LSTM based Conversation Models

Yi Luan[1], *Yangfeng Ji*[2], *Mari Ostendorf*[1]

[1]Department of Electrical Engineering, University of Washington, Seattle, WA 98195
[2]School of Interactive Computing, Georgia Institute of Technology, Atlanta, GA 30332
luanyi@uw.edu, jiyfeng@gatech.edu, ostendor@uw.edu

Abstract

In this paper, we present a conversational model that incorporates both context and participant role for two-party conversations. Different architectures are explored for integrating participant role and context information into a Long Short-term Memory (LSTM) language model. The conversational model can function as a language model or a language generation model. Experiments on the Ubuntu Dialog Corpus show that our model can capture multiple turn interaction between participants. The proposed method outperforms a traditional LSTM model as measured by language model perplexity and response ranking. Generated responses show characteristic differences between the two participant roles.

1. Introduction

As automatic language understanding and generation technology improves, there is increasing interest in building human-computer conversational systems, which can be used for a variety of applications such as travel planning, tutorial systems or chat-based technical support. Most work has emphasized understanding or generating a word sequence associated with a single sentence or speaker turn, potentially leveraging the previous turn. Beyond local context, language use in a goal-oriented conversation reflects the global topic of discussion, as well as the respective role of each participant. In this work, we introduce a conversational language model that incorporates both local and global context together with participant role.

In particular, participant roles (*or* speaker roles) impact content of a sentence in terms of both the information to be communicated and the interaction strategy, affecting both meaning and conversational structure. For example, in broadcast news, speaker roles are shown to be informative for discovering the story structures [1]; they impact speaking time and turn-taking [2]; and they are associated with particular phrase patterns [3]. In online discussions, speaker role is useful for detecting authority claims [4]. Other work shows that in casual conversations, speakers with different roles are likely to use different discourse markers [5]. For the Ubuntu technical support data used in this study, Table 1 illustrates differences in the distributions of frequent words for the poster vs. responder roles. The POSTER role tends to raise questions using words *anyone*, *how*. The RESPONDER role tends to use directive words (*you*, *you're*), hedges (*may*, *might*) and words related to problem solving (*sudo*, *check*).

Specifically, we propose a neural network model that builds on recent work in response generation, integrating different methods that have been used for capturing local (previous sentence) context and more global context, and extending the network architecture to incorporate role information. The model can be used as a language model, as in speech recognition or translation, but our focus here is on response generation. Experiments are conducted with Ubuntu chat logs, using language model perplexity and response ranking, as well as qualitative analysis.

2. Related Work

Data-driven methods are now widely used for building conversation systems. With the popularity of social media, such as Twitter, Sina Weibo, and online discussion forums, it is easier to collect conversation text [6, 7]. Several different data-driven models have been proposed to build conversation systems. Ritter *et al.* [8] present a statistical machine translation based conversation system. Recently, neural network models have been explored. The flexibility of neural network models opens the possibility of integrating different kinds of information into the generation procedure. For example, Sordoni *et al.* [9] present a way to integrate contextual information via feed-forward neural networks. Li *et al.* propose using Maximum Mutual Information (MMI) as the objective function in neural models in order to produce more diverse and interesting responses. Shang *et al.* [10] introduce the attention mechanism into an encoder-decoder network for a conversation model. Most similar to our work is the Semantic Controlled LSTM (SC-LSTM) proposed by Wan *et al.* [11], where a Dialog-act component is introduced into the LSTM cell to guide the generated content. In this work, we utilize the role information to bias response generation without modifying LSTM cells.

Efficiently capturing local and global context remains a open problem in language modeling. Different ways of modeling document-level context has been explored in [12] and [13] based on the LSTM framework. Luan *et al.* [14] proposed a multi-scale recurrent architecture to incorporate both word and turn level context for spoken language understanding tasks. In this paper, we use a similar approach as [16], explicitly using Latent Dirichlet Analysis (LDA) as global-context feature to feed into RNNLM.

Early work on incorporating local context in conversational language modeling is described in [17] conditioned on the most recent word spoken by other speakers. Hutchinson *et al.* [18, 19] improve log-bilinear language model by introducing a multi-factor sparse matrix that could capture speaker role and topic information. In addition, Huang *et al.* [20] show that language models with role information significantly reduce word error rate in speech recognition. Our work differs from these approaches in using an LSTM. Recently, Li *et al.* propose using an additional vector to LSTM in order to capture personal characteristics of a speaker [21]. In this work, we utilize both a global topic vector and role information, where a role-specific weight matrix biases the word distributions for different roles.

$p(w\|Poster)/p(w\|Responder)$	hi, hello, anyone, hey, guys, ideas, thanks, thank, my, how, am, ??, cannot, I'm, says
$p(w\|Responder)/p(w\|Poster)$	you're, your, probably, you, may, might, sudo, ->, search, sure, ask, maybe, most, check, try

Table 1: Top 15 words based on the role likelihood ratio out of the subset with word count > 6k.

3. Model

In this section, we propose an LSTM based framework that integrating participant role and global topic of the conversation. As discussed in section 1, the assumption is, given the same context, each role has its own preference of picking words to generate a response. Each generated response should be both topically related to the current conversation and coherent with the local context.

3.1. Recurrent Neural Network Language Models

We start building a response generation model [9] by using a recurrent neural network language model (RNNLM) [22]. In general, a RNNLM is a generative model of sentences. For a sentence consisted of word sequence $x_1, \ldots, x_I$, the probability of x_i given $x_1, \ldots, x_{i-1} \triangleq \boldsymbol{x}_{\leq i-1}$ is

$$p(x_i|\boldsymbol{x}_{\leq i-1}) \propto g_\tau(\boldsymbol{h}_i) \tag{1}$$

where $\boldsymbol{h}_i$ is the current hidden state and $g_\tau(\cdot)$ is the probability function parameterized by τ:

$$g_\tau(\boldsymbol{h}_i) = \text{softmax}(\mathbf{W}_\tau \boldsymbol{h}_i), \tag{2}$$

where $\mathbf{W}_\tau$ is the output layer parameter. The hidden state $\boldsymbol{h}_i$ is computed recurrently as

$$\boldsymbol{h}_i = \boldsymbol{f}_\theta(x_i, \boldsymbol{h}_{i-1}). \tag{3}$$

$\boldsymbol{f}_\theta(\cdot)$ is a nonlinear function parameterized by θ. We use an LSTM [23] since it is good at capturing long-term dependency, which is an objective for our conversation model.

3.2. Conversation Models with Speaker Roles

To build a conversation model with different participant roles, we extend a RNNLM in two respects. First, to capture the variability from different participant roles, we incorporate role-based information into the generation procedure. Second, to model a conversation instead of single turns, our model adjoins RNNLMs for all turns in sequence to model the whole conversation.

More specifically, consider two adjacent turns[1] $\boldsymbol{x}_{t-1} = \{x_{t-1,i}\}_{i=1}^{N_{t-1}}$ and $\boldsymbol{x}_t = \{x_{t,i}\}_{i=1}^{N_t}$ with their participant role r_{t-1} and r_t respectively. N_t is the number of words in the t-th turn. To build a single model for the entire conversation, we simply concatenate the RNNLMs for all sentences in order. Concatenation changes the way of computing the first hidden state in each utterance (except the first utterance in the conversation). Considering the two turns $\boldsymbol{x}_{t-1}$ and $\boldsymbol{x}_t$, after concatenation, the computation of the first hidden state in turn $\boldsymbol{x}_t$, $\boldsymbol{h}_{t,1}$, is

$$\boldsymbol{h}_{t,1} = \boldsymbol{f}_\theta\left(x_{t,1}, \boldsymbol{h}_{t-1,N_{t-1}}\right). \tag{4}$$

As we will see from section 4, this simple solution can capture the long-term contextual information.

We introduce the role-based information by defining a role-dependent function $g_{\tau,r}(\cdot)$. For example, the probability of $x_{t,i}$ given $x_{t,i-1}, \ldots, x_{1,1} \triangleq \boldsymbol{x}_{\leq t,\leq i-1}$ and its role r_t is

$$p(x_{t,i}|\boldsymbol{x}_{\leq t,\leq i-1}, r_t) \propto g_{\tau,r_t}(\boldsymbol{h}_{t,i}). \tag{5}$$

where the $g_{\tau,r_t}(\cdot)$ is also parameterized by role r_t. In our implementation, we use

$$g_{\tau,r_t}(\boldsymbol{h}_{t,i}) = \text{softmax}(\mathbf{W}_\tau(\mathbf{W}_{r_t}\boldsymbol{h}_{t,i})), \tag{6}$$

where $\mathbf{W}_\tau \in \mathbb{R}^{V\times H}$, $\mathbf{W}_{r_t} \in \mathbb{R}^{H\times H}$, V, H are the vocabulary size and hidden layer dimension respectively. Even $\mathbf{W}_\tau$ is shared across the entire conversation model, $\mathbf{W}_{r_t}$ is role-specific. This linear transformation defined in Eq. 6 is easy to train in practice and appears to capture role information. This model is named the R-CONV model, as the role-based information is introduced in the output layer.

Despite the difference between the two models, they can be learned in the same way, which is similar to training a RNNLM [22]. Following the way of training a language model, the parameters could be learned by maximizing the following objective function

$$\sum_k \sum_t \ell(x_{t,k+1}, \boldsymbol{y}_{t,k}) \tag{7}$$

where $\boldsymbol{y}_{t,k}$ is the prediction of $x_{t,i+1}$. $\ell(\cdot,\cdot)$ can be any loss function for classification task. We choose cross entropy [24] as the loss function $\ell(\cdot,\cdot)$, because it is a popular objective function used in training neural language models.

As a final comment, if we eliminate the role information, R-LDA-CONV will be reduced to an RNNLM. To demonstrate the utility of role-based information, we will use an RNNLM over conversations as a baseline model.

3.3. Incorporating global topic context

In order to capture long-span context of the conversation, inspired by [16], we explicitly include a topic vector representing all previous dialog turns. We use Latent Dirichlet Allocation (LDA) to achieve a compact vector-space representation. This procedure maps a bag-of-words representation of a document into a low-dimensional vector which is conventionally interpreted as a topic representation. For each turn $\boldsymbol{x}_t$, we compute the LDA representation for all previous turns

$$\boldsymbol{s}_t = \boldsymbol{f}_{LDA}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{t-1}) \tag{8}$$

where $\boldsymbol{f}_{LDA}(\cdot)$ is the LDA inference function as in [15]. Then $\boldsymbol{s}_t$ is concatenated with hidden layer $\boldsymbol{h}_{t,i}$ to predict $\boldsymbol{x}_{t,i}$.

$$p(x_{t,i}|\boldsymbol{x}_{\leq t,\leq i-1}) \propto g_\tau\left([\boldsymbol{h}_{t,i}^\top \quad \boldsymbol{s}_t^\top]^\top\right). \tag{9}$$

This model is named LDA-CONV. We assume that by including $\boldsymbol{s}_t$ into output layer, the predicted word would be more topically related with the previous turns, thus allowing the recurrent part to learn more local context information.

[1] In our formulation, we use one turn as the minimal unit as multiple sentences in one turn share the same role.

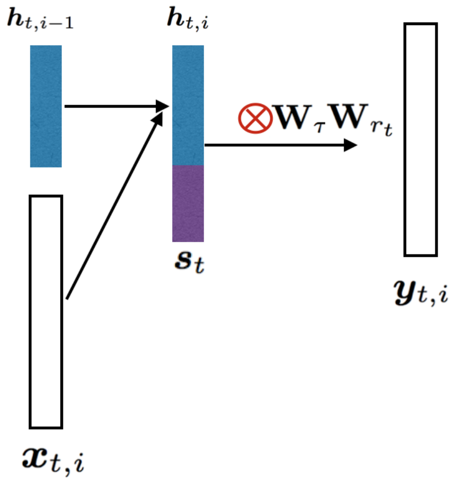

Figure 1: The R-LDA-CONV model. The turn-level LDA feature $\boldsymbol{s}_t$ is concatenated with word-level hidden layer $\boldsymbol{h}_{t,i}$ and the output weight matrix $\mathbf{W}_{r_t}$ is role specific.

When incorporating both the global topic vector and the role factor, the conditional probability of $\boldsymbol{x}_{t,i}$ is

$$p(x_{t,i}|\boldsymbol{x}_{\leq t,\leq i-1}, r_t) \propto g_{\tau,r_t}\left([\boldsymbol{h}_{t,i}^\top \quad \boldsymbol{s}_t^\top]^\top\right). \tag{10}$$

We call this model, illustrated in Figure 1, R-LDA-CONV.

4. Experiments

We evaluate our model from different aspects on the Ubuntu Dialogue Corpus [7], which provides one million two-person conversations extracted from Ubuntu chat logs. The conversations are about getting technical support for various Ubuntu-related problems. In this corpus, each conversation contains two users with different roles, POSTER: the user in this conversation who initializes the conversation by proposing a technical problem; RESPONDER: the other user who tries to provide technical support. For a conversation, we replace the user of each turn with the corresponding role.

4.1. Experimental setup

Our models are trained in a subset of the Ubuntu Dialogue Corpus, in which each conversation contains 6 - 20 turns. The resulting data contains 216K conversations in the training set, 10k conversations in the test set and 13k conversations in the development set. We use a Twitter tokenizer [25] to parse all utterances in the conversations. The vocabulary is constructed on the training set with filtering out low-frequency tokens and replacing them with "*UNKNOWN*". The vocabulary size is fixed to include 20K most frequent words. We did not filter out emoticons, instead we treat them as single tokens.

The LDA model is trained using all conversations in training data, where each conversation is treated as an individual training instance. We use *Gensim* [26] for both training and inference. There are three hyper-parameters in our models: the dimension of word representation K, the hidden dimension H and the number of topics M in LDA model. We use grid search over $K, H \in \{16, 32, 64, 128, 256\}$, $M \in \{50, 100\}$, and select the best combination for each model using the perplexity on the development set. We use stochastic gradient descent with the initial learning rate $\lambda = 0.1$ to train all the models.

Model	K	H	M	Perplexity
Baseline	32	128	-	54.93
R-CONV	256	128	-	48.89
LDA-CONV	256	128	100	51.13
R-LDA-CONV	256	128	50	**46.75**

Table 2: The best perplexity numbers of the three models on the development set.

Metric	Baseline	R-CONV	LDA-CONV	R-LDA-CONV
Recall@1	0.12	0.15	0.13	**0.16**
Recall@2	0.22	0.25	0.24	**0.26**

Table 3: The performance of response ranking with Recall@K.

4.2. Evaluation Metrics

Evaluation on response generation is an emerging research field in data-driven conversation modeling. Due to the variety of possible responses for a given context, it is too conservative to only compare the generated response with the ground truth. For a reasonable evaluation, the n-gram based evaluation metrics including BLEU [27] and ΔBLEU [28] require multiple references for one given context. One the other hand, there are indirect evaluation methods, for example, ranking based evaluation [7, 10] or qualitative analysis [29]. In this paper, we use both ranking-based evaluation (Recall@K [30]) across all models, and leave the n-gram based evaluation for future work. To compute the Recall@K metric of one model given K, the model is used to select the top-K candidates, and it is counted as correct if the ground-truth response is included. In addition to Recall@K, we also evaluate the different models based on test set perplexity.

To understand the chat conversations requires intensive knowledge of Ubuntu even for human readers. Therefore, the qualitative analysis focuses mainly on the capacity of capturing role information, not the justification of responses as valid answers to the technical questions.

4.3. Quantative Evaluation

Experiments in this section compare the performance of LDA-CONV, R-CONV and R-LDA-CONV to the baseline LSTM system.

4.3.1. Perplexity

The best perplexity numbers from the three models are shown in Table 2. R-LDA-CONV gives the lowest perplexity among the four models, nearly 8 points improvement over the baseline model. Comparing role vs. global topic, role has a bigger improvement on perplexity of 11% reduction for role vs. 7% for LDA topic. Combining both leads to a 15% reduction in perplexity. To simplify the comparison, in the following experiments, we only use the best configuration for each model.

4.3.2. Response ranking

The task is to rank the ground-truth response with some randomly-selected sentences for a given context. For each test sample, we use the previous $t-1$ sentences as context, trying to select the best tth sentence. We randomly select 9 turns from other conversations in the dataset, replacing their role with the ground truth label. As we noticed that sentences from the background channel, like "*yes*", "*thank you*", could fit almost all the conversations with various context. To distinguish the background channel from some contentful sentences, we sample the negative examples with the ground-truth sentence length as a constraint — samples with the similar length (± 2 words) are selected as negative examples.

The Recall@K are shown in Table 3. Both R-CONV and LDA-CONV are better than baseline result, while R-LDA-CONV gives the best performance overall. Both role factors and topic feature are acting positively in ranking ground-truth responses. Even though no role information is explicitly used in the baseline model, the contextual information itself could be a useful hint to rank the ground-truth response higher. Therefore, the performance of the baseline model is still better than random guess. Again, role has a bigger effect than topic, and the combination gives the best results, but differences in Recall@K performance are small.

4.4. Qualitative Analysis

For qualitative analysis, the best R-LDA-CONV model is used to generate role-specific responses, and we examined a number of examples to determine whether the generated response fit into the expected speaker role. We include two examples in Table 4 and Table 5 due to the page limitations. For each case, we have responses generated for each of the possible roles: a further question for the POSTER and a potential solution for the RESPONDER.

As we can see from the context part of Table 4, different roles clearly have different behaviors during the conversation. Ignoring the validity of this potential solution, this generated response is consistent with our expectation of the RESPONDER role. The response of POSTER seems quite plausible. The reply of RESPONDER is clearly the right style but more domain information in the topic vector could lead to a more useful solution.

Table 5 shows another example to demonstrate the difference between the POSTER and RESPONDER roles. In this example, the response for the RESPONDER is not a potential solution but a question to the POSTER. Unlike the generated question for the POSTER role in the previous example, the purpose of RESPONDER's question is to ask some further details in order to provide a simpler solution. The POSTER's response also fits well in the local context as well as global topic of ubuntu installation, claiming the difficulty of implementing the POSTER's suggestion. At the same time, the generated responses also show the necessity of incorporating certain domain knowledge into a domain-specific conversation system, which will be explored in future work.

5. Summary

We propose an LSTM-based conversation model by incorporating role factor and topic feature to model different word distribution for different roles. We present three models: R-CONV, LDA-CONV and R-LDA-CONV, by incorporating role factors and topic features into output layer. We evaluate the model using both perplexity and response ranking. Both R-CONV and LDA-CONV outperform the baseline model on all tasks. The model R-LDA-CONV gives the best performance by combining the two components. In addition, the generation results demonstrate the topical coherence and differences in responses associated with different roles. Besides role and topic, our model structure can be generalized to include more supervised information. For future work, we would incorporate supervised domain knowledge into our model to improve the topic relevance of the response.

Role	Utterance
Context	
POSTER	hey people . i have a disk from someone who accidentally overwrote his ext3 partition with a ntfs partition ...... any tips on how to recover files from the erased ext3 partition ? the ntfs partion was created using " fast " formatting (i.e. only the partition table was erased)
RESPONDER	you can rebuild a partition table .. brb
POSTER	is that possible ? how ? and he also started writing to the ntfs partition , so at least some data will be lost , hopefully some is still recoverable
RESPONDER	at that point , he might as well write it off or pay a professional to do it
POSTER	ere4si : is there some guide somewhere on how to make a live cd using the minimal iso ? it doesn't even have a squashfs root filesystem
Generation	
POSTER	can you p me to that ?
RESPONDER	" sudo fdisk -l " then use the fstab of the new permissions . then use that combination of * to recover backup
Ground truth	
RESPONDER	make an iso on usb bootable

Table 4: Response generation example (providing solution)

Role	Utterance
Context	
POSTER	question : i am currently installing ubuntu . in the " prepare partitions " dialog box , should i check both ext3 ' /' and ext3 ' /home ' to be formatted ?
RESPONDER	did you have a previous install of ubuntu on there
POSTER	no
RESPONDER	err kazol_ not mount , write to it i mean
RESPONDER	then you can format them both
Generation	
POSTER	i know , i don't know how to do this . bad , even if i try an encrypted install of ubuntu ... this means roller failed to mount it , so not the default .
RESPONDER	or something similar . are you trying to eject net crapped on there ?
Ground truth	
RESPONDER	ok thanks

Table 5: Response generation example (clarification of the problem)

6. References

[1] Regina Barzilay, Michael Collins, Julia Hirschberg, and Steve Whittaker. The rules behind roles: Identifying speaker role in radio broadcasts. In *AAAI/IAAI*, pages 679–684, 2000.

[2] B Hutchinson, B. Zhang, and M. Ostendorf. Unsupervised broadcast conversation speaker role labeling. In *ICASSP*, pages 5322–5325, 2010.

[3] B. Hutchinson B. Zhang, M. A. Marin and M. Ostendorf. Learning phrase patterns for text classification. *IEEE Trans. Audio, Speech and Language Processing*, 21(6):1180–1189, 2013.

[4] Alex Marin, Bin Zhang, and Mari Ostendorf. Detecting forum authority claims in online discussions. In *Proceedings of the Workshop on Languages in Social Media*, pages 39–47. Association for Computational Linguistics, 2011.

[5] Janet M Fuller. The influence of speaker roles on discourse marker use. *Journal of Pragmatics*, 35(1):23–45, 2003.

[6] Hao Wang, Zhengdong Lu, Hang Li, and Enhong Chen. A Dataset for Research on Short-Text Conversations. In *EMNLP*, pages 935–945, 2013.

[7] Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. The Ubuntu Dialogue Corpus: A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems. *arXiv preprint arXiv:1506.08909*, 2015.

[8] Alan Ritter, Colin Cherry, and William B Dolan. Data-driven response generation in social media. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 583–593. Association for Computational Linguistics, 2011.

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

[10] Lifeng Shang, Zhengdong Lu, and Hang Li. Neural Responding Machine for Short-Text Conversation. *arXiv preprint arXiv:1503.02364*, 2015.

[11] Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Pei-Hao Su, David Vandyke, and Steve Young. Semantically Conditioned LSTM-based Natural Language Generation for Spoken Dialogue Systems. *arXiv preprint arXiv:1508.01745*, 2015.

[12] Yangfeng Ji, Trevor Cohn, Lingpeng Kong, Chris Dyer, and Jacob Eisenstein. Document context language models. *arXiv preprint arXiv:1511.03962*, 2015.

[13] Rui Lin, Shujie Liu, Muyun Yang, Mu Li, Ming Zhou, and Sheng Li. Hierarchical recurrent neural network for document modeling. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 899–907, 2015.

[14] Yi Luan, Shinji Watanabe, and Bret Harsham. Efficient learning for spoken language understanding tasks with word embedding based pre-training. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[15] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[16] Tomas Mikolov and Geoffrey Zweig. Context dependent recurrent neural network language model. In *SLT*, pages 234–239, 2012.

[17] G. Ji and J. Bilmes. Multi-speaker language modeling. In *HLT NAACL*, 2004.

[18] Brian Hutchinson. *Rank and sparsity in language processing*. PhD thesis, 2013.

[19] Brian Hutchinson, Mari Ostendorf, and Maryam Fazel. A Sparse Plus Low-Rank Exponential Language Model for Limited Resource Scenarios. *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, 23(3):494–504, 2015.

[20] Songfang Huang and Steve Renals. Modeling topic and role information in meetings using the hierarchical Dirichlet process. In *Machine Learning for Multimodal Interaction*, pages 214–225. Springer, 2008.

[21] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A persona-based neural conversation model. *arXiv preprint arXiv:1603.06155*, 2016.

[22] Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. In *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010*, pages 1045–1048, 2010.

[23] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[24] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[25] Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A Smith. Improved part-of-speech tagging for online conversational text with word clusters. Association for Computational Linguistics, 2013.

[26] Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA. `http://is.muni.cz/publication/884893/en`.

[27] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

[28] Michel Galley, Chris Brockett, Alessandro Sordoni, Yangfeng Ji, Michael Auli, Chris Quirk, Margaret Mitchell, Jianfeng Gao, and Bill Dolan. deltaBLEU: A Discriminative Metric for Generation Tasks with Intrinsically Diverse Targets. *arXiv preprint arXiv:1506.06863*, 2015.

[29] Oriol Vinyals and Quoc Le. A Neural Conversational Model. *arXiv preprint arXiv:1506.05869*, 2015.

[30] Christopher D Manning, Prabhakar Raghavan, Hinrich Schütze, et al. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge, 2008.