Conversational Contextual Cues:
The Case of Personalization and History for Response Ranking

Rami Al-Rfou and **Marc Pickett** and **Javier Snaider** and **Yun-hsuan Sung** and **Brian Strope** and **Ray Kurzweil**

Google Inc,
1600 Amphitheatre Parkway,
Mountain View, CA 94043, USA
{rmyeid, pickett, jsnaider, yhsung, bps, raykurzweil}@google.com

Abstract

We investigate the task of modeling open-domain, multi-turn, unstructured, multi-participant, conversational dialogue. We specifically study the effect of incorporating different elements of the conversation. Unlike previous efforts, which focused on modeling messages and responses, we extend the modeling to long context and participant's history. Our system does not rely on handwritten rules or engineered features; instead, we train deep neural networks on a large conversational dataset. In particular, we exploit the structure of Reddit comments and posts to extract 2.1 billion messages and 133 million conversations. We evaluate our models on the task of predicting the next response in a conversation, and we find that modeling both context and participants improves prediction accuracy.

1 Introduction

Designing conversational systems is a challenging task and one of the original goals of Artificial Intelligence (Turing, 1950). For decades, conversational agent design was dominated by systems that rely on knowledge bases and rule-based mechanisms to understand human inputs and generate reasonable responses (Weizenbaum, 1966; Parkinson et al., 1977; Wallace, 2009). Data-driven approaches emphasize learning directly from corpora of written or spoken conversations. Recently, this approach gained momentum because of data abundance (Serban et al., 2015b), increasing computational power, and better learning algorithms that automate the feature engineering process (Schmidhuber, 2015; LeCun et al.,

Context 1:	I live in a village.
Context 2:	I live in Chicago.
Input:	*Are you going to watch the bulls?*

Response	Score with	
	Context 1	Context 2
I am planning to visit the farm soon.	**98.35**	93.91
I am going to watch them on TV.	94.24	**95.35**

Table 1: Our ranker utilizing the context to disambiguate the words *watch* and *bulls* and adjusting the scores of the candidate responses accordingly.

2015).

Here, we study how modeling dialogue is influenced by the history within a conversation, and participants' histories across their conversations. Recent work in data-driven models focuses on modeling the next response as a function of the preceding message (Vinyals and Le, 2015; Li et al., 2015). We extend previous models in two new directions. First, we model the history of what has been said before the last message, termed ***context***. This allows the model to include medium-term signals, presumably references and entities, which disambiguate the most recent information. As the conversation continues and the context grows, we expect our model to make better predictions of the next message (See Table 1). Second, to capture longer-term contextual signals, we model each user's personal history across all the conversations in which he or she participated in. We refer to this information as ***personal history***. The model can personalize its predictions depending on specific users' opinions, interests, experiences, and styles of writing or speaking. Both of these contextual signals give us the ability to make better predic-

tions regarding future responses.

Characterizing users, language, discourse coherence, and response diversity requires huge datasets and large models. To gather conversations at scale, we turn to web forums as a source of data. Specifically, we extract conversations from Reddit, a popular social news networking website. The website is divided into sub-forums (subreddits), each of which has its own theme of topics and interests. Registered users can submit URLs or questions, comment on a topic or on other users' comments, and vote on submissions or comments. Unlike previous efforts that used Twitter as a source of conversations (Ritter et al., 2010), Reddit does not have length constraints, allowing more natural text. We extracted 133 million posts from 326K different subforums, consisting of 2.1 billion comments. This dataset is several orders of magnitude larger than existing datasets (Serban et al., 2015b).

Instead of modeling message generation directly, the current work focuses on the ranking task of "*response selection*." At each point in the conversation, the task is to pick the correct next message from a pool of random candidates. Picking the correct next message is likely to be correlated with implicit understanding of the conversation. We use **Precision@k** to characterize the accuracy of the system. We train a deep neural network as a binary classifier to learn the difference between positive, real examples of input / response pairs, and negative, random examples of input / response pairs. The classifier's probabilities are used as scores to rank the candidates. This ranker will choose the response with the highest score.

Unlike generative approaches, where the modeling focus can be dominated by within-sentence language modeling, our approach encourages the system to discriminate between the connections of an actual response to the current conversation, and the lack of connections from a random response. Our model ranks candidates given any subset of the features we discussed so far (i.e., user models, conversation history, or the previous message). We also jointly learn a shared word embedding space (Bengio et al., 2006) and a user embedding space. With this arrangement, the models share common dynamics across users, giving us better models of conversations, and avoiding the need to construct a different

Order	Count	Avg. Score	Unique users	Comment
1	52259	1.84	47537	Thanks!
2	50923	6.85	43808	Yes.
4	35415	7.14	31141	:(
8	26976	2.72	24169	Why?
72	6422	4.93	5838	I don't get it.
88	5285	7.27	5085	I love you.
243	2559	2.07	2482	/s
267	2419	11.10	2132	[deleted]
-	1	3	1	Methane hydrates ALSO destroy ozone and there are huge pulses of those when the ice melts it might be that. Do your own research. Xenon fluoride does react with ozone as does the iodine. If you can't find it online try phoning some atmospheric scientists.

Table 2: Reddit comments, in 2014, sorted by their frequency. Frequent comments tend to be short. Users can up-vote and down-vote the score of a comment.

model for each user.

To summarize, our contributions are:

- We model users' responses over long histories that consist of their contributions over the years in various subforums and discussions. Furthermore, we integrate this model to offer better predictions.
- We model the conversation history beyond the current message. We study the length of the modeled history on the performance of our model.
- We outline a direct path to train and use a discriminative classifier as a response ranker.
- We demonstrate how to use Reddit's comment structure to extract complex dialogues on a large scale. We use a scalable neural network architecture that is able to take advantage of the large data size.

In Section 2, we discuss recent relevant work in data-driven modeling of dialogue systems. Section 3 discusses the Reddit dataset's diversity and scale and the steps we took to process the raw data. In Section 4, we discuss our choices in conversation modeling with deep neural networks. We discuss our experimental setup in Section 5, analyze our results in Section 6, and conclude our discussion in Section 7.

2   Related Work

Ritter et al. (2010) proposed a data-driven approach for building dialogue systems, and they extracted 1.3 million conversations from Twitter with the aim of discovering dialogue acts. Building on the distributional similarities of the vector space model frame-

work, Banchs and Li (2012) built a search engine to retrieve the most suitable response for any input message. Other approaches focused on domain specific tasks such as games (Narasimhan et al., 2015) and restaurants (Wen et al., 2016; Cuayáhuitl, 2016)

Personalizing dialogue systems requires sufficient information from each user and a sufficient user population to define the space. Writing styles quantified by word length, verb strength, polarity, and distribution of dialogue acts have been used to model users (Walker et al., 2012). Other efforts focused on building a user profile based on demographics, such as gender, income, age, and marital status (Bonin et al., 2014). Because Reddit users are pseudo-anonymous, we differ from these approaches by learning the relevant features to model the users' dialogue behavior through embedding each user into a distributed representation.

With the introduction of the sequence-to-sequence framework (Sutskever et al., 2014), many recent learning systems have used recurrent neural networks (RNNs) to generate novel responses given an input message or sentence. For example, Vinyals and Le (2015) proposed using IT desk chat logs as a dataset to train LSTM network to generate new sentences. Sordoni et al. (2015) constructed Twitter conversations limiting the history context to one message. With the help of pre-trained RNN language models, they encoded each message into a vector representation. To eliminate the need for a language model, Serban et al. (2015a) tried end-to-end training on an RNN encoder-decoder network. They also bootstrapped their system with pre-trained word embeddings.

While these systems are able to produce novel responses, it is difficult to understand how much capacity is consumed by modeling natural language versus modeling discourse and the coherence of the conversation. Often responses gravitate to the most frequent sentences observed in the training corpus (Li et al., 2015).

Perplexity, BLEU, and deltaBLEU, adapted from language modeling and machine translation communities, have been used for evaluating novel responses (Yao et al., 2015; Sordoni et al., 2015; Galley et al., 2015). These metrics only measure the response's lexical fluency and do not penalize for incoherent candidates with regard to the conversational

Figure 1: A diagram of the Reddit comment tree structure (Reddit Post). `User_Z` responded to the message produced by `User_Y` (**blue**). If we follow the ancestors of the input message, we can construct several contexts of different lengths (**green**).

discourse. While the search for better metrics is still on going, automatic evaluation of response generation stays an open problem (Shang et al., 2015).

Recall@k or **Precision@k** are commonly used for measuring a ranker's performance on the task of response selection. Typically, a positive response is mixed with random responses, and then the system is asked to score the right response higher than others (Hu et al., 2014; Kadlec et al., 2015). This task measures the model's ability to discriminate what goes together and what does not. As these metrics are better understood, we focus on the response selection task in our modeling effort.

3 Reddit Dataset

As conversational data-driven models are growing in popularity, datasets are increasing in number and size. However, most are small and domain specific. Serban et al. (2015b) surveyed 56 datasets and found that only 9 have more than 100,000 conversations, only one having more than 5 million conversations. This limits the complexity and the capacity of the models we can train. To target open-domain conversations, we need larger datasets. So far, there has been some limited effort to exploit the rich structure of Reddit. For example, Schrading et al. (2015) extracted comments from a small number of subreddits to build a classifier that identifies domestic

(a) Users' comments   (b) Reddit post size

(c) Replies per comment.   (d) Comment depth.

Figure 2: Reddit dataset statistics. For each metric, we calculate the cumulative distribution of comments that satisfy the criteria. For example, Figure 2b shows that $\sim 50\%$ of Reddit posts have less than 100 comments.

abuse content.

Unlike the Ubuntu dataset, logs of technical chat rooms (Lowe et al., 2015), Reddit conversations tend to be more diverse in regard to topics and user backgrounds. There are more than 300 thousands sub-forums (subreddits) with different topics of discussion. Compared to Twitter, Reddit conversations tend to be more natural, as there is no limit on message size (See Table 2).

Figure 1 shows how Reddit conversations are organized as a tree or "Reddit post." Users can comment on each other's comments indefinitely, leading to long conversations, which help us construct significant dialogue history. We construct a conversation by traversing the tree starting at any node, up through its parent and ancestors until we reach the first comment (i.e., the tree's *root*). Since users cannot comment unless they are registered with a unique user name, we use those names as our labels for learning our user embedding vectors to personalize the dialogue system. Note that users tend to be pseudo-anonymous. They do not use their real names and they participate on the website without sharing private identifying information.

Specifically, we use a public crawl of the reddit website[1]. Figure 2 shows that the dataset is hugely

diverse and complex. Figure 2a shows that the website has both irregular contributors and heavy users who have a large number of comments. Unlike the datasets surveyed in (Serban et al., 2015b), a comment can have several user generated responses. While these diverse responses by no means cover the space of reasonable responses, this property may help our models in generalization (See Figure 2c).

4 Models

We define a conversation $\mathcal{C}$ to be a sequence of k pairs of Messages and participants (Authors) $\mathcal{C} \equiv ((M_1, A_1), (M_2, A_2), \ldots, (M_k, A_k))$. Here, a message M_i is a sequence of a variable number of words $M_i = (w_{i1}, w_{i2}, \ldots w_{il})$. A_i and w_j are random variables taking values in the user population $\mathcal{P}_{user}$ and the word vocabulary $\mathcal{V}_{word}$, respectively. $\mathcal{P}_{user}$ is a fixed set of the most frequent authors in reddit, it is basically, a dictionary of their usernames that is used to index the author embedding matrix. Every vector is limited to only one user.

To represent messages we use bag of words technique over recurrent or convolutional networks for its speed and ability to scale to a dataset as large as Reddit. To improve bag of words capability of keeping track of words' order and sentence structure, we define $\mathcal{V}_{ngram}$ to be a dictionary of a subset of ngrams defined over the word vocabulary.

The first step in our modeling is to map each user in our population $\mathcal{P}_{user}$ and each word in our vocabulary $\mathcal{V}_{word}$ to a vector of d dimensions. Specifically, we define $\phi_{user} : A_i \mapsto \mathbb{R}^{d_A}$ to be the embedding of the user A_i and $\phi_{ngram} : (w_i, w_j, \ldots) \mapsto \mathbb{R}^{d_n}$ to be the embedding of the ngram $(w_i, w_j, \ldots)$. For a sequence of k messages, we define the bag of ngrams embedding ($\psi \in \mathbb{R}^{d_n}$) to be the average of the embeddings of the ngrams extracted from all the messages:

$$\psi(M_1, \ldots, M_k) = \frac{1}{L} \sum_{1 \leq j \leq k} \sum_{g \in ngrams(M_j)} \phi_{ngram}(g)$$

(1)

where L is the total number of ngrams extracted from all the messages $\{M_1, \ldots, M_k\}$. Next, for a sequence of message-participant pairs of length k, we define the following features:

Figure 3: Single loss model.

Figure 4: Multi-loss model.

- **Response** : $\mathbf{R} = \psi(M_k)$ where M_k is the last message in the sequence.
- **Input Message**: $\mathbf{I} = \psi(M_{k-1})$ where M_{k-1} is the message that the response is addressing.
- **Context**: $\mathbf{C} = \psi(M_1, M_2, \ldots M_{k-2})$ where $(M_1, M_2, \ldots M_{k-2})$ is the subsequence of messages that preceded the input message.
- **Author**: $\mathbf{A} = \phi_{user}(A_k)$ where A_k is the user who generated the response message.

In Reddit, for each message in the post tree, we consider its parent to be the input message and its parent's ancestors to be the context. The content of the message is the response and the user that wrote the message is the author.

4.1 Response Ranking

To measure the effect of our features on modeling conversations, our task is to select the best response out of a pool of random candidates. This selection process could be viewed as a ranking problem. There are several approaches to ranking: pointwise, pairwise, and list-wise (Liu, 2009). Kadlec et al. (2015) chose pointwise ranking for its simplicity, and we follow the same choice for its speed benefits, which are necessary for training on hundreds of millions of examples. In pointwise ranking, we consider the compatibility of only one candidate at a time. Specifically, we learn a model that estimates the probability of a candidate given a subset of the features $\{\mathbf{I}, \mathbf{C}, \mathbf{A}\}$.

To construct the training dataset, we form pairs of features and responses. For each response appearing in the corpus, we form two pairs. The first is composed of the features with the observed re-

sponse $(\{\mathbf{I}, \mathbf{C}, \mathbf{A}\}, \mathbf{R})$. In the second pair, we replace the response with another random response sampled from our corpus, $(\{\mathbf{I}, \mathbf{C}, \mathbf{A}\}, \mathbf{R}')$. The first pair is used as a positive example and the second is a negative one.

4.2 Single-loss Network

To estimate the probability of the response appearing given the features, we train a binary logistic regression classifier. Figure 3 shows a network that concatenates the previous features into one input vector $input = [\mathbf{I}; \mathbf{C}; \mathbf{A}; \mathbf{R}]$. Then, several hidden layers with Relu non-linearities follow to produce a hidden layer $\mathbf{h}$. Given the hidden layer $\mathbf{h}$, we estimate the probability of the response, as follows:

$$\Pr(\mathbf{R}|\mathbf{I}, \mathbf{C}, \mathbf{A}) \approx \sigma(\mathbf{W}\mathbf{h} + \mathbf{b}) \qquad (2)$$

Where σ is the sigmoid function $\sigma(x) = 1/(1 + e^{-x})$. We call this model a single-loss model because it makes one prediction using all the information available.

4.3 Multi-loss Network

We can formalize the previous single-loss model network further by declaring $\Pr(\mathbf{R}|\mathbf{x}) \approx Network(\mathbf{x})$, where $\mathbf{x}$ is the input feature vector. The network uses a logistic regression layer on top of a feed-forward neural network. Figure 4 shows the multi-loss architecture that could be viewed as a network of networks. This architecture is achieved by replicating the single-loss architecture ($Network$) three times for each feature. Each of the networks makes

predictions taking into account one feature at a time. Furthermore, each network produces a hidden layer ($\mathbf{h^i}$) that will be used in an aggregate network. The aggregate network concatenates the hidden layers from the previous networks, $[\mathbf{h^1; h^2; h^3}]$, to produce a final hidden layer $\mathbf{h^4}$. This allows the final prediction to take advantage of all the features jointly. This network also allows us to measure the performance of each feature alone. This modular architecture facilitates diagnosis of any possible problems during training.

More specifically, the networks represent the following classification problems:

$$\Pr(\mathbf{R|I}) \approx \sigma(\mathbf{W_1 h^1 + b_1})$$
$$\Pr(\mathbf{R|C}) \approx \sigma(\mathbf{W_2 h^2 + b_2})$$
$$\Pr(\mathbf{R|A}) \approx \sigma(\mathbf{W_3 h^3 + b_3})$$
$$\Pr(\mathbf{R|I, C, A}) \approx \sigma(\mathbf{W_4 h^4 + b_4})$$

We use only the final prediction in the evaluation $\Pr(\mathbf{R|A, C, I})$, but we penalize the model with the sum of all predictions' losses.

5 Experimental Setup

We extract 2.1 Billion comments that were posted on the Reddit website between 2007 and 2015. We group the comments by their post page, treating each Reddit post as a tree rooted at the title of the post. We generate a positive example from each comment in the post. The example features are calculated by looking at the message's attributes, its parent, and its ancestors in the tree. We exclude Reddit posts that have more than 1000 comments for computational reasons. Most of these large posts are "Megathreads", each containing hundreds of thousands of comments. We do not generate examples for comments with empty or missing input message features. We also exclude examples where the author is not in our user population $\mathcal{P}_{user}$, or the user profile was deleted. After this filtering, 550 million positive examples are yielded. For each positive example, we generate a negative example by replacing the response feature by a random comment from Reddit.

5.1 Vocabulary

Reddit comments are written in markdown markup language. First, we remove the markdown and then,

tokenize the textual content. We normalize URLs and then include in our vocabulary the most frequent 200K unigrams and 200K bigrams. The count of the least frequent unigram is 1229, and for the least frequent bigram is 27670. For the author embeddings, we construct a user population ($\mathcal{P}_user$) of the most frequent contributing 400K users. The least contributing user created 922 comments. This population is, essentially, a dictionary of the user names.

5.2 Training

We set the ngram embedding and the user embedding space to 300 dimensions. The single loss model consists of one network, while the multiloss network consists of four networks. Each network consists of three hidden layers of size $[500, 300, 100]$. The hidden layer parameters, the ngram embeddings, and the user embeddings are trained jointly, and we use stochastic gradient descent (SGD) to optimize the parameters of our models (Bottou, 1991). The derivatives are estimated using the backpropagation algorithm and the updates are applied asynchronously. The learning rate α for SGD is set to 0.03. The models are implemented using Tensor-Flow (Abadi et al., 2015). Despite that each model is trained on 5 GPUs, the training time still takes several weeks due to data size.

We split our dataset into three partitions: train, dev, and test. The training dataset consists of 90% of the data and the rest is divided between dev and test. We train our classifier for one epoch, which is equal to 1 Billion examples. We stop training our models when we observe no significant improvement in the accuracy of our binary classifier on the dev dataset.

As our binary classifier will be evaluated on ranking candidate responses, we extract 10,000 examples from our test dataset. For each example, we sample $N - 1$ random responses from the pool. Given N of candidates, the classifier is asked to give the highest probability to the positive candidate available in the pool. We report precision (**P@1**) as a metric of quality.

6 Discussion & Results

In this section, we discuss the gains achieved by integrating the conversation history as well as the participants' history into our modeling. We con-

Context Length	Model			
	Single-loss		Multi-loss	
	N		N	
Up To	10	100	10	100
0	74.45	47.92	75.15	49.16
1	78.38	51.80	80.16	55.97
2	79.23	52.30	81.30	56.25
5	78.41	**52.84**	81.35	**56.39**
10	79.32	50.74	81.70	55.25
25	**79.70**	51.70	**81.71**	55.52

Table 3: Precision @ 1 for models trained on different context lengths and tested on two different sizes of candidates pools.

Feature	Model			
	Single-loss		Multi-loss	
	N		N	
	10	100	10	100
message	74.45	47.92	75.15	49.16
message + context	79.70	51.70	81.71	55.52
message + author	79.52	53.03	83.25	60.53
All	**82.72**	**55.91**	**86.60**	**63.53**

Table 4: **P@1** improvement gained by adding author and/or context to the base model. We consider context length to be 25.

messages in the tree.

6.2 Personalization

Table 4 shows larger gains in our rankers' precision when using the author feature compared to the conversational history (context) feature. The multi-loss model improves by 5 points in the task of ranking 100 response candidates. The author vector represents longer historical information than the current conversation history. Personal history could include interests, opinions, demographics, writing style, and personality traits. These could be essential in determining if a response is appropriate.

Finally, if we use all the features available to us, we get further improvement in performance over any of the features used alone. This highlights that the information we recover from each feature is different.

6.3 Multi-loss Vs Single-loss

The motivation behind the multi-loss model is to prevent adaptation between features (Hinton et al., 2012). In the single-loss model, the author feature could be subsumed for many cases with the input message and the context. Only subtle cases may require knowing the author identity to determine if the response is suitable. This slows the learning process of good author vectors. Therefore, the multi-loss network requires that the author vector should capture enough information to perform the prediction task, solely. This architecture extends the *deep supervision* idea where companion objective function is introduced to train intermediate layers in a deep network (Lee et al., 2015). Notice how the author feature outperforms the context feature in all tasks

trast both approaches and contrast their qualities and show a final model that takes advantage of both. Then, we show the effect of increasing the training dataset size on our models performance.

6.1 Length of the Context

How far back do we need to look to improve the quality of our ranking? To test that, we train both models discussed in Section 4 on several datasets with context features that vary in temporal scope. Table 3 shows the Precision @ 1 for both models using two different ranking tasks, the first involves 10 candidates and the second has 100 candidates. Context of length 0 corresponds to using only the input message as a feature. Each model was trained and tested on examples that included a conversation history (**context length**) *up to* m number of messages and not necessarily all the messages in the training or the test included the same history length.

First, we observe clear gains when we integrate the context feature (**C**). **P@1** increases by 4-6 points the moment we include the message that preceded the input message. However, we see a diminishing return as the context increases, particularly when the context is larger than 5 messages. In this case, there could be two factors at work. First, the more messages we use, the larger the number of vectors we average; this tends to blur the features and increase the information loss compared to the insight we gain. Second, we have less training and a smaller number of test examples that could take advantage of a long history. Figure 2d shows that more than 90% of the reddit comments haver a lower depth than 6

Figure 5: Losses contributed by each feature in the multi-loss network. Combining all the features always produce lower loss.

with the introduction of the multi-loss model. The multi-loss model is also easier to debug and probe. By reporting every loss on its own, we can see the development of the network.

Figure 5 shows the loss contributed by each feature. Notice, how the author vector takes more than 100 million examples to start influencing the prediction task. We conjecture that this behavior is the result of two factors. First, the author distribution does not follow Zipf's law, as language does. There is no small number of authors that could cover most of the examples. Second, author vectors depend. indirectly, on the content of the comments they posted. Unless the representation of the language, and consequently messages, are stable, we cannot learn a aggregate representation of the user's set of messages. This multi-stage learning is similar to what McClelland and Rogers (2003) observed in their work.

6.4 New users

In our evaluation we did not consider the case of unknown users. However, if a new user is encountered by our model, we can add a randomly initialized vector as a temporary representation. As the conversation goes on, we can then refine user vector using backpropagation while the rest of the model parameters are fixed. This technique is similar to the paragraph vector's strategy of dealing with new paragraphs after training is finished (Le and Mikolov, 2014).

(a) Learning curve.   (b) Classifier Accuracy Vs. Ranking Quality.

Figure 6: Effect of the training dataset size on the binary classifier accuracy, and therefore, the ranker precision. We used the multi-loss model with all the features and N set to 100.

6.5 Learning Curves

Figure 6a shows the improvement of the classifier accuracy by increasing the training dataset size orders of magnitude at a time. The results we presented so far would not have been possible without the billion examples we extracted from Reddit. It is quite clear that our models would have performed poorly given the other previously used datasets given their small sizes.

Moreover, the accuracy of the binary classifier is correlated highly with the **P@1** of the rankers we evaluated. We found that the pearson correlation between accuracy observed on the dev dataset and **P@1** of the ranker tested on the test dataset is both strong and positive, between $+0.94$ and $+0.99$. Therefore, we may infer the future gains of increasing the size of the dataset on the quality of the ranker (See Figure 6b).

7 Conclusion

Using Reddit, our model is trained on a significantly larger conversational dataset than previously published efforts. We train two scalable neural network models using bags of ngram embeddings and user embeddings. We measure significant improvement in the task of selecting the next response by integrating what has been said in the conversation so far. We study the effect of the length of the conversation history on performance. We also personalize the selection process by learning an identity feature for each user. This yields further improvement as it models the longer history of what a user has said in all conversations. Finally, our multi-loss model shows improvements over the baseline single-loss model using any subset of the features.

References

[Abadi et al.2015] Martın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. 2015. Tensorflow: Large-scale machine learning on heterogeneous systems, 2015. *Software available from tensorflow. org*.

[Banchs and Li2012] Rafael E. Banchs and Haizhou Li. 2012. Iris: a chat-oriented dialogue system based on the vector space model. In *Proceedings of the ACL 2012 System Demonstrations*, pages 37–42, Jeju Island, Korea, July. Association for Computational Linguistics.

[Bengio et al.2006] Yoshua Bengio, Holger Schwenk, Jean-Sébastien Senécal, Fréderic Morin, and Jean-Luc Gauvain. 2006. Neural probabilistic language models. In *Innovations in Machine Learning*, pages 137–186. Springer.

[Bonin et al.2014] Francesca Bonin, Jose San Pedro, and Nuria Oliver. 2014. A context-aware nlp approach for noteworthiness detection in cellphone conversations. In *COLING*, pages 25–36.

[Bottou1991] Léon Bottou. 1991. Stochastic gradient learning in neural networks. In *Proceedings of Neuro-Nîmes 91*, Nimes, France. EC2.

[Cuayáhuitl2016] Heriberto Cuayáhuitl. 2016. Simpleds: A simple deep reinforcement learning dialogue system. *CoRR*, abs/1601.04574.

[Galley et al.2015] Michel Galley, Chris Brockett, Alessandro Sordoni, Yangfeng Ji, Michael Auli, Chris Quirk, Margaret Mitchell, Jianfeng Gao, and Bill Dolan. 2015. deltableu: A discriminative metric for generation tasks with intrinsically diverse targets. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 445–450, Beijing, China, July. Association for Computational Linguistics.

[Hinton et al.2012] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

[Hu et al.2014] Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional neural network architectures for matching natural language sentences. In Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2042–2050. Curran Associates, Inc.

[Kadlec et al.2015] Rudolf Kadlec, Martin Schmid, and Jan Kleindienst. 2015. Improved deep learning baselines for ubuntu corpus dialogs. *arXiv preprint arXiv:1510.03753*.

[Le and Mikolov2014] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In Tony Jebara and Eric P. Xing, editors, *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1188–1196. JMLR Workshop and Conference Proceedings.

[LeCun et al.2015] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature*, 521(7553):436–444.

[Lee et al.2015] Chen-Yu Lee, Saining Xie, Patrick W. Gallagher, Zhengyou Zhang, and Zhuowen Tu. 2015. Deeply-supervised nets. In *AISTATS*, volume 38 of *JMLR Proceedings*. JMLR.org.

[Li et al.2015] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2015. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*.

[Liu2009] Tie-Yan Liu. 2009. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3):225–331.

[Lowe et al.2015] Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. 2015. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. *arXiv preprint arXiv:1506.08909*.

[McClelland and Rogers2003] James L McClelland and Timothy T Rogers. 2003. The parallel distributed processing approach to semantic cognition. *Nature Reviews Neuroscience*, 4(4):310–322.

[Narasimhan et al.2015] Karthik Narasimhan, Tejas Kulkarni, and Regina Barzilay. 2015. Language understanding for text-based games using deep reinforcement learning. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1–11, Lisbon, Portugal, September. Association for Computational Linguistics.

[Parkinson et al.1977] Roger C Parkinson, Kenneth Mark Colby, and William S Faught. 1977. Conversational language comprehension using integrated pattern-matching and parsing. *Artificial Intelligence*, 9(2):111–134.

[Ritter et al.2010] Alan Ritter, Colin Cherry, and Bill Dolan. 2010. Unsupervised modeling of twitter conversations. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 172–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Schmidhuber2015] Jürgen Schmidhuber. 2015. Deep learning in neural networks: An overview. *Neural Networks*, 61:85–117.

[Schrading et al.2015] Nicolas Schrading, Cecilia Ovesdotter Alm, Ray Ptucha, and Christopher Homan. 2015. An analysis of domestic abuse discourse on reddit. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2577–2583, Lisbon, Portugal, September. Association for Computational Linguistics.

[Serban et al.2015a] Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2015a. Building end-to-end dialogue systems using generative hierarchical neural network models. *arXiv preprint arXiv:1507.04808*.

[Serban et al.2015b] Iulian Vlad Serban, Ryan Lowe, Laurent Charlin, and Joelle Pineau. 2015b. A survey of available corpora for building data-driven dialogue systems. *arXiv preprint arXiv:1512.05742*.

[Shang et al.2015] Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. *arXiv preprint arXiv:1503.02364*.

[Sordoni et al.2015] Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. *arXiv preprint arXiv:1506.06714*.

[Sutskever et al.2014] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

[Turing1950] Alan M Turing. 1950. Computing machinery and intelligence. *Mind*, 59(236):433–460.

[Vinyals and Le2015] Oriol Vinyals and Quoc Le. 2015. A neural conversational model. *arXiv preprint arXiv:1506.05869*.

[Walker et al.2012] Marilyn Walker, Grace Lin, and Jennifer Sawyer. 2012. An annotated corpus of film dialogue for learning and characterizing character style. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 1373–1378, Istanbul, Turkey, May. European Language Resources Association (ELRA). ACL Anthology Identifier: L12-1657.

[Wallace2009] Richard S Wallace. 2009. *The anatomy of ALICE*. Springer.

[Weizenbaum1966] Joseph Weizenbaum. 1966. Elizaa computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45.

[Wen et al.2016] T.-H. Wen, D. Vandyke, N. Mrksic, M. Gasic, L. M. Rojas-Barahona, P.-H. Su, S. Ultes, and S. Young. 2016. A Network-based End-to-End Trainable Task-oriented Dialogue System. *ArXiv e-prints*, April.

[Yao et al.2015] Kaisheng Yao, Geoffrey Zweig, and Baolin Peng. 2015. Attention with intention for a neural network conversation model. *arXiv preprint arXiv:1510.08565*.