AMERICAN UNIVERSITY OF BEIRUT

NEGOTIATING BOTS WITH EMPATHY

by
CHRISTIAN PIERRE HOKAYEM

A thesis
submitted in partial fulfillment of the requirements
for the degree of Master of Engineering
to the Department of Electrical and Computer Engineering
of the Maroun Semaan Faculty of Engineering and Architecture
at the American University of Beirut

Beirut, Lebanon
September 2021

AMERICAN UNIVERSITY OF BEIRUT

NEGOTIATING BOTS WITH EMPATHY

by
CHRISTIAN PIERRE HOKAYEM

Approved by:

Dr. Hazem Hajj, Associate Professor Electrical and Computer Engineering	Advisor
Dr. Rabih Jabr, Professor Electrical and Computer Engineering	Member of Committee
Dr. Wassim El Hajj, Associate Professor Computer Science	Member of Committee

Date of thesis defense: August 11, 2021

AMERICAN UNIVERSITY OF BEIRUT

THESIS, DISSERTATION, PROJECT RELEASE FORM

Student Name: Hokayem Christian Pierre
Last First Middle

☑ Master's Thesis ◯ Master's Project ◯ Doctoral Dissertation

☐ I authorize the American University of Beirut to: (a) reproduce hard or electronic copies of my thesis, dissertation, or project; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes.

☑ I authorize the American University of Beirut, to: (a) reproduce hard or electronic copies of it; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes after: **One ___ year from the date of submission of my thesis, dissertation or project.**
Two _✓_ years from the date of submission of my thesis , dissertation or project.
Three ___ years from the date of submission of my thesis , dissertation or project.

Christian H. September 15, 2021
Signature Date

This form is signed when submitting the thesis, dissertation, or project to the University Libraries

Acknowledgements

I would like to first thank Prof. Hazem Hajj for his generous and continuous guidance, support, and motivation throughout my time at AUB. I would also like to thank Prof. Rabih Jabr and Prof. Wassim El Hajj for their feedback, advice, and encouragement. Finally, I would also like to thank my fellow MIND lab colleagues, friends, and family who accompanied me along this journey.

An Abstract of the Thesis of

Christian Pierre Hokayem for Master of Engineering
Major: Electrical and Computer Engineering

Title: Negotiating Bots with Empathy

Automated negotiation is a multi-agent task where one or multiple negotiating bots aim to resolve a conflict or reach a mutually beneficial agreement. Previous approaches have focused on achieving financially optimal outcomes with no consideration for user sentiments. However, as negotiations typically occur within the context of ongoing relationships, maintaining a pleasant overall experience is undeniably crucial. In this thesis, we tackle the problem of an item sale negotiation where a buyer agent seeks to obtain an item from a seller agent. The goal is to develop a seller negotiating bot with the objective of simultaneously maximizing both buyer satisfaction and sale price. We compare two approaches to the problem. The first approach consists of using a single end-to-end Long Short-Term Memory sequence-to-sequence (LSTM seq2seq) model with attention mechanism that takes in previous utterances as input and generates the next utterance. The second approach consists of breaking down the model into 3 parts: a rule-based parser which extracts the negotiation act and sentiment of the received utterance, a seq2seq manager which recommends the next act and sentiment, and a fine-tuned Generative Pre-trained Transformer (GPT-2) generator which transforms the recommended act and sentiment into a complete response. We make use of a mixed learning approach which combines supervised learning with goal-oriented reinforcement learning to efficiently train both the end-to-end model and the manager's decision model. Compared to previous work, the experiment results showed improvement in item representation, consistency of offers, buyer sentiment, empathy, fluency, appropriateness, and human likeness.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, we introduce the topic and outline the existing challenges as well as the main contributions of this thesis.

1.1 Background & Motivation

Negotiation is present in almost every aspect of personal and professional life: scheduling meetings and events, exchanging goods, deciding on restaurants, performing sales and customer support, making management decisions, or even agreeing on a bed time with a child all require negotiation. In a nutshell, negotiation is defined as a back and forth communication which seeks to reach agreement between two or more parties [1]. Automating the negotiation process has been seen as one of "society's key technological challenge for the near future" as it allows for a quick and efficient way to resolve conflicts in various domains such as automated marketplaces, smart grids, service level agreements (SLAs), and even autonomous driving [2].

Most approaches to date focused on bot-to-bot negotiation with rule-based decision models based on game theory [3][4], heuristics [5][6][7][8], or fuzzy logic [9][10] where the goal is to find a Pareto optimal distribution of issues via the exchange of numerical vectors representing offers. More recently, human-like data-driven approaches have been explored [11][12][13]. In these approaches, agents are trained to be capable of holding a dialogue with any other agent (human or bot) and perform actions such as making offers, counteroffers, and accepting or rejecting certain outcomes. The decision model for these agents typically follows a two-step training process: In the first step, the model is trained in a supervised learning (SL) manner to imitate human negotiation behavior and predict the next most likely utterance in a sequence. In the second step, reinforcement learning (RL) is applied to fine-tune the model parameters through simulated interactions and maximize a certain reward function associated with the outcome of the negotiation [11][12].

The reward function varies across different works, but is always associated with a purely financial metric (such as price) with no consideration for customer sentiment. This has driven bots towards deceptive behavior which can be harmful to the seller's image and the customer's long term loyalty [11].

1.2 Problem Description

In this work, we seek to overcome the limitations of prior work. We tackle the problem of building an automated negotiating bot with empathy. We focus on the setting of item sales in English where two agents (a buyer and a seller) negotiate over the price of an item. We aim to automate the actions of the seller agent and have it respond in a way that simultaneously maximizes the sale price and the sentiments of the buyer. Figure 1.1 shows a blackbox representation of the desired bot in action.

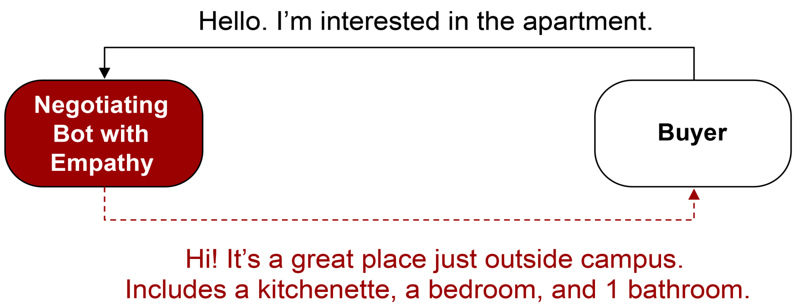

Figure 1.1: Blackbox representation of the automated negotiation bot with empathy. The bot receives a message with an offer from the buyer and must then reply to it with its own empathetic message and offer

We compare the proposed bot to traditional bots in Table 1.1: We see that while a traditional negotiating bot is trained to focus on maximizing its own financial gain with no regards to the sentiment of the user, a negotiating bot with empathy considers user sentiment and seeks to further improve the negotiation experience by guiding the conversation to a more positive tone and by making more appealing offers although these offers may present lower immediate financial gains to the seller.

Table 1.1: Example of a sales negotiation bot with empathy. The traditional negotiation bot ignores user sentiment and fails to reach agreement while the negotiation bot with empathy reacts to negative user sentiment and works to improve it.

Agent	**Traditional Negotiation Bot**	**Negotiation Bot with Empathy**
Human	Hello. I am very interested in the apartment you have for sale.	Hello. I am very interested in the apartment you have for sale.
Bot	Hi. It is a great place with a great location.	Hi. It is a great place with a great location.
Human	I know it is. But the price is a bit too high for students to live in. Would you consider going down to something closer to $2000?	I know it is. But the price is a bit too high for students to live in. Would you consider going down to something closer to $2000?
Bot	How does $2400 sound?	**We love to accommodate students!** How about a special discount down to $2400?
Human	That's still way higher than I can afford!	That's better, but I really can't pay more than $2200!
Bot	<final offer: $2400>	Well, how about $2250! **That's really the best price we can give you.**
Human	<reject>	Wow! Thank you that's great!
Bot		**<final offer: $2250>**
Human		<accept>

1.3 Challenges & Contributions

Designing and building negotiating bots, in general, involves overcoming the following challenges:

1. **Ability to converse in natural language** (namely English): This includes the ability to generate responses which are human-like and demonstrate understanding of the current state of the dialogue. As most negotiations happen within informal settings, the bot must also be able to understand and generate informal language (slang words, abbreviations, emojis, ...).

2. **Ability to bargain**: This includes the ability of being able to make, receive, evaluate, and adapt offers according to the conversation and knowledge of the negotiation issue (i.e. item on sale).

3. **Learning from small datasets**: This includes the ability to achieve the above with limited data as existing datasets tend to be few and small (in the range of a few thousand dialogues).

We seek to address these existing challenges while targeting the following contributions:

1. **Integrating empathy in the automated negotiation process**: This involves tackling the question of how to deliver empathetic responses in a goal-oriented system where the goal may differ or conflict with that of generating empathetic responses. It also involves tackling the question of how to react and adapt the bargaining skills of the model in order to reach improved user sentiment at the end of the negotiation process. We propose an architecture which is capable of understanding and expressing sentiments as well as a reward function which combines both price and sentiment and allows the model to simultaneously optimize for both.

2. **Improving bargaining skills of state-of-the-art models**: This involves improving the existing models' accuracy in representing item information (avoiding misrepresentation or deceitful information). It also involves improving the models' ability to maintain a certain consistency in its offers and avoid deceptive pricing which have both been found to be lacking in previous work. To tackle this challenge, we propose delexicalizing the training data and using slot-filling in combination with mixed learning.

3. **Proposing an evaluation framework for negotiating bots with empathy**: This involves addressing the question of how to evaluate such a bot and its success or failure. To that end, we propose automated metrics which consider the outcome of the negotiation (price and sentiment) as well as the process (presence of inconsistent offers and improper item information) and supplement them with human ratings for evaluating the generated language quality as well as the empathetic capabilities of the bot.

1.4 Thesis Organization

The rest of the thesis is organized as follows: In Chapter 2, we review the current state of the literature and highlight the existing gap. In Chapter 3, we outline our proposed method. In Chapter 4, we detail experiment results. Finally, in Chapter 5, we conclude.

Chapter 2

Literature Review

While work exists on automated negotiating bots and empathetic chatbots, an inter-section between these two remains severely understudied: to date and to our knowledge, no work on negotiating chatbots with empathy exist. Therefore, the focus of this section will be on presenting advancements in both domains.

2.1 Automated Negotiation Bots

Traditional approaches for developing negotiating bots relied on sets of rules which dictate a strategy to follow using a series of if-then statements [10][7][5]. Other approaches looked at the problem from a game-theory point of view and focused on solving the non-cooperative problem of "what is the best, most rational thing to do regardless of how other agents will behave?". One such approach which proved popular was the monotonic concession approach where one or both agents have to make concessions each time an agreement is not reached. Monotonic concession was often applied with Zeuthen strategy which formally defines which agent should conceed and by how much. However, computing the needed outcomes required an exponential number of computations at each negotiating round prompting the need for faster, less computationally expensive methods [4][3].

To answer that need, Zhai et al. [9] explored the solution space through genetic algorithms in combination with monotonic concession. Meanwhile, Koley et al. [8] introduced two heuristics to guide the decision making process: Most Changed Least Preferred (MCLP) and Most Offered Most Preferred (MOMP). Using these heuristics as well as others related to the mean and standard deviations in the offer (model cooperativeness and risk-taking), their model adapted different strategies (based on the Thomas- Kilmann conflict mode Instrument (TKI) [14]) in order to quickly reach optimality. Another approach was proposed by Chaharsooghi et al. [6] who leveraged information from multiple simultaneously occurring negotiations, in order to incorporate current market conditions into the rule-based decision model. Regardless of how they approached it however, all these works had two main limitations. Namely, the inability to negotiate with users in Natural Language (such as English) and the requirement of handcrafting sets of rules for every bot.

Lewis et al. [11] were the first to leverage advances in deep learning in order to overcome those limitations. In their work, they presented an end-to-end model for natural language negotiation agents around the exchange of 3 negotiation issues: Book, Hat, and Ball. The authors used Gated Recurrent Unit (GRU) Recurrent Neural Network (RNN) models to generate responses and offers. They initially trained the model to mimic human negotiating behavior and, in a second step, introduced reinforcement learning to further maximize utility. It was seen that the model performed well but that it also resorted to underhanded tactics such as deceit and manipulation in order to maximize utility. Another major issue that was encountered was degeneracy after reinforcement learning where the model would create incoherent utterances.

To overcome this second issue, He et al. [12] proposed decoupling the end-to-end model into three separate parts: parser, manager, and generator. The parser extracts intents from previous utterances. The manager is trained with supervised learning then reinforcement learning to predict the next intent to take. And the generator converts that intent into a full utterance through a similarity-based retrieval engine. They evaluated different reward functions (utility, fairness, and dialogue length) as well as the human-likeness and the agreement percentage. They found that their modular approach is seen to be more human-like than a reference basic end-to-end model.

Other researchers continued exploring further issues related to automated negotiations: Cheng et al. [15] investigated the robustness of the natural language negotiation models by introducing "adversarial agents" which they optimized to reduce the utility of the opposing automated agent. Meanwhile, Parvaneh et al. [13] worked on integrating multimodal information into the negotiation process using ResNet for image processing and RNN sequence models with attention for price and response generation. Their approach focused more closely on mimicking human-behavior and moved away from goal-based maximization leading to models which may negotiate similarly to humans but not necessarily optimally. In summary, the main limitation of previously developed negotiating bots is that they have not considered customer satisfaction and have completely neglected the positive impact that an empathetic agent can have on user experience [16].

2.2 Empathetic Bots

Many definitions and models for empathy exist, each encompassing different sets of capabilities: mimicry, affective matching, sympathy, altruistic helping, and perspective taking. A popular model that has seen traction in literature is the

Russian doll model which represents empathy as a 3-level hierarchical structure [17]:

- The first level is the capability of understanding others emotions and expressing appropriate responses.
- The second level is emotion regulation conditioned on several modulation factors such as mood, personality, compatibility, likingness, and familiarity.
- The third and deepest level of empathy is the ability to perform perspective taking and appraisal of the situation the other person is in.

Building chatbots capable of capturing some or all of these levels has been of great interest to the research community. Some works have focused on emotion and context recognition using external wearable sensors, video, speech, or a combination of those things to deterministically generate responses [18][19][20]. Other works, focusing mostly on textual data, have presented end-to-end neural approaches for both recognising emotions and generating appropriate responses. For instance, Sun et al. [21] used SeqGANs to generate responses. To model empathy, they included an emotion embedding in the generator and incorporated it into the reward scheme to encourage appropriate emotional responses. Another work was presented by Rashkin et al. [22] where they sought to overcome the issue of limited datasets for empathy by gathering a new dataset called EmpatheticDialogues. They then used that dataset to train their own sequence to sequence model, conditioning its generation on the emotion recognised by their BERT model. Lin et al. [23] would later build on their work to fine-tune the state-of-the-art Generative Pre-trained Transformer (GPT) and surpass their results. Approaching the problem from a different perspective, Shin et al. [24] proposed applying reinforcement learning on the empathetic dialogue system.They fine-tuned BERT to predict the next user sentiment score and then used that predicted score as a reward to update the policy of the model. Their model was seen by human judges to be more empathetic and its responses to be more relevant and fluent when compared to previous works.

Other ways to represent empathy have also been investigated. Asghar et al. [25] proposed the use of a 3-dimensional continuous affective space consisting of Evaluation (E), Potency (P), and Activity (A). Using a BiLSTM network, a response is mapped to a vector in the affective space. The vector's coordinates are then fed into an ACT (Affect Control Theory) module which deterministically computes the response vector. The bot then conditions its response generation model on that obtained vector. Zhou et al. [26] also designed and deployed a large modular chatbot titled XiaoIce. This chatbot relied on machine learning classifiers to detect user intent, emotion, topic, sentiment, and opinion. They combined that with user specific information into what they called an empathy vector and used it to inform the dialogue policy. Their response generation was mainly built on a retrieval model where a database is searched for the most appropriate response. Though, for open-ended dialogue, a sequence to sequence model was used and its decoder was conditioned on the empathy vectors.

All the works in empathy, however are still limited as data-driven approaches tend to focus only on general conversations while goal-oriented systems tend to rely on a fixed set of responses. No work exists which combines both the goal-oriented approach of negotiations and the user sentiment consideration of empathetic chatbots.

2.3 Summary Table

Table 2.1: Qualitative comparison against relevant previous works. The work in this thesis seeks to combine negotiation with empathy while overcoming previous works' limitations.

Reference	**Conversation**	**Bargaining**	**Limited Data**	**Empathy**	**Evaluation**
[5-10]		✓ no dialogue			
[13]	✓	✓ false item information, suboptimal outcomes	✓		✓ outcome's deviation from human data
[11-12]	✓	✓ inconsistent offers, deceptive behavior, false item information	✓		✓ financial outcome, human ratings
[21-26]	✓		✓	✓ open-ended response	✓ response deviation from human data, human rating
This work	✓	✓ proper item information, consistent offers	✓	✓ in goal and response	✓ outcome, process, human ratings

Chapter 3

Proposed Method

3.1 Problem Formulation

We consider the setting of a bilateral item price negotiation where a buyer seeks to obtain an item from an automated seller negotiation bot. Both parties are presented with a scenario which describes an item: name, category, short description (including condition of the item on sale), listing price, and a buyer target price which is only visible to the buyer. The buyer initiates a conversation and dialogues with the seller until they reach one of three possible outcomes:

- A final offer is made by one agent and accepted by the other.
- A final offer is made by one agent and rejected by the other.
- One of the two agents quits prematurely.

Computationally, each negotiation dialogue can be considered an episode with an alternating sequence of utterances. Utterances consist of an action (message, final offer, accept, reject, or quit) and its arguments (price for offers, message content for messages). The goal of the automated seller negotiation bot with empathy is to take in the current dialogue state (x) which represents the previous utterances and an item context (c) and generate a human-like response utterance (y) towards maximizing seller price and buyer sentiment which jointly make up the reward (r). Figure 3.1 shows a high-level description of the problem and the desired functionality of the negotiating bot with empathy.

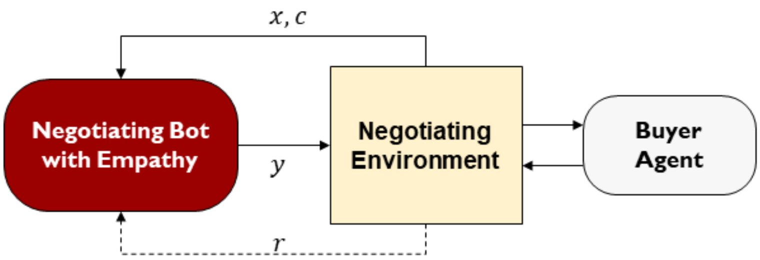

Figure 3.1: High-level description of the role of the negotiating bot with empathy. Given the current dialogue state and item context, the bot must learn to generate a response which maximizes its estimated reward at the end of the negotiation.

3.2 Proposed Models for Negotiation with empathy

We propose and study two different approaches for the automated negotiating seller bot with empathy. The first approach decouples the model into three main components: parser, manager, and generator. While, the second approach consists of using a single end-to-end model which learns directly to map previous utterances to the next utterance. In the following two subsections, we detail both approaches.

3.2.1 Decoupled Model

An overview of the decoupled model for the automated seller negotiation bot with empathy can be seen in Figure 3.2. It is divided into three main components:

- A parser which extracts the coarse act as well as the sentiment expressed by the buyer's utterances. A coarse act is a high-level description of the action taken an agent. It consists of an intent which summarizes the expressed intention of the agent (e.g. introduce themselves, make a counter-offer, ...) and a price argument (if a specific price mention).
- A dialogue manager which receives the extracted intent, price, and sentiment, keeps track of the dialogue state and history (saving current and past intents, prices, and sentiments for both agents), and then prescribes what next intent, price (if any), and sentiment the seller should reply with. The intent and price are used to set the negotiation strategy for the bot while the sentiment sets the tone for an empathetic response.

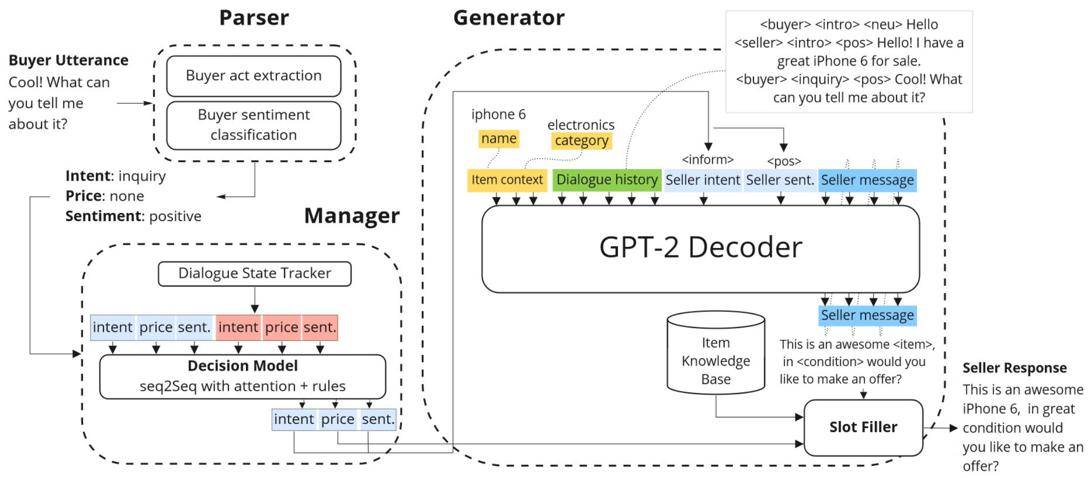

Figure 3.2: Overview of the decoupled model. The parser extracts from the previous utterance the act and sentiment. The manager decides on the response act and sentiment. The generator then turns that decision into a full response.

- A language generator which takes in the prescribed intent, price, and sentiment as well as the preceding dialogue content and item knowledge (name, category, condition) to generate the next message response.

The parser uses regular expressions and sets of rules to assign messages corresponding intents and sentiments while extracting price mentions. We adopt the intents defined by He et al. [12], combine the *greet* and *intro* intents together, and then add further regular expression patterns to reduce the number of unknown utterances in the dataset. A full list of the regular expressions and rules used for intent matching can be found in Table 3.1. We further augment the parser with a 3-level sentiment classifier to decide on the following: negative, neutral, or positive. We use a third party classifier named Valence Aware Dictionary and sEntiment Reasoner (VADER) [27]. VADER is a rule-based sentiment analysis engine which has been found to have good accuracy on social media text which makes it suitable for the type of text found in an online marketplace negotiation (informal language, slang words, emojis, abbreviations, ...). For actions which do not contain any message content for parsing - namely: final offer, accept, reject, and quit - the parser is bypassed and the intent becomes the action itself.

As for the manager, it consists of two components. The first component is the dialogue state tracker which maintains a history of intents $i_0, i_1, ..., i_{t-1}$, prices $p_0, p_1, ..., p_{t-1}$, and sentiments $e_0, e_1, ..., e_{t-1}$ for both agents. Meanwhile, the second component is a decision model which takes in that history and selects the next intent, price, and sentiment for the seller to reply with. The decision model is implemented as an LSTM-based seq2seq model with attention where the dialogue history is the coarse sequence $x = (i_0, p_0, e_0, i_1, p_1, e_1, ..., i_{t-1}, p_{t-1}, e_{t-1})$

Table 3.1: Parser's patterns and rules for intent matching. The parser proceeds through the rules and patterns shown from top to bottom. If a rule is triggered or a pattern matches the given utterance, it is assigned the corresponding intent.

<table>
<tr><th>Intent</th><th>Patterns & Rules</th></tr>
<tr><td>intro</td><td>^hi, ^hello, ^hey, ^hiya, ^howdy,
how are you, (you | m) interested in,
good morning, good evening</td></tr>
<tr><td>init-price</td><td>first mention of a price number in dialogue</td></tr>
<tr><td>vague-price</td><td>no price is mentioned + one of the following patterns:
come down, high[est], low[est], go (higher|lower)</td></tr>
<tr><td>counter-price</td><td>new price is mentioned in an utterance after init-price</td></tr>
<tr><td>insist</td><td>same price is mentioned by the same agent again</td></tr>
<tr><td>inquiry</td><td>^what, ^when, ^where, ^why, ^which, ^who,
^whose, ^how, ^do, ^did, ^does, ^are, ^is,
^would, ^will, ^can, ^could, ^any, ?$</td></tr>
<tr><td>agree</td><td>^ok[ay], ^great, ^perfect, ^wonderful, ^fantastic,
^awesome, ^done, ^deal, ^fine,
^sure, that works, i (can | could | will) (do | take) that,
[you | we] (it (is | 's) | have | got) a deal,
[that] (works | would work) [for me], agree[d],
seems (fair | good | great | perfect | reasonable),
that(is | 's) (fine | fair | good | perfect | reasonable)</td></tr>
<tr><td>inform</td><td>previous coarse dialogue act was inquire or:
^there('s | is | 're | are), ^it('s | is) a, ^it('s | is) in</td></tr>
<tr><td>disagree</td><td>no, not, n't, nothing, dont, sorry,
too (high | low | much | little),
unreasonable, little steep</td></tr>
<tr><td>unknown</td><td>none of the above patterns</td></tr>
</table>

Note: *Rules are shown in italics*

defined by the interactions of both buyer and seller agents and the output of the model is the next vector of items in the sequence $y_t = (i_t, p_t, e_t)$ to be expressed by the seller. For intents which correspond to "final offer", "accept", "reject", or "quit" actions, the generator is not called and instead the action is directly shared with the other agent. Conversely, for intents which correspond to a message action (refer to Table 3.1 for list of intents), the vector y_t is instead passed onto the generator for it to generate a full response message. Note that to prevent the model from learning to make deceptive pricing similar to what was observed in [11] and [12], we add rules on p_t which prevent making or accepting price offers which are not in line with the discussion (e.g. making a final offer for a higher price than what was agreed or accepting an offer below the price currently discussed).

Upon receiving y_t, the generator appends the prescribed intent and sentiment to the item context and the dialogue message history and passes it to a fine-tuned GPT-2 [28] model to generate the next seller utterance. Note that the model is trained to generate delexicalized utterances, or in other words utterances which do not include specific item names, conditions, or price mentions. Instead, they are replaced by placeholder slots which are filled in at inference time through information from the item knowledge base (consisting of the name and item condition extracted from the negotiation scenario) and the price described by the manager. This ensures that the model does not share misleading information related to the item or generate a price different than the one prescribed by the manager.

In Figure 3.4, we see an example of the system in action: The parser receives the utterance "Cool! What can you tell me about it?" and extracts the intent, price, and sentiment from it ("inquiry", "none", and "positive", respectively). It then passes them to the manager. The manager's dialogue state tracker saves the received intent, price, and sentiment and passes them alongside previously received intents, prices, and sentiments to the seq2seq decision model. The decision model then recommends the next intent, price, and sentiment ("inform", "none", and "positive", respectively) to reply with and passes those to the generator. The generator appends these recommendations unto a sequence consisting of the item name and category and previous dialogue content, and uses the fine-tuned GPT2 model to generate the delexicalized seller message. Finally, the slot filler replaces all placeholder slots with the appropriate information and yields the complete seller response.

3.2.2 End-to-end Model

In this approach, we reduce the bot to a single end-to-end model. We use a seq2seq LSTM-based architecture with attention mechanism to predict the next utterance given a sequence of up to three previous utterances and the item context (embeddings for item category, name, and description).

The model architecture is shown in Figure 3.3. The encoder consists of two stacked layers of tanh-activated LSTM units which extract hidden states from the embeddings of the preceding utterances and pass the extracted hidden states to an attention layer. The attention layer weighs these states as well as the the item's context embeddings and passes them to the decoder. The decoder also consists of two stacked layers of tanh-activated LSTM units. Its role is to receive the weighted states and item context and predict the softmax probability of every token at the output layer. As training data is relatively small, we use GloVe pre-trained embeddings [29] to represent input utterances.

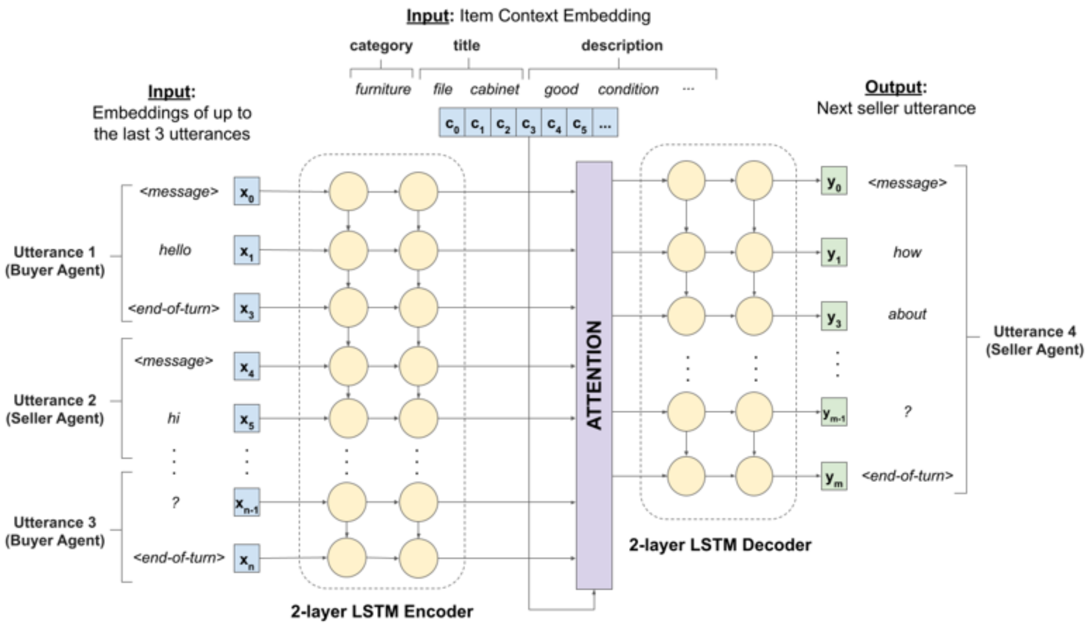

Figure 3.3: Overview of the end-to-end model architecture. The end-to-end model consists of a seq2seq model with an encoder that captures data from previous utterances, an attention layer which learns to assign weights to both encoder states and item context, and a decoder which generates the response utterance.

3.3 Training Method

The end-to-end model must learn which utterances to express to reach a successful negotiation where both price and buyer sentiment are maximized. Similarly, the decoupled model's manager must learn which intents, prices, and sentiments to express. This requires both models to learn an optimal set of parameters θ which would allow them to imitate human sellers while simultaneously finding ways to maximize both sale price and buyer sentiment. Previous work in automated negotiation have approached this mixed task of human imitation and goal-maximization by first training the model to minimize a negative log-likelihood loss L_{MLE} of the ground truth output data y^* and then fine-tuning the model through self-play in order to minimize the reinforcement learning loss L_{RL} and maximize the expected difference between the reward r and a baseline b (b is taken as a moving average of rewards obtained thus far).

$$L_{\text{MLE}}(\theta) = -\sum_x \log p_\theta(y^*|x) \tag{3.1}$$

$$L_{\text{RL}}(\theta) = -\sum_x \log p_\theta(y|x)(r-b) \tag{3.2}$$

A more sample and time efficient alternative to this approach which has seen success in other goal-based generation systems is mixed learning [30] [24]. It uses a loss function L_{mixed} which combines both of the aforementioned losses into one as follows:

$$L_{\text{mixed}} = \gamma L_{\text{RL}} + (1-\gamma)L_{\text{MLE}} \tag{3.3}$$

where γ is a scaling factor used to account for the difference in magnitude between L_{MLE} and L_{RL}.

To ensure that both price and buyer sentiment are jointly maximized, we normalize both and formulate the reward function r as a weighted linear combination of both:

$$r(p, s) = \begin{cases} \alpha\tilde{s} + (1-\alpha)\tilde{p} & \text{agreement} \\ -1 & \text{no agreement} \end{cases} \tag{3.4}$$

$$\tilde{s} = \frac{s - \mu_s}{\sigma_s} \tag{3.5}$$

$$\tilde{p} = \frac{\hat{p} - \mu_{\hat{p}}}{\sigma_{\hat{p}}} \tag{3.6}$$

$$\hat{p} = \frac{p}{p_{\text{seller}}} \tag{3.7}$$

where α is a trade-off parameter that can be tuned, s is a numerical value associated with the final sentiment expressed by the buyer, p is the price agreed upon by both parties, and p_{seller} is the seller's listed price for the item. Note that the mean values μ_s and $\mu_{\hat{p}}$ and the standard deviations σ_s and $\sigma_{\hat{p}}$ are computed based on the observed outcomes in the training set.

3.4 Data pre-processing

Shared pre-processing steps

To ensure proper item representation, we first pre-process the dataset using regular expressions to replace item names, and item conditions with placeholder slots based on the patterns presented in Table 3.2.

We simultaneously also build a knowledge base for each item by performing pattern matching on both the dialogue content and the item context. This lets us form mappings between given item names and item conditions in the item context and a list of possible in-dialogue mentions. An example of such mappings would be between the item name "VIZIO S5451w-C2 5.1 Channel Sound Bar" and the possible references "sound bar", "surround system", and "vizio", and the item condition "mint" and references "mint", "practically new", "flawless", and "excellent". We add to the item name mappings a combination of all possible

Table 3.2: Patterns for extracting item attributes

<table>
<tr><th>Attribute</th><th>Patterns</th></tr>
<tr><td>Name</td><td>how old is (the | this | that | these) ... (you | on | ?), this is a[n] [rare | nice | great | awesome | amazing] ... (with | in | .), the price for (the | this | that | these) ... (is), interested in (my | your | the | this | these | that) ... (? | (i’ve | i have) | (you ’ve | you have), to have this ..., to buy (the | this | that | these) ... (you | i | for), what condition is (the | this | that | these) ... in, it(’s | is) a ... (and it | with | in |), looks like a [rare | nice | great | awesome | amazing] ... , want the ... (i | you | .)</td></tr>
<tr><td>Condition</td><td>in [a[n]] ... condition, [pretty much | practically | brand] new, [barely | never] used</td></tr>
</table>

n-grams of the item name as well as common generic item references, namely: item(s), listing(s), product(s), set(s), and unit(s). We also include the category of the item in the mapping and for items in the housing category, we further add the following generic references: appartment(s), flat(s), house(s), rental(s), home(s), location(s). The generated mappings are then used as ground truth data for evaluation as well as options for the slot filler to insert accurate information.

Meanwhile, to represent price mentions in the end-to-end seq2seq model and in the manager’s seq2seq model, we start by applying min-max normalization such that an agent’s target price is equal to 1 and its bottomline is equal to 0. For buyers, the target price is the buyer target price specified by the negotiation scenario while the bottomline is the listed item price (also specified by the negotiation scenario). For seller agents, the target price is the listed item price while the bottomline is a certain selected portion of that price. For our experiments, we fix the bottomline at 70% of the listed item price. To pre-process the resulting numerical values in a way that is appropriate for use with a seq2seq model, we perform binning according to the approximate normalized value taking into consideration two digits after the decimal point (e.g. normalized values of 0.693 and 0.694 will be represented by the same token: 0.69), then represent each bin by a single token.

Additional pre-processing for decoupled model

For the dialogue manager’s decision model, we associate tokens with each of the 10 intents, 4 non-message actions, and 3 sentiment levels. We then pre-process every dialogue into alternating sequences of intents, prices, and sentiments. An end-of-turn token is again used to indicate a switch in agents. Figure 3.4 shows an example dialogue and the associated training sequence after pre-processing.

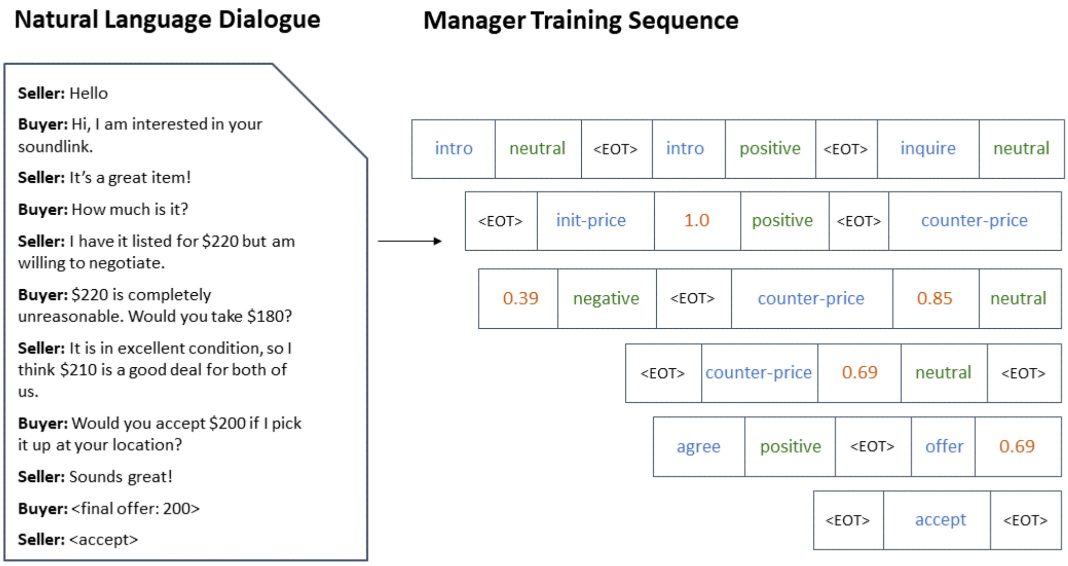

Figure 3.4: Negotiation dialogue being converted into a sequence of tokens appropriate for the manager's seq2seq model. Every message utterance in the dialogue is represented by an intent, price (if any), and sentiment value in the resulting sequence.

For fine-tuning the generator's GPT-2, as the price is directly taken from the manager, we replace all mentions of price by placeholder tokens. Dialogues are then turned into sequences of messages for learning. Each sequence starts with an item name and item condition and then alternates between buyer and seller messages. The associated agent as well as their intent and sentiment are prepended to every message allowing the model to learn to align each message to the preceding intent and sentiment at inference time.

Additional pre-processing for end-to-end model

For the end-to-end model, we associate tokens with each of the 5 actions (message, offer, accept, reject, quit) and build additional vocabulary to accommodate the entirety of the delexicalized dialogue content. For training, we create sequences of utterances from the negotiation dialogues and separate every two subsequent utterances with a special end-of-turn token.

Chapter 4

Experiments & Results

4.1 Dataset

For training and evaluation, we use the CraigslistBargain dataset [12]. This dataset is currently the only dataset publicly available for item sales negotiations that contains rich and diverse language suitable for training a negotiating bot with empathy. It contains 6,682 simulated negotiation dialogues between Amazon Turk Workers over the sale of 1402 item postings from 6 categories (housing, furniture, cars, bikes, phones, and electronics) web-scraped from the Craigslist website. The dataset is split into training, validation, and testing sets as shown in Table 4.1.

Table 4.1: Detailed dataset dialogues split

	Accept	Reject	Quit	Total
Train	4223	344	680	5247
Validation	483	41	73	597
Test	692	52	94	838

Upon inspection of the dialogues, we find that those ending in quit can be detrimental to the learning of the model as they tend to reflect instances were one of the agents became unresponsive, a bug occurred in the system, or one of the two agents made a mistake in the final offer and could not take it back. This was observed on 77 out of 100 randomly selected such dialogues, and thus we chose to filter out entirely all these dialogues. Examples of the evaluated dialogues ending in quit can be seen in Table 4.2.

4.2 Training Procedure

In this section, we detail the training procedure and hyper-parameters used for training both proposed models.

Table 4.2: Examples of extracts from invalid dialogues ending in quit. We can see various observed issues with such dialogues.

Observed Issue	Dialogue
Unresponsive Agent	Seller: there is no way i can go under 300 Buyer: well are there any other perks you can offer at that price? Buyer: hello? are you still there? Could you include delivery? <quit>
Role confused	Buyer: It is the best. so for 5395 you are getting the best Seller: I know that is why I bought it Buyer: omg i got mine confused <quit>
System bug	Buyer: I can't see you cottage. The item up for me is an antique wash bowl. What should we do? Seller: Oh. Well, that's a problem isn't it? My screen shows a cottage rental. Quit? Buyer: I guess I don't know what to do. Seller: Me neither Buyer: <quit>
Mistaken offer	Buyer: <final offer $1475> ... Buyer: ... I am not willing to go over $1850... Seller: You can't make another offer anyway. I can't erase your original offer. I am going to have to quit. Buyer: ok <quit>
Unknown	Buyer: hello Seller: hello <quit>

4.2.1 Decoupled model

For training the seq2seq model of the manager, we use an $n = 300$ dimensional space for representing the embeddings and the hidden states of the encoder and decoder. We then minimize L_{mixed} over 10 epochs using Adagrad with a learning rate of 0.01 and a batch size of 128. For reward computations, we use the VADER compound score [27] of the last buyer message and set $\gamma = 0.5$ as both losses are found to share the same scale of magnitude. As for α, we experiment with different values in the range $[0, 0.1, ..., 0.9, 1]$. At the end of every epoch, we evaluate the loss on the validation set and then save the model with the lowest loss. We find that multiple pareto optimal solutions are possible given different values for α and pick the value 0.6 which maximizes for both price and sentiment and yields the highest reward value on the validation set.

Meanwhile, for the generator, we fine-tune GPT-2 separately on the dialogue message contents from the CraigslistBargain dataset [12]. Using the Adam optimizer, we train for 5 epochs with a batch size of 2 (due to memory limitations) and a learning rate of 5e-4 and observe the loss on the validation set to avoid over-fitting. For our implementation, we use the smallest GPT-2 model which consists of 12 attention heads and 117 million parameters.

4.2.2 End-to-end model

We use a 300-dimensional space to represent words and encoder/decoder hidden states initialized based on the pre-trained GloVe embeddings [29] and use a continuous bag of words approach to learn the embeddings of the item context vector [12]. We train the model using the mixed learning approach outlined in section 3.3 for 10 epochs. We set the learning rate to 0.01, the batch size to 128, γ to 0.5, and experiment with different values of α in the range $[0, 0.1, ..., 0.9, 1]$. At the end of every epoch, we evaluate the loss on the validation set and save the checkpoint with the lowest validation loss. We find that multiple pareto optimal solutions are possible given different values for α and pick the value 0.4 which maximizes for both price and sentiment and yields the highest reward value on the validation set.

We also experiment with training the model using the two-stage training method from previous works:

- For the first stage of the training, we use a learning rate of 0.01, batch size of 128, and train the model for 20 epochs to minimize L_{MLE} using dialogues from the training set [12]. At the end of each epoch, we evaluate the loss on the validation set and choose the checkpoint with the lowest loss.

- In the second stage, we simulate 1500 different dialogues based on the negotiation scenarios present in the training set. We freeze the parameters for the buyer agent to those learned in the first stage, and update the parameters for the seller agent according to Algorithm 1. We experiment with different hyper-parameters and find that a learning rate η of 0.0001, a number of dialogues n_d equal to 1500, and a random sampling decoding temperature of 0.5 lead to good exploration of different utterances and richer dialoguing. We note that, while raising the learning rate and the number of simulated dialogues might lead to improved expected reward, we find that it leads to poorer dialoguing and less generalization capability where the model attempts to exploit the same utterance repeatedly.

Algorithm 1: RL Fine-Tuning Approach

H **Input:** π_θ seller agent policy;
Parameter: η learning rate;
Parameter: n_d # of simulated dialogues;
Initialise baseline reward: $b = 0$;
for $i = 0, 1, ..., n_d$ **do**
  Simulate complete dialogue with buyer agent and generate seller responses $y^{(0)}, y^{(1)}, ..., y^{(T)}$;
  Extract last buyer message sentiment s;
  Extract agreed price p;
  Compute associated reward $R = r(p, s)$;
  Subtract baseline: $G = R - b$;
  Update π_θ parameters using policy gradient $\theta \leftarrow \theta - \eta\, G \sum_{t=0}^{T} \sum_{k=0}^{m} \nabla_\theta \log p_\theta(y_k^{(t)} | y_{0..k-1}^{(t)})$;
  Update baseline reward: $b = b + \frac{G}{i+1}$;
end

4.3 Experimental Setup

For evaluation, we fix the random seed to reduce variance across experiments and simulate conversations based on the 838 scenarios in the test set. For simulating the buyer agent, we use an end-to-end seq2seq model trained in a supervised learning manner to minimize L_{MLE} for the buyer's dialogues and evaluate different seller agent models against it. We study the following models:

- *E2E SL*: An end-to-end model trained only in a supervised learning model (i.e. with only the first stage of the two-stage training) which serves as a baseline for other models. To assess the effect of delexicalization, we train this model on the original dataset which contains mentions of item names and conditions.

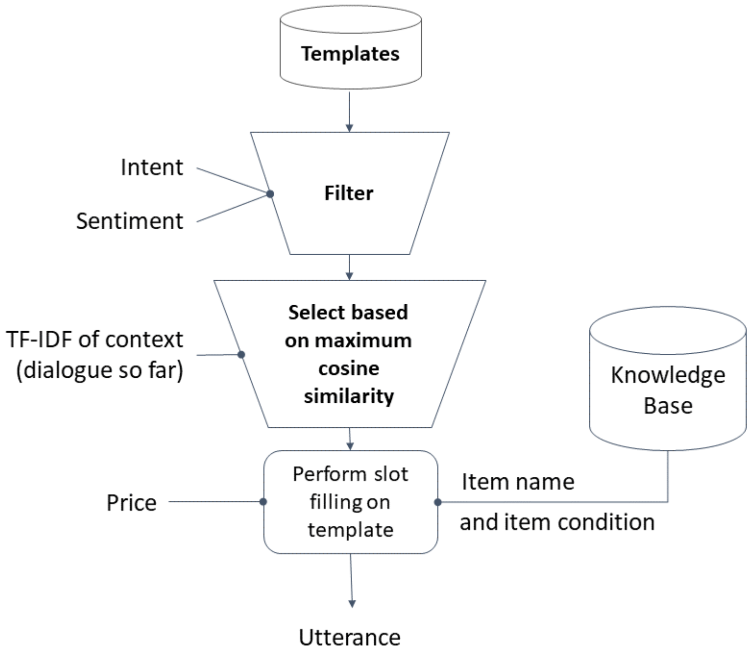

Figure 4.1: Template retrieval system in action

- *E2E RL*: An end-to-end model trained in supervised learning then fine-tuned with reinforcement learning using the proposed reward function.

- *E2E MIXED*: The proposed end-to-end model trained with the mixed learning approach.

- *S2S+TR RL PRICE*: The price-maximizing model proposed by He et al. [12]. Previous work and current state of the art in natural language automated negotiation.

- *S2S+TR MIXED*: A variation of the proposed model which uses template retrieval based on Term Frequency-Inverse Document Frequency (TF-IDF) instead of GPT-2 for utterance generation. Templates are created from the labeled dataset of utterances and selected according to the process outlined in Figure 4.1.

- *S2S+GPT2 MIXED*: The proposed decoupled model trained with the mixed learning approach.

For evaluation and comparison, we compute the following metrics:

- Agreement percentage (**Ag. %**): defined as the ratio of the number of dialogues ending in agreement to the total number of dialogues

- Average price ratio (**Price**): defined as the average value of $\hat{p}$ for all dialogues where agreement was reached

- Average final buyer sentiment (**Sent.**): defined as the average of the VADER compound score for the last buyer message in every dialogue

- Slot filling accuracy (**Slot Acc.**): defined as the ratio of item attribute slots filled correctly to the total number of slots filled in all dialogues. This metric gives us an idea on how well the model is representing the item on sale and whether or not it is providing false information which conflicts with the item description. We focus the evaluation in this work on two item attributes: name and condition.

- Inconsistent seller offers rate (**ISOR**): which we define as the number of the seller's final offers which do not align with the previous discussion divided by the total number of offers made by the seller. A final offer is inconsistent if it has a different value than that of a previous proposition which was made and agreed upon.

Furthermore, we invite 4 human subjects to blindly interact once with 7 selected variations of the discussed models:

- *E2E RL(*$\alpha = 0$*)* which maximizes sale price

- *E2E RL(*$\alpha = 1$*)* which maximizes sentiment alone

- *E2E Mixed(*$\alpha = 0.4$*)* which maximizes both sale price and sentiment

- *S2S+TR RL PRICE* [12] which maximizes price alone

- *S2S+TR Mixed(*$\alpha = 1$*)* which maximizes sentiment alone

- *S2S+TR Mixed(*$\alpha = 0.6$*)* which maximizes both price and sentiment

- *S2S+GPT2 Mixed(*$\alpha = 0.6$*)* which also maximizes both price and sentiment but uses GPT-2 for utterance generation

They negotiate over the sale of the same item and have their dialogues recorded. We then ask 17 different human evaluators to critically rate between 1 (worst) to 5 (best) the dialogues for each model according to the following criteria:

- Empathy: *Do you feel the seller demonstrated empathy towards the buyer's situation in its language?*

- Fluency: *How much fluency in English did the seller express in its utterances? Did the language seem accurate?*

- Appropriateness / Relevance: *Did the responses seem appropriate to the conversation? Were they on-topic?*
- Human-likeness: *Do you think the seller demonstrated reasonable human behavior?*

4.4 Results & Discussion

4.4.1 Comparative analysis

The outcomes from the simulated dialogues for the various seller agent models are shown in Table 4.3. The proposed models for the negotiating bot with empathy are highlighted in grey.

Table 4.3: Summary of performance of different seller models. Proposed models yield good performance on both price and sentiment. They also avoid misrepresentation of items and inconsistent offers.

	Ag. %	**Price**	**Sent.**	**Slot Acc.**	**ISOR**	**ISOR (no rules)**
E2E SL	**99.90%**	0.788	0.161	73.56%	39.70%	N/A
E2E RL α=0	97.90%	0.910	0.198	**100%**	73.75%	N/A
E2E RL α=1	**99.90%**	0.733	**0.445**	**100%**	**-%**	N/A
E2E Mixed α=0.4	94.70%	0.827	0.299	**100%**	53.40%	N/A
S2S+TR RL PRICE	97.60%	**0.999**	0.211	60.50%	99.60%	N/A
S2S+TR Mixed α=1	97.60%	0.757	0.398	**100%**	**-%**	-%
S2S+TR Mixed α=0.6	96.80%	0.809	0.399	**100%**	**0%**	42.1%
S2S+GPT2 Mixed α=0.6	97.40%	0.805	0.398	**100%**	**0%**	42.3%

We observe good agreement rates across all models with two models presenting particularly high values: *E2E SL* and *E2E RL* $\alpha = 1$. For the first model this is explained by the distribution of the data within dataset which heavily discourages rejections while for the later it is explained by its passive behavior where it is open to accept any offer. This passive behavior exhibited by both it and the *S2S+TR Mixed* $\alpha = 1$ comes from ignoring price in the reward function and also leads to them having the lowest values for agreed price ratio. We show an example of this passive behavior in Table 4.4. In it, we can see how these models tend to avoid making offers completely, instead accepting whatever the buyer offers, hence their ISOR of 0 with or without additional rules placed on the model.

Meanwhile, the models trained for maximizing price alone (*E2E RL* $\alpha = 0$ and *S2S+TR RL PRICE*) which correspond to previous work by He et al. [12] obtain the highest values for agreement price ratio but by ignoring buyer sentiment in the reward function they tend to have lower sentiment scores. Tables 4.5 and 4.6 show examples of conversations with both models, we can see a tendency for the models to insist on a single price as well as final offers which aim for higher prices than what was previously discussed in the dialogue.

Table 4.4: Example of passive behavior exhibited by sentiment-maximizing models. The seller bot avoids making offers instead accepting whatever the buyer offers.

Buyer	Hello
Seller	Hi
Buyer	I'm interested in the item but i can't do \$65, would \$40 work?
Seller	**Sure!**
Buyer	Awesome!
Seller	**Great!**
Buyer	<final offer: \$40>
Seller	**<accept>**

Table 4.5: Example of price insistence in price-maximizing models where the seller bot repeats the same offer three times.

Buyer	Hello there
Seller	Hi, are you interested in my Ubiquiti Networks UAP-AC-LITE
Buyer	Yes I am
Seller	Great, I'm asking **65** for it.
Buyer	That's too high.
Seller	**Can you do 65**?
Buyer	No I was thinking more like 60?
Seller	I'd really like **at least 65**
Buyer	Best I can do is 60
Seller	I can do 62 with the extended warranty
Buyer	Okay that sounds great!
Seller	<final offer: \$62>
Buyer	<accept>

Table 4.6: Example of inconsistency in price-maximizing models. Both buyer and seller agree on a price of 184 but the seller sends a final offer of 200.

Buyer	Hello
Seller	Hello there.
Buyer	I am interested in the item you have for sale.
Seller	It's new. I'm asking **200** for it
Buyer	Would you be willing to lower the price?
Seller	How about **200** it works fantastic
Buyer	Seems like a nice piece. Would you take 150?
Seller	**184 and we have a deal!** I'll get it ready for you.
Buyer	Great!
Seller	<final offer: **$200**>

Table 4.7: Comparison of the two generation models for the same action sequence. We observe richer and more relevant responses from the model which employs GPT2.

	S2S+TR Mixed α=0.6	**S2S+GPT2 Mixed α=0.6**
Buyer	Hi there. How are you today?	
Seller	Hello	**Doing well. How are you?**
Buyer	I am interested in the router you have for sale	
Seller	**Great! Hope you are well.**	Great! It is in like new condition and it is absolutely perfect!
Buyer	How does $50 sound?	
Seller	It retails for over 65. It's already a pretty deep discount	**Well I am asking 65 but am willing to negotiate an offer. How does that sound?**
Buyer	Okay then how about 55?	
Seller	Can you go to 58?	How about 58?
Buyer	Sure!	

Finally, the models trained for jointly maximizing both price and sentiment obtain better results on sentiment than models trained for maximizing price and better results on price compared to models which focus only on improved sentiment. *S2S+TR Mixed* $\alpha = 0.6$ and *S2S+GPT2 Mixed* $\alpha = 0.6$ obtain roughly the same outcomes. This is expected as they both share the same decision making model which recommend the same prices, intents, and sentiments to express. The only difference is in the quality of the generated response. We show a side-by-side comparison of generated utterances for the same sequence of manager recommendations in Table 4.7. We observe richer more relevant responses from the GPT2 model, this is expected as it is a significantly larger model with pre-training allowing it to better capture dialogue history as well as generate novel responses.

In terms of slot accuracy, we observe that all models which make use of the proposed delexicalization and slot filling scheme have full accuracy on item representation for name and condition while the baseline model ($E2ESL$) and the previous work's model ($S2S + TRRLPRICE$) which take no special steps to ensure this accuracy have a respective scores of 73.56% and 60.50% indicating the common presence of misleading item information.

We compare the ISOR of the decoupled model with and without the enforcing of the additional rules mentioned in Section 3.2.2. With these rules in place, we see no inconsistent offers being made by the seller agent at all. However, even without rules, we see an improved ISOR compared to previous work. This can be attributed to the mixed learning approach which prevents the model from significantly deviating from human behavior and keeps the generation of offers more closely in line with what would be expected from supervised learning.

4.4.2 Human evaluation

Table 4.8: Summary of human evaluation scores for different seller models. We observe improved empathy, fluency, relevance, and human-likeness from the proposed models.

	Empathy	**Fluency**	**Relevance**	**Human-likeness**
E2E RL α**=0**	3.35	3.68	3.05	3.16
E2E RL α**=1**	3.82	3.71	3.18	2.89
E2E Mixed α**=0.4**	4.11	3.82	3.88	3.82
S2S+TR RL PRICE	2.41	2.94	2.18	2.24
S2S+TR Mixed α**=1**	3.29	2.88	2.29	2.82
S2S+TR Mixed α**=0.6**	3.47	3.12	3.18	3.29
S2S+GPT2 Mixed α**=0.6**	3.71	4.24	4.06	3.88

We present a summary of the results from the human evaluation in Table 4.8.

We start by comparing the end-to-end models together, we find that they share similar fluency ratings but that the proposed model distinguishes itself on empathy, relevance, and human-likeness. For the sentiment maximizing model, the lower relevance and human-likeness ratings are attributed to its passive nature which leads to it having short generic approving responses which while grammatically correct do not reflect a deeper understanding of the dialogue state nor do they reflect human-like behavior. This disconnection from dialogue state and the other agent's situation also lead to it having lower score on empathy. Meanwhile, for the price-maximizing model, the inconsistency in the final offer leads to an overall negative perception of the interaction as a whole. While the model does offer more human-likeness in its negotiation prowess, the fact that it makes offers which are not aligned with the conversation and that it insists on the same price offers leads to it losing out on both relevance and empathy. Finally, the proposed model which maximizes both price and sentiment is seen to be most human-like as it tends to avoid the issues mentioned earlier and provides a balance between a passive attitude and an insistent negotiation strategy.

As for the decoupled models, we see a similar behavior to that of end-to-end models. However, we note that the use template retrieval tends to lead to lower relevance, fluency, and human-likeness. The loss in fluency is not attributed to the template retrieval system itself but rather to the quality of the templates used by it. As the templates are not manually checked, they tend to present grammar and spelling errors. Furthermore, we find that TF-IDF is not enough to fully capture the current dialogue state for the generator, particularly in the novel dialogue situations like those seen during human testing. We show an example dialogue from one of the human testers' negotiations with the *S2S+TR Mixed* model which yields problematic responses and compare it to the dialogue of that same human tester with the *S2S+GPT2 Mixed* model in Table 4.9.

Table 4.9: Comparison of the two generation models showing improved fluency and relevance for the GPT2-based generator.

S2S+TR **Mixed α=0.6**	**S2S+GPT2** **Mixed α=0.6**
Buyer: Hallo Seller: Hello yes its a wonderful item Buyer: Is the product new? Seller: Never **useed** Buyer: All i can do is $50 Seller: I would like get 54 for it. It's brand new. Buyer: How about 51? Seller: You've got a deal Buyer: <final offer: $51> Seller: <accept>	Buyer: Hallo Seller: Hi. How do you like my item? Buyer: Is the product new? Seller: **Yes it is, it is fairly new.** Buyer: How much are you asking? Seller: I am asking $65. Buyer: All I can do is $50. Seller: **I am sorry, how about $54**. **Would you be willing to go up to $54?** Buyer: <final offer: $54> Seller: <accept>

4.4.3 Additional Experiments

In the goal of gaining further insight into the various components of our model, we perform multiple additional experiments.

Choice of reward function

We first investigate the effect of the choice of the reward function and the parameter α on the average outcomes of the resulting model. We compare maximizing the linear combination of normalized sentiment and price shown in eq. 3.4 to maximizing their weighted minimum or in other words the following reward function:

$$r(p, s) = \begin{cases} \min(\alpha\tilde{s}, (1-\alpha)\tilde{p}) & \text{agreement} \\ -1 & \text{no agreement} \end{cases} \tag{4.1}$$

We test training the same end-to-end model with each of the two reward functions for different values of α. We then evaluate against the buyer model by simulating dialogues from the validation set's scenarios. We capture the average agreed price (price) and the average final buyer sentiment (sentiment) on every model, then plot the points on a 2-dimensional space, in order to visualize the Pareto frontier. The results are shown in Figure 4.2.

We find similar performance between both reward functions. However, the linear combination function (sum) is seen to yield a nice convex shape for the Pareto front where lower values of α lead, on average, to favor price over sentiment and higher values of α lead to favor sentiment over price.

Choice of sentiment representation

We consider expanding the scope of s beyond only the last buyer message sentiment score and instead computing it as an exponential moving average (EMA) of all sentiments expressed by the buyer. In other words, given a sequence of buyer sentiments extracted from buyer messages from the beginning until the end of a dialogue $s^{(0)}, s^{(1)}, ..., s^{(T)}$, we compute the final sentiment score recursively as:

$$\text{EMA}(s) = \lambda s^{(T)} + (1-\lambda)\text{EMA}(s^{(T-1)}) \tag{4.2}$$

We fix α to 0.4 and compare the performance of the end-to-end model trained with no sentiment history consideration (E2E RL) in reward to one trained with sentiment history consideration (E2E RL EMA). The sentiments nearing the end of the dialogue play a larger role so we experiment with values of λ in the range of 0.7 to 0.9. We find similar results on price and sentiment for when compared to the extreme case of $\lambda = 1$, or in other words, the previously proposed case of utilizing only the last buyer sentiment.

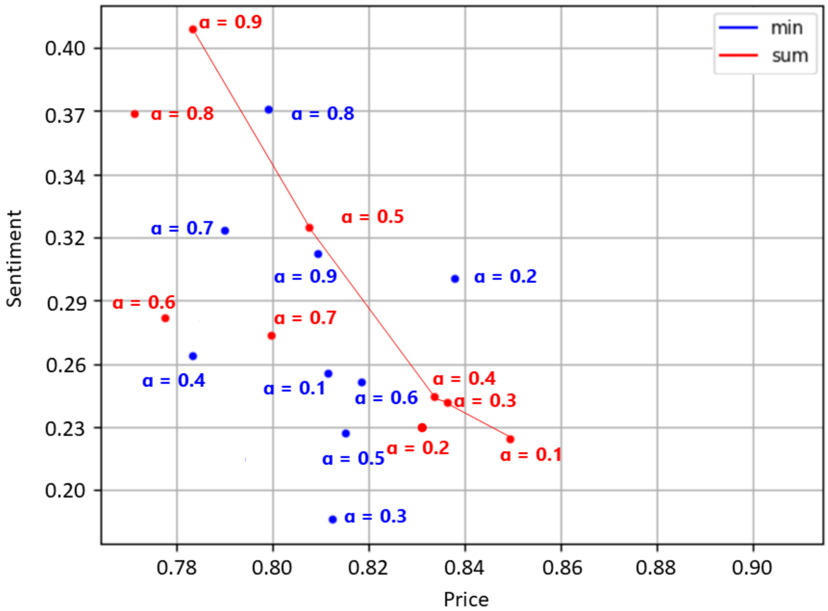

Figure 4.2: 2D visualization of outcomes for the end-to-end model trained with 2 different reward functions. We observe more predictable performance from the linear combination function (sum).

Mixed learning vs traditional approach

We also compare the performance of the model when trained in Supervised Learning (SL), in Supervised Learning followed by Reinforcement Learning (RL), and in the mixed learning approach. We use the following hyper-parameters for each training approach:

- SL: learning rate = 0.01, batch size = 128, epochs = 10
- RL: $\alpha = 0.4$, learning rate = 0.0001, $n_d = 1500$
- Mixed: $\alpha = 0.4$, $\gamma = 0.5$, learning rate = 0.01, batch size = 128, epochs = 10

We then simulate dialogues with the buyer agent over the scenarios in the validation set. The results are extracted and shown in the boxplot of Figures 4.3 and 4.4. We also include the human negotiation data for comparison. We see comparable performance between RL and mixed approaches with the latter outperforming the former on median sentiment and the former outperforming the latter on median price ratio. We note that both outperform the baseline SL model and reach improved buyer sentiment and price when compared to the human negotiations around the same scenarios.

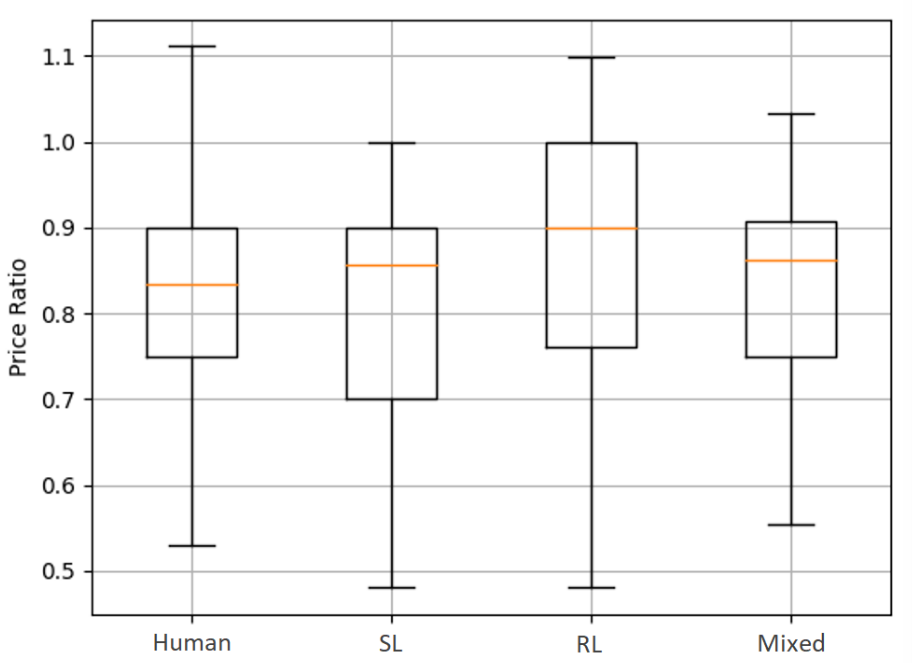

Figure 4.3: Comparison of resulting prices for SL, RL, and mixed model. The RL model tends to negotiate higher prices.

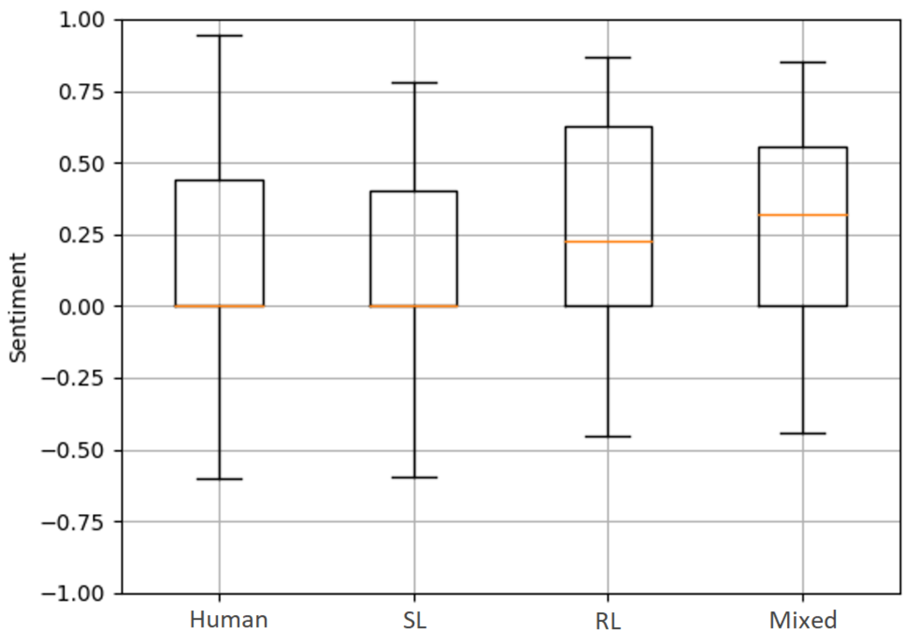

Figure 4.4: Comparison of resulting buyer sentiment for SL, RL, and mixed model. The mixed model obtains better median value on final buyer sentiment.

We also compare the time taken to train each model in Table 4.10. We can see that mixed training has training times which are significantly superior to the two-stage training used in this previous work. This is mainly due to not requiring the simulation of dialogues between it and a pre-trained buyer agent.

Table 4.10: Comparison of training time for two-stage training and mixed training. Mixed training shows significant improvements in training time.

Approach	Training Time	Improvement %
Two-stage training	5,733 seconds	-
Mixed training	1,036 seconds	557%

Parser intent detection evaluation

Finally, we evaluate the intent extraction capabilities of the parser proposed by He et al. [12] and the impact of the proposed added rules and patterns on its performance. As the rules and patterns are designed to match the data in the training set, for evaluation, we randomly select 20 dialogues (123 message utterances) from the dev set and label them. We then compare the predictions of both the original and modified parser to the assigned labels. We show in Table 4.11, a comparison of the performance of the updated parser to that of the original parser on some utterances which were previously labeled as unknown.

Table 4.11: Examples of improved intent extraction after updating the parser

	Original Parser	Updated Parser	Assigned Label
Good, so you're interested in the typewriter?	unknown	intro	intro
ok thanks all yours make apointment for pick up	unknown	agree	agree
that is really low, but I need to get rid of it, so I will take that.	unknown	agree	agree
done, i'm submitting the price now	unknown	agree	agree

We evaluate the accuracy of the intent detection of the parser pre and post modifications. We find that the original parser has an acccuracy of around 81.8%, while the refined parser has an accuracy of 94.7%. This is in large part due to a large proportion of "agree" and "intro" not being detected as such with the older set of rules. For comparison, we show in Figure 4.5 the distribution of the intents in the dataset pre and post parser changes.

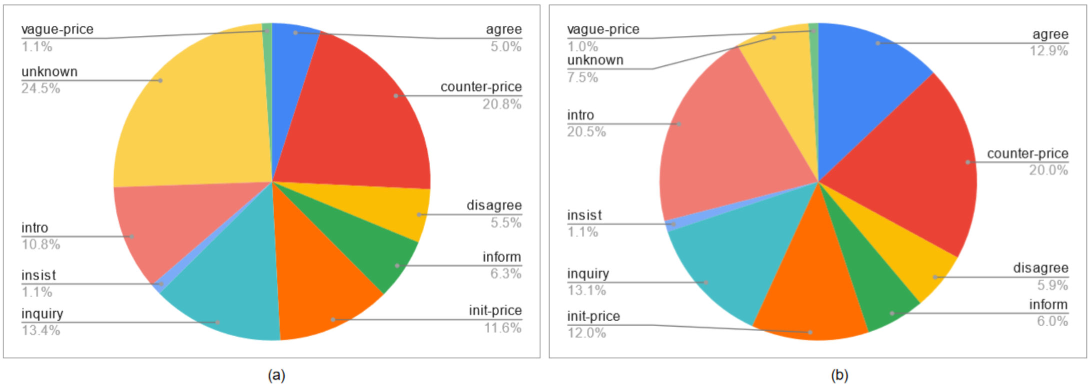

Figure 4.5: Distributions of intents pre (a) and post (b) parser refinements. We observe a significant decrease in unknown utterances with increased presence of correctly identified agree and intro intents.

Chapter 5

Conclusion

In this thesis, we targeted the design and implementation of the first automated sales negotiating bot with empathy. We sought to address the following challenges: (1) Integrate empathy into the automated negotiation process, (2) Improve on the bargaining skills of state-of-the art models by ensuring proper item representations and consistency in offer making, (3) Propose an evaluation framework for negotiating bots with empathy.

We studied two models: an end-to-end model which learns directly from the data to generate appropriate responses and a decoupled model with 3 components: a rule-based parser which extracts the act and sentiment from previous utterances, a seq2seq manager which recommend appropriate response acts and sentiments, and a fine-tuned GPT-2 generator which transforms these recommendations into complete responses. To train both models to jointly maximize both sale price and buyer sentiment while minimizing inconsistency in offers we used a mixed learning approach which includes a reward function encouraging both objectives. Meanwhile, to improve item representation, we built knowledge bases for every item and used delexicalization in combination with slot filling. We set up an evaluation framework which includes automated metrics as well as human evaluation and then used it to evaluate the two proposed models and compare them to previous state of the art as well as baselines and single objective variations. We found that our models presented improved buyer sentiment, item representation accuracy, consistency in offers, empathy, fluency, appropriateness, and human-likeness when compared to others.

Future work includes (1) extending the current framework to integrate empathy into multi-issue negotiations where agents negotiate simultaneously over multiple issues beyond just item price with the ability of adding further issues during negotiation (e.g. delivery, warranty, promo codes, ...), (2) designing automated metrics for measuring empathy in bot responses, and (3) a deeper study on the impact of negotiating bots and potential adoption in real-world businesses.

Bibliography

[1] R. Fisher, W. L. Ury, and B. Patton, *Getting to yes: Negotiating agreement without giving in*. Penguin, 2011.

[2] T. Baarslag, M. Kaisers, E. Gerding, C. M. Jonker, and J. Gratch, "When will negotiation agents be able to represent us? the challenges and opportunities for autonomous negotiators," *International Joint Conferences on Artificial Intelligence*, 2017.

[3] J. Rosenschein and G. Zlotkin, "Consenting agents: Designing conventions for automated negotiation," *AI Magazine*, vol. 15, pp. 29–46, 01 1994.

[4] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge, "Automated negotiation: prospects, methods and challenges," *International Journal of Group Decision and Negotiation*, vol. 10, no. 2, pp. 199–215, 2001.

[5] J.-F. Tang and X.-L. Xu, "An adaptive model of service composition based on policy driven and multi-agent negotiation," in *2006 International Conference on Machine Learning and Cybernetics*, pp. 113–118, IEEE, 2006.

[6] S. Chaharsooghi and Z. Taheri, "An agent based negotiation mechanism considering suppliers bidding in an automated business transaction," in *2010 IEEE International Conference on Industrial Engineering and Engineering Management*, pp. 857–861, IEEE, 2010.

[7] M. J. Hussain, "Parallel processing and intercommunication of multiple negotiation processes in e-commerce for optimal business," in *2014 Fourth International Conference on Communication Systems and Network Technologies*, pp. 583–586, IEEE, 2014.

[8] G. Koley and S. Rao, "Adaptive human-agent multi-issue bilateral negotiation using the thomas-kilmann conflict mode instrument," in *2018 IEEE/ACM 22nd International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, pp. 1–5, IEEE, 2018.

[9] D. Zhai, Y. Wu, J. Lu, and F. Yan, "A fuzzy negotiation model with genetic algorithms," in *Integration and Innovation Orient to E-Society Volume 1*, pp. 35–43, Springer, 2007.

[10] A. Diamah, M. Mohammadian, and B. Balachandran, "Fuzzy utility and inference system for bilateral negotiation," in *2012 2nd International Conference on Uncertainty Reasoning and Knowledge Engineering*, pp. 115–118, IEEE, 2012.

[11] M. Lewis, D. Yarats, Y. N. Dauphin, D. Parikh, and D. Batra, "Deal or no deal? end-to-end learning for negotiation dialogues," *arXiv preprint arXiv:1706.05125*, 2017.

[12] H. He, D. Chen, A. Balakrishnan, and P. Liang, "Decoupling strategy and generation in negotiation dialogues," *arXiv preprint arXiv:1808.09637*, 2018.

[13] A. Parvaneh, E. Abbasnejad, Q. Wu, and J. Shi, "Show, price and negotiate: A hierarchical attention recurrent visual negotiator," *arXiv preprint arXiv:1905.03721*, 2019.

[14] K. W. Thomas, "Thomas-kilmann conflict mode," *TKI Profile and Interpretive Report*, pp. 1–11, 2008.

[15] M. Cheng, W. Wei, and C.-J. Hsieh, "Evaluating and enhancing the robustness of dialogue systems: A case study on a negotiation agent," 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)*, pp. 3325–3335, 2019.

[16] M. Ochs, D. Sadek, and C. Pelachaud, "A formal model of emotions for an empathic rational dialog agent," *Autonomous Agents and Multi-Agent Systems*, vol. 24, no. 3, pp. 410–440, 2012.

[17] O. N. Yalcin and S. DiPaola, "A computational model of empathy for interactive agents," *Biologically inspired cognitive architectures*, vol. 26, pp. 20–25, 2018.

[18] P. Fung, A. Dey, F. B. Siddique, R. Lin, Y. Yang, Y. Wan, and H. Y. R. Chan, "Zara the supergirl: An empathetic personality recognition system," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pp. 87–91, 2016.

[19] A. Ghandeharioun, D. McDuff, M. Czerwinski, and K. Rowan, "Emma: An emotion-aware wellbeing chatbot," in *2019 8th International Conference on Affective Computing and Intelligent Interaction (ACII)*, pp. 1–7, IEEE, 2019.

[20] O. N. Yalcin, "Empathy framework for embodied conversational agents," *Cognitive Systems Research*, vol. 59, pp. 123–132, 2020.

[21] X. Sun, X. Chen, Z. Pei, and F. Ren, "Emotional human machine conversation generation based on seqgan," in *2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia)*, pp. 1–6, IEEE, 2018.

[22] H. Rashkin, E. M. Smith, M. Li, and Y.-L. Boureau, "Towards empathetic open-domain conversation models: A new benchmark and dataset," *arXiv preprint arXiv:1811.00207*, 2018.

[23] Z. Lin, P. Xu, G. I. Winata, F. B. Siddique, Z. Liu, J. Shin, and P. Fung, "Caire: An end-to-end empathetic chatbot.," in *AAAI*, pp. 13622–13623, 2020.

[24] J. Shin, P. Xu, A. Madotto, and P. Fung, "Happybot: Generating empathetic dialogue responses by improving user experience look-ahead," *arXiv preprint arXiv:1906.08487*, 2019.

[25] N. Asghar, I. Kobyzev, J. Hoey, P. Poupart, and M. B. Sheikh, "Generating emotionally aligned responses in dialogues using affect control theory," *arXiv preprint arXiv:2003.03645*, 2020.

[26] L. Zhou, J. Gao, D. Li, and H.-Y. Shum, "The design and implementation of xiaoice, an empathetic social chatbot," *Computational Linguistics*, vol. 46, no. 1, pp. 53–93, 2020.

[27] C. Gilbert and E. Hutto, "Vader: A parsimonious rule-based model for sentiment analysis of social media text," in *Eighth International Conference on Weblogs and Social Media (ICWSM-14). Available at (20/04/16) http://comp. social. gatech. edu/papers/icwsm14. vader. hutto. pdf*, vol. 81, p. 82, 2014.

[28] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," 2019.

[29] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[30] R. Paulus, C. Xiong, and R. Socher, "A deep reinforced model for abstractive summarization," *arXiv preprint arXiv:1705.04304*, 2017.