Agreement Tracking for Multi-Issue Negotiation Dialogues

Anonymous ACL submission

Abstract

Automated negotiation support systems seek to help human negotiators reach more favorable outcomes. When supporting multi-issue negotiations, it is crucial for these support systems to accurately track the agreements reached by the participants in real-time (e.g., an employer and a candidate negotiating over multiple issues such as salary, hours, and promotions before finalizing a job offer). However, existing task formulations are either geared towards differing dialogue paradigms (e.g., dialogue state tracking is aimed at task-oriented dialogues) or generate a single, unstructured output at the end of the dialogue (e.g., meeting summarization). We introduce the novel task of agreement tracking for two-party multi-issue negotiations, in which the goal is to continuously track the agreements over a *structured* state space. Due to the absence of large-scale corpora with turn-level annotations in this domain, we propose a simple, but strong initial baseline for our task based on transfer-learning a T5 model from the dialogue state tracking task on the MultiWOZ 2.4 corpus of task-oriented dialogues. Additionally, we also study the sample-efficiency of our approach by running experiments on smaller fractions of the training data. Our results demonstrate the challenging nature of the agreement tracking task and the need for more data-efficient approaches to solve it.

1 Introduction

Negotiation dialogues are an ubiquitous form of dialogue in our lives and tend to occur in both adversarial and collaborative contexts. However, a long line of foundational research in psychology and business has established that in general, humans tend to be poor negotiators who often fail to maximize favorable outcomes. Consequently, developing capabilities to build automated systems that can support human negotiators has been an active area of research (Prakken, 2006; Wang et al., 2019).

We focus specifically on multi-issue negotiation[1] dialogues, in which the participants negotiate over more than one issue, e.g., salary and hours for a job, or location and time for a meeting. These type of dialogues arise in a wide range of real-life situations that vary widely along the adversarial-cooperative spectrum. Most job negotiations, for example, can be thought of as falling on the adversarial end of the spectrum, since the two parties often have opposing objectives. On the other hand, planning-based meetings, in which participants work towards arriving at a shared plan to achieve a common goal can be characterized as being more cooperative. Regardless of the specific real-world context in which the multi-issue negotiation takes place, the ability to track agreements in real-time is a mission-critical capability for any system aiming to effectively support the participants.

While previous work has looked at related tasks such as dialogue summarization and action item generation at the end of a dialogue, our work is the first to investigate the task of agreement tracking for multi-issue negotiation dialogues over a structured state space (commonly referred to as an "ontology"). Agreement tracking is a state tracking task that requires reasoning over multiple turns in a dialogue. However, it is currently not at all obvious as to how one can leverage existing task formulations, model designs and training objectives for agreement tracking without manually collecting and labelling a substantial amount of in-domain data. We demonstrate this gap in existing methods by reviewing existing work on three closely-related tasks: building negotiation dialogue agents, meeting summarization, and dialogue state tracking for task-oriented dialogues[2].

[1] This type of negotiation is also sometimes referred to as "integrative" negotiation (Zhan et al., 2022).

[2] We will hereafter refer to dialogue state tracking for task-oriented dialogues as just "dialogue state tracking" in line with how the term is commonly used in modern dialogue systems literature.

1. **Negotiation Dialogue Agents:** Research on automated negotiation systems has focused on modeling strategic aspects, such as undervaluing or appealing to the opposite party (Zhou et al., 2019; He et al., 2018; Keizer et al., 2017). However, this is orthogonal to the problem of tracking agreements. End-to-end negotiators that generate responses directly (He et al., 2017), on the other hand, do not generate intermediate structured representations of the dialogue state. The few works that have studied language understanding in this context have done so only at the level of a single utterance (Chawla et al., 2021; Yamaguchi et al., 2021; Frampton et al., 2009); agreement tracking requires reasoning across more than one turn.

2. **Meeting Summarization:** Since the release of the AMI Meeting Corpus (Mccowan et al., 2005), there has been a slew of research on meeting and dialogue summarization (**?**Wang and Cardie, 2013; Rennard et al., 2022; Kryscinski et al., 2020; Liu et al., 2019). However, these approaches differ from agreement tracking in two key aspects. First, summarization is performed only at the end of a dialogue, whereas tracking takes place continuously. Second, summarization focuses on generating a human-readable output of the key decisions made during the meetings, while our task aims to generate a structured representation of the agreements over a fixed ontology for consumption by downstream modules.

3. **Dialogue State Tracking:** Dialogue state tracking has been a long-standing task in task-oriented dialogue systems literature (Williams et al., 2016; Jacqmin et al., 2022; Zhao et al., 2021; Rastogi et al., 2020). Although there has been extensive work in recent years to improve the zero-shot (Lin et al., 2021b; Campagna et al., 2020) and few-shot (Wu et al., 2019) generalization ability of task-oriented dialogue state tracking (DST) models to unseen domains, they are still limited to the paradigm of form-filling dialogues (e.g., restaurant reservation and hotel booking). Hence, the question of "How well can state-of-the-art DST techniques be leveraged to carry out state tracking for vastly differing dialogue paradigms (such as negotiation)?" still remains open-ended.

We offer a solid starting point for future modeling efforts for agreement tracking by introducing a transfer-learning approach based on pre-training a T5 model (Raffel et al., 2020) on the dialogue state tracking task over the MultiWOZ 2.4 corpus. We follow up this step with fine-tuning the model over the agreement tracking task on the GPT-NEGOCHAT corpus, which we introduce in Section 2. Our proposed approach outperforms a T5 model that is fine-tuned only on GPT-NEGOCHAT. Finally, we also investigate the sample-efficiency of our proposed model in low-resource settings by experimenting with various fractional splits of our training data. All our models are evaluated using Joint Slot Accuracy.

The rest of the paper is organized as follows. Section 2 describes the creation and characteristics of GPT-NEGOCHAT, the dataset used for our experiments. Section 3 formally defines the task of agreement tracking, the tokenization scheme, the training objective, and the evaluation metrics used in this study. Section 4 describes our experimental procedure and presents our results. Section 5 presents the related work. Finally, Section 6 talks about the limitations of our work.

2 GPT-Negochat: A Multi-Issue Negotiation Dialogue Corpus

Our choice of corpus is dictated by two key selection criteria. First, we are interested specifically in negotiation dialogues taking place over more than one issue. Second, for reliably evaluating our model, the corpus needs to contain ground-truth annotations for agreements at every dialogue turn. We find two publicly-available corpora satisfying both these criteria: the NEGOCHAT corpus (Konovalov et al., 2016) and the METALOGUE corpus for multi-issue bargaining dialogues (Petukhova et al., 2016). However, we conduct experiments over only the NEGOCHAT corpus as METALOGUE is behind a paywall. NEGOCHAT, on the other hand, is openly accessible and does not impose any copyright restrictions its usage.

The NEGOCHAT corpus (Konovalov et al., 2016) contains 105 crowd-sourced dialogues (1484 utterances) between an Employer and a Candidate negotiating over issues surrounding a job offer such as salary, role, and working hours (the complete ontology of the NEGOCHAT corpus is outlined in

Table 1). While the Employer-side of the conversations is supplied by human participants on Amazon M-Turk[3], an automated agent plays the role of the Candidate. A second human "wizard" is responsible for acting as a live Natural Language Understanding (NLU) module that parses the Employers' utterances into a structured semantic representation. This parsed input is then processed by an automated dialogue manager, which generates a response using a template-based NLG module.

Slot Type	Possible Values
Working Hours	8 hours, 9 hours, 10 hours
Pension Fund	10%, 20%
Job Description	Programmer, Team Manager, Project Manager
Promotion Possibilities	Slow promotion track, Fast promotion track
Salary	90k USD, 60k USD, 120k USD
Leased Car	With leased car, Without leased car, No agreement

Table 1: Ontology of the GPT-Negochat corpus.

Due to the use of a simple, template-based NLG module, the Candidate-side utterances in the NEGOCHAT corpus suffers from a low amount of lexical diversity (see Table 2). To improve the linguistic richness, we create an updated version of the NEGOCHAT corpus with more natural-sounding utterances by utilizing GPT-3 (Brown et al., 2020) to rephrase the original utterances. The GPT-3 prompt used to generate the rephrased utterances is shown in Appendix A. Following this step, we then manually compare each rephrased utterance with the original utterance to verify that there is no change in the meaning of the utterances.

A side-by-side comparison of a dialogue excerpt with the original utterances from NEGOCHAT and the rephrased utterances from GPT-NEGOCHAT is shown in Table 2. A larger set of side-by-side examples is listed in Appendix D. We name this revised version of the NEGOCHAT corpus as "GPT-NEGOCHAT". We also make GPT-NEGOCHAT publicly available on GitHub for use by the broader research community[4].

3 Preliminaries

In this section, we formally define notations as well as the input and output representations that we use for our model.

3.1 Notations

We consider a negotiation dialogue between two participants taking alternating turns with utterances $\{T_1, \ldots, T_N\}$. To ensure that the dialogue follows a strict alternating pattern of utterances, we combine any consecutive utterances made by the same participant into a single utterance. The negotiation takes place over a predefined set of issues, which, in line with terminology used for DST, we will refer to as the domain's "ontology". We also follow the DST framework by representing the issues in our ontology as a set of M slots, $\{s_1, \ldots, s_M\}$. At each dialogue turn t, we define the agreement state A_t as a slot-value mapping between each issue in the ontology and the corresponding value that both participants have agreed upon for it. We denote this slot-value relationship in the agreement state as $A_t(s_j) = v$ (we set v to ϵ if no agreement has been reached so far).

Finally, we associate each utterance T_t with a list of (one or more) dialogue acts, $D_t = [d_t^1, \ldots, d_t^{|d_t|}]$, where d_t^i is one of $\{\text{OFFER}, \text{ACCEPT}, \text{REJECT}, \text{OTHER}\}$. We make use of these dialogue acts for our rule-based algorithm later in Section 4.2. We briefly elaborate on each dialogue act below:

1. OFFER: When making an offer, a participant puts forth one or more slot-value pairs for consideration by the other participant. We formally represent an offer as a list of these key-value pairs: $O(\{(S_1, v_1), \ldots, (S_{n_i}, v_{n_i})\})$. We constrain each issue to a single slot value.

2. ACCEPT and REJECT: While accepting or rejecting an offer, a participant can do so either partially or completely. As an example of partial acceptance, if the Candidate demands an 8-hour workday with a pension of 20%, the Employer might choose to accept the 8-hour workday, but not the 20% pension.

3. OTHER: We put all types of utterances other dialogue acts (e.g., greetings) under the OTHER label as they have no direct effect on the agreement state.

3.2 Representing Dialogue Context

Since the introduction of Transformer-based (Vaswani et al., 2017) generative language models, the common practice for representing dialogue

[3] https://www.mturk.com/

[4] Hidden for review

Speaker	Original Utterance	Rephrased Utterance
Candidate	I would like a position of project manager	I'm interested in a position as a project manager.
Employer	are you sure you wouldnt rather be a programmer?	Are you sure that's the job you're looking for? Wouldn't you prefer to be a programmer?
Candidate	I refuse programmer position. I am expecting a position of project manager	I'm sorry, but I'm not interested in the programmer position. I'm looking for a project manager role instead.
Employer	what about Quality assurance?	What about a Quality Assurance role?
Candidate	I reject qa position. I would like a position of project manager	No, thank you. I'm only interested in a project manager position.

Table 2: An excerpt from a dialogue from the NEGOCHAT corpus alongside its rephrased counterpart from GPT-NEGOCHAT.

Conversation	Agreement State
E: No company car included?	No agreements
C: Right, no car. Let's move on. I was hoping for a pension of 20%.	**Company Car**: No
E: If you work 10 hours, I can offer you a 20% pension - what do you think?	**Company Car**: No
C: No thanks. I'm expecting an 8-hour workday and I want a 10% pension	**Company Car**: No
E: How about a salary of 60K if you agree to the 10% pension?	**Company Car**: No
C: I'm sorry, but I'm looking for a salary of 120,000 and a pension of 20%.	**Company Car**: No
E: What about offering you a fast promotion tract with a 90k salary?	**Company Car**: No
C: No, I'm afraid that won't work for me.	**Company Car**: No
E: Would you be comfortable with a salary of 60 or 90k?	**Company Car**: No
C: 90,000 sounds good to me. Is there anything else we need to discuss?	**Company Car**: No **Salary:** 90,000

Table 3: An excerpt of a dialogue from GPT-NEGOCHAT with turn-level agreement annotations. **C** and **E** stand for Candidate and Employer respectively.

context for dialogue tasks involves concatenating the utterances (potentially separated by some delimiters) within a context window. Dialogue state tracking has also been no exception to this trend.

Full-history based DST models consider as their context the entire dialogue from the beginning up to a given turn (Hosseini-Asl et al., 2020; Feng et al., 2021b; Peng et al., 2021). However, as the number of utterances increases beyond a certain point, they tend to struggle to remember information. Recursive approaches to DST attempt to combat this issue by considering only the utterances within a small window (typically a fixed value, between one to four utterances) as their dialogue context (Lin et al., 2020; Budzianowski and Vulić, 2019; Lei et al., 2018). Rather than predicting the entire dialogue state from scratch at every turn, these approaches use the previously predicted dialogue state as their starting point. For our experiments, we choose a recursive-based approach since the average turn-length of GPT-NEGOCHAT is 34.27, which is significantly higher than that of most task-oriented dialogue corpora.

3.3 Levenshtein Belief Spans

Recent DST approaches are formulated as a conditional text generation problem, with the expected output being the belief spans corresponding to the updated dialogue state. Belief spans offer greater flexibility compared to older classification-based approaches as they can be easily extended to support new slot types, as well as new values for existing slot types, without requiring a complete retraining of the model.

Our tokenization scheme for the agreement tracking task is based on the concept of *Levenshtein Belief Spans*, (or *Lev*s for short), which was proposed by Lin et al. (2020). A *Lev* differs from a traditional belief span in that it only consists of the minimal set of edits that need to be made to the previous dialogue state in order to transform it to the updated one.

We adopt a slightly modified version of the original *Lev* proposed in Lin et al. (2020) for our agreement tracking task, which we will call *A-Lev*. Each *A-Lev* consists of a domain prefix followed by a series of operations. Since our agreement track-

ing task is constrainted to a single domain, the domain prefix is set to either `[gpt-negochat]` or `[multiwoz]` depending on which corpus we are training on. Each individual edit operation is one of the following types: insertion, deletion, and substitution. The final *Lev* is obtained by simply concatenating the domain prefix and the string representations of all the operations. A more thorough description of the method of construction of a *Lev* can be found in the original paper by Lin et al. (2020). Figure 1 shows a tokenized training example used to train our agreement tracking models.

4 Experiments

We begin this section by introducing a rule-based reference model that we will use to characterize our main results. We then introduce our experimental setup, including our choice of backbone model, training objectives, and evaluation metrics.

4.1 Backbone Model (T5)

We choose T5 (Raffel et al., 2020) as our backbone model. T5 is a text-to-text Transformer model consisting of a multi-layer encoder and decoder. It is pre-trained jointly on the self-supervised objectives of autoregressive language modeling, text denoising, and deshuffling. Notably, T5 uses a text-based representation for both its input and output, which allows it to handle a wide variety of NLP tasks without requiring architectural modifications.

Pre-trained T5 models come in multiple variants, each of a different size. We describe the total number of parameters, number of layers, embedding dimension, and the number of attention heads in Table 4. In our work, we only experiment with the T5-small and T5-base variants due to computational budget limitations.

Model	Parameters	# layers	d_model	# heads
Small	60M	6	512	8
Base	220M	12	768	12

Table 4: Properties of the Small and Base variants of the T5 model.

4.2 Rule-Based Agreement Tracker

To provide context for the results of our transfer-learning based model, we develop a rule-based reference model that assumes oracle access to turn-level ground truth annotations. This model helps us to quantify the difficulty of the *multi-turn reasoning* aspect of the agreement tracking task. In other words, our rule-based reference algorithm, ORACLE-TRACKER, is designed to quantify the difficulty of the specific subproblem of *tracking agreements across multiple turns*. It operates under the assumption that the ground-truth values of the dialogue acts (Offer, Accept, Reject, and Other) and accompanying entities are already known (hence the term "oracle").

Agreement tracking models need to solve two subtasks: (1) natural language understanding (NLU) at the utterance level, where they must classify dialogue acts and extract any accompanying slot-value pairs; and (2) tracking agreements over multiple turns. In designing this rule-based reference model, we focus on the demonstrably more difficult second task that involves multi-turn reasoning.

To quantify the relative simplicity of the single-turn NLU problem, we train a T5-base model on just 10 GPT-NEGOCHAT dialogues to classify the dialogue acts and extract corresponding slot-value pairs from each utterance. The resulting model correctly predicts both the dialogue acts and slot-value pairs for 74.33% of the turns in the test set. In contrast, our best-performing model for agreement tracking, also trained on 10 dialogues, achieves a joint slot accuracy of just 20% (see Section 5).

Algorithm 1 Rule-based algorithm (Oracle-Tracker) for agreement tracking

procedure TRACKAGREEMENTS(D)
  $O_A, O_B, C_0 \leftarrow \emptyset$
  for $t \in 1 \dots N$ **do**
    for $d_t^i \in D_t$ **do**
      if $d_t^i =$ AGREE **then**
        $C_t \leftarrow C_{t-1} \bigcup O_{S'_t}$
        $O_{S'_t} \leftarrow \phi$
      else if $d_t^i =$ REJECT **then**
        $O_A, O_B \leftarrow \phi$
      else if $d_t^i =$ OFFER **then**
        $O_A \leftarrow O_A \bigcup O_A^{GT}(t)$
        $O_B \leftarrow O_B \bigcup O_B^{GT}(t)$
      end if
    end for
  end for
  return $\{C_1, \dots, C_T\}$
end procedure

4.3 Transfer Learning from Dialogue State Tracking

Agreement tracking and dialogue state tracking both share the common goal of tracking information across dialogue turns. Although they differ in the type of information they track (agreements vs user goals), our hope is that pre-training a model on dialogue state tracking can equip our language model with transferable representations for the general problem of tracking information across dialogue turns. This representation, in turn, could help the model to adapt to the agreement tracking task in a more sample-efficient manner. The choice of dialogue state tracking as our pre-training task is motivated by the availability of a large amount of publicly annotated data for the task in comparison to agreement tracking.

In our experiments, we compare the joint slot accuracy results of two T5 models to evaluate the extent of this transfer. The first model is fine-tuned only on the agreement tracking task, while the second model is first fine-tuned for the dialogue state tracking task, and then again for agreement tracking.

4.3.1 Multi-Task Training

Multi-task training with a contrastive-learning objective has been widely used to mitigate factual inaccuracies in dialogue summarization and state tracking tasks (Chen et al., 2021; Cao and Wang, 2021; Tang et al., 2022). We describe an analogous setup for agreement tracking that penalizes incorrect outputs by separating them in the model's latent space. Our setup focuses on explicitly teaching our model to discriminate negatively-sampled *A-Lev*s from the correct one.

Since we use a T5 model, which is based on the text-to-text paradigm, we implement the contrastive learning objective by incorporating an additional tokenization scheme during training. Specifically, along with our primary tokenization scheme, which teaches the model to predict the *Lev* given the previous set of agreements along with the dialogue context, we add an auxiliary tokenization scheme that corresponds to our contrastive learning objective. For convenience, we name the primary *Lev*-generation task as GEN and the auxiliary task as CLF. Here, we describe these two trainings by providing their respective tokenization schemes in detail, with examples shown in Figure 1:

1. **GEN**: Our primary task is a conditional generation task that trains a model to generate Levenshtein belief spans for each turn while being conditioned on: 1) the previous agreement state, and 2) the dialogue context.

$$\mathcal{L}_{\text{GEN}} = -log \quad P(A_t|C_t, A_{t-1})$$

2. **GEN + CLF** In our multi-task setup, we supplement the primary task of agreement state prediction (GEN) with an auxiliary task that explicitly trains the model to discern between correct belief spans and negatively sampled incorrect outputs.

   CLF is constructed as a binary classification task that takes in three inputs: 1) the set of agreements as of the previous turn, 2) the dialogue context, and 3) an *A-Lev*, which is randomly set to either the correct output or negatively sampled based on the ontology. The expected output for this task is a boolean "yes" or "no", where "yes" indicates that the *A-Lev* is indeed the correct output and "no" means that the *Lev* is negatively sampled.

$$\begin{aligned}\mathcal{L}_{\text{GEN+CLF}} = &-log\ P(A_t|C_t, A_{t-1}) \\ &- log\ P(Y_t|C_t, A'_t, A_{t-1})\end{aligned}$$

   where A'_t represents a randomly sampled Lev, which can either be a distractor or the correct answer. Y_t is the label of our binary classification task (the binary label is tokenized in the form of "yes" or "no").

4.4 Evaluation Metrics

We evaluate our models using two metrics that are in standard use to evaluate dialogue state tracking models: Joint Slot Accuracy and Joint F1 Score. We briefly describe both of them below.

Joint Slot Accuracy Joint Slot Accuracy is calculated by comparing the predicted agreement at each turn to the ground truth values, with a correct prediction defined as an exact match of all predicted and ground truth values.

Joint F1 The Joint F1 score is calculated in a manner similar to that of Joint Accuracy. The difference is that in the case of a misprediction, rather than using the hard 0 or 1 assigned in the case of Joint Accuracy, the Joint F1 uses a "softer" F1 score by calculating the precision and recall.

summarize: [gpt-negochat] car without job project manager salary 60,000 <EOB> That sounds great. I'm ready to move forward with that offer. I would like a workday of 8 hours <EOU> How about we meet in the middle and you work 9 hours? I'm also open to giving you a 4 day work week. <EOU> Sounds good to me. I was hoping for a pension of 20% though. <EOU> <SOB>

[gpt-negochat] hours 9 </s>

yes or no: [gpt-negochat] job project manager <EOB> I'm interested in taking on the role of project manager. Could you tell me more about what that would involve? <EOU> You got it. <EOU> I'm so pleased you accepted! I was hoping for a salary of 120,000. <EOU> <SOB> [gpt-negochat] hours 10 <EOB> is this belief correct? </s>

no </s>

Figure 1: This figure shows the tokenization schemes corresponding to both the GEN (left) and the CLF (right) tasks. The input for the GEN task consists of the task prefix (green), dataset prefix (blue), belief span representation (red), and concatenated utterances within a window size (yellow). The output includes the dataset prefix (blue) and the updated Levenshtein belief span (dark green). The input in the CLF scheme is mostly identical to that of GEN, except for the task-prefix and the candidate belief span attached at the end. The output in this scheme is only a "yes" or a "no".

4.5 Training Setup

Our models are developed using HuggingFace Transformers and PyTorch Lightning. During the fine-tuning stages, we use a batch size of 32 and apply early stopping against a separate validation set. Adam is used as the optimizer with the learning rate set to 6×10^{-4}. To obtain more reliable estimates, all our results are obtained by averaging over a 3-fold cross-validation scheme. Since sample efficiency is one of our key concerns, we run all our main experiments using 10%, 20%, 30%, 40%, 50%, 75%, and 100% of the training data. We report our hyperparameter search and best-found hyperparameter values in Appendix C.

5 Results

Our experimental results are presented below.

5.1 Rule-Based Reference Model

The results from our rule-based algorithm, ORACLE-TRACKER, are shown in Table 5. We find that it achieves a Joint Slot Accuracy of only 0.5. From this result, we can surmise that it is non-trivial to infer agreements based only on the utterance-level dialogue acts and slot-value pairs.

5.2 Transfer Learning from Dialogue State Tracking

Fine-tuning T5-small and T5-base on MultiWOZ2.4 before GPT-NEGOCHAT improves Joint Slot Accuracy and Joint F1 score compared to fine-tuning on GPT-NEGOCHAT alone. Training on MultiWOZ allows the model to specialize in dialogue-related tasks, like tracking information across turns.

Multi-Task Training The use of the auxiliary binary classification objective (GEN + NEG) does not significantly improve performance as compared to solely training with the generation (GEN) objective. One possible reason for this result could be that the majority of errors made by our model are *false negatives*, whereas the NEG objective is better suited to tackle the problem of *false positives*, or hallucinations. Further experiments with different contrastive learning objectives and negative sampling strategies could shed light on this hypothesis.

Sample Efficiency In Figure 2, we plot the Joint Slot Accuracy obtained by training our models on several different percentage splits of the training dataset. In line with our previous experiments, we run all these experiments using a three-fold split. We find that the larger, T5-base model outperforms the T5-small model in low-resource settings. However, we do not find any significant improvement from using T5-base over T5-small as the percentage of the training split approaches 100%.

6 Related Work

Early approaches (pre deep-learning era) for modeling agreements from meeting transcripts such as Hillard et al. (2003) and Galley et al. (2004) mainly focus on classifying dialogue acts at the utterance level using handcrafted features, but do not attempt to track the slot values associated with those agreements. Bui et al. (2009) propose a labeling scheme similar to ours that includes three dialogue acts: "issue" (raising a new issue), "resolution" (analogous to our offers), and "agreement". Frampton et al. (2009) explicitly consider the real-time aspect of the agreement detection problem by investigating the effect of the context window size on the performance of their incremental agreement-tracking model.

More recent work in this area is focused on gen-

Backbone Model	Fine-Tuning Task(s)	Fine-Tuning Dataset(s)	Joint Acc.	Joint F1
OFFER-ORACLE	-	-	0.5	0.85
T5-small (60M)	GEN	GPT-Negochat	0.32	0.66
	GEN	MWoz + GPT-Negochat	**0.50**	**0.81**
	GEN + CLF	MWoz + GPT-Negochat	**0.53**	**0.80**
T5-base (220M)	GEN	GPT-Negochat	0.47	0.79
	GEN	MWoz + GPT-Negochat	**0.56**	**0.84**
	GEN + CLF	MWoz + GPT-Negochat	**0.54**	**0.86**

Table 5: Results of Agreement Tracking models.

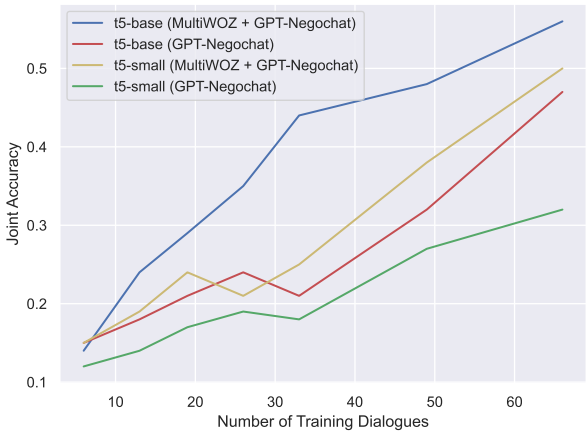

Figure 2: This plot shows the Joint Accuracy trend of four different models when trained with 10, 20, 30, 40, 50, 75, and 100 percentages of the training data. We observe positive effects on the Joint Accuracy from both model size (T5-base is larger than T5-small) as well as an additional step of fine-tuning over DST on MultiWOZ.

erating natural language summaries of meetings and other types of dialogues (Feng et al., 2021a; **?**). Modeling approaches that improve factual correctness in dialogue summarization could be relevant for state tracking tasks as well, as evidenced by Zhao et al. (2021). Jia et al. (2022) group methods to improve factual accuracy in dialogue summarization tasks into three broad strands: (1) injecting pre-processed features (Park et al., 2022), (2) designing self-supervised tasks (Liu and Chen, 2021), and (3) using additional data (Liu et al., 2021; Park et al., 2022).

Finally, although Transformer-based approaches to carrying out DST in task-oriented dialogue continue to push the boundary in both overall as well as few-shot performance (Lee et al., 2021; Peng et al., 2021; Lin et al., 2021b,a), they have yet to demonstrate their efficacy in dialogue paradigms extending beyond task-oriented dialogue, such as collaborative and negotiation dialogues.

7 Limitations and Future Work

Due to our limited computation budget, we did not experiment with larger T5 models such as T5-Large (770M), T5-3B, and T5-11B as well as instruction-tuned variants such as FLAN-T5 (Chung et al., 2022). We also save the exploration of alternative pre-training tasks such as dialogue summarization (Zhang et al., 2020) for future work as our main intention here is to introduce the agreement tracking task formulation and provide a simple, but strong *starting* baseline. Finally, while the proposed GPT-NEGOCHAT improves the linguistic diversity of the synthetic, template-based utterances of the original NEGOCHAT corpus, collecting a fully organic multi-issue negotiation dialogue corpus with turn-level agreement annotations could present a more realistic test-bed to evaluate various agreement tracking approaches.

8 Conclusion

In this work, we introduced the novel task of agreement tracking for multi-issue negotiation dialogues and formulated it as a variant of dialogue state tracking. We found that fine-tuning a language model on the dialogue state tracking task for task-oriented dialogues resulted in improved performance over our naive baseline. In general, state tracking is a capability whose utility extends beyond form-filling dialogues. There is much work to be done to improve the transferability and sample efficiency of dialogue state tracking models to other dialogue paradigms.

Acknowledgements

Placeholder space

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Paweł Budzianowski and Ivan Vulić. 2019. Hello, it's GPT-2 - how can I help you? towards the use of pretrained language models for task-oriented dialogue systems. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 15–22, Hong Kong. Association for Computational Linguistics.

Trung Bui, Matthew Frampton, John Dowding, and Stanley Peters. 2009. Extracting decisions from multi-party dialogue using directed graphical models and semantic similarity. In *Proceedings of the SIGDIAL 2009 Conference*, pages 235–243, London, UK. Association for Computational Linguistics.

Giovanni Campagna, Agata Foryciarz, M. Moradshahi, and Monica S. Lam. 2020. Zero-shot transfer learning with synthesized data for multi-domain dialogue state tracking. In *Annual Meeting of the Association for Computational Linguistics*.

Shuyang Cao and Lu Wang. 2021. CLIFF: contrastive learning for improving faithfulness and factuality in abstractive summarization. *CoRR*, abs/2109.09209.

Kushal Chawla, Jaysa Ramirez, Rene Clever, Gale Lucas, Jonathan May, and Jonathan Gratch. 2021. CaSiNo: A corpus of campsite negotiation dialogues for automatic negotiation systems. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3167–3185, Online. Association for Computational Linguistics.

Sihao Chen, Fan Zhang, Kazoo Sone, and Dan Roth. 2021. Improving faithfulness in abstractive summarization with contrast candidate generation and selection. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5935–5941, Online. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Xiachong Feng, Xiaocheng Feng, and Bing Qin. 2021a. A survey on dialogue summarization: Recent advances and new frontiers.

Yue Feng, Yang Wang, and Hang Li. 2021b. A sequence-to-sequence approach to dialogue state tracking. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1714–1725, Online. Association for Computational Linguistics.

Matthew Frampton, Jia Huang, Trung Bui, and Stanley Peters. 2009. Real-time decision detection in multi-party dialogue. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1133–1141, Singapore. Association for Computational Linguistics.

Michel Galley, Kathleen McKeown, Julia Hirschberg, and Elizabeth Shriberg. 2004. Identifying agreement and disagreement in conversational speech: Use of Bayesian networks to model pragmatic dependencies. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 669–676, Barcelona, Spain.

He He, Anusha Balakrishnan, Mihail Eric, and Percy Liang. 2017. Learning symmetric collaborative dialogue agents with dynamic knowledge graph embeddings. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1766–1776, Vancouver, Canada. Association for Computational Linguistics.

He He, Derek Chen, Anusha Balakrishnan, and Percy Liang. 2018. Decoupling strategy and generation in negotiation dialogues. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2333–2343, Brussels, Belgium. Association for Computational Linguistics.

Dustin Hillard, Mari Ostendorf, and Elizabeth Shriberg. 2003. Detection of agreement vs. disagreement in meetings: Training with unlabeled data. In *Companion Volume of the Proceedings of HLT-NAACL 2003 - Short Papers*, pages 34–36.

Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue. *Advances in Neural Information Processing Systems*, 33:20179–20191.

Léo Jacqmin, Lina M. Rojas Barahona, and Benoit Favre. 2022. "do you follow me?": A survey of recent approaches in dialogue state tracking. In *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 336–350, Edinburgh, UK. Association for Computational Linguistics.

Qi Jia, Siyu Ren, Yizhu Liu, and Kenny Q. Zhu. 2022. Taxonomy of abstractive dialogue summarization: Scenarios, approaches and future directions.

Simon Keizer, Markus Guhe, Heriberto Cuayáhuitl, Ioannis Efstathiou, Klaus-Peter Engelbrecht, Mihai Dobre, Alex Lascarides, and Oliver Lemon. 2017. Evaluating persuasion strategies and deep reinforcement learning methods for negotiation dialogue agents.

Vasily Konovalov, Ron Artstein, Oren Melamud, and Ido Dagan. 2016. The Negochat Corpus of Human-agent Negotiation Dialogues. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3141–3145, Portorož, Slovenia. European Language Resources Association (ELRA).

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online. Association for Computational Linguistics.

Chia-Hsuan Lee, Hao Cheng, and Mari Ostendorf. 2021. Dialogue state tracking with a language model using schema-driven prompting. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4937–4949, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Wenqiang Lei, Xisen Jin, Min-Yen Kan, Zhaochun Ren, Xiangnan He, and Dawei Yin. 2018. Sequicity: Simplifying task-oriented dialogue systems with single sequence-to-sequence architectures. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1437–1447, Melbourne, Australia. Association for Computational Linguistics.

Zhaojiang Lin, Bing Liu, Andrea Madotto, Seungwhan Moon, Paul Crook, Zhenpeng Zhou, Zhiguang Wang, Zhou Yu, Eunjoon Cho, Rajen Subba, and Pascale Fung. 2021a. Zero-shot dialogue state tracking via cross-task transfer.

Zhaojiang Lin, Bing Liu, Andrea Madotto, Seungwhan Moon, Zhenpeng Zhou, Paul Crook, Zhiguang Wang, Zhou Yu, Eunjoon Cho, Rajen Subba, and Pascale Fung. 2021b. Zero-shot dialogue state tracking via cross-task transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7890–7900, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zhaojiang Lin, Andrea Madotto, Genta Indra Winata, and Pascale Fung. 2020. MinTL: Minimalist transfer learning for task-oriented dialogue systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3391–3405, Online. Association for Computational Linguistics.

Chunyi Liu, Peng Wang, Jiang Xu, Zang Li, and Jieping Ye. 2019. Automatic dialogue summary generation for customer service. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery amp; Data Mining*, KDD '19, page 1957–1965, New York, NY, USA. Association for Computing Machinery.

Junpeng Liu, Yanyan Zou, Hainan Zhang, Hongshen Chen, Zhuoye Ding, Caixia Yuan, and Xiaojie Wang. 2021. Topic-aware contrastive learning for abstractive dialogue summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1229–1243, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zhengyuan Liu and Nancy Chen. 2021. Controllable neural dialogue summarization with personal named entity planning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 92–106, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Iain Mccowan, J Carletta, Wessel Kraaij, Simone Ashby, S Bourban, M Flynn, M Guillemot, Thomas Hain, J Kadlec, V Karaiskos, M Kronenthal, Guillaume Lathoud, Mike Lincoln, Agnes Lisowska Masson, Wilfried Post, Dennis Reidsma, and P Wellner. 2005. The ami meeting corpus. *Int'l. Conf. on Methods and Techniques in Behavioral Research*.

Seongmin Park, Dongchan Shin, and Jihwa Lee. 2022. Leveraging non-dialogue summaries for dialogue summarization. In *Proceedings of the First Workshop On Transcript Understanding*, pages 1–7, Gyeongju, South Korea. International Conference on Computational Linguistics.

Baolin Peng, Chunyuan Li, Jinchao Li, Shahin Shayandeh, Lars Lidén, and Jianfeng Gao. 2021. Soloist: Building task bots at scale with transfer learning and machine teaching. *Transactions of the Association for Computational Linguistics*, 9:807–824.

Volha Petukhova, Christopher A. Stevens, Harmen de Weerd, Niels Anne Taatgen, Fokie Cnossen, and Andrei Malchanau. 2016. Modelling multi-issue bargaining dialogues: Data collection, annotation design and corpus. In *International Conference on Language Resources and Evaluation*.

Henry Prakken. 2006. Formal systems for persuasion dialogue. *The knowledge engineering review*, 21(2):163–188.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:8689–8696.

Virgile Rennard, Guokan Shang, Julie Hunter, and Michalis Vazirgiannis. 2022. Abstractive meeting summarization: A survey.

Xiangru Tang, Arjun Nair, Borui Wang, Bingyao Wang, Jai Desai, Aaron Wade, Haoran Li, Asli Celikyilmaz, Yashar Mehdad, and Dragomir Radev. 2022. CONFIT: Toward faithful dialogue summarization with linguistically-informed contrastive fine-tuning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5657–5668, Seattle, United States. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Lu Wang and Claire Cardie. 2013. Domain-independent abstract generation for focused meeting summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1395–1405, Sofia, Bulgaria. Association for Computational Linguistics.

Xuewei Wang, Weiyan Shi, Richard Kim, Yoojung Oh, Sijia Yang, Jingwen Zhang, and Zhou Yu. 2019. Persuasion for good: Towards a personalized persuasive dialogue system for social good. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5635–5649, Florence, Italy. Association for Computational Linguistics.

J. Williams, Antoine Raux, and Matthew Henderson. 2016. The dialog state tracking challenge series: A review. *Dialogue Discourse*, 7:4–33.

Chien-Sheng Wu, Andrea Madotto, Ehsan Hosseini-Asl, Caiming Xiong, Richard Socher, and Pascale Fung. 2019. Transferable multi-domain state generator for task-oriented dialogue systems. In *Annual Meeting of the Association for Computational Linguistics*.

Atsuki Yamaguchi, Kosui Iwasa, and Katsuhide Fujita. 2021. Dialogue act-based breakdown detection in negotiation dialogues. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 745–757, Online. Association for Computational Linguistics.

Haolan Zhan, Yufei Wang, Tao Feng, Yuncheng Hua, Suraj Sharma, Zhuang Li, Lizhen Qu, and Gholamreza Haffari. 2022. Let's negotiate! a survey of negotiation dialogue systems.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 11328–11339. PMLR.

Jeffrey Zhao, Mahdis Mahdieh, Ye Zhang, Yuan Cao, and Yonghui Wu. 2021. Effective sequence-to-sequence dialogue state tracking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7486–7493, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yiheng Zhou, He He, Alan W Black, and Yulia Tsvetkov. 2019. A dynamic strategy coach for effective negotiation. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 367–378, Stockholm, Sweden. Association for Computational Linguistics.

A Prompt to Generate GPT-NEGOCHAT

The following prompt template is fed to the **text-davinci-003** variant of GPT-3 to generate rephrased utterances for GPT-NEGOCHAT:

> Rephrase this while still maintaining the same meaning. Feel free to add some minimal niceties and make it sound less robotic. While rejecting an offer, try to come up with a reason. Make the tone sound like a real job interview: [original utterance]

B Sizes of Dataset Splits

For all our experiments, we follow the following steps:

1. We first perform a three-fold split on our entire dataset, which results in a 66.67% of the samples being assigned to the training split and the remaining to the test split.

2. For each split, we further subdivide the training split into the actual training and the validation split using a 85%-15% split.

3. Effectively, this strategy results in the training, validation, and test splits containing 56.67%, 10%, and 33.33% of the dataset respectively.

C Hyperparameter Tuning

We restrict our hyperparameter sweep to a few values over training-related hyperparameters such as learning rate, batch size, early stopping criterion,

and precision. We also sweep over varying sizes of the context window for representing our dialogue context. To ensure that we do not over-optimize our hyperparameters and stumble upon "lucky" values, we perform an initial sweep once with a fixed configuration and apply the best-found hyperparameter values to all our other experiments.

We report the search spaces and best-found values for our hyperparameters in Table 6.

D A (Longer) Side-by-Side Comparison of Utterances

Table 7 shows an additional list of side-by-side examples of original and rephrased utterances in the NEGOCHAT and GPT-NEGOCHAT datasets respectively.

Hyperparameter	Sweep Range	Best Value
Learning Rate	$1 \times 10^{-4}, 1 \times 10^{-5}, 1 \times 10^{-3}, 6 \times 10^{-4}$	6×10^{-4}
Bath Size	32, 64, 128	32
Early Stopping (Min Delta)	0, 0.001, 0.005, 0.1, 0.5	0
Early Stopping (Patience)	0, 1, 2, 3, 4	4
Gradient Clip Norm	1.0	1.0
Precision	16, 32	32
Context Window Size	1, 2, 3, 4	3

Table 6: Hyperparameter values for all experiments.

Speaker	Original Utterance	Rephrased Utterance
Candidate	I would like a workday of 8 hours	I'm looking for something that would require 8 hours a day.
Employer	cand you do 10?	Can you do 10 hours a day?
Candidate	I disagree about a workday of 10 hours. I would like a workday of 8 hours	I'm not comfortable with that. I'd prefer 8 hours a day.
Employer	what else pension, car?	What else do you need, like a pension or a car?
Candidate	I accept programmer position. Now let's talk about how we can agree on other issues. I want a salary of 120,000?	I'd like to accept the programmer position. How about we discuss the other details, such as salary? I was hoping for something around 120,000.
Employer	I would like to offer you 90,000 for salary.	That's a bit higher than what we had in mind. We were thinking of offering 90,000.
Candidate	I would like a pension of 20%	Could I get a pension of 20%?
Candidate	I accept 60,000. I would like to continue and discuss another issue. I want a position of project manager	Yes, that's fine. I'd also like to discuss another topic - I'm hoping for a project manager position.
Employer	I accept project manager.	Absolutely, we can make that happen.
Candidate	It's good to reach an agreement on project manager	It's great that we can come to an agreement on the project manager.
Candidate	I want a company car	I want a company car
Employer	I do not accept leased car, can you manage without one?	I do not accept leased car, can you manage without one?
Candidate	I reject no company car. I would like a company car	I definitely want a company car.

Table 7: More excerpts from the NEGOCHAT corpus alongside its rephrased counterparts from GPT-NEGOCHAT.