SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2024, Vol. 67, Iss. 3, 132106:1–132106:20
https://doi.org/10.1007/s11432-021-3534-9

Distinct but correct: generating diversified and entity-revised medical response

Bin LI[1], Bin SUN[1], Shutao LI[1*], Encheng CHEN[2], Hongru LIU[3], Yixuan WENG[4], Yongping BAI[5] & Meiling HU[6]

[1]*College of Electrical and Information Engineering, Hunan University, Changsha* 410082*, China;*
[2]*School of Mathematics, Sun Yat-sen University, Guangzhou* 510275*, China;*
[3]*JD Technology, Beijing* 101100*, China;*
[4]*National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing* 100190*, China;*
[5]*Xiangya Hospital of Central South University, Changsha* 410008*, China;*
[6]*Teaching and Research Section of Clinical Nursing, Xiangya Hospital of Central South University, Changsha* 410008*, China*

Received 18 October 2021/Revised 1 February 2022/Accepted 24 June 2022/Published online 21 February 2024

Abstract Medical dialogue generation (MDG) is applied for building medical dialogue systems for intelligent consultation. Such systems can communicate with patients in real time, thereby improving the efficiency of clinical diagnosis. However, predicting correct entities and correctly generating distinct responses remain a great challenge. Inspired by actual doctors' responses to patients, we consider MDG a two-stage task: entity prediction and dialogue generation. For entity prediction, we design an ent-mac post pre-training strategy by leveraging external medical entity knowledge to enhance the pre-trained model. For dialogue generation, we propose an entity-aware fusion MDG method in which predicted entities are integrated into the dialogue generation model through different encoding fusion mechanisms, using information from different sources. Because the diverse beam search algorithm can produce responses with entities that deviate from the predicted entities, an entity-revised diverse beam search is proposed to correct the entities entailed in the generated responses and make the generated responses more distinct. The experimental results on the China Conference on Knowledge Graph and Semantic Computing 2021 (A/B tests) and the International Conference on Learning Representations 2021 (online test) datasets show that the proposed method outperforms several state-of-the-art methods, which demonstrates its practicability and effectiveness.

Keywords medical entity prediction, ent-mac post pre-training strategy, entity-aware fusion medical dialogue generation, encoding fusion mechanism, entity-revised diverse beam search

1 Introduction

The COVID-19 pandemic has been characterized by shortages of medical resources, heavy burdens on medical practitioners, and long waiting times for patients. These problems affect the harmonious development of society and thus need to be alleviated [1]. Therefore, it is necessary to build an automatic response medical dialogue system that can communicate with patients in real time, collect patient information, and automatically make diagnoses according to patients' symptoms [2]. Such a system can improve the efficiency of clinical consultations and significantly reduce the burden on doctors [3].

In recent years, medical dialogue generation (MDG) has attracted increasing attention, owing to its broad application prospects [4–6]. Different from the original task-oriented medical dialogue system, which requires heavy human labor to design templates [2,3], the MDG is expected to generate context-consistent and medically meaningful responses according to annotated dialogue histories. Figure 1 presents a real example of the dialogue between a doctor and a patient. The upper frame is the input of our method, including the medical dialogue history with annotated entities, while the dashed bottom frame is the output of two tasks: entity prediction and response generation. The annotated

* Corresponding author (email: shutao_li@hnu.edu.cn)

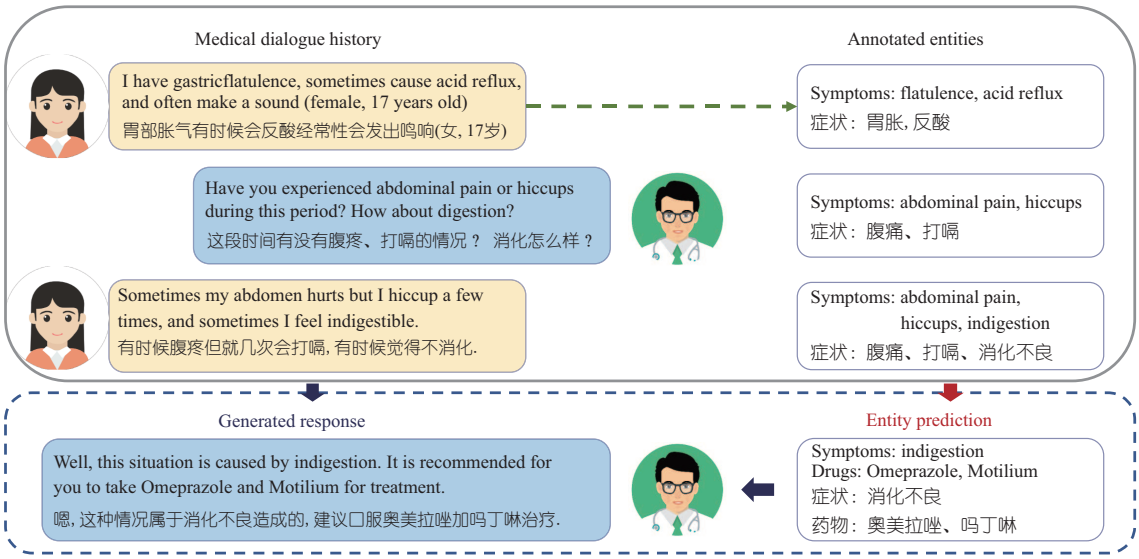

Figure 1 (Color online) An example of medical dialogue generation; the upper solid frame is the input, which includes the medical dialogue history and corresponding annotated entities; the bottom dotted frame represents entity prediction and response generation; the dashed arrow indicates the mapping from non-standard expressions to standard entities; and the solid arrows represent information flow.

entities are professional medical terms mapped from corresponding non-standard expressions (e.g., "胃胀" (flatulence) mapped from "胃部胀气" (gastric flatulence), and "消化不良" (indigestion) mapped from "觉得不消化" (feeling indigestible)). The entity prediction result serves as the medical knowledge to be expressed in the generated response corresponding to a certain dialogue history. The doctor's response combines contextual information and reasoning entities. Usually, the doctor first predicts the possible entities of the patient, such as symptoms, and then organizes the natural language to respond [7].

To develop an MDG system to imitate real doctors (i.e., predict then respond), two critical problems need to be urgently solved: (1) the system should learn how to provide reasonable answers according to correctly predicted medical entity information [2,3], as the medical entity knowledge is often entailed in the response; (2) the system should generate fluent and diversified responses [4–6], as the responses vary with individuals. Therefore, we decompose the MDG into two parts: entity prediction and response generation.

Currently, researchers on the MDG have proposed a variety of solutions, but few studies systematically elaborate on the above two problems. As for medical entity prediction, few studies have considered using external medical knowledge to enhance the pre-trained model [8]. Lack of external knowledge will result in a low entity prediction accuracy [9]. As for response generation, concatenating annotated entities with dialogue history may shorten the maximal sequence length for dialogue history embedding [5], thus making it difficult to cover long dialogues.

To solve the above problems, we design a simple two-stage pipeline to systematically model MDG problems. An ent-mac post pre-training strategy is designed to accurately predict the medical entities by improving the pre-training model with external medical entity knowledge. An entity-aware fusion MDG (EFMDG) method that utilizes different encoding fusion mechanisms to capture entity information without increasing sentence length is proposed. Furthermore, for the decoding process, an entity-revised algorithm is designed to improve the distinction and correctness of responses. The extensive experimental results on the China Conference on Knowledge Graph and Semantic Computing (CCKS) 2021 (A/B tests) and the International Conference on Learning Representations (ICLR) 2021 (online test) datasets show that the proposed method outperforms other state-of-the-art MDG methods in both automatic and manual evaluation, which demonstrates the effectiveness and practicality of the proposed method. The contributions of this paper can be briefly summarized as follows.

• For pre-training the prediction model, an ent-mac post pre-training strategy that augments the model with external medical entity knowledge is designed.

• We propose the EFMDG, which adopts various encoding fusion mechanisms with dialogue context and predicted entity encodings, thereby effectively improving the quality of responses.

• An entity-revised diversity beam search (EDBS) algorithm is designed to improve the diversity of

final responses while preserving the complete predicted entity information.

Organization. Section 2 presents the related works. Section 3 presents the problem formulation and the system framework. Sections 4 and 5 present the methodology and experiments, respectively. Section 6 presents the conclusion and future work.

2 Related work

2.1 Medical entity prediction

The current mainstream approach to medical entity prediction is to fine-tune a pre-trained model [5]. A well-designed pre-training method is required to develop a better entity prediction model [10]. The pre-training method means that the model first performs self-supervised learning on a large-scale corpus, where the semantic representations with general knowledge can be learned. In the end, the model is fine-tuned on the downstream tasks [11]. In recent years, the pre-training model technology has rapidly developed [12]. Numerous effective methods have been adopted to improve the downstream performance. BERT (bidirectional encoder representations from transformers) [13] uses a random mask pre-training method as the mask language model (MLM) task, which helps the model learn the semantic information of the context. The WWM (whole word mask) [14] outperforms the MLM, indicating that richer contextual semantics can be learned from phrases. The Mac (MLM as correction) [9] masks the entity and replaces it with similar words from the external word embedding knowledge during the pre-training stage. Specifically, it reduces the gap between the pre-training and fine-tuning stages by replacing the words with word2vec [15] method of the universal domain. The ERNIE (enhanced representation through knowledge integration) [16] separately masks the token, entity, and phrase during pre-training. This method integrates external knowledge into the contextual representation, thereby improving the general semantic expression ability of the model. However, the concept of generic medical entities covers a wide range of fields, and an entity (e.g., disease and drug names) may have numerous different non-standard expressions [17]. Using diversified entity knowledge improves the model generalization ability in the downstream medical field tasks [16]. Therefore, we design the ent-mac post pre-training strategy with the general medical knowledge contained in medical entities to assist the model in learning stronger semantic representations, thereby achieving better medical prediction accuracy.

2.2 Medical dialogue models

Numerous studies have recently investigated medical dialogue models, including task-oriented and generation-oriented models. Wei et al. [2] first designed dialogue strategies via reinforcement learning (RL) to promote automatic diagnosis. Xu et al. [3] introduced medical knowledge reasoning into a task-oriented dialogue system via RL. Xia et al. [8] proposed deep RL-based generative adversarial learning with regularized mutual information. Liao et al. [18] designed hierarchical RL for automatic disease diagnosis. Although the above studies can accurately capture patients' disease information, they adopt response templates, which require careful design. More recently, automatic dialogue generation has gradually attracted research attention. Zeng et al. [4] proposed several pre-trained language models (e.g., transformer, generative pre-trained transformer (GPT), and BERT-GPT) to generate medical responses that meet COVID-19 medical services. Liu et al. [5] designed an entity-auxiliary sequence-to-sequence model (i.e., hierarchical recurrent encoder-decoder with entity (HRED-Entity)) and a pre-trained language model (i.e., GPT2-Entity), in which the inputs are concatenated with the predicted medical entities. Lin et al. [6] proposed a meta-learning model under low-resource conditions, employing graph networks to capture dependencies between entities and conversations. Although these studies provide feasible solutions to MDG problems, they tend to concatenate the predicted entities with dialogue history [4,5]. The excess concatenated sentences will be truncated, so it becomes difficult to cover all of the information of the long dialogues during the conversation. Different from the previous methods, the EFMDG adopts various encoding fusion mechanisms by utilizing information from different sources, thereby achieving better performance for MDG.

2.3 Dialogue generation decoding algorithm

Many recent studies tend to generate outputs with high grammaticality but low distinction with the greedy method [19,20]. Multinomial sampling as a random sample method aims to increase diversity but

ignores grammatically. Top k sampling involves sorting by probability and zeroing out the probabilities below the k-th token, which improves the overall grammaticality but reduces diversity. Top p sampling [21] involves calculating the cumulative distribution, and the distribution is cut off once it exceeds the predefined probability p. The beam search method [22], as an optimized search method, uses the beam window to improve the optimization space of top k sampling, but short texts or safe responses can be easily generated. Diverse beam search (DBS) [23], as a variant of beam search, uses groups with diversity during response generation, but correct entities deviate during decoding. To alleviate this, we propose a decoding algorithm named EDBS, which ensures the diversity of the generated text and the prediction of more correct entities.

3 Two-stage medical dialogue system

3.1 Problem formulation

In this article, MDG is to generate medical responses containing correct entities, given the medical dialogue history and the annotated entities. In order to solve this problem, the whole MDG is divided into two stages. The first stage is to predict the correct entities as dialogue conditions, and the second stage is to generate responses under conditions of predicted entities.

In the problem of the MDG, the medical dialogue history $U = \{U_0, U_1, \ldots, U_i, \ldots, U_j\}$ is given, where each utterance U_i consists of some tokens and is annotated with corresponding history entity E_H^i. We process the medical dialogue utterances, which includes the dialogue history $\{U_0, U_1, \ldots, U_{j-1}\}$ and current post U_j, into flattened tokens as input $\boldsymbol{x}$. The input tokens $\boldsymbol{x} = \{x_0, x_1, \ldots, x_n\}$, where n is the total input length. The history entity $\boldsymbol{E}_H$ is also utilized as input, where $\boldsymbol{E}_H = \{E_H^0, E_H^1, \ldots, E_H^i, \ldots, E_H^j\}$. In summary, the problem of MDG aims to generate a medical information-rich response sequence $\boldsymbol{y} = \{y_0, y_1, \ldots, y_T\}$ with the predicted entities E_P, where the T is the length of generated response. We design optimal predicted entities E_P^* as the intermediate condition adopted in the prediction and generation. As a result, the above steps are as follows.

(1) Predict correct entities with dialogue history entities:

$$E_P^* \leftarrow \arg\max_{E_P \in \mathcal{E}_c} \Pr\left(E_P \mid \boldsymbol{x}, \boldsymbol{E}_H\right), \tag{1}$$

where the $\mathcal{E}_c$ is the K-dimensional vector space of the entities (K represents the number of medical entities), and E_P^* is the optimal distribution of the predicted entities over entity categories.

(2) Generate correct and medical information-rich responses:

$$\Pr\left(\boldsymbol{y} \mid \boldsymbol{x}, E_P^*\right) = \Pr\left(y_0, y_1, \ldots, y_T \mid \boldsymbol{x}, \boldsymbol{E}_H\right) = \prod_{t=0}^{T} \Pr\left(y_t \mid y_{0:t-1}, \boldsymbol{x}, E_P^*\right). \tag{2}$$

We further take the optimization with the maximum probability of each item in the multiplication in (2). The step is shown in

$$y^* \leftarrow \arg\max \Pr\left(y_t \mid y_{0:t-1}, \boldsymbol{x}, E_P^*\right), \tag{3}$$

where the final optimal token y^* is generated via auto-regression given the previous outputs $y_{0:t}$, the post $\boldsymbol{x}$ and the optimal distribution of predicted entities E_P^*.

3.2 System framework

As shown in Figure 2, the proposed a two-stage medical pipeline system divides the entire pipeline into upstream and downstream. Under the upstream, we first predict the best set of predicted entities, and optimize the results through the optimal F1 threshold search. In the downstream, the predicted entities and the dialogue context are input into the entity-aware fusion dialogue generation model. When decoding, the entity-revised DBS is used to generate diversified and correct responses.

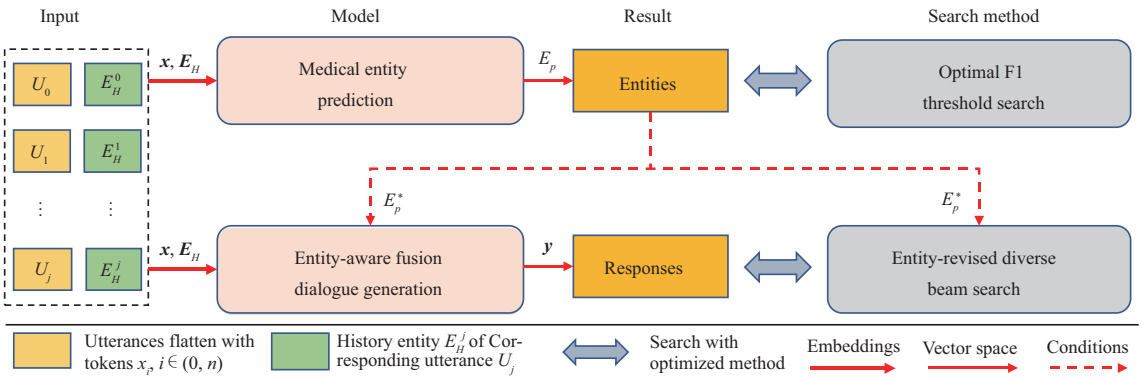

Figure 2 (Color online) The overview of two-stage system framework.

Table 1 The proposed ent-mac post pre-training strategy compared with other methods

Method	输入	Input
Origin	建议你用奥美拉唑的同时，加用吗丁啉或莫沙必利或援生力维.	It is recommended that you use omeprazole and add morpholine or mosapride or trimebutine maleate tablets.
MLM	建[M]你用奥美拉唑的同时，加用吗丁啉或[M]沙必利或援生力维.	It is recommended [M] you use omeprazole and add [M] ##line or mosapride or trimebutine maleate tablets.
WWM	[M][M] 你用奥美拉唑的同时，[M][M] 吗丁啉或莫沙必利或援生力维.	It is [M][M] you use omeprazole and add [M][M] or mosapride or trimebutine maleate tablets.
Mac	推荐你用奥美拉唑的同时，换成吗丁啉或莫沙必利或援生力维.	It is recommended that you apply omeprazole and add morpholine or mosapride or trimebutine maleate tablets.
ERNIE	建议你用[M][M][M][M]的同时，加用吗丁啉或[M][M][M][M]或援生力维.	It is recommended that you use [M] and add morpholine or [M] or trimebutine maleate tablets.
Ent-mac	建议你用雷贝拉唑的同时，加用吗丁啉或瑞巴派特或援生力维.	It is recommended that you use rabeprazole and add morpholine or rebapat or trimebutine maleate tablets.

4 Methodology

4.1 Medical entity prediction

There are two types of entities, i.e., dialogue history entities and predicted entities, in the dialogue framework. The dialogue history entities are annotated from the corresponding utterance and the predicted entities are generated by entity prediction. The medical entities are standard medical terms, which refer to the same medical knowledge with the diversified casual expressions. For some medical knowledge, the professional terms and the casual expressions are quite different in words or characters, especially in English. For example, "胃胀" ("stomach bloating") is mapped into "肠胃胀气" ("flatulence"). Given an utterance, it can be annotated from the mapping function Γ_D with a phrase-entity mapping dictionary D[1)]. Medical entity prediction is important for the MDG which improves the predicted entities for the MDG of the next utterance given the medical dialogue history and annotated entities. In our work, the medical entity prediction is implemented through pre-training and fine-tuning stages. To make better predictions, the ent-mac post pre-training strategy adopted with the optimal F1 threshold search is proposed, which will be introduced in detail as follows in this subsection.

4.1.1 *Ent-mac post pre-training strategy*

Post pre-training method is thought to be a wise choice to enhance the model with domain knowledge [24]. Therefore, the ent-mac post pre-training strategy is designed, adding the medical entity knowledge[2)] [17], where the medical entities are replaced with semantically similar medical entities.

As shown in Table 1, the MLM is used as the origin pre-training method, masking some tokens with the byte pair encoding (BPE) method. The WWM is more suitable for Chinese pre-training, which learns representation about phrases in the corpus, i.e., by masking "建议". The Mac is a good way to help model learning more about similar words, by changing the phrase into others, i.e., by replacing "建议" with "推荐", which is underlined. It is a method that augments corpus with similar words, however,

1) https://github.com/lwgkzl/MedDG/tree/master/MedDG/data.
2) http://thuocl.thunlp.org/.

it is not suitable for special domains, i.e., a medical domain that requires more expert knowledge. The ERNIE provides a masking method via masking the entity, i.e., by masking "奥美拉唑". It also faces some challenges as the type of entity is not medical-specialized but wide sparse. Different from the above method, we further leverage the mac method, by searching semantic similar phrases with external entity knowledge augmenting, i.e., altering the "奥美拉唑" to "雷贝拉唑". Compared with other pre-training methods, our method further considers the different expressions of medical information in the sentence, as well as the relationship between these medical entities.

As shown in Algorithm 1, we design the ent-mac post pre-training algorithm. Specifically, we replace the similar semantic phrases with the vector distance based on the external medical knowledge, and introduce random replacement to increase the robustness of the post pre-training. We also open-source the code of the ent-mac, as well as the pre-trained model, which can be found on the website[3)].

Algorithm 1 Ent-mac post pre-training algorithm

Input: External medical entity knowledge with vocabulary $V = \{V_1, V_2, \ldots, V_m\}$; unsupervised Chinese medical sentences $C = \{C_1, C_2, \ldots, C_n\}$ together with corresponding sentence as string label $L = \{L_1, L_2, \ldots, L_n\}$, where the $C_i = L_i$; Chinese medical word vector $\Omega = \{W_1 : \omega_1, W_2 : \omega_2, \ldots, W_k : \omega_k\}$, k is the vocab size; characters counting threshold ζ.

Output: List of the input and its corresponding string label $\{\{C_1, L_1\}, \{C_2, L_2\}, \ldots, \{C_j, L_j\}\}$, where j is the whole length of the whole input-label pair list.

Initialize the list of input-label pair $Q \leftarrow \emptyset$;
for $t = 1$ **to** n **do**
  $L_t \leftarrow C_t$; // Unsupervised sentence corresponded with their string as label
  char_count = 0; //Start selecting sentence from the front
  for $u = 1$ **to** m **do**
    Generate a random number $\mu \in (0, 1)$;
    if $V_u \in C_t$ **then**
      if $\mu < 0.8$ **then**
        Find the most similar word W^* in Ω according to the vector distance; // W^* = gensim[4)].most_similar('V_u')
        Replace corresponding word V_u in C_t with W^*;
        char_count = char_count + len(W^*);
      else if $\mu < 0.9$ **then**
        Find the random word W' in V; // Pick W' randomly from the vocab V
        Replace corresponding word V_u in C_t with W'; // Random replacement if the probability between 0.8 and 0.9
        char_count = char_count + len(W');
      end if
      if char_count $\geqslant$ len(C_t) $*$ ζ **then**
        Break; // If the conditions are not met, then continue till the end
      end if
    end if
  end for
  Q.append(list(C_t, L_t)); // Append the sentence with the corresponding string label
end for

4.1.2 *Model architechture*

Our medical entity prediction model is shown in Figure 3, where we choose different pre-trained models, including BERT [13], PCL-MedBERT[5)], RoBERTa-wwm-ext [14], Mac-BERT-large [9], and ERNIE [16] as the backbone. The ent-mac post pre-training method is used with online medical dialogues[6)] as continuing pre-training, in order to improve the generalization of the model in medical domain tasks.

We extracted the features H_0 of the last three layers with the concatenation of the CLS vector. Then, the concatenated vector is passed through an attention layer to utilize the information between different layers. The final predicted entity distribution is obtained via Multi-sample dropout[7)], fully perception layer, and sigmoid function. Specifically, we concatenate multiple rounds of dialogue history with history entities, and introduce [SAP] token as a separator to separate history dialogues and history entities, such as the 1st input $\boldsymbol{x}_1 = [\text{CLS}] + U_0 + [\text{SAP}] + E_H^0 + [\text{SEP}] + U_1 + [\text{SAP}] + E_H^1 + [\text{SEP}]$. As a result, the multi-category classification task is designed with the loss function $\mathcal{L}_{\text{p}}(X, T)$, where the final input is X with the corresponding label T (the entities of next sentence). The above steps are defined as follows:

$$\mathcal{L}_{\text{p}}(X, T) = \frac{1}{N} \sum_{k=1}^{N} -w_k \left[t_k \cdot \log \sigma \left(\boldsymbol{x}_k \right) + (1 - t_k) - \log \left(1 - \sigma \left(\boldsymbol{x}_k \right) \right) \right], \tag{4}$$

3) https://github.com/WENGSYX/Chinese-Word2vec-Medicine/.
4) https://github.com/RaRe-Technologies/gensim.
5) https://code.ihub.org.cn/projects/1775.
6) https://github.com/Lireanstar/Medical-Dialogue-Corpus.
7) https://github.com/lonePatient/multi-sample_dropout_pytorch.

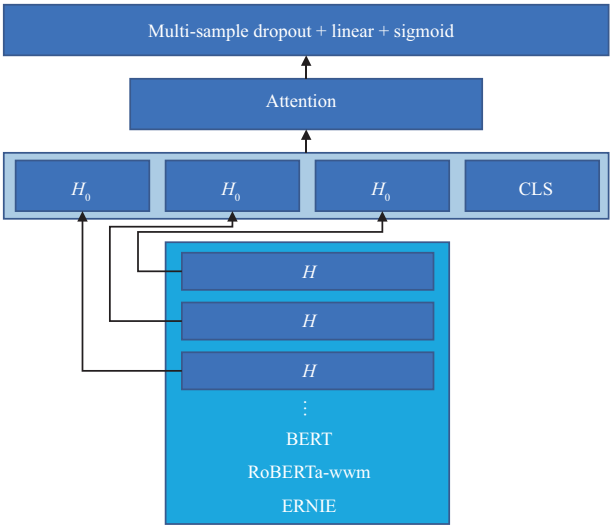

Figure 3 (Color online) Structure of our entity prediction model.

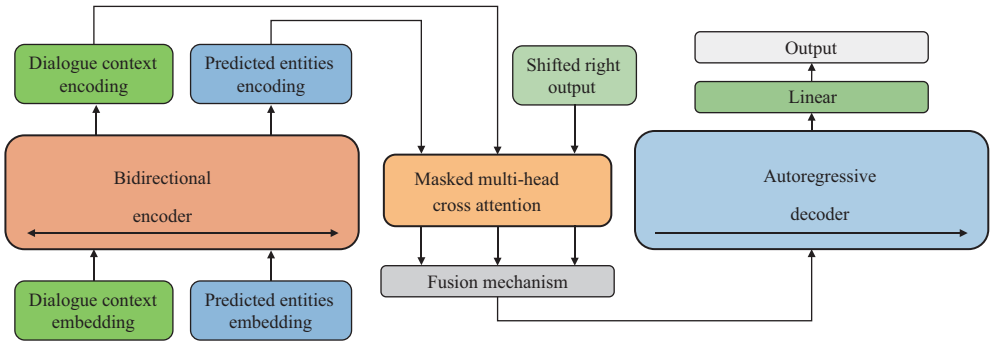

Figure 4 (Color online) Entity-aware fusion dialogue generation model.

where t_k is the target label of $\boldsymbol{x}_k$, σ is the neural network, w_k is the optimal weight performed with F1 threshold search on the validation set, t_k is the target entity and $\boldsymbol{x}_k$ is the input feature, N is the total training number.

4.1.3 *Optimal F1 threshold search*

As the categories are not balanced and the result obtained by cross-entropy loss is not globally optimal, in order to obtain the best F1 index, the optimal weight w_k is designed with the optimal F1 threshold search.

We consider each category of the multi-category problem as a two-category problem. A reasonable threshold can be obtained by threshold searching. More precisely, we can obtain the optimal threshold by adjusting the threshold from 0.3 to 0.6 through the grid search, with the step of 0.001.

4.2 Entity-aware fusion dialogue generation

The entity-aware fusion dialogue generation model is presented in Figure 4. We adopt the encoder-decoder architecture as the backbone. The dialogue context embedding and predicted entities embedding are encoded with a bi-directional encoder. Then, the encodings of the dialogue context and the predicted entities are fused through the masked multi-head cross attention (MMCA) mechanism [25]. Different fusion strategies are designed to fuse information from different sources. Finally, the shifted right output together with the fused encoding is used for final response generation via auto-regression. Moreover, the auxiliary training tasks are designed to bridge the gaps of training objects in the system framework.

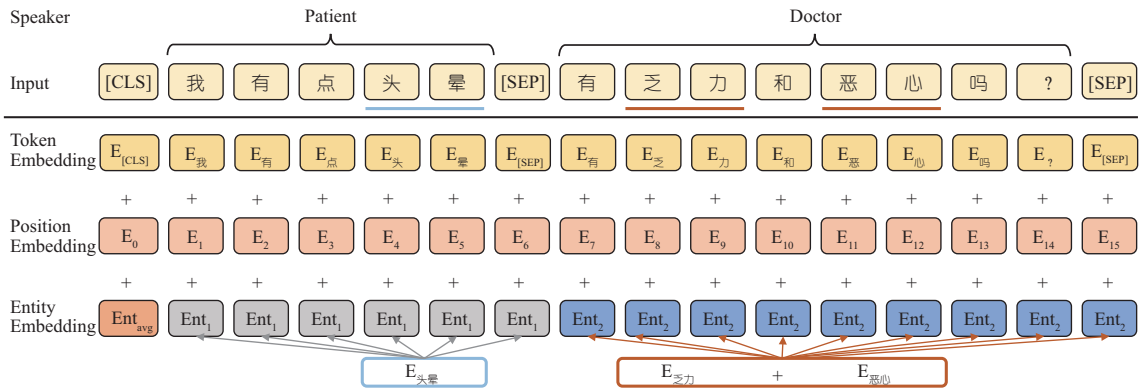

Figure 5 (Color online) Structure of dialogue context embedding.

4.2.1 *Dialogue context embedding*

The context embedding module is presented in Figure 5, where the token embedding, the position embedding, and the entity embedding of the input are added element-wisely together as the dialogue context embedding. The underlines of the input indicate the annotated entities. The notation Ent_k represents the entity embedding of the k-th utterance. The Ent_k is obtained by adding all the embeddings of the annotated entities in the k-th utterance together. Each Ent_k will be replicated to the same length as the corresponding utterance. As shown in Figure 5, there are two utterances representing the speaker of the patient and the doctor respectively. Specifically, the entity embedding is obtained through a one-layer linear perception projection, which is the same dimension as the token embedding. For the first patient utterance, the annotated entity only contains "头晕". As a result, the notation Ent_1 represents the corresponding entity embedding $E_{\text{头晕}}$. For the second doctor utterance, the annotated entities contain "乏力" and "恶心". The notation Ent_2 represents the sum of the entity embeddings, i.e., $E_{\text{乏力}} + E_{\text{恶心}}$. These entity embeddings are replicates to the same length of the corresponding utterance. Moreover, the Ent_{avg} is utilized as the averaged entity embedding head of the entire sentence (divided by the sentence length) for capturing the whole entity information.

4.2.2 *Predicted entities embedding*

The entities are concatenated together and separated by [SEP], which is mapped in the form of tokens by the tokenizer. As a result, the entity embedding is obtained by adding these tokens with the position embedding.

4.2.3 *MMCA*

To collect information from different sources, the MMCA [25] is adopted to encode the dialogue context E_C, the predicted entities E_{ent} and the shifted right previous output $E_{\text{prev.}}$. The equations are shown as follows:

$$O_E = \text{MMCA}\left[E_{\text{prev.}}, E_{\text{ent}}, E_{\text{ent}}\right], \tag{5}$$

$$O_C = \text{MMCA}\left[E_{\text{prev.}}, E_C, E_C\right], \tag{6}$$

where the O_E and the O_C represent the encoding of predicted entities and dialogue context. In order to pay attention to the information of decoded tokens, the previous decoded encoding O_P is obtained through

$$O_P = \text{MMCA}\left[E_{\text{prev.}}, E_{\text{prev.}}, E_{\text{prev.}}\right]. \tag{7}$$

4.2.4 *Encoding fusion mechanism*

After obtaining the O_E, O_C, and O_P, we provide five different fusion strategies to perform our fusion mechanism, including average, source-level weighted, source-dimension weighted, max and min fusion, which are shown as follows.

- Average fusion

$$O_F = \left(O_E + O_C + O_P\right)/3. \tag{8}$$

• Source-level scalar weighted fusion, where we select three trainable scalars for fusing different encoding,

$$O_F = (aO_E + bO_C + cO_P)/(a+b+c), \tag{9}$$

where each variable, i.e., $a, b, c \in (0, 1)$, is learned independently through back propagation.

• Source-dimension weighted fusion, where we select three trainable vectors $\boldsymbol{w}^e, \boldsymbol{w}^c, \boldsymbol{w}^p \in \mathbb{R}^d$, $d = 512$, for fuse different encoding,

$$O_F = (\boldsymbol{w}^e \times O_E + \boldsymbol{w}^c \times O_C + \boldsymbol{w}^p \times O_{\mathrm{P}}) / (\boldsymbol{w}^e + \boldsymbol{w}^c + \boldsymbol{w}^p). \tag{10}$$

• Maximum fusion

$$O_F = \max(O_E, O_C, O_P). \tag{11}$$

• Minimum fusion

$$O_F = \min(O_E, O_C, O_P). \tag{12}$$

4.2.5 *Dialogue generation*

The dialogue generation is processed via auto-regression, and the loss function is shown as follows:

$$\mathcal{L}_D(\varphi) = -\sum_t \log P_\varphi(y_t \mid y_0, \ldots, y_{t-1}, O_E, O_C) = -\sum_t \log P_\varphi(y_t \mid O_F), \tag{13}$$

where i represents the i-th word generated by the decoder, $y_0, \ldots, y_{t-1}$ is the generated tokens, and y_t is the next token. Identically, the input of the decoder also can be represented as the mean fused encoding.

4.2.6 *Auxiliary training tasks*

The experimental datasets used in the pre-training stage come from the vast domains for obtaining a good generating performance, but the model needs to fit the data in the medical domain to ensure the correctness of the responses, thereby influencing the fluency of the generated responses. To bridge the gap of datasets used in two stages, we add the original language model task as the auxiliary training objective.

• Language model task. Inspired by [26], we choose the original language model task to fine-tune the medical dialogue datasets for remedying the gap of the used data in different training stages. We choose the dialogue history as the input to accomplish this auxiliary task, the equation is as follows:

$$\mathcal{L}_{\mathrm{LM}}(\varphi) = -\sum_i \log P_\varphi(x_i \mid x_{i-k}, \ldots, x_{i-1}), \tag{14}$$

where φ represents the parameters of the encoder-decoder model, k is the size of the context window, and $x_{i-k}, \ldots, x_{i-1}$, x_i is the sequence of tokens sampled from the training corpus.

In the training of the proposed model, the ground-truth of the annotated entities is used during the fine-tuning training, while the predicted entities generated by the entity prediction module are used during the testing. An intuitive idea to tackle such difference is to obtain the predicted entities directly from the encoder-decoder model, so that the model can learn the ability to express the predicted entities. Therefore, we add the hierarchical prediction as the second auxiliary training task.

• Hierarchical prediction task. Note that there is a hierarchical structure in the annotated entities shown in Figure 1, i.e., "症状: 胃胀, 反酸" (Symptoms: flatulence, acid reflux). The "Symptoms" represent the medical domain, while the "flatulence, acid reflux" are the medical entities. Consequently, we choose the dialogue context encoding O_C for the hierarchical prediction task

$$\mathcal{L}_{\mathrm{Domain}}(\theta) = -\sum_i \log P_\theta(t_i \mid O_C), \tag{15}$$

$$\mathcal{L}_{\mathrm{Entity}}(\theta) = -\sum_i \log P_\theta(t'_i \mid O_C), \tag{16}$$

where t_i and t'_i represent the ground truth of domain and entity classification respectively, $\mathcal{L}_{\mathrm{Domain}}(\theta)$ represents the loss function of domain types, $\mathcal{L}_{\mathrm{Entity}}(\theta)$ represents the loss function of entity types.

As a result, the final loss function can be represented as

$$\mathcal{L}(\varphi, \theta) = \mathcal{L}_D(\varphi) + L_{\mathrm{LM}}(\varphi) + L_{\mathrm{Domain}}(\theta) + L_{\mathrm{Entity}}(\theta), \tag{17}$$

where $\mathcal{L}(\varphi, \theta)$ is the total loss, the φ and θ represent the parameters of neural network respectively.

4.3 Entity-revised DBS

A DBS is used to generate diversified responses. However, due to the lack of conditional information guidance, the results of the original DBS are often uncontrollable but diversified. Therefore, we design EDBS under the condition of the predicted entity for response generation, adopting the entity modification as a guidance so that the results do not shift from the condition entities E_P. Specifically, the EDBS is designed with multinomial sampling (multinomial) and greedy search sampling (argmax), to balance the diversity and grammaticality. A threshold τ is chosen to ensure the convergence of dialogue generation. In the Algorithm 2, the multinomial softmax in line 16 tends to produce more diversified while the argmax softmax in line 18 apts to produce more grammatically influenced responses. At the beginning of the decoding, it is more likely to choose the multinomial sampling for diversified results. As the generated sentence gets longer, the probability of using greedy sampling is greater, which can ensure the grammaticality of the generated sentence. To ensure that the final result is faithful to the predicted entities, we first adopt the mapping function Γ_D with a phrase-entity mapping dictionary D[1)] to revise the sentence by eliminating the deviated entities contained in the sentence. Then, for the absent predicted entities, their corresponding expressions are added to the revised sentence obtained by the reverse mapping, so that the augmented sentence contains all the predicted entities. Algorithm 2 summarizes the EDBS process. To sum up, we consider the two cases of conditional shifting in dialogue generation, and use the entity-sentence mapping function conditioned with predicted entities to improve the dialogue decoding, correcting it until the final result is obtained.

5 Experiments and results

In this section, the external knowledge dataset and four different datasets for experiments are first introduced. Strong baselines and the corresponding implementation setting are described in detail. The extensive experiments are carried out to demonstrate the effectiveness of our method, including prediction and dialogue generation. Finally, we analyze the ablation studies and the case studies and present the online results of the proposed method. Moreover, we have also established a medical dialogue system under the proposed framework online at the website[8)].

5.1 Datasets

The external entity knowledge comes from the THUOCL dictionary [17]. The main domain list and the corresponding statistics of words in the THUOCL are presented in Table 2, where 157173 words in 11 fields are contained. We choose the vocabularies from the medical field as the external entity knowledge for the ent-mac post pre-training, where the total number of the medical entity knowledge counts 18749. The vocabularies originate from professional medical vocabularies, medical terminologies, and thesaurus of mainstream medical websites. It is processed manually by professional doctors with many rounds to ensure the correctness of the data collection.

All dataset statistics for MDG are shown in Table 3, and the MDG [5] dataset is used for training models. The whole MDG set contains 17864 dialogues, which are divided into training and validation sets according to the ratio of 8:2. Furthermore, three other different test sets are chosen for experiments for evaluations, including CCKS-A, CCKS-B, and ICLR. We adopt these datasets from the competitions as the test data, where the data collections and distributions are similar to the MDG dataset. Among them, the CCKS-A/CCKS-B/ICLR are all collected from the same source[9)] as the MDG dataset, where the data processing format of those datasets is the same as the MDG dataset. It is worth noting that the CCKS-B dataset is processed with real doctors' manual revisions on the original ground truth, which makes it interesting to see whether the model can generate responses in line with the level of human thinking. The results in the ICLR set can only be obtained via submitting online, where the results should be submitted anonymously. It is fair to evaluate the performance of different methods remotely.

5.2 Models for comparsion

We compare our method with several state-of-the-art methods, from three aspects: without additional predicted information, with predicted entities, and the scale of the pre-trained model.

8) http://med.wengsyx.com/.
9) https://www.chunyuyisheng.com.

Algorithm 2 Entity-revised DBS

Input: Beam width B; flattened input tokens $\boldsymbol{x}$; predicted entities E_P with corresponding optimal distribution E_P^*; beam space vector $\boldsymbol{V} = [\mathbf{logits}_1, \ldots, \mathbf{logits}_B]$; beam string list $[W^1, \ldots, W^B]$; beam score list $S = [P^1, \ldots, P^B]$; convergence threshold τ; maximum length N of generated response; phrase-entity mapping dictionary $D = \{s_i : (e_{i,1}, \ldots, e_{i,m}) \mid i = 1, \ldots, k\}$ with the mapping function $e = \Gamma_D(s)$ mapping sentence s into entities e; search completed flag ξ.

Output: Set of solutions $K = [\{W^1, S^1\}, \ldots, \{W^i, S^i\}, i = 1, \ldots, B]$.

Perform top-B decoding algorithm to obtain $[\{y_0^1, P_0^1\}, \{y_0^i, P_0^i\} \mid i = 1, \ldots, B]$, the y_0^i and P_0^i represent the decoding token and corresponding probability at the $t = 0$ respectively;
Initialize the score list $S = [P_0^1, \ldots, P_0^B]$, and beam string $W = [y_0^B, \ldots, y_0^B]$;
Initialize set of solutions $K \leftarrow [\{y_0^1, P_0^1\}, \ldots, \{y_0^B, P_0^B\}]$, y_0^i is the first generated token of i-th beam;
Initialize the logits vector $\boldsymbol{V}$ with zeros, where the size is [B, len(**logits**)];
Initialize decoding completed flag to zeros $\xi = [0, \ldots, 0]$, where the size is [1, B];
for $t = 1$ to $N - 1$ **do**
    for $i = 1$ to B **do**
        if $K[i][0]$ == EOS **then**
            $\boldsymbol{V}[i] \leftarrow \mathbf{0}$; // Set the logits in the beam space vector in the i-th beams to zero vector
            $\xi[i] = 1$; // This flag indicates that the string has been decoded
            continue
        end if
        Generate a random number $\eta \in (0, 1)$;
        if $\eta \leqslant \tau$ **then**
            // Utilize multinomial to increase the diversity
            $P_t^{i*} = \text{multinomial}\,\text{Softmax}\{\Pr\left(y_t^i \mid y_{0:t-1}^i, \boldsymbol{x}, E_P^*\right)\}$;
        else
            $P_t^{i*} = \text{argmax}\,\text{Softmax}\{\Pr\left(y_t^i \mid y_{0:t-1}^i, \boldsymbol{x}, E_P^*\right)\}$;
        end if
        $\boldsymbol{V}[i] = S[i-1] \times P_t^{i*} \times \delta\{\Pr\left(y_t^i \mid y_{0:t-1}^i, \boldsymbol{x}, E_P^*\right)\}$;
    end for
    $Z \leftarrow \text{Flatten}(\boldsymbol{V}_i).\text{top-}B$; // The size of the Flatten($\boldsymbol{V}_i$) is [1, B*len(**logits**)], $Z = [\{y_t^1, P_t^1\}, \ldots, \{y_t^B, P_t^B\}]$
    for $i = 1$ to B **do**
        if $\xi[i] == 1$ **then**
            continue
        end if
        $W[i] = W[i-1] + Z[i][0]$;
        $S[i] = Z[i][1]$;
        $K[i][0].\text{update}(W[i])$;
        $K[i][1].\text{update}(S[i])$;
    end for
    $\tau = \tau$ * 0.9;
end for
// To revise the entities contained in the generated candidates
for b in K **do**
    Divide each b into short sentence list $G = [g_1, \ldots, g_n]$ at sentence granularity separated by punctuation;
    // Delete the entities absent in the predicted entity list
    for g in G **do**
        if $(\Gamma_D(g) \cup E_P)$ is not in E_P **then**
            Delete the absent entities g from G;
        end if
    end for
    // Find the predicted entities absent in response and add into the generated sentence
    $R = \Gamma_D(G)$
    Set the absent entities list $Q \leftarrow \emptyset$;
    for e in E_P **do**
        if e is not in R **then**
            Q.append(e);
        end if
    end for
    if Q is not $\emptyset$ **then**
        Add $\Gamma_D^{-1}(Q)$ into b; // $\Gamma_D^{-1}(Q)$ is the sentence corresponded with the absent predicted entities
    end if
end for

Table 2 Domain and statistics of THUOCL dictionary

Domain	IT	Economics	Idiom	Toponymy	History	Poem	Medicine	Diet	Law	Mobile	Animal
Statistics	16000	3830	8519	44805	13658	13703	18749	8974	9896	1752	17287

• Transformer [4] represents the original sequence-to-sequence (seq2seq) architecture, where the inputs are flattened for dialogue generation without additional predicted information.

• GPT2-Entity [5] uses the architecture of the language model of the maximum length of 300. The notation "1-Entity" represents the flattened inputs that are concatenated with predicted entities.

Table 3 Statistics of the experimental datasets

	MDG	CCKS-A	CCKS-B	ICLR (online)
# Dialogues	17864	2747	1600	5649
# Utterance	385951	29601	15750	80707
# Chars. per dialogue	382.45	379.33	397.99	400.12
# Chars. per utterance	17.70	17.07	19.40	18.60
# Entities per dialogue	12.16	12.01	14.22	12.84
# Entities per utterance	0.56	0.53	0.64	0.60

• HRED-Entity [5] adopts the architecture of the recurrent neural network (RNN) encoder by stacking two hierarchical RNNs, where the flattened inputs are also concatenated with predicted entities.

• BertGPT-Entity [25] uses the seq2seq architecture that adopts the MMCA mechanism when generating. It is initialized with the pre-trained model [4] pre-trained on the medical texts, where we concatenate the flattened inputs with predicted entities.

• T5-Entity [27] has different variants due to the size of different training corpus, i.e., small and base. We implement with pegasus pre-training method [28] and post pre-training method [24] with open-source medical corpus respectively as strong baselines. The baselines of the T5-Entity family are also implemented with flattened inputs concatenated with predicted entities.

• CPM2-prompt [29] is the largest pre-trained model based on the seq2seq architecture, where the predicted entities are used as the prompt tokens for prompting the additional predicted entities for the dialogue generation model.

• The end2end method is also adopted for comparison, where the model is based on the seq2seq architecture. After obtaining the CLS vectors on the encoder, the vectors are used to predict entities under the one perception layer. Finally, the flattened inputs concatenated with the predicted entities are used for training the end2end model.

5.3 Implementation settings

For the post pre-training of entity prediction, the ent-mac post pre-training is adopted with the medical corpus, and the characters counting threshold is set to 0.15. We use the default settings[10] to implement the post pre-training with the proposed ent-mac strategy on four NVIDIA 3090 GPUs. The dataset and codes are publicly available on the website[11].

For the entity prediction fine-tuning, all the baselines use the same configuration, where the stratified learning rate is adopted with an attenuation strategy with decay = $1e^{-4}$. Specifically, a larger value is set for the upper learning rate of the backbone, the internal learning rate of the pre-trained model is smaller, and the closer to the lower layer, the smaller the learning rate, which ranges from lr = $5e^{-4}$ to lr = $1e^{-4}$. We also adopt the FGM adversarial training [30], mixed-precision training[12] with a batch size of 32. The optimal f1 threshold is set to 0.55.

For the dialogue generation, we design a curriculum boost learning strategy [31] to fine-tune the proposed EFMDG model, where the architecture of the model is the seq2seq. The size of the token embedding is 768 and the context window is 512. We fine-tune the EFMDG model with the medical dialogue corpus of training processes with different difficulties. The training steps are as follows.

(1) The original pre-trained model (i.e., BertGPT-Entity) is utilized to initialize the parameters of the encoder and decoder, which is fine-tuned with the cleaned online medical dialogues. Then, we use the boost strategy to train 4 epochs for a total of 5-fold.

(2) The dialogues with entities of the doctor are used for training, so that the generated response will contain the common features of doctors. We use the boost strategy to train 4 epochs for a total of 5-fold.

(3) We further sort out the dialogues with entities of doctors, whose length is greater than 11 (counted on the validation set) to train the dialogue generation model, because these dialogues have more entity characteristics. It is easier for the model to adapt to generating longer sentences. We train 2 epochs for a total of 5-fold.

With the implementation of the batch size of 32 with gradient accumulation. For decoding, the convergence threshold is set to 0.9. The AdamW optimizer [32] is adopted for fine-tuning with 2000

10) https://github.com/ymcui/MacBERT.
11) https://github.com/Lireanstar/Entity-aware-Fusion-Medical-Dialogue-Generation.
12) https://github.com/NVIDIA/apex.

warm-up steps until the learning rate reaches $1e^{-5}$ on four NVIDIA 3090 GPUs, where the random seed of 2022.

For the other baselines the except for the CPM2-prompt, all have a batch size of 32 and a word embedding size of 512 with the AdamW optimizer with an initial learning rate of $1e^{-4}$ and annealed it gradually after 2000 warm-up epoch until it reached $1e^{-5}$. As for the CPM2-prompt, we adopt the original parameter setting[13] for fine-tuning with the training dataset on four NVIDIA 3090 GPUs.

5.4 Evluation settings

5.4.1 *Automatic evluation*

Two indicators in the original evaluation[14] are adopted, including averaged BLEU-1/4 [33] score and Entity-F1, which is to measure response generation quality and entity correctness respectively. Furthermore, the metrics of Distinct-1/2 [34] are implemented to measure the diversity.

5.4.2 *Human evluation*

For the baselines and our model, we randomly picked 200 test cases from the test set. Each generated sentence is scored by three independent persons with a medical background. Artificial responses (golden response) with other generated responses are scored with three metrics: (1) utterance fluency: to decide whether the sentences are fluent, and may be produced by humans; (2) entity correctness: to judge whether the generated sentences contain correct entities; (3) entire quality: a comprehensive index for evaluating sentence quality. The rating scale for each metric ranges from 1 to 5, where 1 represents the worst and 5 represents the best.

5.5 Experimental results

In this part, we conduct a detailed analysis of the automatic results in entity prediction and dialogue generation, as well as manual results in MDG.

5.5.1 *Entity prediction results*

As shown in Table 4, the BERT-base-Chinese and RoBERTa-wwm-ext have similar effects. After the backbone is replaced by ERNIE, the improvement is about 1.06 and 0.76 in the CCKS A and the CCKS B respectively, which indicates that pre-training with external entity knowledge is helpful for improving the F1 scores. The Mac-BERT-large outperforms ERNIE, which shows that similar phrase replacement assists medical prediction. Because the PCL-BERT-wwm model outperforms the others, we choose the PCL-BERT-wwm with ent-mac post pre-training strategy as our final baseline backbone, with improvements of 4.48 and 4.36 in F1 score respectively on the CCKS A/B test set. We then implement the different structures with the backbone of the PCL-BERT-wwm-ent-mac model. As shown in Table 5, the results show that the medical entity prediction adopting the model structure with the concatenated features of the last three layers, attention mechanism, and multi-dropout is more competitive compared with the other baselines.

5.5.2 *Dialogue generation results*

The results of MDG are shown in Tables 6 and 7, respectively. Table 6 shows that the performance of different methods for the ICLR online test set. We can observe that the performance of the original Transformer is relatively low. It can be inferred that without predicted entities, the model has a limited ability to predict the next sentence. Besides, the short length of embedding limits the ability of GPT2-Entity (with small max_len). The original BertGPT achieves better performance than HRED-Entity, which indicates the power of pre-training. The T5-pegasus-Entity achieves high Distinct score than GPT-2-Entity, which means that the responses generated by large pre-trained models have more ability to be diversified. Finally, the proposed EFMDG method is better than other strong baselines. The same trend can also be found in Tabel 7. We also add additional baselines in both the CCKS A/B test set to further experiments. The symbol -post indicates continuing to be pre-trained, and the result shows that continuing pre-training on the training corpus is beneficial to improving the comprehensive score.

13) https://github.com/TsinghuaAI/CPM-2-Finetune.

14) www.biendata.xyz/competition/ccks_2021_mdg/evaluation/.

Table 4 F1 performance of backbone in different datasets

Model	CCKS A F1	CCKS B F1
BERT-base-Chinese [13]	31.23	31.61
RoBERTa-wwm-ext[15] [14]	31.68	31.77
ERNIE [16]	32.29	32.37
Mac-BERT-large [9]	33.64	33.89
PCL-BERT-wwm	34.68	34.87
PCL-BERT-wwm-ent-mac	35.71	35.97

Table 5 Medical entity prediction in ICLR test set

Model	F1	Recall	Accuracy
RNN_CNN [35]	33.29	36.62	30.53
Last_MaxPool	33.17	37.11	29.98
Last3_Avg._Embedding	34.43	38.22	31.32
Last3_Attention [36]	34.79	38.82	31.51
Last3_MulDropout[7]	35.30	37.74	33.16
Last3_Atten_MulDrop	36.39	41.23	32.56

Table 6 Online submitted results of different baselines in ICLR test set

Model	Average	F1	BLEU-1[16]	BLEU-4[16]	Dist.-1	Dist.-2
Transformer [4]	16.18	19.24	34.32	20.33	0.70	6.31
GPT2-Entity [5]	19.58	24.49	40.31	21.13	0.88	11.07
HRED-Entity [5]	20.43	27.00	43.89	23.27	0.77	7.21
BertGPT-Entity [25]	21.98	28.81	46.68	25.11	0.79	8.51
T5-pegasus-base-Entity [27]	23.60	28.76	48.43	26.77	0.84	13.21
EFMDG	25.49	29.58	50.48	28.03	1.23	18.13

Table 7 Performance of different methods in both CCKS-A and CCKS-B test sets

Model	CCKS A				CCKS B			
	Average	F1	BLEU-average	Dist.-average	Average	F1	BLEU-average	Dist.-average
Transformer [4]	12.08	24.71	6.09	5.44	11.29	22.12	6.44	5.32
GPT2-Entity [5]	13.43	25.75	7.30	7.23	12.41	24.41	5.81	7.01
HRED-Entity [5]	13.85	26.42	7.37	7.75	13.11	25.11	6.61	7.61
BertGPT-Entity [25]	13.79	26.57	7.03	7.78	13.69	26.74	6.66	7.69
T5-pegasus-small-Entity [27]	13.42	23.76	9.34	7.15	13.10	24.49	7.71	7.11
T5-pegasus-base-Entity [27]	14.18	25.41	9.43	7.70	13.65	25.47	7.87	7.63
+ post pre-training[17]	14.53	25.55	9.61	8.44	14.29	25.63	8.66	8.58
CPM2-prompt [29]	15.21	26.38	10.04	9.21	15.76	27.10	10.78	9.41
EFMDG	17.67	30.14	12.49	10.37	18.27	30.66	13.17	10.99

With the increasing of model size, i.e., small to the base of the T5-pegasus-Entity, each test score shows an upward trend. As a result, the fine-tuned CPM2-prompt reaches the high averaged score of 15.21 and 15.76 in CCSK A/B tests respectively. It can be found that the proposed method outperforms the CPM2-prompt with the improvement of 2.46 and 2.51 averaged scores in CCSK A/B tests respectively, which indicates the effectiveness of the MDG.

5.5.3 *Manual evaluation results*

The human evaluation result is summarized in Table 8. The performance of the flattened inputs concatenated with predicted entities is better than the one without predicted entities, i.e., Transformer. The maximum length limits the quality of the generated response, i.e., GPT-2-Entity, as the same pre-trained model, obtains relatively low scores. As the size of the pre-trained model increases, all human indicators

15) https://github.com/ymcui/Chinese-BERT-wwm.

16) Since online submission results need to reflect the differences in the scores of each model, the original BLEU 1-4 score value is multiplied by the magnification factor.

17) https://github.com/ZhuiyiTechnology/t5-pegasus.

Table 8 Results of human evaluation, where κ is the average pairwise Cohen's kappa score between annotators

Model	Utterance fluency	Entity correctness	Entire quality
Transformer	3.17	3.01	3.13
GPT2-Entity	3.22	3.11	3.14
HRED-Entity	3.81	3.78	3.78
BertGPT-Entity	3.74	3.81	3.81
T5-pegasus-base-post	4.07	3.84	4.01
CPM2-prompt	4.11	4.19	4.17
EFMDG	4.15	4.21	4.20
Golden response	4.75	4.88	4.81
κ	0.52	0.59	0.57

Table 9 Performance of the proposed method with different structures

Fusion strategy	CCKS-A				CCKS-B			
	Average	F1	BLEU-average	Dist.-average	Average	F1	BLEU-average	Dist.-average
Backbone	15.77	28.72	10.12	8.21	15.49	28.34	9.99	8.14
B + Context embedding	16.42	28.98	11.30	9.21	17.45	29.52	12.54	10.31
B + C. + A.	16.76	29.36	11.50	9.43	17.67	29.87	12.72	10.42
B + C. + SLW.	16.86	29.44	11.66	9.49	17.76	29.93	12.81	10.55
B + C. + SDW.	16.99	29.67	11.77	9.54	17.82	29.99	12.87	10.60
B + C. + Max.	16.69	29.30	11.42	9.35	17.61	29.80	12.66	10.37
B + C. + Min.	16.54	29.12	11.37	9.14	17.48	29.66	12.61	10.17
B + C. + SDW. + EDBS	17.67	30.14	12.49	10.37	18.27	30.66	13.17	10.99
End2end (EFMDG)	16.46	28.97	10.88	9.53	16.70	29.14	11.18	9.77

have an upward trend, which is in line with the result of an automatic evaluation. From the results of different baselines, we can observe that the proposed method outperforms other baselines in all the manual metrics, which indicates that the response generated from the proposed EFMDG is more similar to the way humans respond. As a result, there is still a long way from the generated responses to the real responses of people. What is more, the average pairwise Cohen's kappa [37] scores between annotators range between 0.4 and 0.6 for all metrics, which represents a moderate annotator agreement.

5.6 Ablation studies

5.6.1 *The network architecture*

The ablation experiments are shown in Table 9. The backbone (i.e., B) is the original BertGPT-Entity implemented with the curriculum boost training strategy, which outperforms the original one. This shows that it is helpful for improving the final results to learn through the corpora of different difficulties. It can be found that the proposed context encoding module (i.e., C.) is beneficial to improve the BLEU score but decrease the F1 score, as history entities are equally important for response generation but external entities from the dialogue context may be noise when decoding. The encoding fusion module also improves F1, BLEU, and Distinct effectively, which represents that the information from different sources is beneficial for decoding.

5.6.2 *The fusion strategy*

We also compare different fusion strategies to verify the performance of the fusion mechanism. In Table 9, the source-dimension weighted (i.e., SDW.) strategy has the highest score in all metrics, outperforming the source-level weighted (i.e., SDL.) strategy. It may be that different dimension from the encoding needs different weights to balance during fusion. The maximum (i.e., Max.) and minimum (i.e., Min.) fusion strategies fail to outperform the average fusion (i.e., A.) strategy, which means that a single encoding has limited representation capability and cannot represent information from different sources. It is concluded that the weighted fusion strategy outperforms other fusion strategies because different encodings provide different information for decoding, thus weighted fusion is a better way to balance its information. Furthermore, it turns out that "B + C. + SDW. + EDBS" achieves the best performance in both datasets, which indicates that the proposed structure is necessary for generating better results.

Table 10 Performance of different decoding algorithms

Decoding algorithm	CCKS-A				CCKS-B			
	Average	F1	BLEU-average	Dist.-average	Average	F1	BLEU-average	Dist.-average
Greedy	16.99	29.67	11.77	9.54	17.82	29.99	12.87	10.60
Top_k ($k = 20$)	16.71	29.27	11.49	9.37	17.15	29.44	11.88	10.14
Top_p ($p = 0.9$) [20]	16.65	28.98	11.55	9.41	16.94	28.97	11.55	10.29
Top_k_p ($k = 20$, $p = 0.9$) [20]	16.92	29.54	11.66	9.57	17.58	30.05	12.21	10.49
Multinomial sampling	17.00	29.29	11.80	9.91	17.78	29.99	12.57	10.79
Beam search [22]	17.05	29.39	11.91	9.85	17.88	30.02	12.94	10.67
DBS [23]	17.25	29.65	12.06	10.04	18.04	30.33	12.99	10.79
EDBS	17.67	30.14	12.49	10.37	18.27	30.66	13.17	10.99

5.6.3 *The end2end setting*

It is interesting to see the experimental comparison results between the pipeline method and the end2end method. Comparing the pipeline results in the 8-th row (B + C. + SDW. + EDBS) and the end2end results in the 9-th row from Table 9, it can be found that the proposed two-stage pipeline system outperforms the BertGPT-Entity and the EFMDG method with the end-to-end setting. The reason may be that the ability to predict medical entities of the dialogue generation model is limited for the end2end setting, and the proposed system uses the external entity prediction model to better grasp medical knowledge.

5.6.4 *The decoding algorithm*

We also compare different decoding algorithms with the proposed EDBS. As shown in Table 10, the greedy search outperforms the top_p sampling in the BLEU score, but falls behind in the Dist. score. This represents that the result from the greedy search is more grammatical but less diversified. The multinomial sampling reaches a high Dist. score but fails to get high BLEU scores, as it lacks control conditions. The DBS outperforms the BS in all metrics, which can be a strong decoding baseline. It can be found that the EDBS method has advantages in both entity accuracy and response diversity compared to other methods.

5.6.5 *The convergence threshold τ*

The threshold τ is an important parameter for the EDBS algorithm, balancing the diversity and grammaticality of the generated responses. To investigate its effect on the performance of the proposed method, we conducted the ablation experiments and reported the results in Table 11, where the optimal value of the convergence threshold τ is 0.9. Further conclusions can be observed that (1) the EDBS can revise the entities in the generated response, so the results of the F1 score on the CCKS-A and CCKS-B datasets are not much different. (2) As the value of the convergence threshold τ increases, the BLEU-avg. scores decrease on both datasets while the Dist. score improves. It is because the polynomial function occupies the main advantage that the results are produced with more diversity but less grammaticality. (3) As the value of the convergence threshold τ decreases, the greedy search takes advantage, so the performance of the Dist.-avg. score shows a downward trend. Once the threshold reaches 0, the greedy search dominates the main decoding step. It can be found that without the balancing of the convergence threshold τ, the performance of diversity and grammaticality is relatively poor on both datasets.

5.7 Decoding time and time complexity

Extensive experiments are conducted for evaluating the decoding speed and time complexity of different methods, which is shown in Table 12. As for the decoding speed, it can be seen that the greedy method obtains the fastest decoding speed. Top_k method performs the sorting steps, so the time is slower than the greedy method. Top_p method spends more than in cutting off the pre-defined exceeded probability. Top_k_p combines the above two methods, so the decoding time is slower. The multinomial method sample from a multinomial distribution, where more time will be spent in selecting the diversified probability. As for the beam search decoding method, it costs more time in calculating the local optimal results within the beams. The DBS splits each beam into groups, which costs much time in selecting diversified optimal. The proposed method balances the grammaticality and diversity controlled by the threshold.

Table 11 Ablation study of convergence threshold τ

Value of τ	CCKS-A				CCKS-B			
	Average	F1	BLEU-avg.	Dist.-avg.	Average	F1	BLEU-avg.	Dist.-avg.
50.0	17.25	29.97	11.07	**10.73**	18.00	30.44	12.33	**11.22**
20.0	17.31	30.00	11.24	10.70	18.04	30.46	12.47	11.20
15.0	17.38	30.03	11.44	10.67	18.06	30.47	12.55	11.17
10.0	17.45	30.05	11.79	10.52	18.06	30.48	12.57	11.13
5.0	17.55	30.09	12.11	10.45	18.11	30.52	12.73	11.07
1.0	17.63	30.11	12.37	10.40	18.22	30.57	13.05	11.04
0.9*	**17.67**	**30.14**	**12.49**	10.37	**18.27**	**30.66**	13.17	10.99
0.8	17.64	30.12	12.45	10.35	18.23	30.59	**13.18**	10.91
0.7	17.53	30.08	12.31	10.21	18.10	30.54	12.94	10.76
0.5	17.34	30.05	11.99	9.97	17.99	30.51	12.81	10.58
0.2	17.17	30.04	11.73	9.74	17.91	30.49	12.73	10.52
0.1	17.11	30.01	11.65	9.69	17.87	30.42	12.69	10.51
0.0	17.04	29.99	11.51	9.63	17.85	30.41	12.67	10.47

Table 12 Decoding speed and time complexity of different methods, where the N is the required traversal times, and notations B and G represent the beam width and group number in the beam respectively

Method	Decoding speed (tokens/s)	Time complexity
Greedy	13.1	$O(N)$
Top_k ($k = 20$)	12.7	$O(N)$
Top_p ($p = 0.9$) [20]	11.9	$O(N)$
Top_k_p ($k = 20$, $p = 0.9$) [20]	11.1	$O(N)$
Multinomial sampling	10.4	$O(N)$
Beam search [22]	9.1	$O(NB)$
DBS [23]	8.3	$O(N(B/G)(G-1))$
EDBS	7.9	$O((N-1)B)$

It is worthwhile getting the best evaluation performance at the expense of decoding time. As for the time complexity shown in Table 12, it can be found that the time complexity of the proposed method is $O((N-1)B)$, where the N is the required traversal times and the B is the beam width. The further conclusion can be obtained that the time perplexity of the greedy for the other methods is $O(N)$, which is the same as the top_k, top_p, top_k_p and multinomial sampling method. The beam search needs to traverse each beam (B) for the time complexity of $O(NB)$. The DBS method requires more time on calculating the diversity of each group (G) in the beam. The proposed EDBS is a heuristic method, which takes more time in keeping the grammaticality and diversity during the decoding step by introducing the convergence threshold τ.

5.8 Case studies

Three scripts of the dialogue are provided in Figure 6, where we carefully observe the performance of different models through the cases of responses generated from seven different models. It can be seen that when the number of predictive entities is accurate, the generated responses of different methods are not much different. As the predicted entities deviate from the ground truth, the shortcomings of responses generated by different methods gradually appear. The Transformer model generates a response without any medical entities. The models, including GPT2-Entity, HRED-Entity, BERT-GPT-Entity, and T5-pegasus-base-post, generate responses that lack predicted entities. It can be further inferred that responses generated by a larger model have stronger diversity ability but less controllability. There is a deviation from the predicted entities, the CPM2-prompt generates responses deviating from the predicted entities, i.e., "mucus stool" (黏液便) and "bloody stool" (血便). Our proposed method is more faithful to the predicted entities, generating responses conditioned with the predicted entities. As for the lack of predicted entities, there are wrong or missing entities in responses generated by other methods. The proposed model can generate responses containing revised sentences with predicted information to the maximum extent. Compared to the other methods, the responses generated from the proposed are more correct, controllable, and rich in medical information than other methods. Note that although the

Dialogue context	Generated response	
U_0：总是拉肚子，肚子咕噜噜叫，这几天拉稀严重，请问为什么？ U_0：I always have diarrhea, my stomach rumbling, and severe diarrhea these days, why? Golden Response：您好，请问出现这种情况有多长时间了呢？ Golden Response：Hello, how long has it been since this happened? Predicted entity: Duration（时长） Golden entity: Duration（时长）	Transformer	您好，这种情况多长时间了？ Hello, how long has this situation been?
	GPT2-Entity	您好，这种情况有多久了？ Hello, how long has this situation been?
	HRED-Entity	您好，这种情况多久了？ Hello, how long has this been happening?
	BERTGPT-Entity	您好，这种情况多长时间了？ Hello, how long has this situation been?
	T5-pegasus-base-post	您好，这种情况多长时间了？ Hello, how long has this situation been?
	CPM2-prompt	您好，请问您这种情况多长时间了？ Hello, how long have you been in this situation?
	EFMDG(Our Method)	您好，您出现这种情况多长时间了？ Hello, how long have you been in this situation?
U_2：我拉肚子大概两周了。 U_2：I have had diarrhea for about two weeks. Golden Response：具体哪个部位疼痛？疼之前吃的辛辣刺激食物了？还有没有恶心呕吐腹胀反酸等症状？ Golden Response：Which part of the pain is specific? Did the spicy food that you ate before hurting? Are there any symptoms such as nausea, vomiting and acid reflux? Predicted entity: Location, Nausea, Vomiting, Reflux, Bloating, Heartburn (位置、恶心、呕吐 、反流、腹胀、烧心) Golden entity: Location, Nausea, Vomiting, Reflux (位置、恶心、呕吐 、腹胀、反流)	Transformer	疼多久？ How long does it hurt?
	GPT2-Entity	您是哪个部位疼？ Which part do you hurts?
	HRED-Entity	有没有烧心、反酸、腹胀？ Do you have heartburn, acid reflux and bloating?
	BERTGPT-Entity	具体哪里疼？最近吃什么了？ Where does it hurt? What have you eaten recently?
	T5-pegasus-base-post	您是哪个部位疼？有没有吃辣的？ Which part of you hurts? Are there any spicy ones?
	CPM2-prompt	有没有吃辣的食物，有没有呕吐、发热、腹痛、腹胀、黏液便、血便等症状？ Have you eaten spicy food and have you been vomiting, fever, abdominal pain, bloating, mucus stool and bloody stool, etc.?
	EFMDG(Our Method)	具体哪里疼，有没有反酸、烧心、恶心、呕吐、腹胀等症状？ Where does it hurt? Do you have symptoms such as acid reflux, heartburn, nausea, vomiting and bloating?
.....		
U_6：请问医生我该怎么做才能恢复好？ U_6：May I ask the doctor what should I do to recover from the illness well? Golden Response：像您的这种情况还是考虑肠道的问题，至于有没有炎症？还需要做一个大便常规的检测来看一下。不过根据现在的症状，我建议首先注意饮食，适当的活动，增强体质，凉的不要吃，需要多喝温水，补充液体！可以服用金双歧等肠道益生菌调节肠道功能。 Golden Response：In your case, you still consider the intestinal problems. As for whether there is inflammation? Also need to do a routine stool test to take a look. However, based on the current symptoms, I suggest that you first pay attention to your diet, appropriate activities, strengthen your physical fitness, do not eat cold, you need to drink more warm water and supplement fluids! You can take intestinal probiotics such as Golden Bifidus to regulate intestinal function. Predicted entity: Stool routine, Golden Bifidus (便常规、金双歧) Golden entity: Stool routine, Golden Bifidus, Probiotics (便常规、金双歧、益生菌)	Transformer	建议您注意饮食。 It is recommended that you pay attention to your diet.
	GPT2-Entity	您的肠道有问题，多喝温水很关键。 There is a problem with your intestines, it is important to drink plenty of warm water.
	HRED-Entity	考虑到您这种情况应该是肠道有问题，建议做个大便常规。 Considering that your situation should be due to intestinal problems, it is recommended to do a stool routine.
	BERTGPT-Entity	考虑到您现在的症状，可以服用金双歧，需要控制饮食多运动。 Considering your current symptoms, you can take Golden Bifidus, and you need to control your diet and exercise more.
	T5-pegasus-base-post	鉴于您现在的情况，可口服金双歧缓解，具体什么原因还需要进一步检查之后才能判断。 In view of your current situation, you can take orally Golden Bifidus to relieve, the specific reason needs to be further checked before you can judge.
	CPM2-prompt	像您这种情况建议注意饮食，但不确定有没有炎症，建议服用金双歧调节肠道。 In your case, it is recommended to pay attention to your diet, but you are unsure whether there is inflammation. It is recommended to take Golden Bifidus to regulate the intestinal tract.
	EFMDG(Our Method)	考虑您现在这种情况应该是肠道的问题，可以口服金双歧清淡饮食，多运动。建议做个大便常规。 Considering that your current situation should be an intestinal problem, you can take Golden Bifidus orally, and it is recommended to make a stool routine.

Figure 6 Examples of three dialogue scripts. U_i represents the i-th utterance of the dialogue. Due to the space limitation, we only present the first two dialogues and the last dialogue as the example, where the correct predicted entities are underlined.

predicted entity may be biased, the generated sentence still has medical significance, as the dialogue generation model can only fit the sub-optimal results in the corpus.

We also provide the histogram of predicted entities of script 2 of Figure 6, which are shown in Figure 7. It can be concluded that after searching for the optimal f1 threshold of 0.55, each predicted entity score greater than 0.55 in the histogram will be selected for the response generation. The optimal F1 threshold is selected by considering the co-occurrence probability of entities in the medical dialogue dataset, thus indicating that both precision and recall are optimal. Further observation shows that although there will be redundant entities such as "heartburn" (with the predicted probability of 0.5671), it is still in

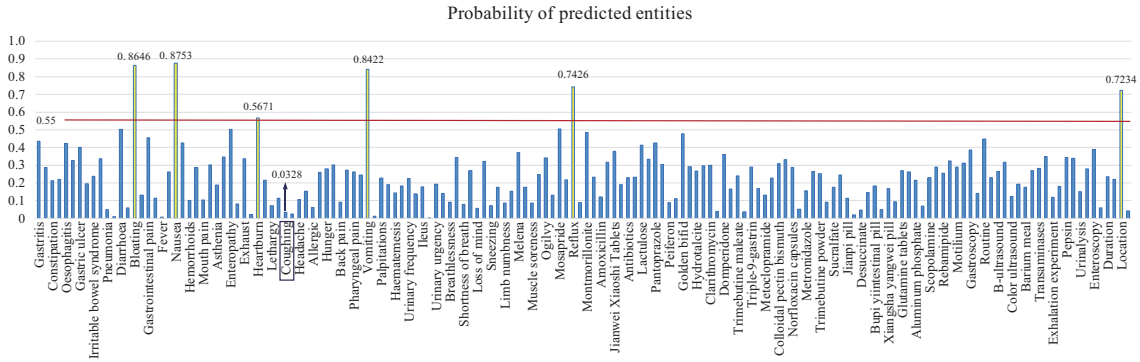

Figure 7 (Color online) Histogram of the predicted entities of script 2.

Table 13 Results of CCKS competition

Rank	Test A		Test B	
	Team name	Score	Team name	Score
1	VPAI_Lab	21.24432	VPAI_Lab	21.8283
2	ParlAI	20.65078	sfeng	21.4424
3	sfeng	20.00431	iseesaw	21.0741
4	iseesaw	19.03350	White_jingling	20.0248
5	vivo	17.46282	Running snails	19.0828

Table 14 Results of ICLR workshop competition

Rank	Phase A					Phase B				
	Team name	Score	F1	BLEU1	BLEU4	Team name	Score	F1	BLEU1	BLEU4
1	VPAI_Lab	32.43	23.67	47.90	25.73	VPAI_Lab	36.03	29.58	50.48	28.03
2	jwanglvy	32.09	18.07	49.66	28.53	jackey	35.28	31.72	49.09	25.04
3	zxxflyfish	31.08	21.51	47.27	24.46	LHS	34.24	25.89	48.23	28.60

line with medical common sense. Irrelevant entities like "Cough" (framed in blue) would indicate a low probability and are unlikely to be selected, so the MDG will not contain these entities. The conclusions are drawn that the redundant entities may appear in the responses, but the irrelevant entities are much more unlikely to be contained with the proposed method.

For the case of the actual performance, we present the competition results shown in Tables 13 and 14. The proposed method won both the 2021 CCKS entity-aware MDG competition and the 2021 ICLR workshop machine learning for preventing and combating pandemics (MLPCP) Track 1 competition. These results demonstrate that the proposed method is effective and solid.

6 Conclusion and future work

In this paper, we propose a pipeline framework for MDG. The framework consists of two parts: medical entity prediction and entity-aware dialogue generation. In our framework, we first optimize the entity prediction model post pre-trained with the ent-mac optimizing with F1 threshold search, then utilize the predicted entities with the proposed encoding fusion mechanism, which controls the information from different sources. We improve the original DBS with the entity-revised method, which proves to be effective in improving the quality of the final response. The proposed method outperforms other competitive baselines in the CCKS-A/B and the ICLR online test sets, demonstrating the effectiveness and practicality of the proposed method. In the future, we will consider using the knowledge graph to infer the predicted entity and try different encoding fusion mechanisms with medical knowledge entities, to further improve the correctness and quality of the generated response.

Acknowledgements This work was supported by National Key Research and Development Project (Grant No. 2018YFB1305200), National Natural Science Foundation of China (Grant No. 62171183), and Project of Hunan Provincial Health Commission (Grant No. 202114010841).

References

1 Zhang S H, Cai Y, Li J. Visualization of COVID-19 spread based on spread and extinction indexes. Sci China Inf Sci, 2020, 63: 164102
2 Wei Z Y, Liu Q L, Tou B L, et al. Task-oriented dialogue system for automatic diagnosis. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, 2018. 201–207
3 Xu L, Zhou Q X, Gong K, et al. End-to-end knowledge-routed relational dialogue system for automatic diagnosis. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2019. 7346–7353
4 Zeng G T, Yang W M, Ju Z Q, et al. Meddialog: a large-scale medical dialogue dataset. In: Proceedings of Conference on Empirical Methods in Natural Language Processing, 2020. 9241–9250
5 Liu W G, Tang J H, Qin J H, et al. MedDG: a large-scale medical consultation dataset for building medical dialogue system. 2020. ArXiv:2010.07497
6 Lin S, Zhou P, Liang X D, et al. Graph-evolving meta-learning for low-resource medical dialogue generation. 2020. ArXiv:2012.11988
7 Jiang F, Jiang Y, Zhi H, et al. Artificial intelligence in healthcare: past, present and future. Stroke Vasc Neurol, 2017, 2: 230–243
8 Xia Y, Zhou J B, Shi Z H, et al. Generative adversarial regularized mutual information policy gradient framework for automatic diagnosis. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2020. 1062–1069
9 Cui Y M, Che W X, Liu T, et al. Revisiting pre-trained models for Chinese natural language processing. In: Proceedings of Conference on Empirical Methods in Natural Language Processing: Findings, 2020. 657–668
10 Rogers A, Kovaleva O, Rumshisky A. A primer in bertology: what we know about how BERT works. Trans Assoc Comput Linguist, 2020, 8: 842–866
11 Qiu X P, Sun T X, Xu Y G, et al. Pre-trained models for natural language processing: a survey. Sci China Tech Sci, 2020, 63: 1872–1897
12 Han X, Zhang Z Y, Ding N, et al. Pre-trained models: past, present and future. AI Open, 2021, 2: 225–250
13 Devlin J, Chang M W, Lee K, et al. BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2019. 4171–4186
14 Cui Y M, Che W X, Liu T, et al. Pre-training with whole word masking for chinese bert. 2019. ArXiv:1906.08101
15 Mikolov T, Sutskever I, Chen, K, et al. Distributed representations of words and phrases and their compositionality. In: Proceedings of Advances in Neural Information Processing Systems, 2013. 3111–3119
16 Zhang Z Y, Han X, Liu Z Y, et al. ERNIE: enhanced language representation with informative entities. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019. 1441–1451
17 Han S Y, Zhang Y H, Ma Y S, et al. THUOCL: Tsinghua Open Chinese Lexicon. 2016. http://thuocl.thunlp.org/
18 Liao K B, Liu Q L, Wei Z Y, et al. Task-oriented dialogue system for automatic disease diagnosis via hierarchical reinforcement learning. 2020. ArXiv:2004.14254
19 Kulikov I, Miller A H, Cho K, et al. Importance of search and evaluation strategies in neural dialogue modeling. In: Proceedings of the 12th International Conference on Natural Language Generation, 2019. 76–87
20 Fan A, Lewis M, Dauphin Y. Hierarchical neural story generation. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, 2018. 889–898
21 Holtzman A, Buys J, Du L, et al. The curious case of neural text degeneration. 2019. ArXiv:1904.09751
22 Cohen E, Beck C. Empirical analysis of beam search performance degradation in neural sequence models. In: Proceedings of International Conference on Machine Learning PMLR, 2019. 1290–1299
23 Vijayakumar A K, Cogswell M, Selvaraju R R, et al. Diverse beam search: decoding diverse solutions from neural sequence models. 2016. ArXiv:1610.02424
24 Gururangan S, Marasović A, Swayamdipta S, et al. Don't stop pretraining: adapt language models to domains and tasks. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020. 8342–8360
25 Lewis M, Liu Y, Goyal N, et al. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020. 7871–7880
26 Brown T B, Mann B, Ryder N, et al. Language models are few-shot learners. 2020. ArXiv:2005.14165
27 Raffel C, Shazeer N, Roberts A, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. J Mach Learn Res, 2020, 21: 5485–5551
28 Zhang J Q, Zhao Y, Saleh M, et al. Pegasus: pre-training with extracted gap-sentences for abstractive summarization. In: International Conference on Machine Learning, 2020. 11328–11339
29 Zhang Z Y, Gu Y X, Han X, et al. CPM-2: large-scale cost-effective pre-trained language models. 2021. ArXiv:2106.10715
30 Miyato T, Dai A M, Goodfellow I. Adversarial training methods for semi-supervised text classification. 2016. ArXiv:1605.07725
31 Drucker H, Cortes C, Jackel L D, et al. Boosting and other ensemble methods. Neural Comput, 1994, 6: 1289–1301
32 Loshchilov I, Hutter F. Fixing weight decay regularization in ADAM. 2017. ArXiv:1711.05101
33 Chen B, Cherry C. A systematic comparison of smoothing techniques for sentence-level BLEU. In: Proceedings of the 9th Workshop on Statistical Machine Translation, 2014. 362–367
34 Li J W, Galley M, Brockett C, et al. A diversity-promoting objective function for neural conversation models. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2016. 110–119
35 Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014. 580–587
36 Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. In: Proceedings of Advances in Neural Information Processing Systems, 2017. 5998–6008
37 Randolph J J. Free-Marginal Multirater Kappa (multirater K): an alternative to Fleiss' fixed-marginal multirater Kappa. 2005. https://eric.ed.gov/?id=ED490661