PoliSe: Reinforcing Politeness using User Sentiment for Customer Care Response Generation

Mauajama Firdaus[1 *†], **Asif Ekbal**[2] **and Pushpak Bhattacharyya**[3]
[1] Department of Computing Science, University of Alberta, Canada
[2]Department of Computer Science and Engineering, Indian Institute of Technology Patna, India
[3]Department of Computer Science and Engineering, Indian Institute of Technology Bombay, India
[1]mauzama.03@gmail.com; [2]asif@iitp.ac.in; [3]pushpakbh@gmail.com

Abstract

The interaction between a consumer and the customer service representative greatly contributes to the overall customer experience. Therefore, to ensure customers' comfort and retention, it is important that customer service agents and chatbots connect with users on social, cordial and empathetic planes. In the current work, we automatically identify the sentiment of the user and transform the neutral responses into polite responses conforming to the sentiment and the conversational history. Our technique is basically a reinforced multi-task network- the primary task being 'polite response generation' and the secondary task being 'sentiment analysis'- that uses a Transformer based encoder-decoder. We use sentiment annotated conversations from Twitter as the training data. The detailed evaluation shows that our proposed approach attains superior performance compared to the baseline models.

1 Introduction

Human-machine interactions have increased rapidly assisting humans in their everyday lives. With the growth in Artificial Intelligence (AI) and Natural Language Processing (NLP), chatbots and personal assistants, such as Microsoft's Cortana, Apple's Siri, etc., have predominantly become a part of our daily lives. Thus, research in recent years has been on modulating biases, styles, and control in text generation to enhance these interactions.

Customer care is an essential tool used by companies to provide guidance, and assistance and in building stable customer relations. The ease of access, ease of following up, and immediacy of social media has made it a strong platform for companies and applications to interact with their customers. In this regard, conversational agents such as shopping agents, customer service agents, and personal assistants have become extremely famous for handling the various queries of a customer and providing suitable responses to them.

Assisting the customer through social media channels is attaining high popularity. The main reason behind this is the fact that the important elements of social media are its immediacy, transparency, ease of following up, and providing a human-like feel to the company/brand. In this platform, we see the usage of polite and emphatic language, which is the center of our current study.

For the growth of any company or application, customer care agents must be cordial and amicable to the customer. Thus along with handling queries, the agents need to provide customer satisfaction by greeting, empathizing, appreciating feedback, apologizing at the right time, and thus building a strong relationship with the customer.

Recently, incorporating politeness in responses has been investigated (Golchha et al., 2019; Madaan et al., 2020; Niu and Bansal, 2018) to make the dialogue agent more human-like and interactive. Also, the sentiments of the user are important for properly addressing the customer needs (Shi and Yu, 2018; Firdaus et al., 2022b, 2021) to assist in creating smooth and cordial conversations. Therefore, for politely responding to the user it is important to have knowledge about user sentiment to avoid frustrating experiences and help build better systems.

The usage of the user feedback in the form of sentiments is crucial to get contextually correct polite responses as presented in Table 1. For example, if the user has a negative sentiment towards the customer care system, then the possible polite response should be towards apology, assurance, and empathy rather than greet or appreciation. For the first example, the sentiment of the user is *negative*,

* The work was done while the author was in IIT Patna

Dialog Context	User Sentiment	Generic Response	Polite Response	Polite Behaviour
Hey, i got food poisoning from your inflight meal on sunday	Negative	Send us a dm	That's disappointing to hear, we are sorry please send us a dm.	Apology
I need the software update urgently, the battery lasts literally half a day	Negative	How can we help?	Don't worry, we are here for you, please say how can we help?	Assurance
Dear this new update is awesome, got great new apps!	Positive	The update has many features.	Thank you very much, please checkout the exciting features in the update.	Appreciation
Order 2 zinger box meals n got free popcorn chicken, yayyyy	Positive	Enjoy your meal.	That's nice to hear, enjoy your meal.	Acknowledge
How do i go about getting a monthly ride pass ?	Neutral	We have send the link	Hello, good morning we have send the link.	Greet

Table 1: Examples of polite responses in accordance to the user sentiments

therefore politeness in the form of appreciation, greeting or acknowledgment could lead to a wrong response making the customer angrier towards the customer care agent.

Similarly, if the user has a positive sentiment towards the customer care application, then the customer care agent should be able to converse politely showing appreciation and greeting. Previously, researchers focused on changing the sentiment of a given sentence (Li et al., 2018) using style transfer techniques. Here, we use sentiment as feedback for generating polite responses. Therefore, we propose the task of sentiment-guided polite response generation. To the best of our knowledge, this is one of the first works that jointly predict the sentiments from the user utterances and uses the predicted sentiment as feedback for generating polite responses.

Due to the unavailability of sentiment-annotated customer care conversations, we annotate the Courteously Yours Customer Care Dataset (CYCCD) (Golchha et al., 2019) with sentiment labels. To address the task of sentiment-guided polite response generation we design an end-to-end framework that identifies the sentiment of the user and uses the sentiment knowledge for generating polite responses.

We employ a hierarchical transformer network that captures the utterance as well as contextual information for generating responses. While we utilize a BERT-based classifier for predicting the sentiments. For incorporating politeness we use task-specific rewards that reinforce polite behavior in the generated customer care responses.
The key contributions of our current work are:

- We propose the task of generating polite responses in accordance with user sentiments.
- We develop a multi-task end-to-end hierarchical network to identify the sentiments and use the predicted information for generating the contextually correct polite responses.
- We design task-specific rewards that assist in the proposed task and ensure that the content is preserved while incorporating correct and interactive polite responses.

2 Related Work

Natural language generation (NLG) module has been gaining importance in several applications such as dialogue systems (Vinyals and Le, 2015; Shen et al., 2018; Wu et al., 2018; Serban et al., 2017a; Zhang et al., 2018; Li et al., 2016), question answering systems (Reddy et al., 2017; Duan et al., 2017), and many other natural language interfaces. To help the users achieve their desired goals, response generation provides the medium through which a conversational agent can communicate with its user.

In (Serban et al., 2017b), the authors have proposed a hierarchical encoder-decoder model for capturing the dependencies in the utterances of a dialogue. Emotion classification and analysis (Herzig et al., 2016) in customer support dialogue is important for a better understanding of the customer and to provide better customer support. Lately, several works have been carried out on controlled text generation (Hu et al., 2017; Li et al., 2017; Subramanian et al., 2017; Fedus et al., 2018; Peng et al., 2018) to generate responses with desired attributes.

Style transfer has been an emerging field in natural language processing (NLP). In (Rao and Tetreault, 2018), a dataset has been introduced for formality style transfer. Unsupervised text style transfer has encouraged in transforming a given text without parallel data (Shen et al., 2017; Carlson et al., 2017; Fu et al., 2018; Li et al., 2018; Niu and Bansal, 2018). One of the early works in politeness (Gupta et al., 2007) was based upon making the conversational agents more affective and socially intelligent by incorporating different politeness strategies in the responses.

Recently in (Niu and Bansal, 2018), the authors proposed a neural framework that could induce politeness in chit-chat conversations in the absence of parallel data. Lately, (Golchha et al., 2019) presented a method for increasing user satisfaction by inducing courteous phrases in the customer-care responses by exploiting reinforced pointer networks.

One of the recent studies presented in (Madaan et al., 2020) devised a tag and generate framework for converting non-polite sentences into polite ones. Recent works on politeness focuses on modeling politeness across languages (Firdaus et al., 2020), gender and age group (Firdaus et al., 2022a), predicting variations in politeness (Mishra et al., 2022b) and building a politeness adaptive system (Mishra et al., 2022a).

From the existing literature on politeness, we can conclude that politeness in conversational agents is essential for increasing the social and affective understanding of conversational agents. Our current work differs from the existing baselines as we include user feedback in the form of sentiment information to enhance the quality of generation and to make the responses contextually coherent with the dialog.

3 Methodology

In this section, we formally define the problem statement and give a detailed description of our proposed methodology.

3.1 Problem Formulation

In our current work, we address the task of identifying the sentiments from the user utterances and using the sentiment information to transform the generic customer care responses into polite responses which are contextually appropriate to the dialog history and the user sentiments.

Precisely, given the dialog history D having a sequence of sentences $(s_1, s_2, \ldots, s_N)$ where each sentence s_n is a sequence of words $u_1, u_2, \ldots, u_M$, the task is to simultaneously identify the sentiments from the user sentence $s_{(N-1)}$ and utilize the detected sentiment to transform the generic customer care s_N response into a polite response s'_N.

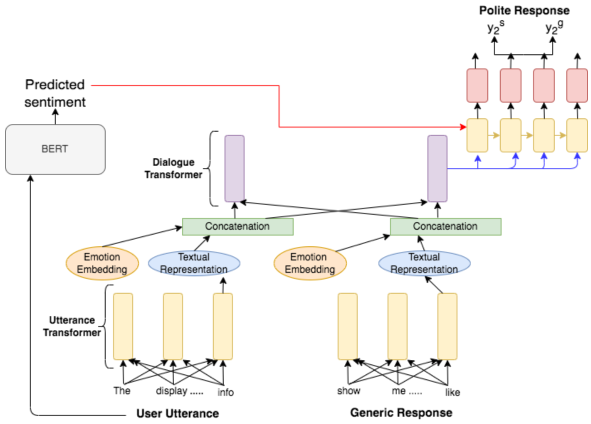

Figure 1: Architectural diagram of our proposed multi-task framework that simultaneously identifies the sentiment and generates the polite response in accordance with the predicted sentiment

3.2 Methodology

Our proposed network is based upon the Transformer Encoder-Decoder (TED) architecture (Vaswani et al., 2017) as shown in Figure 1. We utilize a hierarchical transformer having two encoders: one of which is used to encode the sentences named as sentence encoder while we use another Transformer to encode the output of the sentence encoder to capture the dialog context. We apply the softmax activation function on the sentence encoder to capture the sentiment information from the user utterance. The predicted sentiment information along with the contextual information is used to initialize the Transformer decoder.

We design task-specific rewards to ensure that the users' sentiments and politeness are induced appropriately in the generated responses. In our present work, we focus on the sentiment of the user and the way it affects the politeness quotient of the customer care agent/bot response. For this, we consider the emotional information of the user that, in turn, helps in identifying the correct sentiment of the user. As we predict the sentiment of the user, the customer care agent's emotional information is provided as shown in Figure 1 for generating consistent politeness, emotion, and sentiment-aware responses.

Sentence Encoder. As the transformer encoder has multiple layers and each layer is composed of a multi-head self attentive sub-layer followed by a feed-forward sub-layer with residual connections (He et al., 2016) and layer normalization (Ba et al., 2016), we use it to encode the sentences in a given dialog. For intricate details on the Transformer network, we refer the interested readers to (Vaswani et al., 2017). To learn the representation of s_n, $s_n = (u_{k,1}, u_{k,2}, ..., u_{k,n'})$ is first mapped into continuous space

$$T_u = (t_1^i, t_2^i, \dots, t_{|s_n|}^i); where[T_j^i = e(w_j^i) + p_j] \quad (1)$$

where $e(u_j^i)$ and p_j are the word and positional embedding of every word u_j^i in an utterance, respectively. For words, we use Glove embeddings and adopt the sine-cosine positional embedding (Vaswani et al., 2017) as it performs better and does not introduce additional trainable parameters.

The utterance encoder (a Transformer) converts T_u into a list of hidden representations $h_1^i, h_2^i, \dots, h_{|s_n|}^i$. We use the last hidden representation $h_{|s_n|}^i$ (i.e. the representation at the EOS token) as the textual representation of the utterance s_n. Similarly, to the representation of each word in s_n, we also take into account the utterance position. Therefore, the final textual representation of the utterance s_n is:

$$h_i^s = h_{|s_n|}^i + p_i \quad (2)$$

Note that the words and sentences share the same positional embedding matrix. We also capture the emotional embeddings of every sentences using the output distribution from DeepMoji (Felbo et al., 2017) which is pre-trained on the emoji prediction task in a similar manner as (Golchha et al., 2019).

The emotional embedding of every sentence is represented as $e_{s,n}$. The final representation of any sentence is given by the concatenation of the emotional representation as well as the last hidden representation of the sentence.

$$h_i^{sen} = h_i^s + e_{s,n} \quad (3)$$

Context Encoder. The context encoder is yet another Transformer, but it is applied on the utterance level. After running the transformer on the sequence of sentence representation $h_1^{sen}, {}_2^{sen}, \dots, h_{|D|}^{sen}$, we obtain the context-sensitive utterance representations $\hat{D} = (\hat{d}_1, \hat{d}_2, \dots, \hat{d}_{|D|})$.

After achieving the representation of a given dialogue using a hierarchical transformer network, we employ a transformer decoder to generate the polite response in accordance to the contextual information and the sentiment information.

Sentiment Classification Network. For identifying sentiments, we propose a sentiment classification network. The input to this network is the user utterance and the output is a predicted sentiment for this utterance. A BERT classifier is applied on the user utterance followed by a softmax output layer which gives the sentiment prediction. The sentiment classifier is trained by minimizing the negative log-likelihood

$$\mathcal{L}_{SE} = -\sum_{se=1}^{N} y_{se} \log \tilde{y_{se}} \quad (4)$$

Decoder: To generate the polite response with the predicted sentiment, we employ a Transformer decoder (Vaswani et al., 2017) as shown in Figure 1. The Transformer decoder used in our current work is slightly different from the original in which two multi-head attention layers were employed to incorporate both the contexts in encoder and decoder.

But in our case, we only need one to learn the decoder context, since the context in encoder is a vector (i.e., $\hat{d}_i$). We predict the polite responses $Y_k = (y_0^k, y_1^k, y_2^k, \dots, y_{|Y_k|}^k)$ one word per step (y_0^k) is an artificially added BOS (beginning of sentence token). At the j^{th} step, we predict y_j^k given $y_0^k, \dots, y_{j-1}^k$, predicted sentiment SE_i and the context representation $\hat{D}$. By applying word and positional embeddings to $(y_0^k, \dots, y_{j-1}^k)$, we obtain $\tilde{E}_{1:j-1}^k = (\tilde{e_0^k}, \dots, \tilde{e_{j-1}^k})$. Then, we apply multi-head attention sub-layer to $\tilde{E}_{1:j-1}^k$:

$$\begin{aligned} \tilde{h_{j-1}} = MultiHead(q_{j-1}, K_{j-1}, V_{j-1}); \\ q_{j-1} = W^Q \tilde{e_{j-1}^k}; \\ K_{j-1} = W^K \tilde{E}_{1:j-1}^k; \\ V_{j-1} = W^V \tilde{E}_{1:j-1}^k \end{aligned} \quad (5)$$

where q_{j-1}, K_{j-1}, V_{j-1} are the input query, key and value matrices of the multi-head attention function (Vaswani et al., 2017), respectively. Also, W^Q, W^K and W^V are the weight matrices. To attain polite responses in accordance to the sentiment, we include the information of $\hat{D}$ and sentiment SE_i by addition:

$$\tilde{x_{j-1}} = \tilde{h_{j-1}} + \hat{d}_k + SE_i \quad (6)$$

We also apply a feedforward sub-layer (one hidden layer with ReLU (Glorot et al., 2011) activation function) after $\tilde{x_{j-1}}$ in a similar manner as (**?**):

$$\tilde{g_{j-1}} = W_2^{ff} max(0, W_1^{ff} \tilde{x_{j-1}} + b_1) + b_2 \quad (7)$$

Note that the transformer decoder can have multiple layers by applying Equation (4) to (6) multiple times but we show only the computation of one layer for the simplicity. The probability of y_j^k given $y_0^k, \ldots, y_{j-1}^k$, SE_i and the context representation $\hat{D}$ is:

$$p(y_j^k|y_0^k, \ldots, y_{j-1}^k, \hat{d}_k, SE_i) = softmax(W^O g_{\tilde{j-1}}) \tag{8}$$

Model Training: As used in (Paulus et al., 2017), we jointly use reinforcement learning (RL) and machine learning (ML) to train our model in a similar manner as (Golchha et al., 2019). If $\tilde{y} = \{\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_{n'}\}$ is the gold output tokens for the given generic response tokens h_i^{sen} and conversation history $\hat{D}$, the maximum-likelihood objective using teacher forcing is given by:

$$L_{MLE} = -\sum_{t=1}^{n'} \log p(\tilde{y}_t|\tilde{y}_1, \ldots, \tilde{y}_{t-1}, h_i^{sen}, \hat{D}) \tag{9}$$

In addition to maximum likelihood error training, we also use RL to learn from maximising discrete metrics that are task-specific (which we design as the rewards). We use the self-critical policy gradient algorithm suggested in (Rennie et al., 2017) for training the network.

The reward obtained by the inference time algorithm (which performs greedy decoding) is baselined for the REINFORCE (Williams, 1992) algorithm, without the need for training a "critic" network for estimating the value functions. During training, two output sequences are produced: y^s, obtained by sampling $p(y_t^s|y_1^s, \ldots, y_{t-1}^s, x)$ probability distribution, and y^g, the baseline output, obtained by greedily maximizing the output probability distribution at each time step.

$$L_{RL} = (r(y^g) - r(y^s)) \sum_{t=1}^{n'} \log p(y_t^s|y_1^s, \ldots, y_{t-1}^s, h_i^{sen}, \hat{D}) \tag{10}$$

Our reward function $r(y)$, used for evaluating y against the gold standard output is

$$r(y, \tilde{y}) = \lambda_1 \cdot r1(y, \tilde{y}) + \lambda_2 \cdot r2(y, \tilde{y}) + \lambda_3 \cdot r3(y, \tilde{y}) \tag{11}$$

The final reward function is the weighted mean of the three terms as given below:

(i). BLEU metric $r1$: Ensures the content matching between the generated response and the ground-truth response.

(ii). Sentiment consistency $r2$: Measured by the cosine similarity of the sentiment prediction distributions of the user utterance and generated responses (using pre-trained BERT classifier). It ensures that the sentiment states of the generated polite response is consistent with the user sentiment.

(iii). Politeness accuracy $r3$: The politeness accuracy of the generated response is computed using the pre-trained BERT based politeness classifier (trained on Stanford Politeness Corpus (Danescu-Niculescu-Mizil et al., 2013)). The responses having scores greater than 0.7 are considered as polite.

We first pre-train using the maximum likelihood (ML) objective and then using a mixed objective function with a reduced learning rate:

$$\mathcal{L}_{gen} = \eta L_{RL} + (1 - \eta) L_{MLE}, \tag{12}$$

Joint Training (JT): We jointly train the entire model by simultaneously minimizing the sentiment classification loss and generation loss. The final loss of the model is:

$$\mathcal{L}_{Joint} = \mathcal{L}_{SE} + \mathcal{L}_{gen} \tag{13}$$

3.3 Baseline Models:

To demonstrate the effectiveness of our proposed model, we compare with the previous state-of-the-art (SoTA) models:

Seq2Seq: It is the standard encoder-decoder framework with attention mechanism that has been widely used in generation, machine translation etc. (Sutskever et al., 2014).

HRED: It is a hierarchical encoder-decoder model proposed for text based dialogue systems (Serban et al., 2015).

Polite-RL: We implement the Polite-RL framework to induce politeness in responses in a similar manner as (Niu and Bansal, 2018).

PT-TGA: We implement the politeness transfer framework presented in (Madaan et al., 2020) that uses a tag and generate approach to incorporate politeness.

PG-RL: We also take the reinforced pointer generator network employed in (Golchha et al., 2019) as one of the baselines to infuse politeness in generic responses.

To demonstrate the effectiveness of each of the components in the proposed model, we experiment

with different model variants.
HT: In this model, we use the hierarchical transformers to induce politeness in responses without the RL rewards and sentiment information.
HT + RL: In this framework, we include the RL based rewards in the hierarchical model for polite response generation.
HT + RL + SE: In this framework, we provide the sentiment information to the hierarchical Transformer encoder-decoder model along with RL rewards without jointly training for both the tasks.

For sentiment classification, we train several classifiers such as CNN, LSTM, Bi-LSTM on the CYCCD dataset for predicting sentiments into 3 classes. We also employ RoBERTa (Liu et al., 2019) as one of the baselines.

	Train	*Valid*	*Test*
# Conversation	130898	19762	39665
# Utterances	168534	24724	49788

Table 2: CYCCD Dataset Statistics

4 Datasets and Experiments

In this section we explain the dataset used for experimentation and briefly provide the implementation details and evaluation metrics.

4.1 Dataset:

For our current work, we use the CYCCD dataset (Golchha et al., 2019)[1] having interactions between customers and professional customer care agents of companies on their Twitter handles. The CYCCD Twitter data was taken from the dataset made available on Kaggle by Thought vector. We use the generic and polite annotated version of the CYCCD dataset in a similar manner as (Golchha et al., 2019). As the CYCCD dataset was not annotated for sentiment, therefore we do the sentiment annotations for the dataset.

To annotate the CYCCD dataset with sentiments, we employ crowd-workers from Amazon Mechanical Turk (AMT) that labels every utterance with the provided set of sentiment labels (i.e., positive, negative, neutral). For labeling the utterances, the workers were asked to follow the instructions and guidelines provided for annotation. Some of the significant guidelines for annotation were as follows: (i). Every utterance of a given dialogue was to be marked with the provided sentiment labels; (ii) In addition, the workers were asked to provide the overall sentiment for every sentence in an utterance as well. For cases where we found different annotations in sentiment for a particular sentence, we remove them from the dataset, and we also drop the entire conversation to maintain coherence among the utterances.

A majority voting scheme was used for selecting the final sentiment for every sentence. We observe a multi-rater Kappa (McHugh, 2012) agreement ratio of approximately 75% for the sentiment, which can be considered as reliable. The final CYCCD data statistics is provided in Table 2. The sentiment distribution of the CYCCD dataset is provided in Figure 2.

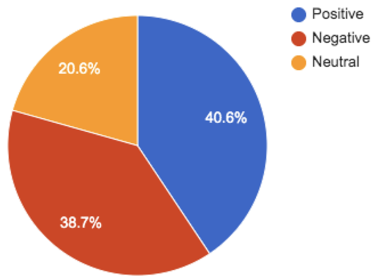

Figure 2: Sentiment distribution in the CYCCD dataset

4.2 Implementation Details:

All the implementations were done using the PyTorch[2] framework. We use the dropout (Srivastava et al., 2014) with probability 0.45. We initialize the model parameters randomly using a Gaussian distribution with the Xavier scheme (Glorot and Bengio, 2010). The hidden size for all the layers is 512. We employ AMSGrad (Reddi et al., 2019) as the optimizer for model training to mitigate the slow convergence issues.

We use uniform label smoothing with $\epsilon = 0.1$ and perform gradient clipping when the gradient norm is above 5. We use 300-dimensional word-embedding initialized with Glove (Pennington et al., 2014) embedding pre-trained on Twitter. We train with batches of size 16, and use the same size for beam search decoding. We use $\eta = 0.99$ (similar to (Paulus et al., 2017)) for the joint loss. For the reward function, the values of λ_1, λ_2 and λ_3 are 0.34, 0.33 and 0.33, respectively.

[1] https://github.com/Mauajama/Courteously-Yours

[2] https://pytorch.org/

4.3 Automatic Evaluation Metrics:

To evaluate the model at the relevance and grammatical level, we report the results using the standard metrics like Perplexity (Chen et al., 1998), Rouge-L (Lin, 2004) and BLEU-4 (Papineni et al., 2002). We also report the Politeness Accuracy as a metric to measure the degree of politeness in the responses.

We compute the politeness score using a pre-trained classifier, BERT (Devlin et al., 2018)[3] for measuring the degree of politeness in the generated responses similar to (Niu and Bansal, 2018). The classifier takes as input the generated response and generates a probability value giving us the politeness accuracy of the generated response. To evaluate the sentiment classification performance, we use the traditional metrics such as F1 score.

4.4 Manual Evaluation Metrics:

We recruit six annotators (in a similar manner as (Shang et al., 2015; Tian et al., 2019)) from a third party company, having high-level language skills. We sampled 250 responses per model for evaluation with the utterance and the conversational history provided for generation. First, we evaluate the quality of the response on two conventional criteria: *Fluency* and *Relevance*. These are rated on a five-scale, where 1, 3, 5 indicate unacceptable, moderate, and excellent performance, respectively, while 2 and 4 are used for unsure.

Secondly, we evaluate the politeness quotient of a response in terms of *Politeness Appropriateness* metric that measures whether the politeness induced in the response is in accordance with the user sentiment and the dialogue history. Here, 0 indicates irrelevant or contradictory, and 1 indicates consistent with the provided persona and dialogue context.

We compute Fleiss' kappa (Fleiss, 1971) to measure inter-rater consistency. The Fleiss' kappa for fluency and relevance are 0.53 and 0.49, indicating moderate agreement. For politeness appropriateness, we obtain 0.65 as the kappa score indicating substantial agreement.

5 Result and Analysis

In this section, we provide the experimental results for both sentiment classification and polite response generation. The proposed model performs significantly better than the other baselines for all the evaluation metrics and the improvement in each model is statistically significant compared to the other models for both the tasks[4].

The sentiment classification results are provided in Table 3. From the results, it is evident that the BERT based classifier outperforms all the other models for sentiment classification task with an improvement of more than 15 points compared to the Bi-LSTM. Hence, it can be concluded that the BERT classifier correctly identifies the sentiment of a sentence.

Model	S-F1
LSTM	71.06
CNN	69.90
Bi-LSTM	74.87
BERT (Devlin et al., 2018)	89.74
RoBERTa (Liu et al., 2019)	88.89

Table 3: Classification scores of sentiment on CYCCD data. Here, S-F1 denotes the weighted average F1 score of sentiment.

Model Description		PPL	BLEU-4	Rouge-L	PA
Existing Approaches	*Seq2Seq* (Sutskever et al., 2014)	1.112	0.145	0.278	0.38
	HRED (Serban et al., 2015)	1.085	0.198	0.308	0.45
	Polite-RL (Niu and Bansal, 2018)	1.028	0.224	0.321	0.69
	PT-TGA (Madaan et al., 2020)	1.032	0.251	0.332	0.68
	PG-RL (Golchha et al., 2019)	1.018	0.264	0.339	0.73
Proposed Approach	**HT + RL + SE (Joint Training)**	**1.004**	**0.275**	**0.352**	**0.77**
Ablation Study	*HT*	1.015	0.269	0.343	0.70
	HT + RL	1.008	0.272	0.349	0.74
	HT + RL + SE	1.006	0.273	0.350	0.75

Table 4: Automatic evaluation results. Here, PPL: Perplexity, PA: Politeness accuracy, HT: Hierarchical transformer, SE: Sentiment, RL: Reinforcement learning

In Table 4, we provide the automatic evaluation results for the proposed framework and the existing baselines. From the table, it is evident that the proposed network outperforms the *Seq2Seq* and *HRED* frameworks in terms of all the metrics. In the case of politeness, there is an enormous improvement in comparison to the Seq2Seq and HRED frameworks with more than 30% accuracy.

In comparison to the *Polite-RL* framework, our proposed network is capable of preserving the information and generating correct responses as the BLEU-4 score is significantly higher. It can be concluded that the generation of polite responses does not comprise the information present in the

[3]The classifier is trained on the Stanford Politeness Corpus (Danescu-Niculescu-Mizil et al., 2013) and achieves an accuracy of 92%.

[4]we perform statistical significance tests (Welch, 1947) and it is conducted at 5% (0.05) significance level

Dialogue Context	Sentiment Information	Generic Response	Generated Polite Examples
My order doesnot have fries in it, that sucks!	Negative	we'll follow up with the store	**HT:** We will check **HT + RL:** Please, wait we'll follow up with the store. **HT + RL + SE:** Sorry, we'll follow the order with the store **Proposed:** Sorry for the inconvenience, please wait while we follow with the store.
The new iphone has awesome display	Positive	enjoy your new iphone!	**HT:** Ohh ur new phone, great **HT + RL:** That's nice about your iphone **HT + RL + SE:** Thanks and enjoy your phone. **Proposed:** Thanks for your kind words and enjoy your new phone!

Table 5: Examples of polite responses generated by different models according to the sentiment information

Model Description		F	R	PA
Existing Approaches	*Seq2Seq* (Sutskever et al., 2014)	3.82	3.73	48%
	HRED (Serban et al., 2015)	3.86	3.78	52%
	Polite-RL (Niu and Bansal, 2018)	3.91	3.79	61%
	PT-TGA (Madaan et al., 2020)	4.03	3.85	64%
	PG-RL (Golchha et al., 2019)	4.11	4.06	67%
Proposed Approach	**HT + RL + SE (Joint Training)**	**4.23**	**4.17**	**75%**
Ablation Study	*HT*	4.09	4.03	65%
	HT + RL	4.16	4.09	71%
	HT + RL + SE	4.19	4.12	73%

Table 6: Human evaluation results. Here, PA: Politeness Appropriateness, HT: Hierarchical Transformer, SE: Sentiment, RL: Reinforcement Learning

ground-truth generic response. In the case of automatic evaluation in Table 4, our method shows a notable drop in the perplexity scores, thereby ensuring grammatically correct responses generated by the framework. By introducing sentiment information in our proposed framework, we see the growth in the performance compared to the *PG-RL* network establishing the importance of sentiment information for generating polite responses.

In Table 4, we provide the ablation study of our proposed network. From the table, it is visible that the Hierarchical Transformer network performs better than the *Seq2Seq* and *HRED* frameworks, but lacks politeness in comparison to the existing *PG-RL* baseline. This is because the model is trained without politeness and sentiment rewards.

On adding the RL training to the hierarchical network, we see that there is an improvement in terms of all the metrics. An increase in BLEU-4, Rouge-L, and politeness accuracy scores signifies that the rewards used for training the framework contribute towards better generation.

Finally, in *HT + RL + SE* adding the sentiment information of the user helps in improving the performance of the overall network. Joint training helps in simultaneously improving the performance of the end-to-end network for generating sentiment-guided polite responses.

From Table 6, it is evident that the proposed method generates grammatically correct responses as the fluency score is the highest. Similarly, the relevance score in the case of our proposed network is greater than all the existing and baseline approaches signifying that the generated responses are contextually correct according to the dialogue history.

As the primary aim of our current work is to generate politeness according to the sentiment information, the politeness appropriateness metric helps evaluate the proposed task for all the networks. The proposed framework has the highest politeness score compared to all the existing and baseline frameworks. Also, through ablation study in case of manual evaluation, it can be established that the proposed framework having joint training and RL rewards helps the overall architecture to generate polite responses according to the users' sentiments.

In Table 5, we provide a few examples and their corresponding generated responses by the different models. It exhibits that while the existing *Seq2Seq* framework generates shorter and less polite responses, the *HRED* framework can generate more complete and informative responses. The responses generated by the proposed model exhibit different variations in politeness according to the user sentiments. The variation following the different sentiments is visible in the responses, thereby achieving the desired task.

After performing a detailed quantitative and qualitative analysis of the generated responses, we came across a few mistakes committed by the baselines and the proposed frameworks.
Some of the commonly occurring errors are:
(i) Repetition: The responses generated by the baselines and the proposed framework sometimes repeat the information or generate $< unk >$ tokens. For example, in the following generated response,

there is a repetition: "*Thanks, could you dm us more info info info...*"

. ***(ii) Loss of information:*** In some cases, the models generate responses that, though having politeness factor in them, yet is incapable of providing the complete information in accordance to the ongoing conversation. For example: Gold:"*We appreciate the concern regarding the new update, how could I help*"; Predicted: "*Hello, good afternoon! How may I help?*".

(iii) Sentiment inconsistency: The generated responses having politeness in them at times do not correspond to the specified user sentiment. For example: the user sentiment is negative and the Predicted: "*Thank you ma'am, we love it...*" response is not consistent to it. This is due to the fact that the multi-task framework has wrongly predicted the sentiment of the user therefore causing the agent to respond incorrectly.

6 Conclusion

In our current work, we propose the task of transforming a generic customer care response into a polite response according to the sentiment information of the user and consistent to the context history. For the proposed task, we design a hierarchical Transformer network that captures the user and dialog context simultaneously. We create a multi-task network that identifies the sentiment information and uses the predicted sentiment information for generating the corresponding polite response. We use reinforcement learning based rewards to incorporate politeness in the customer care responses.

Experiments on the sentiment annotated CY-CCD dataset proves that the proposed network not only identifies the sentiment information but also generate the contextually correct polite responses in accordance to the user sentiments. Hence, it can be concluded that the polite generation of responses is dependent on the sentiments of the user to ensure that the generation is correct and relevant.

7 Ethical Declarations

All the resources used in this paper are publicly available. The dataset used in this paper is used only for the purpose of academic research. There is nothing to disclose that warrant the ethical issues.

Acknowledgement

Authors acknowledge the partial support of the I-Hub Anubhuti-IIITD Foundation to carry out this research. Authors also acknowledge the partial support from the project "Sevak-An Intelligent Indian Language Chatbot", sponsored by Imprint, SERB.

References

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Keith Carlson, Allen Riddell, and Daniel Rockmore. 2017. Zero-shot style transfer in text using recurrent neural networks. *arXiv preprint arXiv:1711.04731*.

Stanley Chen, Douglas H. Beeferman, and Ronald Rosenfeld. 1998. Evaluation metrics for language models.

Cristian Danescu-Niculescu-Mizil, Moritz Sudhof, Dan Jurafsky, Jure Leskovec, and Christopher Potts. 2013. A computational approach to politeness with application to social factors. *arXiv preprint arXiv:1306.6078*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Nan Duan, Duyu Tang, Peng Chen, and Ming Zhou. 2017. Question generation for question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 866–874.

William Fedus, Ian Goodfellow, and Andrew M Dai. 2018. Maskgan: Better text generation via filling in the _. *arXiv preprint arXiv:1801.07736*.

Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Mauajama Firdaus, Asif Ekbal, and Pushpak Bhattacharyya. 2020. Incorporating politeness across languages in customer care responses: Towards building a multi-lingual empathetic dialogue agent. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4172–4182.

Mauajama Firdaus, Umang Jain, Asif Ekbal, and Pushpak Bhattacharyya. 2021. Seprg: Sentiment aware emotion controlled personalized response generation. In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 353–363.

Mauajama Firdaus, Arunav Shandilya, Asif Ekbal, and Pushpak Bhattacharyya. 2022a. Being polite: Modeling politeness variation in a personalized dialog agent. *IEEE Transactions on Computational Social Systems*.

Mauajama Firdaus, Nidhi Thakur, and Asif Ekbal. 2022b. Sentiment guided aspect conditioned dialogue generation in a multimodal system. In *European Conference on Information Retrieval*, pages 199–214. Springer.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation. In *AAAI*.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, pages 249–256.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323.

Hitesh Golchha, Mauajama Firdaus, Asif Ekbal, and Pushpak Bhattacharyya. 2019. Courteously yours: inducing courteous behavior in customer care responses using reinforced pointer generator network. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 851–860.

Swati Gupta, Marilyn A Walker, and Daniela M Romano. 2007. How rude are you?: Evaluating politeness and affect in interaction. In *International Conference on Affective Computing and Intelligent Interaction*, pages 203–217. Springer.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778.

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 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 64–73.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. 2017. Toward controlled generation of text. In *ICML*.

Jiwei Li, Will Monroe, Alan Ritter, Daniel Jurafsky, Michel Galley, and Jianfeng Gao. 2016. Deep reinforcement learning for dialogue generation. In *EMNLP*.

Jiwei Li, Will Monroe, Tianlin Shi, Alan Ritter, and Daniel Jurafsky. 2017. Adversarial learning for neural dialogue generation. In *EMNLP*.

Juncen Li, Robin Jia, He He, and Percy Liang. 2018. Delete, retrieve, generate: A simple approach to sentiment and style transfer. In *NAACL-HLT*.

Chin-Yew Lin. 2004. Rouge: a package for automatic evaluation of summaries.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Aman Madaan, Amrith Setlur, Tanmay Parekh, Barnabas Poczos, Graham Neubig, Yiming Yang, Ruslan Salakhutdinov, Alan W Black, and Shrimai Prabhumoye. 2020. Politeness transfer: A tag and generate approach. *arXiv preprint arXiv:2004.14257*.

Mary L McHugh. 2012. Interrater reliability: The kappa statistic. *Biochemia medica: Biochemia medica*, 22(3):276–282.

Kshitij Mishra, Mauajama Firdaus, and Asif Ekbal. 2022a. Please be polite: Towards building a politeness adaptive dialogue system for goal-oriented conversations. *Neurocomputing*, 494:242–254.

Kshitij Mishra, Mauajama Firdaus, and Asif Ekbal. 2022b. Predicting politeness variations in goal-oriented conversations. *IEEE Transactions on Computational Social Systems*.

Tong Niu and Mohit Bansal. 2018. Polite dialogue generation without parallel data. *TACL*, 6:373–389.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. Association for Computational Linguistics.

Romain Paulus, Caiming Xiong, and Richard Socher. 2017. A deep reinforced model for abstractive summarization. *CoRR*, abs/1705.04304.

Yehong Peng, Yizhen Fang, Zhiwen Xie, and Guangyou Zhou. 2018. Topic-enhanced emotional conversation generation with attention mechanism. *Knowledge-Based Systems*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. 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 1532–1543.

Sudha Rao and Joel Tetreault. 2018. Dear sir or madam, may i introduce the gyafc dataset: Corpus, benchmarks and metrics for formality style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 129–140.

Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. 2019. On the convergence of adam and beyond. *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

Sathish Reddy, Dinesh Raghu, Mitesh M Khapra, and Sachindra Joshi. 2017. Generating natural language question-answer pairs from a knowledge graph using a rnn based question generation model. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, volume 1, pages 376–385.

Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1179–1195.

Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2015. Hierarchical neural network generative models for movie dialogues. *arXiv preprint arXiv:1507.04808*, 7(8):434–441.

Iulian Vlad Serban, Tim Klinger, Gerald Tesauro, Kartik Talamadupula, Bowen Zhou, Yoshua Bengio, and Aaron C Courville. 2017a. Multiresolution recurrent neural networks: An application to dialogue response generation. In *AAAI*, pages 3288–3294.

Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron C Courville, and Yoshua Bengio. 2017b. A hierarchical latent variable encoder-decoder model for generating dialogues. In *AAAI*, pages 3295–3301.

Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. 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 1577–1586.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *Advances in Neural Information Processing Systems*, pages 6830–6841.

Xiaoyu Shen, Hui Su, Shuzi Niu, and Vera Demberg. 2018. Improving variational encoder-decoders in dialogue generation. In *AAAI*.

Weiyan Shi and Zhou Yu. 2018. Sentiment adaptive end-to-end dialog systems. *arXiv preprint arXiv:1804.10731*.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Sandeep Subramanian, Sai Rajeswar, Francis Dutil, Chris Pal, and Aaron Courville. 2017. Adversarial generation of natural language. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 241–251.

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.

Zhiliang Tian, Wei Bi, Xiaopeng Li, and Nevin L Zhang. 2019. Learning to abstract for memory-augmented conversational response generation. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3816–3825.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

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

Bernard L Welch. 1947. The generalization ofstudent's' problem when several different population variances are involved. *Biometrika*, 34(1/2):28–35.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256.

Xianchao Wu, Ander Martinez, and Momo Klyen. 2018. Dialog generation using multi-turn reasoning neural networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 2049–2059.

Hainan Zhang, Yanyan Lan, Jiafeng Guo, Jun Xu, and Xueqi Cheng. 2018. Reinforcing coherence for sequence to sequence model in dialogue generation. In *IJCAI*, pages 4567–4573.