Sentiment Forecasting in Dialog

Zhongqing Wang[†], Xiujun Zhu[†], Yue Zhang[‡], Shoushan Li[†], and Guodong Zhou[†]
[†] Natural Language Processing Lab, Soochow University, Suzhou, China
[‡] Westlake University, Hangzhou, China
wangzq@suda.edu.cn, 1300821865@qq.com
yue.zhang@wias.org.cn, {lishoushan, gdzhou}@suda.edu.cn

Abstract

Sentiment *forecasting* in dialog aims to predict the polarity of next utterance to come, and can help speakers revise their utterances in sentimental utterances generation. However, the polarity of next utterance is normally hard to predict, due to the lack of content of next utterance (yet to come). In this study, we propose a Neural Sentiment Forecasting (NSF) model to address inherent challenges. In particular, we employ a *neural simulation model* to simulate the next utterance based on the context (previous utterances encountered). Moreover, we employ a *sequence influence model* to learn both pair-wise and seq-wise influence. Empirical studies illustrate the importance of proposed sentiment forecasting task, and justify the effectiveness of our NSF model over several strong baselines.

1 Introduction

Developing intelligent chatbots is of great appealing to both the industry and the academics. However it is challenging to build up such an intelligent chatbot which involves a series of high-level natural language processing techniques, such as sentiment analysis of utterances in dialog.

Previous studies on sentiment classification focus on determining polarity (positive or negative) in a single document (Pang and Lee, 2008; Amplayo et al., 2018). In comparison, only few studies focus on determining polarity of utterances in dialog (Herzig et al., 2016; Majumder et al., 2018). However, all of these studies focus on determining the polarity of existing utterances. It may be more important to predict the polarity of *next* utterance yet to come. Given the example in Figure 1, although B expresses a positive sentiment in second utterance, A still shows a negative sentiment in his response. In this case, if B know that A would be very upset after his first utterance, he may revise his utterance to let A feel more comfortable. Hence, predicting the polarity of the next utterance can help a speaker to improve their utterances, which is important in automatic dialogue, such as customer service.

To the above purpose, we propose a new task, calls *sentiment forecasting* in dialog, which aims to predict the polarity of next utterance yet to come. In this paper, we focus on tackling two inherent challenges, one is how to simulate next utterance, and the other is how to learn the influence from context towards next utterance. The motivations behind are that, since next utterance is yet to come, it would be helpful if we can simulate next utterance from the context. Moreover, the polarity of next utterance can be much influenced by the context, it is necessary to consider the influence of the context for next utterance.

In this paper, we propose a Neural Sentiment Forecasting (NSF) model to address above challenges. In particular, a *neural simulation model* is employed to simulate next utterance based on existed utterances. In addition, a *hierarchical influence model* is employed to learn the influence from existed utterances by considering both pair-wise and sequence-wise influence between existed utterance sequence and next utterance. Empirical studies illustrate the importance of proposed sentiment forecasting task in dialog, and also show the effectiveness of our proposed NSF model over several strong baselines.

Proceedings of the 28th International Conference on Computational Linguistics, pages 2448–2458
Barcelona, Spain (Online), December 8-13, 2020

A: John, I've asked you not to smoke in here! I don't want to see you smoking in my office again.

B: I'm sorry, Ms. Fairbanks. I won't let it happen again.

A: **Negative** (*That's what you said the last time! If you want to smoke, you'll have to use your break time and go outside!*)

Figure 1: Example of dialog for sentiment forecasting.

2 Related Work

Our task is related to document-level sentiment classification (Pang and Lee, 2008) for various neural network models have been used, including convolutional neural networks (Kim, 2014), recursive neural network (Socher et al., 2013) and recurrent neural network (Teng et al., 2016; Tai et al., 2015). More recently, researches focus on aspect level sentiment analysis (Tang et al., 2016; Tay et al., 2018; Huang and Carley, 2018) and user-based or product-based sentiment classification (Chen et al., 2016; Song et al., 2017; Amplayo et al., 2018).

Different from document-level sentiment classification, sentiment classification in dialog aims to detect polarity of utterances in dialog by considering the influence of the whole dialog. As a prime study, Ruusuvuori (2012) stated that sentiment plays a pivotal role in conversations. Zhang et al. (2011) studied the impact of sentimental text on the customer's perception of the service agent. On the basic, Herzig et al. (2016) used SVM to classify sentiment in customer support dialogs by integrating both text based turn and dialog features. Recently, Cerisara et al. (2018) proposed a multi-task hierarchical recurrent network to classify sentiment and dialog act jointly. Majumder et al. (2018) proposed a recurrent neural networks to track of the individual states throughout the dialog and employed this information for sentiment classification.

Different from previous studies which focus on detecting the polarity of existing utterances, we propose a novel and important task to forecast polarity of next utterance yet to come.

3 Neural Sentiment Forecasting Modeling

As illustrated in Figure 1, given a existed utterance sequence $\{u_1, u_2, ..., u_{n-1}\}$ in a dialog d, we aim to predict the polarity (positive, negative, or neutral) of next utterance u_n. Note that, next utterance u_n does not exist in dialog, and we do not know the polarity of utterances in existed dialog sequence.

Figure 2 shows the overview of proposed Neural Sentiment Forecasting (NSF) model. We first learn the representation of each utterance u_i. Secondly, we employ a *neural simulation model* to simulate the representation of next utterance $\hat{u}_n$ based on the existed utterance representations. Thirdly, we employ a *hierarchical sequence influence model* to learn the influence from existed utterance sequence to the next utterance $\hat{u}_n$. Finally, we predict polarity $\hat{y}$ of next utterance $\hat{u}_n$ based on the simulation model and the influence from existed utterances. In the following, we discuss these issues one by one.

3.1 Existed Utterances Representation

Firstly, we need to learn the representation of existed utterances in dialog. Given a utterance u_i with m words $\{w_1, w_2, ..., w_m\}$, we transform each token w_i into a real-valued vector x_i using the word embedding vector of w_i (Mikolov et al., 2013). We employ LSTM model (Hochreiter and Schmidhuber, 1997) over u_i to generate a hidden vector sequence $(h_1, h_2, ..., h_m)$. At each step t, the hidden vector h_t of LSTM model is computed based on the current vector x_t and the previous vector h_{t-1}, and $h_t = \mathrm{LSTM}(x_t, h_{t-1})$. In particular, the initial state and all stand LSTM parameters are randomly initialized and tuned during training. In this way, we can use $H_i = h_m$ as the representation for utterance u_i in the context.

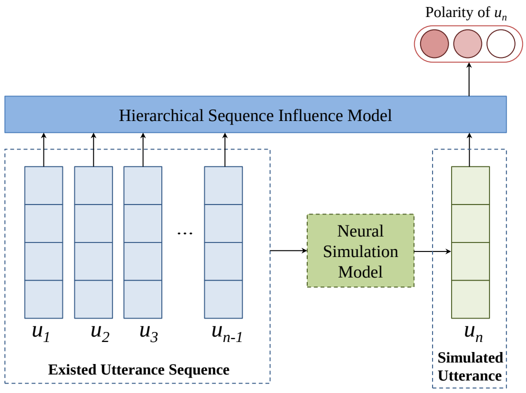

Figure 2: Overview of the neural sentiment forecasting model.

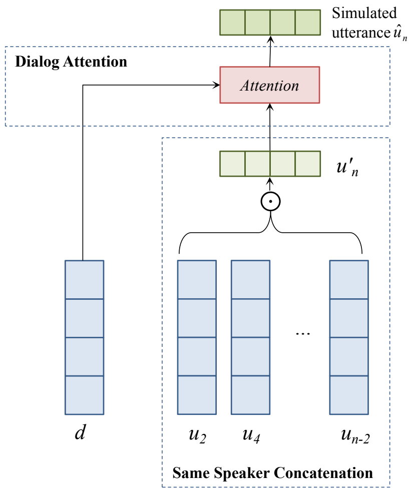

Figure 3: Architecture of neural utterance simulation model.

3.2 Neural Utterance Simulation Model

After we learn the representation $H = \{H_1, H_2, ..., H_{n-1}\}$ from existed utterance sequence, we employ a neural utterance simulation model to simulate the next utterance $\hat{u}_n$ from H, and the overview of proposed neural utterance simulation model is illustrated in Figure 3. In particular, since the polarities of utterances from the same speaker are correlated, *same speaker concatenation* model is used to concatenate the utterances from same speaker of u_n as a basic simulation. Moreover, since the polarity of u_n is influenced by its context in dialog, *dialog attention* model is employed to consider the influence from dialog sequence for simulating u_n.

Same Speaker Concatenation. To construct a basic simulation of u'_n, we concatenate the utterances from same speaker of u_n:

$$u'_n = H_2 \odot H_4 \odot ... \odot H_{n-2} \tag{1}$$

where $\{H_2, H_4, ..., H_{n-2}\}$ denotes the sequence which is from the same speaker of u_n, and u'_n is a basic simulation of u_n.

Dialog Attention. After getting the basic simulation u'_n, we then use dialog attention model to learn influence from utterance sequence to u'_n for simulating u_n.

First, we learn the dialog representation d by concatenating the utterance sequence $\{H_1, H_2, ..., H_{n-1}\}$:

$$d = H_1 \odot H_2 \odot ... \odot H_{n-1} \tag{2}$$

Then, we use attention mechanism to learn influence from dialog representation d to the basic simulation u'_n. The attention model outputs a continuous vector $\hat{u}_n \in R^{d\times 1}$ recurrently by feeding the hidden

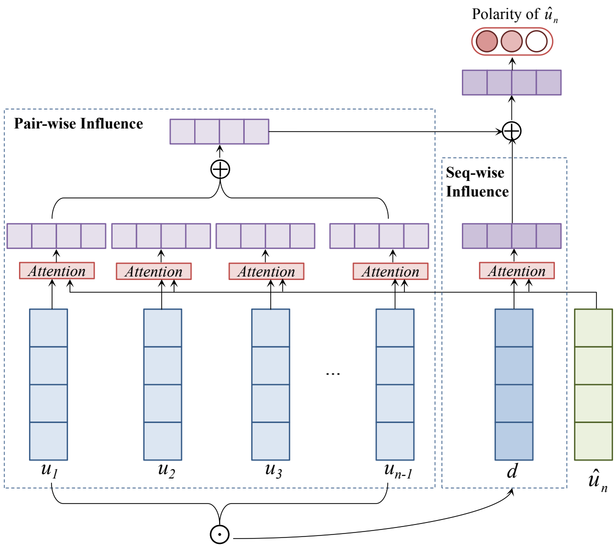

Figure 4: Architecture of hierarchical sequence influence model.

representation vectors $u'_n = \{h_{n1}, h_{n2}, ..., h_{nm}\}$ as inputs. Specifically, $\hat{u}_n$ is computed as a weighted sum of h_{nj} $(0 \leq j \leq m)$, namely

$$\hat{u}_n = \sum_{j}^{m} \alpha_j h_{nj} \tag{3}$$

where $\alpha_j \in [0, 1]$ is the weight of h_{nj}, and $\sum_j \alpha_j = 1$. For each piece of hidden state $h_{dj} \in d$ from the hidden representation of the dialog representation, the scoring function is calculated as follows:

$$v_j = \tanh(W h_{dj} + b) \tag{4}$$

$$\alpha_j = \frac{\exp(v_j)}{\sum_k \exp(v_k)} \tag{5}$$

In this way, the vector $\hat{u}_n$ is learned as the simulation of u_n from the existed utterance sequence H.

3.3 Hierarchical Sequence Influence Model

After we simulate the next utterance $\hat{u}_n$ from existed utterance sequence, we employ a hierarchical sequence influence model to learn influence from existed utterance sequence d to the simulated next utterance $\hat{u}_n$. The overview of hierarchical sequence influence model is shown in Figure 4.

In the hierarchical sequence influence model, we consider both pair-wise and sequence-wise influence model to learn the influence from existed utterance sequence to the simulated next utterance. The *pair-wise* influence model is used to learn the influence from each utterance to the next utterance, and the *sequence-wise* influence model is used to learn influence from utterance sequence to the next utterance. Finally, we integrate the representations from pair-wise and sequence-wise influence model into a unified representation to learn influence from whole utterance sequence collectively.

3.3.1 Pair-wise Influence Model

Firstly, we employ attention mechanism to learn the pair-wise influence from utterance u_i to the simulated next utterance $\hat{u}_n$. The pair-wise attention model outputs a continuous vector $v_i^p \in R^{d \times 1}$ recurrently by feeding the hidden representation vectors $H_n = \{h_{n1}, h_{n2}, ..., h_{nm}\}$ from $\hat{u}_n$ as inputs. Specifically, v_i^p is computed as a weighted sum of h_{nj} $(0 \leq j \leq m)$, namely

$$v_i^p = \sum_{j}^{m} \beta_j h_{nj} \tag{6}$$

where m is the hidden variable size, $\beta_j \in [0, 1]$ is the weight of h_{nj}, and $\sum_j \beta_j = 1$. For each piece of hidden state $h_{ij} \in H_i$ from the hidden representation of u_i, the scoring function is calculated as follows:

$$v_j = \tanh(Wh_{ij} + b) \tag{7}$$

$$\beta_j = \frac{\exp(v_j)}{\sum_k \exp(v_k)} \tag{8}$$

Here, the vector v_i^p is used as the representation of influence from u_i to $\hat{u}_n$.

3.3.2 Sequence-wise Influence Model

After we learn the pair-wise influence from each utterance to the simulated next utterance, we propose a sequence-wise influence model to learn the influence from whole utterance sequence to next utterance using attention mechanism. The sequence-wise attention model outputs a continuous vector $v^s \in R^{d\times 1}$ recurrently by feeding the hidden representation vectors $H_n = \{h_{n1}, h_{n2}, ..., h_{nm}\}$ from $\hat{u}_n$ as inputs. Specifically, v^s is computed as a weighted sum of h_{nj} ($0 \leq j \leq m$), namely

$$v^s = \sum_j^m \gamma_j h_{nj} \tag{9}$$

where $\gamma_j \in [0, 1]$ is the weight of h_{nj}, and $\sum_j \gamma_j = 1$. For each piece of hidden state h_j from the hidden representation of the dialog representation d (Eq. 2), the scoring function is calculated as follows:

$$v_j = \tanh(Wh_j + b) \tag{10}$$

$$\gamma_j = \frac{\exp(v_j)}{\sum_k \exp(v_k)} \tag{11}$$

Here, the vector v^s is used as the representation of the sequence-wise influence from the whole utterance sequence d to $\hat{u}_n$.

3.3.3 Integrating Pair-wise and Sequence-wise Influence

After we learn the representation $\{v_1^p, v_2^p, ..., v_{n-1}^p\}$ from pair-wise influence model, and the representation v^s from the sequence-wise influence model, we should integrate them into an uniform representation for sentiment forecasting of next utterance.

We first concatenate the representation $\{v_1^p, v_2^p, ..., v_{n-1}^p\}$ from pair-wise influence model into an uniform pair-wise representation v^p:

$$v^p = v_1^p \oplus v_1^p \oplus ... \oplus v_{n-1}^p \tag{12}$$

We then concatenate pair-wise representation v^p with sequence-wise representation v^s into an uniform representation v:

$$v = v^p \oplus v^s \tag{13}$$

Here, we use v as the representation of simulated next utterance $\hat{u}_n$ by considering both pair-wise and sequence-wise influence for forecasting polarity of u_n.

3.4 Sentiment Forecasting of Next Utterance

After we learn the representation v of simulated next utterance $\hat{u}_n$ with both pair-wise and sequence-wise influence, we employ a multi-layer perceptron model to learn the polarity (positive, negative, or neutral) of it. Since there are three sentiment categories, our task can be considered as a multi-label classification task. Formally, giving an input vector v, a hidden layer is used to induce a set of high-level features as follow:

$$H_P = \sigma(W_p^h v + b_p^h), \tag{14}$$

H_P is used as inputs to a softmax output layer:

$$P_P = \mathrm{softmax}(W_p H_P + B_P) \tag{15}$$

Here, W_p^h, b_p^h, W_p, and B_p are model parameters.

3.5 Model Training

Given the utterance sequence $\{u_1, u_2, ..., u_{n-1}\}$ in a dialog d_i and the pre-defined polarity y_i of next utterance $\hat{u}_n$, our training objective is to minimize the cross-entropy loss over a set of training examples $(d_i, y_i)|_{i=1}^{N}$, with a ℓ_2-regularization term,

$$J(\theta_y) = -\sum_{i=1}^{N}\sum_{j=1}^{K} y_i \log \hat{y}_i + \frac{\lambda}{2}||\theta_y||^2 \quad (16)$$

where $\hat{y}_i$ is the predicted label, θ_y is the set of model parameters and λ is a parameter for L2 regularization.

4 Experimentation

4.1 Data and Setting

In all the experiments, the DailyDialog (Li et al., 2017) dataset is used to study the importance of sentiment forecasting task, and evaluate the performance of proposed NSF model. The dataset contains 13,118 multi-turn dialogs, the speaker turns are roughly 8, and the average tokens per utterance is about 15.

Since there are six kinds of emotion[1] in the original dataset, and some emotions occupy less than 5%, we thus merge all the emotions into three sentiment categories: positive(joy), negative(other emotions), and neutral (no emotion).

To construct a sentiment rich dataset, we only select the dialogs which contain at least one emotional utterance, we then get 7,395 dialogs. We randomly separate the dataset into training/test sets with 4,435/2,960 dialogs. For each dialog, we select top-4 utterances for our experiments: the top-3 utterances are considered as existed utterances ($\{u_1, u_2, u_3\}$), the last utterance is considered as unknown utterance (u_n). Note that, we do not know the content of u_n, and we should forecast the sentiment of it in experiments.

The vocabulary size is 9,888, the embedding size sets to 64, and the hidden size of all the model sets to 32. Here, all the model parameters are optimized by AdaGrad (Duchi et al., 2011).

F1-measure (F1.) are used to evaluate the performance of proposed model in each sentiment category (positive, and negative), and micro-average F1-measure is used to evaluate the overall performance.

4.2 Experimental Results

In this subsection, we present experiment results to illustrate the importance of sentiment forecasting task, and show the effectiveness of proposed NSF model.

4.2.1 Comparison with Baselines

We first shows the results of proposed Neural Sentiment Forecasting (NSF) model with several strong baselines, where

- LSTM^i is a single utterance based sentiment forecasting model, it uses LSTM model to learn the representation of u_i ($1 \leq i \leq 3$) (Section 3.1), and then employs the representation of u_i to forecasting sentiment of next utterance u_n.

- LSTM^{seq} is a sequence based sentiment forecasting model, it employs a LSTM model to learn dialog representation d from the existed utterance sequence $\{u_1, u_2, u_3\}$ (Eq. 2), and then employ the dialog representation d to forecast sentiment of u_n.

- *ICON* takes one utterance with previous k utterances as input, and uses a GRU model for modeling inter-personal dependency in previous utterances and stores all history with one memory network (Hazarika et al., 2018).

[1]There exist six categories of emotion in the dataset: joy, anger, disgust, fear, sadness, and surprise. Besides, many utterances do not express any emotion (i.e., neutral).

Table 1: Comparison with baselines.

	Pos F1.	**Neg F1.**	**Avg F1.**
LSTM^1	0.545	0.285	0.415
LSTM^2	0.506	0.310	0.408
LSTM^3	0.558	0.273	0.415
LSTM^{seq}	0.563	0.342	0.453
ICON	0.529	0.293	0.411
DialogRNN	0.540	0.358	0.449
NSF	0.586	0.387	0.486

- *DialogRNN* employs recurrent neural networks to keep track of the individual states of utterances and uses this information for sentiment classification in dialog (Majumder et al., 2018). It report best results in dialog sentiment classification.

Note that, since ICON and DialogRNN in Table 1 were designed for dialog sentiment classification instead of sentiment forecasting, we use u_2 to simulate u_n for these two models[2].

From the results in Table 1, we can find that the performance of sequence based LSTM^{seq} is better than utterance based LSTM^i, it indicates that the importance of dialog sequence for forecasting sentiment, and it also suggest us to consider the influence of whole dialog sequence for sentiment forecasting of next utterance.

DialogRNN outperforms utterance based LSTM^i, it also indicates the importance of dialog sequence for sentiment classification.

The proposed NSF model outperforms all other baselines significantly, it shows that we should consider both neural simulation and influence of dialog structure for forecasting sentiment of next utterance.

4.2.2 Comparison with Different Simulation Models

We then analyze the effectiveness of various neural simulation model, where

- UniSim^i is a basic single utterance simulation model, which use u_i to simulate u_n, and forecast polarity of it.
- $\text{DualSim}^{i,j}$ is a dual utterances simulation model, which concatenates the representation of u_i and u_j to simulate u_n, and forecast polarity of it.
- $\text{SeqSim}^{d \to i}$ is a sequence based simulation model, it employs attention mechanism to learn the influence from dialog representation d to u_i for simulating u_n, and forecast polarity of it. It has been discussed in Section 3.2.

From the results in Table 2, we can find that: UniSim^i which only employs single utterance to simulate next utterance cannot achieve a well performance.

$\text{DualSim}^{i,j}$ which employs two utterances to simulate next utterance outperforms the single utterance based UniSim^i, it indicates effectiveness of context for simulating next utterance.

The sequence attention based $\text{SeqSim}^{d \to i}$ outperforms both single utterance and dual utterance based models, and $\text{SeqSim}^{d \to 2}$ outperforms all other simulation model. It indicates the effectiveness of dialog attention and the same speaker's utterance (u_2 and u_4 are from the same speaker). Hence, we use $\text{SeqSim}^{d \to 2}$ as the proposed neural simulation model in this studies.

4.2.3 Comparison with Different Influence Models

After we learn the simulated next utterance $\hat{u}_n$ from $\text{SeqSim}^{d \to 2}$ in dialog, we then analyze influence of dialog sequence for next utterance with different neural influence model, where

[2]Since u_2 and $u_n(u_4)$ are from the same speaker, there is high probability that they have the same polarity. It has been proven in Table 5.

Table 2: Comparison with different simulation model.

	Pos F1.	**Neg F1.**	**Avg F1.**
UniSim^1	0.545	0.285	0.415
UniSim^2	0.506	0.310	0.408
UniSim^3	0.558	0.273	0.415
$\mathrm{DualSim}^{1,2}$	0.533	0.349	0.441
$\mathrm{DualSim}^{1,3}$	0.554	0.317	0.435
$\mathrm{DualSim}^{2,3}$	0.545	0.330	0.437
$\mathrm{SeqSim}^{d\to1}$	0.540	0.358	0.449
$\mathrm{SeqSim}^{d\to2}$	0.544	0.366	0.455
$\mathrm{SeqSim}^{d\to3}$	0.538	0.348	0.443

Table 3: Comparison with different influence model.

	Pos F1.	**Neg F1.**	**Avg F1.**
$\mathrm{UniIf}^{1\to n}$	0.533	0.349	0.441
$\mathrm{UniIf}^{2\to n}$	0.554	0.317	0.435
$\mathrm{UniIf}^{3\to n}$	0.545	0.330	0.437
PairIf	0.540	0.358	0.449
SeqIf	0.544	0.366	0.455
NSF	0.586	0.387	0.486

- $\mathrm{UniIf}^{i\to n}$ employs attention mechanism to learn the influence from utterance u_i to next utterance u_n.
- *PairIf* learns pair-wise influence towards u_n, by concatenating the representations from $\mathrm{UniIf}^{i\to n}$, it has been discussed in Section 3.3.1.
- *SeqIf* learns sequence-wise influence from utterance sequence to u_n, it has been discussed in Section 3.3.2.

From the results in Table 3, we can find that $\mathrm{UniIf}^{i\to n}$ which only considers the influence from single utterance u_i cannot achieve a well performance.

The pair-wise influence model PairIf outperforms all the single utterance influence model, it indicates the importance of pair-wise influence model for sentiment forecasting. In addition, sequence-wise influence model SeqIf outperforms PairIf, it shows that sequence-wise influence model is much more important than pair-wise model.

Finally, the proposed NSF model outperforms all other influence models significantly, it shows that we should consider both pair-wise and sequence-wise influence for sentiment forecasting of next utterance.

5 Analysis and Discussion

In this section, we give some statistic and analysis to discuss our motivations and illustrate the importance of propose sentiment forecasting task.

5.1 Sentiment Correlation between Existed Utterances and Next Utterance

We analyze the sentiment correlation between next utterance u_4 and existed utterance u_i ($u_i \in \{u_1, u_2, u_3\}$) in dialog. Table 5 shows conditional probability $P(u_n|u_i)$: given the polarity of u_i, the conditional probability of u_n ($n = 4$) with the same polarity of u_i. From the table, we can find that polarity of all existed utterances are correlated with u_n. In addition, we find that average conditional probability of $P(u_4|u_2)$ is much higher than other utterances, it may due to that u_2 and u_4 are from the same speaker, the polarity of utterances from same speaker may not much change in most situations.

Table 4: Outputs of LSTM^{seq} and NSF.

Dialog	LSTM^{seq}	NSF
A: Didn't you event want to go to the cinema? B: Not really. I watched televison for an hour. A: What was on television last night? B: Positive (*Boxing. It was excellent.*)	Neutral	Positive
A: Can I help you? B: Have got the last ONXIU magazine A: Yes, but it's checked out. B: Negative (*What a pity! I missed it again.*)	Positive	Negative
A: Why are you so quiet? B: My girlfriend just broke up with me. A: You must feel terrible now. B: Negative (*Yeah.*)	Neutral	Negative

Table 5: Pair-wise correlations between next and existed utterance.

	u_4		
	Positive	Negative	Average
u_1	0.534	0.120	0.327
u_2	0.557	0.366	0.462
u_3	0.642	0.136	0.389

5.2 Case Study

We select three examples from the testing data to illustrate the effectiveness of proposed NSF model compared with the LSTM^{seq} model in Table 4.

In the first example, we can find that although next utterance is not related with the existed utterances, the proposed NSF model still predicts correct polarity. It may due to that NSF can simulate next utterance based on existed utterance sequence.

In the second example, the polarity of next utterance is related with third utterance in second example, the proposed NSF predicts correct polarity by considering pair-wise influence between third utterance and next utterance. Meanwhile, by considering sequence-wise influence from existed utterances, the proposed NSF predicts correct polarity in the third examples.

In summary, NSF is much more effective by considering both neural simulation and influence of dialog structure for forecasting sentiment of next utterance.

6 Conclusion

In this paper, we propose a novel but important task, called sentiment forecasting in dialog, which aims to forecast the polarity of next utterance to come. There are two challenges in this task, one is that how to simulate the next utterance for predicting its' polarity, and another is how to learn the influence of existing utterance sequence for forecasting next utterance's polarity. In this paper, we propose a Neural Sentiment Forecasting (NSF) model to address above challenges. In particular, a neural simulation model is used to simulate the next utterance based on existed utterances sequence. In addition, a hierarchical influence model is used to learn the influence of existing utterance by considering both pair-wise and sequence-wise influence. Empirical studies illustrate the importance of our proposed sentiment forecasting task, and show the effectiveness of our proposed NSF model over several strong baselines.

Acknowledgments

We thank our anonymous reviewers for prudent advice. This work is supported by the National Natural Science Foundation of China (No. 61806137, No. 61702149).

References

Reinald Kim Amplayo, Jihyeok Kim, Sua Sung, and Seung-won Hwang. 2018. Cold-start aware user and product attention for sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2535–2544.

Christophe Cerisara, Somayeh Jafaritazehjani, Adedayo Oluokun, and Hoa T. Le. 2018. Multi-task dialog act and sentiment recognition on mastodon. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 745–754.

Huimin Chen, Maosong Sun, Cunchao Tu, Yankai Lin, and Zhiyuan Liu. 2016. Neural sentiment classification with user and product attention. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1650–1659.

John C. Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159.

Devamanyu Hazarika, Soujanya Poria, Rada Mihalcea, Erik Cambria, and Roger Zimmermann. 2018. Icon: Interactive conversational memory network for multimodal emotion detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2594–2604. Association for Computational Linguistics.

Jonathan Herzig, Guy Feigenblat, Michal Shmueli-Scheuer, David Konopnicki, Anat Rafaeli, Daniel Altman, and David Spivak. 2016. Classifying emotions in customer support dialogues in social media. In *Proceedings of the SIGDIAL 2016 Conference, The 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 13-15 September 2016, Los Angeles, CA, USA*, pages 64–73.

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

Binxuan Huang and Kathleen M. Carley. 2018. Parameterized convolutional neural networks for aspect level sentiment classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1091–1096.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1746–1751.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. Dailydialog: A manually labelled multi-turn dialogue dataset. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017, Taipei, Taiwan, November 27 - December 1, 2017 - Volume 1: Long Papers*, pages 986–995.

Navonil Majumder, Soujanya Poria, Devamanyu Hazarika, Rada Mihalcea, Alexander F. Gelbukh, and Erik Cambria. 2018. Dialoguernn: An attentive RNN for emotion detection in conversations. *CoRR*, abs/1811.00405.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 3111–3119.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.

Johanna Ruusuvuori. 2012. Emotion, affect and conversation. *The handbook of conversation analysis*, pages 330–349.

Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL.*

Kaisong Song, Wei Gao, Shi Feng, Daling Wang, Kam-Fai Wong, and Chengqi Zhang. 2017. Recommendation vs sentiment analysis: A text-driven latent factor model for rating prediction with cold-start awareness. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 2744–2750.

Kai Sheng Tai, Richard Socher, and Christopher D. Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1556–1566.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016. Effective lstms for target-dependent sentiment classification. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 3298–3307.

Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5956–5963.

Zhiyang Teng, Duy-Tin Vo, and Yue Zhang. 2016. Context-sensitive lexicon features for neural sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1629–1638.

Lu Zhang, Lee B Erickson, and Heidi C Webb. 2011. Effects of emotional text on online customer service chat.