Training Millions of Personalized Dialogue Agents

Pierre-Emmanuel Mazaré, Samuel Humeau, Martin Raison, Antoine Bordes
Facebook
{pem, samuelhumeau, raison, abordes}@fb.com

Abstract

Current dialogue systems are not very engaging for users, especially when trained end-to-end without relying on proactive reengaging scripted strategies. Zhang et al. (2018) showed that the engagement level of end-to-end dialogue models increases when conditioning them on text personas providing some personalized back-story to the model. However, the dataset used in (Zhang et al., 2018) is synthetic and of limited size as it contains around 1k different personas. In this paper we introduce a new dataset providing 5 million personas and 700 million persona-based dialogues. Our experiments show that, at this scale, training using personas still improves the performance of end-to-end systems. In addition, we show that other tasks benefit from the wide coverage of our dataset by fine-tuning our model on the data from (Zhang et al., 2018) and achieving state-of-the-art results.

1 Introduction

End-to-end dialogue systems, based on neural architectures like bidirectional LSTMs or Memory Networks (Sukhbaatar et al., 2015) trained directly by gradient descent on dialogue logs, have been showing promising performance in multiple contexts (Wen et al., 2016; Serban et al., 2016; Bordes et al., 2016). One of their main advantages is that they can rely on large data sources of existing dialogues to learn to cover various domains without requiring any expert knowledge. However, the flip side is that they also exhibit limited engagement, especially in chit-chat settings: they lack consistency and do not leverage proactive engagement strategies as (even partially) scripted chatbots do.

Zhang et al. (2018) introduced the PERSONA-CHAT dataset as a solution to cope with this issue. This dataset consists of dialogues between pairs of agents with text profiles, or *personas*, attached to each of them. As shown in their paper, conditioning an end-to-end system on a given persona improves the engagement of a dialogue agent. This paves the way to potentially end-to-end personalized chatbots because the personas of the bots, by being short texts, could be easily edited by most users. However, the PERSONA-CHAT dataset was created using an artificial data collection mechanism based on Mechanical Turk. As a result, neither dialogs nor personas can be fully representative of real user-bot interactions and the dataset coverage remains limited, containing a bit more than 1k different personas.

In this paper, we build a very large-scale persona-based dialogue dataset using conversations previously extracted from REDDIT[1]. With simple heuristics, we create a corpus of over 5 million personas spanning more than 700 million conversations. We train persona-based end-to-end dialogue models on this dataset. These models outperform their counterparts that do not have access to personas, confirming results of Zhang et al. (2018). In addition, the coverage of our dataset seems very good since pre-training on it also leads to state-of-the-art results on the PERSONA-CHAT dataset.

2 Related work

With the rise of end-to-end dialogue systems, personalized trained systems have started to appear. Li et al. (2016) proposed to learn latent variables representing each speaker's bias/personality in a dialogue model. Other classic strategies include extracting explicit variables from structured knowledge bases or other symbolic sources as in (Ghazvininejad et al., 2017; Joshi et al., 2017; Young et al., 2017). Still, in the context of per-

[1] `https://www.reddit.com/r/datasets/comments/3bxlg7/`

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2775–2779
Brussels, Belgium, October 31 - November 4, 2018. ©2018 Association for Computational Linguistics

sonal chatbots, it might be more desirable to condition on data that can be generated and interpreted by the user itself such as text rather than relying on some knowledge base facts that might not exist for everyone or a great variety of situations. PERSONA-CHAT (Zhang et al., 2018) recently introduced a dataset of conversations revolving around human habits and preferences. In their experiments, they showed that conditioning on a text description of each speaker's habits, their *persona*, improved dialogue modeling.

In this paper, we use a pre-existing REDDIT data dump as data source. REDDIT is a massive online message board. Dodge et al. (2015) used it to assess chit-chat qualities of generic dialogue models. Yang et al. (2018) used response prediction on REDDIT as an auxiliary task in order to improve prediction performance on natural language inference problems.

3 Building a dataset of millions of persona-based dialogues

Our goal is to learn to predict responses based on a persona for a large variety of personas. To that end, we build a dataset of examples of the following form using data from REDDIT:

- Persona: [*"I like sport"*, *"I work a lot"*]
- Context: *"I love running."*
- Response: *"Me too! But only on weekends."*

The persona is a set of sentences representing the personality of the responding agent, the context is the utterance that it responds to, and the response is the answer to be predicted.

3.1 Preprocessing

As in (Dodge et al., 2015), we use a preexisting dump of REDDIT that consists of 1.7 billion comments. We tokenize sentences by padding all special characters with a space and splitting on whitespace characters. We create a dictionary containing the 250k most frequent tokens. We truncate comments that are longer than 100 tokens.

3.2 Persona extraction

We construct the persona of a user by gathering all the comments they wrote, splitting them into sentences, and selecting the sentences that satisfy the following rules: (i) each sentence must contain between 4 and 20 words or punctuation marks, (ii) it contains either the word *I* or *my*, (iii) at least one verb, and (iv) at least one noun, pronoun or adjective.

To handle the quantity of data involved, we limit the size of a persona to N sentences for each user. We compare four different setups for persona creation. In the *rules* setup, we select up to N random sentences that satisfy the rules above. In the *rules + classifier* setup, we filter with the rules then score the resulting sentences using a bag-of-words classifier that is trained to discriminate PERSONA-CHAT persona sentences from random comments. We manually tune a threshold on the score in order to select sentences. If there are more than N eligible persona sentences for a given user, we keep the highest-scored ones. In the *random from user* setup, we randomly select sentences uttered by the user while keeping the sentence length requirement above (we ignore the other rules). The *random from dataset* baseline refers to random sentences from the dataset. They do not necessarily come from the same user. This last setup serves as a control mechanism to verify that the gains in prediction accuracy are due to the user-specific information contained in personas.

In the example at the beginning of this section, the response is clearly consistent with the persona. There may not always be such an obvious relationship between the two: the discussion topic may not be covered by the persona, a single user may write contradictory statements, and due to errors in the extraction process, some persona sentences may not represent a general trait of the user (e.g. *I am feeling happy today*).

3.3 Dataset creation

We take each pair of successive comments in a thread to form the context and response of an example. The persona corresponding to the response is extracted using one of the methods of Section 3.2. We split the dataset randomly between training, validation and test. Validation and test sets contain 50k examples each. We extract personas using training data only: test set responses cannot be contained explicitly in the persona.

In total, we select personas covering 4.6m users in the rule-based setups and 7.2m users in the random setups. This is a sizable fraction of the total 13.2m users of the dataset; depending on the persona selection setup, between 97 and 99.4 % of the training set examples are linked to a persona.

4 End-to-end dialogue models

We model dialogue by next utterance retrieval (Lowe et al., 2016), where a response is picked among a set of candidates and not generated.

4.1 Architecture

The overall architecture is depicted in Fig. 1. We encode the persona and the context using separate modules. As in Zhang et al. (2018), we combine the encoded context and persona using a 1-hop memory network with a residual connection, using the context as query and the set of persona sentences as memory. We also encode all candidate responses and compute the dot-product between all those candidate representations and the joint representation of the context and the persona. The predicted response is the candidate that maximizes the dot product.

We train by passing all the dot products through a softmax and maximizing the log-likelihood of the correct responses. We use mini-batches of training examples and, for each example therein, all the responses of the other examples of the same batch are used as negative responses.

4.2 Context and response encoders

Both context and response encoders share the same architecture and word embeddings but have different weights in the subsequent layers. We train three different encoder architectures.

Bag-of-words applies two linear projections separated by a $\tanh$ non-linearity to the word embeddings. We then sum the resulting sentence representation across all positions in the sentence and divide the result by $\sqrt{n}$ where n is the length of the sequence.

LSTM applies a 2-layer bidirectional LSTM. We use the last hidden state as encoded sentence.

Transformer is a variation of an End-to-end Memory Network (Sukhbaatar et al., 2015) introduced by Vaswani et al. (2017). Based solely on attention mechanisms, it exhibited state-of-the-art performance on next utterance retrieval tasks in dialogues (Yang et al., 2018). Here we use only its encoding module. We subsequently average the resulting representation across all positions in the sentence, yielding a fixed-size representation.

		hits@k		
	Persona	k=1	k=3	k=10
IR Baseline	No	5.6	9.9	19.5
BOW	No	51.7	64.7	77.9
BOW	Yes	53.9	67.9	81.9
LSTM	No	63.1	75.6	87.3
LSTM	Yes	66.3	79.5	90.6
Transformer	No	69.1	80.7	90.7
Transformer	Yes	**74.4**	**85.6**	**94.2**

Table 1: Test results when classifying the correct answer among a total of 100 possible answers.

4.3 Persona encoder

The persona encoder encodes each persona sentence separately. It relies on the same word embeddings as the context encoder and applies a linear layer on top of them. We then sum the representations across the sentence.

We deliberately choose a simpler architecture than the other encoders for performance reasons as the number of personas encoded for each batch is an order of magnitude greater than the number of training examples. Most personas are short sentences; we therefore expect a bag-of-words representation to encode them well.

5 Experiments

We train models on the persona-based dialogue dataset described in Section 3.3 and we evaluate its accuracy both on the original task and when transferring onto PERSONA-CHAT.

5.1 Experimental details

We optimize network parameters using Adamax with a learning rate of $8e{-4}$ on mini-batches of size 512. We initialize embeddings with FastText word vectors and optimize them during learning.

REDDIT LSTMs use a hidden size of 150; we concatenate the last hidden states for both directions and layers, resulting in a final representation of size 600. Transformer architectures on reddit use 4 layers with a hidden size of 300 and 6 attention heads, resulting in a final representation of size 300. We use Spacy for part-of-speech tagging in order to verify the persona extraction rules. We distribute the training by splitting each batch across 8 GPUs; we stop training after 1 full epoch, which takes about 3 days.

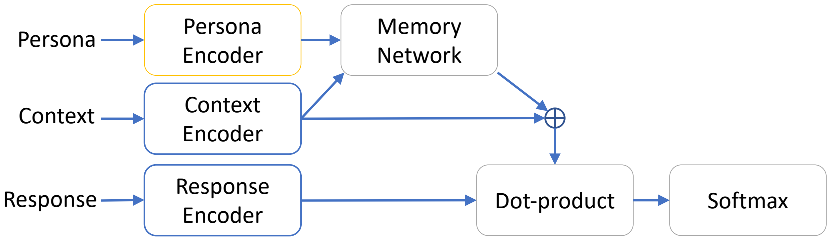

Figure 1: Persona-based network architecture.

Context (Persona)	*Predicted Answer*
Where do you come from?	
(I was born in London.)	*I'm from London, studying in Scotland.*
(I was born in New York.)	*I'm from New York.*
What do you do?	
(I am a doctor.)	*I am a sleep and respiratory therapist.*
(I am an engineer.)	*I am a software developer.*

Table 2: Sample predictions from the best model. In all selected cases the persona consists of a single sentence. The answer is constrained to be at most 10 tokens and is retrieved among 1M candidates sampled randomly from the training set.

PERSONA-CHAT We used the revised version of the dataset where the personas have been rephrased, making it a harder task. The dataset being only a few thousands samples, we had to reduce the architecture to avoid overfitting for the models trained purely on PERSONA-CHAT. 2 layers, 2 attention heads, a dropout of 0.2 and keeping the size of the word embeddings to 300 units yield the highest accuracy on the validation set.

IR Baseline As basic baseline, we use an information retrieval (IR) system that ranks candidate responses according to a TF-IDF weighted exact-match similarity with the context alone.

5.2 Results

Impact of personas We report the accuracy of the different architectures on the reddit task in Table 1. Conditioning on personas improves the prediction performance regardless of the encoder architecture. Table 2 gives some examples of how the persona affects the predicted answer.

Influence of the persona extraction In Table 3, we report precision results for several persona extraction setups. The *rules* setup improves the results somewhat, however adding the persona classifier actually degrades the results. A possible interpretation is that the persona classifier is trained only on the PERSONA-CHAT revised personas, and that this selection might be too narrow and lack diversity. Increasing the maximum persona size also improves the prediction performance.

N	Persona selection	hits@1
0	–	69.1
20	rules + classifier	70.7
20	rules	71.3
100	rules + classifier	72.5
100	rules	**74.4**
100	random from user	73.8
100	random from dataset	66.9

Table 3: Retrieval precision on the REDDIT test set using a Transformer and different persona selection systems. N: maximum number of sentences per persona.

Transfer learning We compare the performance of transformer models trained on REDDIT and on PERSONA-CHAT on both datasets. We report results in Table 4. This architecture provides a strong improvement over the results of (Zhang et al., 2018), jumping from 35.4% hits@1 to 42.1%. Pretraining the model on REDDIT and then fine-tuning on PERSONA-CHAT pushes this score to 60.7%, largely improving the state of the art. As expected, fine-tuning on PERSONA-CHAT reduces the performance on REDDIT. However, directly testing on PERSONA-CHAT the model trained on REDDIT without fine-tuning yields a very low result. This could be a consequence of a discrepancy

	Validation set	
Training set	PERSONA-CHAT	REDDIT
PERSONA-CHAT	42.1	3.04
REDDIT	25.6	**74.4**
FT-PC	**60.7**	65.5
IR Baseline	20.7	5.6
(Zhang et al., 2018)	35.4	–

Table 4: hits@1 results for the best found Transformer architecture on different test sets. FT-PC: REDDIT-trained model fine-tuned on the PERSONA-CHAT training set. To be comparable to the state of the art on each dataset, results on PERSONA-CHAT are computed using 20 candidates, while results on REDDIT use 100.

between the style of personas of the two datasets.

6 Conclusion

This paper shows how to create a very large dataset for persona-based dialogue. We show that training models to align answers both with the persona of their author and the context improves the predicting performance. The trained models show promising coverage as exhibited by the state-of-the-art transfer results on the PERSONA-CHAT dataset. As pretraining leads to a considerable improvement in performance, future work could be done fine-tuning this model for various dialog systems. Future work may also entail building more advanced strategies to select a limited number of personas for each user while maximizing the prediction performance.

References

Antoine Bordes, Y-Lan Boureau, and Jason Weston. 2016. Learning end-to-end goal-oriented dialog. *arXiv preprint arXiv:1605.07683*.

Jesse Dodge, Andreea Gane, Xiang Zhang, Antoine Bordes, Sumit Chopra, Alexander Miller, Arthur Szlam, and Jason Weston. 2015. Evaluating prerequisite qualities for learning end-to-end dialog systems. *arXiv preprint arXiv:1511.06931*.

Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen-tau Yih, and Michel Galley. 2017. A knowledge-grounded neural conversation model. *CoRR*, abs/1702.01932.

Chaitanya K Joshi, Fei Mi, and Boi Faltings. 2017. Personalization in goal-oriented dialog. *arXiv preprint arXiv:1706.07503*.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A persona-based neural conversation model. *CoRR*, abs/1603.06155.

Ryan Lowe, Iulian V Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. On the evaluation of dialogue systems with next utterance classification. *arXiv preprint arXiv:1605.05414*.

Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*, volume 16, pages 3776–3784.

Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, and Rob Fergus. 2015. End-to-end memory networks. *Proceedings of NIPS*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Tsung-Hsien Wen, David Vandyke, Nikola Mrksic, Milica Gasic, Lina M Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2016. A network-based end-to-end trainable task-oriented dialogue system. *arXiv preprint arXiv:1604.04562*.

Yinfei Yang, Steve Yuan, Daniel Cer, Sheng-yi Kong, Noah Constant, Petr Pilar, Heming Ge, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Learning semantic textual similarity from conversations. *CoRR*, abs/1804.07754.

Tom Young, Erik Cambria, Iti Chaturvedi, Minlie Huang, Hao Zhou, and Subham Biswas. 2017. Augmenting end-to-end dialog systems with commonsense knowledge. *arXiv preprint arXiv:1709.05453*.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? *CoRR*, abs/1801.07243.