interpretable methods for doc alignment in dialogue

Anonymous ACL submission

Abstract

TBD

1 Introduction

In many customer-facing dialogue applications, customer service interactions must follow safety guidelines. These guidelines often have a natural sequential order (Chen et al., 2021): If a customer is locked out of their account and requests a password reset, the agent must first verify that the customer is indeed the owner of the account.

All agents, whether human or robot, should follow safety guidelines. As a result, safety guidelines are often written in natural language.

Our goal is to train dialogue agents that not only follow a set of natural language guidelines, but explain their actions by pointing to the guidelines. This allows others to verify whether the guidelines have been followed.

We propose a dialogue model that explain sequential decisions with guideline-aligned rationales, and does not require rationale labels.

Experiments show that our model is accurate, intepretable, and works at a range of supervision levels.

We present results on three datasets, ranging over a variety of guideline styles. In ABCD, the guidelines are given to us Chen et al. (2021). In SGD, we write the guidelines ourselves, using the generative model to aid development. In doc2dial, we show that our method works for alignment to general document-guided dialogue as well.

2 Related work

The adaptation of large languge models to task-oriented dialogue has allowed for impressive results in zero-shot generalization, where models are tested in scenarios that they have not previously seen (). The key idea behind this success is the use of a natural language interface: specify scenario-specific details using natural language, and take advantage of the generalization abilities of large language models.

Rule following (SHARC, doc2dial)

Reading manuals for generalization (RTFM, Kartik's work).

3 Problem setup

Our goal is to, given a guideline-grounded dialogue x between a customer and agent, align each turn in x to a sentence z from document d from guidelines D.

4 Method 1: Dialogue given document

We start with a generative model of dialogue that justifies its actions by aligning to the guidelines as a whole, without sentence alignments.

The model first UNIFORMLY chooses a document in the guideline $d \sim p(d)$, then generates the dialogue $x \sim p(x \mid d)$. This yields the joint distribution $p(x, d) = p(x \mid d)p(d)$.

We perform training by optimizing the log marginal likelihood

$$\log \sum_{d} p(x, d). \tag{1}$$

We perform inference online via Bayes' rule:

$$\operatorname*{argmax}_{d} p(d \mid x) = \operatorname*{argmax}_{d} p(x \mid d)p(d), \tag{2}$$

which allows us to make predictions for any prefix of x.

Why not choose document alignments iid at every agent turn? We found that using a document to generate only the next agent turn resulted in

poor unsupervised accuracy. In particular, the document choice model would converge to a uniform distribution. Additionally, we found that many single turns were well-explained by a large number of different documents. The document should instead explain the dialogue as a whole, rather than an individual turn. It is these two points that led us to consider generating full dialogues given a single document.

4.1 Approximate inference in training

We perform additional experiments with an inference network $q(d|x)$ to speed up training. We approximate the following objective:

$$\log \sum_d p(x,d) - KL[p(d \mid x)||q(d \mid x)]. \quad (3)$$

This differs from the VAE objective, $\log p(x) - KL[q(d|x)||p(d|x)]$, as we use the forward KL. The forward KL is known to control the error of importance sampling (Chatterjee and Diaconis, 2018), and has been found to be useful in variational inference (Jerfel et al., 2021; Bornschein and Bengio, 2014). In particular, the reverse KL is mode-seeking, allowing the $q(d|x)$ to prefer an incorrect document even if $p(d|x)$ prefers the correct one.

We make approximations to equation 3 for computational efficiency. Rather than relying on importance sampling, we rely on a top-k approximation, and use this top-k approximation for approximating the evidence $p(x)$, posterior $p(z|x)$, and $q(z|x)$:

$$\log \sum_d p(x,d) \approx \log \sum_{d \in \tilde{D}} p(x,d)$$

$$KL[p(d \mid x)||q(d \mid x)] \approx KL[\tilde{p}(d \mid x)||\tilde{q}(d \mid x)]$$

$$\tilde{p}(d \mid x) = \frac{p(x,d)}{\sum_{d \in \tilde{D}} p(x,d)}$$

$$\tilde{q}(d \mid x) = \frac{q(d \mid x)}{\sum_{d \in \tilde{D}} q(d \mid x)},$$

so that the partition functions fo $\tilde{p}, \tilde{q}$ are approximated over $\tilde{D} = \text{argtopk}\, q(d \mid x)$.

We then optimize this objective with respect to the parameters of p and q via gradient descent, treating $\tilde{D}$ as a constant because the topk operation is not differentiable.

Why not VAE training? We found VAE training with the usual ELBO required a baseline and achieved worse accuracy with a leave-one-out baseline. There are two differences: First, our objective naturally uses a regret baseline, while the leave-one-out baseline can be interpreted as the advantage. Second, the VAE objective optimizes reverse KL, which has been shown to work worse than the forward KL (Jerfel et al., 2021).

Relation to wake sleep The wake-phase aims to find

$$\operatorname*{argmax}_p \mathbb{E}_{x \sim D} \left[\mathbb{E}_{z \sim q(z|x)} \left[\frac{\log p(x,z)}{q(z \mid x)} \right] \right], \quad (4)$$

and the sleep phase finds

$$\operatorname*{argmin}_q \mathbb{E}_{x,z \sim p(x,z)} \left[\log \frac{p(z \mid x)}{q(z \mid x)} \right], \quad (5)$$

the forward KL divergence between the model distribution p and q. In contrast, we optimize a unified objective for both p and q, and do not require samples from the generative model.

4.2 Parameterization

We parameterize $p(x|d)$ with a sequence to sequence model such as BART. The prior $p(d)$ is a uniform distribution. The inference network $q(d|x) \propto \langle \text{enc}(x), \text{enc}(d) \rangle$ encodes x, d with a transformer such as RoBERTa.

4.3 Experiments

We present an experiment on unsupervised document classification with a generative model of dialogue given document. We evaluate the document accuracy $d|x$ right before first agent action. An example of the first agent action is pulling up the customer's account, at which point the agent should be following a specific document in the guidelines. The agent should know what the correct document is before taking an action.

We truncate documents and dialogues to 256 tokens. If the first agent action does not happen in the first 256 tokens of the dialogue, we evaluate the document prediction at the 256th token.

We provide a fully supervised skyline as reference, which directly models $p_s(d|x) \propto \langle \text{emb}(d), \text{RoBERTa}(x) \rangle$, where we use a label embedding of the document d that does not see the document's text (a RoBERTa encoding of the document performed similarly). As a baseline, we use BM25 which is based on lexical similarity.

model	Training objective	N	doc acc	training time
Skyline supervised	$\log p_s(d\|x)$	8,034	90.7	-
Baseline lexical	-	0	34.1	-
Doc2Dialogue	$\log p(x)$	0	71.8	20h
Doc2Dialogue	$\log p(x) - KL[p\|\|q]$	0	75.8	14h

Table 1: Results for document classification with a generative model at the first agent action in a conversation. Documents and dialogues are truncated to the first 256 tokens. The number of labeled training examples is N.

In table 1, we see that both full marginalization and approximate inference outperform the lexical baseline, but do not reach the supervised skyline. One cause of this is that a portion of the documents in the guidelines are lexically identical to one another, resulting in a performance drop for the unsupervised models. The supervised skyline does not use the documents themselves and therefore does not suffer from this issue. Additionally, the unsupervised model has issues distinguishing between similar documents, such as changing the name of an account versus changing the address.

5 Method 2: Dialogue, sentence given document

The next step is to model sentence alignments given the oracle document. We start with a very simple alignment model, which breaks up the dialogue into turns $x = (x_1, \ldots, x_T)$. For every turn, we first choose an alignment uniformly $p(z_t|d)$, then generates the utterance for that turn $p(x_t|x_{<t}, z_t)$ using BART (the encoder input tokens are given by z_t and decoder input tokens by $x_{<t}$). The graphical model is given in figure 1.

We directly optimize the marginal likelihood

$$\log p(x_{1:T} \mid d) = \log \prod_t \sum_{zt} p(x_t \mid x_{<t}, z_t) p(z_t \mid d). \quad (6)$$

[1]

For each turn, we perform inference via Bayes' rule

$$\operatorname*{argmax}_{z_t} p(z_t \mid x, d) = \operatorname*{argmax}_{z_t} p(x_t \mid x_{<t}, z_t) p(z_t \mid d), \quad (7)$$

where $p(z_t|d)$ is uniform and can be dropped.

[1] A natural extension of this model is to instead model $p(z_t|x_{<t}, d)$, we can approximate the marginal likelihood by taking the top-k elements of this more informative prior.

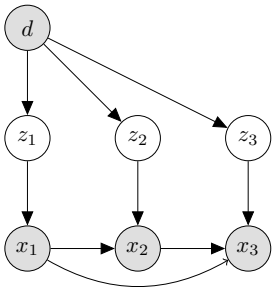

Figure 1: Graphical model for the first approach to sentence alignment.

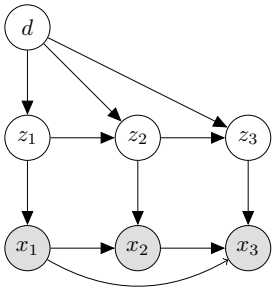

Figure 2: Graphical model for full document and span alignments.

5.1 Experiments

References

Jörg Bornschein and Yoshua Bengio. 2014. Reweighted wake-sleep.

Sourav Chatterjee and Persi Diaconis. 2018. The sample size required in importance sampling. *The Annals of Applied Probability*, 28(2):1099–1135.

Derek Chen, Howard Chen, Yi Yang, Alex Lin, and Zhou Yu. 2021. Action-based conversations dataset: A corpus for building more in-depth task-oriented dialogue systems. *CoRR*, abs/2104.00783.

Ghassen Jerfel, Serena Wang, Clara Wong-Fannjiang, Katherine A. Heller, Yian Ma, and Michael I. Jordan. 2021. Variational refinement for importance sampling using the forward kullback-leibler divergence. In *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161 of

Proceedings of Machine Learning Research, pages 1819–1829. PMLR.

A Experiment 1: Document classification from observed dialogue + action output

A.1 Research question

Can we do document classification in document-driven dialogue with no document labels?

A.2 Experiment

We present experiments on unsupervised document classification with a generative model of dialogue given document. We evaluate the document accuracy $d|x$ right before first agent action. An example of the first agent action is pulling up the customer's account, at which point the agent should be following a specific document in the guidelines. The agent should know what the correct document is before taking an action.

A.3 Models

- Skyline: supervised

$$p(d \mid x) \propto \langle \text{emb}(d_{label}), \text{BERT}(x) \rangle$$

- Baseline: lexical BM25
- Approximate marginalization with D^*: $\log \sum_{d \in D^*} p(x \mid d) p(d)$
  - D^* = {true d^*, 3 hard lexical negatives based on d^*, 3 random negatives}
  - Inference via Bayes' rule: $\text{argmax}_d \, p(x_{1:t}|d)$ where t is the index of the first agent action.
- Full marginalization over D: $\log \sum_{d \in D} p(x|d)p(d)$
  - all docs D
  - Inference via Bayes' rule
- Approximate marginalization with $q(d \mid x)$: $\log \sum_{d \in \tilde{D}} p(x|d)p(d) - KL[\tilde{p}(d \mid x)||\tilde{q}(d \mid x)]$
  - $\tilde{D}$ from top16 $q(d|x)$
  - Inference via Bayes'

model	acc	time
Skyline supervised $p(d|x)$	90.7	-
Baseline lexical	34.1	-
Approx marg w/ D^*	78	6h
Full marg w/ D	71.8	20h
Approx marg w/ top16 $q(d \mid x)$	75.8	14h

Table 2: Results for document classification with a generative model at the first agent action in a conversation.

A.4 Results

In table 2, we see that

- Full marg does better than lexical baseline, but worse than approximate marg over Z*.
- Approximate marg with q does better than full marg, but worse than approximate marg over Z*.

The approximate marg w/ Z* doing best is surprising, since the training setup of full marg (all D) is closer to the testing setup (all D), as $D^* \subset D$.

There are a few possible causes

1. There are reasonable negative documents in $D \setminus D^*$ that have $p(x|d) > p(x|d^*)$
2. The gradient will be larger for D^* since it is smaller, meaning the support will be smaller and the posterior will be sharper (gradient = posterior).

Error analysis found the the model does make mistakes in full marg and approx marg with q due to the existence of distracting negatives not considered in D^*. However, there are also other errors in approx marg with q that are not explained by this.

A.5 Model selection

Since we do not assume access to labeled examples, we perform model selection using the validation likelihood $p(x)$. The validation likelihood always decreases over the course of training, meaning we simply take the last model checkpoint after 10 epochs. This results in a decent amount of overfitting, meaning the presented results are not the highest accuracies over the course of training.

model	N	doc acc
Skyline supervised $p(d\|x)$	All	90.65
Approx marg w/ top16 $q(d\|x)$	0	74.2
Approx marg w/ top16 $q(d\|x)$	10	-
Approx marg w/ top16 $q(d\|x)$	50	-
Approx marg w/ top16 $q(d\|x)$	100	-

Table 3: Results for document classification with a generative model. N is the number of labeled examples seen during training.

B Experiment 2: Semi-supervised document classificaiton

B.1 Question

When writing a manual, we need at least one labeled example for each document in the manual. How many labeled examples do we need to recover supervised performance?

B.2 Experiment

We perform early document detection in the semi-supervised setting, where we have 50 paired (x, d) examples, and the remaining examples only have access to x. We report accuracy on $d|x$ at the first agent action.

B.3 Models

- Skyline: fully supervised $p(d|x)$
- Baseline: Approx marg w/ $q(d|x)$ with no d labels
- Approx marg w/ top16 $q(d|x)$ with $\{10, 50, 100\}$ d labels

Ways of utilizing labeled examples:

- Warm start: Train p and q on these labeled examples
- Interleave: Every M steps, train p and q on these examples

In order to prevent overfitting to the labeled examples, we use interleaving (find citation for KL reg / semisup).

B.4 Results

In table 3, we see that TBD

model	N	span acc
Baseline BM25	0	-
Approx marg w/ top16 $q(a\|x, d)$	0	-
Approx marg w/ top16 $q(a\|x, d)$	10	-
Approx marg w/ top16 $q(a\|x, d)$	50	-

Table 4: Results for document classification with a generative model. N is the number of labeled examples seen during training.

C Experiment 3: Span alignment

C.1 Question

When writing a manual, we need at least one labeled example for each document in the manual. How many labeled examples do we need to recover supervised performance?

C.2 Results

D Experiment 4: Downstream AST accuracy

E Experiment 5: Downstream response generation

F Experimental setup

G Results

H Sasha questions about doc classification (12/27)

- Isn't your model p(x, z)
  - Yes, the model is $p(x \mid z)p(z)$, with $p(z)$ uniform.
- I don't really like this experiment, because it seems to test two different things: 1) keeping the z^* in the true set, 2) approximating the marginalization. A clean experiment would be Full Marginalization vs. Approx Marginalization during training. The one that keeps around Z^* is a skyline at best, and maybe at worst not informative.
  - The approximate marginalization with Z^* will not be included in the final results, but was useful for debugging full marg and will be useful for debugging the VAE setting.

  - That said, this is a clean experiment. Only one thing is changed: the set of negatives. Approx marg w/ Z^* uses z^* and some negatives, while full marg uses z^* and all negatives. Full marg vs VAE approx marg would change both the negatives as well as whether z^* is guaranteed to be present.

- I would like your conclusions to be a little bit more clear about things like speed and methods. Is Full Marg reasonable or not?
  - Speed: Full marg takes between 5-10 hours to reach peak validation document accuracy This is reasonable for this setting, but will become a limitation in models that must perform both sentence and document marginalization.
  - General resaonableness: Full marg is reasonable as long as it fits within memory constraints. It is reasonable for this dataset, but may not be for the other datasets.

- The name "Approx Marg" does not really make sense here, as again approx would be a version of this with the $Z*$
  - Approximate marginalization with Z^* describes the setting $\log \sum_{z \in Z^*} p(x, z), Z^* \subset Z$. Marginalization over Z is approximated over the restriction Z^*. I believe this is a precise description without jargon.

- You are much too early to worry about hyperparams, that discussion should not even be here yet.
  - I managed to get accuracy up a few points, but nothing major. Other learning rate settings resulted in very poor performance for this experiment, as fine-tuning is sensitive to hyperparameters.

- I don't really get this line "This is surprising, since the training setup (all Z) is closer to the testing setup (all Z), as Z* is a strict subset of Z". This doesn't seem surprising to me?
  - It is hard to predict whether approximate marginalization with Z^* vs full marginalization with Z would yield a better model.
  - $p(x|z)$ will learn to prefer z^* if $p(x|z^*)$ is better than other $p(x|z)$, since the gradient of the log marginal likelihood objective is the posterior $p(z|x)$ and the model has a uniform $p(z)$.
  - When would approx marg w/ Z* do better? If Z^* contains hard negatives z with $p(x|z^*) > p(x|z)$ but not negatives $p(x|z^*) < p(x|z)$, so that the model doesnt learn to prefer those hard negatives over the true z^*. This seems to be the case here.
  - When would approx marg w/ Z^* do worse? If Z^* misses some hard negatives with $p(x|z^*) > p(x|z)$.

- please provide a section in these documents with "parameterization". Is $p(z)$ parameterized?
  - $p(z)$ is uniform and therefore has no learnable parameters.