

# A Knowledge-enhanced Two-stage Generative Framework for Medical Dialogue Information Extraction

Zefa Hu<sup>1,2†</sup>, Ziyi Ni<sup>1,2†</sup>, Jing Shi<sup>2</sup>, Shuang Xu<sup>2</sup> and Bo Xu<sup>1,2</sup>

<sup>1</sup>School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China.

<sup>2</sup>Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China.

<sup>†</sup>These authors contributed equally to this work.

## Abstract

This paper focuses on term-status pair extraction from medical dialogues (MD-TSPE), which is essential in diagnosis dialogue systems and the automatic scribe of electronic medical records (EMRs). In the past few years, works on MD-TSPE have attracted increasing research attention, especially after the remarkable progress made by generative methods. However, these generative methods output a whole sequence consisting of term-status pairs in one stage and ignore integrating prior knowledge, which demands a deeper understanding to model the relationship between terms and infer the status of each term. This paper presents a knowledge-enhanced two-stage generative framework (KTGF) to address the above challenges. Using task-specific prompts, we employ a single model to complete the MD-TSPE through two phases in a unified generative form: we generate all terms the first and then generate the status of each generated term. In this way, the relationship between terms can be learned more effectively from the sequence containing only terms in the first phase, and our designed knowledge-enhanced prompt in the second phase can leverage the category and status candidates of the generated term for status generation. Furthermore, our proposed special status “not mentioned” makes more terms available and enriches the training data in the second phase, which is critical in the low-resource setting. The experiments on the Chunyu and CMDDD datasets show that the proposed method achieves superior results compared to the state-of-the-art models in the full training and low-resource settings.

**Keywords:** Medical dialogue understanding, information extraction, text generation, knowledge-enhanced prompt, low-resource setting, data augmentation.

**Citation:** Z. Hu, Z. Ni, J. Shi, S. Xu, B. Xu. A knowledge-enhanced two-stage generative framework for medical dialogue information extraction. *Machine Intelligence Research*, vol.21, no.1, pp.153–168, 2024.  
<http://doi.org/10.1007/s11633-023-1461-5>.

**Published version:** <https://link.springer.com/journal/11633>.

**Email:** {huzefa2018, niziyi2021, shijing2014, shuang.xu, xubo}@ia.ac.cn

## 1 Introduction

Extracting terms and their statuses from medical dialogues has received increasing attention in the past few years [1–6]. The extracted information is beneficial to automatic electronic medical records (EMRs) generation [7], which reduces the burden of doctors to document EMRs [8–10]. In addition, information extraction is a backbone module of a typical diagnosis dialogue system, where the diagnosis is inferred from the dialogue history [11–14].

Notably, conversation-style medical data in complex speech patterns is challenging because various medical information, such as symptoms, surgeries, and tests, are scattered in a whole turn, even multiple turns. In addition, colloquial expressions of terms in medical dialogues vary from formal expressions, which further increases the task’s difficulty. As shown in Table 1, the expression of **Chest pain** is scattered in the overall sentence, and **Dyspnea** is mentioned in the dialogue through the synonymous phrase **short of breath**. Moreover, the status of each term may be changed as the conversation progresses. The word **No** indicates the **absent** of both **cardiopalmus** and **dyspnea**, which requires a good understanding of the dialogue. The status of **thyroid function test** changes from **suggest** to **done** according to the last turn.

Facing the above challenges, many works [7, 15–20] have been proposed, which can be generally grouped into two categories: classification-based methods and generative methods. One way [7, 20] of the classification-based methods takes each term candidate as the input to model the semantic interaction between the medical dialogue and the candidate, then determines the status of the term candidate by classification. However, this way treats each term independently, ignoring their relationship. Another way [15–17] is to decompose the task into multiple stages, including term detection, term normalization, and status inference. It first detects colloquial expressions of terms in dialogue, then maps the colloquial expressions to formal expressions and infers the status. However, term detection in the first stage requires token-level annotation. The generative methods [17–19] cast the task as a sequence generation problem regarding formal expressions of terms and status candidates as a target vocabulary and generating terms and the corresponding status sequentially. In

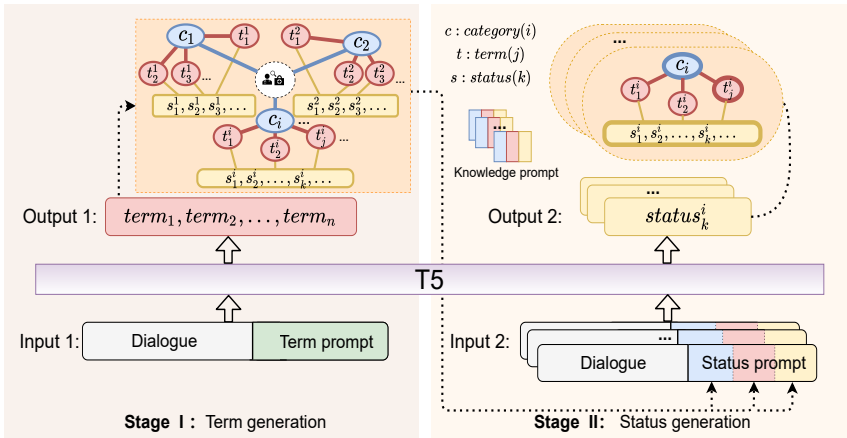
**Table 1:** An example of term-status pair extraction from medical dialogues.

Dialogue	
Patient:	我有房颤，怎么治疗？ I had atrial fibrillation, how to treat it?
Doctor:	有没有心慌、气短？ Do you feel palpitation or short of breath?
Patient:	没有，但胸口会不舒服，经常疼 No, but my chest is uncomfortable, always has bouts of pain
Doctor:	先完善一下甲状腺功能检查，正常的话，建议射频消融治疗 First, complete the thyroid function test. If it is normal, radiofrequency ablation is recommended.
Patient:	检查已经做过了。 The examination has been done.
Term-Status Pairs Label	
房颤:	阳性 (Atrial fibrillation: appear)
心慌:	阴性 (Cardiopalmsus: absent)
呼吸困难:	阴性 (Dyspnea: absent)
胸痛:	阳性 (Chest pain: appear)
射频消融:	医生建议 (Radiofrequency ablation: suggest)
甲状腺功能:	患者已做 (Thyroid function test: done)

this way, generative methods need not explicitly detect terms. Consequently, it requires only the final formal label rather than token-level BIO (begin-in-out) annotations. At the same time, sequence generation can model the relationship between the terms.

Although existing generative methods have many advantages compared with other methods, there are still some limitations: (1) Complex relationship modelling among terms. The output sequence of these generative methods consists of terms and status. If the model explores the association between terms through this sequence, it must understand the corresponding status simultaneously. (2) Prior schema information is neglected. These generative methods output the sequence based only on the dialogue history. The schema information for status generation can only be learned through mapping from medical dialogue to the output sequence. In fact, some status candidates are predefined in medical datasets that could guide the model to generate status but are ignored. (3) Lack of training data in the low-resource setting. The training data of these generative methods must be fully annotated (including terms and status). These methods preclude the model from using the existing data that only has the term annotation, which is unfriendly to the low-resource setting.

In this paper, we propose a novel knowledge-enhanced two-stage generative framework (KTGF). The overview of KTGF is shown in Figure 1. In our framework, we complete term-status pair extraction from medical dialogues (MD-TSPE) through two phases in a unified form of the sequence to sequence generation: we generate all terms first and then generate the status of each generated term. We also plug different task-specific prompts after the dialogue context and take them together as the model input. In this way, the generation of terms and their status are decoupled, leading to greater flexibility of the framework that introduces at least three advantages: (1) The sequence contains



**Figure 1:** Overview of KTGF.  $t_j^i$  and  $s_k^i$  indicate the  $j$ -th term and the  $k$ -th status in the  $i$ -th category, respectively. KTGF takes T5 as the backbone and generates terms and their status in two stages. In each stage, KTGF concatenates medical dialogue with a subtask prompt as input. In the second stage, KTGF uses the terms generated in the first stage to retrieve prior task knowledge to enhance the prompt, which enables the model to generate status effectively.

only terms in the first phase, which enables the model to learn the relationship between terms more effectively without considering the status. (2) The prior schema information can be fully utilized. We integrate the status candidates and the category of generated terms defined in the schema into the prompt of the second phase and name it the knowledge-enhanced prompt. It allows task-aware contextualization and can guide the model to generate status more effectively. (3) As terms and their statuses are generated separately, the framework can learn from data that only has term annotations in the first phase, which can partially alleviate the data sparsity problem. In addition, we design another status, “not mentioned”, to further enrich the data and enhance the training of the second phase. We evaluate KTGF on two medical dialogue datasets, Chunyu [7] and CMDD [15]. Comparisons against previous state-of-the-art methods show that KTGF performs best in both the full training and the low-resource setting.

Our main contributions can be summarized as follows<sup>1</sup>:

- We propose a novel knowledge-enhanced two-stage generative framework (KTGF) to explore the term-status pair extraction from medical dialogues(MD-TSPE). Our framework only needs to consider the relationship between terms in the first phase. Then it generates the corresponding

<sup>1</sup> Our codes are available at <https://github.com/FlyingCat-fa/KTGF>.

status from the dialogue in the second phase with the help of prior schema knowledge in the prompt.

- KTGF can utilize data with only term annotation, which is critical in the low-resource setting. The extra-designed status further enriches the status-related data to alleviate the data sparsity problem.
- We evaluate KTGF on Chunyu [7] and CMDD [15]. Our approach achieves new state-of-the-art results on both datasets. Ablation studies and further analysis demonstrate the advantages of our two-stage generation strategy and prior knowledge utilization.

The remainder of this paper is organized as follows. Section 2 presents previous works related to medical information extraction, prompting language models, and training on supplementary data. Section 3 introduces the problem definition and proposes our KTGF. Section 4 describes the datasets, compared baselines, experimental settings, and results. In Section 5, further analysis and a discussion are presented. Finally, the conclusion is shown in Section 6.

## 2 Related Work

### 2.1 Medical Information Extraction

Medical Information Extraction has great potential in the fields of biomedicine and NLP. One of the classic tasks is the *i2b2 challenge*. It constructs a small corpus of written discharge summaries and designs a variety of tasks to extract different kinds of information [21]. Many efforts have also focused on extracting entities and relations from medical texts [22, 23], which is also taken as one of the tasks for Chinese Biomedical Language Understanding Evaluation (CBLUE) [24]. In contrast, extracting information directly from medical dialogues has been an emerging research field in recent years [4, 25, 26], focusing on colloquialism and multi-turn dialogue interaction. Given the patient-doctor conversation, medical information, including clinical terms [5, 27] and their properties [16, 17], or treatment regimens [28] can be extracted and used to generate electronic medical records (EMRs) [2, 29], reducing the burden on doctors of creating narrative reports [8–10]. Information extraction from medical dialogues is also a backbone module of the typical diagnosis system [11–14], in which the status information of medical terms plays a key role.

Among the works on medical dialogue information extraction, [2] extracts information from dialogue through a pipeline consisting of various heuristics such as character matching, regular expressions, and other task-specific heuristics. [29] proposed generating EMRs from an automatic speech recognition (ASR) directly, but it achieved poor performance. Recently, classification-based methods have been proposed, including multi-stage methods [15–17] and matching-based classification methods [7, 20]. The multi-stage methods identify the term by a sequence tagging model in the first stage and then conduct term normalization and status inference by two multi-layer perception classifiers. However, these methods require token-level redundant annotation in the first

stage. In the matching-based classification methods, [7] takes each term, corresponding category, and status candidates as input to match the semantics with the medical dialogue, and then designs an aggregate module to consider the utterance interaction. SAFE [20] also takes each term candidate as input. The method develops a multi-task learning method to model the speaker's identity. It further proposes a co-attention fusion module with graph networks to match the semantics between the medical dialogue and the term. However, these matching-based classification methods do not consider the relationship between terms. The generative method is considered an excellent way to address the problems in classification methods. The generative methods [17–19] regard formal medical terms and status as the target vocabulary and generate a single sequence containing terms and their status, which gain better performance compared with classification-based methods. Based on these generative methods, we aim to further enhance the modeling of relationships among terms and fully use prior schema knowledge.

## 2.2 Prompting Language Models

Combining the prompt with pre-trained language models is an active line of research [30]. [31] uses prompts for generation in the zero-shot setting and obtains excellent performance. [32] designed task-specific prompts for training and testing to learn multiple tasks simultaneously. Many works also focus on automatically optimizing discrete prompts [33, 34] or directly using learnable continuous prompts [35, 36]. In addition, to prompt engineering, some works also use available prior task information or knowledge as task-specific prompts [37–40]. Our proposed prompting method inspired by these considerations considers both multi-task and knowledge prompts. Moreover, we further improve the prompt to enrich training data in the low-resource setting.

## 2.3 Training on Supplementary Data

Previous works have shown that the performance of the target task can be improved through supplementary training on tasks with intermediate-labelled data [41–43]. [41] and [42] focus on GLUE natural language understanding benchmark [44], and [43] improves the task-oriented dialogue system through an augmented training phase with intermediate-labelled data. In this work, we aim to boost the performance of term-status pair extraction from medical dialogues through more supplementary data, including not only the intermediate-labelled data, but also the additional data available owing to our designed special status.

# 3 Methodology

In this section, we will define the term-status pair extraction from medical dialogues (MD-TSPE) task and then introduce our approach, including an overview of the approach, the two-stage generative framework, prompt design with prior schema knowledge, and training strategies.

### 3.1 Problem Definition

**Medical Dialogue  $D$ :** The medical dialogue is the raw text containing multiple sentences alternated between the patient and the doctor, namely  $D = (l_1, l_2, \dots, l_n)$ , which consists of a sequence of word tokens  $(w_1, w_2, \dots, w_m)$ .

**Term  $T$ :** The term set  $T$  is predefined according to medical knowledge graphs. Each term  $t_i$  can be mentioned in the medical dialogues in different ways, such as formal expression, spoken word expression, and the expression scattered in the whole sentence or even across multiple sentences. Each term belongs to a predefined category in the dataset.

**Status  $S$ :** The status is essential because it specifies more detailed information about the term, and status candidates depend on each term's category. Furthermore, the status of each term is changeable during the whole medical dialogue.

Given Medical Dialogue  $D$ , the objective of MD-TSPE is to extract a set of term-status pairs  $\{\dots, (t_i, s_i), \dots\}$ , where  $t_i \in T$  is the term mentioned in the dialogue, and  $s_i \in S$  is the corresponding status of  $t_i$ .

### 3.2 Overview of Our Approach

Previous generative methods on MD-TSPE cast the task as a sequence generation problem in one stage. In this case, the medical dialogue  $D$  is taken as input to the model, and term-status pairs  $\{\dots, (t_i, s_i), \dots\}$  are generated sequentially. This approach has been adopted in many works, such as LSTM-based or transformer-based models [17, 19]. However, generating a single output sequence consisting of both terms and their status is more likely to hardly model the relationship among terms and suffers optimization issues with decoding long and complex sequences, which results in poor performance. [18] leverages contrastive learning to filter out the unfaithful term-status pairs after generating complex and long output sequences. However, this method also needs to generate long and complex sequences, resulting in missing some term-status pairs. Even so, considering the common use of the one-stage generation approach, we first include this basic framework as our preliminary experiments.

Then, we design a knowledge-enhanced two-stage generative framework (KTGF) for MD-TSPE, of which each stage has its corresponding task. In the first stage, we aim to generate a simplified sequence containing only the terms to model the relationship among them and then generate the corresponding status of each generated term in the second stage. Thus, the complex sequence in the one stage generation approach is disintegrated into multiple sequences. To make the form of our framework unified in two stages, we plug different task-specific prompts after the dialogue context and take them together as the model input. In addition, prior knowledge can enrich the task-specific prompt in the status generation phase and further benefit generative methods. Moreover, we design a special status that improves the status prompt to expand the status-related training data in the low-resource setting.

### 3.3 A Two-stage Generative Framework for MD-TSPE

Given the medical dialogue, we concatenate each utterance into a word token sequence  $D = (w_1, w_2, \dots, w_m)$ . The basic idea is to encode the medical dialogue  $D$ , obtain its contextualized representations, and generate the medical terms and their status in two stages. Notably, the generation in two stages can be fulfilled in a unified generative model. In the second stage, each generated term in the first stage needs to be appended to medical dialogue  $D$ . However, the generation of two stages is actually different subtasks. To address this issue, we adopt the prompting approach, where elaborately designed prompt tokens are concatenated with the original medical dialogue. The knowledge of generated terms is incorporated into the prompt of the second stage. It has been confirmed that prompting is an adequate paradigm to leverage the prior information for specific tasks [37, 38], or the knowledge of pre-trained language models (PLMs) to solve various tasks [33, 45]. After incorporating the prompts, the original medical dialogue  $D$  can be extended to a task-specific sequence, denoted as  $\tilde{D}$ :

$$\tilde{D} = \underbrace{w_1, \dots, w_m}_{\text{word tokens}}, \underbrace{p_1, \dots, p_n}_{\text{prompt tokens}}. \quad (1)$$

Given the task-specific sequence  $\tilde{D}$  as the input, we add a special token [SOS] to represent the beginning of each target sequence. Furthermore, a special end-of-sequence [SEP] is also appended to the end of the target sequence. The target sequence of the first stage consists of all terms mentioned in the medical dialogue  $D$ , and we use a special token “,” to separate different terms. The status of the generated term that can be found from the prompt is the target sequence of the second stage.

We employ an encoder-decoder transformer architecture T5 [32] as the backbone of our framework. This framework first embeds the input sequence as the input vector representation:

$$H^0 = \text{Emb}(S), \quad (2)$$

where  $S$  denotes the embedded sequence. Based on the input representation  $H^0$ , both the transformer encoder and decoder employ  $L$  stacked transformer blocks, and  $l$ -th block learns a new sequence representation  $H^l = H_1^l, \dots, H_n^l$  from  $H^{l-1} = H_1^{l-1}, \dots, H_n^{l-1}$ . Each layer of the encoder has two sub-layers. The first is a multi-head self-attention mechanism, and the second is a small position-wise fully connected feed-forward network. Layer normalization [46] is applied to the input of each sub-layer, and the applied layer normalization is a simplified version where no additive bias is utilized. Following layer normalization, a residual skip connection adds each sub-layer’s input to its output. In addition to the applied two sub-layers in each encoder layer, the decoder layer inserts a third sub-layer, which has the same structure as the first sub-layer and performs attention over the output of the encoder. The following are the details of each sub-layer.



The first sub-layer is multi-head self-attention, which computes independent self-attention representations with multiple individual heads. For each head, the input consists of queries  $Q$  and keys  $K$  of dimension  $d_k$ , values  $V$  of dimension  $d_v$ . The output is a weighted sum of values, and the weights are calculated by the scaled dot-product between queries  $Q$  and keys  $K$ . Since the transformer mainly relies on the attention mechanism without recurrence, the transformer adds a scalar as the position embedding parameter to the corresponding logit for computing the attention weights. The learnable position embedding is assigned based on the relative positions between queries  $Q$  and keys  $K$  and shared across all layers in the model. The calculation of a self-attention head is shown as follows:

$$\text{Attention}(Q, K, V) = \text{softmax} \left( \frac{\mathbf{QK}^\top}{\sqrt{d_k}} + \mathbf{M} + \mathbf{PE} \right) \mathbf{V}, \quad (3)$$

$$\mathbf{M}_{ij} = \begin{cases} 0, & \text{visible} \\ -\infty, & \text{invisible} \end{cases} \quad (4)$$

where  $d_k$  is the dimension of the key.  $\mathbf{M}, \mathbf{PE} \in \mathbb{R}^{n \times n}$  are an attention mask and the position embedding matrix, and  $n$  is the sequence length. The attention mask  $\mathbf{M}$  can make a key invisible or no contribution to a query by setting the attention score  $-\infty$ .

Multi-head attention allows the model to compute multiple subspace representations in parallel and concatenate them.

$$\text{head}_i = \text{Attention}(H^{l-1}W_i^Q, H^{l-1}W_i^K, H^{l-1}W_i^V), \quad (5)$$

$$\text{MHAttention} = \text{ConCat}(\text{head}_1, \dots, \text{head}_h)W^O, \quad (6)$$

where  $H^{l-1}$  and  $h$  are the input of the  $l$ -th block and the number of attention heads, respectively.  $W_i^Q, W_i^K, W_i^V$  and  $W^O$  are learnable parameters.

The second sub-layer is a simple position-wise fully connected feed-forward network. Following layer normalization, a residual skip connection adds the sub-layers' input to their output.

$$O^l = H^{l-1} + \text{LayerNorm}(\text{MHAttention}(H^{l-1})), \quad (7)$$

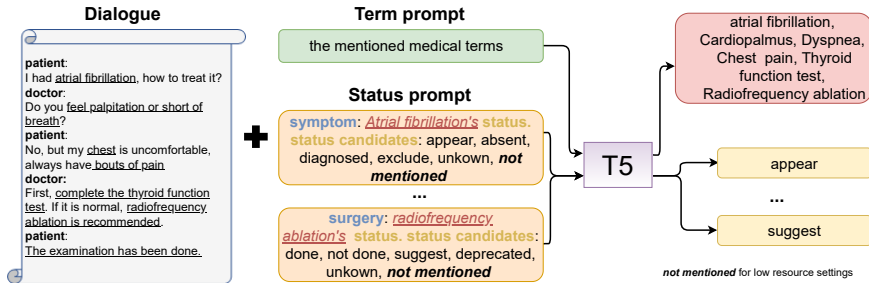
$$H^l = O^l + \text{LayerNorm}(\text{FFN}(O^l)), \quad (8)$$

where FFN is the feed-forward layer:

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2, \quad (9)$$

where  $W_1, W_2, b_1$  and  $b_2$  are trainable model parameters.

Based on the transformer-based encoder-decoder architecture, T5 designs an unsupervised objective that randomly samples and then drops out the sampled tokens in the input sequence. A single sentinel special token replaces all



**Figure 2:** The design of our prompt. In the term generation stage, the prompt is employed for a better understanding of the term generation subtask. In the status generation stage, we use the generated term to obtain the category and status candidates, which are utilized to enhance the prompt. Moreover, we add a special status, "not mentioned", for the low-resource setting. Therefore, the term not mentioned in the dialogue can also have its corresponding status, which augments the status-related training data. The medical dialogue is from Table 1.

consecutive spans of dropped-out tokens. The target is to recover all corresponding dropped-out spans of tokens delimited by the same sentinel special tokens used in the input sequence.

Taking the pre-trained T5 as the backbone, we further train our model to generate the term and status sequences in two stages. Each training sample is represented as follows:

$$d = (\tilde{D}_t, y), \quad (10)$$

where  $t \in \{\text{term generation, status generation}\}$ , which denotes the subtask that sample  $d$  belongs to.  $\tilde{D}_t$  is the subtask input that consists of the medical dialogue and the subtask prompt.  $y$  denotes the subtask output text. We train the model with the maximum likelihood objective. Given the training sample  $d = (\tilde{D}_t, y)$ , the objective  $\mathcal{L}_\Theta$  is defined as:

$$\mathcal{L}_\Theta = - \sum_{i=1}^y \log P_\Theta(y_i | y_{<i}; \tilde{D}_t), \quad (11)$$

where  $\Theta$  is the model parameters.

### 3.4 Knowledge-enhanced Prompt

Our framework uses the same T5 to generate terms and status in two stages, which requires the model to learn two different subtasks. Therefore, we design task-specific prompts to make better use of the knowledge of the pre-trained T5 and jointly learn the two subtasks. We further incorporate the prior schema information of each generated term as additional prompt tokens for status generation. This paper further considers the low-resource setting. In reality, data

with term and status annotation is usually more difficult to obtain. Compared with the existing generative methods, our model has the advantage of being able to learn from other data containing only term annotation in the first stage. Moreover, we provide a status-enhancing method that improves the status prompt to expand the status-related training data in the low-resource setting. Next, we describe the specific prompting designs of the two stages in detail.

**Prompt for term generation.** Term generation aims to generate all terms mentioned in the medical dialogue. The prompting design of term generation mainly enhances the textual semantics for better medical dialogue understanding and term generation. Here we set “the mentioned medical terms” as the prompt and add it after the medical dialogue.

**Prompt for status generation.** After obtaining the generated term, we further generate the corresponding status of each generated term in the status generation stage. The prompt for status generation is designed to enhance the term semantics and predict status more accurately. For each generated term, we retrieve the prior schema information, including the category and status candidates of the term. The information is leveraged to construct a knowledge-enhanced status prompt, which is formally denoted as: “**Category: term’s status. Status candidates: candidate 1, candidate 2, etc ...**.” The knowledge-enhanced prompt advances task-aware contextualization and guides the model to generate status more effectively.

Moreover, we improve the knowledge-enhanced prompt for status generation in the low-resource setting. Specifically, we add a special status, “not mentioned”, to the status candidates. In this way, many terms not mentioned in a medical dialogue can match the special status “not mentioned” as a term-status pair. Therefore, the training data of status generation increases, and the model can learn how to generate the correct status according to the status candidates in the low-resource setting. The design of term and status prompts is also shown in Figure 2.

### 3.5 Training Strategies

We employ multi-task learning approach to train a single unified model in the two-stage generative framework and use a mini-batch based optimization approach. We adopt different prompting strategies for full-data and low-resource settings. In the full-data setting, we adopt the knowledge-enhanced prompt without additional special status. In the low-resource setting, we improve the knowledge-enhanced prompt by introducing a special status as described in Section 3.4. KTGF is trained in two stages in the low-resource setting. The model is first trained on a mixed dataset consisting of the training set and external corpora containing only term annotations, and then further fine-tuned on the training set for better performance. In this way, the partially annotated data containing only terms can also be leveraged in the low-resource setting. More experimental details are introduced in Section 4.3.

Dataset	Domain	Dialogue	Window	Term	Status
Chunyu	Cardiology	1,120	18212	71	18
CMDD	Pediatrics	2,067	87005	149	3

**Table 2:** The statistics of Chunyu dataset and CMDD dataset.

## 4 Experiments

### 4.1 Datasets and Metrics

We evaluate our proposed framework on two benchmark datasets: Chunyu [7] and CMDD [15], both of which are Chinese medical dialogue datasets for entity-status pair extraction.

Chunyu contains 1,120 dialogues on cardiovascular diseases. A schema is defined in the dataset, which contains four main categories: symptom, surgery, test, and other info. Each category includes medical terms and possible statuses. The original Chunyu dataset provides two settings for the status, including the coarse-grained status independent of categories and the fine-grained status dependent on categories [7]. This work uses the fine-grained status better to express the status semantics of terms from different categories. For example, surgery includes the term, radiofrequency ablation, and its status candidates consist of done, not done, suggest, deprecated, and unknown. The schema has 71 predefined terms and 18 predefined statuses. Chunyu is annotated by a window-to-information approach with the mentioned terms and corresponding status, which focuses on the dialogue context in the window and does not need word-level sequence labelling information. Specifically, the dialogues in the dataset are divided into pieces using a sliding window. Each window consists of 5 turns of the dialogue. The sliding step is set to 1. In this way, the dataset obtains 18212 annotated windows.

CMDD has 2,067 annotated dialogues on four pediatric diseases, which contain 149 predefined terms and 3 predefined statuses. All of these terms belong to the symptom category, and the status candidates are True, False, and Uncertain. Each term-related character in the dialogue turn is annotated by the word-level begin-in-out (BIO) label. In the original data, multiple adjacent sentences belonging to the same speaker are considered multiple turns. Following [19], we merged these adjacent sentences from one speaker into a single dialogue turn. We also split the medical dialogues into windows and kept the window size and sliding step the same as Chunyu.

We followed [19] to further process the status annotation in the two datasets. In the original annotation of the Chunyu dataset, one window contained several statuses for the same term. Compared to the historical status in medical dialogues, the latest status of each term is more informative. We only preserve the latest status, which is more suitable for the medical dialogue system in the experimental setting [19]. In fact, the historical status of each term in the current window can be obtained through the latest status of the term in the historical window, so there is no historical status information loss. In the CMDD

dataset, the repetitive medical term-status pairs in one window are removed. Table 2 summarizes the statistics of the Chunyu dataset and CMDD dataset.

Following [7], we use the micro-average precision, recall, and F1-score as the evaluation metrics. We report the results of extracted medical information in term and full evaluations. Term evaluation only considers the correctness of the terms, regardless of their status. Full evaluation means that both terms and the corresponding status must be strictly correct. The evaluation results for both window-level and dialogue-level on the test set are reported. In line with [7], we merge the results of windows within the same dialogue for the dialogue-level evaluation and update the previous labels with the latest ones. Notably, not all medical windows have golden labels; sometimes, the windows are not associated with medical terms in the dataset schema. In this case, the empty prediction is regarded as correct, and then the precision, recall, and F1 are set to 1; otherwise, they are 0.

## 4.2 Baseline Models

We compared our proposed method KTGF against classification-based and generative models for MD-TSPE.

**SAT** [17]: a multi-stage classification-based method. It identifies the term spans by a sequence tagging model trained with token-level annotation in the first stage. Then, it employs two multi-layer perception classifiers to normalize the terms and infer their status from the hidden representation of identified term spans.

**BERT-SAT**: a multi-stage classification-based method similar to SAT, except that the encoder is replaced by BERT [47].

**MIE** [7]: a multi-label classification model that obtains a term-specific representation and a status-specific representation for each utterance through an attention mechanism to determine whether the term-status pair belongs to the dialogue. MIE also models the multi-turn interaction simultaneously.

**Transformer**: an encoder-decoder based generative method that treats the medical dialogue as the input and outputs a sequence containing the terms and their status.

**MGT** [19]: a multi-granularity transformer model that models MD-TSPE similar to Transformer. However, unlike Transformer, MGT can simultaneously capture the role-enhanced interaction across turns and integrate mixed granularity representations to model the dialogue context fully.

**CGT** [18]: a generative model based on UniLM [48] that combines Transformer with contrastive learning to generate medical terms and their status in one stage.

**T5**: a generative method based on pre-trained T5 [32], the same backbone as our KTGF, but it generates all outputs in one stage, including medical terms and their status.

### 4.3 Implementation Details

We obtain experimental results of MIE based on the released codes. For the baseline models, including SAT, BERT-SAT, Transformer, and MGT, we adopt the same settings as [19] and cite the results of these models from the paper. In these models, the encoder of Transformer and MGT are all pre-trained with masked language modelling objective (MLM) [47] on the mix of Chunyu and CMDD datasets. Considering that CGT does not release the source code, we only report the full results on the Chunyu dataset cited from the original paper [18].

We report the results of T5 and KTGF with two backbones: the Chinese T5-small and T5-base [49]. The models are the base versions by default if not explicitly specified. We implemented T5 and KTGF based on HuggingFace’s Transformers library [50] and conducted experiments using an Nvidia RTX3090 GPU. AdamW [51] is employed as the optimizer, and the weight decay is set to 0.01. We set the initial learning rate to  $2e - 5$ , and the warm up step is 1000. The batch size is 32, and the total number of epochs are 100 and 300 for KTGF and T5, respectively. We adopt the greedy search for generation and select the checkpoints performing best on the valid set for test evaluation. In the low-resource setting, to augment the training data in the term generation phase, we collect two annotated corpora that only contain term annotation, including MSL [5] and MedDG [6]. In total, there are over 0.24M examples.

### 4.4 Full Data Evaluation

**Table 3:** Window-level evaluation results on the Chunyu and CMDD datasets.

Model	Chunyu						CMDD					
	Term			Full			Term			Full		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
MIE	90.36	87.58	80.7	70.32	66.47	66.81	90.85	86.48	87.71	81.76	73.7	79.58
SAT	-	-	-	-	-	-	88.5	87.8	87.2	75.1	75.3	74.2
BERT-SAT	-	-	-	-	-	-	90.7	89.6	89.2	75.2	75.2	74.2
Transformer	92.2	90.3	90.3	69.3	68.2	68	90.1	88.5	88.7	74.3	72	72.4
MGT	93.8	88.6	90.1	75.3	71.7	72.7	92.8	90.2	90.7	80	76.8	77.6
CGT	-	-	-	80.53	78.83	79.42	-	-	-	-	-	-
T5 small	93.15	91.73	91.5	74.77	74.28	73.78	93.02	94.34	93.21	83.28	84.34	83.41
T5 base	92.19	92.64	91.49	75.49	76.26	75.11	93.28	94.28	93.31	83.58	84.36	83.57
<b>KTGF small</b>	99.07	95.15	96.65	77.42	74.53	75.63	<b>99.32</b>	<b>98.16</b>	<b>98.59</b>	88.35	87.4	87.75
<b>KTGF base</b>	99.11	<b>95.72</b>	<b>97.05</b>	<b>84.31</b>	<b>81.44</b>	<b>82.55</b>	98.95	97.47	98.01	<b>88.85</b>	<b>87.58</b>	<b>88.05</b>
<b>w/o status</b>	<b>99.19</b>	95.2	96.75	80.94	77.85	79.05	99.03	97.48	98.06	88.62	87.33	87.81
<b>w/o category</b>	99.03	95.57	96.91	84.18	81.34	82.44	-	-	-	-	-	-
<b>w/o knowledge</b>	99.11	95.25	96.72	80.68	77.68	78.82	-	-	-	-	-	-

Table 3 and Table 4 summarize the results for window-level and dialog-level on two experimental medical datasets, respectively, as described in Section 4.1. The SAT method can not be evaluated on the Chunyu dataset because it needs token-level annotation. The window-level results in Table 3 shows that our

**Table 4:** Dialogue-level evaluation results on the Chunyu and CMDD datasets.

Model	Chunyu						CMDD					
	Term			Full			Term			Full		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
MIE	94.36	87.58	89.53	72.02	66.21	67.83	93.64	84.89	87.99	77.15	70.09	72.61
T5 small	90.1	93.37	90.78	72.69	77.71	74.05	87.73	96.46	91.16	71.09	77.77	73.7
T5 base	89.4	95.46	91.29	72.67	80.12	75.16	88.42	96.48	91.52	72.69	78.83	75.04
<b>KTGF small</b>	100	95.38	97.32	77.35	77.61	76.86	<b>100</b>	<b>98.48</b>	<b>99.13</b>	80.45	79.21	79.74
<b>KTGF base</b>	<b>100</b>	<b>97.52</b>	<b>98.57</b>	85.63	<b>84.53</b>	<b>84.71</b>	100	97.65	98.67	<b>82.46</b>	<b>80.47</b>	<b>81.33</b>
<b>w/o status</b>	100	95.99	97.69	82.68	82.21	81.98	100	97.74	98.72	80.95	79.13	79.92
<b>w/o category</b>	100	96.81	98.14	<b>86.14</b>	83.87	84.62	-	-	-	-	-	-
<b>w/o knowledge</b>	100	95.79	97.57	84.08	81.98	82.55	-	-	-	-	-	-

KTGF base achieves an F1-score of 82.55 on the Chunyu dataset and 88.05 on the CMDD dataset in the full training setting, which gains significant improvement for the complex medical conversation compared to existing methods. Specifically, on the Chunyu dataset, KTGF outperforms CGT, the previous best result, by a large margin of 3.13. On the CMDD dataset, our model outperforms the MGT model by 10.45. The improvement demonstrates the power of the two-stage generation with the knowledge-enhanced prompt. The dialogue-level evaluation results are shown in Table 4, further demonstrate the excellent performance of our proposed method. On the Chunyu dataset, the dialogue-level evaluation typically outperforms the window-level evaluation, while the opposite is observed for the CMDD dataset. We attribute this difference to the distinct distributions of the two datasets. In the Chunyu dataset, each dialogue has fewer rounds (16.26 on average), and errors in extracting medical terms or the corresponding status based on the current window can be corrected by subsequent windows with more information, leading to higher performance at the dialogue-level evaluation. Conversely, in the CMDD dataset, each dialogue has more rounds (42.09 on average), and many dialogue windows do not mention any medical terms. The models' accurate predictions on these empty windows may result in an overestimation of its performance at the window-level evaluation, while the dialogue-level evaluation can avoid this situation. Considering that many baselines only provide window-level evaluation results, we compare our model with them based on the window-level evaluation results by default if not explicitly specified.

*Comparison with classification-based methods:* The SAT model is a classic baseline. The BERT-SAT model employs BERT as the encoder for better contextual representation, which improves the performance of the SAT model on term evaluation but not on full evaluation. This phenomenon illustrates that only incorporating general contextual representation does not assist the model in understanding the complex dialogue interaction, which is critical for status inference. In our KTGF model, prior information is incorporated into KTGF, including category and status candidates. Therefore, the KTGF model can more effectively infer the status of each term from the complex dialogue

interaction. The MIE model also takes prior information into consideration. Specifically, to extract term-status pairs from medical dialogue, the MIE model incorporates term candidates and status candidates as part of the input and obtains a term-specific representation and a status-specific representation for each utterance through an attention mechanism. However, the MIE model performs poorly on term evaluation. For example, the gap in the F1-score on the full evaluation between the MIE model and MGT is 5.89 on the Chunyu dataset, and the gap in the term evaluation between them is enlarged to 9.4. It demonstrates that the term-specific representation hardly carries out the colloquial expression of medical terms, and the representation also ignores the correlation between terms. In our KTGF model, the terms are generated sequentially, which models the relationship between terms naturally.

*Comparison with generative methods:* From Table 3 we can see that our KTGF model demonstrates better performance than other generative models on term evaluation and further gains a larger margin on full evaluation. For example, using the same backbone model, the KTGF model improves the F1 score by 5.56 compared with T5 (97.05 vs. 91.49) on term evaluation and gains a larger margin by 7.44 (82.55 vs. 75.11) on full evaluation. The huge success is attributed to our two-stage generation framework and the knowledge-enhanced prompt. Based on the two-stage generation framework, we simplify the output sequence to only contain the terms in the first stage, which makes it more effective to learn the relationship among terms and gains better results on term evaluation. Moreover, the two-stage framework integrates prior knowledge more flexibly. With the proposed knowledge-enhanced prompt, the framework utilizes the category and status candidates in the second stage to better understand the status subtask and gains a larger margin on full evaluation.

## 4.5 Low-Resource Evaluation

**Table 5:** Low-resource evaluation on the Chunyu dataset. The F1-scores of both term and full evaluations are shown.

Model	1%		5%		10%	
	Term	Full	Term	Full	Term	Full
MIE	27.74	19.25	43.74	32.35	68.28	51.12
T5 base	61.39	35.72	77.08	47.27	82.37	57.35
<b>KTGF base</b>	73.02	43.96	84.8	52.41	90.1	64.26
<b>w/ term data</b>	78.79	44.96	<b>88.19</b>	55.72	91.87	65.7
<b>w/ both</b>	<b>79.27</b>	<b>45.24</b>	87.84	<b>56.27</b>	<b>94.87</b>	<b>67.81</b>

Owing to the two-stage generation, our KTGF model can learn from the data that only have term annotation in the first phase. Moreover, we designed the special status “not mentioned” to enrich the training data in the second phase. To investigate the generalization ability of KTGF equipped with these



data augmentation methods, we evaluate it in a more challenging low-resource scenario. In the low-resource scenario, only a small percentage of training data is used. Here we select three different percentages as 1%, 5%, and 10%. Referring to the original dataset division for training, validation, and test sets based on the whole dialogue [7], we directly sample the whole dialogue. All the windows in these sampled dialogues are used as the low-resource training set. The trained model is then evaluated with the original test set. We compare KTGF with the classification-based model MIE and the generative model T5 on the Chunyu dataset.

The results are shown in Table 5. As seen, KTGF consistently outperforms for all baseline models by a large margin. Notably, the performance gain of KTGF is even the most significant when 1% training samples are used. For example, compared with the T5 model, our KTGF model improves the full F1-score by 6.91 (64.26 for KTGF vs. 57.35 for T5) when trained with 10% samples and improves it by a larger margin of 8.24 (43.96 for KTGF vs. 35.72 for T5) when trained with 1% samples. The results demonstrate that our model can model the MD-TSPE task more efficiently through two-stage generation and knowledge-enhanced prompt in an extremely low-resource setting.

We further explore the generalization ability of KTGF equipped with data augmentation. Specifically, w/ term data denotes that the KTGF base only augments the training data in the term generation phase, while w/ both means that the special status is further employed to enrich the training data in the status generation phase after term data augmentation. From Table 5, we can see that the additional training data with term annotation improve the performance in all the metrics. This demonstrates that the two-stage generation of our KTGF can effectively leverage the data that only contain term annotation to alleviate the data sparsity problem. Then we enrich the training data in the status generation phase, and the performance is further boosted. This indicates that our design of special status promotes KTGF learning how to generate the correct status according to the status candidates.

## 5 Analysis and discussion

### 5.1 Ablation study

We integrate the prior schema information containing the categories and status candidates of the terms to prompt status generation. To analyse the contribution of the prior schema information, we evaluate some KTGF base variants where different prior information is removed, and the results are shown in Table 3 and Table 4. Specifically, w/o status denotes that the KTGF base, which do not use the status information and only use the category information, while w/o category is the opposite of w/o status; w/o knowledge means that neither status nor category information is utilized. Since the CMDD dataset only contains the symptom category, it is unnecessary to distinguish terms from different categories. Therefore, we do not use the category information in this dataset.

From Table 3, it can be seen that each time after removing a kind of relevant prior knowledge, the performance of the KTGF model declines, indicating that all kinds of knowledge are helpful for the MD-TSPE task. On the Chunyu dataset, the F1-score drops from 82.55 to 79.05 after removing the status candidates, which is a larger reduction (3.5) compared with the w/o category (0.11). That is because the status candidates intuitively include the information that the generated output needs, which makes the model learn the status generation more effectively. The performance after removing both categories and status candidates is the worst. Table 4 also shows that incorporating both status and category information gives KTGF the best results on the dialogue-level evaluation.

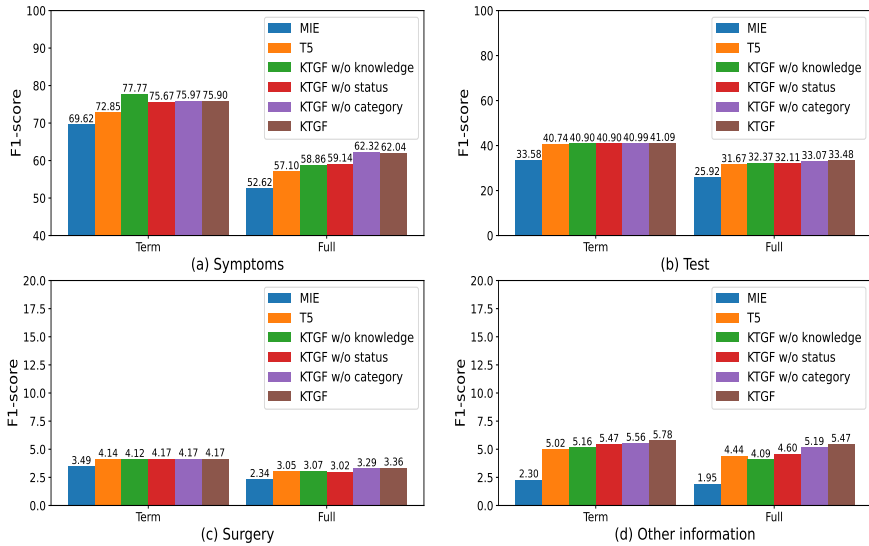
We also note the small impact of introducing prior schema information on term generation. As shown in Table 3, both the precision on the Chunyu, all metrics on the CMDD dataset are improved after removing the status candidates, which is different from the full evaluation. The preliminary analysis is that the status candidates depend on the categories and have different effects on different categories of terms. For further analysis, we conduct the evaluation on different categories, and the details are left to Section 5.2.

In addition, compared with the results on the CMDD dataset, more remarkable improvement is achieved on the Chunyu dataset after introducing the prior schema information. One possible reason for explaining this phenomenon is that the Chunyu dataset has more categories and more status candidates, and the learning of complex prior task knowledge is more dependent on the knowledge-enhanced prompt.

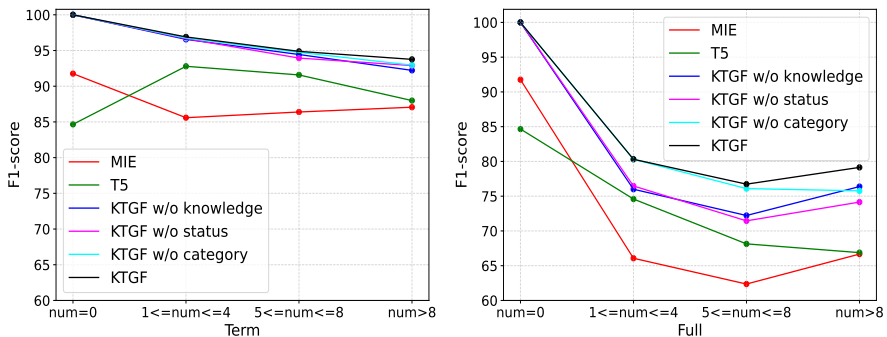
## 5.2 Comparison w.r.t Different categories

Considering that the prior schema information contains the category and the status candidates of terms, and the status candidates also depend on the categories, we are curious about the effectiveness of the knowledge on different categories. Here we evaluate the Chunyu test set based on the categories, and the empty prediction is not taken into account. The Chunyu dataset has four categories (Symptom, Test, Surgery, and Other Info), and the evaluation results of different categories are shown in Figure 3.

As seen, the performance of all models on different categories varies greatly, but it is basically consistent with the amount of data for corresponding categories in the dataset. The category “Symptom” occurs most frequently, while the “Surgery” and “Other info” rarely appear. For all categories, our proposed KTGF outperforms the baseline models MIE and T5. After utilizing the categories and status candidates, KTGF performs best both on term evaluation and full evaluation in three of the four categories, except “Symptom”. Considering that “Symptom” occurs most frequently in the dataset among all the categories, we analyse that it is the utilization of both category and status candidates that makes the model pay more attention to the long-tail category. Therefore, the performance of the category “Symptom” is reduced, which results in KTGF w/o knowledge performing the best on term evaluation. Even so, compared



**Figure 3:** The evaluation of different categories on F1-score.



**Figure 4:** The evaluation for different numbers of mentioned terms on F1-score.

with KTGF w/o knowledge, the KTGF w/o category still performs better on full evaluation, demonstrating that status knowledge can be helpful for the category “Symptom” on status generation.

### 5.3 Comparison w.r.t The numbers of mentioned terms

The two-stage generation of our KTGF model simplifies the output sequence to only contain the terms in the first stage and gains better results on term

evaluation. Here we further explore the effectiveness of the KTGF model on different numbers of mentioned terms, which determines the length and complexity of the output sequence in the term generation phase. All the experiments are performed on the Chunyu dataset, and the results are presented in Figure 4. “ $num = 0$ ” means that the test examples do not have golden labels. “ $1 \leq num \leq 4$ ” means there are no more than four terms mentioned in the test examples.

It can be observed that the performance of the classification-based model MIE has no obvious relationship with the number of mentioned terms in the term evaluation. With the increase in the number of mentioned terms, the performance of the T5 model decreases significantly both on term evaluation and full evaluation. Even the F1-score of the T5 model in “ $num = 0$ ” only is 85, which means the model can not identify the examples that do not have golden terms. Compared with the generative model T5, the performance of the two-stage generation decreases more slowly as the number of mentioned terms increases on term evaluation. Even the downwards trend of KTGF has changed on full evaluation. These results show that KTGF is more robust both in the situation of no terms and many terms during term generation and eliminates the interference of the number of generated terms during status generation. Unsurprisingly, after utilizing the categories and status candidates, the KTGF model gains the best results on full evaluation.

#### 5.4 Comparison w.r.t The changed status during dialogue interaction

A common scenario is that the status of the term is changed through the interaction of speakers. Sometimes the status of the term may not appear immediately after the term, and status inference requires a better understanding of dialogue interaction. We filter the examples that contain these terms from the Chunyu test set and evaluate them with different models. The results are shown in Table 6. From the results, we can see that our KTGF model achieves the best performance for all metrics. Compared with the T5 model, the KTGF model even gains a margin of 9.57 (73.48 for KTGF vs. 63.91 for T5), which is more than the margin of 7.44 obtained on overall data (82.55 for KTGF vs. 75.11 for T5) for full test evaluation. This demonstrates that our model can better understand more complicated dialogues where the status changes as dialogue progresses.

#### 5.5 Case Study

To more qualitatively compare the performance of the KTGF model with other baseline models, we select an example from the test set of Chunyu, as shown in Table 7. There are five terms mentioned in this example, **electrocardiogram** and **myocardial enzyme** belong to the test, the category of **chest pain** and **cold** is the symptom, and **smoking** belongs to other information.

**Table 6:** Evaluation of the examples with the changed status.

Model Status	Term			Full		
	Precision	Recall	F1-score	Precision	Recall	F1-score
MIE	93.4	81.23	85.09	58.85	59.51	57.43
T5 base	94.91	93.43	93.32	64.86	64.08	63.91
<b>KTGF base</b>	<b>100</b>	95.23	97.21	<b>75.69</b>	<b>71.92</b>	<b>73.48</b>
<b>w/o status</b>	100	94.78	96.92	71.31	67.66	69.15
<b>w/o category</b>	100	<b>95.3</b>	<b>97.28</b>	75.24	71.69	73.19
<b>w/o knowledge</b>	100	94.98	97.04	71.62	67.97	69.46

As seen, all baselines ignore **smoking**, while our KTGF model produces it and its corresponding status correctly. This demonstrates the satisfactory ability of KTGF to capture the terms of rare categories. In addition, the T5 model misjudges the breath mentioned in the spoken description of chest pain as **dyspnea**, which is not the case in the KTGF model. It shows that the KTGF model can better understand the colloquial expression in medical dialogues. Moreover, the last sentence changes the statuses of many terms, and only our model predicts all of them correctly, which verifies the effectiveness of our KTGF model in complex dialogue interaction scenarios.

## 6 Conclusion

This paper proposes a knowledge-enhanced two-stage generative framework (KTGF) for term-status pair extraction from medical dialogues (MD-TSPE). In the framework, we generate all terms first and then generate the status of each generated term in the second phase. We further design the knowledge-enhanced prompt in the second phase to leverage the category and status candidates of the generated term. For the low-resource setting, we design a special status “not mentioned”, which makes more terms available and enriches the training data in the second phase.

We evaluated our KTGF model on two Chinese medical dialogue datasets, Chunyu and CMDD. The experiments show that KTGF surpasses the previous state-of-the-art results by 5.55 and 5.28 for Chunyu, and 3.13 and 4.48 for CMDD on term and full evaluations, respectively. It demonstrates the advantages of both two-stage generation and knowledge-enhanced prompt in complex dialogue scenarios compared with existing classification-based models and generative models. Low-resource experiments show that two-stage generation can leverage data with only term annotation, and improve the term generation performance. Our designed special status “not mentioned” can further enhance status generation. The effectiveness of different components of the proposed framework is also illustrated by the ablation study and further discussion. Based on the analysis in this paper, we hope that more works can be inspired to complete better MD-TSPE as well as other similar tasks through step-by-step generation and find ways to integrate richer prior knowledge into the model.

**Table 7:** A case inferred by baseline models and the proposed model.

<b>Dialogue</b>	
Patient:	22 22 years old.
Doctor:	做过检查没有。怎么个疼法? Have you done any inspections? What's the details of the pain?
Patient:	去年这个时候我也会有一种情况。但是做了很多检查都与心脏无关。是吃心可舒好转的。按理说我的疼应该和心脏没关系。具体疼痛我也不知道从哪里出来的。你深呼吸，左胸会疼。就这3天。会不会是因为天气变化 It also appeared last year. But many tests have shown that it has nothing wrong with the heart. It takes a turn for the better after taking Xinkeshu Capsule. My pain should have nothing to do with my heart. I don't know where the pain comes from. My left chest would pain when I took a deep breath. It appears just for three days. Could it be from the changed weather?
Doctor:	感冒了么。心电图，心肌酶做了吗。抽烟么 Have you caught a cold? Have you done electrocardiogram and myocardial enzyme? Do you smoke ?
Patient:	没有 None.
<b>Term-Status Pairs Label</b>	
胸痛:	阳性(Chest pain: appear)
心电图:	患者未做(Electrocardiogram: not done)
感冒:	阴性(Cold: absent)
吸烟:	异常(Smoking: abnormal)
心肌酶:	患者未做(Myocardial enzyme: not done)
<b>MIE</b>	
胸痛:	阳性(Chest pain: appear)
心电图:	患者未做(Electrocardiogram: not done)
感冒:	未知(Cold: unknown)
心肌酶:	患者未做(Myocardial enzyme: not done)
心肌酶:	未知(Myocardial enzyme: unknown)
<b>T5 base</b>	
胸痛:	阳性(Chest pain: appear)
心电图:	患者未做(Electrocardiogram: not done)
感冒:	阴性(Cold: absent)
心肌酶:	患者已做(Myocardial enzyme: done)
呼吸困难:	阳性(Dyspnea: appear)
<b>KTGF base</b>	
胸痛:	阳性(Chest pain: appear)
心电图:	患者未做(Electrocardiogram: not done)
感冒:	病人阴性(Cold: absent)
吸烟:	异常(Smoking: abnormal)
心肌酶:	患者未做(Myocardial enzyme: not done)

## Declaration of competing interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

This work is supported by the Key Research Program of the Chinese Academy of Sciences under Grant (No.ZDBS-SSW-JSC006) and the National Natural Science Foundation of China (No.62206294).

## References

- [1] Patel, P., Davey, D., Panchal, V., Pathak, P.: Annotation of a large clinical entity corpus. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018, pp. 2033–2042. Association for Computational Linguistics. <https://doi.org/10.18653/v1/d18-1228>
- [2] Finley, G., Edwards, E., Robinson, A., Brenndoerfer, M., Sadoughi, N., Fone, J., Axtmann, N., Miller, M., Suendermann-Oeft, D.: An automated medical scribe for documenting clinical encounters. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 2-4, 2018, Demonstrations, pp. 11–15. Association for Computational Linguistics. <https://doi.org/10.18653/v1/n18-5003>
- [3] Finley, G., Salloum, W., Sadoughi, N., Edwards, E., Robinson, A., Axtmann, N., Brenndoerfer, M., Miller, M., Suendermann-Oeft, D.: From dictations to clinical reports using machine translation. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 3 (Industry Papers), pp. 121–128. Association for Computational Linguistics. <https://doi.org/10.18653/v1/n18-3015>
- [4] Quiroz, J.C., Laranjo, L., Kocaballi, A.B., Berkovsky, S., Rezazadegan, D., Coiera, E.: Challenges of developing a digital scribe to reduce clinical documentation burden. *NPJ digital medicine* **2**(1), 1–6 (2019)
- [5] Shi, X., Hu, H., Che, W., Sun, Z., Liu, T., Huang, J.: Understanding medical conversations with scattered keyword attention and weak supervision from responses. In: The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, pp. 8838–8845. AAAI Press
- [6] Liu, W., et al.: MeddG: A large-scale medical consultation dataset for building medical dialogue system. arXiv preprint arXiv:2010.07497 (2020)

- [7] Zhang, Y., Jiang, Z., Zhang, T., Liu, S., Cao, J., Liu, K., Liu, S., Zhao, J.: MIE: A medical information extractor towards medical dialogues. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020, pp. 6460–6469. Association for Computational Linguistics. <https://doi.org/10.18653/v1/2020.acl-main.576>
- [8] Sinsky, C., Colligan, L., Li, L., Prgomet, M., Blike, G.: Allocation of physician time in ambulatory practice: A time and motion study in 4 specialties. *Annals of Internal Medicine* **166**(11) (2016)
- [9] Wachter, R., Goldsmith, J.: To combat physician burnout and improve care, fix the electronic health record. *Harvard Business Review* (2018)
- [10] Arndt, B.G., Beasley, J.W., Watkinson, M.D., Temte, J.L., Tuan, W.J., Sinsky, C.A., Gilchrist, V.J.: Tethered to the ehr: Primary care physician workload assessment using ehr event log data and time-motion observations. *Annals of Family Medicine* **15**(5), 419 (2017)
- [11] Wei, Z., Liu, Q., Peng, B., Tou, H., Chen, T., Huang, X.-J., Wong, K.-F., Dai, X.: Task-oriented dialogue system for automatic diagnosis. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 201–207 (2018)
- [12] Kao, H.-C., Tang, K.-F., Chang, E.: Context-aware symptom checking for disease diagnosis using hierarchical reinforcement learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32 (2018)
- [13] Xu, L., Zhou, Q., Gong, K., Liang, X., Tang, J., Lin, L.: End-to-end knowledge-routed relational dialogue system for automatic diagnosis. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 7346–7353 (2019)
- [14] Peng, Y.-S., Tang, K.-F., Lin, H.-T., Chang, E.: Refuel: Exploring sparse features in deep reinforcement learning for fast disease diagnosis. *Advances in neural information processing systems* **31** (2018)
- [15] Lin, X., He, X., Chen, Q., Tou, H., Wei, Z., Chen, T.: Enhancing dialogue symptom diagnosis with global attention and symptom graph. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019, pp. 5032–5041. Association for Computational Linguistics. <https://doi.org/10.18653/v1/D19-1508>
- [16] Du, N., Wang, M., Tran, L., Lee, G., Shafran, I.: Learning to infer entities, properties and their relations from clinical conversations. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 4979–4990 (2019)
- [17] Du, N., Chen, K., Kannan, A., Tran, L., Chen, Y., Shafran, I.: Extracting symptoms and their status from clinical conversations. In: Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long



- Papers, pp. 915–925. Association for Computational Linguistics. <https://doi.org/10.18653/v1/p19-1087>
- [18] Ye, H., Zhang, N., Deng, S., Chen, M., Tan, C., Huang, F., Chen, H.: Contrastive triple extraction with generative transformer. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 14257–14265 (2021)
  - [19] Li, M., Xiang, L., Kang, X., Zhao, Y., Zhou, Y., Zong, C.: Medical term and status generation from chinese clinical dialogue with multi-granularity transformer. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **29**, 3362–3374 (2021)
  - [20] Xia, Y., Shi, Z., Zhou, J., Xu, J., Lu, C., Yang, Y., Wang, L., Huang, H., Zhang, X., Liu, J.: A speaker-aware co-attention framework for medical dialogue information extraction. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pp. 4777–4786 (2022)
  - [21] Uzuner, Ö., South, B.R., Shen, S., DuVall, S.L.: 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association* **18**(5), 552–556 (2011)
  - [22] Lai, T., Ji, H., Zhai, C., Tran, Q.H.: Joint biomedical entity and relation extraction with knowledge-enhanced collective inference. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 6248–6260 (2021)
  - [23] Su, Y., Wang, M., Wang, P., Zheng, C., Liu, Y., Zeng, X.: Deep learning joint models for extracting entities and relations in biomedical: a survey and comparison. *Briefings in Bioinformatics* **23**(6) (2022)
  - [24] Zhang, N., Chen, M., Bi, Z., Liang, X., Li, L., Shang, X., Yin, K., Tan, C., Xu, J., Huang, F., *et al.*: Cblue: A chinese biomedical language understanding evaluation benchmark. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 7888–7915 (2022)
  - [25] Happe, A., Pouliquen, B., Burgun, A., Cuggia, M., Le Beux, P.: Automatic concept extraction from spoken medical reports. *International Journal of Medical Informatics* **70**(2-3), 255–263 (2003)
  - [26] Hu, Z., Chen, X., Wu, H., Han, M., Ni, Z., Shi, J., Xu, S., Xu, B.: Matching-based term semantics pre-training for spoken patient query understanding. In: ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1–5 (2023). IEEE
  - [27] Shi, X., *et al.*: Understanding patient query with weak supervision from doctor response. *IEEE Journal of Biomedical and Health Informatics* (2021)
  - [28] Yan, J., Wang, Y., Xiang, L., Zhou, Y., Zong, C.: 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), pp. 1490–1499 (2020)
  - [29] Finley, G., Salloum, W., Sadoughi, N., Edwards, E., Robinson, A.,

- Axtmann, N., Brenndoerfer, M., Miller, M., Suendermann-Oeft, D.: From dictations to clinical reports using machine translation. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers), pp. 121–128 (2018)
- [30] Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., Neubig, G.: Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys* **55**(9), 1–35 (2023)
- [31] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., *et al.*: Language models are unsupervised multitask learners. *OpenAI blog* **1**(8), 9 (2019)
- [32] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J., *et al.*: Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **21**(140), 1–67 (2020)
- [33] Gao, T., Fisch, A., Chen, D.: Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723* (2020)
- [34] Jiang, Z., Xu, F.F., Araki, J., Neubig, G.: How can we know what language models know? *Transactions of the Association for Computational Linguistics* **8**, 423–438 (2020)
- [35] Lester, B., Al-Rfou, R., Constant, N.: The power of scale for parameter-efficient prompt tuning. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 3045–3059 (2021)
- [36] Li, X.L., Liang, P.: Prefix-tuning: Optimizing continuous prompts for generation. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 4582–4597 (2021)
- [37] Lee, C.-H., Cheng, H., Ostendorf, M.: Dialogue state tracking with a language model using schema-driven prompting. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 4937–4949 (2021)
- [38] Hu, S., Ding, N., Wang, H., Liu, Z., Wang, J., Li, J., Wu, W., Sun, M.: Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 2225–2240 (2022)
- [39] Han, X., Zhao, W., Ding, N., Liu, Z., Sun, M.: Ptr: Prompt tuning with rules for text classification. *AI Open* **3**, 182–192 (2022)
- [40] Chen, X., Zhang, N., Xie, X., Deng, S., Yao, Y., Tan, C., Huang, F., Si, L., Chen, H.: Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction. In: Proceedings of the ACM Web Conference 2022, pp. 2778–2788 (2022)
- [41] Phang, J., Févry, T., Bowman, S.R.: Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks. *arXiv preprint*

- arXiv:1811.01088 (2018)
- [42] Aghajanyan, A., Gupta, A., Shrivastava, A., Chen, X., Zettlemoyer, L., Gupta, S.: Muppet: Massive multi-task representations with pre-finetuning. arXiv preprint arXiv:2101.11038 (2021)
- [43] Su, Y., Shu, L., Mansimov, E., Gupta, A., Cai, D., Lai, Y.-A., Zhang, Y.: Multi-task pre-training for plug-and-play task-oriented dialogue system. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 4661–4676 (2022)
- [44] Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S.: Glue: A multi-task benchmark and analysis platform for natural language understanding. In: Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, pp. 353–355 (2018)
- [45] 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. *Advances in neural information processing systems* **33**, 1877–1901 (2020)
- [46] Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016)
- [47] Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), pp. 4171–4186. Association for Computational Linguistics. <https://doi.org/10.18653/v1/n19-1423>
- [48] Dong, L., Yang, N., Wang, W., Wei, F., Liu, X., Wang, Y., Gao, J., Zhou, M., Hon, H.: Unified language model pre-training for natural language understanding and generation. In: Wallach, H.M., Larochelle, H., Beygelzimer, A., d’Alché-Buc, F., Fox, E.B., Garnett, R. (eds.) *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 13042–13054 (2019). <https://proceedings.neurips.cc/paper/2019/hash/c20bb2d9a50d5ac1f713f8b34d9aac5a-Abstract.html>
- [49] Zhao, Z., Chen, H., Zhang, J., Zhao, W.X., Liu, T., Lu, W., Chen, X., Deng, H., Ju, Q., Du, X.: Uer: An open-source toolkit for pre-training models. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations, pp. 241–246 (2019)
- [50] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., *et al.*: Huggingface’s transformers: State-of-the-art natural language processing. arXiv preprint arXiv:1910.03771 (2019)
- [51] Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)