GUD-IR: Generative Retrieval for Semiparametric Models

Aman Madaan [*], **Niket Tandon** [*†], **Peter Clark**[†], **Yiming Yang**
Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA, USA
[†] Allen Institute for Artificial Intelligence, Seattle, WA, USA
{amadaan,yiming}@cs.cmu.edu
{nikett,peterc}@allenai.org

Abstract

When a large language model produces undesirable output after deployment, recent semiparametric methods have relied on a growing database of how a similar error was resolved in the past. This paper highlights current issues in retrieving similar errors from the database, and addresses these issues with a novel generation-inspired retrieval technique, called GUD-IR (Generated UnDerstanding for explainable IR). On two existing datasets: SocialChem101 (ethical reasoning) and OBQA (science QA), we found that off-the-shelf modern retrieval methods fail to distinguish lexically dissimilar but semantically similar erroneous situations. On both the datasets, GUD-IR provides nearly 10% absolute precision gains in retrieval quality over strong baselines. Our approach is a step towards deploying semiparametric methods as a low-cost utility enhancement for very large pre-trained LMs.

1 Introduction

Large LMs such as GPT-3 and T5 are powerful, but can commit mistakes that are obvious to humans. Recent semiparametric methods such as (Madaan et al., 2022) record a growing memory or database of cases where the model failed and was provided with an understanding of the input in the form of feedback (e.g., underlying principle or knowledge). Given a new query $\mathbf{x}$, they look up the database of understanding $\mathbf{u}$ or knowledge from similar, prior queries that apply to the new query. We found that this lookup is non-trivial and error-prone because a data point in the database can be lexically dissimilar but semantically similar and vice-versa. Consider situation $\mathbf{x}_1$: *tom hated skating because he had no sense of balance* and situation $\mathbf{x}_2$: *jordyn was trying to improve her cooking skills*. These situations are lexically dissimilar, yet a common *rule-of-thumb* binds them. Specifically, they can both be explained by $\mathbf{u}$: *practicing more to improve your skills*. We find that off-the-shelf IR methods fail to recognize this similarity of intent, leading to irrelevant database entries applied to the input, degrading model performance.

To address these retrieval issues, we propose **GUD-IR** (Generated UnDerstanding for explainable IR). The key intuition for our approach relies on substituting $f_\theta : \mathbf{x} \rightarrow \mathrm{R}^d$ (latent space projection) with $f_\theta : \mathbf{x} \rightarrow \mathbf{u}$ (generated understanding of $\mathbf{x}$). Concretely, instead of learning a function that maps a question to a d dimensional vector, we train a generative model that directly maps an input to a rough understanding. The generated rough understanding is then used as a key to retrieve a relevant understanding from the database using any off-the-shelf retrieval method. This two-step *generate-then-retrieve* procedure has benefits: (i) it alleviates sparsity issues that we found latent space projection methods were unable to deal with[1] (ii) the overall retrieval becomes explainable and debuggable.

Our approach is inspired and supported by the recent success of generate and retrieve (Mao et al., 2021) methods. However, despite the similarity, the methods have different goals: Mao et al. (2021) leverage generative models for query expansion, whereas our goal is explainable input understanding. Moreover, their implementation is geared towards open-domain QA, while ours is towards explainable input understanding. Thus, it is non-trivial to adapt similar ideas to our tasks effectively. To this end, our contributions are:

- We highlight the IR issues in semiparametric methods.

[*]Equal Contribution

[1]e.g., there are only eight popular emotions but can lead to a large number of diverse situations. Hence, many inputs can map to the same principle $\mathbf{u}$. This mapping becomes increasingly difficult for a model as the specificity of $\mathbf{u}$ increases, because of sparsity issues. This is exacerbated when the input situations are diverse and previously unseen.

- We present GUD-IR, a simple and effective retriever to enhance the utility of semip. models.

2 Generative IR (GUD-IR)

One of the key strengths of existing methods such as MEM-PROMPT (Madaan et al., 2022) is its ability to leverage the understanding provided on earlier inputs $\mathbf{x}$ to improve a future input. This is achieved by retrieving an understanding from memory $\mathcal{M}$ using $\mathbf{x}$ as the key. An underlying assumption of this process is that similar inputs will admit similar understanding, allowing us to use the understanding provided for one situation on another. For two input situations $\mathbf{x_i}$ and $\mathbf{x_j}$ with respective understanding $\mathbf{u}_i$ and $\mathbf{u}_j$, this assumption can be stated as:

$$x_i \sim x_j \implies \mathbf{u}_i \sim \mathbf{u}_j$$

The Social chemistry (ethical reasoning dataset) (Forbes et al., 2020) fails to meet this assumption because lexically dissimilar situations might have the same understanding, thus posing a unique challenge for our method. As a concrete example, consider an input situation $\mathbf{x_i}$: *tom hated skating because he had no sense of balance* – with an understanding $\mathbf{u}_i$: *practicing more when you want to improve your skills*. Suppose that our system has already seen $\mathbf{x_i}$ and has received an understanding $\mathbf{u}_i$ (i.e., there is an entry in $\mathcal{M}$: $\mathbf{x_i} \rightarrow \mathbf{u}_i$). Next, suppose a user enters a new situation $\mathbf{x_j}$: *jordyn was trying to improve her soccer skills*. As usual, MEM-PROMPT will try to retrieve understanding for a *similar* situation. However, such retrieval is going to be challenging, because $\mathbf{x_i}$ (*tom hated skating because he had no sense of balance*) has little to no overlap with $\mathbf{x_j}$ (*jordyn was trying to improve her soccer skills*). Consequently, MEM-PROMPT may fail to retrieve the relevant understanding $\mathbf{u}_i$ or worse, may retrieve a misleading understanding.

The fact that similarity of two inputs $(\mathbf{x}_i, \mathbf{x}_j)$ does not imply similarity of the understanding $(\mathbf{u}_i, \mathbf{u}_j)$ makes vanilla retrieval non-viable for our setting. We deal with this challenging situation with two different solutions of increasing complexity.

2.1 Initial approach: Fine-tuning with u similarity

Since the surface level similarity of input situations is not enough to capture similarity of respective understanding, we attempt to learn a function f_θ that will map similar inputs $\mathbf{x}_i$ and $\mathbf{x}_j$ to similar representations if the corresponding understanding $\mathbf{u}_i$ and $\mathbf{u}_j$ are close to each other, and vice-versa. A natural choice is training an embedding function $f : \mathbf{x} \rightarrow \mathrm{R}^d$ supervised by $\texttt{cos}(\mathbf{u}_i, \mathbf{u}_j)$ where $\texttt{cos}$ is the cosine similarity ($\texttt{cos}(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a}^T\mathbf{b}}{|\mathbf{a}||\mathbf{b}|}$). Thus, the objective function is:

$$\mathcal{L}_\theta = (\texttt{cos}(f_\theta(\mathbf{x}_i), f_\theta(\mathbf{x}_j)) - \texttt{cos}(\mathbf{u}_i, \mathbf{u}_j))^2$$

Intuitively, this objective function will encourage the similarity between the inputs $(\texttt{cos}(f_\theta(\mathbf{x}_i), f_\theta(\mathbf{x}_j)))$ to be high when the corresponding understanding are similar, and vice-versa.

Understanding retrieval proceeds as follows: an input $\mathbf{x_i}$ is embedded using f_θ, and $f_\theta(s_i)$ is then used to retrieve an understanding from the memory, with the hope that representations $f_\theta(s_i)$ and $f_\theta(s_j)$ will be similar after the training.

While in principle this objection function should be enough to learn informative representations, we found the training to be unstable. We attribute this to the fact that two extremely dissimilar situations can have identical understanding. Thus, it might be unrealistic to train similarity functions that can capture all possible cases where the same understanding applies to two situations. After observing the unstable training using this method, we experiment with a generative model for the task, described next.

2.2 Proposed approach GUD-IR: Retrieve understanding similar to a generated one

Our goal is to retrieve an understanding $\mathbf{u}$ that applies to the given input $\mathbf{x_i}$. To achieve this, we match $\mathbf{x_i}$ to the closest entry $\mathbf{x}_{mem}$ in $\mathcal{M}$. However, $\mathbf{x}$ and $\mathbf{x}_{mem}$ can be in very different spaces, even different language dialects, thus making lexical matching ineffective and the initial approach in Section §2.1 must deal with sparsity issues. For example, when translating $\mathbf{x}$ in Indian English to $\mathbf{x}_{mem}$ in American English, the previous methods fail because the query space is very different from the document space. One could imagine that it would help to look up a large Indian-American English phrase-book. This is the motivation behind our approach.

The key intuition for our approach relies on substituting $f_\theta : \mathbf{x} \rightarrow \mathrm{R}^d$ with $f_\theta : \mathbf{x} \rightarrow \mathbf{u}$. That is, instead of learning a function that maps a question to a d dimensional vector, we train a generative model

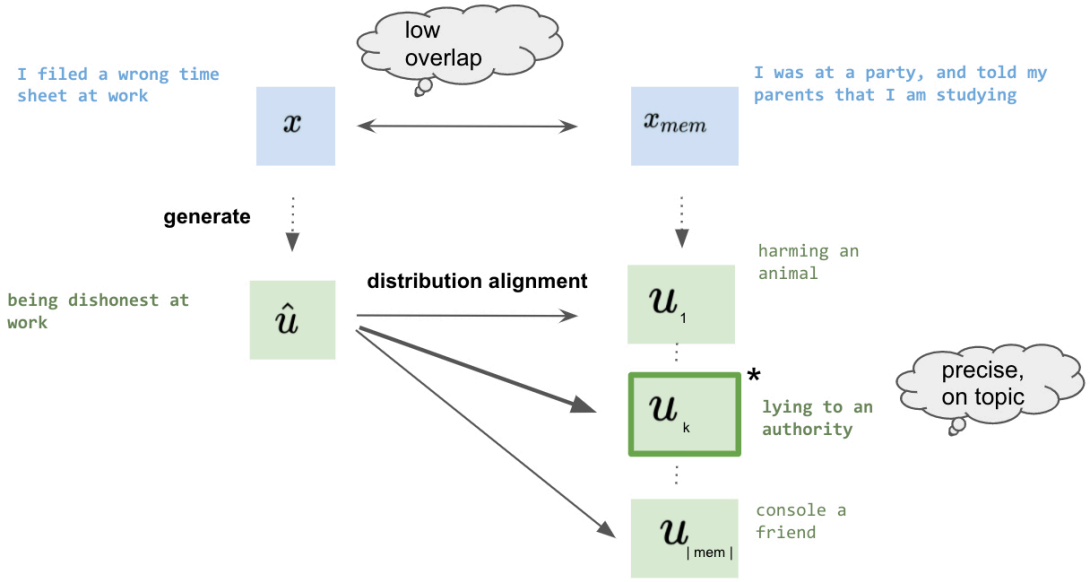

Figure 1: **Overview of GUD-IR**. To retrieve a relevant understanding that applies to $\mathbf{x}$, GUD-IR first generates an understanding $\hat{\mathbf{u}}$ using a generative model. This is then aligned with a database/ memory of understandings $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_{|mem|}$. The best matching understanding $\hat{\mathbf{u}}^*$ is then used for $\mathbf{x}$. Thus, GUD-IR decomposes the retrieval problem $\mathbf{x} \rightarrow \mathbf{u}$ into two sub-problems: (i) generate a rough understanding ($\mathbf{x} \rightarrow \hat{\mathbf{u}}$) and (ii) search a database/ memory for the closest understanding $\hat{\mathbf{u}}^* = \arg\min_{j \in [1,|mem|]} |\hat{\mathbf{u}} - \mathbf{u}_j|$.

that directly maps an input situation a rough understanding. The generated understanding is then used as a key to retrieve a relevant understanding from the training set.

Specifically, we train a sequence-to-sequence model, (e.g., BART or T5), that maps each input $\mathbf{x}$ to a corresponding output $\mathbf{u}$. The understanding is now retrieved in a two step process:

1. The generative model f_θ is used to generate a noisy understanding for $\mathbf{x_i}$, $\hat{\mathbf{u}}$.
2. $\hat{\mathbf{u}}$ is used as a key to *search* over the set of already present understandings, to retrieve the nearest one.

Intuitively, our generative IR model transforms the lookup problem to a mapping and search problem: we first transform the input to the space of understandings, then search over the set of already present understanding. The search normalizes the generated $\mathbf{u}$ into a real $\mathbf{u}^*$ by generating $\mathbf{u}$ that is expected to be similar to the $\mathbf{u}$ that the model is attempting to find. Figure 1 presents an overview of our *generation then reshape* approach (GUD-IR).

3 Experiments

Tasks We consider two tasks of varying complexity of $\mathbf{x} \rightarrow \mathbf{u}$.

- Task 1: SOCIAL-CHEM-101: A subset of social norms dataset[2] (Forbes et al., 2020) provided by Madaan et al. (2022). There are 32,000 data points (30,000 train, 1,000 dev, 1,000 test) of the type $\mathbf{x} \rightarrow \mathbf{u}$[3]. This is the more challenging dataset because semantically dissimilar $\mathbf{x}$ can have the same underlying $\mathbf{u}$.
- Task 2: OBQA - The Open Book QA (OBQA) dataset (Mihaylov et al., 2018). There are ∼6,000 data points (5,000 train, 500 dev, 500 test) of the type $\mathbf{x} \rightarrow \mathbf{u}$.

Table 1 presents examples of the task. From a semiparametric model perspective, the main interest is to use past understanding on future instances. The table picks a sample case where at test time, the model can make use of a train-time understanding $\mathbf{u}$ for a similar point $\mathbf{x}$.

Baselines We compare GUD-IR with two different baselines, $\mathbf{x} \rightarrow \mathbf{x}$, $\mathbf{x} \rightarrow \mathrm{R}^d$ to compare with our approach $\mathbf{x} \rightarrow \mathbf{u}$:

- **BM25** is a standard and strong retrieval baseline based on word tokens in a probabilistic retrieval framework. $\mathbf{x}_1$ and $\mathbf{x}_2$ are matched lexically.
- **SENT-BERT** is a popular sentence similarity base-

[2] https://github.com/mbforbes/social-chemistry-101

[3] additionally SOCIAL-CHEM-101 has answer labels about social judgment (it is bad, it is ok, it is good) – we do not present any results on the answer label in this paper

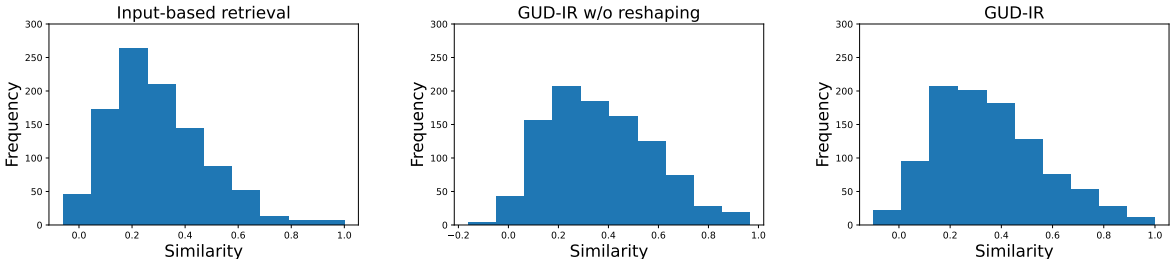

Figure 2: Distribution of similarity scores between expected $\mathbf{u}^*$ and $\hat{u}$ for retrieval (left) and GUD-IR (right). The similarity scores are higher using GUD-IR.

Dataset	$\mathbf{x} \rightarrow \mathbf{u}$
SOCIAL-CHEM-101 (train time)	$\mathbf{x}$: Turning my blender on at 3AM $\mathbf{u}$: making noise at night
SOCIAL-CHEM-101 (*test time*)	$\mathbf{x}$: Play drums late at night $\mathbf{u}$: ?
OBQA (train time)	$\mathbf{x}$: Wind causes stones to weather down to pebbles $\mathbf{u}$: wind causes erosion
OBQA (*test time*)	$\mathbf{x}$: Wind depletes dunes at the beach $\mathbf{u}$: ?

Table 1: Examples of two tasks. On an unseen/ test point $\mathbf{x}$, GUD-IR can rely on a database/ memory of train time/ past $\mathbf{u}$.

	Exact-match
BM-25	46.2
SENT-BERT	41.2
GUD-IR w/o reshaping	53.4
GUD-IR	54.5

Table 2: Results on OBQA using GUD-IR and OBQA.

line where the query is encoded using Sentence transformers (Reimers and Gurevych, 2019). To match $\mathbf{x}_1$ and $\mathbf{x}_2$ it uses cosine dist. on the vector representations with a threshold of 0.9

Metrics

- For SOCIAL-CHEM, we use the similarity score of the retrieved understanding and ground truth clarification for evaluation. Specifically, we compute the semantic similarity between retrieved understanding $\hat{\mathbf{u}}$ and the ground truth $\mathbf{u}^*$ using sentence transformers, $\texttt{sim}(\mathbf{u}^*, \hat{\mathbf{u}})$. Exact match scores are very low in this setting because the task is harder.
- For OBQA, we use the exact match metric.

Results On SOCIAL-CHEM-101 we plot the distribution of $\texttt{sim}(\hat{u}, \hat{u}^*)$ over the test set. Figure 2 shows this distribution using GUD-IR and using surface-level similarities. The probability mass shifts towards a higher similarity range for GUD-IR.

On OBQA dataset, there is a substantial boost over BM25 baseline, and it is also clear that reshaping/ distribution alignment does help. Overall we get 8.3% exact match boost over BM25 (see Table 2).

4 Conclusion

This paper presented the IR issues in semiparametric methods and an effective approach GUD-IR to generate and then retrieve. On two diverse tasks, we show improvements in IR performance using GUD-IR. (Madaan et al., 2022) shows that end task performance would improve with IR, and thus our results on better IR imply a better end task performance. An important future work is to generate $\hat{\mathbf{u}}$ using a constrained vocabulary present in the database/memory. Overall, this paper is a step towards deploying semiparametric models, and effectively improve the model performance over time.

References

Maxwell Forbes, Jena D. Hwang, Vered Shwartz, Maarten Sap, and Yejin Choi. 2020. Social chemistry 101: Learning to reason about social and moral norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 653–670, Online. Association for Computational Linguistics.

Aman Madaan, Niket Tandon, Peter Clark, and Yiming Yang. 2022. Memory-assisted prompt editing to improve gpt-3 after deployment. *ArXiv*, abs/2201.06009.

Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. 2021. Generation-augmented retrieval for open-domain question answering. *ArXiv*, abs/2009.08553.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct elec-

tricity? a new dataset for open book question answering. In *EMNLP*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *EMNLP*.