electronics

MDPI

Article

Traditional Chinese Medicine Knowledge Graph Construction Based on Large Language Models

Yichong Zhang and Yongtao Hao *

CAD Research Center, Tongji University, Shanghai 200092, China; yichongzyc@163.com
* Correspondence: hyongtao492@163.com

Abstract: This study explores the use of large language models in constructing a knowledge graph for Traditional Chinese Medicine (TCM) to improve the representation, storage, and application of TCM knowledge. The knowledge graph, based on a graph structure, effectively organizes entities, attributes, and relationships within the TCM domain. By leveraging large language models, we collected and embedded substantial TCM–related data, generating precise representations transformed into a knowledge graph format. Experimental evaluations confirmed the accuracy and effectiveness of the constructed graph, extracting various entities and their relationships, providing a solid foundation for TCM learning, research, and application. The knowledge graph has significant potential in TCM, aiding in teaching, disease diagnosis, treatment decisions, and contributing to TCM modernization. In conclusion, this paper utilizes large language models to construct a knowledge graph for TCM, offering a vital foundation for knowledge representation and application in the field, with potential for future expansion and refinement.

Keywords: traditional Chinese medicine; large language modeling; knowledge graph; interdisciplinary research

check for updates

Citation: Zhang, Y.; Hao, Y. Traditional Chinese Medicine Knowledge Graph Construction Based on Large Language Models. *Electronics* **2024**, *13*, 1395. https://doi.org/10.3390/electronics13071395

Academic Editor: Arkaitz Zubiaga

Received: 17 February 2024
Revised: 29 March 2024
Accepted: 3 April 2024
Published: 7 April 2024

1. Introduction

Traditional Chinese Medicine (TCM) embodies the unique wisdom of the Chinese nation regarding life, health, and medical treatment [1]. With rich theoretical knowledge and clinical expertise, TCM holds significant academic and practical value. As an integral part of Traditional Chinese Medicine, TCM has accumulated centuries of abundant experience and knowledge. However, challenges persist in the modernization and intelligent application of TCM, hindering the effective utilization of knowledge and information within the field [2].

With the development of computer technology and related theories, the reorganization and utilization of TCM knowledge information through advanced modern technology have gained recognition, leading to notable achievements in relevant research. The integration of advanced ontological theories and techniques from the field of computer science into the study of TCM knowledge organization, constructing a Chinese medicine ontology, and achieving the knowledge–based restructuring of Chinese medicine information can provide a foundational data structure for data mining and knowledge discovery in the field of TCM [3].

In the era of big data, Knowledge Graphs (KGs) serve as crucial data resources for knowledge management and applications, playing a key role in various fields such as semantic retrieval, knowledge inference, decision–making, question–answering, and system recommendations. In 2012, Google introduced the concept of the KG and applied it to search engines [4]. Since then, the KG has been widely employed in various domains. A Chinese Medicine KG is a structured knowledge base modeling and representing concepts, entities, and relationships in the field of TCM. It aids doctors, researchers, and patients in better understanding and utilizing knowledge in TCM.

The traditional process of constructing knowledge graphs often relies on extensive manual operations and expert knowledge, which can lead to inefficiency and errors when dealing with massive data and complex relationships [5]. Knowledge graph construction typically involves various methods such as manual construction, automatic construction, and semi–automatic construction. Manual construction involves domain experts manually inputting entities, attributes, and relationships, but this method is time–consuming and labor–intensive, with limited applicability. Automatic construction utilizes information extraction techniques to extract knowledge from structured and unstructured data, but may face challenges in terms of accuracy and completeness. Semi–automatic construction combines manual and automatic approaches, guiding experts through construction using assisting tools or algorithms and achieving certain effectiveness.

Currently, with the development of large language models (LLMs), their application in knowledge graph construction has gradually become a research hotspot. LLMs possess outstanding representation learning capabilities, extracting rich semantic information from text through learning from extensive corpora. Introducing LLMs into the knowledge graph construction process allows for the automated extraction of entities, attributes, and relationships from text, significantly reducing the manual annotation workload, improving construction efficiency, and ensuring accuracy [6]. Furthermore, LLMs can handle multimodal data, such as text and images, providing support for the richness and diversity of knowledge graphs.

The current developmental status indicates that LLMs are emerging as powerful tools for knowledge graph construction. This is attributed not only to their capability to handle and analyze extensive unstructured textual data but also to their adaptability to specific domains through pre–training and fine–tuning [7]. This adaptability ensures both the quality of construction and a significant enhancement in the automation and efficiency of the construction process.

The objective of constructing a knowledge graph for TCM is to structurally represent and link entities, relationships, and attributes related to TCM, forming a comprehensive and accurate network of TCM knowledge. Such a knowledge graph can assist healthcare professionals in disease differentiation and treatment, support clinical decision–making, and provide rich data for TCM research. Furthermore, a TCM knowledge graph facilitates the integration of TCM with modern medicine, opening new possibilities for interdisciplinary medical research and applications.

Despite several studies focusing on the construction of TCM knowledge graphs, the field still faces numerous challenges. The primary challenges include the complexity, diversity, and ambiguity of TCM knowledge, which increase the difficulty of accurately characterizing and correlating various types of knowledge. Additionally, given the vast and decentralized nature of the TCM knowledge system, effectively collecting, integrating, and storing relevant knowledge poses a challenging task. Furthermore, ensuring the timeliness and updateability of the knowledge graph is crucial as TCM knowledge continues to evolve and develop [8].

The literature [9–12] commonly employs natural language processing and machine learning techniques for the automated construction of knowledge graphs. However, these studies still face challenges in accurately identifying entities, extracting relationships, and achieving comprehensive coverage of domain knowledge. They heavily rely on significant manual intervention to rectify errors and enhance data quality, leading to an increase in human resource costs. Addressing these limitations, this study aims to enhance the accuracy of the automated extraction process and reduce the dependency on human resources by adopting advanced LLMs and fine–tuning them with domain expert knowledge. This approach is expected to automatically extract high–quality knowledge structures from extensive Chinese medicine text data and effectively transform them into a format suitable for knowledge graph construction, thereby promoting the automation and intelligence of TCM knowledge graph development.

The main contributions of this paper include the following:

(1) Adopting LLMs for named entity recognition, utilizing few–shot learning techniques to achieve high accuracy in identification, significantly reducing the cost of manual annotation, and laying a solid foundation for the construction of an accurate Chinese medicine knowledge graph.
(2) Constructing a knowledge graph of TCM, which not only contributes to the preservation and dissemination of TCM knowledge but also facilitates the integration of traditional knowledge with modern technology, opening up new possibilities for the innovation and development of TCM.
(3) In the experimental evaluation phase, this study systematically validates and assesses the proposed named entity recognition method using real–world Chinese medicine domain text data. Experimental results demonstrate a significant improvement in the efficiency and accuracy of the knowledge extraction process.

2. Related Work

2.1. Construction of Traditional Chinese Medicine Knowledge Graph

A knowledge graph is a semantic network that maps the real world onto the data world, composed of nodes and edges. It can be understood as a semantic network consisting of numerous knowledge points and their interconnecting relationships. Alternatively, it can be simplified as a "multi–relational graph," encompassing various types of nodes and edges. In a knowledge graph, entities, representing real–world entities, are used to denote nodes, and relationships are employed to express edges, indicating certain connections between different entities. Entities and relationships typically possess their respective attributes.

In 2012, Google first introduced the concept of the knowledge graph and applied it to its search engine. Since the inception of the knowledge graph concept, numerous researchers have undertaken substantial efforts to construct large–scale, high–quality knowledge graphs. Knowledge graphs can be broadly categorized into general knowledge graphs and domain–specific knowledge graphs. In the realm of research, there exist various general knowledge graphs and extensive public knowledge repositories. This paper focuses on a domain–specific knowledge graph, specifically addressing TCM knowledge.

In the field of TCM knowledge, numerous researchers have already constructed relevant knowledge graphs. Cheng et al. developed a medical knowledge graph for stroke [4]; Wang et al. designed a knowledge graph–based monitoring system for Traditional Chinese Medicine prescriptions [10]; Yang et al. built a TCM knowledge graph based on Chinese classical texts [11]; Zheng et al. created a deep learning–based TCM knowledge graph platform, TCMKG [12]. While these knowledge graphs have significantly contributed to applied research in this domain, most of them are manually constructed by domain experts or rely heavily on unstructured data during integration, resulting in lower accuracy. The data sources for the knowledge graph constructed in this paper come from various online sources, including structured, semi–structured, and unstructured data. Compared to existing knowledge repositories, our graph has a simpler structure, facilitating easier extraction and achieving higher accuracy. Given the limited availability of TCM data for training, which can lead to reduced accuracy, this paper utilizes few–shot LLMs to construct the database, reducing the annotation workload significantly and minimizing human labor.

2.2. Named Entity Recognition

Named entity recognition (NER) refers to the identification and extraction of entities with specific meanings from a given text, typically involving the tasks of determining entity boundaries and determining entity categories or attributes [13].

In the field of NER, there are mainly four approaches: rule–based methods, statistical–model–based methods, neural network–based methods, and pre–trained–model–based methods.

Rule–based methods utilize predefined rules and patterns to identify entities. For instance, regular expressions can be employed to match strings with specific patterns as

entities. The advantage of this approach lies in its simplicity and intuitiveness, but it requires manual rule crafting and may be challenging to cover all possible cases.

Statistical–model–based methods employ machine learning algorithms such as Conditional Random Fields (CRFs) [14] and Hidden Markov Models (HMMs) [15] for named entity recognition. Statistical models identify entities by learning the mapping relationship from input text to output labels (entity categories). This approach takes into account contextual information and relationships between features but necessitates a substantial amount of annotated data for model training.

With the rise of deep learning, neural network–based methods have made significant advancements in NER. Models based on Recurrent Neural Networks (RNNs) and Long Short–Term Memory (LSTM) [16] networks are widely applied. These models can capture context information and sequence relationships, thereby enhancing the accuracy of named entity recognition.

2.3. Large Language Models

Recently, the emergence of pre–trained language models has further advanced the development of NER. Through pre–training on large–scale text corpora, these models can learn rich language representations, including entity information. By fine–tuning specific tasks, pre–trained models can achieve outstanding NER performance. One widely used pre–trained language model based on the Transformer architecture is BERT (Bidirectional Encoder Representations from Transformers) [17]. Building upon BERT, several pre–trained models tailored for the biomedical domain have been developed to perform NER tasks. For instance, BioBERT is based on BERT and further pre–trained on biomedical data, followed by fine–tuning medical text data for NER [18]. ClinicalBERT is another model pre–trained on clinical medical text data and fine–tuned for tasks like medical entity recognition [19]. MedBERT, a domain–specific pre–trained model for the medical field, is based on the Transformer architecture and pre–trained on medical literature and clinical data. MedBERT demonstrates strong performance in medical entity recognition tasks, identifying entities such as diseases, drugs, and treatment methods [20]. These models have found extensive applications in building knowledge graphs in the medical domain.

GPT (Generative Pre–trained Transformer), as a pre–trained language model, can be fine–tuned or applied to specific domains or tasks to obtain large language models (LLMs) that better understand and generate text in those domains, providing more accurate predictions and generation results. GPT has demonstrated impressive performance in various NLP tasks [21]. In November 2022, OpenAI introduced ChatGPT [22] as an extension of GPT–3, one of the state–of–the–art NLP models at that time. ChatGPT has exhibited excellent performance across various NLP tasks.

With the growing interest in ChatGPT among the general public, experts from various fields are exploring its application in their respective domains, aiming to reduce human labor consumption, and the field of medicine is no exception. According to surveys, ChatGPT achieved an accuracy rate of approximately 60% in the United States Medical Licensing Examination [20]. Ni et al. [23] investigated the ability of LLMs to comprehend instructions and perform text–structured tasks. They proposed adding a prefix and suffix instruction before inputting text into the LLM to indicate the required information extraction (IE) task. Through testing on different datasets, it was demonstrated that a simple instruction could enable the LLM to perform comparably to other state–of–the–art methods on the dataset. However, despite the promising performance of LLMs on various natural language processing tasks, their performance on named entity recognition (NER) remains lower than supervised baselines. Even with the addition of instructions, the accuracy of the task cannot be guaranteed. This is due to the nature of NER as a sequence labeling task, while LLM is primarily a text generation model. In this study, we address this gap by transforming the sequence labeling task into a generative task that LLMs can easily adapt to [24]. Specifically, through few–shot prompts, we guide LLMs to annotate relevant entities related to TCM in sentences using the special symbol “【】”.

For large–scale knowledge extraction tasks, it is necessary to rely on the API (Application Programming Interface) of LLMs. The iFLYTEK Spark Cognitive Large Model is a Chinese natural language processing full–stack platform introduced by iFLYTEK, a Chinese tech giant. It is currently the largest Chinese pre–trained language model in the world, with over 1000 billion parameters, covering more than 1000 billion characters of Chinese text data. It possesses powerful general language representation capabilities, performing better than or close to human level on multiple public datasets; it has a rich Chinese knowledge base, significantly outperforming other models on Chinese–question–answering datasets; it has flexible generation capabilities, able to generate various types and styles of Chinese texts according to user needs and preferences, scoring higher than other models on Chinese generation datasets; it also has an open platform and interface.

In the field of Chinese natural language processing, the iFLYTEK Spark Cognitive Large Model has demonstrated superior performance compared to ChatGPT. Specifically, according to the experimental results in Section 4, this model has shown better application performance in the field of TCM, surpassing the effects of ChatGPT. Moreover, unlike ChatGPT, which requires a fee to access its API, the iFLYTEK Spark Cognitive Large Model offers a free API service, significantly reducing the financial burden of research tasks. In view of the above factors, this study ultimately decided to employ the iFLYTEK Spark Cognitive Large Model to carry out various research tasks.

2.4. Few–Shot Prompting

The burgeoning paradigm of few–shot prompting has increasingly been recognized as a viable strategy to circumvent the dependence on voluminous task–specific training datasets. Distinguishing itself from conventional few–shot learning techniques, which necessitate the fine–tuning of models with scarce supervision [25,26], few–shot prompting is a technique applied to support the model through input and output examples. The technique does not require large amounts of training data and the model uses pre–given training to give the desired answer when prompted for output.

Comprehensive studies have corroborated the adeptness of few–shot prompting across a spectrum of disciplines. Notably, within the ambit of natural language processing (NLP), empirical findings have substantiated that the incorporation of elementary yet task–pertinent prompts within input sequences significantly augments the performance of pre–trained language models on designated tasks, obviating the need for additional fine–tuning [21]. This approach has demonstrated its superiority particularly in scenarios where the procurement of numerous annotated examples is infeasible.

In essence, few–shot prompting presents an innovative and pragmatic solution to the data intensiveness traditionally associated with the training of machine learning models. As research in this area progresses, the potential applications of few–shot prompting are being extended, holding promise for enhanced efficiency and adaptability in a multitude of NLP tasks and beyond.

3. Algorithm Implementation

In this section, we introduce the construction process of the TCM knowledge graph, which is illustrated in Figure 1. The process consists of knowledge acquisition, knowledge extraction, knowledge fusion, and data storage.

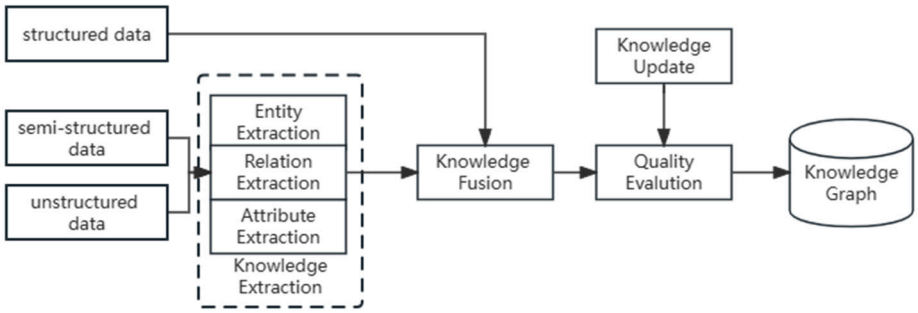

Figure 1. Framework for constructing TCM knowledge graph. The application of LLMs in entity extraction and relation extraction tasks is among them.

3.1. Data Collection and Data Cleaning

Data resources can be categorized into three types based on their structural characteristics: structured data, referring to data conforming to a fixed format and commonly stored in relational databases; semi–structured data, such as tables and lists in web pages, which possess a certain structure but lack standardization; unstructured data, like web page text, which lacks predefined organizational formats.

In the process of constructing the TCM knowledge graph, the main data sources include semi–structured and unstructured data. Specifically, semi–structured data are primarily obtained from Baidu Baike (Baidu Baike_Global Leading Chinese Encyclopedia (www.baidu.com (16 April 2023))) and Zhongyi Baodian ([Zhongyi Baodian] Complete Collection of Famous Traditional Chinese Medicine Books, Online Reading of Traditional Chinese Medicine Books, Online Learning of Traditional Chinese Medicine (www.zhongyibaodian.com (16 April 2023))); unstructured data are mainly sourced from web pages related to TCM and TCM on the internet, such as Zhongyi Zhongyao Wang (TCM Online_Traditional Chinese Medicine Network (www.zhzyw.com (16 April 2023))). We utilize thematic web crawling techniques to obtain TCM–related text data from specified websites and save them in .txt format.

The extraction process for semi–structured data is as follows: The crawler technology crawls the semi–structured data in the information box (InfoBox) in the Baidu Encyclopedia. Figure 2 shows the InfoBox in Baidu Encyclopedia using "Gardenia" as an example. Starting with the initial page provided, we crawl clickable web pages in a manner similar to breadth–first search. We save the obtained page information as an HTML (Hyper Text Markup Language) file in web format. Subsequently, use the Xpath selector to extract the contents of the InfoBox in the saved web page. We organize the data, including the basicInfo–item name and basicInfo–item value, into an Excel file according to their correspondence. These semi–structured data will be stored in the database as attributes of the entity and together with the entity form a property triplet, such as $< entity, attribute\ name, attribute\ value >$.

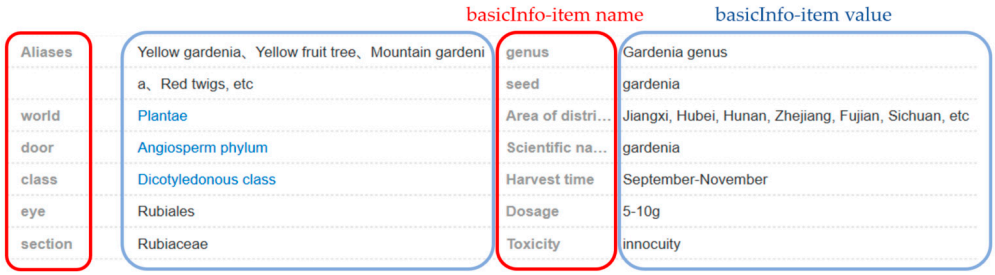

Figure 2. Illustration of an example of semi–structured data extraction from Baidu Baike. This section pertains to the InfoBox section, where the red–boxed area represents basicInfo–item name, and the data within the blue boxes are basicInfo–item values. This section primarily serves the purpose of constructing property triplets.

For unstructured textual data, we view the web page format and use generalized web crawling techniques to access data information of all pages after that by using the breadth–first search strategy, starting from a preset URL (Uniform Resource Locator) and continuously extracting new URLs from the URL queue. This information is then saved as a text file for subsequent knowledge extraction operations.

To ensure that the data obtained met high quality standards, a meticulous data cleaning process was conducted. In this study, data cleaning was carried out according to a series of strict filtering criteria, which mainly consisted of identifying and removing outliers. Outliers in this context mainly refer to possible HTML tags, URL links, special characters, garbled or inconsistently encoded data, and recurring data. The purpose of this data cleaning process is to provide carefully filtered input data for the subsequent knowledge extraction phase, thereby improving the overall data quality.

3.2. Prompt Construction

In this study, we adopt the methodology of few–shot prompting to perform knowledge extraction tasks. The task is divided into three coherent stages: (1) task description: this stage involves the precise articulation and definition of the knowledge to be extracted, informing large language models (LLMs) of the nature of the task at hand; (2) few–shot demonstration: in this phase, by presenting a limited number of examples as guidance, the model is directed to understand the expected output structure and content, with different examples provided according to the task's specificity; (3) input sentence: the final stage requires the model to generate a structured knowledge output based on the provided input sentences, utilizing the information and framework acquired in the previous two stages. Through this staged approach, we aim to utilize few–shot prompting techniques to achieve effective knowledge extraction using LLMs.

3.2.1. Named Entity Recognition

The prompt structure for named entity recognition comprises three integral components, as delineated below. An exemplification of prompt instances is provided in Figure 3.

(1) Task description.

"Your task is to use brackets "【】" to select the [*entity type*] entities in the given sentences. Here are some examples:"

In the task of entity recognition, LLMs are required to identify and bracket the entities specified by [*entity type*] from the provided statements. The [*entity type*] in this instruction indicates the description of the desired information category. This study involves multiple types of entities, implying that the extraction task needs to be completed through multiple iterations. This process essentially transforms a multi–class classification problem into several binary classification problems.

(2) Few–shot demonstration.

To ensure uniformity in the output format of the Large Language Model (LLM) and facilitate subsequent processing, three examples are provided to standardize the input and output forms. The three examples cover three scenarios: (a) the sentence does not contain any specific type of entity; (b) the sentence contains only one specific type of entity; (c) the sentence contains multiple specific types of entities. Each example consists of an input sequence X and the corresponding output sequence Y.

Regarding the output format, if the input sequence X does not contain the target entity, the output sequence Y should directly copy the input sequence X. Conversely, if the input sequence X contains one or more target entities, the special brackets "【" and "】" should be used to annotate the entities.

(3) Input sentence.

你的任务是用“【】”框选出给定句子中的疾病实体。以下是一些例子：
Your task is to use brackets "【】" to select the disease entities in the given sentences. Here are some examples:

Input： 齿龈出血称为齿衄，又称为牙衄、牙宣。
Input: Bleeding from the gums is referred to as gum bleeding, and is also known as tooth bleeding or gum oozing.
Output： 齿龈出血称为【齿衄】，又称为【牙衄】、【牙宣】。
Output: Bleeding from the gums is referred to as【gum bleeding】, and is also known as【tooth bleeding】or【gum oozing】.

Input： 喘病是以症状命名的疾病，既是独立性疾病，也是多种急、慢性疾病过程中的症状。
Input: Asthma is a disease named after its symptoms, it is not only an independent disease but also a symptom in the process of various acute and chronic diseases.
Output：【喘病】是以症状命名的疾病，既是独立性疾病，也是多种急、慢性疾病过程中的症状。
Output:【Asthma】is a disease named after its symptoms, it is not only an independent disease but also a symptom in the process of various acute and chronic diseases.

Input： 喘息气促，咳嗽，咯痰，胸部膨满，胀闷如塞等是疾病的证候特征。
Input: Breathlessness, shortness of breath, coughing, expectoration (phlegm production), chest fullness, and a sensation of chest tightness are characteristic symptoms of the disease.
Output： 喘息气促，咳嗽，咯痰，胸部膨满，胀闷如塞等是疾病的证候特征。
Output: Breathlessness, shortness of breath, coughing, expectoration (phlegm production), chest fullness, and a sensation of chest tightness are characteristic symptoms of the disease.

Input： 咳嗽是以发出咳声、伴有咳痰为主要表现的一种疾病。
Input: Cough is a disease characterized by the emission of cough sounds and accompanied by expectoration..
Output：【咳嗽】是以发出咳声、伴有咳痰为主要表现的一种疾病。
Output: 【Cough】is a disease characterized by the emission of cough sounds and accompanied by expectoration.

Figure 3. Named entity recognition prompt example. The prompt consists of three components: (1) Task description (green part): located at the top of the table, it indicates that the current task for the iFLYTEK Spark Cognitive Large Model is to recognize disease entities using linguistic knowledge. (2) Few–shot demonstrations (red part): in the middle of the table, it provides reference instances of input—-output pairs for different results produced by the iFLYTEK Spark Cognitive Large Model. (3) Input sentences (blue part): at the bottom of the table, it represents the input sentences and output results, with the results generated by the iFLYTEK Spark Cognitive Large Model highlighted in bold.

3.2.2. Matching Prompts Based on Text Similarity

During the NER phase, multiple samples were designed for each prompt category to enhance relevance to the textual content. To achieve effective extraction for seven different entity types, thirty prompt samples were devised for each entity type. Specifically, each set of prompt samples for a given entity type included three different coverage scenarios, with ten samples under each scenario, ensuring a comprehensive evaluation of the model's performance in various contexts.

In the sample selection process, the Sentence–BERT model [27] was employed to compute the semantic similarity between the target text and candidate samples in the sample

set, determining the most suitable prompt. The Sentence–BERT model utilizes a Siamese network structure to generate fixed–length sentence embeddings rich in semantic information. Subsequently, it employs cosine distance, Manhattan distance, or Euclidean distance measurement methods to calculate text similarity. The model exhibits significant advantages in computational efficiency compared to other models.

3.2.3. Self–Validation

Self–validation is mainly used to verify whether the entity extracted from a given sentence belongs to a particular entity type. An example of a self–validation prompt is shown in Figure 4.

你的任务是验证给定句子中用"【】"框出来的实体是否属于疾病实体。以下是一些例子：
Your task is to verify whether the entity in the given sentence, which is framed with "【】", belongs to the disease. Here are some examples:

Input： 给定的句子：【咳嗽】痰多，痰稠色白，晨起或饭后咳甚痰多。
其中句子中用"【】"框出来的实体是否属于疾病实体？回答"是"或"否"
Input: Given sentence: Frequent 【cough】 with thick white phlegm, especially after waking up or meals.
The entities enclosed in "【】" in the sentence, do they belong to disease entities? Answer with "Yes" or "No."
Output： 否
Output: No

Input： 给定的句子：【咳嗽】是指以肺气上逆、咳出痰液为主症的一种疾病。
其中句子中用"【】"框出来的实体是否属于疾病实体？回答"是"或"否"
Input: Given sentence: 【Cough】 refers to a condition characterized by the upward reversal of lung qi and the predominant symptom of expectorating phlegm..
The entities enclosed in "【】" in the sentence, do they belong to disease entities? Answer with "Yes" or "No."
Output： 是
Output: Yes

Input： 给定的句子：【咳嗽】是以发出咳声、伴有咳痰为主要表现的一种疾病。
其中句子中用"【】"框出来的实体是否属于疾病实体？回答"是"或"否"
Input: Given sentence: 【Cough】 is a condition characterized primarily by the emission of cough sounds and the presence of sputum.
The entities enclosed in "【】" in the sentence, do they belong to disease entities? Answer with "Yes" or "No."
Output：是
Output: Yes

Figure 4. Named entity recognition self–verification example. This prompt consists of three parts: (1) Task description (green part): located at the top of the table, it indicates that the current task for the iFLYTEK Spark Cognitive Large Model is to determine whether a given entity is a disease entity. (2) Few–shot demonstrations (red part): positioned in the middle of the table, it provides a few examples for reference to showcase the few–shot learning approach employed by the iFLYTEK Spark Cognitive Large Model. (3) Input sentences (blue part): found at the bottom of the table, it represents the input sentences and corresponding output results, with the results generated by the iFLYTEK Spark Cognitive Large Model highlighted in bold.

(1) Task description.

"Your task is to verify whether the entity in the given sentence, which is framed with "【】", belongs to the [*entity type*]. Here are some examples:" Here, [*entity type*] refers to the category of the target entity to be determined, such as disease entities in this case.

(2) Few–shot demonstrations.

To standardize the output format of LLMs and facilitate the parsing of subsequent data, two example inputs and their expected outputs are provided for each verification. The demonstrations include two scenarios: (a) the provided sentence contains entities that do not belong to the specified entity type; (b) the provided sentence contains entities belonging to the specified entity type. Each demonstration consists of an input sequence X and an expected output sequence Y. Regarding the output format, it is specified as follows: if the entities contained in the input sequence X match the specified [*entity type*], the output is "Yes"; otherwise, if they do not match, the output is "No".

(3) Input sentence.

The given sentence, entities, and entity types.

3.2.4. Entity Relationship Extraction

Figure 5 exemplifies the process of entity relationship extraction. Specifically, the entity relationship extraction task involves relationship judgment prompts that contain the following components:

(1) Task description.

"Your task is to verify whether the given relationship is extracted from the given sentence for the [*relation type*]. Here are some examples:" The relationship extraction task in this study aims to validate the presence of a specific relationship type in the given sentence. Here, the term "relation type" refers to the specific type of relationship to be verified.

(2) Few–shot demonstrations.

To guide the output format of the LLM and facilitate subsequent parsing, we provide two input–output examples. The examples cover two scenarios: (1) instances where two specified entities in the given sentence do not conform to the established relationship type, and (2) instances where two specified entities in the given sentence satisfy the established relationship type. Each example consists of an input sequence X and an expected output sequence Y. In defining the output format, if the input sequence X indicates the existence of the relationship type r between entities e1 and e2, the expected output sequence Y should respond with "Yes"; otherwise, it should output "No".

(3) Input sentence.

The given sentence, entities, and relationship.

3.3. Knowledge Extraction

In the process of constructing a knowledge graph, knowledge extraction is a critical phase aimed at distilling structured information from a vast amount of data, encompassing core elements such as named entities, entity attributes, and entity relationships. The realization of this phase relies not only on cutting–edge natural language processing techniques but also integrates innovative approaches from the fields of deep learning and information extraction to ensure the efficient transformation and processing of unstructured data.

In this study, relevant entities within the field of TCM were successfully identified through named entity recognition techniques. Subsequently, entity relationship extraction methods were employed to reveal the unique relationships among these entities in TCM. Ultimately, by processing semi–structured data, the extraction of relevant attributes associated with the identified entities was achieved, laying a solid foundation for the construction of a knowledge graph rich in domain–specific knowledge.

你的任务是验证给定句子中用“【】”框出来的实体是否属于疾病实体。以下是一些例子：
Your task is to verify whether the entity in the given sentence, which is framed with "【】", belongs to the disease. Here are some examples:

Input： 给定的句子：【咳嗽】痰多，痰稠色白，晨起或饭后咳甚痰多。
其中句子中用“【】”框出来的实体是否属于疾病实体？回答“是”或“否”
Input: Given sentence: Frequent 【cough】 with thick white phlegm, especially after waking up or meals.
The entities enclosed in "【】" in the sentence, do they belong to disease entities? Answer with "Yes" or "No."
Output： 否
Output: No

Input： 给定的句子：【咳嗽】是指以肺气上逆、咳出痰液为主症的一种疾病。
其中句子中用“【】”框出来的实体是否属于疾病实体？回答“是”或“否”
Input: Given sentence: 【Cough】 refers to a condition characterized by the upward reversal of lung qi and the predominant symptom of expectorating phlegm..
The entities enclosed in "【】" in the sentence, do they belong to disease entities? Answer with "Yes" or "No."
Output： 是
Output: Yes

Input： 给定的句子：【咳嗽】是以发出咳声、伴有咳痰为主要表现的一种疾病。
其中句子中用“【】”框出来的实体是否属于疾病实体？回答“是”或“否”
Input: Given sentence: 【Cough】 is a condition characterized primarily by the emission of cough sounds and the presence of sputum.
The entities enclosed in "【】" in the sentence, do they belong to disease entities? Answer with "Yes" or "No."
Output：是
Output: Yes

Figure 5. Entity relationship extraction example. The prompt consists of three parts: (1) Task description (green part): at the top of the table, it indicates that the current task for the iFLYTEK Spark Cognitive Large Model is to determine, using linguistic knowledge, whether a given relationship is a manifestation relationship. (2) Few–shot demonstrations (red part): in the middle of the table, a few examples are provided for reference to demonstrate how iFLYTEK Spark Cognitive Large Model performs in few–shot scenarios. (3) Input sentences (blue part): at the bottom of the table, it presents input sentences and their corresponding output results, with the output from the iFLYTEK Spark Cognitive Large Model highlighted in bold.

3.3.1. Entity Normalization

Due to the data collected from various websites, the use of the same entity may have different terminologies. Therefore, entity standardization is needed to map original terms to standard terms and further create entities through the inheritance of standard terms.

This paper focuses on the field of TCM. By consulting the literature, we have established entity types and identifiers within the entity relationship recognition model, totaling seven categories, disease, symptom, drug, prescription, diet, treatment methods, and etiology and pathogenesis, as shown in Table 1.

3.3.2. Named Entity Recognition

Named entity recognition (NER) refers to the identification of named entities from text and serves as the foundation for information extraction. The results of NER directly impact the outcomes of entity relationship extraction and attribute extraction. In this study,

we conducted named entity extraction on the acquired data using few–shot prompts. The specific implementation is shown in Section 3.2.1.

Figure 3 illustrates the prompt instances used for performing named entity recognition, consisting of three parts. Table A1 in Appendix A illustrates instances of errors in NER.

Table 1. Entity types.

Chinese Name	English Name	Interpretation
疾病	Disease	Entities used to describe various diseases or pathological states in TCM.
症状	Symptom	Subjective feelings or objective manifestations experienced by patients during the course of a disease, such as fever.
药材	Drug	Various herbs, plants, minerals, and animal tissues used in TCM.
方药	Prescription	Combinations of herbs used in TCM prescriptions.
饮食	Diet	Specific foods and food combinations emphasized in TCM health preservation and regulation.
治则治法	Treatment methods	Methods used for treating diseases in TCM.
病因病机	Etiology and pathogenesis	Refers to the causes and mechanisms of disease occurrence and development. Etiology refers to the factors leading to the onset of a disease, such as external wind–cold, emotional imbalances, etc. Pathogenesis refers to the mechanisms and patterns of disease development, such as qi stagnation and blood stasis, yin deficiency with yang excess, etc.

3.3.3. Self–Validation

Due to the challenges posed by hallucination or overprediction issues in LLMs, there is a notable tendency in LLMs for named entity recognition (NER) tasks. Specifically, even when provided with demonstrations, LLMs tend to excessively and overconfidently label empty data as entities [23,28,29]. To mitigate the impact on the results, we designed a self–validation cue to verify that the entity extracted from a given sentence belongs to a specific entity type; the composition of the cue is described in Section 3.2.3. Figure 4 shows an example of an extracted disease entity.

3.3.4. Entity Relationship Extraction

Entity relation extraction, as an essential task in information extraction, refers to the extraction of predefined entity relationships from unstructured text, based on entity recognition. The relationships between entity pairs can be formalized as relationship triplets $< e1,\ r,\ e2 >$, where e1 and e2 are entities, and r belongs to the target relationship set $R\{r1,\ r2, r3 \ldots, ri\}$. The task of relation extraction is to extract relationship triplets $< e1,\ r,\ e2 >$ from natural language text, thereby extracting textual information [30]. For the entity relation extraction part, the LLM is used to obtain entity relationships through a question–and–answer format.

This paper involves seven types of relationships between entities, manifestation, category, treatment, administration, composition, dietary therapy, and induction, as shown in Table 2. The determination of the subject and object of entities in this paper is based on their order of appearance in the text, where the entity that appears first is considered the subject, and the one that appears later is the object.

Table 2. Relationship types.

Relationship	Entity		Example
	Subject	Object	
Manifestation	Disease	Symptom	<风痰袭窍型慢性咳嗽, 咳痰> <Wind–Phlegm Invading the Lung Type Chronic Cough, Cough with Phlegm>
Category	Disease	Disease	<咳嗽, 肺阴亏虚型咳嗽> <Cough, Lung Yin Deficiency Cough >
Treatment	Treatment methods	Disease	<润肺止咳, 肺阴亏虚型咳嗽> <Lung–Moistening and Cough–Relieving Therapy, Lung Yin Deficiency Cough>
Administration	Disease	Prescription	<风盛挛急型慢性咳嗽, 苏黄止咳汤> <Wind Excess and Spasmodic Chronic Cough, Su Huang Zhi Ke Tang >
Composition	Drug	Prescription	<紫苏叶, 苏黄止咳汤> <Perilla Leaf, Su Huang Zhi Ke Tang>
Dietary therapy	Diet	Disease	<葱梨汤, 肺气虚寒咳嗽> <Green Onion and Pear Decoction, Lung Qi Deficiency Cough>
Induction	Etiology and pathogenesis	Disease	<风邪犯肺, 风盛挛急型咳嗽> <Invasion of Wind–Cold in the Lungs, Wind Excess and Spasmodic Cough>

In the context of entity relationships, this paper adopts an inquiry–based approach. For the entities extracted during the previous step of named entity recognition, for any two entities, e1 and e2, we determine the potential relationship r between these two entities based on their types and the types of subject and object in the eight relationship types.

If r does not exist, no inquiry is made; otherwise, we ascertain whether $< e1, r, e2 >$ represents a relationship extracted from the text in the form of a question.

Section 3.2.4 describes in detail the composition and roles of the various parts of the few–shot prompt, and Figure 5 shows an example used for entity relationship extraction, using relationship judgment as an example. Table A2 in Appendix A illustrates instances of errors in entity relationship extraction.

3.3.5. Attribute Extraction

Attribute extraction refers to the task of extracting relevant attributes or features of entities from text. The attribute extraction in this paper primarily derives from semi–structured data found on Baidu Baike and the Chinese Traditional Medicine and Medicinal Herbs website. As shown in Figure 6, the left side represents the extracted data, while the right side displays the processed results. The format used is as follows: $Entity - Attribute\ Type - > Attribute\ Value$.

3.4. Knowledge Fusion

After the knowledge extraction process, entity attributes are extracted from structured data, while entities, relationships between entities, and some entity attributes are extracted from semi–structured and unstructured data. Subsequently, entity fusion is performed to eliminate duplicate information and correct any erroneous data that may have occurred during the extraction process. To achieve this, we extract Chinese medicine synonyms pairs from "Classification and Codes of Traditional Chinese Medicine Diseases" and "Clinical Terminology of Traditional Chinese Medicine" to construct a thesaurus. For any two named entities, a search is conducted in the thesaurus, and if matching synonym pairs are found, the two named entities are considered similar. Afterward, the same entity is used to represent these two similar entities.

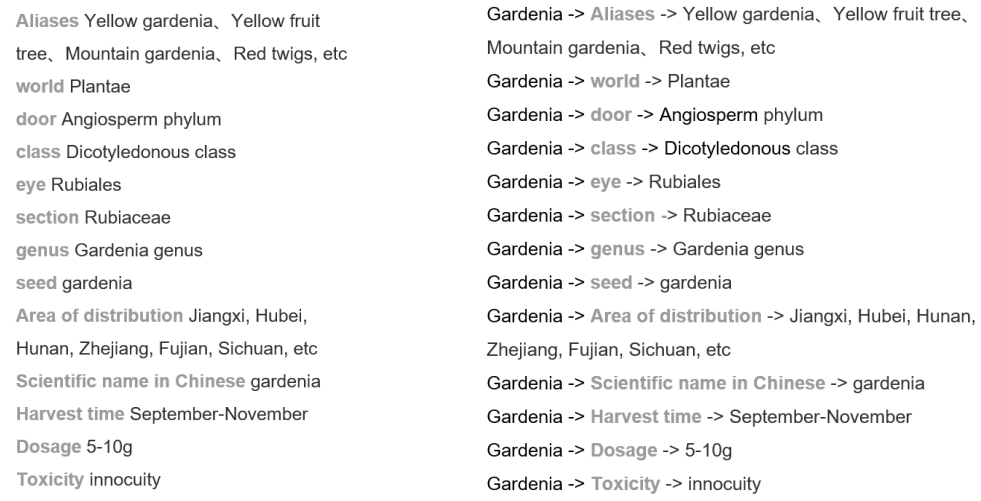

Figure 6. Extraction results of partial data from Baidu Baike's InfoBox. The left image shows the extraction of information from the InfoBox, mainly including two parts, "basicInfo–name" and "basicInfo–value" within the InfoBox, separated by ":". The right image displays the processed data extracted from the InfoBox in the form of property triplets, represented as $(Entity1-Relation->Entity2)$.

3.5. Data Storage

In this paper, the processed data are stored in the form of triplets. In the context of constructing a knowledge graph, the ontology and representation of the knowledge graph can be formalized using standardized languages such as the Resource Description Framework (RDF) and Web Ontology Language (OWL). These languages provide a standardized semantic foundation aimed at facilitating knowledge sharing and reuse across applications and platforms. Specifically, the RDF provides a flexible and scalable data model for encoding information, expressing associations between entities and their properties in the form of triplets. The OWL further enhances the expressive power of the RDF, introducing richer modeling primitives and logical constructs to precisely define complex knowledge systems and support automated reasoning.

However, despite the significant advantages of RDF and OWL in terms of semantic interoperability and machine understandability, they may face performance bottlenecks when dealing with large–scale datasets. In contrast, graph databases optimize access efficiency for large amounts of graph–structured data, with a design focus on achieving efficient data storage, indexing, and querying performance. Graph databases represent entities and their relationships through nodes and edges, similar to the data model of RDF, but typically provide more direct methods to support real–time analysis and management of large datasets.

Given the requirements for data processing and retrieval efficiency in practical applications of knowledge graphs, we choose to adopt graph databases as the solution for data storage. Graph databases not only effectively store and manage vast amounts of entity and relationship data but also support complex queries, thereby meeting the demand for high–performance data access in the later stages of knowledge graph applications. Additionally, the selection of graph databases also facilitates future analysis and mining of graph–structured data that may be involved.

We have chosen the Neo4j graph database for data storage. In the graph, nodes represent entities related to TCM and traditional Chinese herbs. Relationships in the triplets are represented as edges in the graph, indicating some form of relationship between two nodes by pointing from one node to another. These edges point from the subject to the object. Attribute triplets are stored in the graph database in the format $<entity, attribute\ category, attribute>$.

4. Experimental Results and Analysis

4.1. Knowledge Extraction

In this study, we employ the iFLYTEK Spark Cognitive Large Model as the backbone architecture of the LLM to underpin the entire experimental process.

4.1.1. Data Preparation

In the data preparation phase of this study, we employed web crawling techniques to extract relevant textual datasets from specified web pages. Subsequently, a series of data cleaning procedures were applied to the collected corpus to ensure data quality. We randomly selected 200 records from the cleaned dataset as the test set and manually labeled the entities and relationships between them. These 200 records cover different entity types and relationship types to ensure the comprehensiveness and representativeness of the experimental analysis.

Specifically, data crawling was primarily targeted at professional websites in the field of TCM, such as Zhongyi Zhongyao Wang (TCM Online_Traditional Chinese Medicine Network (www.zhzyw.com (16 April 2023))). For the initially acquired data, we conducted preliminary classification based on predefined features. Subsequently, utilizing LLMs, entities and their relationships were extracted from the preprocessed corpus. Finally, based on the results extracted by the language models, we constructed a relationship triplets dataset.

4.1.2. Evaluation Criteria

Standard evaluation measures such as precision (*P*), recall (*R*), and F1–score were employed.

$$P = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \times 100\% \quad (1)$$

$$R = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \times 100\% \quad (2)$$

$$F1_{score} = \frac{2\mathrm{PR}}{\mathrm{P} + \mathrm{R}} \times 100\% \quad (3)$$

Among them, the meanings of TP, FP, and FN are as follows:
TP (True Positive): predicts the correct answer.
FP (False Positive): wrongly predicts other classes as this class.
FN (False Negative): the label of this class is predicted to be the label of other classes.

4.1.3. NER Experimental Results and Analysis

In the NER task, the experiment is based on crawled data. We randomly selected 200 instances from the crawled data as a test set for testing. For these 200 data, we looked for experts in TCM to label the data to ensure the accuracy of the test. The experimental results are delineated in Table 3, where instances highlighted in bold denote the optimal findings. This convention is consistently applied across Tables 4, 5 and 7 as well, ensuring clarity and ease of interpretation.

Table 3. NER Experimental Results (%).

	Precision	**Recall**	**F1–Score**
iFLYTEK Spark + matched few–shot prompts	**93.168%**	87.772%	90.361%

Table 4. Comparison of NER experimental results with and without self–validation (%).

	Precision	**Recall**	**F1–Score**
iFLYTEK Spark without Self–Verification	87.826%	**88.596%**	88.210%
iFLYTEK Spark with Self–Verification	**93.168%**	87.772%	**90.361%**

Table 5. Comparison of NER experimental results of ChatGPT model, ChatGPT + random few–shot prompts, ChatGPT + matched few–shot prompts, iFLYTEK Spark model, iFLYTEK Spark + random few–shot prompts, and iFLYTEK Spark + matched few–shot prompts (%).

	Precision	Recall	F1–Score
ChatGPT	90.850%	81.287%	85.802%
ChatGPT + random few–shot prompts	90.446%	83.041%	86.585%
ChatGPT + matched few–shot prompts	92.258%	83.626%	87.730%
iFLYTEK Spark	82.954%	85.380%	84.150%
iFLYTEK Spark + random few–shot prompts	91.950%	86.842%	89.323%
iFLYTEK Spark + matched few–shot prompts	**93.168%**	**87.772%**	**90.361%**

In this study, we compare the extraction accuracies of two LLMs, ChatGPT and iFLYTEK Spark, in a TCM entity recognition task, the results are shown in Table 4. We also compare the performance of the models in three different learning scenarios: zero–shot prompts, random few–shot prompts, and matched few–shot prompts (as shown in Table 5).

Based on the experimental results, the following conclusions can be drawn: The self–verification improves the performance of the iFLYTEK Spark NE. Although a slight decrease in recall rate is observed, the comprehensive performance metric—namely, the F1 score—experiences a notable enhancement. This indicates that the self–verification strategy has a positive effect on reducing False Positives, while also optimizing overall efficiency. While the slight decrease in recall rate may raise some concern, the improvement in the F1 score emphasizes the effectiveness of this technique in enhancing the overall recognition efficiency of the system.

From the experimental outcomes, it can be observed that LLMs, in the absence of specific guidance, universally exhibit a hallucination phenomenon. This behavior is characterized by the tendency of these models to incorrectly identify non–relevant data as specific entities. The use of prompts, coupled with a self–validation mechanism, proved to be an effective intervention in mitigating this issue. Furthermore, the results indicate that through this optimized strategy, the iFLYTEK Spark model demonstrated superior performance across several evaluation metrics compared to ChatGPT.

4.1.4. Entity Relationship Extraction Experimental Results and Analysis

In the experimental phase of entity relation extraction, this study conducted an analysis on a test dataset consisting of 200 samples. Initially, based on the entities present in the test data, we categorized them into seven main categories according to the possible relationship types between entities. For each category, we employed corresponding few–shot demonstrations.

Given that the relation extraction task in this study closely resembles a binary classification problem, namely determining the presence or absence of a relationship, we selected classification accuracy as the evaluation metric. This choice allows us to directly assess the performance of the model based on its affirmative or negative judgment of relationships. Table 6 shows the results of entity relationship extraction.

Table 6. Entity relationship extraction experimental results.

	Accuracy
iFLYTEK Spark + few–shot prompts	94.0928%

In this study, we delve into and compare the performance of two LLMs—ChatGPT and iFLYTEK Spark—in the task of TCM entity relation extraction. To comprehensively evaluate model performance, two distinct learning scenarios were specifically designed: zero–shot prompt and few–shot prompts (as shown in Table 7).

Table 7. Comparison of entity relationship extraction experimental results of ChatGPT model, ChatGPT + few–shot prompts, iFLYTEK Spark model, and iFLYTEK Spark + few–shot prompts (%).

	Accuracy
ChatGPT	91.772%
ChatGPT + few–shot prompts	93.671%
iFLYTEK Spark	91.561%
iFLYTEK Spark + few–shot prompts	**94.0928%**

In analyzing the experimental results, we observed that, owing to the highly structured nature of the dataset itself and the explicit overlapping relationships between entities, different methods exhibited a high degree of consistency in classification accuracy on this dataset, generally reaching a high level. This phenomenon suggests that, in datasets with clear structural characteristics, various methods may tend to converge in terms of classification performance.

4.2. Data Visualization Storage

Using the above experiments, we can obtain the entities and their relationships contained in the TCM knowledge graph. In this paper, the entities and relationships obtained from the data are stored in the Neo4j graph database. Figure 7 illustrates a small part of the graph database using "sinusitis with pattern of wind–heat in lung channel" as an example. Figures A1 and A2 in Appendix A exemplify additional instances.

Figure 7. "Sinusitis with Pattern of Wind–heat in Lung Channel" and its Partially Related Entities in TCM knowledge graph. Different colored circles represent different entities, and the text on the arrows indicates the relationship between the two entities, with the arrow pointing from the subject to the object.

5. Conclusions and Future Work

The present research proposes a novel approach for constructing a TCM knowledge graph based on the few–shot learning paradigm of LLMs. Initially, this study comprehensively outlines the data crawling process, with a specific emphasis on the strategies employed for acquiring TCM–related textual datasets using web crawling techniques. Subsequently, it provides a detailed exposition of the methods for extracting entities and their relational information from the acquired data. It highlights the key steps in information

extraction by integrating large language models and few–shot learning strategies, with a thorough analysis of the advantages of LLMs. In the experimental process, this research utilizes the iFLYTEK Spark Cognitive Large Model, evaluating its performance based on various metrics. Ultimately, this study adopts the Neo4j graph database as the data storage solution, storing extracted entities, attributes, and relationships in triplet form for effective data management and querying. Additionally, the structural aspects of the TCM knowledge graph are visually presented through data visualization methods.

Based on an in–depth analysis of entity relation extraction results, this study observes that across datasets with significant structural characteristics, various classification methods tend to exhibit consistent performance. This phenomenon suggests that traditional classification accuracy, as an evaluation metric, may not fully reveal performance differences among these methods within such specific datasets. Hence, for a more effective assessment and comparison of the efficiency and effectiveness of different classification methods in handling datasets with similar structural properties, future research efforts need to explore and develop more sensitive evaluation criteria.

In light of this, we plan to continuously optimize the knowledge extraction process in our future work, aiming to enhance the accuracy of data extraction. Additionally, we will strive to continually update and improve the database, gradually expanding its scale to provide a more comprehensive and diverse data resource for entity relation extraction.

Author Contributions: Writing—original draft, Y.Z.; writing—review and editing, Y.Z. and Y.H.; supervision, Y.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. Examples for NER, where the error information is colored in red and the expected correct output is colored in green.

Your task is to use brackets "【】" to select the drug entities in the given sentences. Here are some examples:
Input: The therapeutic effect of Lianhua Qingwen capsules in clearing internal heat is very significant. Ingredients such as gypsum, guanzhong, honeysuckle, and forsythia in the prescription can all clear internal heat. Output: The therapeutic effect of Lianhua Qingwen capsules in clearing internal heat is very significant. Ingredients such as 【gypsum】, 【guanzhong】, 【honeysuckle】, and 【forsythia】 in the prescription can all clear internal heat.
Input: After the local burn is cooled by rinsing with cold water, fresh aloe vera is used to wash with warm water. Output: After the local burn is cooled by rinsing with cold water, fresh 【aloe vera】 is used to wash with warm water.
Input: Recently, authoritative medical research institutions in China have conducted experiments and clinical studies on Lianhua Qingwen capsules. Output: Recently, authoritative medical research institutions in China have conducted experiments and clinical studies on 【Lianhua Qingwen capsules】.
Input: There is a "plague" character in the name of Lianhua Qingwen capsules, and "plague" is actually a type of infectious disease like influenza.
Output: There is a "【plague】" character in the name of Lianhua Qingwen capsules, and "【plague】" is actually a type of infectious disease like influenza. Output: There is a "plague" character in the name of Lianhua Qingwen capsules, and "plague" is actually a type of infectious disease like influenza.

Table A2. Examples for entity relationship extraction, where the error information is colored in red and the expected correct output is colored in green.

Your task is to verify whether the given relationship is extracted from the given sentence for the categorical relationship. Here are some examples:
Input: Given Sentence: The symptoms of blood stasis obstructive palpitations include palpitations, uneasy feeling in the chest, dark complexion, purplish tongue or presence of petechiae, winding of sublingual veins, and rough or knotted pulse. Is blood stasis obstructive palpitations a type of palpitations? Answer "Yes" or "No". Output: Yes Input: Given Sentence: The symptoms of blood stasis obstructive palpitations include palpitations, uneasy feeling in the chest, dark complexion, purplish tongue or presence of petechiae, winding of sublingual veins, and rough or knotted pulse. Is palpitations a type of blood stasis obstructive palpitations? Answer "Yes" or "No". Output: No Input: Given Sentence: Diet stagnation type abdominal pain is due to stagnation of food in the stomach, leading to obstruction of the fu qi (digestive function). Is abdominal pain a type of diet stagnation type abdominal pain? Answer "Yes" or "No".
Output: Yes Output: No

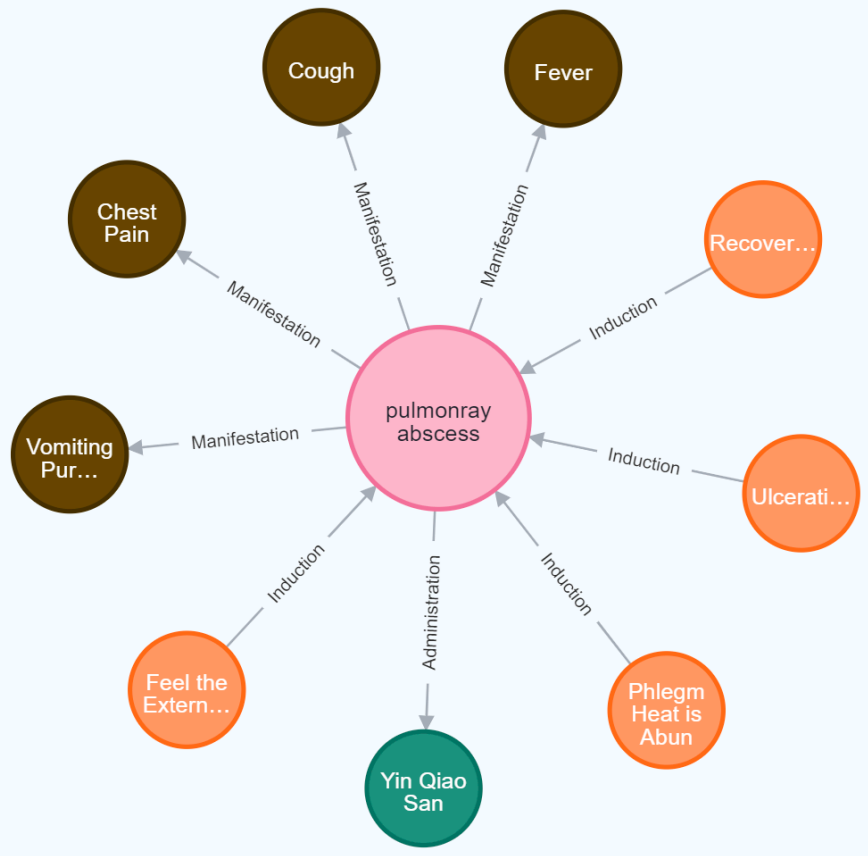

Figure A1. "Pulmonray Abscess" and its Partially Related Entities in TCM knowledge graph. Different colored circles represent different entities, and the text on the arrows indicates the relationship between the two entities, with the arrow pointing from the subject to the object.

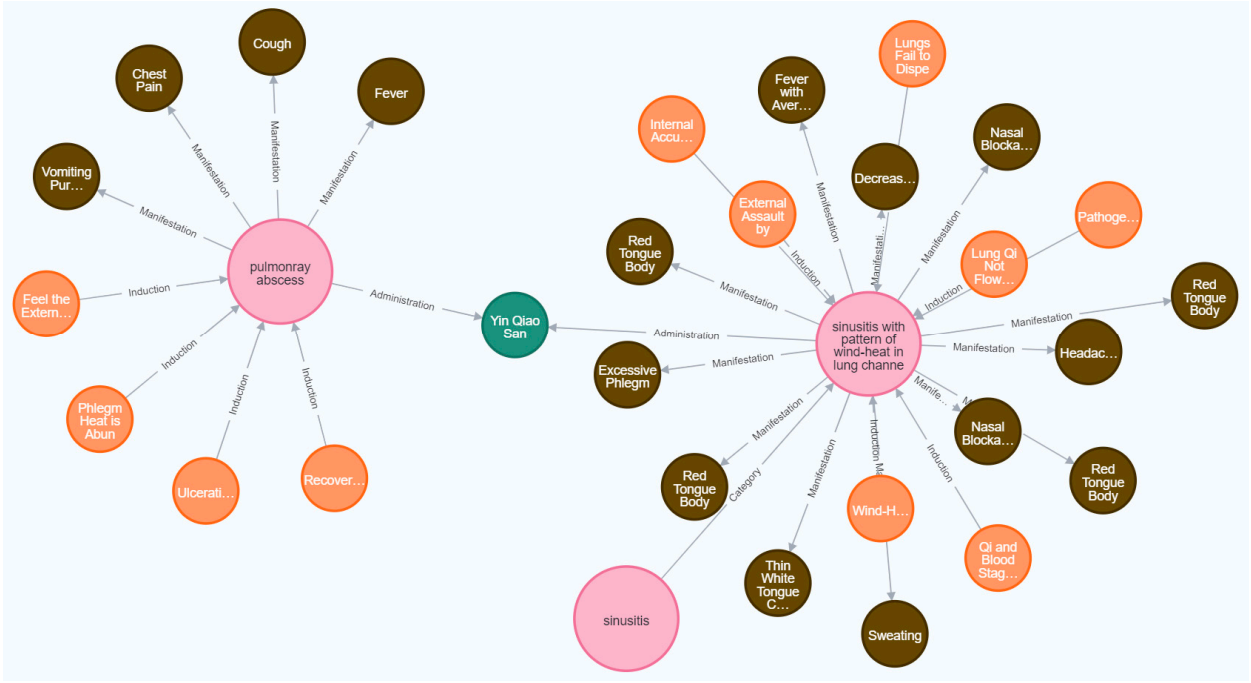

Figure A2. "Yin Qiao San" and its Partially Related Entities in TCM knowledge graph. Different colored circles represent different entities, and the text on the arrows indicates the relationship between the two entities, with the arrow pointing from the subject to the object.

References

1. Sun, X.; Xing, Y. Constructing a health management system with Chinese characteristics: Conceptualization of the construction of the "Cure for Future Diseases" health project. *J. Guangzhou Univ. Tradit. Chin. Med.* **2010**, *27*, 517–551.
2. He, H.; Henderson, J.; Ho, J.C. Distributed tensor decomposition for large scale health analytics. In Proceedings of the World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; pp. 659–669.
3. Liu, Y.; Duan, H.; Sui, Z. A data base study on knowledge discovery of unrelated literature—The construction of linguistic knowledge base of ancient Chinese medicine literature as an example. *J. Intell.* **2006**, *9*, 21–26.
4. Singhal, A. Introducing the Knowledge Graph: Things, Not Strings. Available online: https://blog.google/products/search/introducing-knowledge-graph-things-not/ (accessed on 30 November 2022).
5. Chen, H.; Luo, X. An automatic literature knowledge graph and reasoning network modeling framework based on ontology and natural language processing. *Adv. Eng. Inform.* **2019**, *42*, 100959. [CrossRef]
6. Krishna Kommineni, V.; König–Ries, B.; Samuel, S. From human experts to machines: An LLM supported approach to ontology and knowledge graph construction. *arXiv* **2024**, arXiv:2403.08345.
7. Raiaan, M.A.K.; Fatema, K.; Khan, I.U.; Azam, S.; ur Rashid, M.R.; Mukta, M.S.H.; Jonkman, M.; De Boer, F. A lightweight robust deep learning model gained high accuracy in classifying a wide range of diabetic retinopathy images. *IEEE Access* **2023**, *11*, 42361–42388. [CrossRef]
8. Xu, T.; Guo, C.; Du, L.; Xu, J.; Zhang, P.; Feng, X.; Li, M. A Method for Traditional Chinese Medicine Knowledge Graph Dynamic Construction. In Proceedings of the 5th International Conference on Big Data Technologies, Qingdao, China, 23–25 September 2022; pp. 196–202.
9. Cheng, B.; Zhang, J.; Liu, H.; Cai, M.; Wang, Y. Research on medical knowledge graph for stroke. *J. Healthc. Eng.* **2021**, *2021*, 5531327. [CrossRef]
10. Xiong, W.; Cao, J.; Zhou, X.; Du, J.; Nie, B.; Zeng, Z.; Li, T. Design and evaluation of a prescription drug monitoring program for Chinese patent medicine based on knowledge graph. *Evid.–Based Complement. Altern. Med.* **2021**, *2021*, 9970063. [CrossRef]
11. Zhou, Y.; Qi, X.; Huang, Y.; Ju, F. Research on construction and application of TCM knowledge graph based on ancient Chinese texts. In Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence–Companion Volume, Thessaloniki, Greece, 14–17 October 2019; pp. 144–147.
12. Zheng, Z.; Liu, Y.; Zhang, Y.; Wen, C. TCMKG: A deep learning based traditional Chinese medicine knowledge graph platform. In Proceedings of the 2020 IEEE International Conference on Knowledge Graph (ICKG), Nanjing, China, 9–11 August 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 560–564.
13. Liu, B.; Xiao, X.; Zou, B.; Zhou, Z.; Zheng, L.; Tan, J. A BERT–BiLSTM–CRF Named Entity Recognition Model for Traditional Chinese Medicine Cases Incorporating Chinese Character Radicals. *biomedRxiv* **2023**, biomedRxiv:202303.00004.

14. He, Y.; Luo, C.; Hu, B. A Geographic Named Entity Recognition Method Based on the Combination of CRF and Rules. *Comput. Appl. Softw.* **2015**, *32*, 179–185.
15. Han, X.; Huang, D. Study of Chinese Part–of–Speech Tagging Based on Semi–Supervised Hidden Markov Model. *Small Microcomput. Syst.* **2015**, *36*, 2813–2816.
16. Sherstinsky, A. Fundamentals of recurrent neural network (RNN) and long short–term memory (LSTM) network. *Phys. D Nonlinear Phenom.* **2020**, *404*, 132306. [CrossRef]
17. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre–training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:181004805.
18. Lee, J.; Yoon, W.; Kim, S.; Kim, D.; Kim, S.; So, C.H.; Kang, J. BioBERT: A pre–trained biomedical language representation model for biomedical text mining. *Bioinformatics* **2020**, *36*, 1234–1240. [CrossRef] [PubMed]
19. Huang, K.; Altosaar, J.; Ranganath, R. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *arXiv* **2019**, arXiv:190405342.
20. Rasmy, L.; Xiang, Y.; Xie, Z.; Tao, C.; Zhi, D. Med–BERT: Pretrained contextualized embeddings on large–scale structured electronic health records for disease prediction. *NPJ Digit. Med.* **2021**, *4*, 86. [CrossRef]
21. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few–shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
22. OpenAI Introducing Chatgpt. OpenAI. Available online: https://openai.com/blog/chatgpt (accessed on 30 November 2022).
23. Ni, X.; Li, P. Unified Text Structuralization with Instruction–tuned Language Models. *arXiv* **2023**, arXiv:2303.14956.
24. Hu, Y.; Ameer, I.; Zuo, X.; Peng, X.; Zhou, Y.; Li, Z.; Li, Y.; Li, J.; Jinag, X.; Xu, H. Zero–shot clinical entity recognition using chatgpt. *arXiv* **2023**, arXiv:2303.16416.
25. Li, F.-F.; Fergus, R.; Perona, P. One–shot learning of object categories. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *28*, 594–611.
26. Fink, M. Object classification from a single example utilizing class relevance metrics. *Adv. Neural Inf. Process. Syst.* **2004**, *17*, 449–456.
27. Reimers, N.; Gurevych, I. Sentence–bert: Sentence embeddings using siamese bert–networks. *arXiv* **2019**, arXiv:1908.10084.
28. Wang, S.; Sun, X.; Li, X.; Ouyang, R.; Wu, F.; Zhang, T.; Li, J.; Wang, G. Gpt–ner: Named entity recognition via large language models. *arXiv* **2023**, arXiv:2304.10428.
29. Braverman, M.; Chen, X.; Kakade, S.; Narasimhan, K.; Zhang, C.; Zhang, Y. Calibration, entropy rates, and memory in language models. In Proceedings of the International Conference on Machine Learning, Virtual, 13–18 July 2020; pp. 1089–1099.
30. E, H.-H.; Zhang, W.-J.; Xiao, S.-Q.; Cheng, R.; Hu, Y.-X.; Zhou, X.-S.; Niu, P.-Q. Survey of entity relationship extraction based on deep learning. *J. Softw.* **2019**, *30*, 1793–1818.