Extractive Medical Entity Disambiguation with Memory Mechanism and Memorized Entity Information

Guobiao Zhang[1,4] , Xueping Peng[2] , Tao Shen[2] , Guodong Long[2] , Jiasheng Si[1,4] , Libo Qin[3] , Wenpeng Lu[1,4]*

[1]Key Laboratory of Computing Power Network and Information Security, Ministry of Education, Shandong Computer Science Center, Qilu University of Technology (Shandong Academy of Sciences)

[2]Australian Artificial Intelligence Institute, University of Technology Sydney, Sydney, Australia

[3]School of Computer Science and Engineering, Central South University, Changsha, China

[4]Shandong Provincial Key Laboratory of Computing Power Internet and Service Computing, China

guobiao.zhang@foxmail.com, {wenpeng.lu,jiashengsi}@qlu.edu.cn

Abstract

Medical entity disambiguation (MED) aims to ground medical mentions in text with ontological entities in knowledge bases (KBs). A notable challenge of MED is the long medical text usually contains multiple entities' mentions with intricate correlations. However, limited by computation overhead, many existing methods consider only a single candidate entity mention during the disambiguation process. As such, they focus only on local MED optimal while ignoring the sole-mention disambiguation possibly boosted by richer context from other mentions' disambiguating processes – missing global optimal on entity combination in the text. Motivated by this, we propose a new approach called Extractive Medical Entity Disambiguation with **M**emory **M**echanism and **M**emorized **E**ntity Information (**M^{3}E**). Specifically, we reformulate MED as a text extraction task, which simultaneously accepts the context of medical mentions, all possible candidate entities, and entity definitions, and it is then trained to extract the text span corresponding to the correct entity. Upon our new formulation, 1) to alleviate the computation overhead from the enriched context, we devise a memory mechanism module that performs memory caching, retrieval, fusion and cross-network residual; and 2) to utilize the disambiguation clues from other mentions, we design an auxiliary disambiguation module that employs a gating mechanism to assist the disambiguation of remaining mentions. Extensive experiments on two benchmark datasets demonstrate the superiority of M^{3}E over the state-of-the-art MED methods on all metrics[1].

1 Introduction

Associating medical mentions in a given biomedical text with their corresponding correct entities from a reference knowledge base (KB) is a historical and challenging task in natural language processing (NLP) and biomedical domains (French and McInnes, 2023; Zhu et al., 2023; Kartchner et al., 2023), formally known as medical entity disambiguation (MED). For instance, consider the following sentence: "*Insulinum is an important factor in the treatment of DM*". The medical mention "*DM*" could refer to the entity "*diabetes mellitus*" or "*diabetic microangiopathy*" in UMLS (Bodenreider, 2004). The MED system should accurately map the mention "*DM*" to the corresponding entity "*diabetes mellitus*". In addition, MED has extensive applications in diverse downstream tasks of medical NLP, including medical question answering (Bae et al., 2024), medical dialogue (Valizadeh and Parde, 2022; Priya et al., 2023), medical information extraction (Landolsi et al., 2023).

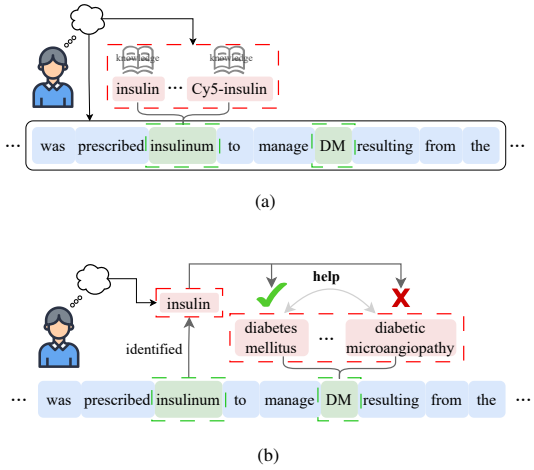

Figure 1: (a) When encountering the medical mention "insulinum" in the sentence, humans will simultaneously consider the context of mention, all possible candidate entities such as "insulin", "Cy5-insulin", and the semantic knowledge of these entities to determine the correct entity "insulin". (b) Once the medical mention "Insulinum" is identified as "insulin", humans will utilize it as semantic clues to further infer the entity of the mention "DM" as "diabetes mellitus".

In recent years, a series of disambiguation meth-

* Corresponding author

[1]The source code and datasets of this paper can be obtained from https://github.com/Stubborn-z/MMME.

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 13811–13822

ods have been proposed in the literature of deep representation learning. They can be roughly categorized into matching and generative MED, improving benchmark performance in a row. First, MED was defined as a matching problem between medical mentions and candidate entities. As such, some works (Zhu et al., 2020; Lu et al., 2024) utilized dual encoders to separately encode medical mentions and candidate entities, adopting attention mechanisms to enhance the interaction between them. However, these methods only consider a single candidate entity for a medical mention and do not fully utilize semantic knowledge in KBs. To mitigate this, another research line presents generative MED methods(Yuan et al., 2022a,b) with sacrifice of efficiency, which utilize generative pre-trained language models, inputting medical mentions with context and outputting corresponding entities through pre-training and fine-tuning.

Nonetheless, these generative MED methods still suffer from a lack of context information as they only consider the context of the current medical mention, while ignoring richer context from other mentions' disambiguating processes. For example, as shown in Figure 1(a), when encountering a medical mention in a sentence, humans simultaneously consider the context of the mention, all possible candidates and the semantic knowledge of entities, focusing on global information to comprehensively understand the whole context and determine the correct entity. What's worse, these methods usually disambiguate each medical mention individually without considering the semantic clues when disambiguating other entities. As such, they focus only on local MED optimization, overlooking the potential benefits of richer context from the disambiguation processes of other mentions. This results in missing the global optimization of entity combinations within the text. Continue the example above: as depicted in Figure 1(b), humans utilize the memorized entity information as disambiguation clues to further infer the remaining medical mentions. Therefore, the disambiguation pattern of existing MED methods is inconsistent with human cognitive behaviors.

Inspired by the extractive entity disambiguation in the general domain (Barba et al., 2022) and the human cognitive behaviors in Figure 1, we propose a new paradigm that reformulates MED as a medical text extraction task. Unlike previous medical entity disambiguation methods, we introduce a memory mechanism module to alleviate the computational burden of our newly formulated MED task, enabling the model to simultaneously consider the context of medical mentions, all possible candidate entities and the semantic knowledge of candidates during the disambiguation process. Additionally, we propose an auxiliary disambiguation module that leverages the semantic disambiguation clues from memorized entities, i.e., the richer knowledge from the disambiguation process of other mentions to assist in disambiguating the remaining medical mentions. Our approach achieves globally optimal disambiguation by focusing on global MED optimization

Our main contributions are three-fold:

- We reformulate MED as an extractive disambiguation task for both effective and efficient purposes, leading to a new MED method called extractive medical entity disambiguation with memory mechanism and memorized entity information (M^3E).
- To preserve the efficiency with enriched context, we propose a memory mechanism module that performs memory caching, retrieval and fusion to alleviate computational burdens for our newly formulated MED. This facilitates the proposed auxiliary disambiguation module that leverages semantic disambiguation clues from memorized entities to assist in disambiguating the remaining mentions.
- We conduct experiments on two public benchmark datasets and verify the superiority of our approach through comparisons with representative and existing state-of-the-art works.

2 Related Work

Matching MED. Matching MED methods treat MED as a matching problem, which usually encodes medical mentions and candidate entities separately and then evaluates their matching score to predict correct entities. Early methods (Francis-Landau et al., 2016; Deng et al., 2019) employ CNN (Kim, 2014) to capture the semantic similarity between source documents and candidate entities. However, they often focus solely on the representations of contexts and entities, neglecting to leverage the knowledge from medical KBs and overlooking the modeling of interactions between medical mentions and entities. To address these deficiencies, LATTE (Zhu et al., 2020) introduces latent type knowledge as auxiliary supervision and

employs a cross-attention mechanism to model intrinsical interactions. Although simple and effective, it overlooks fine-grained entity-entity interactions. Prompt-BioEL(Xu et al., 2023) first employs a bi-encoder initialized with SAPBERT (Liu et al., 2021) to jointly learn representations of medical mentions and candidate entities to retrieve potential candidates and then adopts a re-ranking model based on prompt tuning with entity-entity interactions to identify the correct entity.

Generative MED. Generative MED methods usually utilize generative language models pre-trained on various medical corpora as fundamental architectures, and finetune them to infer correct entities of medical mentions. For example, BioBART (Yuan et al., 2022a) is a biomedical autoregressive generative language model pre-trained on PubMed abstracts, which is further fine-tuned on the medical entity disambiguation task, yielding promising results. GenBioEL (Yuan et al., 2022b) first proposes KB-guided pre-training to inject synonyms and definition knowledge into the generative language model and then proposes synonym-aware fine-tuning to select correct entities for mentions.

3 Methodology

3.1 Task Definition

Inspired by recent trends of entity disambiguation in the general domain (Barba et al., 2022), we formulate MED as a medical text extraction task: given a context with medical mentions, all possible candidate entities along with their definitions from medical KBs, a model has to extract the text span corresponding to the correct entity's definition. Formally, let $\mathbb{C} = \{w_1, \ldots, w_l\}$ be a medical context consisting of l words, which includes N ambiguous mentions, denoted as $\mathbb{M} = \{m_1, \ldots, m_N\}$. For the i-th mention m_i in $\mathbb{M}$, according to medical KBs, its potential candidate entities and their definitions are expressed as $\mathbb{E} = \{e_1, \ldots, e_n\}$, $\mathbb{D} = \{d_1, \ldots, d_n\}$, where n represents the number of candidate entities. We concatenate the medical context $\mathbb{C}$, all possible candidate entities $\mathbb{E}$ and entity definitions $\mathbb{D}$ together, and train the model to extract the text span $[i_e, j_e]$ corresponding to the correct entity of the mention m_i.

3.2 Model Architecture

The overall architecture of the proposed M^{3}E model is shown in Figure 2, comprising four core components: *knowledge augmenting module*, *memory mechanism module*, *auxiliary disambiguation module* and *prediction module*. First, to comprehensively understand the medical text, the *knowledge augmenting module* simultaneously accepts the context of medical mentions, all potential candidate entities, along with their definitions from UMLS. This inevitably increases the input length. Then, to alleviate the computational burdens caused by the increased input length, the *memory mechanism module* performs memory caching, retrieval, fusion and cross-network residual. Subsequently, the *auxiliary disambiguation module* employs a gating mechanism to leverage the memorized entities as semantic clues to assist in disambiguating remaining medical mentions. Finally, the *prediction module* extracts the text span corresponding to the correct entity of the target mention. Next, we will describe these four modules in detail.

3.3 Knowledge Augmenting Module

Some recent works have verified that semantic knowledge in UMLS is crucial for enhancing the representation of medical mentions and candidate entities (Zhu et al., 2020; Zhang et al., 2022). Therefore, in order to mimic human cognitive behavior shown in Figure 1(a) and provide sufficient semantic knowledge to our model, we devise a knowledge augmenting module to simultaneously accept the context of medical mentions, all potential candidate entities along with their definitions from UMLS. In detail, we augment candidate entities by incorporating the definition of entities from UMLS, and then concatenate the context of medical mentions, all possible candidate entities, and their entity definitions together as the input of our model, described as:

$$\begin{aligned} I = \texttt{<s>}\, w_1 \ldots \texttt{<t>}\, m_i \,\texttt{</t>}\, \ldots w_l \,\texttt{</x>} \\ e_1, d_1 \ldots e_n, d_n \,\texttt{</s>} \end{aligned} \tag{1}$$

where <s> and </s> are the special symbols that surround the entire input, <t> and </t> are the special symbols that surround the target medical mention m_i, and </x> is a special symbol that separates the context with candidate entities and their definitions.

3.4 Memory Mechanism Module

In order to efficiently encode the input text I, we employ a medical pre-trained language model, as it excels in understanding medical text at a deeper

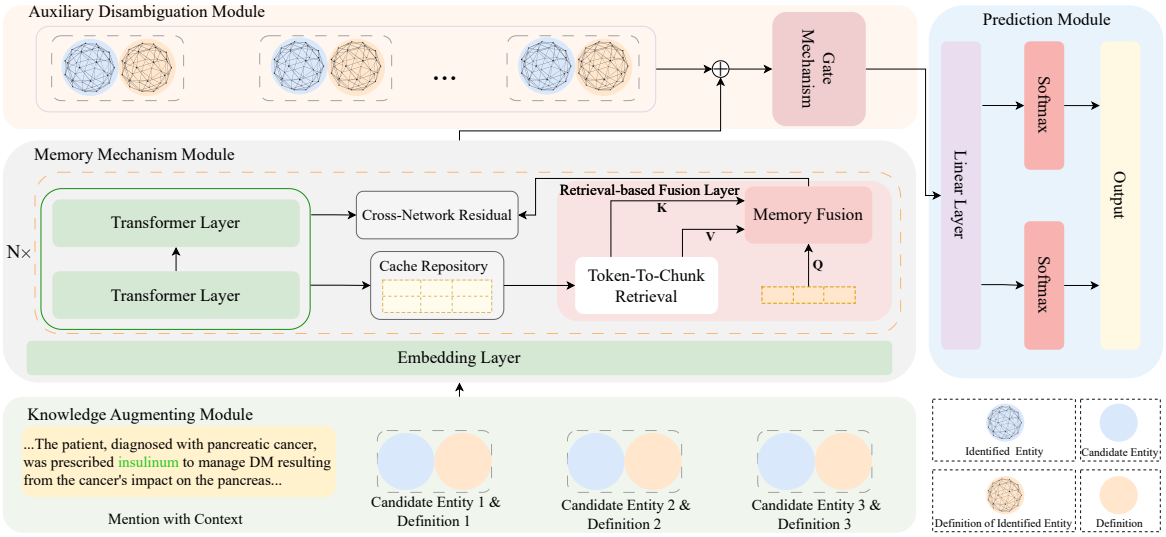

Figure 2: The architecture of the proposed M^{3}E model, which consists of four core modules: (1) knowledge augmenting module, which simultaneously accepts the context of medical mentions, all potential candidate entities along with their definitions from UMLS; (2) memory mechanism module, which performs memory caching, retrieval, fusion and cross-network residual connections to alleviate the computational burdens; (3) auxiliary disambiguation module, which employs a gate mechanism to leverage memorized entities as semantic clues to disambiguate the remaining medical mentions; (4) prediction module, which extracts the text span corresponding to the correct entity of the target mention.

level and performs well in long sequence modeling. Nevertheless, as we simultaneously feed the context of medical mentions, and all potential candidate entities along with their definitions from UMLS into our model, the length of the input text increases significantly, resulting in substantial memory requirements and computational burdens. To address this challenge, drawing inspiration from LongMem (Wang et al., 2023), we propose a memory mechanism module, which consists of memory caching, retrieval, fusion and cross-network residual.

3.4.1 Memory Caching

As shown on the memory mechanism module in Figure 2, we employ the memory embedding layer to encode the input I and acquire the hidden state representation $\mathbf{H}_m^0$:

$$\mathbf{H}_m^0 = \{\mathbf{h}_f^1, \ldots, \mathbf{h}_f^u\} = \text{EmbeddingLayer}(I) \quad (2)$$

where $\mathbf{H}_m^0 \in \mathbb{R}^{d \times u}$, d represents the dimension of each hidden state and u represents the number of hidden units. We freeze the pre-trained language model, allowing the transformer layers to perform forward passes without conducting any gradient calculations after obtaining $\mathbf{H}_m^0$. During the forward propagation of the model, the memory mechanism module cache the key-value pairs of self-attention, which represent the context tokens (i.e., the current input context information) adjacent to the current input token, into a cache repository for memory caching. This process avoids recalculating the same context information in each forward propagation step, significantly reducing computational redundancy.

3.4.2 Memory Retrieval

Subsequently, the retrieval-based fusion layer retrieves the relevant key-value pairs from the cache repository through token-to-chunk retrieval. Specifically, we divide the cache repository into a certain number of chunks, with each chunk containing a fixed number of attention key-value pairs. Mean pooling is then conducted along the chunk size dimension for each chunk to generate a mean-pooled vector for retrieval. Next, the dot product between the query vector of the current input token and the mean-pooled vector of each candidate chunk is computed. Based on the dot product value, the memory chunk most relevant to the current input token is retrieved to prepare for subsequent fusion operations. Further, the chunks of self-attention key-value pairs are linearly projected to the attention matrices $\mathbf{K}$, $\mathbf{V}$ through two weight matrices $\mathbf{W}^K$, $\mathbf{W}^V$ respectively.

$$\mathbf{K} = \mathbf{W}^K(\text{TokenToChunk}(\mathbf{H}_m^0)) \quad (3)$$

$$\mathbf{V} = \mathbf{W}^V(\text{TokenToChunk}(\mathbf{H}_m^0)) \quad (4)$$

where $\mathrm{TokenToChunk}$ represents the above token-to-chunk retrieval operation, which can greatly speed up the retrieval process.

3.4.3 Memory Fusion

The retrieval-based fusion layer employs the attention mechanism to achieve memory fusion, which enables each token to attend to both current contexts and retrieved memory contexts:

$$\begin{aligned}\mathbf{H}^1_{fus} = \mathrm{sigmoid}(g) \cdot (\mathrm{softmax}(\frac{\mathbf{Q} \cdot \mathbf{K}^{\mathrm{T}}}{\sqrt{\mathbf{d}_k}})\mathbf{V}) \\ + (1 - \mathrm{sigmoid}(g)) \cdot \mathbf{H}^0_m\end{aligned} \tag{5}$$

where $\mathbf{H}^1_{fus}$ is the output of the first retrieval-based fusion layer, g is a trainable gating vector, $\mathbf{Q}$ is obtained by linearly projecting the $\mathbf{H}^0_m$ through the query matrix $\mathbf{W}^Q$, $\mathrm{sigmoid}$ and $\mathrm{softmax}$ are activation function.

Following a single memory caching, retrieval and fusion, it deletes the attention key-value pair at the front of the queue and adds the current attention key-value pair to the tail of the queue. This mechanism facilitates repository updates, ensuring that the cache repository always preserves the latest context for the current input.

3.4.4 Cross-network Residual

In order to leverage the knowledge from the frozen pre-trained language model, we perform cross-network residual connections between transformer layers. Different from the residual operations in LongMem (Wang et al., 2023), our memory mechanism module uses adjacent transformer layers for residual connections, facilitating more comprehensive knowledge transfer. Additionally, we do not perform the concatenation operation as the most relevant contextual information has been considered in the iterative fusion process. In this way, the lightweight retrieval-based fusion layer achieves fast convergence of knowledge transferred from pre-trained parameters and can undergo continuous training in an efficient manner:

$$\mathbf{H}^1_m = \mathbf{H}^1_{fus} + (\mathbf{H}^2_t - \mathbf{H}^1_t) \tag{6}$$

where $\mathbf{H}^1_m$ is the memory representation obtained through retrieval-based fusion process. $\mathbf{H}^1_t$ and $\mathbf{H}^2_t$ represent the hidden state representation of the first transformer layer and the second transformer layer, respectively.

3.4.5 Iteration

The series of operations involved in the aforementioned retrieval-based fusion is formally defined as the function f_θ, as follows:

$$\mathbf{H}^1_m = f_\theta(\mathbf{H}^0_m) \tag{7}$$

We iterate the memory caching, retrieval, fusion and cross-network residual connection process multiple times, allowing the model to focus on global information and significantly alleviating the computational overhead:

$$\mathbf{H}^l_m = f_\theta(\mathbf{H}^{l-1}_m),\ \forall l \in [1, L] \tag{8}$$

where $\mathbf{H}^l_m$ represents the memory representation acquired through the l-th retrieval-based fusion. l represents the number of memory retrieval fusion processes. Finally, the final memory representation $\mathbf{H}_m$ is obtained via the memory mechanism module.

3.5 Auxiliary Disambiguation Module

To mimic human cognitive behavior shown in Figure 1(b), we devise an auxiliary disambiguation module that caches memorized entities and their definitions in the disambiguation process of other mentions to assist in disambiguating the remaining medical mentions. Specifically, the auxiliary disambiguation module caches the hidden state representation $\mathbf{H}_{ide}$ of the identified entities and entity definitions after completing the preceding disambiguation:

$$\mathbf{H}_{ide} = \{\mathbf{h}^{e_1}_{ide}, \mathbf{h}^{d_1}_{ide}, \ldots, \mathbf{h}^{e_i}_{ide}, \mathbf{h}^{d_i}_{ide}\} \tag{9}$$

where e_i is the i-th memorized entity, d_i is the i-th entity definition. Subsequently, the final hidden state representation is generated by concatenating $\mathbf{H}_{ide}$ and $\mathbf{H}_m$ together, and then passing through GRU (Chung et al., 2014), described as:

$$\mathbf{H} = \mathrm{GRU}([\mathbf{H}_{ide}; \mathbf{H}_m]) \tag{10}$$

3.6 Prediction Module

The prediction module extracts the text span corresponding to the correct entity of the target medical mention. First, it employs a linear layer to perform a linear transformation on $\mathbf{H}$:

$$\begin{aligned}\mathbf{Z} &= \mathbf{W}^{\mathsf{T}}\mathbf{H} + \mathbf{b} \\ \mathbf{Z}^s &= [\mathbf{Z}_{11}, \ldots, \mathbf{Z}_{1n}] \\ \mathbf{Z}^e &= [\mathbf{Z}_{21}, \ldots, \mathbf{Z}_{2n}]\end{aligned} \tag{11}$$

where $\mathbf{W} \in \mathbb{R}^{d\times 2}$ and $\mathbf{b} \in \mathbb{R}^2$ are trainable parameters. $\mathbf{Z}^s$ and $\mathbf{Z}^e$ represent the logits of each token, indicating whether it is the start or the end of the text span corresponding to the correct entity of the target mention.

Subsequently, it feeds the logits of $\mathbf{Z}^s$ and $\mathbf{Z}^e$ into softmax to obtain the probability distribution, and then performs a product operation on the probability distributions of the start and end position to generate the probability of the pair $[i_e, j_e]$, described as:

$$\mathbf{P}[i_e] = \text{softmax}(\mathbf{Z}_s) \tag{12}$$

$$\mathbf{P}[j_e] = \text{softmax}(\mathbf{Z}_e) \tag{13}$$

$$\mathbf{P}[i_e, j_e] = \mathbf{P}[i_e] \times \mathbf{P}[j_e] \tag{14}$$

where $\mathbf{P}[i_e]$, $\mathbf{P}[j_e]$ represent the probability that i_e is the correct starting position or j_e is the correct ending position, respectively. $\mathbf{P}[i_e, j_e]$ represents the probability of a medical text span in the input text starting at i_e and ending at j_e (i.e., the probability that a medical mention corresponds to the correct entity).

Finally, the prediction module outputs the pair with the maximum probability score:

$$s = \text{argmax}(\mathbf{P}[i_e, j_e]) \tag{15}$$

3.7 Training

We train the proposed M^{3}E model by summing two cross-entropy losses calculated at the start and end position, described as:

$$\mathcal{L}_s = -\mathbf{Z}^s_{i_e} + \log \sum_v^n \exp(\mathbf{Z}^s_v) \tag{16}$$

$$\mathcal{L}_e = -\mathbf{Z}^e_{j_e} + \log \sum_v^n \exp(\mathbf{Z}^e_v) \tag{17}$$

$$\mathcal{L} = \mathcal{L}_s + \mathcal{L}_e \tag{18}$$

where $\mathcal{L}_s$ and $\mathcal{L}_e$ are the losses of the starting position and the ending position of the candidate entity, respectively. $\mathcal{L}$ is the total loss. $\mathbf{Z}^s_{i_e}$ and $\mathbf{Z}^e_{j_e}$ are the scores corresponding to the correct start and end positions.

4 Experiment

4.1 Datasets and Baselines

We evaluate M^{3}E using three public benchmark datasets: MedMentions[2], BC5CDR[3] and NCBI Disease[4]. The details of the datasets are shown in Appendix A.1.

To evaluate the performance of M^{3}E, we compare our method with representative methods as well as state-of-the-art approaches. Detailed descriptions of the baselines are provided in Appendix A.2.

4.2 Implementation Details

We implement the proposed M^{3}E model using the PyTorch framework and adopt BioBART (Yuan et al., 2022a) as the base model. We also employ the $base$ and $large$ variants of pre-trained weights from HuggingFace library, called M^{3}E$_{base}$ and M^{3}E$_{large}$. More implementation details are provided in Appendix A.3.

4.3 Main Results

We compare our approach with representative and state-of-the-art baselines. The experimental results are summarized in Table 1. According to the table, we have the following observations.

Firstly, among the Matching MED methods, BIOSYN exhibits inferior performance compared to later approaches. This discrepancy arises due to subsequent methods incorporating additional knowledge information from medical KBs and enhancing the interaction not only between medical mentions and candidate entities but also between entities. Among the Generative MED methods, BART yields lower results compared to the latter methods. This is because the subsequent methods use a large amount of medical knowledge for pretraining on medical corpora. The aforementioned observations prove that integrating more semantic information into the model is crucial for MED.

Secondly, on the three public benchmark datasets, our M^{3}E model consistently outperforms all competitive baselines on all evaluation metrics. The superiority of our M^{3}E model can be attributed to two key factors. On one hand, our model reformulates MED as a text extraction task, which can simultaneously accept medical contexts, candidate entities and their definitions, comprehensively understanding the context to judge the correct entities. On the other hand, our model utilizes a memory mechanism module to alleviate the model burden due to increased input length, while leveraging memorized entities to assist in disambiguating the remaining mentions.

[2]https://github.com/chanzuckerberg/MedMentions

[3]https://www.ncbi.nlm.nih.gov/research/bionlp/Data/

[4]https://www.ncbi.nlm.nih.gov/

	Model	MedMentions			BC5CDR			NCBI Disease		
		Precision	Recall@1/@5	F1	Precision	Recall@1/@5	F1	Precision	Recall@1/@5	F1
Matching MED	BIOSYN (Sung et al., 2020)	67.12	66.42/77.50	68.49	70.10	68.84/82.75	72.21	74.61	77.98/82.10	72.53
	LAATE (Zhu et al., 2020)	88.23	86.55/88.94	85.61	88.28	87.00/88.56	86.37	90.53	89.87/91.00	87.91
	Zhu (Zhu et al., 2021)	69.52	68.14/80.41	67.48	88.29	88.01/89.33	87.10	89.85	90.12/92.03	88.14
	Cross-Domain (Varma et al., 2021)	68.85	73.12/80.71	78.22	88.74	90.00/91.50	86.80	90.21	90.48/91.10	88.13
	B-LBConA (Yang et al., 2023)	88.51	87.16/89.20	86.47	89.19	88.24/90.93	87.37	90.88	90.98/92.07	89.70
	Prompt-BioEL (Xu et al., 2023)	89.16	86.85/88.67	87.65	90.27	91.26/92.20	88.63	91.11	91.38/92.00	91.46
Generative MED	BART_{base} (Lewis et al., 2020)	64.25	68.61/82.29	67.46	86.15	88.18/89.49	85.34	88.21	89.80/90.70	86.19
	BART_{large} (Lewis et al., 2020)	68.16	70.00/82.85	68.14	88.21	89.47/90.60	85.83	89.15	90.61/91.00	88.52
	BioBART_{base} (Yuan et al., 2022a)	81.63	80.11/83.25	80.58	89.10	89.81/91.20	86.21	90.18	91.24/91.50	90.61
	BioBART_{large} (Yuan et al., 2022a)	82.81	81.21/84.24	82.37	90.07	89.77/91.22	87.27	91.49	91.00/92.22	90.15
	BioGPT (Luo et al., 2022)	81.96	80.82/82.51	81.46	89.20	89.84/91.85	86.38	89.90	91.01/92.26	88.42
	ClinicalT5_{base} (Lu et al., 2022)	81.18	80.13/82.50	80.89	89.44	90.71/91.39	85.80	90.12	91.21/91.89	88.40
	$\text{ClinicalT5}_{large}$ (Lu et al., 2022)	83.94	80.54/83.00	82.05	90.51	90.88/92.11	85.08	91.44	91.09/92.25	89.71
	GenBioEL (Yuan et al., 2022b)	88.56	88.19/89.54	87.74	91.14	91.08/92.30	88.64	92.23	92.82/93.00	91.27
Our	$\textbf{M}^3\textbf{E}_{base}$	89.21	88.40/90.04	88.56	92.01	91.29/92.30	89.51	93.01	92.41/92.91	92.16
	$\textbf{M}^3\textbf{E}_{large}$	**89.60**	**88.87/90.14**	**88.71**	**92.26**	**91.43/92.39**	**89.70**	**93.15**	**93.18/93.30**	**92.28**
	Improvement(%)	1.04	0.68/0.6	0.97	0.92	0.35/0.09	1.06	1.22	0.36/0.31	1.01

Table 1: Performance on the MedMentions, BC5CDR and NCBI disease datasets in comparison with the SOTA models.It is worth noting that we take the average of the 5 experimental performances as our final result. We mark in **bold** the best scores and underline the suboptimal one. The improvement is calculated against the $\text{M}^3\text{E}_{large}$ and best-performing baseline (GenBioEL), and the difference in performance between $\text{M}^3\text{E}_{large}$ and GenBioEL is statistically significant ($p < 0.01$).

4.4 Ablation Study

To investigate the importance of each key component of M^3E, we perform an ablation study by comparing the performance of M^3E and that of its variants. We consider the following variants: i) $\textbf{M}^3\textbf{E}^{-de}$: It removes the definitions of entities in the knowledge augmenting module, considering only the medical context of mentions and their candidate entities; ii) $\textbf{M}^3\textbf{E}^{-Me}$: It removes the memory mechanism module, relying solely on BioBART to encode the long input text segments; iii) $\textbf{M}^3\textbf{E}^{-Au}$: It removes the auxiliary disambiguation module, disregarding the semantic clues associated with the memorized entities. The experimental results are presented in Figure 3, and several noteworthy observations can be drawn. Firstly, comparing M^3E with M^3E^{-de} verifies the importance of entity definitions, as the latter performs inferiorly. Secondly, comparing M^3E with M^3E^{-Me} highlights the indispensability of the memory mechanism module, as the latter exhibit inferior performance. Lastly, comparing M^3E with M^3E^{-Au} reveals a significant performance gap, emphasizing the crucial role of memorized entities in providing important clues for disambiguating the remaining mentions.

4.5 Fine-grained Results on Frequency-specific Entities

To investigate the performance of our model on the entities with different frequencies, we conduct a fine-grained analysis on frequency-specific entities. We create three subsets[5] from MedMentions and BC5CDR datasets based on entity frequencies: i) **MFE**, which contains all the instances in the test set that the target medical mention is associated with its most frequent entity in the training corpus; ii) **LFE**, which contains all the instances in the test set that the target medical mention is associated with its least frequent entity in the training corpus; iii) **Unseen**, which contains all the instances in the test set that the target medical mention is never seen in the training corpus.

Model	MedMentions			BC5CDR			NCBI		
	MFE	LFE	Unseen	MFE	LFE	Unseen	MFE	LFE	Unseen
GenBioEL	93.61	50.15	78.00	93.40	52.15	81.10	93.40	52.15	81.10
$\text{M}^3\text{E}_{base}$	93.82	56.16	82.61	94.21	56.80	83.44	94.21	56.80	83.44
$\text{M}^3\text{E}_{large}$	**93.82**	**58.53**	**84.39**	**95.68**	**59.38**	**88.12**	**94.21**	**57.24**	**86.32**

Table 2: F1 score of GenBioEL, $\text{M}^3\text{E}_{base}$ and $\text{M}^3\text{E}_{large}$ on MFE, LFE and Unseen datasets created from MedMentions, BC5CDR and NCBI respectively. The best scores are marked in **bold**.

In Table 2, we report the results of the three best-performing models: GenBioEL, $\text{M}^3\text{E}_{base}$ and $\text{M}^3\text{E}_{large}$. Firstly, it is observed that the F1 score of all three models exceeds 93 on the MFE dataset, indicating their excellent disambiguation abilities on frequent entities. Secondly, $\text{M}^3\text{E}_{base}$ and $\text{M}^3\text{E}_{large}$ perform significantly better than GenBioEL on both LFE and Unseen datasets. Specifically, on

[5]The manually created dataset is available at https://github.com/Stubborn-z/MMME/data/subsets/.

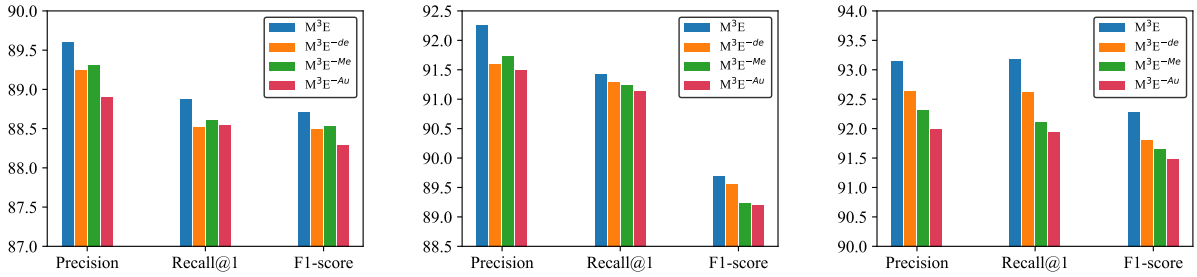

Figure 3: Ablation experimental results on MedMentions (left), BC5CDR (middle) and NCBI (right) datasets.

the LFE and Unseen datasets, M^3E_{large} achieves approximately 8 and 5 points higher F1 scores than GenBioEL, respectively. This highlights the strong generalization ability of our proposed method, demonstrating its effectiveness in handling rare or unseen candidate entities.

4.6 Complexity and Efficiency Analysis

To investigate the complexity of our model, we conduct a fair comparison of its parameter size and memory requirement with existing methods on the MedMentions dataset. To analyze the efficiency, we compare its training and inference time with existing methods on the three benchmark datasets. The detailed experiments are provided in Appendix A.4.

4.7 Error Analysis

To investigate the proposed M^3E model further and gain insights for future work, we conduct an error analysis on the test sets across three datasets and discover errors caused by insufficient fine-grained knowledge.

Input	...specific cytotoxic effectors as a potential remedy for... effector T cells refer to cells, a subset of T lymphocyte... cytotoxic effectors refer to cells or molecules that... ...
Gold Entity	effector T cells
Predicted Entity	cytotoxic effector

Table 3: A representative instance for error analysis. We mark the medical mention in , the golden entity and its definition in / , the predicted entity and its definition in / .

According to the statistical results, out of the 5714 medical mentions of the test set, M^3E makes incorrect disambiguation on 241 instances. We conduct a detailed analysis on some representative instances. As shown in Table 3, we observe that the two candidate entities related to the medical mention *"effectors"*, namely *"effector T cells"* and *"cytotoxic effector"*, share similar meanings and entity definitions, leading to our model being unable to utilize sufficient differentiation information for accurate disambiguation. To this end, we strongly believe that future research might benefit from focusing on enriching entities' information by adding fine-grained knowledge.

5 Conclusion

In this paper, we propose a novel MED framework called extractive medical entity disambiguation with Memory Mechanism and Memorized Entity Information (M^3E). Our new paradigm reformulates MED as a text extraction task, which simultaneously accepts the context of mentions, all possible candidate entities and entity definitions. To alleviate the computational burden caused by the increased input length, we devise a memory mechanism module, which performs memory caching, retrieval, fusion, and cross-network residual connections to enhance the model's efficiency by capturing and utilizing global context information more effectively. Additionally, we implement an auxiliary disambiguation module, which leverages memorized entities' semantic clues in the disambiguation process of other mentions, aiding in the disambiguation of the remaining mentions. Therefore, our method achieves globally optimal disambiguation by emphasizing global MED optimization. Extensive experimental results on three public benchmark datasets demonstrate that M^3E consistently outperforms the representative and state-of-the-art MED baselines. Notably, our model achieves an F1 score approximately 1 point higher than the suboptimal baseline on all datasets, showing great potential for further exploration.

Limitations

Although our work is the first to adopt the extractive medical entity disambiguation paradigm and has achieved great success, significantly improving performance compared to matching and generative disambiguation methods, it has the following limitations. First, our model still struggles to distinguish between similar entities, whose ability to capture subtle difference between candidate entities remains limited. Second, due to budget constraints, we have not conducted comparative experiments with large-scale medical language models. Therefore, improving the model's ability to distinguish subtle differences between similar entities and exploring the impact of large medical language models on MED tasks are very interesting directions for future work.

Ethical Statement

Medical entity disambiguation (MED) is essential in natural language processing and biomedical domains. It supports various downstream medical applications, aiding in medical-related decision-making, enhancing medical information extraction, and improving the accuracy of medical question answering. We believe that the potential for misuse of this MED technology is low. Our technology is developed using publicly available datasets, adhering the data use guidelines and ensuring no copyright infringement.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No.62376130), Shandong Provincial Natural Science Foundation (Grant No.ZR2022MF243), Program of New Twenty Policies for Universities of Jinan (Grant No.202333008), Program of Innovation Improvement of Shandong (Grant No.2023TSGC0182 and Grant No.2023TSGC0274), the Key Laboratory of Computing Power Network and Information Security, Ministry of Education (Grant No.2023ZD032), and Pilot Project for Integrated Innovation of Science, Education, and Industry of Qilu University of Technology (Shandong Academy of Sciences) (Grant No.2024ZDZX08 and Grant No.2021JC02010).

References

Seongsu Bae, Daeun Kyung, Jaehee Ryu, Eunbyeol Cho, Gyubok Lee, Sunjun Kweon, Jungwoo Oh, Lei Ji, Eric Chang, Tackeun Kim, et al. 2024. Ehrxqa: A multi-modal question answering dataset for electronic health records with chest x-ray images. *Advances in Neural Information Processing Systems*, 36.

Edoardo Barba, Luigi Procopio, and Roberto Navigli. 2022. Extend: Extractive entity disambiguation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 2478–2488.

Olivier Bodenreider. 2004. The Unified Medical Language System (UMLS): Integrating biomedical terminology. *Nucleic Acids Research*, 32(suppl_1):D267–D270.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Allan Peter Davis, Thomas C Wiegers, Michael C Rosenstein, and Carolyn J Mattingly. 2012. Medic: A practical disease vocabulary used at the comparative toxicogenomics database. *Database*, 2012:bar065.

Pan Deng, Haipeng Chen, Mengyao Huang, Xiaowen Ruan, and Liang Xu. 2019. An ensemble cnn method for biomedical entity normalization. In *Proceedings of the 5th workshop on BioNLP open shared tasks*, pages 143–149.

Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. Ncbi disease corpus: a resource for disease name recognition and concept normalization. *Journal of Biomedical Informatics*, 47:1–10.

Matthew Francis-Landau, Greg Durrett, and Dan Klein. 2016. Capturing semantic similarity for entity linking with convolutional neural networks. In *Proceedings of the 15th Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1256–1261.

Evan French and Bridget T McInnes. 2023. An overview of biomedical entity linking throughout the years. *Journal of Biomedical Informatics*, 137:104252.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. In *Proceedings of the 9th International Conference on Learning Representations*, pages 20–41.

David Kartchner, Jennifer Deng, Shubham Lohiya, Tejasri Kopparthi, Prasanth Bathala, Daniel Domingo-Fernández, and Cassie S Mitchell. 2023. A comprehensive evaluation of biomedical entity linking models. In *Proceedings of the 28th Conference on Empirical Methods in Natural Language Processing*, page 14462.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 19th Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751.

Mohamed Yassine Landolsi, Lobna Hlaoua, and Lotfi Ben Romdhane. 2023. Information extraction from electronic medical documents: State of the art and future research directions. *Knowledge and Information Systems*, 65(2):463–516.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. 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*, pages 7871–7880.

Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. 2016. BioCreative V CDR task corpus: A resource for chemical disease relation extraction. *Database*, 1:10.

Fangyu Liu, Ehsan Shareghi, Zaiqiao Meng, Marco Basaldella, and Nigel Collier. 2021. Self-alignment pretraining for biomedical entity representations. In *Proceedings of the 22nd Conference of the North American Chapter of the Association for Computational Linguistics*, pages 4228–4238.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. 2020. On the variance of the adaptive learning rate and beyond. In *Proceedings of the 9th International Conference on Learning Representations*.

Qiuhao Lu, Dejing Dou, and Thien Nguyen. 2022. ClinicalT5: A generative language model for clinical text. In *Findings of the Association for Computational Linguistics*, pages 5436–5443.

Wenpeng Lu, Guobiao Zhang, Xueping Peng, Hongjiao Guan, and Shoujin Wang. 2024. Medical entity disambiguation with medical mention relation and fine-grained entity knowledge. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation*, pages 11148–11158.

Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. 2022. BioGPT: Generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics*, 23(6).

Sunil Mohan and Donghui Li. 2019. MedMentions: A large biomedical corpus annotated with UMLS concepts. In *Proceedings of the 1st Automated Knowledge Base Construction*, pages 1–13.

Priyanshu Priya, Kshitij Mishra, Palak Totala, and Asif Ekbal. 2023. PARTNER: A persuasive mental health and legal counselling dialogue system for women and children crime victims. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence*, pages 6183–6191.

Mujeen Sung, Hwisang Jeon, Jinhyuk Lee, and Jaewoo Kang. 2020. Biomedical entity representations with synonym marginalization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3641–3650.

Mina Valizadeh and Natalie Parde. 2022. The AI doctor is in: A survey of task-oriented dialogue systems for healthcare applications. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 6638–6660.

Maya Varma, Laurel Orr, Sen Wu, Megan Leszczynski, Xiao Ling, and Christopher Ré. 2021. Cross-domain data integration for named entity disambiguation in biomedical text. In *Findings of the Association for Computational Linguistics*, pages 4566–4575.

Weizhi Wang, Li Dong, Hao Cheng, Xiaodong Liu, Xifeng Yan, Jianfeng Gao, and Furu Wei. 2023. Augmenting language models with long-term memory. *Advances in Neural Information Processing Systems*, 36.

Zhenran Xu, Yulin Chen, and Baotian Hu. 2023. Improving biomedical entity linking with cross-entity interaction. In *Proceedings of the 37th AAAI Conference on Artificial Intelligence*, pages 13869–13877.

Siyu Yang, Peiliang Zhang, Chao Che, and Zhaoqian Zhong. 2023. B-LBConA: A medical entity disambiguation model based on bio-linkbert and context-aware mechanism. *BMC Bioinformatics*, 24(1):1–18.

Hongyi Yuan, Zheng Yuan, Ruyi Gan, Jiaxing Zhang, Yutao Xie, and Sheng Yu. 2022a. BioBART: Pretraining and evaluation of a biomedical generative language model. In *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 97–109.

Hongyi Yuan, Zheng Yuan, and Sheng Yu. 2022b. Generative biomedical entity linking via knowledge base-guided pre-training and synonyms-aware fine-tuning. In *Proceedings of the 23rd Conference of the North American Chapter of the Association for Computational Linguistics*, pages 4038–4048.

Sheng Zhang, Hao Cheng, Shikhar Vashishth, Cliff Wong, Jinfeng Xiao, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2022. Knowledge-Rich self-supervision for biomedical entity linking. In *Findings of the Association for Computational Linguistics*, pages 868–880.

Ming Zhu, Busra Celikkaya, Parminder Bhatia, and Chandan K Reddy. 2020. LATTE: Latent type modeling for biomedical entity linking. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, pages 9757–9764.

Tiantian Zhu, Yang Qin, Qingcai Chen, Baotian Hu, and Yang Xiang. 2021. Enhancing entity representations with prompt learning for biomedical entity linking. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence*, pages 4036–4042.

Tiantian Zhu, Yang Qin, Qingcai Chen, Xin Mu, Changlong Yu, and Yang Xiang. 2023. Controllable contrastive generation for multilingual biomedical entity linking. In *Proceedings of the 28th Conference on Empirical Methods in Natural Language Processing*, pages 5742–5753.

A Appendix

A.1 Dataset Summary and Statistics

- **MedMentions**: It is the largest biomedical entity disambiguation dataset, consisting of 4,392 abstracts from PubMed, with over 350,000 mentions linked to UMLS concepts (Mohan and Li, 2019).
- **BC5CDR**: It consists of 1,500 articles from PubMed, with 4,409 annotated chemicals and 5,818 diseases, containing over 28,000 mentions linked to MeSH concepts. In addition, we map these mentions to UMLS concepts (Li et al., 2016).
- **NCBI Disease**: It provides manually annotated disease mentions in each document, with each Concept Unique Identifier (CUI) mapped into the MEDIC dictionary (Davis et al., 2012). Following the work of BIOSYN (Sung et al., 2020), we adopt the consistent version of MEDIC, which contains 11,915 CUIs and 71,923 synonyms from the MeSH and OMIM ontologies (Doğan et al., 2014). Additionally, we map these mentions to UMLS concepts.

The detailed statistical information is provided in Table 4.

Dataset	Statistics	Train	Dev	Test
MedMentions	#Documents	2,635	878	879
	#Mentions	211,029	71,062	70,405
	#Entities	20,830	6,941	6,953
BC5CDR	#Documents	900	300	300
	#Mentions	17,135	5,710	5,714
	#Entities	5,489	1,830	1,830
NCBI Disease	#Documents	592	100	100
	#Mentions	5,134	787	960
	#Entities	1,924	480	482

Table 4: Statistics of experimental datasets.

A.2 Descriptions of Baselines

We category baselines into two groups: matching MED and generative MED.

A.2.1 Matching MED

- **BIOSYN**: It uses iterative candidate selection together with synonym marginalization techniques to optimize the representation of medical entity (Sung et al., 2020).
- **LATTE**: It introduces latent type knowledge and employs a cross-attention mechanism to model interactions betweenmentions and entities (Zhu et al., 2020).
- **Zhu**: It proposes a two-stage algorithm to enhance entity representations using prompt learning and leveraging contextual knowledge (Zhu et al., 2021).
- **B-LBConA**: It uses Bio-LinkBERT to encode medical mentions and entities, while using a bidirectional attention mechanism to capture the interactive information between them (Yang et al., 2023).
- **Cross-Domain**: It introduces a cross-domain data integration method to transfer general knowledge into the medical domain (Varma et al., 2021).
- **Prompt-BioEL**: It proposes a prompt learning-based re-ranking method that simultaneously represents the context and all candidate entities (Xu et al., 2023).

A.2.2 Generative MED

- **BART, BioBART, BioGPT** and **ClinicalT5**: They are all representative medical pretrained language model, which are adopted as baselines and are finetuned on our datasets (Lewis et al., 2020; Yuan et al., 2022a; Luo et al., 2022; Lu et al., 2022).
- **GenBioEL**: It introduces the injection of synonyms and definition knowledge into the pretraining and finetuning of biomedical language models (Yuan et al., 2022b).

A.3 Experiment Parameters

To ensure a fair comparison between our model and the baselines, we finetuned the parameters of all models consistently on the validation dataset. Specifically, we initialized the parameters of each baseline model according to the experimental settings from the original paper, and then finetuned them on the MED validation dataset to achieve optimal performance. Similar to the previous work of

EXtEnD (Barba et al., 2022), we use the output of the last decoder to represent the input tokens and calculate the probability distributions of the start and end tokens. Additionally, we finetune M^{3}E with the RAdam (Liu et al., 2020) optimizer with a learning rate set to $2e^{-5}$ for at most 200,000 steps, and a gradient clipping of 1.0 (He et al., 2020). We employ 10 steps of gradient accumulation and a batch size consisting of 1024 tokens. We evaluate the model's performance on the validation dataset every 2000 steps, enforcing patience of 15 evaluation rounds. Our model was trained for 10 epochs using a GeForce RTX 3090 GPU, which required around 21 hours on the MedMentions dataset and approximately 6 hours on the BC5CDR dataset.

A.4 Complexity and Efficiency Analysis

To investigate the complexity of our model, we conduct a fair comparison of its parameter size and memory requirement with the best-performing baselines (i.e., Prompt-BioEL and GenBioEL) on the MedMentions dataset. Table 5 depicts experimental results.

Method	Parameter Size (M)	Memory Requirement (GB)
Prompt-BioEL	242	27.1
GenBioEL	416	36.2
M^{3}E$_{large}$	410	30.2

Table 5: Parameter Size and Memory Requirement (GB) of different methods on the MedMentions dataset.

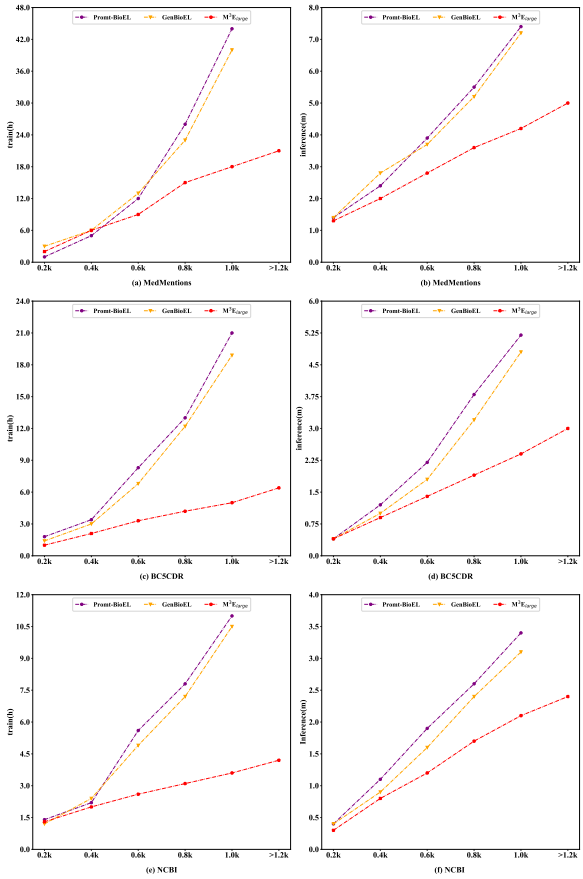

Figure 4: Efficiency comparison run on MedMentions, BC5CDR and NCBI dataset.

According to Table 5, as the representative of the Matching MED baselines, Prompt-BioEL requires the least memory with the smallest parameter size. Although Prompt-BioEL shows advantages in model complexity, its performance is inferior to generative methods. Both GenBioEL and M^{3}E$_{large}$ are generative MED methods, which can outperform Prompt-BioEL. Compared to GenBioEL, our M^{3}E$_{large}$ model achieves better performance with less parameters and memory requirement. This improvement is attributed to the memory mechanism module we proposed, which effectively alleviates the model's computational burden.

Additionally, we evaluate the efficiency of our M^{3}E model by comparing its training and inference times with existing methods on three benchmark datasets. The experimental results are reported in Figure 4. We have the following observations. Firstly, as the number of input tokens increases, the training and inference times of Prompt-BioEL, GenBioEL, and M^{3}E$_{large}$ increase to varying degrees across the three datasets. This is due to the greater computational burdens required as input length grows. Secondly, compared to Prompt-BioEL and GenBioEL, our M^{3}E$_{large}$ model exhibits a stable linear growth in training and inference times as the number of tokens increases, whereas Prompt-BioEL and GenBioEL tend to show a quadratic growth trend, demonstrating signification efficiency advantages of M^{3}E. This improvement is attributed to the memory mechanism module we proposed, which eases the model's burden through retrieval fusion operations and accelerates both training and inference. Lastly, once the number of input tokens reaches 1k, Prompt-BioEL and GenBioEL cannot function on the experimental workstation with a single GeForce RTX 3090 GPU, whereas our M^{3}E$_{large}$ model continues to run normally with only limited increases in training and inference times. This verifies that compared to Prompt-BioEL and GenBioEL, our model supports longer input token lengths, thanks to its robust memory mechanism module.