# RRNorm: A Novel Framework for Chinese Disease Diagnoses Normalization via LLM-Driven Terminology Component Recognition and Reconstruction

**Yongqi Fan**[♠], **Yansha Zhu**[♠], **Kui Xue**[♢], **Jingping Liu**[♠]*, **Tong Ruan**[♠]*
[♠]School of Information Science and Engineering, East China University of Science and Technology, Shanghai, China
[♢]Shanghai Artificial Intelligence Laboratory, Shanghai, China
{johnnyfans,sanchan}@mail.ecust.edu.cn, xuekui@pjlab.org.cn
{jingpingliu,ruantong}@ecust.edu.cn

## Abstract

The Clinical Terminology Normalization aims at finding standard terms from a given termbase for mentions extracted from clinical texts. However, we found that extracted mentions suffer from the multi-implication problem, especially disease diagnoses. The reason for this is that physicians often use abbreviations, conjunctions, and juxtapositions when writing diagnoses, and it is difficult to manually decompose. To address this problem, we propose a Terminology Component Recognition and Reconstruction strategy that leverages the reasoning capability of large language models (LLMs) to recognize the components of terms, enabling automated decomposition and transforming original mentions into multiple atomic mentions. Furthermore, we adopt the mainstream "Recall and Rank" framework to apply the benefits of the above strategy to the task flow. By leveraging the LLM incorporating the advanced sampling strategies, we design a sampling algorithm for atomic mentions and train the recall model using contrastive learning. Besides the information about the components is also used as knowledge to guide the final term ranking and selection. The experimental results show that our proposed strategy effectively improves the performance of the terminology normalization task and our proposed approach achieves state-of-the-art on the experimental dataset. We release our code and data on the repository RRNorm.

## 1 Introduction

Clinical Terminology Normalization (CTN) plays an important role in clinical natural language processing (Schulz et al., 2019). CTN aims at mapping non-standard clinical mentions such as some fragments extracted from clinical texts, to standard terms within the certain knowledge base (Bodenreider et al., 2018) such as the International Classification of Diseases and Related Health Problems 10-th Revision (ICD-10), which provides the foundation for downstream tasks in the clinical domain. Standardized clinical terms, which are the standard expression of a terminology concept, are applied in tasks such as medical research, data analysis, and healthcare quality improvement that utilize electronic health record information.

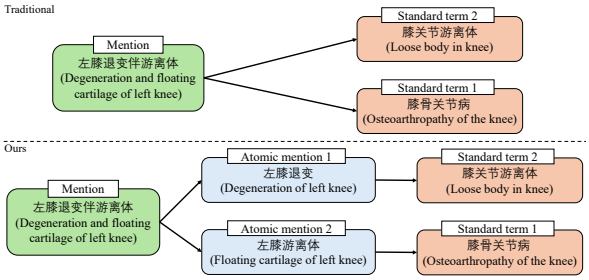


Figure 1: A comparative overview of our proposed approach as opposed to the traditional approach.

More specifically, in the Chinese medical domain, the CTN task faces the challenge of "multi-implication" (Liang et al., 2021; Yan et al., 2020) which means that one mention contains multiple meanings. The non-standardized writing habits such as abbreviation, hyphenation, and juxtaposition lead to this issue and cause inconsistent granularity problems, which affects the effectiveness of traditional term normalization methods based on the embedding model.

Previous work has primarily focused on situations where uni-implication mentions that correspond to only one standard term, are in the majority, such as the CHIP 2019 task that used the International Statistical Classification of Diseases and Related Health Problems 9-th Revision (ICD-9) as the knowledge base (Yan et al., 2020). In CHIP 2019, multi-implication mentions constituted only 4% of the total. Their approaches to solving such problems include two-stage methods involving the "Recall and Re-rank" framework as well as the generative approach. However, previous studies

*Corresponding authors.



*Findings of the Association for Computational Linguistics ACL 2024*, pages 9162–9175

rarely researched the original mentions when facing datasets with the "multi-implication" problem accounting for the majority. Multiple meanings can lead to semantic ambiguity when compared to uni-implication terms, and make it difficult to train the model. If the intermediate results after decomposition can be obtained, transforming the original task into multiple normalization tasks with uni-implications, the difficulty of the task will decrease. Figure 1 presents the difference between our method and the traditional method.

Figure 2 representative multi-implication examples of the Chinese clinical normalization task from the CHIP-CDN dataset. Case 1 shows the simplest scenario: there are clear separation between different atomic mentions. The multi-implication problem in case 1 can be resolved using traditional word segmentation methods or term recognition methods based on sequence labeling. Conversely, the following two cases cannot be segmented or recognized using traditional methods. In case 2, the original mention omitted the repeated occurrence of the affected body part *knee*. Both word segmentation and term recognition result in information loss. The most challenging case, case 3 denotes a semantic implication relation, which cannot be accomplished by direct splitting.

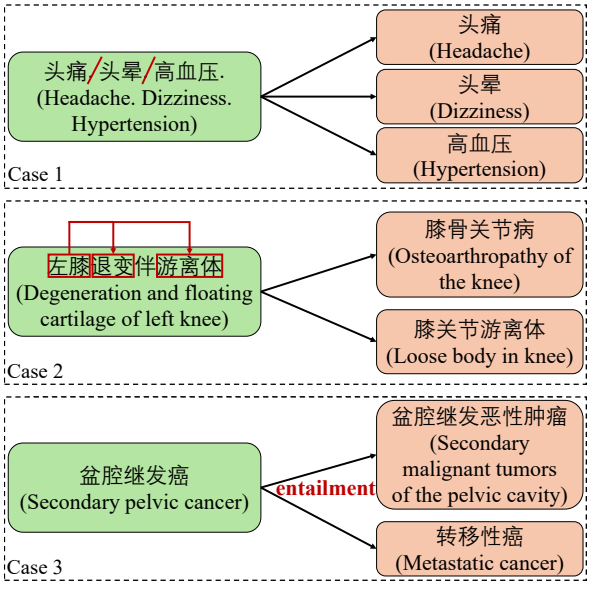


Figure 2: Three cases of different muli-implication examples.

After analysis, the samples without clear boundaries in CHIP-CDN account for a large number of cases and cannot be ignored. The distribution of the multi-implication samples in the train set and validation set of CHIP-CDN are shown in Table 1. It can be observed that multi-implication mentions constitute more than half of the total, and for multi-implication mentions, the inseparable cases make up more than 70%. This result demonstrates the necessity of research on multi-implication problems.

To address the defect caused by the multi-implication problem mentioned above, we propose a "Terminology Component Recognition and Reconstruction" strategy based on LLM, which helps identify the atomic mentions implied within a long Chinese mention. This module primarily consists of two main steps, First, we utilized prompt engineering technology to recognize the specific types of components by LLM, the components include such as affected body parts and disease content are identified at a finer granularity. Then we manually crafted specific rules to restructure the components obtained in the recognition step. The order of components appeared in the original mention and the type of components were taken into consideration during reconstruction.

| Datasets | Separable | Inseparable | Multi-implication |
|---|---|---|---|
| Train | 946 | 2318 | 3264 |
| Validation | 288 | 741 | 1029 |

Table 1: The distribution of the multi-implication samples of CHIP-CDN. A mention is multi-implication if it contains multiple meanings. The manifestation is that one mention corresponds to multiple standard terms."Separable" means that there are obvious separators in the original mention, such as semicolons "; ".

Furthermore, we propose a new framework for terminology normalization based on the Terminology Component Recognition and Reconstruction (RR) strategy, named RR-Norm. In addition to designing the "RR" module, we apply the gains from "RR" to the "Recall and Re-rank" process, implementing the "Atomic-Sampling-based Contrastive Learning" module and "Knowledge-Guided Term Selection" module to finish the CTN task.

Additionally, we constructed a new dataset "CHIP-CDN-RR" primarily composed of uni-implication mentions during the implementation of the Atomic Sampling part, which aligns more closely with traditional term normalization tasks and effectively reduces the complexity of the original task. We demonstrated its effectiveness by applying this dataset to various baseline methods.

Overall, our contributions are as follows.

- We propose the "Terminology Component Recognition and Reconstruction" strategy based on the LLM. We use in-context learning to let the LLM learn to recognize, split, and group the components of the original mention, and then through reconstruction, we obtain a set of uni-implication atomic mentions.

- We propose a terminology normalization framework based on the "Terminology Component Recognition and Reconstruction" strategy and achieve performance improvement in the CTN task. Including the "RR" module, the "Atomic Sampling" algorithm is designed to obtain high-quality positive and negative samples and train more effective recall models using contrastive learning, and the "Knowledge-Guided Term Selection" module that leverages the attention mechanism to capture the knowledge in the constituent information.

- We constructed an almost uni-implication dataset CHIP-CDN-RR to map the annotated answers in the original dataset to the reorganized atomic mentions during the implementation of the atomic sampling stage, and the final uni-implication ratio was improved by 37%.

## 2 Related Work

The most traditional method of clinical terminology normalization is accomplished through retrieval methods, such as BM25, and Edit Distance. With the development of deep learning, the majority of text similarity approaches are calculated by generating word vectors. Leal et al. (2015) employs a similarity search based on Lucene's implementation of Levenshtein and N-gram distances. Leaman et al. (2013) proposed a linear pairwise model for the representation of medical terms.

Many of the term normalization studies consider the task of normalizing the terminology to involve a multi-classification problem as well. Limsopatham and Collier (2016) introduced the convolutional and recurrent neural network architecture. Niu et al. (2019) presented a multi-task character-level attentional network that learned character structure features. Methods based only on retrieval or classification are not very accurate, hence the recall and re-rank method was introduced, which means another model is trained to re-rank the candidate terms obtained by the recall model. Ji et al. (2020) first conducted the BM25 scores as the recall evaluation and proposed a term normalization architecture by fine-tuning the existing BERT models. Xu et al. (2020) proposed an architecture consisting of a candidate generator and a list-wise ranker based on BERT. (Liu et al., 2020) provided an ABTSBM method for ICD-9 terminology standardization.

However, it has been found that there are multi-implication issues in the field of Chinese healthcare. To solve this problem, Sui et al. (2022); Zhang et al. (2023)have added several prediction modules to the original normalization framework. Among them, using multiple classification methods inevitably encounters long tail problems. Yan et al. (2020) suggested a sequence generative framework to directly generate all the corresponding medical procedure terms. The generative method achieves this task well by avoiding number prediction, but the method tends to be inefficient. Liang et al. (2021) considered introducing a tagging task when predicting the implication number. Inspired by this, when considering introducing word segmentation to solve problems, we found that the method based on rule (Liu et al., 2012; Gai et al., 2014) is difficult to cope with the complexity of Chinese medical terminology (Ding et al., 2021). The sequence annotation method (Zhao et al., 2006) relies on high-quality datasets and cannot address the Ellipsis problem of Chinese medical terminology.

## 3 Methods

In this section, we will introduce the proposed framework for Chinese terminology normalization based on terminology component recognition and reconstruction strategy, including the "RR" module to pre-process the original mention, the recall module to get candidates, and the selection module to find the appropriate standard terms, shown in Figure 3.

### 3.1 RR: Recognition and Reconstruction

The objective of this module is to decompose an original multi-implication mention into atomic mentions to reduce task complexity. For instance, when presented with an original mention such as *Degeneration of the left knee with loose body*, it is expected to be segmented into two atomic mentions: *Degeneration of left knee*, and *Loose body in left knee*. These two mentions can be separately mapped to the standard terms: *Osteoarthropathy of the knee* and *Loose body in knee*, which corre-

| Type | Meaning | Abbr. |
|---|---|---|
| Disease Content | Possible symptoms, lesions, and conditions within the scope of onset | DC |
| Disease Scope | Anatomical sites where lesions occur | DS |
| Operation Content | Treatment methods or examination methods | OC |
| Modifier | Words describing the degree and nature of the condition, or directional terms indicating the location within the scope of onset | Mo |
| Separator Word | Separator word or delimiter | SEP |
| Invalid Content | Meaningless description | IC |

Table 2: The specific meanings of predefined components types, where "Abbr." notes the abbreviation.The predefined component types are a summary result obtained by manually analyzing the dataset. Specifically, we sampled 10% of the multi-implication mentions in the train set and dev set respectively for observation, and summarization, and let LLM try to identify the component types and carry out manual corrections (foundation-based medical knowledge). By organizing, we consolidated some meaningless and uncategorizable types into "Invalid Content" (roughly constitutes 6%) and finally gave six types

spond to the original mention. Echoing the goals above, we design the RR module that implements the segmentation of mentions by the recognition and reconstruction of terminology components.

#### 3.1.1 Component Recognition

Given an input original mention $m$ or a term, by prompt engineering, ChatGPT is used to recognize component sequence $\{c_1, c_2, ...c_n\}$ and corresponding specific type sequence $\{t_1, t_2, ..., t_n\}$, where $t_i$ belongs to a predefined type set. The specific content and meanings of the type set can be found in Table 2. From this step, we can get the corresponding components tables, as shown in the upper part of Figure 3, and this structured and ordered knowledge will be used to reconstruct the atomic mentions as well as to train the term selection model.

The prompt for component recognition contains basic task definitions, task output formats, and predefined component types. While we use in-context learning to enhance LLM to understand the predefined types, we provide manually selected demonstration examples and restrict the output format. Additionally, we have added an emphasis section to reiterate the task requirements. This step has proven effective in enhancing the quality of results during practical use. The specific prompt is available in Appendix A. In addition, we require the output to be in JSON format, so it is convenient to perform further post-processing of the results to avoid generating incorrect, missing, or redundant components.

#### 3.1.2 Atomic Mention Reconstruction

Given a sequence of components $c_1, c_2, ...c_n$ and corresponding type sequence $t_1, t_2, ..., t_n$, this module will reconstruct components into atomic mentions $m_1, m_2, ..., m_i$. In this section, we will introduce the detailed algorithm of each stage. The overall approach consists of Knowledge Enhancement, Rule-based Combination, and Fact-checking.

**Knowledge Enhancement** To reduce errors in the recognition module and ensure the accuracy and professionalism of the reconstruction, we intend to introduce domain knowledge to enhance its capability. Firstly, we performed knowledge distillation from ChatGPT by using prompt engineering to obtain a mapping dictionary that maps some expanded synonyms and abbreviations to a standardized terminology component set. The standardized terminology component set is constructed based on the ICD-10 standard terminology base through the "Component Recognition" module in Section 3.1. The distillation prompt is shown in Appendix A. We use these standard terminology components as a supplement to the component table. If the recognition module recognizes components that appear in synonyms and abbreviations above, these less professional expressions will be replaced with corresponding standard components.

**Rule-based Combination** In this stage, we will enumerate all atomic mentions as comprehensively as possible based on a specific set of rules and the component tables, which contain component sequences and type sequences, the detailed procedure is illustrated in Algorithm 1.

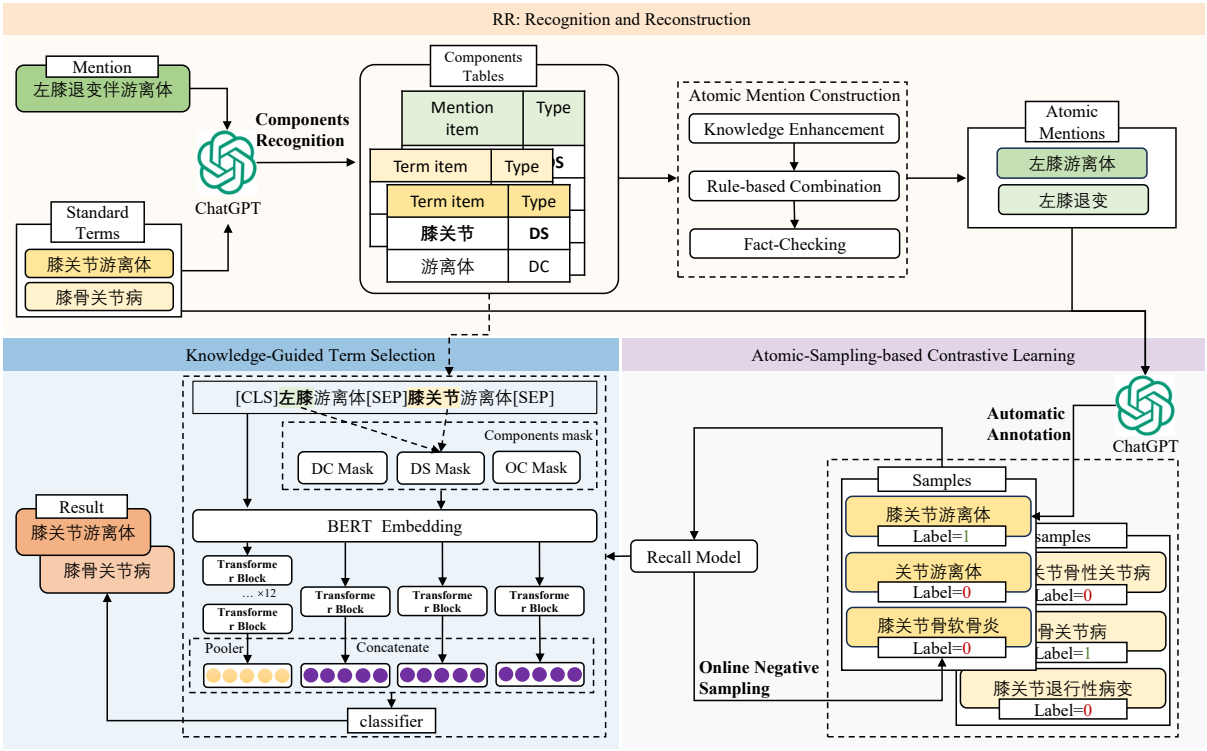


Figure 3: The workflow of our system. In addition to the original mention, standard terminology will also be recognized through ChatGPT. This step is to perform special mask operations based on the composition category during the selection stage.

The algorithm processes the component and type sequence in order according to the original mention. The algorithm maintains a starting position $start$, which determines the starting position of searching for prefix sequences. Whenever the algorithm encounters a component of the category "Disease Content (DC)", it will form one or more prefix sequences starting from the starting position. The prefix sequence of "disease content (DC)" and "operation content (OC)" follows two rules:

1. All "modifiers (Mo)" encountered will be added to the prefix sequence.

2. Each encountered "disease scope (DS)" forms a new prefix sequence.

Afterward, each prefix sequence will be concatenated with the current component and added to the atomic mentions set. More specifically, when encountering "separators (SEP)", the algorithm will not add any atomic terms, but will adjust the starting position based on the component types at both ends of the "separator (SEP)". Let us denote the atomic mentions set as $\mathcal{M}$.

Due to the complexity of Chinese expressions, the correctness of atomic references is not considered at this stage. In general, for example, the reference "*The left knee degeneration with loose body*" can be empirically recognized by medical experts as "*Left knee Degeneration*" and "*Left knee loose body*". After parsing, we obtain the rule that the subsequence "*DS-DC1-SEP-DC2*" should be constructed as "*DS-DC1*" and "*DS-DC2*". However, there are also some special cases, such as "*Chest pain and diabetes mellitus* ", that cannot be applied to the previous rule. In this case, its subsequence "*DS-DC1-SEP-DC2*" should be constructed as "*DS-DC1*" and "*DC2*". Hence, strict rules cannot be generalized as this may lead to semantic loss or errors. When compared with strict rules and arbitrary combinations, the loose rules we adopt can provide reasonable combinations to a certain extent and also ensure the number of combinations to prevent semantic loss.

**Fact-Checking** In the end, we need to do fact-checking for the constructed set of atomic mentions, removing irrational combinations. As illustrated in Algorithm 2, we perform a vector similarity search in the standard terminology base $\mathbf{T}$ for each atomic mention from $\mathcal{M}$. For any atomic mention, we focus on its highest score, denoted $s_{\max}$. We will consider that this atomic mention theoretically does not exist and should be abandoned if $s_{\max}$ is less than the set threshold $\tau$, after

this step, we get the final atomic mention set $\hat{\mathcal{M}}$.

**Algorithm 1:** Algorithm of Rule-based Combination

**Input:** a sequence of components $S_c$, specific type sequence $S_t$, rule base $\mathcal{R}$
**Output:** atomic mentions set $\mathcal{M}$

$\mathcal{M}$=$Set()$
$start$=0
**foreach** $c_i$ *in* $S_c$, $t_i$ *in* $S_t$ **do**
  **if** $t_i$ == *"DC" or* $t_i$ == *"OC"* **then**
    $c_i \leftarrow$ findPreMo($start$,$i$) + $c_i$;
    $P \leftarrow$ findPreDS($start$,$i$);
    **foreach** $pre_{ds}$ *in* $P$ **do**
      $\mathcal{M}$.append($pre_{ds} + c_i$);
    **end**
    $start \leftarrow i$;
  **end**
  **else if** $t_i$ == *"SEP"* **then**
    $start \leftarrow$ adjustStart($i-1$,$i+1$);
  **end**
**end**

**Algorithm 2:** Algorithm of Fact-Checking

**Input:** atomic mention set $\mathcal{M}$
standard terminology base $T$, threshold $\tau$
**Output:** the fact-checked atomic mention set $\hat{\mathcal{M}}$

$\hat{\mathcal{M}} = set()$
**foreach** $m$ *in* $\mathcal{M}$ **do**
  calculateSim$(m, T) \rightarrow s \in S$;
  $s_{\max} = max(S, 0)$;
  **if** $s_{max} > \tau$ **then**
    $\hat{\mathcal{M}}$.append$(m)$;
  **end**
**end**

### 3.2 Atomic-Sampling-based Contrastive Learning

The goal of this module is to train an embedding-based recall model using contrastive learning, which is used to recall several candidate terms for the input mentions from a large-scale standard terminology base for the next term selection step. The most important of these is the selection of positive and negative samples, so we design an atomic sampling algorithm using the LLM, based on the atomic mentions obtained above and advanced online negative sampling techniques proposed by (Liang et al., 2021).

A multi-implication sample in the existing dataset would only provide the original mention and its corresponding standard term, but after the RR module we have decomposed the original mention into multiple atomic mentions, so to get the positive sample, we devised an automatic annotation method to solve the mapping between atomic mentions and standard terms to obtain training data that does not require additional manual annotation. For an original mention, we provide its atomic mentions set and its standard terms in order as input to ChatGPT and let it select the best matching atomic mention for a standard term. A specific prompt template is provided in the appendix A. During the implementation, we constructed an almost uni-implication dataset based on CHIP-CDN, named CHIP-CDN-RR, the relevant details will be presented in 4.1.

For the negative samples, we use the online negative sampling strategy, i.e., before the start of each training epoch, we use the current model to get the vector set $\{v^m\}$, $m \in \hat{\mathcal{M}}$, and the vector set $\{v^t\}$, $e \in \mathbf{T}$. We use the L2 distance between vectors as a metric, for an atomic mention, terms that are close but not standard answers as negative samples, plus the ground truth of the samples, and this constructs the data for the updated one batch.

We trained the recall model based on the SBERT (Reimers and Gurevych, 2019) framework and contrastive learning, with the atomic term serving as anchor $a$ and the correct answer as positive $p$. For any two inputs $i$ and $j$, either mentions or terms, let $D_{i,j} = \left\|v^i - v^j\right\|_2$ denote the L2 distance between them. Then we use the triplet loss, which is shown in formula 1 for the training of the encoder, where $m$ is a manually set hyperparameter indicating the margin in contrastive learning. During the prediction phase, we will recall the $k$ standard terms with the closest distance as the candidate for each atomic mention.

$$\mathcal{L}_{trip.} = max\left(D_{a,p} - D_{a,n} + m, 0\right) \quad (1)$$

### 3.3 Knowledge-guided Term Selection

Given an atomic mention $m$ and corresponding candidates set $C$, we utilized a BERT-based classification model with guidance on identical components to select the term that matches the atomic mention.

For a given input pair $(m, c)$, along with the corresponding component sequences$(s_m, s_c)$ obtained

by the "Component Recognition" module in Section 3.1, we pre-process them to get four kinds of inputs. The first one contains full content, where the original $m$ and $c$ are tokenized by connecting them with "[SEP]". The remaining inputs focus on the specific component types by masking out other types of tokens, including disease content, disease scope, and operation content.

As shown in Figure 3, for the first input, we use the pooler output of BERT as the classification feature, and the other three use the same BERT Embedding Layer but connect different randomly initialized transformer blocks to get different features. Finally, we concatenate these four features together as the input to the classification MLP layer. The binary Cross-Entropy loss is used for the training of the classifier.

# 4 Experiment

## 4.1 Dataset

The CHIP-CDN dataset aims at normalizing the terminologies from the final diagnoses of Chinese electric medical records based on the International Classification of Diseases (ICD-10), which was first released in CHIP2020 and is collected in the Chinese Biomedical Language Understanding Evaluation Benchmark (CBLUE)[1] (Zhang et al., 2021). This is a typical dataset suffering from the "multi-implication" problem that contains 6,000 training samples, 2,000 validation samples, and 10,000 test samples.

| Datasets | Uni-implication | Multi-implication | Total |
|---|---|---|---|
| $\text{CDN}_{train}$ | 2736 | 3264 | 6000 |
| $\text{CDN-RR}_{train}$ | 7915 | 1672 | 9587 |

Table 3: Comparison of CHIP-CDN vs. CHIP-CDN-RR for multi-implication problem on training data.

As mentioned in 3.2 above, We constructed the CHIP-CDN-RR as our experimental dataset to validate our ideas, which consists mainly of the uni-implication training data we constructed. Table 3 shows the comparison between CHIP-CDN and CHIP-CDN-RR on the training set.

## 4.2 Implementation details

We implement our approach on 2 NVIDIA GeForce 3090 GPUs, saving the checkpoints that performed best on the validation set. We use gpt-3.5-turbo-1106 (OpenAI, 2023) as the basic LLM, the temperature is set as 0, and the seed is set as 42.

Following the setting of (Zhang et al., 2023) we adopt the "chinese-roberta-wwm-ext-base" (Cui et al., 2019) as the backbone of the recall model and adopt the "bert-base-chinese" as the backbone of the term selection model. For both the recall model and the term selection model, we use the Adam optimizer, the initial learning rate is 2e-5, batch size is set as 64, and the max length of input tokens is 64. During the online negative sampling stage of the recall model, the number of samples for each atomic mention is 20.

## 4.3 Experimental setup and result analysis

**For the recall part**, we validate the effectiveness of term component Recognition and Reconstruction (RR) on three models, namely the traditional BM25 model, the fixed M3E (Wang Yuxin, 2023) embedding model, the trained triplet SBERT (Reimers and Gurevych, 2019) based on contrast learning and the trained triplet SBERT with online negative sampling strategy (Liang et al., 2021). We use "HR@num" as the evaluation metric, i.e., the hit rate of recalled candidate terms that contain the correct answer, and the "num" is the number of recalled candidate terms.

| Approach | RR | HR@10 | HR@20 | HR@40 |
|---|---|---|---|---|
| BM25 | ✗ | 42.00 | 49.95 | 57.80 |
| | ✓ | 44.15 | 53.05 | 61.40 |
| M3E | ✗ | 40.40 | 48.90 | 57.00 |
| | ✓ | 51.30 | 59.80 | 67.35 |
| SBERT | ✗ | 74.45 | 82.30 | 88.10 |
| | ✓ | 76.25 | 83.80 | 89.75 |
| SBERT$^{\dagger}$ | ✗ | 82.25 | 87.15 | 91.00 |
| | ✓ | **84.30** | **88.70** | **92.25** |

Table 4: Comparison of hit rates between with or without "RR" on different recall models on the validation set of CHIP-CDN.Among them, SBERT$^{\dagger}$ uses online negative sampling for training, while SBERT uses random sampling. To keep it fair, if an original mention contains $k$ atomic mentions, we recall $num/k$ candidate terms for each atomic mention separately, so that for the same sample, with or without RR, the number of recalled candidates is the same.

As shown in Table 4, we compared the hit rates of different recall methods on the CHIP-CDN vali-

[1] https://tianchi.aliyun.com/dataset/95414

| Approach | Micro-F1 | Precision | Recall |
|---|---|---|---|
| M3E + BERT-rank | 55.40 | - | - |
| M3E + RoBERTa-rank(Cui et al., 2019) | 56.40 | 55.30 | 57.50 |
| M3E + Ernie-rank(Sun et al., 2021) | 59.28 | **62.27** | 56.57 |
| RR + M3E + BERT-rank | 57.97 | 55.03 | 61.25 |
| RR + M3E + RoBERTa-rank | 59.29 | 56.59 | 62.10 |
| RR + M3E + Ernie-rank | **62.05** | 58.15 | **66.52** |
| ConstraintDecoding(Yan et al., 2020) | 56.64 | **58.82** | 54.61 |
| RR + ConstraintDecoding | **58.92** | 57.59 | **60.32** |
| DependencyTree + GNN + rank(Zhang et al., 2023) | 63.10 | **61.80** | 64.60 |
| Recall + BERT-rank | 61.98 | 61.49 | 62.49 |
| Recall$^{\dagger}$ + BERT-rank | 63.21 | 61.58 | 64.93 |
| RR + Recall$^{\dagger}$ + BERT-rank | 63.70 | 58.78 | **69.54** |
| RR + Recall$^{\dagger}$ + BERT-rank$^{*}$ | **64.20** | 60.03 | 68.75 |

Table 5: "RR" represents the complete recognition and reconstruction module. "M3E"represents term recall based on (Wang Yuxin, 2023). "Recall$^{\dagger}$" represents term recall based on contrastive learning and online negative sampling. "BERT-rank$^{*}$" represents a BERT classification with guidance from component sequences.

dation set since the test set is not public. Obviously, in terms of performance, the trained model outperforms the fixed embedding model, which outperforms the traditional BM25 model. And the SBERT which uses the advanced online sampling strategy performs the best. What's more, our proposed terminology component recognition and reconstruction (RR) strategy brings a steady improvement over all kinds of recall methods.

This phenomenon proves our conjecture that the multi-implication problem causes the semantics of mentions to be blurred by multiple meanings. This leads to performance degradation when matching with a uni-implication term, and clear semantics improve performance when a multi-implication mention is decomposed into multiple atomic mentions recalled individually.

**For the entire normalization task**, regarding the baseline, we choose a fixed "M3E" as the recall model and different pre-trained models, e.g. BERT and its families, as the ranking model. First, we study the impact of different base models, we compare BERT (Devlin et al., 2018), Roberta (Liu et al., 2019), and Ernie (Sun et al., 2021). Then we compare the impact of "RR" in different task paradigms. In addition to the "Recall and Re-rank" paradigm in the upper part of Table 5, we investigate the generative paradigm presented by (Yan et al., 2020) in the middle of Table 5. Finally, we also compare state-of-the-art approach (Zhang et al., 2023) on the experimental dataset with our proposed approach and validate the impact of each module through some ablation studies. We used the official indicators provided by CBLUE (Zhang et al., 2021) to calculate the Micro-F1 score, Precision, and Recall as the evaluation metrics.

As shown in Table 5, it can be seen that every setting improved in the F1-score after using the "RR" strategy. The baseline method of "Recall and Re-rank" that showed considerable improvement achieved an increase of about 3% in the F1 score regardless of which base pre-trained model is used. Of the three pre-trained models, Ernie Sun et al. (2021) benefited from its training in incorporating the medical knowledge graph and therefore performed the best. The effect on the generative paradigm is the same. Additionally, with the same basic pre-trained model as the state-of-the-art normalization approach, our proposed approach also improves and achieves state-of-the-art.

Among all indicators, the improvement in recall is the most significant. Analyzing each module one by one, we find that both the online sampling strategy and the RR strategy help to improve the performance of recall, i.e., recalling as many terms as possible that are similar to the mentions. However, this can lead to the final ranking model or selection model having difficulty distinguishing between ground truth and hard negative, which is why the precision metric decreases. However, even if there may be a slight decrease in precision, it can be compensated by the recall rate to achieve

the effect of improving the F1 score. Moreover, we propose the knowledge-guided term selection module that leverages the attention mechanism to extract information about the components of both mentions and terms, balancing precision and recall.

## 5 Conclusion

In this paper, to address the hindrance caused by the multi-implication problem, we propose the Terminology Component Recognition and Reconstruction (RR) strategy based on the LLM. Furthermore, we propose a competitive terminology standardization framework that uses the benefits of the strategy and achieves state-of-the-art performance.

Specifically, we design the "Mention Decomposition" module to leverage an LLM to decompose a raw mention into multiple atomic mentions. The "Atomic Sampling" algorithm for atomic mentions is then designed, combining online negative sampling and LLM reasoning to obtain positive and negative samples of atomic mentions, and the recall model is trained using contrastive learning. Finally, use the term components as knowledge and use the attention mechanism to train a "Knowledge-Guided Term Selection" model. Experimentally we verified that the "RR" strategy mitigates the semantic ambiguity defects brought by the multi-implication problem and improves the performance of normalization.

## 6 Limitations

In this section, we focus on the limitations and risks of our proposed strategy and framework. In our Terminology Component Recognition and Reconstruction strategy, the LLM is used to obtain the composition of terms by inference. We eliminate the effects of randomness by setting a minimum temperature and a fixed random number seed but fully address the nuances introduced by the ChatGPT system. Therefore we limit the results obtained from large model inference to the scope of the CTN task through content and format constraints to avoid harmful information and hallucinations. Meanwhile, the component types we predefined are specific to disease diagnosis terms and do not apply to other terms. One possible idea is to design more generalized types by referring to syntactic structures in the clinical domain.

Another risk is that although we have designed meticulous approaches to perform the reconstruction of atomic mentions and the annotation of the positive samples of atomic mentions, the fact that it is an automated method of annotation without the intervention of an expert may lead to errors. However, these are harmless to the overall task as intermediate results.

## 7 Acknowledgments

Thank the anonymous reviewers for their excellent feedback. This work was supported by the National Key Research and Development Program of China (Grant 2023YFF1204904), the fund of the National Natural Science Foundation of China (No. 62306112), the Shanghai Sailing Program (No. 23YF1409400), and Shanghai Pilot Program for Basic Research (No. 22TQ1400100-20).

## References

Oliver Bodenreider, Ronald Cornet, and Daniel J Vreeman. 2018. Recent developments in clinical terminologies—snomed ct, loinc, and rxnorm. *Yearbook of medical informatics*, 27(01):129–139.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Ziqing Yang, Shijin Wang, and Guoping Hu. 2019. Pre-training with whole word masking for chinese bert. *arXiv preprint arXiv:1906.08101*.

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

Yi Ding, Fei Teng, Pan Zhang, Xianxu Huo, Qiao Sun, and Yan Qi. 2021. Research on text information mining technology of substation inspection based on improved jieba. In *2021 International Conference on Wireless Communications and Smart Grid (ICWCSG)*, pages 561–564. IEEE.

Rong Li Gai, Fei Gao, Li Ming Duan, Xiao Hui Sun, and Hong Zheng Li. 2014. Bidirectional maximal matching word segmentation algorithm with rules. In *Advanced materials research*, volume 926, pages 3368–3372. Trans Tech Publ.

Zongcheng Ji, Qiang Wei, and Hua Xu. 2020. Bert-based ranking for biomedical entity normalization. *AMIA Summits on Translational Science Proceedings*, 2020:269.

André Leal, Bruno Martins, and Francisco M Couto. 2015. Ulisboa: Recognition and normalization of medical concepts. In *proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 406–411.

Robert Leaman, Rezarta Islamaj Doğan, and Zhiyong Lu. 2013. Dnorm: disease name normalization with pairwise learning to rank. *Bioinformatics*, 29(22):2909–2917.

Ming Liang, Kui Xue, Qi Ye, and Tong Ruan. 2021. A combined recall and rank framework with online negative sampling for chinese procedure terminology normalization. *Bioinformatics*, 37(20):3610–3617.

Nut Limsopatham and Nigel Collier. 2016. Normalising medical concepts in social media texts by learning semantic representation. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 1014–1023.

Huidan Liu, Minghua Nuo, Wei-na Zhao, Jian Wu, and Yeping He. 2012. Segt: A practical tibetan word segmentation system. *Journal of Chinese information processing*, 26(1):97–103.

Yijia Liu, Bin Ji, Jie Yu, Yusong Tan, Jun Ma, and Qingbo Wu. 2020. An advanced icd-9 terminology standardization method based on bert and text similarity. In *The International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery*, pages 1868–1879. Springer.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Jinghao Niu, Yehui Yang, Siheng Zhang, Zhengya Sun, and Wensheng Zhang. 2019. Multi-task character-level attentional networks for medical concept normalization. *Neural Processing Letters*, 49(3):1239–1256.

OpenAI. 2023. New models and developer products announced at devday. Technical report.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Stefan Schulz, Robert Stegwee, and Catherine Chronaki. 2019. Standards in healthcare data. *Fundamentals of Clinical Data Science*, pages 19–36.

Xuhui Sui, Kehui Song, Baohang Zhou, Ying Zhang, and Xiaojie Yuan. 2022. A multi-task learning framework for chinese medical procedure entity normalization. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8337–8341. IEEE.

Yu Sun, Shuohuan Wang, Shikun Feng, Siyu Ding, Chao Pang, Junyuan Shang, Jiaxiang Liu, Xuyi Chen, Yanbin Zhao, Yuxiang Lu, et al. 2021. Ernie 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation. *arXiv preprint arXiv:2107.02137*.

He sicheng Wang Yuxin, Sun Qingxuan. 2023. M3e: Moka massive mixed embedding model.

Dongfang Xu, Zeyu Zhang, and Steven Bethard. 2020. A generate-and-rank framework with semantic type regularization for biomedical concept normalization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8452–8464.

Jinghui Yan, Yining Wang, Lu Xiang, Yu Zhou, and Chengqing Zong. 2020. A knowledge-driven generative model for multi-implication chinese medical procedure entity normalization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1490–1499.

Ningyu Zhang, Mosha Chen, Zhen Bi, Xiaozhuan Liang, Lei Li, Xin Shang, Kangping Yin, Chuanqi Tan, Jian Xu, Fei Huang, et al. 2021. Cblue: A chinese biomedical language understanding evaluation benchmark. *arXiv preprint arXiv:2106.08087*.

Yuhong Zhang, Kezhen Zhong, and Guozhen Liu. 2023. A novel method for medical semantic word sense disambiguation by using graph neural network. In *2023 9th International Symposium on System Security, Safety, and Reliability (ISSSR)*, pages 263–272. IEEE.

Hai Zhao, Changning Huang, and Mu Li. 2006. An improved chinese word segmentation system with conditional random field. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 162–165.

## A The Specific Prompts

Here are the specific contents of the prompts used in this paper, including the prompt for terminology component recognition, the prompt for knowledge enhancement from the standard termbase, and the prompt for atomic positive sampling, they are shown in Figure A1, Figure A2 and Figure A3.

The prompts for three types of tasks consist of two parts: instruction and demonstration. In the instruction part, we will introduce the definition and objective of the task, as well as the requirements for the output format to LLM. Specifically, for relatively complex tasks, we provide a wealth of background knowledge to help LLM understand. We also manually selected some typical demonstration examples to enhance the understanding.

**The prompt for terminology component recognition.** For the task of terminology component recognition, it is essential for the LLM to accurately identify components based on the types we have predefined. We begin by explicitly stating this objective in our task definition. Since there is a risk of the LLM misinterpreting these predefined component types, potentially causing errors or misrepresentations, we include a comprehensive description of each type within the instructions. To further aid understanding, we offer several examples showcasing the expected output categories and formats.

**The prompt for knowledge enhancement.** For knowledge enhancement, our objective is to identify standard terminology components and then use the LLM to generate a list of synonyms and abbreviations. We have consolidated these two tasks into a single prompt, instructing the LLM to directly supply synonyms and abbreviations for designated component types within the standard terminology. To minimize misunderstandings, we include clear definitions of each component type.

**The prompt for atomic positive sampling.** Atomic positive sampling is a straightforward task, for which we defined the process without requiring background knowledge. It is important to note that, in this prompt, the definitions of "mention" and "candidates" differ from those earlier in our article. As demonstrated, "mention" refers to a standard term that corresponds to the original mention, while "candidate list" consists of atomic mentions derived from the original mention. The aim is to ensure that each standard term is matched with a corresponding atomic term, thus preventing any omissions of standard terms. This prompt design helps to avoid confusion in the LLM.

(a) Chinese version

**user:**
你要完成一个组成成分识别任务。以最细粒度按照“发病内容”、“发病范围”、“操作内容”、“修饰词”、“分割词”、“无效内容”识别。
以下是各个识别类别的含义：
发病内容：可能存在的发病范围的症状、病变、状态等，以及“心脏病”、“高血压”等固定疾病名称。
发病范围：产生病变的解刨部位。
操作内容：放疗、化疗等治疗手段，或胃镜检查等检查手段。
修饰词：描述发病内容程度、性质的词。或者描述发病范围的方位词
分割词：“伴”、“并”等分割词、分隔符
无效内容：“待查”、“怀疑”等无意义描述，或检查结果、病因等非疾病、检查内容，或发病范围的方位、区域。

按照输入的内容逐个识别，只以JSON格式输出，例如
[
{"原词":"恶性","类别":"修饰词"},
{"原词":"卵巢","类别":"发病范围"},
{"原词":"癌","类别":"发病内容"},
{"原词":"化疗后","类别":"操作内容"},
{"原词":"，伴","类别":"分割词"},
{"原词":"尿频","类别":"发病内容"},
{"原词":"可能性大","类别":"无效内容"},
]
注意：必须按照输出的格式并且仅仅以JSON list形式输出，不要额外描述，不要缺少原词也不要增加额外的词语。
输入:{mention}
输出:

(b) English version

**user:**
You need to complete a Component Recognition task. Identify at the finest granularity according to "disease content", "disease scope", "operation content", "modifier", “separator", and "invalid content".
The following are the meanings of each recognition category:
Disease content: Symptoms, lesions, and conditions that may exist within the scope of the disease, as well as fixed disease names such as "heart disease" and "hypertension".
Disease scope: The site of dissection where the lesion occurs.
Operation content: Treatment methods such as radiotherapy and chemotherapy, or examination methods such as gastroscopy.
Modifier: a word that describes the degree and nature of the disease. Or directional words describing the scope of the disease.
Separator: "companion", "union" and other segmentation words, separators.
Invalid content: meaningless descriptions such as "to be investigated" and "suspected", or non disease or examination content such as examination results and causes, or the direction and area of the disease scope.

Identify each input one by one and output only in JSON format, for example
[
{"Original word": "malignant", "Category": "modifier"},
{"Original word": "Ovary", "Category": "Scope of incidence"},
{"Original word": "cancer", "category": "incidence content"},
{"Original word": "After chemotherapy", "Category": "Operation content"},
{"Original word": ", companion", "Category": "Segmented word"},
{"Original word": "Urinary frequency", "Category": "Disease content"},
{"Original word": "Possibility is high", "Category": "Invalid content"}
]
Attention: It is necessary to output in the format of the example and only in the form of a JSON list, without additional descriptions, missing original words, or adding additional words.
Input:{mention}
Output:

Figure A1: The specific prompt for terminology component recognition.

(a) Chinese version

**user:**
请你完成一个同义词映射工作。根据我给你的一个临床术语，只用JSON格式给出它所含疾病、部位、病因的同义词俗称、英文缩写。
所含疾病：去除该临床术语中的部位、病因等部分剩余的疾病内容，若不存在则为Null
部位：该临床术语发病的部位，若不存在则为Null\n"
病因：该临床术语发病的原因，若不存在则为Null\n"
例如：
输入：肺恶性肿瘤
输出
{
"所含疾病": {
"name":"恶性肿瘤",
"synonyms":["肿瘤","癌"],
"abbreviation": "MT"},
"部位": {
"name":"肺",
"synonyms":["肺部","上肺"],
"abbreviation": None},
"病因": None
}
注意：必须只输出JSON格式的结果，不要其他描述内容。
'"name"必须是所给临床术语中的一部分，"synonyms"的值必须是List类型，"abbreviation"给出最常见的缩写或Null。
输入：{q}

(b) English version

**user:**
Please complete a synonym mapping task. Based on a clinical term I gave you, provide synonyms and abbreviations for the "Diseases included", "Location", and "Etiology" in JSON format only.
Diseases included: Remove the remaining disease content from the clinical terminology, including the location, etiology, etc. If it does not exist, it is null
Location: The site of onset of this clinical term, if it does not exist, it is null
Etiology: The clinical term refers to the cause of the disease, which is null if it does not exist
For example:
Input: Lung Malignant Tumor
Output
{
" Diseases included ":{
"Name": "Malignant tumor",
"Synonyms": ["tumor", "cancer"],
"Abbreviation": "MT"},
Location:{
"Name": "Lung",
"Synonyms": ["lungs", "upper lungs"],
"Abbreviation": None},
"Etiology": None
}
Note: Only output results in JSON format and do not include any other descriptive content.
"Name" must be a part of the given clinical terminology, the value of "synonyms" must be of type List, and "abbreviation" provides the most common abbreviation or null.
Input: {q}

Figure A2: The specific prompt for knowledge enhancement to get synonyms and abbreviations of standard components.

(a) Chinese version

**user:**
请你完成一个同义词挑选工作。我会给你一个mention,以及一个JSON list格式的candidates列表。你需要在candidates列表的范围内，选择一个最适合mention的一个或多个同义词。
输出格式为一个JSON list，由最合适的一个或多个candidate为值，值不能为空\
示例：
mention：膝骨关节病
candidate_list：
[
"左膝退变[退行性病变]",
"左膝游离体[关节内游离体]“
],
输出
[
"左膝退变[退行性病变]“
]
注意：必须只输出JSON格式的结果，不要其他描述内容。
输出范围必须在candidates内，不能修改内容

mention：{mention}
candidates：{candidate_list}

(b) English version

**user:**
Please complete a synonym selection task. I will give you a mention and a list of candidates in JSON list format. You need to select one or more synonyms that are most suitable for the Mention within the scope of the Candidates list.
The output format is a JSON list, with the most suitable candidate or candidates as values, and the values cannot be empty
Example:
mention: Knee osteoarthritis
candidate_list:
[
"Left knee degeneration [degenerative disease]",
" Left knee free body [intra-articular free body]",
Output
[
“Left knee degeneration [degenerative disease]”
]
Note: Only output results in JSON format and do not include any other descriptive content.
The output range must be within candidates and cannot be modified

Mention: {mention}
Candidates: {candidate_list}

Figure A3: The specific prompt for atomic positive sampling.