Variational Learning for Unsupervised Knowledge Grounded Dialogs

Mayank Mishra* , **Dhiraj Madan** , **Gaurav Pandey** and **Danish Contractor**
IBM Research AI
mayank.mishra1@ibm.com, {dmadan07, gpandey1}@in.ibm.com, danish.contractor@ibm.com

Abstract

Recent methods for knowledge grounded dialogs generate responses by incorporating information from an external textual document. These methods do not require the exact document to be known during training and rely on the use of a retrieval system to fetch relevant documents from a large index. The documents used to generate the responses are modeled as latent variables whose prior probabilities need to be estimated. Models such as RAG and REALM, marginalize the document probabilities over the documents retrieved from the index to define the log likelihood loss function which is optimized end-to-end.

In this paper, we develop a variational approach to the above technique wherein, we instead maximize the Evidence Lower bound (ELBO). Using a collection of three publicly available open-conversation datasets, we demonstrate how the posterior distribution, that has information from the ground-truth response, allows for a better approximation of the objective function during training. To overcome the challenges associated with sampling over a large knowledge collection, we develop an efficient approach to approximate the ELBO. To the best of our knowledge we are the first to apply variational training for open-scale unsupervised knowledge grounded dialog systems.

1 Introduction

In this paper we focus our attention on the task of generating responses, grounded on information present in a large collection of external textual documents [Lewis *et al.*, 2020]. In real-world scenarios, the exact document that one must access for generating the response is often unknown and one only has access to conversation logs and a document collection. Hence during training, given a dialog context, the primary challenge is first figuring out the correct document needed to generate the response, and then using that document for generating the actual response.

A straightforward baseline approach would be to use an out-of-the-box retriever (for instance, a tf-idf based retriever such as BM25 [Robertson *et al.*, 1994] or a neural retriever such as DPR [Karpukhin *et al.*, 2020]) for first retrieving the document and then using a retrieved document for generating the response. While this is fairly easy to implement, it cannot be trained in an end-to-end manner and thus, the retriever never improves as the model learns to generate responses.

To overcome this limitation, methods such as RAG [Lewis *et al.*, 2020], model documents as latent variables and learn a distribution over these variables (Figure 1a). This distribution is referred to as the *document prior*. Specifically, the document-prior distribution is defined by querying a knowledge index [Johnson *et al.*, 2017; Karpukhin *et al.*, 2020] using the dialog context (history), and then converting the retrieval scores of the top-k[1] documents into a probability distribution. The response-likelihood can be defined using any neural language generator such as GPT2 [Radford *et al.*, 2019]. It then performs a marginalisation of the latent variable over the retrieved documents to compute the approximate probability of the response, given the context. The negative log likelihood under this approximation forms the loss function to train.

However, one of the weaknesses of this approach is that, using the document-prior to query the index during training, ignores crucial information present in the ground-truth response which could have aided document retrieval. As a result, the response-likelihood network parameters may receive a weaker signal during training, which, in-turn, can cause models to try reducing their dependence on external knowledge by 'memorizing', especially if the correct document is rarely fetched by the retriever.

Variational Retrieval-Augmented Generation (VRAG): In this paper, we propose an approach that overcomes this limitation. We incorporate the ground truth response with the dialog context for retrieving the documents during training in a secondary retriever. This increases the chances of retrieving the correct document during training. The distribution over the documents defined by this retriever is referred to as the *document posterior*. The document posterior guides the training of the document prior while the documents sampled by the posterior are fed to the decoder for generating the re-

*Contact Author

[1] typically k = 5-10.

sponse. Such a formalism emerges naturally in the variational setting, wherein the evidence lower bound (ELBO) is optimized instead of the maximum likelihood objective. Hence, we refer to the model as Variational Retrieval-Augmented Generation (VRAG).

One of the advantages of variational training is that it provides a low variance estimate of the objective (as compared to sampling from the document-prior distribution), for the same number of samples. Although this has been used in supervised settings ([Chen *et al.*, 2020] and [Kim *et al.*, 2020a]), we note that directly training under the variational objective may be prohibitively expensive in case of a very large[2] document collection (sampling an element from the posterior distribution, would require retrieval scores for each document in the index collection). In fact, related approaches for variational training therefore only use a small set of pre-retrieved documents [Lian *et al.*, 2019; Li *et al.*, 2020] to overcome this bottleneck. In particular, such approaches first use an out-of-the box retriever to fetch a small set of documents (typically 5-10 documents) from the entire document collection. The methods then learn a prior as well as posterior distribution over the small set of pre-retrieved documents only by optimizing the variational objective.

A major weakness of this approach is that the out-of-the-box retriever does not benefit from training. As a result, if the recall of the out-of-the box retriever is low, that is, the correct document is not present in the pre-retrieved subset for most of the training data, the mapping from the document to the response learnt will be highly noisy and not very useful (we also demonstrate this experimentally in this paper).

Contributions: In this paper we describe our approach called VRAG or Variational Retrieval-Augmented Generation[3] which allows us to extend variational optimization to cases where documents are retrieved from large document collections. Instead of pre-retrieving a small set of documents to facilitate variational training, we retrieve documents from the entire collection but perform a summation over the top-k retrieved documents from the posterior distribution as well as the prior distribution to approximate the variational objective (Figure 1b). Top-k retrieval can be performed efficiently using an index for nearest neighbor search such as Faiss [Johnson *et al.*, 2017], and we find that this simple trick performs significantly better than other approaches. We present experiments on three, publicly available, conversational QA datasets and we show that variational training helps build better knowledge grounded dialog systems. Our experiments show that not only does VRAG perform better on the end-task, it also learns a better retriever. To the best of our knowledge, we are the first to apply variational training for open-scale unsupervised knowledge grounded dialog systems.[4]

[2]Collections can have millions of documents

[3]We provide the code and the supplementary material at https://github.com/mayank31398/VRAG and https://arxiv.org/abs/2112.00653 respectively.

[4]Concurrently, [Paranjape *et al.*, 2022] also use variational training with RAG to generate responses

2 Background

As is commonly done in dialog modeling tasks, we represent the collection of dialogs as a set of context (dialog history) and response pairs; $\mathcal{T} = \{(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})\}_{i=1}^{m}$ where each context $\boldsymbol{x}^{(i)}$ as well as its response $\boldsymbol{y}^{(i)}$ is a sequence of tokens. Further let $\mathcal{D} = \{\boldsymbol{d}_j\}_{j=1}^{N}$ be a set of documents in the form of a large indexed document collection. We assume that each context-response pair requires exactly 1 document $\boldsymbol{d}_j \in \mathcal{D}$ (where $1 \leq j \leq N$) to generate the corresponding response. Let $\boldsymbol{z}^{(i)}$ denote a discrete variable which indicates the document (from the indexed collection) needed for training instance i i.e, $\boldsymbol{d}_{\boldsymbol{z}^i} \in \mathcal{D}$. We can now model the joint likelihood of a response and document pair $(\boldsymbol{y}^{(i)}, \boldsymbol{z}^{(i)})$ as $p(\boldsymbol{y}^{(i)}, \boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)}) = p(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})p(\boldsymbol{y}^{(i)}|\boldsymbol{x}^{(i)}, \boldsymbol{z}^{(i)})$.

In the absence of document-level supervision, the $\boldsymbol{z}^{(i)}$ variables are unknown or 'latent'. Here we will be maximizing $\log p(\boldsymbol{y}^{(i)}|\boldsymbol{x}^{(i)}) = \sum_{\boldsymbol{z}} p(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})p(\boldsymbol{y}^{(i)}|\boldsymbol{z}^{(i)}, \boldsymbol{x}^{(i)})$. However, since an explicit summation over the entire document collection can be computationally intractable, one needs to resort to a few approximation techniques. For ease of notation, we will drop the superscript (i) from now on.

Retrieval based approaches: Approaches such as RAG [Lewis *et al.*, 2020] and REALM [Guu *et al.*, 2020], maintain an index which allows one to retrieve the top-k documents with high prior probability. The objective is then approximated as a sum over these retrieved documents.

Specifically, the document-prior distribution $p(\boldsymbol{z}|\boldsymbol{x})$ is defined based on scores returned by the Dense Passage Retriever (DPR) [Karpukhin *et al.*, 2020].

The top-k most relevant documents (S_k^p) for a query (dialog context) are retrieved from an index that allows efficient retrieval [Johnson *et al.*, 2017] using MIPS search. We denote the approximate document-prior distribution, normalized over the set S_k^p, as $\hat{p}(\boldsymbol{z}|\boldsymbol{x})$. The overall objective for generating the response can then be written as $\log\left[\sum_{\boldsymbol{z}\in S_k^p} \hat{p}(\boldsymbol{z}|\boldsymbol{x})p(\boldsymbol{y}|\boldsymbol{z}, \boldsymbol{x})\right]$. RAG suffers from a drawback that it does not use the information from responses in order to retrieve documents for a given training instance.

Variational techniques: An alternative approach is to maximize a variational lower bound on the objective. Here we need to define an Evidence Lower Bound (ELBO) on the likelihood as $\log p(\boldsymbol{y}|\boldsymbol{x}) \geq \mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})}\left[log\left(\frac{p(\boldsymbol{y},\boldsymbol{z}|\boldsymbol{x})}{q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})}\right)\right]$. This lower bound holds for any distribution q. Variational autoencoders [Kingma and Welling, 2013] define another network to model the distribution q. To train such networks, the ELBO is split as:

$$\mathbb{E}_{\boldsymbol{z}\sim q}\left[\log p(\boldsymbol{y}|\boldsymbol{z}, \boldsymbol{x})\right] - KL\left[q||p(\boldsymbol{z}|\boldsymbol{x})\right] \tag{1}$$

The first term is an expectation that can be estimated by sampling documents from the document-posterior $q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{y})$ distribution. The response-likelihood network is then run only using these sampled documents. One can either use re-parameterization trick (with Gumbel softmax distribution) [Jang *et al.*, 2016; Maddison *et al.*, 2016] or policy gradient method to back propagate through the sampling step. However, in order to sample a document, one would need to compute the entire distribution over the documents. This can be

prohibitively expensive when using a large document collection. The second term (KL-divergence) is also computed as an explicit sum, given access to prior and posterior probability distributions, and is also intractable for large document collections.

In summary, variational training which uses the posterior distribution to retrieve documents while training, can help retrieve more relevant documents for training the response-generator. One of the trivial ways to extend these approaches (for large document corpus) is to identify the candidate knowledge documents for each training instance via an existing out-of-the-box retriever ([Li *et al.*, 2020], [Lian *et al.*, 2019]). One can then create prior and posterior distributions on the restricted set of documents. However this does not allow us to train the retriever and we also show in our experiments, that the trained distribution in such a setting does a poor job of generalizing as a retriever (Section 4).

In our approach we generalize the variational technique to open domain setting without fixing or pre-retrieving those candidate documents. In order to do so, we would first need to be able to compute the ELBO objective more efficiently (over the entire document corpus) as described in the next section.

3 Variational RAG (VRAG)

Variational training involves using both, the document-prior ($p(\boldsymbol{z}|\boldsymbol{x})$) and document-posterior distributions ($q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})$). We model each of these based on scores from a Dense Passage retriever (DPR) [Karpukhin *et al.*, 2020] i.e.

$$p(\boldsymbol{z}|\boldsymbol{x}) = \text{softmax}\left(f(\boldsymbol{z})^T g(\boldsymbol{x})\right) \tag{2}$$

$$\text{and } q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y}) = \text{softmax}\left(f(\boldsymbol{z})^T h(\boldsymbol{x},\boldsymbol{y})\right) \tag{3}$$

where f and g are parameterized representations of documents ($\boldsymbol{z}$) and dialog contexts ($\boldsymbol{x}$). $h(\boldsymbol{x},\boldsymbol{y})$ denotes the joint embedding of the context-response pair. These are created using neural models such as BERT [Devlin *et al.*, 2018]. We use a single network to compute the document embeddings $f(\boldsymbol{z})$ for both prior and posterior.

In order to efficiently compute expectation and KL divergence terms in the ELBO objective (Equation 1), we need to approximate the above distributions. To do so, we maintain an index on document embeddings. This allows us to retrieve the set of top-k documents under prior and posterior distributions (equations 2 and 3) using MIPS search [Johnson *et al.*, 2017]. Note that since the query embeddings g and h are trainable, the retriever is trained over the epochs as well. We denote the sets of top documents under prior and posterior distributions by S_k^p and S_k^q respectively.

The overall cost is then computed using Evidence Lower Bound (Equation 1). Here for the first term, we normalize $\hat{q}$ over the set of top-k documents (denoted by S_k^q) returned by the index when queried using the posterior (i.e, the response and the dialog context). The first term is then approximated as $\sum_{\boldsymbol{z}\in S_k^q} \hat{q}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y}) \log p(\boldsymbol{y}|\boldsymbol{z},\boldsymbol{x})$. To approximate the KL-divergence we use the top-k knowledge instances retrieved by querying the index using the dialog-context (for prior) and the context-response pair (for posterior). We then take union of the two sets S_k^p and S_k^q to form the set S_{KL}. We use this set (S_{KL}) to approximate the KL-divergence. Thus, the KL-divergence in Equation 1 is given by:

$$KL\left[\hat{q}||\hat{p}\right] = \sum_{\boldsymbol{z}\in S_{KL}} \hat{q}(\boldsymbol{z}) \log\left(\frac{\hat{q}(\boldsymbol{z})}{\hat{p}(\boldsymbol{z})}\right), \tag{4}$$

where the approximate posterior ($\hat{q}$) and prior ($\hat{p}$) in this case are obtained by normalizing the retrieval scores on S_{KL}. The intuition behind this approximation is that the documents with low posterior probability do not contribute much to the KL objective and hence, can safely be ignored. Similar to other variational models, VRAG is trained end-to-end.

3.1 Architecture

We now describe the neural networks used to model the prior and posterior distributions, and the response generator.

Prior Distribution Encoders: We need to create document and context representations defined by functions $f$and g, respectively for modelling the prior. We pass the input context through BERT model [Devlin *et al.*, 2018] to create context representation. We use special markers to separate the turns. The embedding at final layer of $[CLS]$ token is passed through a linear layer to create context representation. Similarly the document is passed through (separate) BERT Model to create a document representation at $[CLS]$ token.

Posterior Distribution Encoders: Similar to the modeling of the prior distribution, we use another (separate) BERT to model the posterior. Here we create the input representation of the context-response pair ($\boldsymbol{x}$, $\boldsymbol{y}$), where a special marker is used to separate the context and response.

Response Likelihood: We use the standard sequence to sequence formulation for response likelihood, where $\log p(\boldsymbol{y}|\boldsymbol{z},\boldsymbol{x}) = \sum_j \log p(\boldsymbol{y}_j|\boldsymbol{y}_{<j},\boldsymbol{z},\boldsymbol{x})$. Here, we use the GPT2 [Radford *et al.*, 2019] model as our decoder with input sequence consisting of context and response.

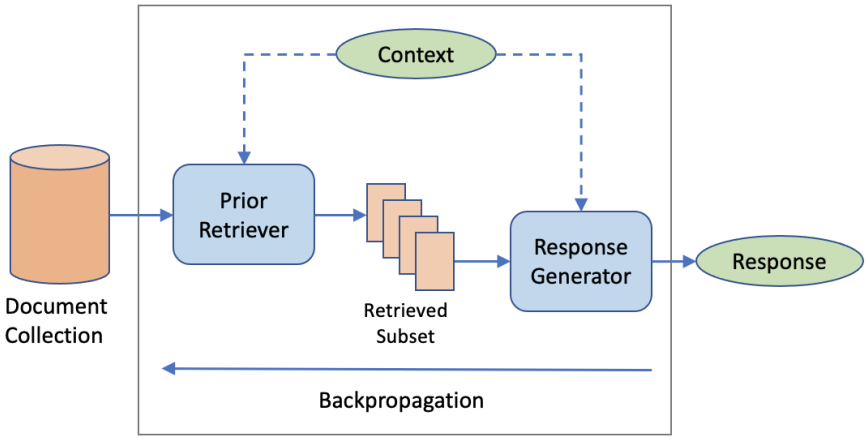
3.2 Training Details

We train our network to maximize the ELBO objective (Equation 1). We initialize our document-prior (for both RAG and VRAG) and document-posterior (for VRAG) networks with the pretrained DPR-Multiset model[5] pre-trained using data from the Natural Questions [Kwiatkowski *et al.*, 2019], TriviaQA [Joshi *et al.*, 2017] etc.

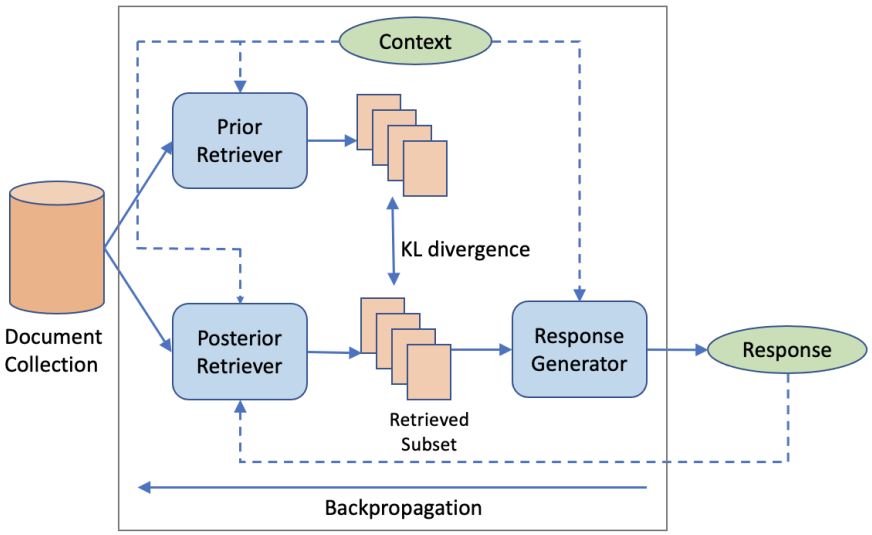
During the training of the model, it can be difficult to rebuild the document index after every change to the document representation parameters in f, therefore similar to Lewis *et al.*, the parameters in f are kept constant.

We used early stopping with $patience = 5$ on recall of the validation sets to prevent overfitting of models. The loss was optimized using AdamW Optimizer [Loshchilov and Hutter, 2017]. We also found it useful to continue training the response-likelihood for both RAG and VRAG after the joint training is complete. This is because while training, the decoder-likelihood function often lags behind prior (for RAG) and posterior (for VRAG).

[5]This can be used to initialize a model in Hugging Face, see https://huggingface.co/transformers/model_doc/dpr.html



(a) RAG [Lewis *et al.*, 2020]



(b) VRAG (Our approach)

Figure 1: (a) RAG does not use information from the responses to retrieve documents. (b) Our approach - VRAG which is trained end-to-end and uses the response to train a posterior distribution which guides the prior distribution

3.3 Response Generation

At test time we need to generate the response for a given context by first retrieving a document – thus, in both RAG and VRAG models we use the trained document-prior model to retrieve the top-k documents using the dialog context as query. We experiment with two different decoding strategies to generate the response:

1. **Top Document Decoding:** In this case the document with the highest prior probability is used to condition the generator. The response is then generated using beam search (beam width=3) on the trained GPT-2 response generation model; the most likely beam is taken to be the prediction of the model. We refer to this method as the 'top-1 decoding' in our experiments.
2. **Top-k Documents Decoding:** Here top (k=5) documents are retrieved from the prior distribution, say $\boldsymbol{z}_1, ..., \boldsymbol{z}_5$. A beam search is then run to generate the top response from each of these say $\boldsymbol{r}_1, ..., \boldsymbol{r}_5$. We use the estimate of $p(\boldsymbol{r}_i|\boldsymbol{x}) \approx p(\boldsymbol{z}|\boldsymbol{x})p(\boldsymbol{r}_i|\boldsymbol{z}, \boldsymbol{x})$. The most likely response under the estimated distribution is taken to be the response generated by the model[6]. We refer to this as 'top-5 decoding' in our experiments.

4 Experiments

Our experiments aim to answer the following questions: (1) Does Variational RAG (VRAG), which uses samples from the approximated document-posterior distribution, perform better than vanilla RAG? (2) Does the quality of generated responses improve by decoding using the top-k documents? (3) How do the trained document retriever modules for RAG

[6]This is the same as "Fast Decoding" as defined in [Lewis *et al.*, 2020].

and VRAG compare with each other? (4) Are the quality of samples returned by the document-posterior of VRAG better than document-prior of RAG as hypothesized? (5) How does VRAG compare with other approximations for variational training?

4.1 Datasets

OR-QuAC [Qu *et al.*, 2020]: This dataset is a modified version of the QuAC [Choi *et al.*, 2018] dataset. The dataset consists of dialog conversations, where each conversation is associated with the top-5 most relevant documents (retrieved using TF-IDF [Robertson *et al.*, 1994] based BM25 ranking) from the QuAC dataset. To create an open-scale collection for our task, we index the set of all the documents available in the train, validation and test splits. In some cases of the test and validation set, the ground-truth document may be missing in the top-5 list associated with each conversation. In such cases, we obtain the ground-truth document from the original QuAC dataset and add it to the indexed collection.

DSTC9 [Kim *et al.*, 2020b]: This dialog dataset was released as part of the DSTC9 challenge. The dataset comprises of dialog conversation turns in which the system: (i) needs to identify the turns in which to consult a collection for textual FAQs, (ii) retrieve the FAQ if required, (iii) and then generate the response based on the retrieved FAQ. The universe of knowledge documents in this dataset is the set of FAQs and each FAQ also includes the entity name because the same question can occur multiple times for different entities (eg: "Is parking available?"). The training dataset consists of conversations based on 4 different domains i.e hotels, restaurants, trains and taxis. The test dataset contains an additional domain, attractions, which is not found in the train and validation splits.

DoQA [Campos *et al.*, 2020]: This dataset comprises of open-ended dialog conversations on different domains like cooking, travel and movies. Unlike, the OR-QuAC and DSTC9 datasets, most questions in this dataset are not factoid/specific questions, and are open-ended. We only use the cooking split for both training and testing.

We preprocess all the datasets by removing all the examples if the ground truth response is "CANNOTANSWER" (unanswerable). Each 'instance' refers to a context-response pair.

Question-Answering Task: The OR-QuAC dataset also contains non-contextual variations of questions at each dialog turn and we use them in a QA setting (referred to as OR-QuAC-QA).

4.2 Baselines

Apart from our approach (VRAG), we also study the performance of RAG, as well as, a pipeline model which uses the pre-trained DPR-Multiset Retriever and a GPT2 based decoder which is fine-tuned to generate responses. We refer to this as the DPR + GPT2 baseline in our experiments. In addition, we also show the importance of using a trainable retriever for variational setting by comparing gainst a variational model where a fixed set of candidate documents are retrieved using a pre-trained DPR as a retriever (conditioned on context and response) during training [Lian *et al.*, 2019; Li *et al.*, 2020]. This model retrieves from the entire document collection using the prior distribution during inference. We refer to this baseline as Variational Model with pre-retrieval ("pre-VM"). Thus, the distributions being trained do not change the set of candidate documents used as training progresses. All models use their respective document-prior distributions during testing.

Dataset	Model	R@1	R@5	MRR@5
OR-QuAC [Qu *et al.*, 2020]	DPR	19.8	49.0	0.315
	pre-VM	17.71	38.61	0.264
	RAG	24.2	54.9	0.366
	VRAG	**26.0**	**56.9**	**0.388**
DSTC9 [Kim *et al.*, 2020b]	DPR	13.2	34.3	0.208
	pre-VM	28.32	42.76	0.339
	RAG	69.4	84.3	0.758
	VRAG	**73.3**	**87.0**	**0.792**
DoQA [Campos *et al.*, 2020]	DPR	54.3	71.7	0.612
	pre-VM	60.81	77.43	0.672
	RAG	61.8	79.0	0.687
	VRAG	**62.4**	**79.1**	**0.691**
OR-QuAC-QA [Qu *et al.*, 2020]	DPR	9.6	30.7	0.174
	pre-VM	6.27	17.26	0.104
	RAG	15.0	38.1	0.239
	VRAG	**16.4**	**41.0**	**0.261**

Table 1: Comparison of VRAG and RAG models in terms of retrieval accuracy (document recall $R@K$ and MRR scores) on all datasets.

4.3 Evaluation Metrics

For each of our experimental runs, we report the Mean Reciprocal Rank@5 (MRR@5), Recall@1 (R@1), Recall@5 (R@5) to evaluate prior's performance. We also report BLEU scores (both with top-1 and top-5 decoding) to assess the performance of generator. The BLEU-1 and BLEU-4 scores in our tables are denoted by B-1 and B-4 respectively. We also consider "BLEU-penalized" scores (indicated by BP-1 anad BP-4) which consider the BLEU score at a given test instance as 0 if the document retrieved for the given instance is incorrect. These help ensure that a model is not able to produce a high score by memorizing on a particular given domain.

4.4 Results

As can be seen in Table 1, VRAG outperforms RAG on each dataset for document retrieval. We also note that both RAG and VRAG significantly improve the performance of the initial DPR based retriever. Table 2 shows the performance of the models on the response generation task. Note that all results in Table 2 are obtained after further fine tuning the generator networks (after joint training is complete). The results show that the VRAG model outperforms the RAG model on language generation tasks (BLEU metrics) on all datasets except DoQA. We believe this difference is due to the nature of documents used in this dataset - other datasets have a lot more fact based questions to be answered using knowledge in documents, while more than 66% of the questions in the DoQA dataset are non-fact based and open-ended.

In addition, it is also possible that the RAG model is memorizing and overfitting on this dataset. For instance, see gains in penalized BLEU scores (BP-1 and BP-4) of RAG and VRAG over DPR in OR-QuAC and DSTC9 datasets in Table 2 – the relative gain of VRAG (over RAG) is significantly

		top-1 decoding				top-5 decoding			
Dataset	**Model**	**B-1**	**B-4**	**BP-1**	**BP-4**	**B-1**	**B-4**	**BP-1**	**BP-4**
OR-QuAC [Qu *et al.*, 2020]	DPR + GPT2	13.65	6.11	4.41	3.11	16.06	7.97	11.36	7.37
	RAG	12.88	5.94	4.60	3.03	15.39	7.64	11.72	7.21
	pre-VM	11.44	4.87	3.52	2.36	13.55	6.26	9.17	5.89
	VRAG	**13.97**	**7.58**	**5.61**	**4.02**	**16.30**	**9.11**	**13.10**	**8.72**
DSTC9 [Kim *et al.*, 2020b]	DPR + GPT2	31.84	7.21	4.37	1.08	31.81	7.17	11.14	2.60
	RAG	33.28	8.26	25.87	6.86	33.30	8.27	28.75	7.45
	pre-VM	31.57	7.16	9.92	2.66	31.87	7.29	14.7	3.98
	VRAG	**33.49**	**8.70**	**26.49**	**7.57**	**33.51**	**8.67**	**29.80**	**8.03**
DoQA [Campos *et al.*, 2020]	DPR + GPT2	21.26	14.31	17.83	14.20	22.60	15.73	20.53	15.62
	RAG	**23.59**	**17.04**	20.86	16.92	**24.27**	**17.73**	**22.75**	**17.60**
	pre-VM	23.33	16.70	20.47	16.5	23.79	17.21	22.24	17.07
	VRAG	23.38	17.02	**20.91**	**16.94**	23.29	16.88	21.93	16.80
OR-QuAC-QA [Qu *et al.*, 2020]	DPR + GPT2	9.11	1.88	1.81	1.07	10.18	2.61	4.95	2.38
	RAG	9.12	2.31	2.36	1.44	10.39	2.97	5.75	2.75
	pre-VM	6.96	1.17	1.11	0.67	7.84	1.75	3.05	1.67
	VRAG	**9.64**	**2.93**	**2.83**	**1.66**	**10.65**	**3.49**	**6.73**	**3.36**

Table 2: Comparison of VRAG and RAG models in terms of BLEU and BLEU-penalized on all datasets after decoder fine-tuning.

	B-1 (top-1 decoding)	B-1 (top-5 decoding)
DPR	-42.28%	-48.64%
RAG	-36.77%	-45.19%
pre-VM	-41.89%	-44.20%
VRAG	-43.22%	-49.88%

Table 3: Percentage drop in BLEU score when correct documents have been removed on OR-QuAC [Qu *et al.*, 2020] dataset.

higher. This suggests that VRAG model is more likely to generate the response using the correct document and not by merely memorize on a given domain.

Alternative approximations for Variational Training: In Table 1 we see that the recall scores of pre-VM model are much worse than our VRAG model. This observation validates our hypothesis that, because the retrieved samples are not improved during training, the prior and posterior distributions in pre-VM end up focusing only on the few (potentially incorrect) initially-retrieved documents. This is especially problematic if the initial recall is low. As training progresses, this may worsen the distributions over the initial model (eg: see DSTC9 recall scores for DPR and pre-VM in Table 1) because the correct document isn't present in the retrieved sample, thus giving it an incorrect signal.

Effect of Top-5 Decoding: From Table 2 we find that in almost all cases, using top-5 decoding to generate responses performs better than using the single (best scored) document to generate responses. This indicates that models are to able to incorporate information from the correct document even if it is not returned as the top-ranked document.

Benefit of Document Posterior: In Section 3, we motivated the VRAG model by suggesting that using the posterior to sample documents while training the decoder could help train a better model. We find that recall of the document posterior in VRAG is nearly 14-70% higher than the document prior of RAG (depending on the dataset). Further, we find that the use of the responses by VRAG (posterior), to query the index during training, results in significantly better retrieval accuracy than RAG (prior) at every epoch during training. While this is perhaps intuitive and expected, our results on recall in Table 2 demonstrate the VRAG (prior) which is trained indirectly via the KL-divergence with VRAG (posterior) also outperforms the RAG (prior). We attribute this gain to the fact that the generator in VRAG learns to focus well on the generated documents which further reinforces the posterior network (and indirectly the prior network through KL term) to improve.

Study of Memorization: One of the issues in such an unsupervised learning is that generator may fail to use the retrieved knowledge. Instead the parameters might have been trained to internally model the external knowledge sources themselves, refered to as 'memorization'. In order to study memorization in the models, we compared the results on response generation of all models in the absence of the correct document – if the correct document is missing, the models should perform very poorly. A good performance even without obtaining correct document, would indicate lesser reliance on external knowledge and hence a higher tendency for 'memorization'. We, thus rebuild the document index without including any of the documents from the test set and then re-evaluate the performance of our models. Table 3, shows the drop in BLEU scores for each of the models after removing the correct document on OR-QuAC dataset. As can be seen, the percentage drop is highest in the VRAG model indicating a higher usage of knowledge instance and thus, possibly lesser memorization.

5 Conclusion

In this paper we described an approach to run variational training on knowledge grounded dialog with a large corpus. Our experiments on three conversational QA datasets indicate that variational training is helpful as it produces better document samples while training. We find that our model, VRAG (having access to superior samples from posterior while training), not only generates better responses, it also learns a better retriever (prior distribution).

We believe that such sampling approximations could also be helpful in other tasks; for instance, it could also be interesting to apply them to other approaches such as Reinforcement Learning to the setting of a large corpus. This would require overcoming similar challenges in sampling as we did in this paper for variational training.

References

[Campos *et al.*, 2020] Jon Ander Campos, Arantxa Otegi, Aitor Soroa, Jan Deriu, Mark Cieliebak, and Eneko Agirre. DoQA - accessing domain-specific FAQs via conversational QA. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7302–7314, Online, July 2020. Association for Computational Linguistics.

[Chen *et al.*, 2020] Xiuyi Chen, Fandong Meng, Peng Li, Feilong Chen, Shuang Xu, Bo Xu, and Jie Zhou. Bridging the gap between prior and posterior knowledge selection for knowledge-grounded dialogue generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3426–3437, 2020.

[Choi *et al.*, 2018] Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. QuAC: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2174–2184, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

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

[Guu *et al.*, 2020] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. Realm: Retrieval-augmented language model pre-training. *arXiv preprint arXiv:2002.08909*, 2020.

[Jang *et al.*, 2016] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[Johnson *et al.*, 2017] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *arXiv preprint arXiv:1702.08734*, 2017.

[Joshi *et al.*, 2017] Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*, 2017.

[Karpukhin *et al.*, 2020] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*, 2020.

[Kim *et al.*, 2020a] Byeongchang Kim, Jaewoo Ahn, and Gunhee Kim. Sequential latent knowledge selection for knowledge-grounded dialogue. *arXiv preprint arXiv:2002.07510*, 2020.

[Kim *et al.*, 2020b] Seokhwan Kim, Mihail Eric, Karthik Gopalakrishnan, Behnam Hedayatnia, Yang Liu, and Dilek Hakkani-Tur. Beyond domain APIs: Task-oriented conversational modeling with unstructured knowledge access. In *Proceedings of the 21th Annual Meeting of SIGDIAL*, pages 278–289, 1st virtual meeting, July 2020. Association for Computational Linguistics.

[Kingma and Welling, 2013] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[Kwiatkowski *et al.*, 2019] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

[Lewis *et al.*, 2020] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *arXiv preprint arXiv:2005.11401*, 2020.

[Li *et al.*, 2020] Linxiao Li, Can Xu, Wei Wu, YUFAN ZHAO, Xueliang Zhao, and Chongyang Tao. Zero-resource knowledge-grounded dialogue generation. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 8475–8485. Curran Associates, Inc., 2020.

[Lian *et al.*, 2019] Rongzhong Lian, Min Xie, Fan Wang, Jinhua Peng, and Hua Wu. Learning to select knowledge for response generation in dialog systems. *arXiv preprint arXiv:1902.04911*, 2019.

[Loshchilov and Hutter, 2017] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[Maddison *et al.*, 2016] Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

[Paranjape *et al.*, 2022] Ashwin Paranjape, Omar Khattab, Christopher Potts, Matei Zaharia, and Christopher D Manning. Hindsight: Posterior-guided training of retrievers for improved open-ended generation. In *International Conference on Learning Representations*, 2022.

[Qu *et al.*, 2020] Chen Qu, Liu Yang, Cen Chen, Minghui Qiu, W. Bruce Croft, and Mohit Iyyer. *Open-Retrieval Conversational Question Answering*, page 539–548. Association for Computing Machinery, New York, NY, USA, 2020.

[Radford *et al.*, 2019] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[Robertson *et al.*, 1994] Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. Okapi at trec-3. In *TREC*, 1994.