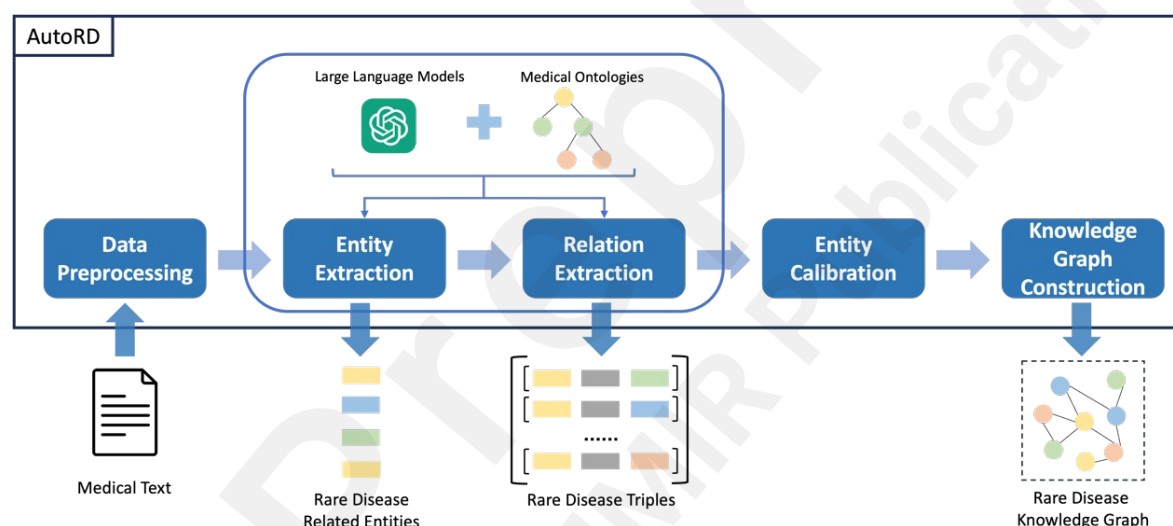
To improve the medical understanding of LLMs, we incorporated three medical ontologies into AutoRD: Orphanet Rare Disease Ontology (ORDO) [10], HPO-ORDO Ontology Module (HOOM) [10], and Mondo Disease Ontology (Mondo) [24].

For assessing the entity and relation extraction capabilities of AutoRD, the RareDis-v1 dataset [25] was employed. Prior to use, this dataset underwent several reprocessing steps including manual review and revision of annotation errors followed by reshuffling. We have named this new dataset RareDis2023.

## AutoRD Framework

We present AutoRD, an innovative system designed to automatically extract rare disease information from medical texts and create a knowledge graph. The AutoRD framework is illustrated in (Figure 1) and consists of a pipeline structure that includes data preprocessing, entity extraction, relation extraction, entity calibration, and knowledge graph construction. The extraction steps, which include entity and relation extraction, are the core components of the system. In these steps, we utilize large language models, along with medical ontologies, to effectively extract information from the texts.

Figure 1. The AutoRD framework processes medical texts as input data and outputs entities related to rare diseases and rare disease triples, which are the results of the extraction process. Subsequently, it constructs a knowledge graph based on these extraction results. During the entity and relation extraction steps, ontology-enhanced large language models are utilized to enhance performance.



## Task Definition

Given a medical text  $T$ , AutoRD is designed to first extract entities  $E = \{E_1, E_2, \dots, E_n\}$  and relations  $R = \{R_1, R_2, \dots, R\}$ , and then output a knowledge graph  $KG$  based on  $E$  and  $R$ .

The entity types listed in (Table 1) include 'rare\_disease', 'disease', 'symptom\_and\_sign', and 'anaphor'. We group 'symptom' and 'sign' together because they both represent phenotypic abnormalities that may suggest a disease or medical condition. Distinguishing between them is not crucial in the context of rare disease research.

Table 1. Entity type in the entity extraction task. Definitions and examples of all entity types: 'rare\_disease', 'disease', 'symptom\_and\_sign', and 'anaphor'. The definitions and examples are based on the original RareDis dataset definitions. [25]

Entity Type	Definition	Example
rare_disease	Diseases which affect a small number of people compared to the general population. A disease is often considered to be rare when it affects less than 1 in 2000 individuals. [26]	acquired aplastic anemia, Fryns syndrome, giant cell myocarditis
disease	An abnormal condition of a part, organ, or system of an organism resulting from various causes such as infection, inflammation, environmental factors, or genetic defect, and characterized by a patterned group of signs and/or symptoms.	cancer, Alzheimer, cardiovascular disease
symptom_and_sign	Signs and symptoms are abnormalities that may suggest a disease. A symptom is a physical or mental problem that a person experiences that may indicate a disease or condition; it is a subjective finding reported by the patient. In contrast, a sign is an observable or otherwise discoverable feature that is considered abnormal.	fatigue, dyspnea, pain inflammation, rash, abnormal heart rate, hypothermia
anaphor	Pronouns, words, or nominal phrases that refer to a rare disease (which is the antecedent of the anaphor)	This disease, These diseases

Relation types are displayed in (Table 2), which include 'produces', 'increases\_risk\_of', 'is\_a', 'is\_acron', 'is\_synon', and 'anaphora.' Each relation type represents a specific kind of relationship between a subject and an object, both of which can be any medical term entity.

Table 2. Relation types in the entity extraction task. Definitions of all relation types: 'produces', 'increases\_risk\_of', 'is\_a', 'is\_acron', 'is\_synon', 'anaphora'. The definitions are based on those in the original RareDis dataset. [25]

Relation Type	Definition
produces	Relation between any disease and a sign or a symptom produced by that disease.
increases_risk_of	Relation between a disease and a disorder, in which the presence of the disease increases the likelihood of the presence of the disorder.
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.
anaphora	Relation between an antecedent and an anaphor entity. The antecedent must be a rare disease.



## Data Preprocessing

Before the system performs entity and relation extraction, we first preprocess the data due to the token limit of LLMs. In our system, we use GPT-4, which has a token limit of 8,000. Our maximum length for prompts is approximately 1,000 tokens, including the length of both input and output in the prompt slot. We divide lengthy input documents into segments containing fewer than 2,000 tokens to adhere to the token limit. To minimize entity relations across segments, we recognize that relations typically occur within a single natural paragraph. Therefore, we segment documents at natural paragraph boundaries, ensuring each segment contains fewer than 2,000 tokens. For paragraphs that have been segmented, we re-extract relations from the segmented middle portion to identify new relations.

We process medical knowledge data from ontology files downloaded from official websites. ORDO encompasses rare diseases that have been discovered up to the present day. From this ontology, we extract the names and definitions of all rare diseases. Mondo offers a unified medical terminology covering various medical concepts, from which we extract all disease, symptom, and sign concepts along with their definitions. Additionally, HOOM is an ontology that annotates the relationships between clinical entities and phenotypic abnormalities and reports their frequencies of occurrence. We extract information from HOOM as triples, consisting of (Rare Disease, Frequency, Phenotype). After preprocessing the ontology files, we can easily integrate medical knowledge from these ontologies into LLMs to enhance their knowledge base.

Data in the RareDis dataset also requires preprocessing for evaluation. The input texts in RareDis are all shorter than 512 tokens and consist of single paragraphs from medical literature, which contains a total of 1,040 data elements (texts and labels). We have corrected some errors in the annotations of the original dataset. To evaluate performance and compare it with the fine-tuning baseline, we divided the dataset into training, validation, and test sets in a 6:2:2 ratio, resulting in 624, 208, and 208 entries, respectively. The training set is used for training fine-tuning models and selecting some exemplars for LLMs, while the validation set is utilized to select the best fine-tuning models during training. In alignment with our task definition, we have merged 'Symptom' and 'Sign' from the original dataset into one entity type, 'symptom\_and\_sign'. We have named the newly processed dataset RareDis2023.

## Entity Extraction

After preprocessing, AutoRD subsequently carries out entity extraction. We drew inspiration from the concept of chain-of-thought (CoT) [27] to structure the entity extraction process. CoT proposes that tackling complex problems step by step can enhance the performance of LLMs. Similarly, we divided entity extraction into three sub-steps. In each step, an LLM completes a specific, smaller task. This division of the task allows us to integrate external medical knowledge more effectively from ontologies into the LLMs during this process. The three steps are: extract medical terms, extract more terms, and extract entities.

The first step, extract medical terms, extracts basic medical terms from the text. We only employ a string-match algorithm with negation detection in this process. A dictionary is built from the medical ontology Mondo. We use Mondo here because it encompasses nearly all standard medical terms. For each text, we use a string-match algorithm to search any medical terms in ontologies and save candidate medical terms as temporary results. For negation detection, we initially make a list of negation keywords manually, followed by the creation of a regular expression template. This template is then used to identify these keywords and extract the complete terms together. In this

ontology, many terms include a free-text definition useful for model comprehension, so we make use of this information in subsequent steps.

The next step, extracting more terms, utilizes LLMs. The prompt template can be found in the left side of (Figure 2). We input the original text and medical terms extracted from the previous step into the LLMs. The LLMs then output additional medical terms. These include terms that are medically relevant but did not directly match an ontology term, including lemmatizations. This process leverages the strong language comprehension capabilities of LLMs for more flexible term extraction. In this step, LLMs also identify anaphors. In the prompt, we first outline the specifics of the current task and provide clear definitions of the entity types. Additionally, we include guidelines for the LLMs on identifying entities which are difficult to recognize. This part is significant and can be continuously improved by medical experts based on the performance of the LLMs and the results they produce. In many cases, LLMs may have misunderstandings in this task, so we need to use prompts to adjust for and correct their interpretations. Finally, we define the output format of the LLMs to be easily parsed, such as in JSON format.

Figure 2. Content of all prompt templates in AutoRD. This figure presents the simplified content of all prompts to provide a clear framework of the prompt structure. The black text represents the original text of the instructions. Grey text indicates a summary of each part of the instructions. Blue text highlights the prompt slots, where external information and inputs can be inserted.

Extract More Terms	Extract Entities	Extract Relations	Calibrate Entities
<pre> <b>## Task:</b> (Extract entities: "anaphor", "medical_term"...)</pre> <p><b>## Definition:</b> (Define entities, including "rare_disease", "disease", "symptom_and_sign", "anaphor", "medical_term".)</p> <p><b>## Notice:</b> (Remind the model of some considerations...)</p> <p>(Provide extracted "medical_terms".) {medical_terms}</p> <p><b>## Input Text:</b> (Provide text.) {text}</p> <p><b>## Output format</b> <b>## Entity Representation:</b> (Define entity representation.)</p> <p><b>## Your output should be a single JSON object in the following format:</b> (Define the format of response.)</p> <p><b>## Output:</b></p>	<pre> <b>## Task:</b> (Extract entities: "rare_disease", "disease", "symptom_and_sign", "anaphor"...)</pre> <p><b>## Definition:</b> <b>## Entity:</b> (Define entities, including "rare_disease", "disease", "symptom_and_sign", "anaphor".)</p> <p><b>## Notice:</b> (Remind the model of some considerations...)</p> <p><b>## Examples</b> Here are some examples of input text and target entities: {examples}</p> <p><b>## Input (Give Passage):</b> {text}</p> <p><b>## Extracted Medical Terms:</b> {medical_terms}</p> <p><b>## Extracted Anaphor:</b> {anaphor}</p> <p><b>## Rare Disease Knowledge:</b> {rare_disease_knowledge}</p> <p><b>## Output format</b> <b>## Entity Representation:</b> (Define entity representation.)</p> <p><b>## Your output should be a single JSON object in the following format:</b> (Define the format of response.)</p> <p><b>## Output:</b></p>	<pre> <b>## Task:</b> (Extract relations: "produces", "increases_risk_of", "is_a", "is_acron", "is_synonym", "anaphora"...)</pre> <p><b>## Definition:</b> <b>## Entity:</b> (Define entities, including "rare_disease", "disease", "symptom_and_sign", "anaphor".)</p> <p><b>## Relation:</b> (Define relations, including "produces", "increases_risk_of", "is_a", "is_acron", "is_synonym", "anaphora".)</p> <p><b>## Notice:</b> (Remind the model of some considerations...)</p> <p><b>## Input (Give Passage):</b> {text}</p> <p><b>## Examples</b> Here are some examples of input text and target entities: {examples}</p> <p><b>## Input (Give Passage):</b> {text}</p> <p><b>## Extracted Entities:</b> {entities}</p> <p><b>## Rare Disease Knowledge:</b> {rare_disease_knowledge}</p> <p><b>## Output format</b> <b>## Entity Representation:</b> (Define entity representation.)</p> <p><b>## Relation Representation:</b> (Define relation representation.)</p> <p><b>## Your output should be a single JSON object in the following format:</b> (Define the format of response.)</p> <p><b>## Output:</b></p>	<pre> <b>## Task:</b> (Verify all extracted entities and relations...)</pre> <p><b>## Definition:</b> <b>## Entity:</b> (Define entities, including "rare_disease", "disease", "symptom_and_sign", "anaphor".)</p> <p><b>## Relation:</b> (Define relations, including "produces", "increases_risk_of", "is_a", "is_acron", "is_synonym", "anaphora".)</p> <p><b>## Notice:</b> (Remind the model of some considerations...)</p> <p><b>## Input (Give Passage):</b> {text}</p> <p><b>## Extracted Entities:</b> {entities}</p> <p><b>## Extracted Relations:</b> {relations}</p> <p><b>## Output format</b> <b>## Entity Representation:</b> (Define entity representation.)</p> <p><b>## Relation Representation:</b> (Define relation representation.)</p> <p><b>## Your output should be a single JSON object in the following format:</b> (Define the format of response.)</p> <p><b>## Output:</b></p>

The final step, entity extraction, also involves the use of LLMs. The instruction prompt template can be found in the central part of (Figure 2). The input for this step includes the medical terms and anaphors extracted during the previous step, while the output is comprised of all extracted entities with their appropriate categorizations. The framework of the prompt is formatted like earlier steps; however, additional external information is incorporated into the prompt slots, including medical terms, exemplars, anaphors, and rare disease knowledge. Here, 'rare disease knowledge' refers to terms that can be definitively classified as rare diseases, achieved by matching candidate entities with terms in ORDO.

Furthermore, we utilize the concept of in-context learning (ICL). [28] ICL employs exemplars to enhance the performance of LLMs. Each exemplar is a gold input-output pair, demonstrating the correct method of processing input and generating output for LLMs. This approach is beneficial for guiding the output format of LLMs and providing them with reference material and knowledge to inform their responses. Exemplars can be randomly selected from the training set. After completing

these three steps, we can extract entities from text using LLMs.

## Relation Extraction

In our methodology, relation extraction is conducted after entity extraction. All identified entities are fed into LLMs, which then output the extracted relations. The prompt template used for instructing the LLMs is depicted in the central part of (Figure 2). The underlying logic of this process is akin to that of entity extraction. In the prompt, we initially provide an overview of the current task and establish clear definitions for both entity and relation types. We also include additional considerations for the LLMs to consider during relation extraction. Finally, we define the output format for the LLMs, which is structured to be easily parsed in JSON format. The prompt also contains examples of relation extraction.

For the extraction of rare disease knowledge, we utilize HOOM, an ontology which consists of rare disease-phenotype triples. This ontology provides information on symptoms and signs associated with rare diseases. We employ rare diseases as keys to construct a dictionary, enabling the identification of related triples through string matching. This external medical knowledge aids the LLMs in acquiring information about existing relationships between rare diseases and certain phenotypes.

## Entity Calibration

Our goal is to construct a knowledge graph based on the extraction results. We consider that many entities might not have defined relationships with other entities. Moreover, after analyzing the extraction results, we observed that entities without any relationships are more likely to be irrelevant or falsely ascribed as medical entities within the context. For instance, the system identifies the term 'disorder' during the entity extraction phase. However, in the relation extraction, the system fails to detect any 'anaphora' or other relations, indicating that it is merely a generic term and can be disregarded in this context. In other instances, some false symptoms and signs are also effectively eliminated. Therefore, we introduce entity calibration as an additional step after relation extraction. The prompt template for this task can be seen on the right side of (Figure 2). In this step, we provide all results obtained from the previous steps and use the LLMs to reanalyze the relationships, filtering out unrelated entities. By combining the results from both entity and relation extraction phases, we obtain the comprehensive outcome of the entire extraction process.

## Knowledge Graph Construction

After extracting entities and relations, we postprocess the data to prepare for knowledge graph construction. This includes aligning entities, which involves merging identical nodes in the knowledge graph. For each triple, we assess whether the subjects or objects are the same. We begin by converting the names to lowercase and then determining if they match. Additionally, we transform all anaphoric relations to their original names.

After postprocessing, we can easily construct the knowledge graph based on these triples. Specifically, we utilize Neo4j [29] for this purpose. Neo4j is a highly flexible and scalable graph database, designed to store and process complex networks of data. It enables efficient querying and management of interconnected information. Using the API of Neo4j, we add the rare disease triples into the graph database one by one. As a result, we can visualize our rare disease knowledge graph within the Neo4j platform.

## Evaluation

For the entity and relation extraction component, we quantitatively evaluate the performance of AutoRD using the processed RareDis2023 dataset.

Regarding our method, AutoRD, we specifically selected 'gpt-4-0613', a version of GPT-4 from OpenAI, for the LLMs. We set the LLM's temperature to 0 to ensure the most stable output. For each prediction with exemplars, we randomly choose 5 exemplars from the training set. The performance of AutoRD is evaluated exclusively on the test set, and detailed prompt templates are available in the source code.

To analyze the improvement our method brings compared to using only LLMs, we evaluate the performance of the base LLM. We use the same LLM, 'gpt-4-0613', and maintain all other settings identical to AutoRD. The detailed prompt template can be found in the source code.

For our fine-tuning model baseline, we selected BioClinicalBERT [30]. Entity recognition is performed through token classification based on BIO labels, and relationships are identified by concatenating the embeddings of two entities, followed by a linear classification. This model is trained on the training set, optimized according to the validation set, and finally evaluated on the test set. Detailed experimental settings are available in the source code.

In terms of evaluation metrics, we use Precision, Recall, and F1 metrics in a named entity recognition setting. For entity and relation extraction, we measure entity F1 and relation F1 respectively. The final overall results are represented by the overall F1 score, calculated as the mean of entity F1 and relation F1. We consider replicated entities in our extraction measurements, which are instances of the same entity occurring at different positions within the text. If the name of an extracted entity is correct, we regard it as true. The evaluation of relation extraction is not limited to correctly identified entities and for all true entities.

In the test set, the entity instances include the follow number of each type: 463 'disease', 1054 'rare\_disease', 1255 'symptoms\_and\_signs,' and 334 'anaphor'. Numbers of each type of relation include 1261 'produces', 62 'increases\_risk\_of', 188 'is\_a', 55 'is\_acron', 22 'is\_synon', and 331 'anaphora'.

For the knowledge graph construction, we conduct qualitative experiments and provide examples of the knowledge graph results. In this case, we only use the extraction results from the test dataset of RareDis2023.

## Results

### Main Experimental Results

The primary experimental results are presented in (Table 3), which includes comparisons of entity and relation extraction performance among BioClinicalBERT (a fine-tuning model), Base GPT-4 (a base LLM), and AutoRD (our method). Overall, AutoRD achieves an overall F1 score of 47.3%. The system demonstrates superior performance over these two baselines, with an improvement of 0.8% in overall F1 score compared to the fine-tuning model and a 14.4% improvement compared to the base LLM. Recall is deemed more important than precision in this context because human effort can be used to validate extracted results. The primary goal is to extract all gold entities initially. In terms

of recall, our overall recall improved by 18.4% compared to Base GPT-4 and by 6.6% compared to the fine-tuning models. For each extraction objective, AutoRD achieves an overall entity extraction F1 score of 56.1% ('rare\_disease': 83.5%, 'disease': 35.8%, 'symptom\_and\_sign': 46.1%, 'anaphor': 67.5%) and an overall relation extraction F1 score of 38.6% ('produces': 34.7%, 'increases\_risk\_of': 12.4%, 'is\_a': 37.4%, 'is\_acronym': 44.1%, 'is\_synonym': 16.3%, 'anaphora': 57.5%).

Table 3. The main experimental results of entity and relation extraction. Our methods surpass both the fine-tuning model (BioClinical-BERT) and the base LLM (Base GPT-4) in terms of overall F1 score (%).

Method	type		Precision	Recall	F1
BioClinical BERT	Entity	rare_disease	80.5	87.7	83.9
		disease	53.2	46.0	49.3
		symptom_and_sign	62.3	62.5	62.4
		anaphor	89.9	93.7	91.7
		entity_overall	70.9	72.0	71.4
	Relation	produces	49.7	13.6	21.4
		increases_risk_of	0.0	0.0	0.0
		is_a	80.0	4.3	8.1
		is_acron	0.0	0.0	0.0
		is_synon	0.0	0.0	0.0
		anaphora	82.9	23.3	36.3
		relation_overall	57.0	13.4	21.7
	overall		64.0	42.7	46.5
	Entity	rare_disease	94.8	38.4	54.7
disease		22.5	59.8	32.7	
symptom_and_sign		48.7	41.7	44.9	
anaphor		45.2	69.5	54.7	
entity_overall		43.2	46.3	44.7	
Base GPT4	Relation	produces	26.5	3.3	5.8
		increases_risk_of	9.6	8.1	8.8
		is_a	33.0	30.9	31.9
		is_acron	17.1	21.8	19.2
		is_synon	0.0	0.0	0.0
		anaphora	41.7	55.3	47.6
		relation_overall	32.5	15.6	21.1
	overall		37.9	30.9	32.9
	Entity	rare_disease	93.1	75.6	83.5
		disease	26.6	54.9	35.8
symptom_and_sign		45.8	46.5	46.1	
anaphor		59.0	79.0	67.5	
entity_overall		51.8	61.1	56.1	
AutoRD (Ours)	Relation	produces	37.2	32.4	34.7
		increases_risk_of	11.8	13.1	12.4
		is_a	41.4	34.0	37.4
		is_acron	49.2	40.0	44.1

is_synon	12.8	22.7	16.3
anaphora	52.4	63.7	57.5
relation_overall	39.8	37.5	38.6
overall	45.8	49.3	<b>47.3</b>

First, comparing Base GPT-4 with BioClinicalBERT reveals that BioClinicalBERT, with its ample training data, aligns well with the original dataset's distribution and excels in multiple metrics. However, LLMs were not trained to fit this distribution. This discrepancy leads to issues such as misclassifications and ambiguous entity boundaries in LLMs. In relation extraction, however, the fine-tuned model fails to detect relations like 'increases\_risk\_of', 'is\_acron', and 'is\_synon', whereas Base GPT-4 detects all but 'is\_synon'. This demonstrates the strong zero-shot capabilities of LLMs, especially for relations sparsely represented in the training set.

When comparing AutoRD with BioClinicalBERT, it is apparent that while our method falls somewhat short in entity extraction, it excels in relation extraction. Specifically, the entity F1 score is 15.1% lower than this baseline, but the relation F1 score is 16.9% higher. This results in an overall performance that is 0.8% better. Relation extraction plays a pivotal role in the construction of knowledge graphs, as it is essential towards understanding the underlying relationships between entities. AutoRD leverages the few-shot learning capability of LLMs to better analyze relationships between medical entities. However, we observed a lower precision in AutoRD, primarily because it identifies too many entities as 'diseases' and sometimes misclassifies 'symptom\_and\_sign' according to its extraction results.

Furthermore, when comparing AutoRD with Base GPT-4, it is evident that our method significantly improves performance by 18.4%. The most notable improvement is a 37.2% increase in the recall of rare disease entities from Base GPT-4. Base GPT-4, with its poor analysis capability and lack of sufficient medical knowledge, struggles to identify all types of rare diseases. Overall, our approach demonstrates substantial improvements in most metrics.

## Ablation Study

Table 4. The results of the ablation experiment. It clearly shows that each key component contributes to the improvement of AutoRD in terms of F1 score (%). The symbol ( $\nabla$ ) represents the magnitude of the performance drop.

Method		Entity F1	$\nabla$	Relation F1	$\nabla$	Overall F1	$\nabla$
AutoRD (Ours)		<b>56.1</b>	-	<b>38.6</b>	-	<b>47.3</b>	-
AutoRD	w/o	53.8	-2.3	36.1	-2.5	45.0	-2.3
Knowledge							
AutoRD	w/o	52.9	-3.2	34.9	-3.7	43.9	-3.4
Exemplars							
AutoRD w/o Notice		44.7	-11.4	33.7	-4.9	39.2	-8.1

We conduct an ablation study to analyze the contribution of various components within AutoRD to the overall system. The results are presented in (Table 4), which clearly shows that each key component contributes to the improvement of AutoRD. Note that 'Knowledge' represents the external knowledge sourced from medical ontologies, while 'Notice' refers to the reminders for the LLMs. This study suggests that AutoRD can effectively utilize knowledge from both medical ontologies and exemplars. The 'Notice' component brings a significant improvement of 8.1% in overall F1. The

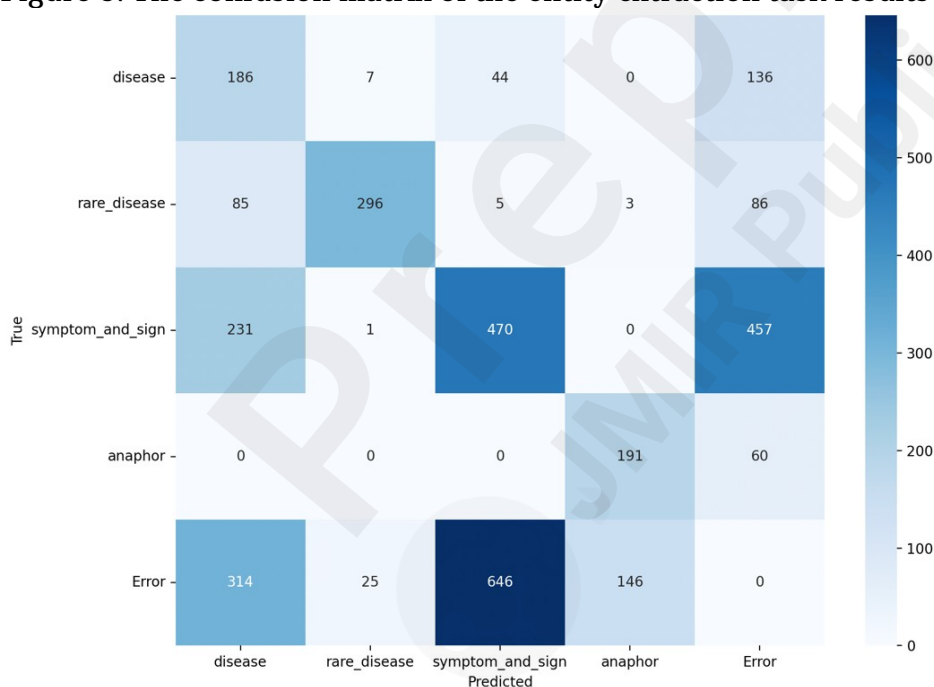
current notices for LLMs in AutoRD have been carefully fine-tuned, demonstrating their effectiveness in adjusting and correcting for the LLMs' interpretations for extraction tasks.

## Error Analysis

We perform error analysis for each entity and relation type. To illustrate the distribution of extraction results, we use two confusion matrices: one for entity extraction results and another for relation extraction results. The results are shown in (Figure 3) and (Figure 4), respectively. The term 'Error' in the 'Predicted' axis refers to entities that have been incorrectly extracted, whereas 'Error' in the 'True' axis denotes real entities that were not extracted. We exclude replicated entities to simplify the computation of the confusion matrices.

In the entity extraction confusion matrix, there is significant confusion between the categories of 'disease' and 'rare\_disease', possibly due to overlapping textual features. The 'symptom\_and\_sign' category exhibits high classification accuracy but is also prone to being misclassified as 'Error', suggesting the need for more distinctive features or additional contextual information in the dataset. The 'Anaphor' category was accurately classified with fewer errors, indicating that the system effectively captures its linguistic features. However, many predicted entities are incorrectly extracted and are labeled as 'Error', indicating that LLMs tend to extract more information with a low precision.

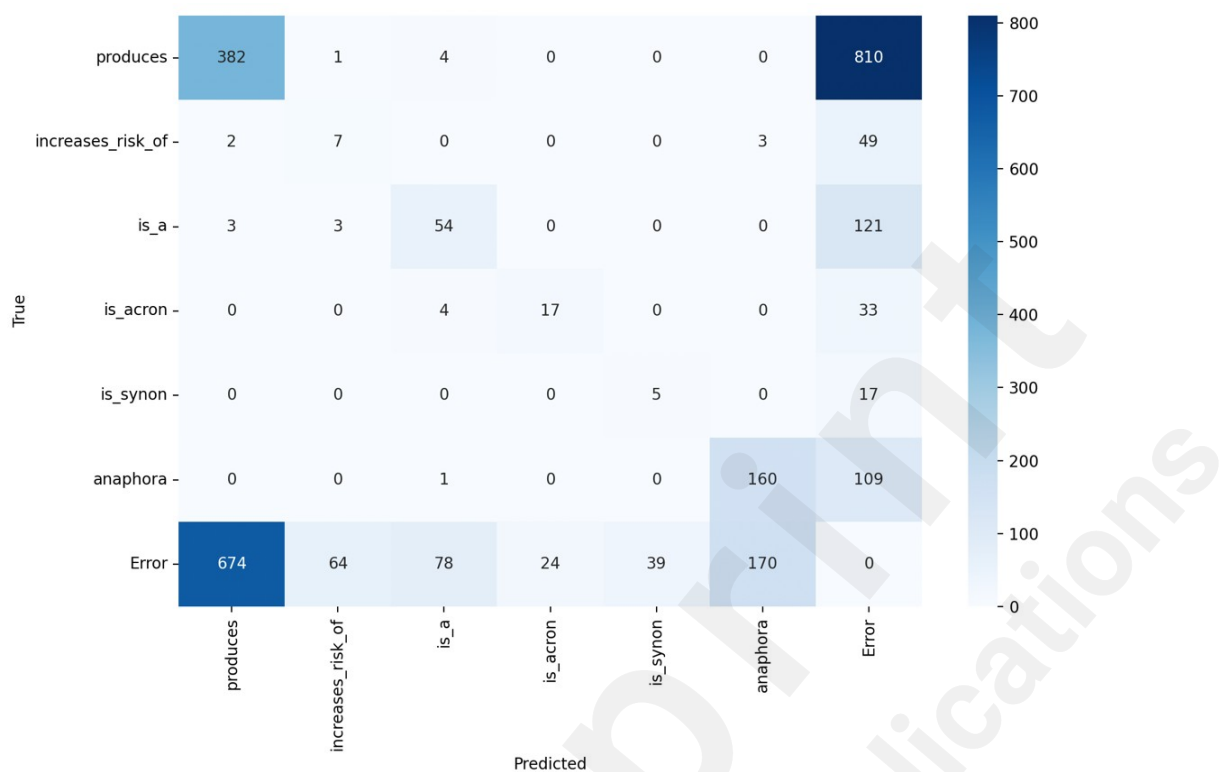
Figure 3. The confusion matrix of the entity extraction task results in RareDis2023.



The confusion matrix for the relation extraction indicates varying degrees of performance across different categories. The 'produces' relation is often identified correctly but also often misclassified as 'Error', indicating recognition issues. 'Increases\_risk\_of' is more frequently an 'Error' than correct, demonstrating the recognition difficulty of this relation. 'Is\_a' has moderate success but high error rates as well. 'Is\_acron' and 'is\_synon' rarely hit true positives and mostly fall into 'Error', possibly due to acronym variability and synonym recognition failure. 'Anaphora' resolution is relatively accurate but also misclassified, hinting at context comprehension challenges. The 'Error' category's high rate of both true and false positives is primarily affected by the false results from

entity extraction step before it.

Figure 4. The confusion matrix of the relation extraction task results in RareDis2023.

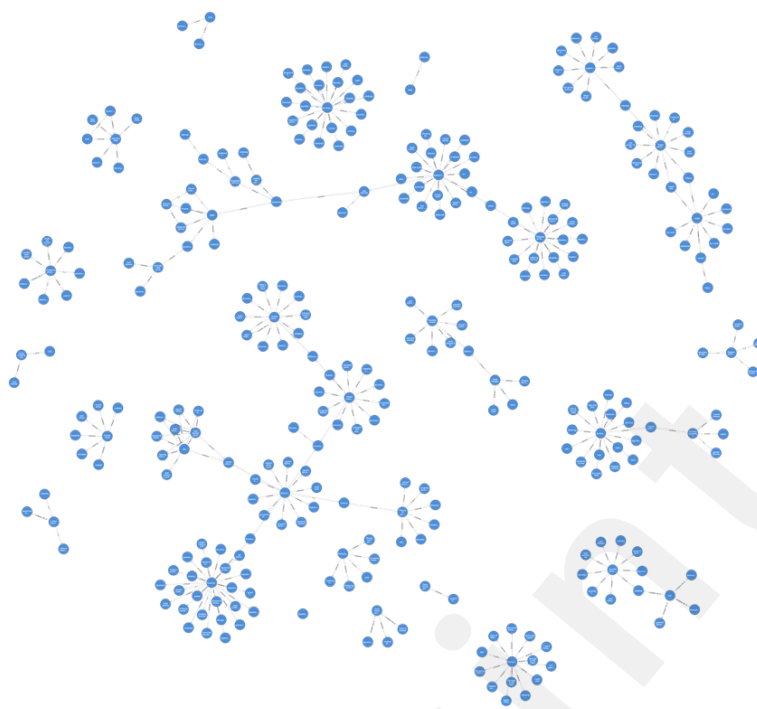


Qualitative Results

Qualitative results are showcased in (Figure 5), which depicts all the extracted results from the RareDis2023 dataset. Our qualitative results have been validated by medical experts and have shown satisfactory outcomes. This visualization provides a global perspective, highlighting the relationships among various rare diseases and their associated signs and symptoms in a concise knowledge graph.

Figure 5. The example of constructed knowledge graph from RareDis2023. The result is a clear and well-structured knowledge graph.





Specific extraction results of the knowledge graph are depicted in (Figure 6). This figure offers visualizations from a local perspective, illustrating an ideal structure of the knowledge graph. In this structure, rare diseases are positioned at the center of radial formations, with connections extending to entities like symptoms and signs. For example, in , the rare disease 'Turcot syndrome' is associated with 'abdominal pain', 'bleeding', 'fatigue', etc.

Figure 6. An example provides a detailed view of a specific local section of the constructed knowledge graph.



Additionally, we experimented with training specialized medical LLMs and compared their

performance. Specifically, we utilized Camel-Platypus2-70B [31], a healthcare-tailored model that is an extension of Llama-2 [32], through continuous training. Our experiments revealed that, without specific training, this type of model struggles to execute complex tasks such as joint entity and relation extraction. It appears that the inherent medical knowledge is not readily applicable in these scenarios.

## Discussion

### Principal Results

Our experimentation demonstrates the effectiveness of our proposed system, AutoRD. It significantly improves upon the base LLM and even outperforms fine-tuning models without requiring any training. Within several designs, the incorporation of medical ontologies has notably enhanced the LLMs by addressing gaps in medical knowledge. Furthermore, the results achieved in knowledge graph construction by our system are commendable. We highlight the advantage of LLMs in low-resource scenarios like rare disease extraction, showcasing their vast potential. Our meticulously designed system, AutoRD, substantiates this claim. The emergence of LLMs is generating unparalleled opportunities in the phenotyping of rare diseases. These models facilitate the automatic identification and extraction of concepts related to these diseases. Our prompts are easily adjustable due to their clear structure, allowing for simple modifications. Additionally, medical knowledge derived from external sources can be updated at any time within the AutoRD system.

### Limitations

Nevertheless, there is considerable potential for further improvement with respect to AutoRD. For instance, integrating advanced text processing tools and specialized medical tools into our system could amplify its capabilities. In the future, we can deploy more powerful medical LLMs as base models to enhance medical understanding. Moreover, medical experts can contribute more tailored prompts to improve LLMs' performance.

Our work has potential limitations and avenues for extension. For example, we have only evaluated AutoRD on a single dataset, so the results may not fully reflect the system's performance across the entire spectrum of rare diseases or in other long-text scenarios. Additionally, the prompts we designed are intuitive, but there is still room for continuous tuning and experimentation of different prompts. We acknowledge that AutoRD may not be the optimal LLM application for this task, yet it significantly improves upon the baseline performance of LLMs. This work aims at demonstrating the potential of LLM applications in the healthcare field.

## Conclusions

We developed AutoRD as an automatic and end-to-end system specifically designed for extracting information about rare diseases from text and building knowledge graphs. This system incorporates various innovative designs, including ontology-enhanced LLMs, to augment its medical knowledge base. The effectiveness of AutoRD has been validated through experimental evaluations, highlighting the valuable applications of LLMs in healthcare. In the future, we intend to explore how to broaden the application of LLMs to other healthcare processes in low-resource environments, including the identification of potential rare diseases in patients. Furthermore, ongoing research is focused on improving the performance of LLMs with respect to informing and augmenting medical decision-making strategies.

## Acknowledgements

Lang Cao designed the main method, wrote codes, conducted experiments, and wrote the manuscript. Dr. Adam Cross and Professor Jimeng Sun participated in the design, analysis, and discussion of the experiment and helped to improve the method. Dr. Adam Cross and Professor Jimeng Sun also assisted with revisions to the manuscript.

## Conflicts of Interest

None declared.

## Abbreviations

AutoRD: Automatic Rare Disease Mining System

CoT: chain-of-thought

GPT: Generative Pre-trained Transformer

HOOM: HPO-ORDO Ontology Module

HPO: Orphanet Human Phenotype Ontology

LLM: Large Language Model

UMLS: Unified Medical Language

Mondo: Mondo Disease Ontology

NER: Named Entity Recognition

NLP: Natural Language Processing

ORDO: Orphanet Rare Disease Ontology

## Multimedia Appendix 1

N/A

## References

1. Marwaha S, Knowles JW, Ashley EA, et al. A guide for the diagnosis of rare and undiagnosed disease: beyond the exome. *Genome Med.* 2022 Feb 28;14(1):23. doi: 10.1186/s13073-022-01026-w
2. Boat TF, Field MJ, et al. Rare Diseases and Orphan Products: Accelerating Research and Development. Washington, DC: National Academies Press; 2011.
3. Nguengang Wakap S, Lambert DM, Olry A, et al. Estimating cumulative point prevalence of rare diseases: analysis of the Orphanet database. *Eur J Hum Genet.* 2020 Feb;28(2):165–173.
4. Haendel M, Vasilevsky N, Unni D, et al. How many rare diseases are there? *Nat Rev Drug Discov.* 2020 Feb;19(2):77-78. doi: 10.1038/d41573-019-00180-y
5. U.S. Government Accountability Office. Rare Diseases: Although Limited, Available Evidence Suggests Medical and Other Costs Can Be Substantial. Report # GAO-22-104235. 2021. Available from: <https://www.gao.gov/assets/gao-22-104235.pdf>.
6. Tisdale A, Cutillo CM, Nathan R, et al. The IDEaS initiative: pilot study to assess the impact of rare diseases on patients and healthcare systems. *Orphanet J Rare Dis.* 2021;16:1–18.
7. Ferreira CR. The burden of rare diseases. *Am J Med Genet A.* 2019 Jun;179(6):885-892. doi: 10.1002/ajmg.a.61124
8. Bodenreider O. The Unified Medical Language System (UMLS): integrating biomedical

- terminology. *Nucleic Acids Res.* 2004 Jan 1;32(Database issue):D267-70. doi: 10.1093/nar/gkh061.
9. Köhler S, Gargano M, Matentzoglou N, et al. The Human Phenotype Ontology in 2021. *Nucleic Acids Res.* 2021 Jan 8;49(D1):D1207-D1217. doi: 10.1093/nar/gkaa1043.
  10. Orphanet. [Internet]. Available from: <https://www.orpha.net/consor/cgi-bin/index.php>. Accessed November 21, 2023.
  11. Zhao WX, Zhou K, Li J, et al. A Survey of Large Language Models. 2023. arXiv:2303.18223 [cs.CL].
  12. OpenAI. GPT-4 Technical Report. 2023. arXiv:2303.08774 [cs.CL].
  13. Karabacak M, Margetis K. Embracing Large Language Models for Medical Applications: Opportunities and Challenges. *Cureus.* 2023 May 21;15(5):e39305. doi: 10.7759/cureus.39305.
  14. Kasai J, Kasai Y, Sakaguchi K, et al. Evaluating GPT-4 and ChatGPT on Japanese Medical Licensing Examinations. 2023. arXiv:2303.18027 [cs.CL].
  15. Nori H, King N, McKinney SM, et al. Capabilities of GPT-4 on Medical Challenge Problems. 2023. arXiv:2303.13375 [cs.CL].
  16. Kung TH, Cheatham M, Medenilla A, et al. Performance of ChatGPT on USMLE: Potential for AI-assisted medical education using large language models. *PLoS Digit Health.* 2023;2(2):e0000198.
  17. Li M, Zhang R. How far is Language Model from 100% Few-shot Named Entity Recognition in Medical Domain. 2023. arXiv:2307.00186 [cs.CL].
  18. Sanjak J, Binder J, Yadaw AS, Zhu Q, Mathé EA. Clustering rare diseases within an ontology-enriched knowledge graph. *J Am Med Inform Assoc.* 2023 Sep; [Epub ahead of print]. doi: 10.1093/jamia/ocad186.
  19. Alsentzer E, Li MM, Kobren SN, et al. Deep learning for diagnosing patients with rare genetic diseases. *medRxiv.* 2022 Dec. [Online]. Available from: <https://www.medrxiv.org/content/10.1101/2022.12.07.22283238>.
  20. Rashid R, Copelli S, Silverstein JC, Becich MJ. REDCap and the National Mesothelioma Virtual Bank—a scalable and sustainable model for rare disease biorepositories. *J Am Med Inform Assoc.* 2023 Jul;30(10):1634-1644. doi: 10.1093/jamia/ocad132.
  21. Datta S, Lee K, Paek H, et al. AutoCriteria: a generalizable clinical trial eligibility criteria extraction system powered by large language models. *J Am Med Inform Assoc.* 2023 Nov; [Epub ahead of print]. doi: 10.1093/jamia/ocad218.
  22. Shyr C, Hu Y, Harris PA, et al. Identifying and Extracting Rare Disease Phenotypes with Large Language Models. 2023. arXiv:2306.12656 [cs.CL].
  23. Oniani D, Hilsman J, Dong H, et al. Large Language Models Vote: Prompting for Rare Disease Identification. 2023. arXiv:2308.12890 [cs.CL].
  24. Vasilevsky NA, Matentzoglou NA, Toro S, et al. Mondo: Unifying diseases for the world, by the world. *medRxiv.* 2022 Apr 13. doi: 10.1101/2022.04.13.22273750.
  25. Martínez-deMiguel C, Segura-Bedmar I, Chacón-Solano E, et al. The RareDis Corpus: A Corpus Annotated with Rare Diseases, Their Signs and Symptoms. *J Biomed Inform.* 2022 Jan;125:C.
  26. Sinan I, Mihdawi M, Farahat AR, Fida M. Knowledge and Awareness of Rare Diseases Among Healthcare Professionals in the Kingdom of Bahrain. *Cureus.* 2023 Oct 25;15(10):e47676. doi: 10.7759/cureus.47676.
  27. Wei J, Wang X, Schuurmans D, et al. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. 2023. arXiv:2201.11903 [cs.CL].
  28. Dong Q, Li L, Dai D, et al. A Survey on In-context Learning. 2023. arXiv:2301.00234 [cs.CL].
  29. Neo4j. [Internet]. Available from: <http://neo4j.org>. Accessed November 27, 2023.

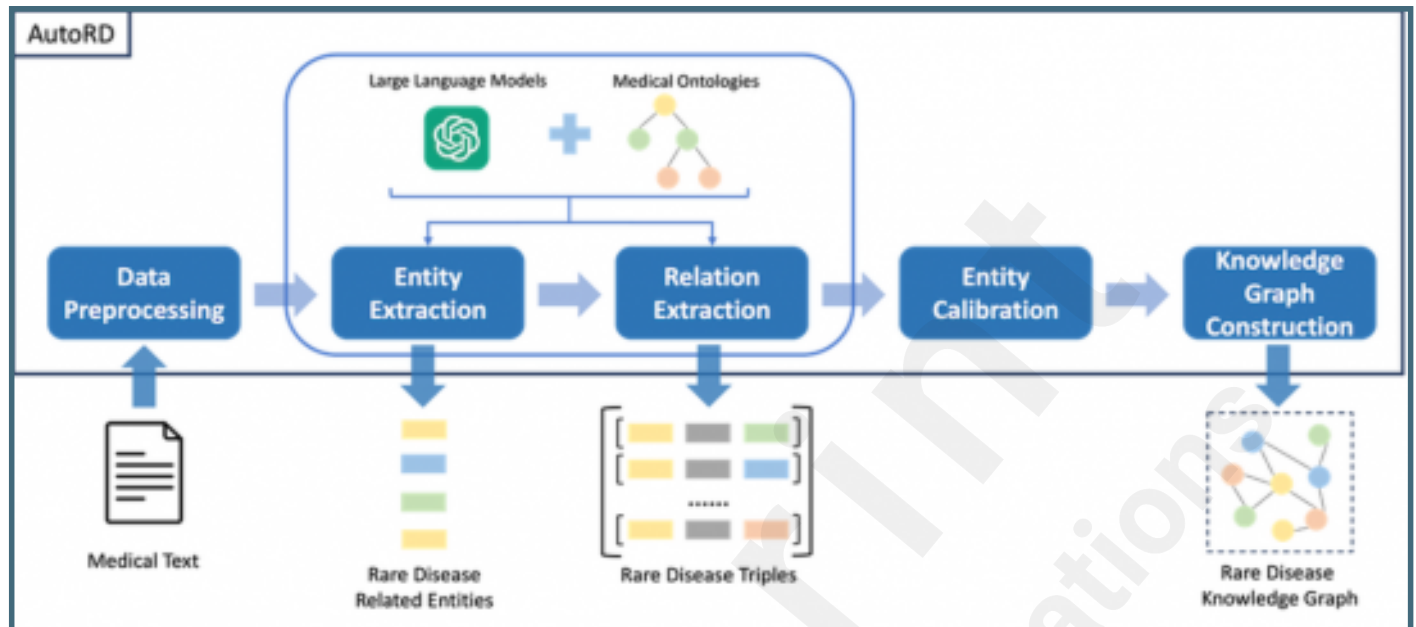
30. Alsentzer E, Murphy JR, Boag W, et al. Publicly Available Clinical BERT Embeddings. 2019. arXiv:1904.03323 [cs.CL].
31. Lee AN, Hunter CJ, Ruiz N. Platypus: Quick, Cheap, and Powerful Refinement of LLMs. 2023. arXiv:2308.07317.
32. Touvron H, Martin L, Stone K, et al. Llama 2: Open Foundation and Fine-Tuned Chat Models. 2023. arXiv:2307.09288.



## Supplementary Files

## Figures

The AutoRD framework processes medical texts as input data and outputs entities related to rare diseases and rare disease triples, which are the results of the extraction process. Subsequently, it constructs a knowledge graph based on these extraction results. During the entity and relation extraction steps, ontologies-enhanced large language models are utilized to enhance performance.

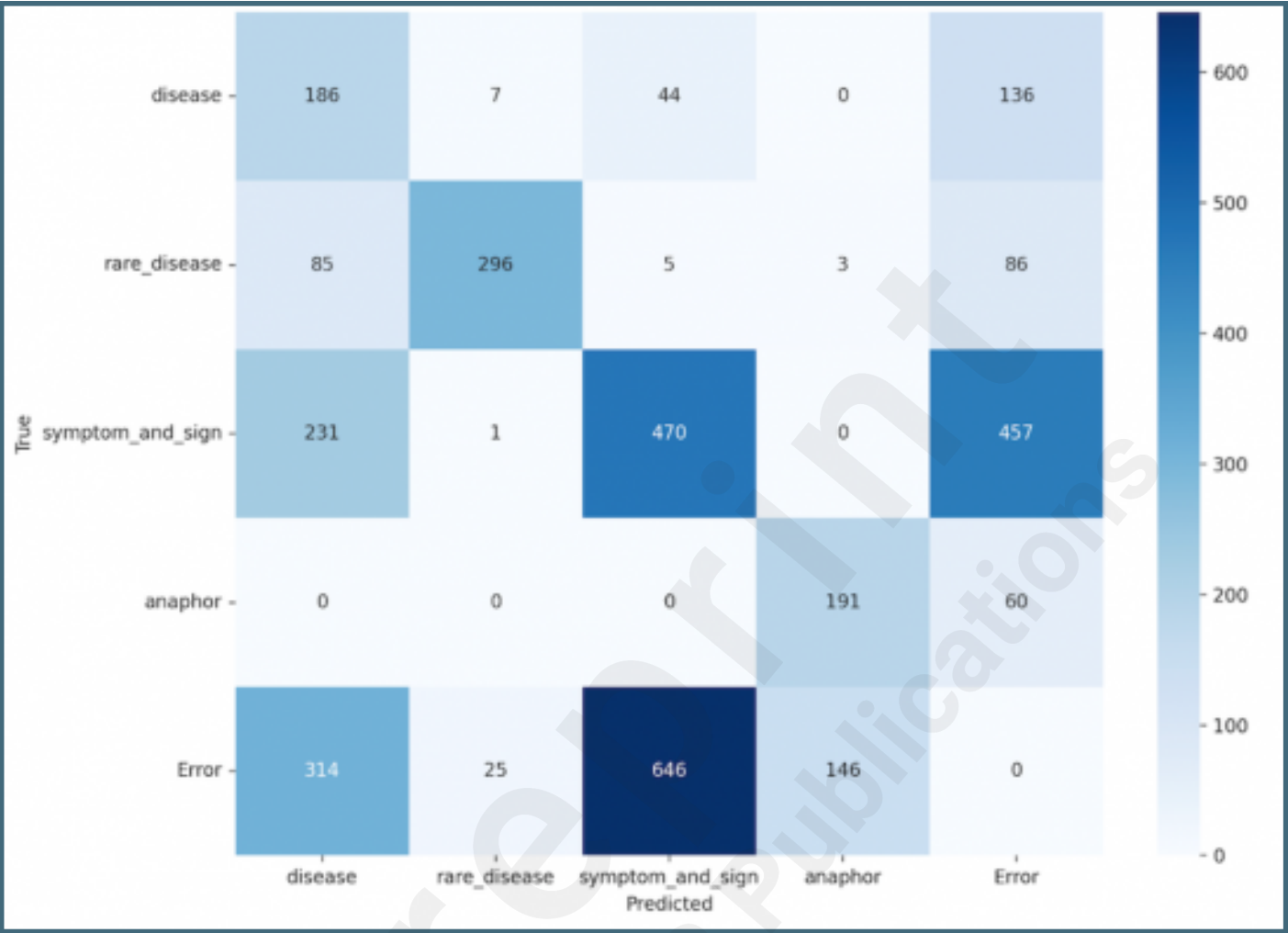




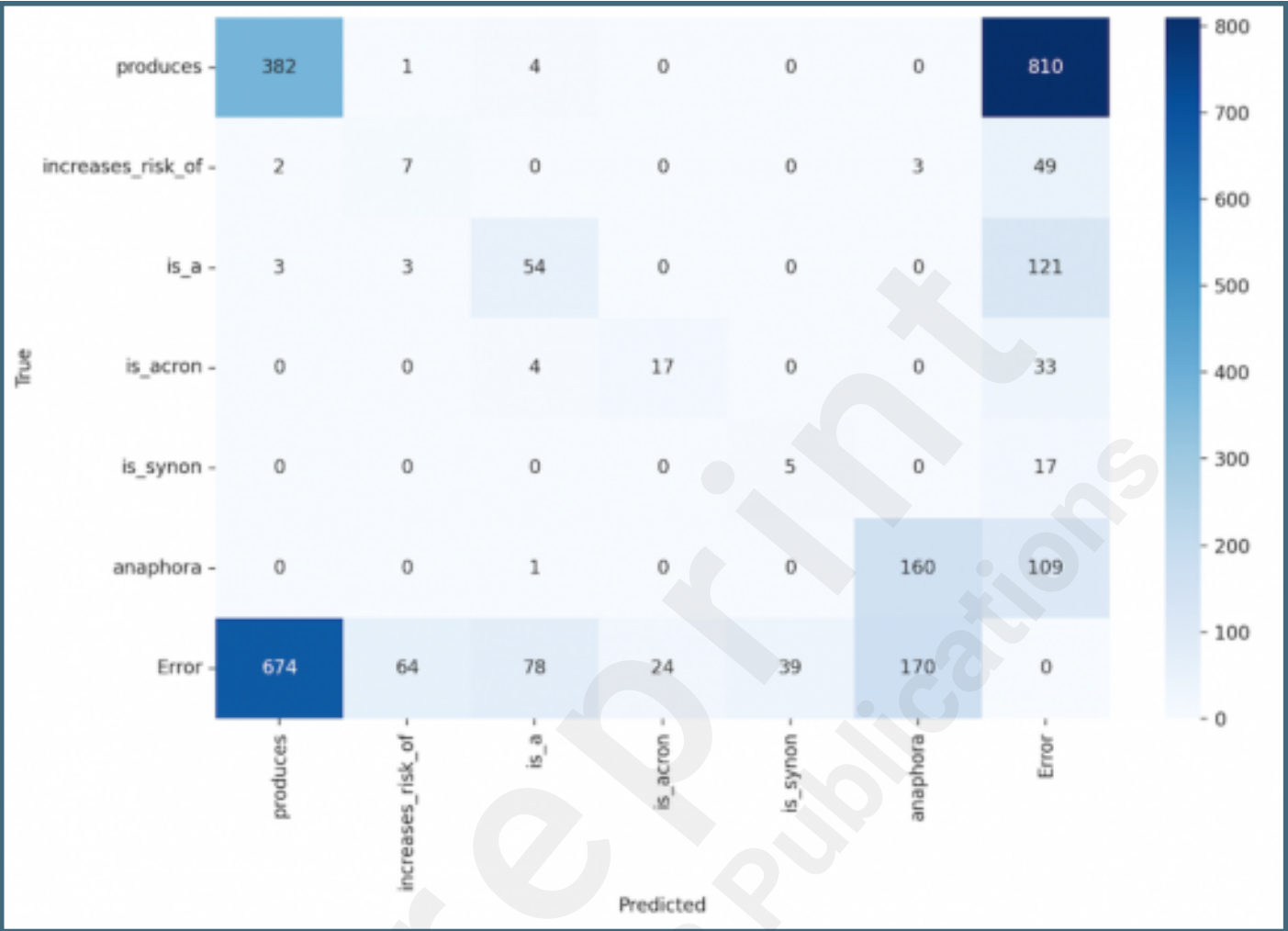
Content of all prompt templates in AutoRD. This figure presents the simplified content of all prompts to provide a clear framework of the prompt structure. The black text represents the original text of the instructions. Grey text indicates a summary of each part of the instructions. Blue text highlights the prompt slots, where external information and inputs can be inserted.

Extract More Terms	Extract Entities	Extract Relations	Calibrate Entities
<b>### Task:</b> (Extract entities: "anaphor", "medical_term"...) <b>## Definition:</b> <b>## Entity:</b> (Define entities, including "rare_disease", "disease", "symptom_and_sign", "anaphor", "medical_term".) <b>## Notice:</b> (Remind the model of some considerations...) (Provide extracted "medical_terms".) <b>## Input Text:</b> (Provide text.) <b>## Output format</b> <b>## Entity Representation:</b> (Define entity representation.) <b>## Your output should be a single JSON object in the following format:</b> (Define the format of response.) <b>## Output:</b>	<b>### Task:</b> (Extract entities: "rare_disease", "disease", "symptom_and_sign", "anaphor"...) <b>## Definition:</b> <b>## Entity:</b> (Define entities, including "rare_disease", "disease", "symptom_and_sign", "anaphor".) <b>## Notice:</b> (Remind the model of some considerations...) <b>## Examples</b> Here are some examples of input text and target entities: <b>## Input (Give Passage):</b> {text} <b>## Extracted Medical Terms:</b> {medical_terms} <b>## Extracted Anaphor:</b> {anaphor} <b>## Rare Disease Knowledge:</b> {rare_disease_knowledge} <b>## Output format</b> <b>## Entity Representation:</b> (Define entity representation.) <b>## Your output should be a single JSON object in the following format:</b> (Define the format of response.) <b>## Output:</b>	<b>### Task:</b> (Extract relations: "produces", "increases_risk_of", "is_a", "is_acron", "is_synon", "anaphora"...) <b>## Definition:</b> <b>## Entity:</b> (Define entities, including "rare_disease", "disease", "symptom_and_sign", "anaphor".) <b>## Relation:</b> (Define relations, including "produces", "increases_risk_of", "is_a", "is_acron", "is_synon", "anaphora".) <b>## Notice:</b> (Remind the model of some considerations...) <b>## Input (Give Passage):</b> {text} <b>## Examples</b> Here are some examples of input text and target entities: <b>## Input (Give Passage):</b> {text} <b>## Extracted Entities:</b> {entities} <b>## Rare Disease Knowledge:</b> {rare_disease_knowledge} <b>## Output format</b> <b>## Entity Representation:</b> (Define entity representation.) <b>## Relation Representation:</b> (Define relation representation.) <b>## Your output should be a single JSON object in the following format:</b> (Define the format of response.) <b>## Output:</b>	<b>### Task:</b> (Verify all extracted entities and relations...) <b>## Definition:</b> <b>## Entity:</b> (Define entities, including "rare_disease", "disease", "symptom_and_sign", "anaphor".) <b>## Relation:</b> (Define relations, including "produces", "increases_risk_of", "is_a", "is_acron", "is_synon", "anaphora".) <b>## Notice:</b> (Remind the model of some considerations...) <b>## Input (Give Passage):</b> {text} <b>## Extracted Entities:</b> {entities} <b>## Extracted Relations:</b> {relations} <b>## Output format</b> <b>## Entity Representation:</b> (Define entity representation.) <b>## Relation Representation:</b> (Define relation representation.) <b>## Your output should be a single JSON object in the following format:</b> (Define the format of response.) <b>## Output:</b>

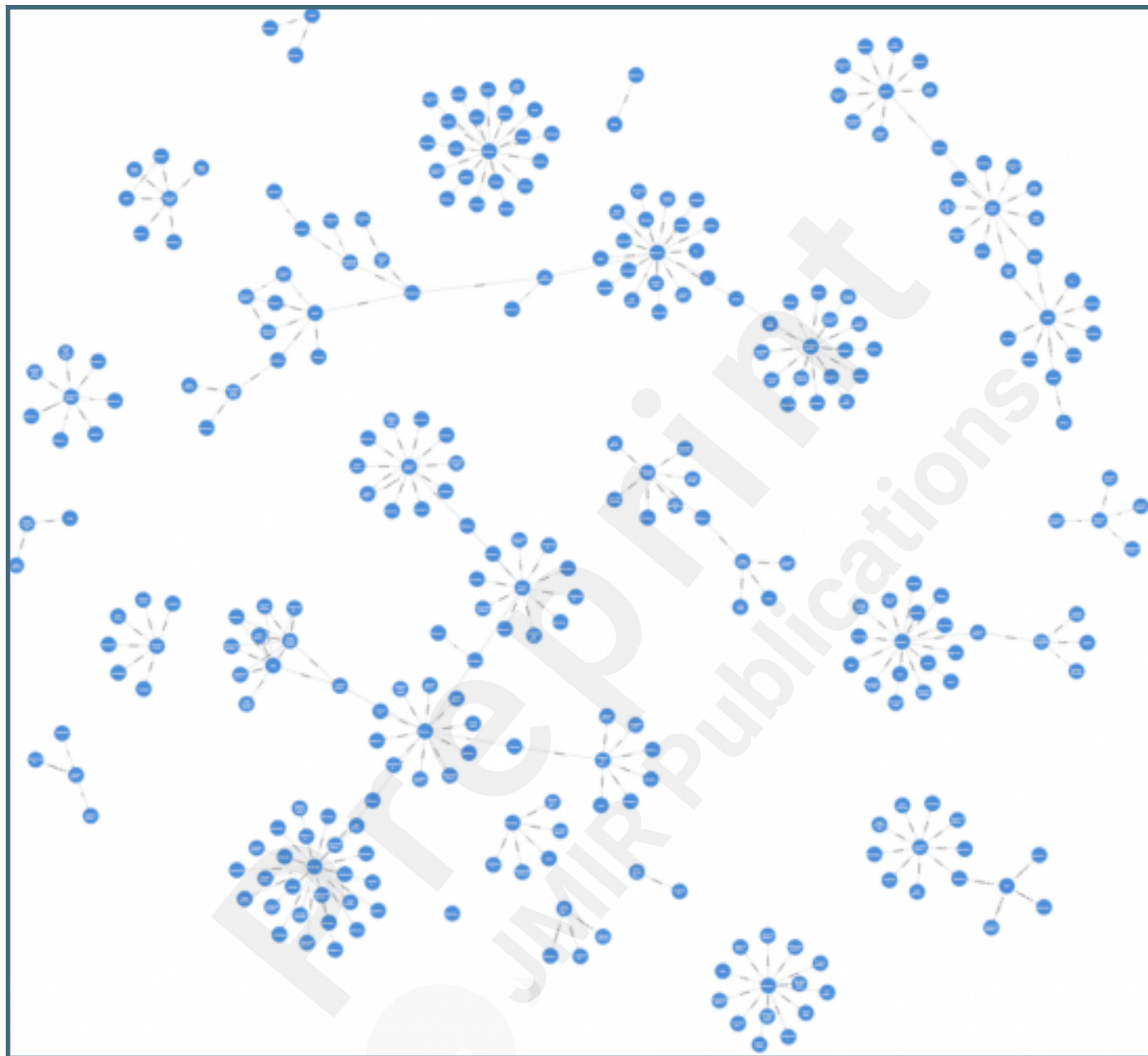
The confusion matrix of the entity extraction task results in RareDis2023.



The confusion matrix of the relation extraction task results in RareDis2023.



The example of constructed knowledge graph from RareDis2023. The result is a clear and well-structured knowledge graph.



This example provides a detailed view of a specific local section of the constructed knowledge graph.

