DictLLM: Harnessing Key-Value Data Structures with Large Language Models for Enhanced Medical Diagnostics

Yiqiu Guo[1,2], **Yuchen Yang**[2,4], **Ya Zhang**[2,3]✉, **Yu Wang**[2,3]✉, **Yanfeng Wang**[2,3]
[1]Fudan University [2]Shanghai AI Laboratory [3]Shanghai JiaoTong University
[4]University of Science and Technology of China

Abstract

Structured data offers a sophisticated mechanism for the organization of information. Existing methodologies for the text-serialization of structured data in the context of large language models fail to adequately address the heterogeneity inherent in key-value structured data. These methods are not ideal and frequently result in larger input sizes and poor adaptability to input changes. In this paper, we introduce DictLLM, an innovative framework designed to improve the modeling of key-value structured data, like medical laboratory reports, for generating medical diagnoses. DictLLM integrates three key components: (1) group positional encoding to maintain permutation invariance, (2) hierarchical attention bias to capture the inherent bias in structured data, and (3) an optimal transport alignment layer that aligns the embedding generated by the dictionary encoder with the LLM, thereby producing a sequence of fixed-length virtual tokens. We carry out experiments using various LLM models on a comprehensive real-world medical laboratory report dataset for automatic diagnosis generation, our findings illustrate that DictLLM significantly outperforms established baseline methods and few-shot GPT-4 implementations in terms of both Rouge-L and Knowledge F1 scores. Furthermore, our evaluation of the framework's scalability and robustness, through a series of experiments, underscores its exceptional capability in accurately modeling the complex key-value data structure of medical dictionary data.

1 Introduction

The integration of large language models (LLMs) into natural language processing (NLP) has marked a paradigm shift, enabling unprecedented advancements across diverse applications. Recent explorations into applying LLMs to structured data processing, such as graphs, dictionaries, and tables, highlight their potential beyond traditional text analysis. Notably, efforts like tabular data classification in Hegselmann et al. (2023), graph-based node classification in Tang et al. (2023) , and intelligent Excel table querying in KuB , have paved the way for innovative applications. Yet, the application of LLMs in processing medical lab reports, a cornerstone in clinical diagnostics, exposes significant challenges. These reports, structured as key-value pairs, are critical for diagnosis but diverge substantially from the data types traditionally handled by LLMs due to their unique structure and information content.

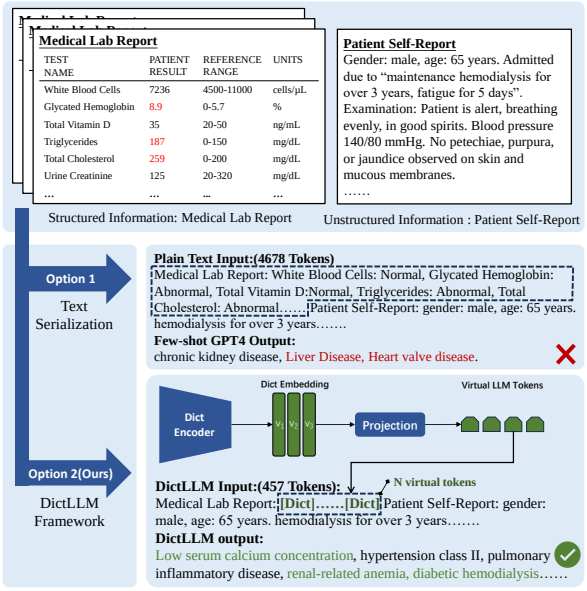


Figure 1: Our DictLLM Framework for medical lab report-assisted diagnosis generation. The framework uses a hierarchical dict encoder to encode the medical lab report, and an optimal transport alignment layer to align the embedding generated by the dict encoder and the text encoder.

Medical lab reports are pivotal in clinical decision-making, capturing patient test results in a structured format that facilitates diagnosis. Unlike

✉: Corresponding author.

Findings of the Association for Computational Linguistics ACL 2024, pages 10231–10241
August 11-16, 2024 ©2024 Association for Computational Linguistics

the linear, narrative flow of natural language, these reports are characterized by two distinct features:

- Structural Heterogeneity: They are organized as key-value pairs, allowing for permutation invariance where the sequence of entries does not affect the informational content.
- Information Density Heterogeneity: These reports encapsulate densely packed, discrete data, contrasting with the more continuous and narrative nature of text.

Existing methods, primarily based on converting structured data into a linear token sequence, inadequately capture these nuances. Such serialization not only risks losing structural fidelity but also scales poorly due to token limits in LLMs, highlighting a critical gap in current methodologies.

DictLLM emerges as a novel framework tailored to address these challenges, marrying the structured precision of medical lab reports with the analytical depth of LLMs. By innovatively leveraging a hierarchical dict encoder inspired by advancements in set transformation, DictLLM transcends traditional serialization approaches. It introduces a dict tokenizer to convert complex numerical data into interpretable medical labels, a group positional encoding to maintain the inherent permutation invariance of lab report data, and hierarchical attention mechanisms to adeptly handle the reports' information density.

Our contributions offer a significant leap forward in medical diagnostics:

- We introduce a hierarchical dict encoder that adeptly models the structured nature of medical lab reports, preserving their key-value integrity and enhancing robustness to variations in report formatting.
- The introduction of an optimal transport alignment layer aligns dict encoder embeddings with LLM outputs, optimizing the efficiency of input representation and addressing the challenge of token count scalability.
- Comparative analysis with leading LLMs on a comprehensive dataset of real-world medical lab reports demonstrates DictLLM's superior performance, showcasing notable improvements in Rouge-L and Knowledge F1 scores, indicative of its enhanced diagnostic accuracy and relevance extraction capabilities.

In aligning closely with the medical diagnostic process's intricacies, DictLLM not only highlights the untapped potential of LLMs in processing structured medical data but also sets a new benchmark for precision and efficiency in automated medical diagnosis. This approach not only underscores the framework's novelty but also its practical significance, promising to bridge the gap between current LLM capabilities and the complex demands of healthcare diagnostics.

2 Related work

2.1 Tabular data representation learning

Tabular data representation learning aims to learn a dense representation for tabular data. Deng et al. (2020) introduces the Masked Entity Recovery (MER) objective for pre-training the Table Encoder, aiming to capture the semantics and knowledge in large-scale unlabeled data. Yang et al. (2022) highlights that linearizing table structures would encode the order of the table's rows and columns with an unwanted bias. Chen et al. (2023) introduces a hypergraph-enhanced table representation learning framework to model the inherent inductive bias of tabular structures. Ye et al. (2023) introduce cross-table pretraining into the tabular data representation learning, to capture the cross-table knowledge. Du et al. (2022) propose learning enhanced representations for tabular data via neighborhood propagation. These study highlights the importance of modeling the structural properties of tabular data. However, these approaches do not harness the capabilities of large language models and are not designed to explicitly capture the heterogeneity of medical lab reports.

2.2 Large language model for structural data

With the emergence of large language models Touvron et al. (2023) Zeng et al. (2022) Mialon et al. (2021), there have been numerous efforts to leverage them for processing structured data tasks. Han et al. (2023) propose ChartLlama, a multimodal llava-based model for chart understanding and generation task. Hegselmann et al. (2023) introduce TabLLM, an text serialization-based framework that leverages LLMs for data-efficient tabular classification. However, this approach can only handle small-scale classification tasks, which is not suitable for generation tasks. Ope (2023)propose OpenTab, an open-domain end-to-end table reasoning framework, which leverages a retriever to fetch

relevant tables, employs a coder to generate programs as intermediary reasoning steps, and assigns the task of deriving the final solution to a reader. However, the retrieval-augmented paradigm can be limited by the performance of the retrieval module, especially for tasks requiring specific domain knowledge. Zhu et al. (2024) propose TAT-LLM, A specialized language model for discrete reasoning over tabular and textual data, serve as a pioneering example of specializing smaller language models for specific tasks. The GraphGPT proposed by Tang et al. (2023) comes closest to our work. This method employs a graph encoder and a text encoder to encode the structural information and the textual information of the graph and propose a dual-stage graph instruction tuning paradigm. Our work distinguishes itself from these studies by focusing on the design of a carefully designed hierarchical dict encoder to model the heterogeneous structure of medical lab reports.

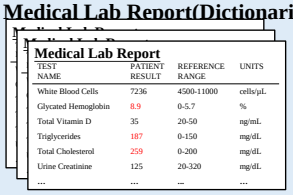
3 Approach

3.1 Problem Formalization

Task: Report-Assisted Diagnosis Generation

Input

Medical Lab Report(Dictionaries):



Medical Lab Report

TEST NAME	PATIENT RESULT	REFERENCE RANGE	UNITS
White Blood Cells	7236	4500-11000	cells/μL
Glycated Hemoglobin	8.9	0-5.7	%
Total Vitamin D	35	20-50	ng/mL
Triglycerides	187	0-150	mg/dL
Total Cholesterol	259	0-200	mg/dL
Urine Creatinine	125	20-320	mg/dL
...	...	...	...

Patient's Self-Report(Text):
Gender: Female Age: 82 Main symptoms and signs at admission: Asthma after activity for more than 1 year, aggravated for 2 months……

Output

Final Diagnosis(Text):
Renal insufficiency, moderate anemia, pulmonary inflammatory disease……

Figure 2: An example of the input and output of the medical lab report-assisted diagnosis generation task.

As shown in Figure 2, the task of report-assisted diagnosis generation involves creating a diagnosis based on a patient's self-reported symptoms and medical laboratory reports. Suppose we have a patient's medical laboratory report. We can formalize this report as a set of dictionaries, denoted as $\mathcal{D} = \{D_1, D_2, \ldots, D_n\}$, where each D_i can be formalized as $D_i = \{(k_{ij}, v_{ij})\}_{j=1}^{m}$, where k_{ij} and v_{ij} represent the key and value of the j-th key-value pair in the i-th dictionary, respectively. The text information of the patient's self report can be formalized as a sequence of tokens, denoted as $\mathcal{T} = \{t_1, t_2, \ldots, t_m\}$, where t_i represents the i-th token in the sequence. The goal of the report-assisted diagnosis generation task is to generate the final diagnosis of the patient, denoted as $\mathcal{Y} = \{y_1, y_2, \ldots, y_k\}$, where y_i represents the i-th token in the sequence.

3.2 Framework

In the pipeline of a text-serialization based method, the dictionaries are converted into a single natural-language string using a fixed template. However, this approach is sub-optimal for structured data like dictionary due to the structural heterogeneity between structured data and natural language.

To address this, We propose the DictLLM Framework. As shown in Figure 3, the DictLLM Framework consists of three main components: a hierarchical dict encoder, an optimal transport alignment layer, and a large language model. The hierarchical dict encoder and optimal transport alignment layer encode medical laboratory reports into several virtual tokens $\mathcal{T}_v$, the virtual tokens are then concat with the text tokens $\mathcal{T}$, and the combined tokens are fed into a large language model for generation.

3.3 Hierarchical Dict Encoder

Drawing inspiration from recent advancements such as SetTransformer Lee et al. (2019), TURL Deng et al. (2020), and Tapas Herzig et al. (2020), we harness the BERT's self-attention architecture Devlin et al. (2018) to model the intricate interactions within dictionaries. To effectively adapt to the unique data attributes of medical laboratory reports, the dict encoder incorporates dict tokenizer, relative position encoding and hierarchical attention biases. In the following sections, we will describe the them in detail.

3.3.1 Dict Tokenizer: tokenize numerical values in lab report

Dict tokenizer turns dictionaries into a series of token ids. To align with the behavior of medical practitioners in actual medical practice, we propose converting detailed numerical values in the laboratory reports into special medical labels. For a numerical attribute v_{ij}, the dict tokenizer maps it to a single token v'_{ij} (e.g., [NORMAL], [POSITIVE], [NEGATIVE]). We have defined a total of 13 such special medical labels, with a detailed list provided in the appendix. To be more specific, given a set of

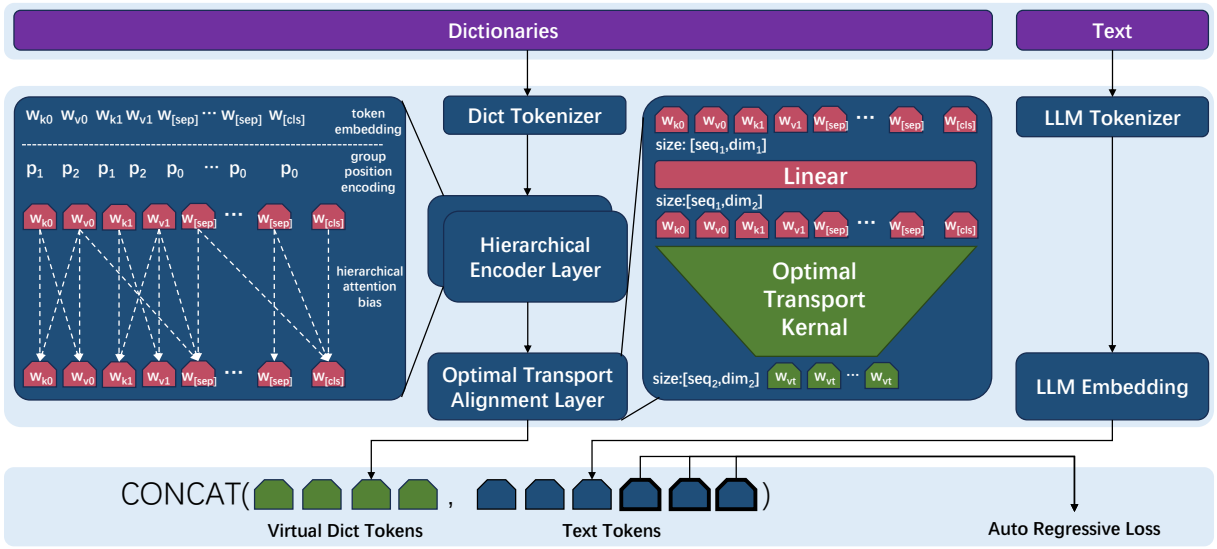


Figure 3: DictLLM Framework for report-assisted diagnosis generation. The medical lab report is first tokenized and encoded by the dict encoder. The embedding generated by the dict encoder are then aligned with the text embedding generated by the large language using the optimal transport alignment layer. The aligned embedding are then fed into the large language model to generate the final diagnosis.

dictionaries $\mathcal{D} = \{D_1, D_2, \ldots, D_k\}$ that contains k dictionary, where each $D_i = \{(k_{ij}, v_{ij})\}_{j=1}^{m}$ is a set that contains m key-value pairs. The dict tokenizer function f_t maps the whole $\mathcal{D}$ into a series of token ids t, denoted as:

$$f_t(\mathcal{D}) \rightarrow t = \{t_1, t_2, \ldots, t_n\}$$

3.3.2 Group Positional Encoding: maintain permutation invariance

After tokenization, the discrete token ids will be embedded into continuous vectors, which will be fed into the hierarchical encoder layer. We follow the standard practice of using a token embedding W and add a positional encoding P to the token embedding.

To model the permutation invariance of key-value pairs in laboratory reports, we have established a group positional encoding $P_{group} = \{p_{\text{pos}_1}, p_{\text{pos}_2}, \ldots, p_{\text{pos}_n}\}$. This encoding ensures that perturbation in the relative positions of elements within a dictionary do not impact the embedding generated by the dict encoder. Given the distinct characteristics of medical laboratory reports as dictionary-structured data, we propose the following assumption:

Assumption: *For a laboratory report D containing m key-value (k, v) pairs, changing the relative positions of these (k, v) pairs within D does not affect the final diagnosis.*

We implement P_{group} by resetting the index of positional ids at the beginning of each key-value pair, where pos_i represents the positional id for the ith token. Let W_{emb} be the embedding matrix of the dict encoder, The initial dict embedding h_0 is denoted as:

$$h_0 = W_{emb}(t) + P_{group}$$

3.3.3 Hierarchical Attention Bias: model structural inductive bias

Medical laboratory reports distinguish themselves from natural language in that, the correlation among items within a single report is significantly stronger than the correlation among items across different reports. (e.g. Test items on the same urine report are more likely to collectively indicate kidney-related diseases) We propose incorporating hierarchical attention bias to model the structural inductive bias of medical laboratory reports.

Specifically, tokens within the same dictionary are visible to each other, while tokens from different dictionaries are not. The special token [sep] is used to separate different dictionaries, and the special token [cls] is used to represent the whole dictionary. These special tokens are visible to each other, and they are visible to all tokens in their own dictionary. As illustrated in the Figure 3, tokens connected by dashed lines are visible to each other, while others are not.

The initial embedding will then pass through multiple hierarchical encoder layers(HierEnc) to obtain the final embedding. The hierarchical attention bias is implemented as a attention mask M,

which is a $n \times n$ matrix, where n is the sequence length. A hierarchical encoder layer consists of a Hierarchical Self-Attention (HierAttn) layer and a MLP layer, denoted as:

$$\mathit{HierEnc}(h_l) = h_l + \mathit{HierAttn}(h_l) + \mathit{MLP}(h_l)$$

$$\mathit{HierAttn}(h_l) = \mathit{softmax}(\frac{QK^T + M}{\sqrt{d_K}})V$$

$$M_{ij} = \begin{cases} 1 & t_i, t_j \in D \\ 1 & t_i, t_j \in \{[sep], [cls]\} \\ 0 & otherwise \end{cases}$$

After passing through the hierarchical encoder layers, the final dict embedding h_L is obtained.

3.4 Optimal Transport Alignment Layer

Algorithm 1 Optimal Transport Alignment Layer

Input source embedding $h_s \in \mathbb{R}^{m \times a}$
Output target embedding $h_t \in \mathbb{R}^{n \times b}$

initialize (trainable) reference points $z \in \mathbb{R}^{n \times b}$
initialize positive definite kernel Φ.
$h_r \in \mathbb{R}^{m \times b} \longleftarrow \Phi(h_s)$
$TP \in \mathbb{R}^{n \times m} \longleftarrow \mathit{sinkhorn}(h_r, z)$
$h_t \in \mathbb{R}^{n \times b} \longleftarrow TP \times h_r$

To deal with the heterogeneous information density between medical laboratory reports and natural language, we propose an optimal transport alignment layer to align the embedding generated by the dict encoder with those generated by the LLM, producing a list of fixed-length virtual tokens.

Natural language organized information in a sequential, dense and coherent manner, while information in medical laboratory reports are sparse and discrete. A naive approach such as using a linear layer may not be the optimal solution. Optimal transport is a mathematical framework that provides a principled way to align two sets of points in a high-dimensional space, which is widely used to alignment problems. Grave et al. (2018)

We utilize a recently proposed technique called optimal transport kernel Mialon et al. (2021) (OTK). OTK first utilize a positive definite kernel (i.e. in our implementation, a linear function) to embed the source set into a reproducing kernel Hilbert space (RKHS), then sinkhorn algorithm, which is a differentiable approximation of the optimal transport plan, is used to compute the optimal transport plan between the source set and a trainable reference set, which introduce non-linear transformation on source features. The detailed process of is described in algorithm 1.

Let the embedding output by the dict encoder be denoted as $h_L \in \mathbb{R}^{m \times a}$, where a is the number of tokens in the dict embedding, and b is the dimension of the token embedding. Our goal is to map it to a fixed-length virtual token $\mathcal{T}_v = \{t_{v1}, t_{v2}, \ldots, t_{vn}\} \in \mathbb{R}^{n \times b}$, where n is the number of virtual tokens, and b is the dimension of the large language model's token embedding.

4 Experiments Setup

4.1 Data Description

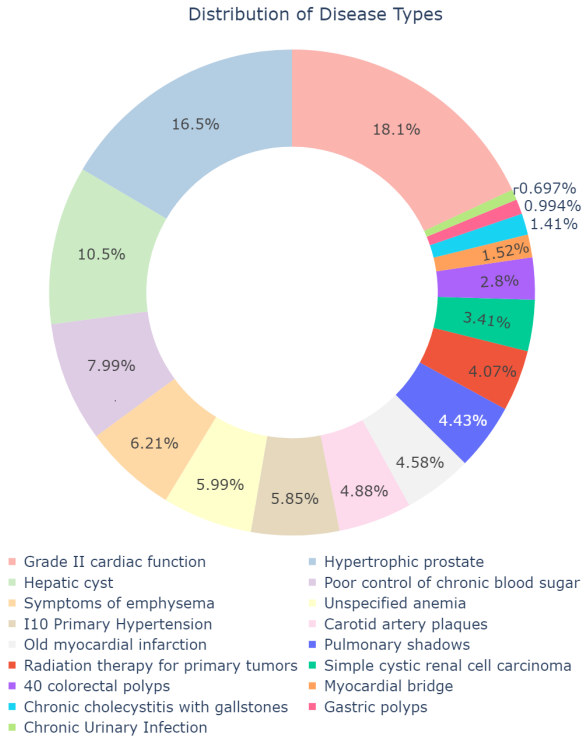


Figure 4: Distribution of different types of disease in the dataset.

The dataset we use in our experiment is a large-scale chinese real-world medical lab report dataset. We collect the dataset from a real-world hospital, which contains a large number of medical lab reports and the corresponding final diagnosis. The dataset contains a total of 11, 290 medical lab reports, and each report is associated with several final diagnosis. The original dataset is highly imbalanced in terms of the number of the disease types. We only keep the disease types that appears more than 0.1% of the time in the dataset. The dataset contains a wide range of disease types, as shown in Figure 4. The statistics of the dataset are

shown in Table 1.

num of cases	mean num of text token/case	mean num of lab report item/case
11,290	450.82	16.23

Table 1: Statistical information of datasets.

4.2 Baseline Methods

Text-Serialization For text-serialization method, we use a fixed template to serialize the medical lab report into a sequence of tokens. In our experiment, we separate each item in dict with comma, and use a special token to separate each dict. Then the model will be trained with the standard supervised fine-tuning paradigm.

GPT-4 We also evaluate the performance of GPT-4 on this task in zero-shot and few-shot settings. The template we use is the same as the one we use in baseline.

4.3 Implementation Details

For our model implementation, we primarily rely on the PyTorch and Transformers libraries. In terms of the Text-Serialization method, we convert medical lab reports into plain text at the dataset level and then train the model using the standard supervised fine-tuning paradigm. For our proposed DictLLM framework, we train the dict encoder and base large language models jointly. We choose internlm-7b-base and baichuan2-7b-base as our base models due to their superior performance in Chinese. We utilize the AdamW optimizer with a learning rate of $2e-5$ and a total batch size of 128. We apply a warmup ratio of 0.01, and the training process spans 6 epochs. Notably, we did not conduct any hyperparameter search in our experiment. Regarding the dataset, we split it into training and testing sets, using 90% of the data for training and the remaining 10% for testing.

4.4 Evaluation Metrics

We use the following metrics to evaluate the performance of the methods we proposed in this paper:

Rouge-L Rouge-L is a metric that measures the similarity between two sequences. It is widely used in the text generation task.

Knowledge F1 We also use the knowledge F1 score to evaluate the performance of the methods we proposed in this paper. Knowledge F1 score is a metric that measures the quality of the generated sequence in terms of the knowledge it contains. In our experiment, we implement the knowledge F1 score as the harmonic mean of precision and recall of the correct diagnosis in the generated sequence.

5 Results

5.1 Main Results

Table 2 shows the main results of our experiment. As we can see, the proposed DictLLM framework outperforms the baseline methods in terms of both Rouge-L and Knowledge F1 score in all settings. The performance of our method is consistent across different backbone models. The results demonstrate that our proposed DictLLM framework is effective in modeling the heterogeneous structure of medical lab reports and generating the final diagnosis.

Notably, GPT-4 achieve poor performance in both zero-shot and few-shot settings, and there is a large gap between the performance of GPT-4 and the finetuned large language models. The main reason is report-assisted diagnosis generation task is that the task requires the model to have a good understanding of the specialized medical terminology, which is rare in the training data of GPT-4.

The gap between the performance of the text-serialization method and our proposed DictLLM framework in baichuan-7b is smaller than that in internlm-7b, which is mainly due to the better backbone model performance of baichuan-7b.

5.2 Scalability to Input Length

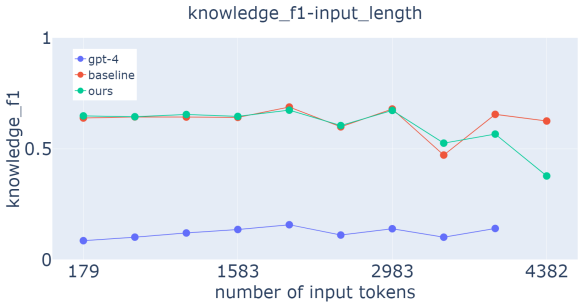


Figure 5: The knowledge F1 score of different methods with respect to the number of input tokens. Other results are detailed in the appendix A.

We also evaluate the scalability of several method to the input length on the backbone of internlm-7b. In real-world medical lab reports, the number of items in the report can be very large, and the length of the report may exceed the limitation of the max token length of large language models.

Method		Rouge-L			Knowledge		
		P	R	F1	P	R	F1
GPT-4	zero-shot	5.65	32.87	8.64	8.82	28.85	12.70
	few-shot	5.80	33.45	8.99	9.54	32.11	13.84
INTERNLM-7B	zero-shot	3.59	4.25	3.83	4.38	3.95	4.05
	few-shot	5.89	5.35	5.29	5.99	5.32	5.53
	finetune	51.89	45.19	46.69	50.90	46.03	47.43
	DictLLM	68.67	63.15	**64.24**	68.68	64.09	**65.24**
BAICHUAN-7B	zero-shot	6.14	8.24	6.93	7.71	7.39	7.40
	few-shot	8.35	12.67	9.83	10.19	9.27	9.58
	finetune	67.15	63.18	63.13	67.32	64.42	64.51
	DictLLM	67.26	63.39	**63.28**	67.50	64.65	**64.61**
HUATUOGPT-7B	finetune	66.53	61.83	62.03	64.16	59.64	60.30
	DictLLM	67.84	63.19	**63.48**	65.97	61.60	**62.31**

Table 2: **Main Results** We compare the performance of our DictLLM framework with several baseline methods on the medical lab report-assisted diagnosis generation task. We report the Rouge-L and Knowledge F1 scores. The best results are in bold. The detail of the evaluation metrics can be found in section4.4.

Large input length would also lead to large training time and memory requirement, which could be a bottleneck for the model to be deployed in real-world applications.

As shown in Figure 6, the performance of the text-serialization method decreases significantly as the input length increases due to the large input token size. In contrast, our proposed DictLLM framework effective compress the input token number and achieve consistent performance across different input lengths, demonstrating a better scalability of our method to the input length.

5.3 Robustness to Input Perturbation

Besides scalability, Robustness to input perturbation is also an important property for the model to be deployed in real-world applications. Input perturbation refers to the random permutation of the items in the medical lab report. In the ideal situation, the model should generate the same diagnosis for the same medical lab report, regardless of the order of the items in the report. To evaluate the robustness of the model to input perturbation, we conduct an experiment to compare the performance of different methods before and after perturbation. We report the performance and the relative change of the generated text before and after perturbation in Table 3. The metric RC (i.e. Relative Change) is calculated as the $1 - RougeL_{f1}$ score between the text generated before and after the perturbation.

As is shown in Table 3, the performance of the text-serialization method decreases after perturbation, while the performance of our proposed DictLLM framework is the most stable across different backbone models. We also observe that the relative change of the generated text before and after perturbation is the smallest for our proposed DictLLM framework, demonstrating the robustness of our method to input perturbation.

5.4 Ablation study

5.4.1 Ablation over the main components

We conduct ablation study to demonstrate the effectiveness of the model components in our proposed DictLLM framework. For the ablation of group positional encoding, we replace it with the standard sequential positional encoding. For the ablation of optimal transport alignment layer, we replace it with the a simple linear layer. For the ablation of hierarchical attention bias, we just simply remove it from the model. Table 4 shows the ablation study results.

Overall, the results show that each component in our proposed DictLLM framework contributes to the performance of the model. Among all the components, deleting the hierarchical attention bias leads to the largest performance drop, demonstrating the importance of the hierarchical attention bias in capturing the structural inductive bias of medical lab reports.

Method	Before Perturbation						After Perturbation						
	Rouge-L			Knowledge			Rouge-L			Knowledge			RC↓
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	
GPT-4	5.65	32.87	8.64	8.82	28.85	12.70	5.77	32.21	8.64	8.79	27.69	12.43	36.30
TEXT-SERIALIZATION	53.43	46.31	47.76	51.89	47.43	48.60	52.69	45.77	47.24	52.22	47.63	48.82	11.31
DICTLLM	68.53	63.52	64.22	68.42	64.40	65.11	68.61	63.64	64.33	68.51	64.49	65.20	1.71

Table 3: **Perturbation Results** We compare the performance of our DictLLM framework with baseline methods on the medical lab report-assisted diagnosis generation task before and after perturbation. RC denotes the relative change of the generated text before and after perturbation.

Method	Rouge-L			Knowledge		
	P	R	F1	P	R	F1
DICTLLM	68.67	63.15	64.24	68.68	64.09	65.24
- position encoding	67.25	60.96	62.23	67.29	62.23	63.46
- attention bias	66.15	60.61	61.40	66.19	61.69	62.53
- alignment layer	68.53(0.44)	61.15(0.16)	62.91(0.35)	67.18(1.39)	60.70(1.24)	62.52(1.44)

Table 4: Ablation study of DictLLM framework.

5.4.2 Ablation over the virtual token number

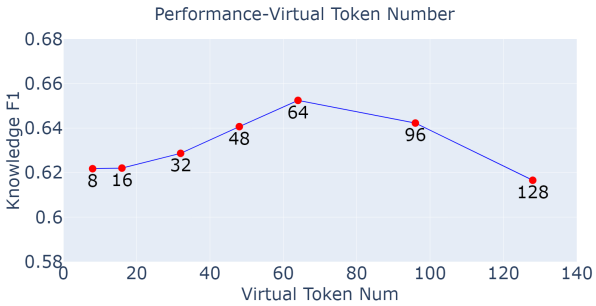


Figure 6: Ablation study of virtual token ber.

The number of the virtual token is a hyperparameter in our proposed DictLLM framework. We conduct an ablation study to evaluate the performance of the model with different virtual token number. As shown in Figure 6, the performance of the model increases as the virtual token number increases.

However, the increase of the virtual token number also leads to the slightly increase of the model size and the memory requirement. We choose 64 as the virtual token number in our experiment, which achieves a good trade-off between the performance and the memory requirement.

6 Conclusion

In this paper, We propose a novel framework called DictLLM, which is an efficient and effective framework for modeling the heterogeneous structure of structured data, to deal with the report-assisted diagnosis generation task. Our comprehensive empirical studies on real-world datasets reveal that a carefully designed encoder, which individually encodes structured data, significantly enhances model performance on downstream tasks, demonstrating advantages in scalability and robustness.

Limitation The DictLLM framework is specifically designed for processing dictionary-structured data and requires some effort to further extend it to more complex tabular data. Additionally, although DictLLM has reduced training and inference overhead compared to text-serialization methods, it still demands significant computational resources.

Acknowledgements

This work is supported by the National Key R&D Program of China (No. 2022ZD0160702), STCSM (No. 22511106101, No. 18DZ2270700, No. 21DZ1100100), 111 plan (No. BP0719010), STCSM (No. 21511101100), and State Key Laboratory of UHD Video and Audio Production and Presentation. This research received partial support from Shanghai Ninth People's Hospital, which provided the medical lab report data. We also extend our thanks to Dr. Ran Li for the valuable advice and expertise in medicine.

References

ChatExcel. https://chatexcel.com/.

2023. OpenTab: Advancing Large Language Models as Open-domain Table Reasoners. In *The Twelfth International Conference on Learning Representations*.

Pei Chen, Soumajyoti Sarkar, Leonard Lausen, Balasubramaniam Srinivasan, Sheng Zha, Ruihong Huang, and George Karypis. 2023. HYTREL: Hypergraph-enhanced Tabular Data Representation Learning.

Xiang Deng, Huan Sun, Alyssa Lees, You Wu, and Cong Yu. 2020. TURL: Table Understanding through Representation Learning.

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

Kounianhua Du, Weinan Zhang, Ruiwen Zhou, Yangkun Wang, Xilong Zhao, Jiarui Jin, Quan Gan, Zheng Zhang, and David Wipf. 2022. Learning Enhanced Representations for Tabular Data via Neighborhood Propagation.

Edouard Grave, Armand Joulin, and Quentin Berthet. 2018. Unsupervised Alignment of Embeddings with Wasserstein Procrustes.

Yucheng Han, Chi Zhang, Xin Chen, Xu Yang, Zhibin Wang, Gang Yu, Bin Fu, and Hanwang Zhang. 2023. ChartLlama: A Multimodal LLM for Chart Understanding and Generation.

Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. 2023. TabLLM: Few-shot Classification of Tabular Data with Large Language Models.

Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Eisenschlos. 2020. TaPas: Weakly Supervised Table Parsing via Pre-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4320–4333, Online. Association for Computational Linguistics.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam R. Kosiorek, Seungjin Choi, and Yee Whye Teh. 2019. Set Transformer: A Framework for Attention-based Permutation-Invariant Neural Networks.

Grégoire Mialon, Dexiong Chen, Alexandre d'Aspremont, and Julien Mairal. 2021. A Trainable Optimal Transport Embedding for Feature Aggregation and its Relationship to Attention.

Jiabin Tang, Yuhao Yang, Wei Wei, Lei Shi, Lixin Su, Suqi Cheng, Dawei Yin, and Chao Huang. 2023. GraphGPT: Graph Instruction Tuning for Large Language Models.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and Efficient Foundation Language Models.

Jingfeng Yang, Aditya Gupta, Shyam Upadhyay, Luheng He, Rahul Goel, and Shachi Paul. 2022. TableFormer: Robust Transformer Modeling for Table-Text Encoding.

Chao Ye, Guoshan Lu, Haobo Wang, Liyao Li, Sai Wu, Gang Chen, and Junbo Zhao. 2023. CT-BERT: Learning Better Tabular Representations Through Cross-Table Pre-training.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, Weng Lam Tam, Zixuan Ma, Yufei Xue, Jidong Zhai, Wenguang Chen, Peng Zhang, Yuxiao Dong, and Jie Tang. 2022. GLM-130B: An Open Bilingual Pre-trained Model.

Fengbin Zhu, Ziyang Liu, Fuli Feng, Chao Wang, Moxin Li, and Tat-Seng Chua. 2024. TAT-LLM: A Specialized Language Model for Discrete Reasoning over Tabular and Textual Data.

A Appendix

A.1 Case Study

Here are two cases created by DictLLM, showcasing specific instances where DictLLM excels and where it faces challenges:

A.2 Special Medical Labels

As is shown in the table 5 , We define a total of 13 special medical labels to convert detailed numerical values in the laboratory reports into special medical labels.

Labels
[NORMAL]
[ABNORMAL]
[HI NORMAL]
[LT NORMAL]
[POSITIVE]
[NEGATIVE]
[POSITIVE]
[POSITIVE+]
[POSITIVE++]
[POSITIVE-]
[POSITIVE–]
[SENSTIVE]
[RESISTANT]
[INTERMEDIATE]

Table 5: **Special Medical Labels.**

A.3 Prompt for Zero-shot and Few-shot Generation

Zero-shot prompt:

```
Please output the patient's discharge
diagnosis based on the given laboratory order
and patient information. Each disease should
be separated by a Chinese comma and then
output a period. Do not output anything else.
Example output: Low-risk mild hypertension,
elevated serum uric acid concentration, stage
5 chronic kidney disease. Laboratory test
report: {} Patient information: {}
```

Few-shot prompt:

```
Please output the patient's discharge
diagnosis based on the given laboratory order
and patient information. Each disease should
be separated by a Chinese comma and then
output a period. Finish. Do not output
anything else. Examples:  Laboratory test
report: {} Patient information: {}
```

Input: Gender: Male Age: 91 years old Main symptoms and signs on admission: Recurrent cough and sputum for more than 6 years, worsening for 3 days, clear mind, flat air, finger pulse oxygen saturation 95% (nasal cannula oxygen inhalation 4L/min), double The lung breath sounds were low, and obvious crackles could be heard in both lungs. The heart rate was 80 bpm, with a regular rhythm and no obvious murmur. The abdomen is soft, without tenderness or rebound tenderness. There is a 4*5cm round mass in the right groin, which is soft and non-tender. There was no edema in both lower limbs, the dorsalis pedis artery was palpable, the nasogastric tube and urinary catheter were in place and unobstructed.
DictLLM's output: Hyperplastic prostate, anemia unspecified, liver cyst
Ground Truth: Hyperplastic prostate, anemia unspecified, liver cyst

Input: Gender: Male Age: 69 years old Main symptoms and signs on admission: Chest pain, pending investigation for 1 week, refreshed and calm. The heart rhythm is regular and there is no murmur. The breath sounds in both lungs were clear, with less obvious dry and wet rales. The abdomen was flat and soft, with no tenderness or rebound tenderness in the entire abdomen, the liver and spleen were not reachable under the ribs, shifting dullness (-), and low bowel sounds.
DictLLM's output: 40 colorectal polyps, myocardial bridge
Ground Truth: Chronic cholecystitis with gallstones

Figure 7: Case study of the generated diagnoses, showcasing specific examples where DictLLM performs well and where it struggles.

A.4 Scalability to Input Length: Detailed Results

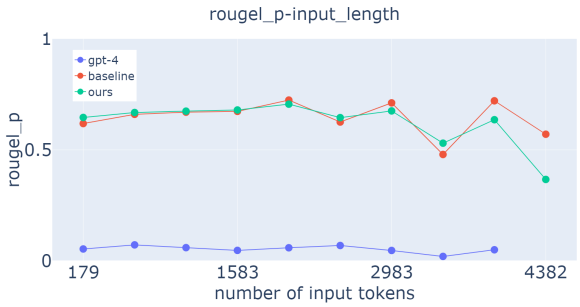


Figure 8: The Rouge-L precision score of different methods with respect to the number of input tokens.

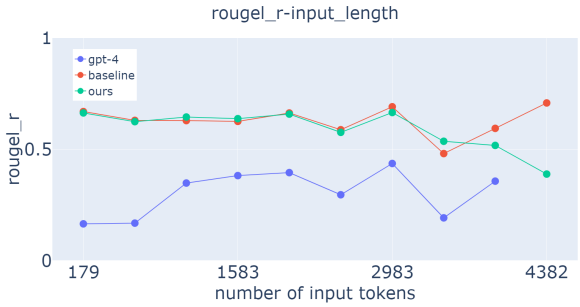


Figure 9: The Rouge-L recall score of different methods with respect to the number of input tokens.

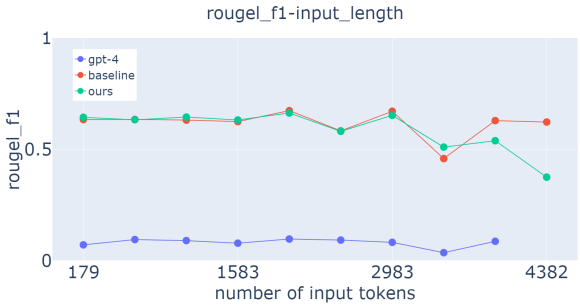


Figure 10: The Rouge-L F1 score of different methods with respect to the number of input tokens.

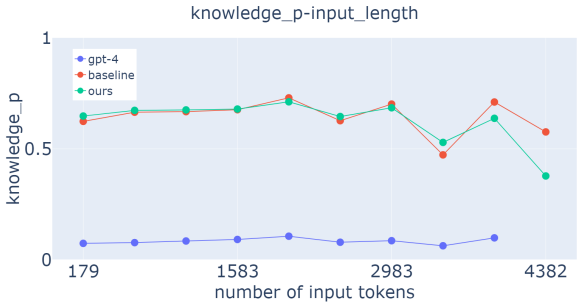


Figure 11: The knowledge precision score of different methods with respect to the number of input tokens.

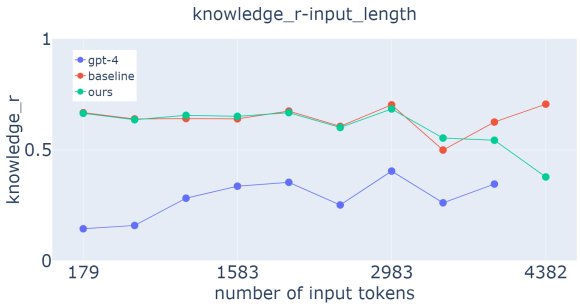


Figure 12: The knowledge recall score of different methods with respect to the number of input tokens.