# Leveraging Ensemble Techniques and Metadata for Subjective Knowledge-grounded Conversational Systems

**Seongho Joo**[*1] **Kang-il Lee**[*1] **Kyungmin Min**[*1] **Joongbo Shin**[2]
**Janghoon Han**[2] **Seungpil Won**[2] **Kyomin Jung**[1,3]
[1]Seoul National University, [2]LG AI Research, [3]SNU-LG AI Research Center
{seonghojoo, 4bkang, kyungmin97, kjung}@snu.ac.kr
{jb.shin, janghoon.han, seungpil.won}@lgresearch.ai

## Abstract

The goal of DSTC11 track 5 is to build task-oriented dialogue systems that can effectively utilize external knowledge sources such as FAQs and reviews. This year's challenge differs from previous ones as it includes subjective knowledge snippets and requires multiple snippets for a single turn. We propose a pipeline system for the challenge focusing on entity tracking, knowledge selection and response generation. Specifically, we devise a novel heuristic to ensemble the outputs from the rule-based method and neural model for entity tracking and knowledge selection. We also leverage metadata information in the knowledge source to handle fine-grained user queries. Our approach achieved the first place in objective evaluation and the third place in human evaluation of DSTC11 track 5.

## 1 Introduction

Task-oriented dialogue (TOD) systems aim to provide users with specific services, such as making restaurant reservation or booking hotel. Typically, these systems utilize domain-specific APIs, which are limited to predefined functionalities. However, there are scenarios where a user's request go beyond the coverage of domain API. For instance, the user might ask about alcohol availability at a restaurant or the availability of free WiFi at a hotel. To handle such cases, TOD system that can refer to external knowledge base, or knowledge-grounded TODs, has drawn much attention (Wu et al., 2019; Kim et al., 2020).

Previous studies on knowledge-grounded TODs have focused on factual knowledge, such as FAQs about restaurants or hotels. In practice, users usually show interest not only in factual information but also in subjective information, involving the experiences, opinions, and preferences of other customers (Zhao et al., 2023). To address this aspect, DSTC11 track 5 specifically focuses on building a dialogue system that can effectively utilize subjective knowledge snippets. The subjective knowledge snippets consist of customer reviews concerned with hotels and restaurants, such as "If only the room was a bit cleaner, it would have been perfect" and "We loved the atmosphere from the moment we walked in." Furthermore, in some cases, multiple knowledge snippets need to be retrieved in order to adequately respond to a single user request, adding an additional layer of complexity to DSTC11 track 5.

To this end, we propose a pipeline system that addresses four subtasks of knowledge-grounded TOD: (1) knowledge-seeking turn detection, (2) entity tracking, (3) knowledge selection, and (4) response generation. For entity tracking, we propose a novel heuristic that combines the results of rule-based and neural entity matching models. In knowledge selection, we leverage given metadata to precisely retrieve relevant review snippets when the user's request is highly specific, such as asking about a particular menu of a restaurant. For response generation, we propose a data augmentation method using a Large Language Model (LLM) to fine-tune the response generation model, enabling it to handle cases with mixed opinions more effectively. Overall schema of our pipeline is illustrated in Figure 1.

Our approach demonstrates significant improvement over baseline method in terms of the accuracy of knowledge selection and the quality of response generation. Furthermore, our submission achieved the top in objective evaluation and the third place in human evaluation of DSTC11 track 5.

## 2 Task Description

DSTC11 Track 5 is about Task-oriented Conversational Modeling with Subjective Knowledge. The dialogue dataset used in this track is an extension of MultiWoz 2.1(Eric et al., 2020), where reviews

[*] Equal contribution.



*Proceedings of The Eleventh Dialog System Technology Challenge*, pages 206–215

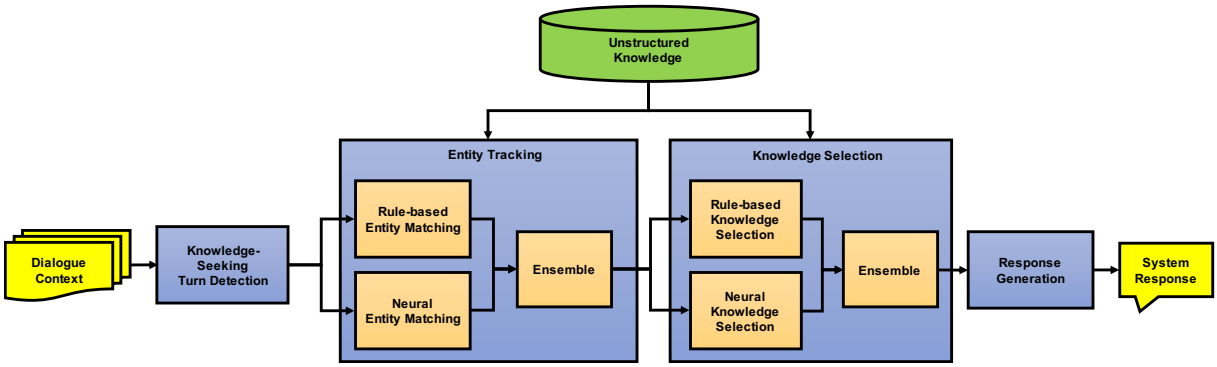


Figure 1: Overall pipeline of our conversational model.

for the exact location contain different opinions. Subjective opinions make the task more challenging than DSTC9 track 1 (Kim et al., 2020), which relied only on factual knowledge for generating responses.

The overall task can be divided into four sequential subtasks, which are as follows: knowledge-seeking turn detection, entity tracking, knowledge selection, and response generation. Knowledge-seeking turn detection determines whether subjective knowledge is required to generate the response based on the dialogue context and the user's last utterance. Entity tracking is the task of identifying the entity relevant to the user's last utterance, which helps reduce the number of candidate knowledge snippets. Knowledge selection involves finding relevant knowledge snippets to user's request from the narrowed-down knowledge pool (Reviews, FAQs) obtained by the entity tracking. Finally, response generation is the task of generating a response of the user's request grounded on the selected knowledge snippets. Given that the knowledge snippets often have conflicting opinions about entities, it is crucial to reflect both positive and negative reviews whenever possible.

The challenge dataset is structured in the following way. Firstly, it includes a dialog dataset that contains conversations between a user and a system. Secondly, there is a Knowledge dataset comprising reviews and FAQs for each domain/entity. In this context, "domain" refers to "hotel" and "restaurant", and "entity" refers to specific names such as a hotel name (e.g. CITYROOMZ) or a restaurant name (e.g. ROSA'S BED AND BREAKFAST). Each entity has its own set of review documents and FAQs. The review documents comprise sentences and also store metadata that offers additional information about each review, such as the type of traveler (e.g. couples, etc.), specific dishes (e.g. beef wellington, etc.), and beverages (e.g. beer, ale, etc.). The FAQs are stored as pairs of questions and answers.

## 3 Method

In this section, we explain the methods for each subtask: knowledge-seeking turn detection, entity tracking, knowledge selection, and response generation. The overall pipeline of our subtask is described in the Figure 1.

### 3.1 Knowledge-Seeking Turn Detection

Knowledge-seeking turn detection is a binary classification task that determines whether subjective knowledge is needed given dialogue context. We use the DeBERTa-v3-base (He et al., 2023) model as our backbone. We feed the dialogue context into the pretrained DeBERTa-v3 model and get output vectors by mean pooling. Afterward, we pass the pooled vector through a linear layer and use softmax to construct a binary classifier. This method is the same as the given baseline method in DSTC11 track 5. The model was fine-tuned by binary cross-entropy loss.

### 3.2 Entity Tracking

Entity tracking is a method that aims to find relevant entities based on the dialogue context and the user's request. Accurately extracting relevant entities is crucial for optimizing the performance and efficiency of a dialogue system. The more accurate the entity tracking, the fewer knowledge candidates need to be retrieved in the knowledge selection step, resulting in more precise and efficient knowledge selection process. Therefore, we perform both rule-based entity matching and neural entity matching to achieve accurate entity tracking

and then ensemble those entity matching results.

#### 3.2.1 Rule-based Entity Matching

We develop our own rule-based entity matching method based on the baseline entity matching method provided in DSTC11 track 5. In each dialogue turn in dialogue context, we perform fuzzy n-gram matching (Jin et al., 2021; Zhao et al., 2023) with all entities (143 entities). Fuzzy n-gram matching finds the longest contiguous matching sub-sequence between each dialogue turns and entities, then calculates a matching ratio. If the matching ratio exceeds a predefined threshold, we store the entity and its dialogue history turn. Based on our observation, we noticed that the more recent an entity appears, the more relevant it tends to be to the user's request. To tackle this, we track entities from the most recent dialogue history turn selected by fuzzy matching. This approach enables us to identify and prioritize the entities that are most likely to be relevant to the user's current request.

#### 3.2.2 Neural Entity Matching

In neural entity matching, we regard the task as a sentence pair classification problem. We use the DeBERTa-v3-large model to find relevant entities. The dialogue history and entity are fed into the DeBERTa model as an input of sentence pair classification task. The training and inference are conducted in the same way as in knowledge-seeking turn detection.

Unlike knowledge-seeking turn detection, there is no negative samples provided by the dataset. Thus, we build our own set of negative samples using 4 sampling strategies with various difficulty: (1) rule-based entity matching negative, (2) similar-name negative, (3) in-domain negative, and (4) random negative. These multi-scale negatives improve the robustness of our neural entity matching model.

Rule-based entity matching negatives are sampled from aforementioned rule-based entity matching algorithm's false positives. These samples serve as hard negatives that are very informative for the neural model. Similar-name negatives are sampled from the set of entities with at least 50% token overlap with ground truth entity. In-domain negatives are sampled from the entities with the same domain and random negatives are randomly sampled from the whole entity set.

#### 3.2.3 Ensemble

Here, we present our heuristic ensemble method that combines the outputs of rule-based entity matching and neural entity matching. We have noticed that entity annotation errors are quite common, particularly when multiple entities are involved in a single turn. These annotation errors deteriorate the neural entity matching performance in scenarios where multiple entities are present.

In contrast, the advantage of rule-based entity matching lies in its robustness against annotation errors. It has the ability to track multiple entities fairly when they appear in a single dialogue turn. However, rule-based entity matching lacks the capability to understand the dialogue context and only extracts entities from the most recent dialogue turn selected by fuzzy matching. From this perspective, neural entity matching excels as it can utilize the understanding of the dialogue context to extract entities that appear in past dialogue turns.

We have observed that when the neural model tracks only one entity in the dialogue context, it has a high level of confidence in the prediction. Consequently, if the neural model predicts only one entity, we utilize the prediction of the neural model. For dialogue contexts where the neural model extracted multiple entities, we use the entities extracted by rule-based entity matching. This heuristic of ensembling the results of two distinct models demonstrates strong performance compared to baseline entity matching.

### 3.3 Knowledge Selection

Knowledge selection involves identifying relevant knowledge snippets to address user inquiries effectively. Initially, the candidate knowledge snippets are narrowed down through entity tracking. The retrieval process is then conducted on these refined candidates using two distinct approaches: neural knowledge selection and rule-based knowledge selection.

#### 3.3.1 Neural Knowledge Selection

In neural knowledge selection, context is critical when dealing with pronouns in review sentences. Without the context provided by previous sentence, it is tough to understand a review sentence containing a pronoun. Thus, we concatenate the preceding sentence in front of each candidate sentence to ascertain the relevance. In conclusion, a pair of consecutive review sentences form what we refer to as "consecutive knowledge snippets".

We use DeBERTa-v3-large as the backbone model. We construct the model input as follows: [CLS] + dialog history + [SEP] + consecutive knowledge snippets + [SEP], where [CLS] and [SEP] represent special tokens. For FAQs, consecutive knowledge snippets are constructed by concatenating the question and answer pairs. Then, binary classification is performed by mean pooling the last hidden states to determine the relevance of each knowledge snippet and the user's request.

#### 3.3.2 Rule-based Knowledge Selection

The unstructured knowledge base provided in DSTC11 track 5 allows us to harness metadata for knowledge selection. The metadata let us know in advance which dish or drink the restaurant review document was related to. We also have found that the user's request is often associated with a specific drink or dish. In this case, utilizing metadata is a very efficient and accurate method.

To achieve this, we collect all entities related to dishes and drinks derived from the metadata, hereafter referred to as the "metadata entity set." We adopt a fuzzy n-gram matching approach to compare the user's latest utterance with the metadata entity set, facilitating the decision on whether to leverage metadata for the knowledge selection step. If the metadata is found to be relevant, an additional round of fuzzy n-gram matching is performed between all candidate knowledge snippets and the corresponding metadata to select relevant knowledge snippets.

Fuzzy n-gram matching effectively identifies specific words' presence but falls short in determining the context. To address this limitation, we employ DeBERTa-v3-large. This neural model explores the knowledge snippets and retrieves the most relevant knowledge snippet with the highest logit value among the knowledge snippets within the document containing the metadata detected by fuzzy matching. In other words, for each document containing relevant metadata, the neural model identifies the most suitable knowledge snippet. Finally, the output of rule-based knowledge selection is constructed by the union of knowledge snippets obtained through fuzzy n-gram matching and those selected by the neural model.

#### 3.3.3 Ensemble

In the DSTC11 track 5 challenge, metadata includes information such as the names of dishes and drinks. If the user's last utterance contains metadata, we utilize the rule-based knowledge selection method. Sometimes, the metadata is mentioned in the user's last utterance but not found in any candidate knowledge snippets. In these cases, no knowledge snippet would be retrieved through fuzzy n-gram matching. In such exceptional cases, no knowledge snippets would be selected, and we rely on the results obtained from the neural knowledge selection. Also, we employ the neural knowledge selection when the user's last utterance does not contain any metadata.

### 3.4 Response generation

Response generation is the core task of task-oriented dialogue generation where the system response is generated based on the knowledge snippets, dialogue context, and the input utterance of the user. The main issue of response generation is properly reflecting all relevant knowledge snippets. However, we have observed that response generation models often neglect some of the knowledge snippets that are relevant to the user utterance and context history. Consequently, the model sometimes reflects only negative reviews regarding the domain or only positive reviews even if there exist both negative and positive reviews about the place.

#### 3.4.1 Pseudo labels

To remedy the aforementioned problem, we generated pseudo labels that indicate whether given knowledge snippets have mixed opinions or not. In here, the 'mixed' means that the reviews both have positive and negative nuances about the domain. To generate the pseudo labels of knowledge snippets, the recently celebrated GPT-3 (text-davinci-003) model was used.

```
Determine whether the following
reviews contain conflicting
options related to the context:
Context: <Question 1>
Reviews: <Knolwedge Snippets1>
Opinions are conflicting: true
.
.
.
Context: <QuestionN>
Revies: <Knolwedge SnippetsN>
Opinions are conflicting: false
Context: <Question>
Reviews: <Test snippets>
Opinions are conflicting:
```

Figure 2: Example of the prompt for in-context learning

| Method | Exact Match | Over Prediction | Under Prediction | Incorrect |
|---|---|---|---|---|
| Baseline | 0.9076 | 0.0351 | 0.0129 | 0.0444 |
| Rule-based | 0.9316 | 0.0377 | 0.0081 | 0.0226 |
| Neural | 0.8987 | 0.0869 | 0.0002 | 0.0122 |
| Baseline + Neural | 0.9494 | 0.0189 | 0.0129 | 0.0189 |
| Rule-based + Neural | **0.9545** | 0.0233 | 0.0085 | 0.0137 |

Table 1: Results of entity tracking task on custom test set. If the predicted entity set is $P$ and the ground-truth entity set is $G$, Exact Match, Over Prediction, Under Prediction, and Incorrect occurs when $P = G$, $G \subset P$, $P \subset G$, and $(P - G \neq \emptyset) \cap (G - P \neq \emptyset)$, respectively. The numbers in the table represent the proportion that each case occupies in the entire test set.

The pseudo labels are generated via in-context learning. First, pairs of knowledge snippets and user utterances are labeled manually by humans. Each sample (a knowledge snippet and a user utterance) is given a true label if the snippets have mixed opinions, and a false label otherwise. The knowledge snippets of our interest get the pseudo label by GPT-3 via in-context learning with the samples like Figure 2.

We also filter out trivial knowledge snippet cases like there is only a single ground truth knowledge snippet for the turn. After the in-context learning, pseudo labels for ground truth knowledge snippets are generated. These labels are treated as special tokens along with other tokens to generate the response.

### 3.4.2 Augmentation

Another speculation for why the model is vulnerable when knowledge snippets have mixed reviews is that the training set is biased towards simple cases like all reviews are positive or negative compared to mixed cases. This assumption guides us to generate mixed snippets where grounded knowledge snippets both have positive and negative reviews about the place.

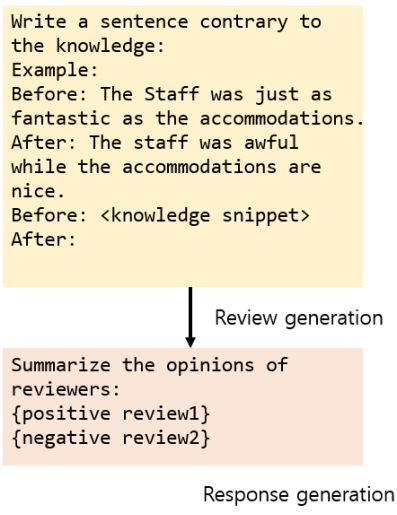


Figure 3: Example of data augmentation

We again employ the large language models to generate mixed reviews based on a positive or negative reviews. We instruct the GPT-3 (text-davinci-003) to write a review contrary to the knowledge. For example, if the input utterance is "`The staff was just as fantastic as the accommodation`", we expect that the output utterance from GPT-3 should have the opposite meaning like "`The staff was awful while the accommodations are nice`". Like the pseudo-label generation, in-context learning is used for the review generation. Finally, the response for query and snippets are generated by prompt `Summarize the opinions of reviewers`.

Although it is an effective approach for generating cases where opinions are mixed, the response might differ by a large margin from the ground truth response which leads to deteriorated BLEU score. The overall process is described in Figure 3. We observe that there is little drop in BLEU score when a small amount of augmented samples are added, but the BLEU dropped by a large margin when a large amount of augmented samples are added to the dataset.

## 4 Experiments

### 4.1 Entity Tracking

Our work focuses on two different approaches in the Entity Tracking task. The first approach aims to increase the recall score to minimize the chances of missing relevant entities. However, this approach has challenges in accurately selecting the appropriate knowledge during the selection stage. The second approach prioritizes increasing the exact match score to facilitate accurate knowledge selection. We concentrated on the latter approach and

designed an ensemble method, described in section 3.2.3, to enhance the exact match ratio.

To evaluate the performance of our proposed method, we conduct experiments comparing various entity matching techniques: (1) baseline entity matching, (2) rule-based entity matching, (3) neural entity matching, (4) ensemble of baseline entity matching with neural entity matching, and (5) ensemble of rule-based entity matching with neural entity matching. To ensure the consistency of evaluation, we create a custom test set specifically for this purpose. The custom test set is designed to have no duplicate logs and contains unseen entities, accounting for 10% of the total entities.

Our findings, presented in Table 1, demonstrate that when applied to baseline method, our rule-based method improves exact match ratio from 0.91 to 0.93. This difference can be attributed to the fact that we lowered the threshold from 0.95 to 0.85. With a threshold of 0.95, even minor typos in the log's entity would result in exclusion, whereas lowering the threshold to 0.85 allowed for the inclusion of such cases. Analysis of the log data reveals a substantial occurrence of entity name typos, supporting this adjustment's effectiveness.

The neural entity matching approach alone exhibits a low exact match score, especially in multi-entity tracking. Our proposed method, which combines rule-based entity matching with neural entity matching, overcomes this weakness and the results demonstrate strong performance which make improvement in the exact match performance compared to the baseline method, with the score increasing from 0.91 to 0.95.

### 4.2 Knowledge Selection

In Knowledge Selection task, we use the ground truth label of the Entity tracking task as input to verify the performance of our proposed neural-based knowledge selection. We compared the performance of DeBERTa-v3-base with only one candidate knowledge snippet as the baseline method and DeBERTa-v3-base with our proposed consecutive knowledge snippets.

Table 2 shows that giving consecutive knowledge snippets is better in terms of precision, recall, F-1 score, and exact match ratio. We adopt a consecutive knowledge snippet approach and aim to analyze the impact of model size on performance. Specifically, we compare the performance of two models, DeBERTa-v3-base and DeBERTa-v3-large, to determine if increasing the model size would improve results.

| Method | Precision | Recall | F-1 score | Exact match |
|---|---|---|---|---|
| Baseline-DeBERTa-base | 0.9596 | 0.9416 | 0.9505 | 0.8555 |
| Consecutive-DeBERTa-base | 0.9661 | 0.9533 | 0.9597 | 0.8662 |
| Consecutive-DeBERTa-large | 0.9626 | **0.9638** | 0.9632 | 0.8935 |
| Ensemble | **0.9714** | 0.9553 | **0.9633** | **0.9009** |

Table 2: Results of Neural Knowledge Selection

The evaluation results shows that utilizing a larger model enhances the retrieval performance. Based on this finding, we select the DeBERTa-v3-large model for knowledge selection, considering its superior performance in capturing relevant information.

Moreover, we further improve our system by ensembling the DeBERTa-v3-large model through a majority voting scheme. This ensemble model is trained and validated on three separate dataset splits, allowing for diversified learning. Notably, this ensemble approach substantially enhanced the F-1 score and exact match accuracy.

Our experiments demonstrate the efficacy of employing consecutive knowledge snippets and the advantages of leveraging a larger model, DeBERTa-v3-large, for knowledge selection. Furthermore, our ensemble strategy based on multiple training partitions led to significant performance improvements, as indicated by the enhanced F-1 score and exact match accuracy.

### 4.3 Generation

We've used the T5-large model as the baseline in this section. First, we conducted the experiment

| Model | BLEU | Mix failed |
|---|---|---|
| Baseline | **0.111** | 52/211 |
| 1600 Aug | 0.106 | 32/211 |
| 3200 Aug | 0.104 | 28/211 |
| 4800 Aug | 0.105 | **22/211** |
| Baseline+Pseudo | 0.103 | 53/211 |
| Baseline+Pseudo+Aug | 0.102 | 45/211 |
| T5-3B+Pseudo | 0.101 | 51/211 |
| T5-3B+Pseudo+Aug | 0.95 | 28/211 |

Table 3: Performance on DSTC11 validation set with the data augmentation. "Mix failed" denotes the portion of samples that fail to reflect both positive and negative samples. number+'Aug' denotes the augmentation method with n additional augmented samples. 'Pseudo' denotes the model trained with pseudo labels.

| | Knowledge Selection Ensemble(Voting) | Response Generation | | |
|---|---|---|---|---|
| | | Data Augmentation | Pseudo Labeling | Composition |
| Entry 0 | Single Model | YES | YES | NO |
| Entry 1 | 3 out of 6 votes required for Selection | YES | YES | NO |
| Entry 2 | 6 out of 6 votes required for Selection | YES | YES | NO |
| Entry 3 | 6 out of 6 votes required for Selection | NO | NO | NO |
| Entry 4 | 6 out of 6 votes required for Selection | NO | NO | YES |

Table 4: Our submission entries in DSTC11 track 5. Pseudo labeling refers to incorporating a "mixed" label, identifying whether the selected knowledge snippets possess both positive and negative senses. Composition means that the model generates individual responses for each knowledge snippets and combines the responses using InstructGPT.

to verify if the augmentation approach is effective. As a first step, we first gather samples that have both positive and negative reviews from the held out DSTC11 dataset. We then compare model responses with ground truth responses from the filtered-out samples. In addition to BLEU score, we count how many samples the model succeeds to reflect both positive and negative reviews.

Table 3 shows the evaluation results of various generation methods. Overall, the augmentation approach generates lower BLEU score samples due to the style dissimilarity. However, the model with an augmentation approach shows better performance regarding mixed reviews. We have not seen a conspicuous change in BLEU score or human evaluation as we increase the number of augmentation samples. Here, 'pseudo' denotes the pseudo labels, and 'aug' denotes the data augmentation. Although T5-3B model with pseudo labels and data augmentation showed superior performance regarding reflecting both positive and negative reviews, its BLEU score drops significantly compared to the baseline.

### 4.4 Submission Results

The evaluation in the DSTC11 track 5 challenge involved three subtasks: knowledge-seeking turn detection (task 1), knowledge selection (including Entity tracking) (task 2), and response generation (task 3). Each participating team has the opportunity to submit a maximum of five entries. In our submissions, we employ the following method:

We use the DeBERTa-v3-base model provided in the baseline for turn detection in all five entries. All submitted entries use ensemble method for entity tracking and knowledge selection respectively. The details of the ensemble method using majority voting in the knowledge selection are described in Table 4 for each entry. In addition, the details of each entry submitted in the response generation are also reported in Table 4. Out of the 14 teams, our team is Team 13. Our submission entries ranked the highest score in precision, F-1, and exact match in knowledge selection and BLEU score in response generation. Additionally, our model secured the first place in the accuracy category of the human evaluation and obtained third place overall.

Among 5 entries, Entry ID 3 was selected as the best system and evaluated by humans. As shown in Table 5 and Table 6, our approach achieved the first place in objective evaluation and the third place in human evaluation of DSTC11 track 5.

## 5 Related Work

Task-oriented dialogue systems are conversational systems designed to solve specific tasks or purposes requested by users. Traditional TOD systems have solved these problems by dividing them into subtasks using a pipleline architecture (Young et al., 2013). The pipeline architecture consists of natural language understanding (NLU) (Liu and Lane, 2016; Koh et al., 2023), dialogue state tracking (DST) (Nouri and Hosseini-Asl, 2018; Lee et al., 2021), dialogue policy (Peng et al., 2017), and natural language generation (NLG) (Wen et al., 2015; Peng et al., 2020) modules. By solving these four modules sequentially, the system generates dialogues that fulfill the user's objectives. End-to-end neural models for task-oriented dialogue (TOD) systems, which build a conversational system using a single unified model, have also been continuously researched and developed (Hosseini-Asl et al., 2020; Ham et al., 2020; Lee, 2021).

Task-oriented dialogue systems have predominantly relied on pre-defined application programming interfaces (APIs) and structured databases to generate user-specific responses (Zhao et al., 2023). However, this approach has limitations as it relies on having prior knowledge of user needs and encounters challenges in storing all the necessary

| Method | | Task1: Turn Detection | | | Task2: Knowledge Selection | | | | Task3: Response Generation | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Team ID | Entry ID | Precision | Recall | F1 | Precision | Recall | F1 | Exact Match | BLEU | METEOR | ROUGE-1 | ROUGE-2 | ROUGE-L |
| Baseline | | 0.9661 | 0.9979 | 0.9980 | 0.7901 | 0.7877 | 0.7889 | 0.3906 | 0.1004 | 0.1748 | 0.3520 | 0.1430 | 0.2753 |
| 6 | 0 | 0.9968 | **0.9996** | 0.9982 | 0.8039 | 0.8775 | 0.8391 | 0.5547 | 0.1017 | **0.1894** | 0.3629 | 0.1478 | 0.2804 |
| **13 (Ours)** | 0 | 0.9964 | 0.9982 | 0.9973 | 0.8341 | 0.8716 | 0.8524 | **0.6567** | 0.1024 | 0.1826 | 0.3638 | 0.1524 | 0.2868 |
| | 1 | 0.9964 | 0.9982 | 0.9973 | 0.8511 | 0.8581 | **0.8546** | 0.6474 | 0.1017 | 0.1830 | 0.3630 | 0.1530 | 0.2870 |
| | 2 | 0.9964 | 0.9982 | 0.9973 | **0.8590** | 0.8449 | 0.8519 | 0.6432 | 0.1017 | 0.1819 | 0.3618 | 0.1514 | 0.2865 |
| | 3 | 0.9964 | 0.9982 | 0.9973 | **0.8590** | 0.8449 | 0.8519 | 0.6432 | **0.1081** | 0.1819 | 0.3652 | 0.1528 | 0.2872 |
| | 4 | 0.9964 | 0.9982 | 0.9973 | **0.8590** | 0.8449 | 0.8519 | 0.6432 | 0.0931 | 0.1840 | 0.3591 | 0.1484 | 0.2808 |
| 14 | 0 | 0.9979 | 0.9989 | 0.9984 | 0.7856 | 0.8035 | 0.7944 | 0.4183 | 0.1066 | 0.1748 | 0.3599 | 0.1577 | 0.2899 |

Table 5: Objective evaluation results in DSTC11 track 5.

| Rank | Team ID | Entry ID | Accuracy | Appropriateness | Average |
|---|---|---|---|---|---|
| Ground-truth | | | 2.9189 | 3.6422 | 3.2806 |
| 1 | 6 | 0 | 2.9095 | **3.6596** | **3.2846** |
| 2 | 8 | 0 | 2.9005 | 3.6535 | 3.2770 |
| **3** | **13 (Ours)** | **3** | **2.9100** | 3.6321 | 3.2710 |
| 4 | 2 | 3 | 2.8908 | 3.6487 | 3.2697 |
| 5 | 7 | 4 | 2.9046 | 3.6348 | 3.2697 |
| 6 | 12 | 2 | 2.8856 | 3.6518 | 3.2687 |
| 7 | 14 | 0 | 2.8912 | 3.6427 | 3.2670 |
| Baseline | | | 2.8715 | 3.6348 | 3.2531 |

Table 6: Human evaluation results in DSTC11 track 5.

knowledge to generate desired responses, making it impractical in real-world scenarios. Therefore, there are lots of attempts to go beyond the limitations of APIs by utilizing knowledge from unstructured documents. DSTC9 utilized domain FAQ knowledge (Kim et al., 2020), while Chen et al. (2022) employed knowledge from Wikipedia articles, both of which served as external knowledge sources rooted in factual information. However, in DSTC11 track 5, the goal is to use user review data as subjective knowledge to meet the user's needs in a TOD system.

Knowledge-grounded dialogue generation is not only relying on the dialogue context but also leveraging external knowledge (Liu et al., 2018; Dinan et al., 2019). By leveraging external knowledge alongside the dialogue context, it is possible to generate more meaningful responses to users. To provide meaningful answers, it is crucial to obtain relevant external knowledge, making this task akin to a form of retrieval task. The key is how to determine the relevance score between a request and external knowledge, similar to the significance of assessing the relevance score between a query and a document in Question Answering task. There are two ways to measure relevance scores using dense vectors. First, there is the method of using bi-encoder models (Das et al., 2016; Chang et al., 2020; Karpukhin et al., 2020). By employing bi-encoder models, the relevance score can be calculated using inner product or cosine distance. However, the bi-encoder model lacks cross-attention between the query and document, which limits the ability to capture deeper relationships. Another method for calculating relevance scores is to utilize a cross-encoder model (He et al., 2021). This allows token-level interaction between the query and document, resulting in better retrieval performance. In this work, we used the cross-encoder approach to retrieve knowledge relevant to the user's utterance.

## 6 Conclusion

In this work, we proposed a pipeline system for DSTC11 track 5: Task-oriented Conversational Modeling with Subjective Knowledge, mainly focusing on entity tracking, knowledge selection and response generation. Our approach involves the following methods: (1) an heuristic ensemble method to combine the outputs from the rule-based method and neural model, (2) a knowledge selection method based on metadata information, and (3) data augmentation method to tackle mixed opinions in review snippets. As a final result, our submission ranked the top in objective evaluation and the third place in human evaluation.

## Acknowledgement

K. Jung is with ASRI, Seoul National University, Korea. This work was supported in part by LG AI Research. This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) [NO.2021-0- 02068, Artificial Intelligence Innovation Hub (Artificial Intelligence Institute, Seoul National University) & NO.2021-0-01343, Artificial Intelligence Graduate School Program (Seoul National University)]

## References


Wei-Cheng Chang, Felix X. Yu, Yin-Wen Chang, Yiming Yang, and Sanjiv Kumar. 2020. Pre-training tasks for embedding-based large-scale retrieval. In *International Conference on Learning Representations*.

Zhiyu Chen, Bing Liu, Seungwhan Moon, Chinnadhurai Sankar, Paul Crook, and William Yang Wang. 2022. KETOD: Knowledge-enriched task-oriented dialogue. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2581–2593, Seattle, United States. Association for Computational Linguistics.

Arpita Das, Harish Yenala, Manoj Chinnakotla, and Manish Shrivastava. 2016. Together we stand: Siamese networks for similar question retrieval. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 378–387, Berlin, Germany. Association for Computational Linguistics.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of wikipedia: Knowledge-powered conversational agents. In *International Conference on Learning Representations*.

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, Adarsh Kumar, Anuj Goyal, Peter Ku, and Dilek Hakkani-Tur. 2020. MultiWOZ 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 422–428, Marseille, France. European Language Resources Association.

Donghoon Ham, Jeong-Gwan Lee, Youngsoo Jang, and Kee-Eung Kim. 2020. End-to-end neural pipeline for goal-oriented dialogue systems using gpt-2. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 583–592.

Huang He, Hua Lu, Siqi Bao, Fan Wang, Hua Wu, Zhengyu Niu, and Haifeng Wang. 2021. Learning to select external knowledge with multi-scale negative sampling.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTav3: Improving deBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*.

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.

Di Jin, Seokhwan Kim, and Dilek Hakkani-Tur. 2021. Can I be of further assistance? using unstructured knowledge access to improve task-oriented conversational modeling. In *Proceedings of the 1st Workshop on Document-grounded Dialogue and Conversational Question Answering (DialDoc 2021)*, pages 119–127, Online. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Seokhwan Kim, Mihail Eric, Karthik Gopalakrishnan, Behnam Hedayatnia, Yang Liu, and Dilek Hakkani-Tur. 2020. Beyond domain APIs: Task-oriented conversational modeling with unstructured knowledge access. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 278–289, 1st virtual meeting. Association for Computational Linguistics.

Hyukhun Koh, Haesung Pyun, Nakyeong Yang, and Kyomin Jung. 2023. Multi-view zero-shot open intent induction from dialogues: Multi domain batch and proxy gradient transfer. *arXiv preprint arXiv:2303.13099*.

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.

Yohan Lee. 2021. Improving end-to-end task-oriented dialog system with a simple auxiliary task. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1296–1303, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Bing Liu and Ian Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. *Interspeech 2016*, pages 685–689.

Shuman Liu, Hongshen Chen, Zhaochun Ren, Yang Feng, Qun Liu, and Dawei Yin. 2018. Knowledge diffusion for neural dialogue generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1489–1498, Melbourne, Australia. Association for Computational Linguistics.

Elnaz Nouri and Ehsan Hosseini-Asl. 2018. Toward scalable neural dialogue state tracking. In *NeurIPS 2018, 2nd Conversational AI workshop*.

Baolin Peng, Xiujun Li, Lihong Li, Jianfeng Gao, Asli Celikyilmaz, Sungjin Lee, and Kam-Fai Wong. 2017. Composite task-completion dialogue policy learning

via hierarchical deep reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2231–2240, Copenhagen, Denmark. Association for Computational Linguistics.

Baolin Peng, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, and Jianfeng Gao. 2020. Few-shot natural language generation for task-oriented dialog. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 172–182, Online. Association for Computational Linguistics.

Tsung-Hsien Wen, Milica Gašić, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1711–1721, Lisbon, Portugal. Association for Computational Linguistics.

Chien-Sheng Wu, Richard Socher, and Caiming Xiong. 2019. Global-to-local memory pointer networks for task-oriented dialogue. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Steve Young, Milica Gašić, Blaise Thomson, and Jason D Williams. 2013. Pomdp-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*, 101(5):1160–1179.

Chao Zhao, Spandana Gella, Seokhwan Kim, Di Jin, Devamanyu Hazarika, Alexandros Papangelis, Behnam Hedayatnia, Mahdi Namazifar, Yang Liu, and Dilek Hakkani-Tur. 2023. "what do others think?": Task-oriented conversational modeling with subjective knowledge.