Construction of Hierarchical Structured Knowledge-based Recommendation Dialogue Dataset and Dialogue System

Takashi Kodama, Ribeka Tanaka,* Sadao Kurohashi
Graduate School of Informatics, Kyoto University
Yoshida-honmachi, Sakyo-ku, Kyoto, 606-8501, Japan
{kodama, kuro}@nlp.ist.i.kyoto-u.ac.jp

Abstract

We work on a recommendation dialogue system to help a user understand the appealing points of some target (e.g., a movie). In such dialogues, the recommendation system needs to utilize structured external knowledge to make informative and detailed recommendations. However, there is no dialogue dataset with structured external knowledge designed to make detailed recommendations for the target. Therefore, we construct a dialogue dataset, Japanese Movie Recommendation Dialogue (JMRD), in which the recommender recommends one movie in a long dialogue (23 turns on average). The external knowledge used in this dataset is hierarchically structured, including title, casts, reviews, and plots. Every recommender's utterance is associated with the external knowledge related to the utterance. We then create a movie recommendation dialogue system that considers the structure of the external knowledge and the history of the knowledge used. Experimental results show that the proposed model is superior in knowledge selection to the baseline models.

1 Introduction

In recent years, research on recommendation dialogue systems, which systems recommend something to users through dialogues, has attracted much attention. Here, we focus on movie recommendations. A recommendation dialogue consists of two phases: (1) the user's preferences are elicited, and a movie is selected from several candidates, and (2) in-depth information is provided for the selected movie. We focus on the latter phase in this study.

To provide in-depth information, the use of external knowledge is crucial. There has been much research on incorporating external knowledge in dialogue, and many kinds of knowledge-grounded dialogue datasets have been proposed (Dinan et al., 2019; Liu et al., 2020). These datasets often use plain texts or knowledge graphs as external knowledge. If the hierarchically structured knowledge is available in recommendation dialogues, it allows for more appropriate knowledge selection and informative response generation. However, there is no dialogue dataset with hierarchically structured knowledge to provide rich information for a single target (e.g., a movie).

To address the aforementioned problem, we propose a dialogue dataset, Japanese Movie Recommendation Dialogue (**JMRD**), in which recommendation dialogues are paired with the corresponding external knowledge. This dialogue dataset consists of about 5,200 dialogues between crowd workers. Each dialogue has 23 turns on average. We can say that our dataset provides in-depth movie recommendations utilizing various knowledge about a movie, with relatively a large number of dialogue turns. Specifically, as shown in Figure 1, one speaker (recommender) recommends a movie to the other speaker (seeker). Only the recommenders can have access to the knowledge about the movie, and they should use the external knowledge as much as possible in their utterances. The recommenders are asked to annotate the knowledge they used when sending their utterance. This procedure enables us to associate every recommenders' utterances with the corresponding external knowledge. The external knowledge is hierarchically structured into knowledge types common to all movies (e.g., "Title", "Released Year") and giving knowledge contents for each movie (e.g., "Rise of Planet of the Apes", "August 5, 2011").

We also propose a strong baseline model for the constructed dataset. This model considers the history of knowledge types/contents, noting that the order in which each piece of knowledge is used is essential in recommendation dialogues. The exper-

*Currently affiliated with Ochanomizu University, 2-1-1 Otsuka, Bunkyo-ku, Tokyo, 112-8610, Japan (E-mail: tanaka.ribeka@is.ocha.ac.jp).

Proceedings of the Second DialDoc Workshop on Document-grounded Dialogue and Conversational Question Answering, pages 83 - 92

imental results show that our proposed model can select appropriate knowledge with higher accuracy than the baseline method.

Our contributions are three-fold.

- We construct a movie recommendation dialogue dataset associated with hierarchically structured external knowledge.
- We propose a strong baseline model, which selects knowledge based on hierarchically structured knowledge, for our dataset.
- To the best of our knowledge, we are the first to construct a human-to-human dialogue dataset based on external knowledge in Japanese.

2 Related Work

Recommendation dialogue has long attracted attention. However, most of them are goal-oriented dialogues in which the user's preferences are elicited from multiple recommendation candidates, and a recommendation target is decided according to that preferences (Bordes et al., 2017; Li et al., 2018). Li et al. (2018) propose REDIAL, a human-to-human movie recommendation dialogue dataset. The recommender presents several movies in one dialogue while inquiring about the seeker's preferences. Kang et al. (2019) collect GoRecDial dataset in a gamified setting where experts decide on a movie similar to the seekers' preference among a small set of movies (= five movies) in a minimal number of turns. OpenDialKG (Moon et al., 2019) is a recommendation and chit-chat dataset linking open-ended dialogues to knowledge graphs. In this study, we focus on the recommendation dialogue, which provides in-depth information about a movie rather than deciding which movie to recommend.

Research on the knowledge-grounded dialogue has also been growing in the last few years. Zhou et al. (2018) collect a human-to-human chit-chat dialogue dataset by utilizing Wikipedia articles of 30 famous movies. This dataset is unique in that it has two dialogue settings: either only one of the participants can see the knowledge, or both of them can see it. Moghe et al. (2018) also collect chit-chat dialogues about movies based on multiple types of knowledge: plot, review, Reddit comments, and fact table. Wizard of Wikipedia (Dinan et al., 2019) is an open-domain chit-chat dialogue dataset based on Wikipedia articles on 1,365 topics. It has become a standard benchmark in this research field. Su et al. (2020) collect a large Chinese chit-chat dialogue dataset (246,141 dialogues with 3,010,650 turns) about movies. Other dialogue datasets with external knowledge in Chinese are DuConv (Wu et al., 2019), KdConv (Zhou et al., 2020), and DuRecDial (Liu et al., 2020). DuConv (Wu et al., 2019) combines dialogues with knowledge graphs to track the progress of the dialogue topic. KdConv (Zhou et al., 2020) is also a chit-chat dialogue corpus that consists of relatively long dialogues to allow deep discussions in multiple domains (movies, music, and travel). Liu et al. (2020) focus on multiple dialogue types (e.g., QA, chit-chat, recommendation) and collect a multi-domain dialogue dataset associated with a knowledge graph. Compared to these studies, our work differs in that it uses hierarchically structured knowledge that contains both factoid (e.g., title) and non-factoid (e.g., review) information to make recommendations.

3 Japanese Movie Recommendation Dialogue

We choose movies as the domain for the recommendation dialogue because movies are interesting to everyone and facilitate smooth dialogue. In addition, movie recommendation dialogue is open-domain in nature according to the variety of movie topics, and it is a preferable property for NLP research. In this section, we explain the construction method of the JMRD.

3.1 External Knowledge Collection

The external knowledge is mainly collected from web texts such as Wikipedia. First, we select 261 movies based on the box-office revenue ranking. [1] For each of these movies, we collect movie information as external knowledge.

The external knowledge consists of seven knowledge types: title, released year, director, cast, genre, review, and plot, as shown in Figure 1. The title, released year, director, cast, and plot are extracted from the Wikipedia article of each movie (we allow at most one director and two casts). For the director and the casts, a brief description is also extracted from the first paragraph of each person's Wikipedia article. For the genre, we use the genre classification of Yahoo! Movies. [2] Reviews are collected

[1] http://www.eiren.org/toukei/index.html

[2] https://movies.yahoo.co.jp/

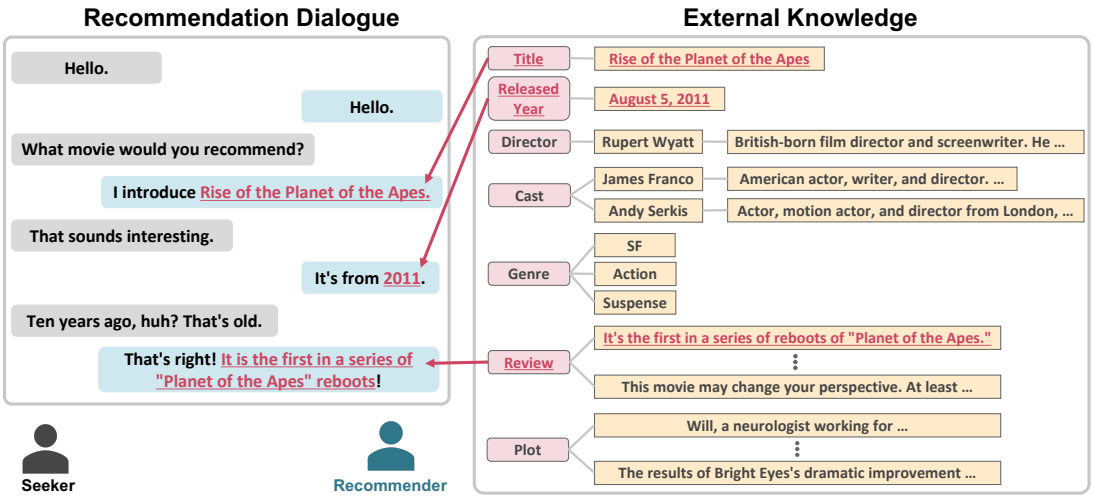

Figure 1: An example of JMRD dataset. The underlined parts of the external knowledge indicate the knowledge items used in the dialogue.

by crowdsourcing using Yahoo! Crowdsourcing. [3] Each worker selects a movie that he or she has seen from a list of 261 movies and writes down three recommendations for the selected movie. As a result, we collected an average of 16.5 reviews per movie.

We split the plot into sentences and present only the first ten sentences (or all sentences if fewer than ten) to reduce the burden of the recommender. Besides, we use the reviews written by the workers as it is, without splitting the sentences. We randomly selected five reviews between 15 and 80 characters long for each movie from the collected reviews. Those five reviews are used as the reviews for that movie.

3.2 Dialogue Collection

3.2.1 Settings

The two workers engaging in the movie recommendation dialogue have different roles: one is the **recommender**, and the other is the **seeker**. The flow of the dialogue takes place as follows:

1. Either the recommender or seeker can initialize the conversation.

2. The recommender decides which movie to recommend from the movie list. The recommender can choose a movie he or she wants to recommend or a movie that matches the seeker's preference obtained from a few message exchanges. The recommender can access the movie knowledge after deciding the movie to recommend. On the other hand, the seeker is only shown the chat screen and cannot access knowledge about the movie.

3. The recommender is instructed to use the presented knowledge as much as possible to recommend the movie. When the recommender sends their utterance, they must select the knowledge referred to by the utterance (multiple selection is allowed). For the utterance that does not use any knowledge, such as greetings, the recommender can select the "no knowledge" option.

4. The seeker is only instructed to enjoy and learn more about the recommended movie, and they can talk freely. This instruction refers to that of Wizard of Wikipedia (Dinan et al., 2019).

5. The dialogue lasts at least 20 turns after the movie is selected and can be terminated after 20 turns.

3.2.2 Dialogue Collection System

ParlAI (Miller et al., 2017) is a framework for collecting real-time chats in crowdsourcing. However, it is not easy to perform Japanese tasks with the Amazon Mechanical Turk used in ParlAI. Therefore, we build a new framework for dialogue collection, which incorporates crowdsourcing services where more native Japanese speakers can be gathered. In our framework, when workers access the specified URL for dialogue collection, pair match-

[3] https://crowdsourcing.yahoo.co.jp/

# dialogues	5,166
# utterances (R)	57,714
# utterances (S)	59,160
# movies	261
# workers	322
Avg. # turns per dialogue	22.6
Avg. # words per utterance (R)	23.8
Avg. # words per utterance (S)	6.9
Avg. # knowledge used per utterance	1.3
Avg. # knowledge used per dialogue	10.8

Table 1: Statistics of JMRD. R and S denote recommender and seeker respectively. We use Juman++ (Tolmachev et al., 2020) for word segmentation.

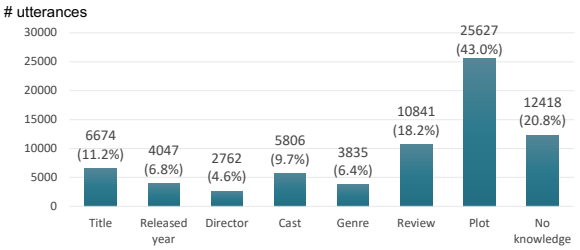

Figure 2: Distribution of external knowledge used.

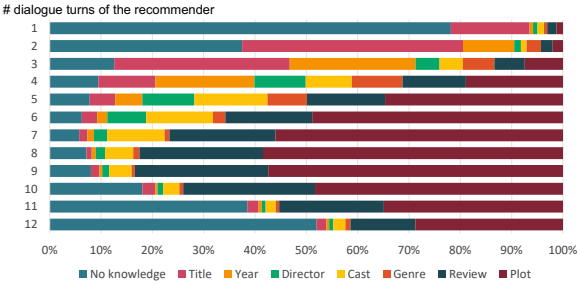

Figure 3: Distribution of external knowledge used in each dialogue turn of the recommender. The information up to turn 12 is shown here.

ing is performed, and a chat room is created for the worker to interact in real-time.

3.2.3 Statistics

The statistics are shown in Table 1. Our dataset consists of 5,166 dialogues with 116,874 turns. The average number of words per utterance of the recommender is more than three times larger than that of the seeker. It is probably because the recommender needs to talk more than the seeker to provide information to recommend a movie. The average number of knowledge items per utterance is 1.3, and the recommender tends to mention each knowledge item separately. There were on average 10.8 different types of knowledge used per dialogue, indicating that we could collect dialogues with various types of external knowledge.

	Q1	Q2	Q3	Q4	Q5
Recommender	4.36	4.00	3.94	4.01	-
Seeker	4.26	3.83	2.72	-	3.82

Table 2: Results of the questionnaire.

Figure 2 shows the distribution of the knowledge types used. The number of utterances that did not use any knowledge was only about 20% of the total, indicating that most utterances use some kind of external knowledge. In addition, non-factoid texts such as reviews and plots tend to be used more frequently.

Furthermore, Figure 3 shows the distribution of the knowledge used in each dialogue turn of the recommender. In the early part of the dialogue, there are many utterances without knowledge, such as greetings or utterances that mention the title. The recommenders often use factoid information such as released year, director, and cast in the middle of the dialogue. In the later part, non-factoid information such as reviews and plots are often used to convey specific content. In addition, after ten turns, the percentage of "No knowledge" increased again, as more generic recommendations such as "please check it out" are used. As can be seen from this analysis, our dataset is capable of analyzing human recommendation strategies.

3.2.4 Post-task Questionnaire

We ask the dialogue participants to answer the following post-task questionnaire in some of the collected dialogues (= 4,410 dialogues).

Q1: Do you like movies?

Q2: Did you enjoy the dialogue?

Q3: Do you know the movie you recommended (or that was recommended to you)?

Q4: Do you think you have recommended the movie well?

Q5: Do you want to watch the recommended movie?

All questions are answered on a 5-point Likert scale, with five being the best and one being the worse. The choices for Q1, Q2, Q4, and Q5 are [agree/somewhat agree/neutral/somewhat disagree/disagree]. The choices for Q3 are [have seen

the movie and remember the contents well/have seen the movie and remember some the contents/have never seen the movie but know the plot/have never seen the movie but know only the title/do not know at all]. Q4 is for recommenders only, and Q5 is for seekers only.

Table 2 shows the results of the questionnaire. We found that most of the workers were highly interested in the topic of movies (Q1), and both recommenders and seekers enjoyed the dialogue, although it was relatively long, more than 20 turns (Q2). In addition, from Q3, we can see that the recommenders recommended movies they knew, while the seekers were often recommended movies they did not know. Finally, from Q4 and Q5, it was confirmed that the collected dialogues sufficiently achieved the purpose of movie recommendation.

4 Proposed Model

4.1 Outline

Each dialogue $\mathcal{D} = \{(x^l, y^l)\}_{l=1}^{L}$ in the dataset is paired with a knowledge pool $\mathcal{K} = (\mathbf{k_t}, \mathbf{k_c})$ about the movie recommended in that dialogue, where x^l, y^l is the utterance of the seeker and recommender at turn l and L is the number of turns in $\mathcal{D}$. In addition, $\mathbf{k_t}$ $(= \{k_{t,1}, \ldots, k_{t,m}, \ldots, k_{t,M}\})$ are the knowledge types, $\mathbf{k_c}$ $(= \{k_{c,1}, \ldots, k_{c,n}, \ldots, k_{c,N}\})$ are knowledge contents, and M, N are the number of knowledge types and knowledge contents contained in $\mathcal{K}$, respectively. At turn l, given the dialogue context (= the current seeker's utterance x^l and the last recommender's utterance y^{l-1}), the previously selected knowledge types $\{\hat{k}_t^1, \ldots, \hat{k}_t^{l-1}\}$, and previously selected knowledge contents $\{\hat{k}_c^1, \ldots, \hat{k}_c^{l-1}\}$, our target is to select a piece of knowledge $\hat{k}_c^l$ from $\mathbf{k_c}$ and generate response y^l utilizing $\hat{k}_c^l$. We call the previously selected knowledge types the "knowledge type history" and the previously selected knowledge contents the "knowledge content history" in this paper.

Figure 4 shows the overview of the proposed model. The proposed model mainly consists of the Encoding Layer, the Knowledge Selection Layer, and the Decoding Layer. We describe each of the components in the following sections.

4.2 Encoding Layer

The encoding layer is used to obtain the following representations: dialogue context, knowledge types, knowledge contents, knowledge type history, and knowledge content history. We use BERT (Devlin et al., 2019) as the encoder. For encoding the dialogue context, we obtain the hidden state $H^{x^l y^{l-1}}$ via BERT, and then perform average pooling to obtain $h^{x^l y^{l-1}}$ (Cer et al., 2018):

$$H^{x^l y^{l-1}} = BERT(x^l, y^{l-1}) \tag{1}$$

$$h^{x^l y^{l-1}} = avgpool(H^{x^l y^{l-1}}) \in \mathbb{R}^d \tag{2}$$

where d is the hidden size. We insert [SEP] between x^l and y^{l-1}, and insert [CLS] and [SEP] at the beginning and the end of the entire input string, respectively.

In the case of knowledge types, we insert [CLS] and [SEP] at the beginning and the end of the input string, respectively. After that, we get $\{h^{k_{t,m}}\}_{m=1}^{M}$ by feeding it to BERT in the same way. For the knowledge contents, we input the knowledge type in addition to the knowledge contents, following the method of Dinan et al. (2019). We insert a new special token [KNOW SEP] between the knowledge type and the knowledge content and further insert [CLS] and [SEP] at the beginning and the end of the input string, respectively. The resulting string is input to BERT to obtain $\{h^{k_{c,n}}\}_{n=1}^{N}$ likewise. We also compute the representations of knowledge type history $\{h^{\hat{k}_t^i}\}_{i=1}^{l-1}$ and that of knowledge content history $\{h^{\hat{k}_c^i}\}_{i=1}^{l-1}$.

4.3 Knowledge Selection Layer

We encode the knowledge type history via the transformer encoder (Vaswani et al., 2017). This transformer encoder (we call this "knowledge type encoder") adds a positional embedding for each turn (= turn embedding) to the input so that the model reflects in which turn each knowledge type was used (Meng et al., 2021). We concatenate the last output of this encoder $h_{trans}^{\hat{k}_t^{l-1}}$ with the hidden state of the dialogue context $h^{x^l y^{l-1}}$ as the query, and regard $\{h^{k_{t,m}}\}_{m=1}^{M}$ as the key. The attention over knowledge types $a_t \in \mathbb{R}^M$ is calculated as follows:

$$\begin{aligned}
a_t &= [a_{t,1}, \ldots, a_{t,m}, \ldots, a_{t,M}] \\
&= softmax(Q_t {K_t}^\top) \\
Q_t &= MLP([h_{trans}^{\hat{k}_t^{l-1}}; h^{x^l y^{l-1}}]) \\
K_t &= MLP([h^{k_{t,1}}, \ldots, h^{k_{t,M}}]) \\
[h_{trans}^{\hat{k}_t^1}, \ldots, h_{trans}^{\hat{k}_t^{l-1}}] &= KTE([h^{\hat{k}_t^1}, \ldots, h^{\hat{k}_t^{l-1}}])
\end{aligned} \tag{3}$$

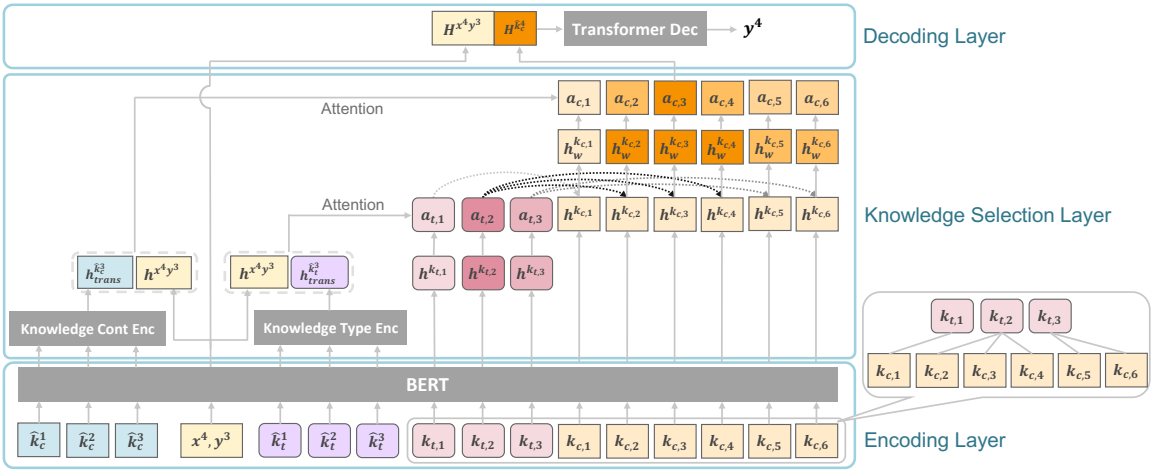

Figure 4: Overview of the proposed model. In this figure, the model generates the response y^4 at time $l = 4$. Knowledge Cont Enc, Knowledge Type Enc, and Transformer Dec denote the knowledge content encoder, the knowledge type encoder, and the transformer decoder, respectively.

where $MLP(\cdot)$ is a multilayer perceptron, KTE is the knowledge type encoder, and $[\cdot;\cdot]$ is the vector concatenation operation.

We compute the weighted hidden state of the knowledge contents $\{h_w^{k_{c,n}}\}_{n=1}^N$ based on the calculated attention a_t. This weighted hidden state is used to calculate the attention over the knowledge contents. Suppose the number of knowledge contents belonging to the m-th knowledge type is N_m, and the same weight $a_{t,m} \in a_t$ is given to all of them. In that case, the M-dimensional a_t can be extended to the N-dimensional $a'_t \in \mathbb{R}^N$ as follows, because N_m satisfies $\sum_{m=1}^M N_m = N$:

$$a'_t = [a_{t,1}, \ldots, \underbrace{a_{t,m}, \ldots, a_{t,m}}_{N_m}, \ldots, a_{t,M}] \quad (4)$$

Using a'_t, the weighted hidden states of the knowledge contents $\{h_w^{k_{c,n}}\}_{n=1}^N$ can be obtained as follows:

$$[h_w^{k_{c,1}}, \ldots, h_w^{k_{c,N}}] = a'_t[h^{k_{c,1}}, \ldots, h^{k_{c,N}}] \quad (5)$$

The knowledge content history is encoded by the transformer encoder as well. This transformer encoder, which we call "knowledge content encoder", has the same setting as the knowledge type encoder, but they do not share any parameters. We concatenate the last output of the encoder $h_{trans}^{\hat{k}_c^{l-1}}$ with $h^{x^l y^{l-1}}$ as the query, and regard the weighted hidden states of knowledge contents $\{h_w^{k_{c,n}}\}_{n=1}^N$ as the key. We can calculate the attention over the knowledge contents $a_c \in \mathbb{R}^N$ as follows:

$$\begin{aligned} a_c &= softmax(Q_c {K_c}^\top) \\ Q_c &= MLP([h_{trans}^{\hat{k}_c^{l-1}}; h^{x^l y^{l-1}}]) \\ K_c &= MLP([h_w^{k_{c,1}}, \ldots, h_w^{k_{c,N}}]) \\ [h_{trans}^{\hat{k}_c^1}, \ldots, h_{trans}^{\hat{k}_c^{l-1}}] &= KCE([h^{\hat{k}_c^1}, \ldots, h^{\hat{k}_c^{l-1}}]) \end{aligned} \quad (6)$$

where KCE is the knowledge content encoder. Finally, we select a knowledge content $\hat{k}_c^l$ at time l from the probability distribution of a_c.

4.4 Decoding Layer

At time l, the dialogue context x^l, y^{l-1} and the knowledge content $\hat{k}_c^l$ selected by the knowledge selection layer, are input to the transformer decoder to generate the response y^l. Specifically, we feed the concatenated embedding $H^{x^l y^{l-1} \hat{k}_c^l} = [H^{x^l y^{l-1}}; H^{\hat{k}_c^l}]$ to the decoder. The word generation probability $p(y_j^l)$ over the vocabulary V when the decoder generates the j-th word can be written as follows:

$$\begin{aligned} p(y_j^l) &= softmax(MLP(h_{dec}^{l,j})) \in \mathbb{R}^{1\times|V|} \\ h_{dec}^{l,j} &= TD(H^{x^l y^{l-1} \hat{k}_c^l}, emb(y_{<j}^l))) \in \mathbb{R}^{1\times d} \end{aligned} \quad (7)$$

where TD is the transformer decoder, $y_{<j}^l$ are the words generated up to the j-th word, $emb(y_{<j}^l)$ are the word embeddings of $y_{<j}^l$, which is initialized with the word embedding of BERT.

We use copy mechanism (Gu et al., 2016; See et al., 2017) to make it easier to generate knowledge

words and follow the method used in Meng et al. (2021).

4.5 Learning Objective

Similar to Dinan et al. (2019), we combine the negative log-likelihood loss for the generated response $\mathcal{L}_{nll}$ with the cross-entropy loss for knowledge selection $\mathcal{L}_{knowledge}$ modulated by a weight λ, which is the hyperparameter. The final loss function $\mathcal{L}$ is as:

$$\mathcal{L} = (1-\lambda)\mathcal{L}_{nll} + \lambda\mathcal{L}_{knowledge} \quad (8)$$

5 Experiments

5.1 Settings

We randomly split the dialogues into the train (90%), validation (5%), and test sets (5%). Input texts are truncated to the maximum input length of 64 tokens for dialogue contexts and knowledge contents and 5 tokens for knowledge types. In addition, a maximum of 20 turns of knowledge history can be entered for both knowledge types and knowledge contents. Our proposed dataset may have multiple pieces of knowledge associated with a recommender's utterance, but we use only one of them in this study for simplicity. In the case of an utterance with multiple knowledge items, we select the one with the highest Jaccard coefficient in the word set of the recommender's utterance and each knowledge as the correct knowledge. To input "No knowledge," we use the special token [NO KNOW] in place of knowledge type and content.

5.2 Baseline

We use an end-to-end Transformer Memory Network (TMN) (Dinan et al., 2019) as baseline. This model encodes the dialogue context and each knowledge respectively and selects knowledge by calculating the dot-product attention between them. It also performs end-to-end response generation using the selected knowledge. To make a fair comparison with our proposed model, we have replaced the original transformer encoder with a BERT encoder. We call this model TMN BERT.

As a baseline to consider knowledge history, we add the knowledge content encoder to TMN BERT and concatenate its output with the hidden states of the dialogue context. We call this model TMN BERT+KH. Knowledge selection is made by calculating the attention between the knowledge candidates and the concatenated hidden states. Other conditions are the same as in TMN BERT.

In addition, we use Random baseline that selects knowledge randomly.

5.3 Implementation Details

We use the NICT BERT Japanese pre-trained model (with BPE) [4] as the encoder. This BERT is also used to initialize the word embedding in the transformer decoder. The transformer encoders for knowledge type and knowledge content, and the transformer decoder have the same architecture, consisting of 2 attention heads, 5 layers, and the size of the hidden layer is 768 and the filter size is 3072. We train the models for 100 epochs with a batch size of 512 and 0.1 gradient clipping. We do early stopping if no improvement of the validation loss is observed for five consecutive epochs. All models are learned with Adam optimizer (Kingma and Ba, 2015) with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and an initial learning rate = 0.00005. We use an inverse square root learning rate scheduler with the first 1,000 steps allocated for warmup. In addition, we set the hyperparameter λ to 0.95. At decoding, we use beam search with a beam of size 3. We add a restriction to prevent the same bigram from being generated multiple times.

5.4 Evaluation Metrics

We evaluate the models with automatic evaluation metrics. For knowledge selection, we use accuracy (**Acc**). For response reproducibility, we measure **BLEU**$_{tgt}$**-4** (Papineni et al., 2002), which is the 4-gram overlap between a generated response and a target response. We also use unigram F1 (**F1**) following the evaluation setting in Dinan et al. (2019). Additionally, we use **Jaccard** and **BLEU**$_{know}$**-4** to evaluate whether the knowledge is reflected in the generated response. **Jaccard** is the Jaccard coefficient of the set of words in the generated response and the set of words in the selected knowledge content. **BLEU**$_{know}$**-4** is the BLEU-4 computed between the generated response and the selected knowledge content.

5.5 Results and Analysis

The results of knowledge selection are shown in Table 3. The results show that our proposed method outperformed the baselines. TMN BERT+KH, which adds a mechanism to consider knowledge history to the baseline TMN BERT, is almost the same as TMN BERT in Acc. On the other hand,

[4] https://alaginrc.nict.go.jp/nict-bert/index.html

	knowledge selection	response reproducibility		knowledge reflection	
	Acc	F1	BLEU$_{tgt}$-4	Jaccard	BLEU$_{know}$-4
Random	4.18 (0.15)	24.05 (0.26)	4.63 (0.16)	5.87 (0.18)	0.47 (0.07)
TMN BERT	48.81 (0.25)	**42.97** (0.16)	**21.03** (0.70)	38.36 (0.81)	24.94 (1.36)
TMN BERT+KH	48.66 (0.06)	42.74 (0.46)	20.68 (0.56)	38.23 (0.94)	25.08 (1.29)
Ours	**49.72** (0.44)	42.92 (0.71)	20.78 (0.69)	**39.35** (1.41)	**25.88** (1.35)

Table 3: The evaluation results. Scores are the mean of three runs of the experiment with different random seeds, and standard deviations are shown in parentheses. The bold scores indicate the best ones over models.

	Dialogue	Knowledge
Recommender$_1$:	Nice to meet you.	No knowledge
Seeker$_1$:	Hello.	-
Recommender$_2$:	I am pleased to meet you.	No knowledge
Seeker$_2$:	What movies do you recommend?	-
TMN BERT	I will introduce a movie called Do You Like Disney Movies?	Danny Ocean immediately breaks his parole rules (no interstate movement) and reunites with his partner Rusty Ryan in Los Angeles. He confides in Ryan about a new theft scheme he had hatched while in prison. (Plot)
Ours:	Today I will introduce Ocean's Eleven.	Ocean's Eleven (Title)
Gold:	How about Ocean's Eleven?	Ocean's Eleven (Title)

Table 4: Examples of generated responses by our model and the baseline model. Subscript numbers indicate the number of turns in the dialogue. The knowledge type is indicated in parentheses in the knowledge column.

our proposed method improves Acc, suggesting the importance of considering knowledge structurally.

The results of response generation are also shown in Table 3. The proposed method did not perform well in terms of reproducibility for target responses. However, this should not be a major problem because it is known that it is inappropriate to measure reproducibility in dialogue evaluation (Liu et al., 2016). On the other hand, the proposed model performed the best for knowledge reflection. We believe this improvement is due to selecting knowledge more correctly according to the dialogue context and knowledge history.

5.6 Case Study

Table 4 shows an example of knowledge selection and response generation. TMN BERT, which does not consider knowledge history, selects the plot even though it is at the beginning of the dialogue. Moreover, the generated utterance does not reflect the selected knowledge. On the other hand, our proposed model introduces the movie title that has not yet been mentioned in this dialogue by considering the knowledge history.

As illustrated by the generated response of TMN BERT, the generated utterances may not reflect the selected knowledge or may contain words inconsistent with the selected knowledge. This problem is known as the hallucination problem (Roller et al., 2020; Shuster et al., 2021), and we leave the solution to this problem as future work.

6 Conclusion

We proposed JMRD, a hierarchically structured knowledge-based movie recommendation dialogue dataset. We also proposed an end-to-end dialogue system that utilizes the hierarchically structured knowledge of knowledge types and contents to perform knowledge selection and generate responses as a strong baseline for our dataset. The experimental results show that our model can select more appropriate knowledge than baselines.

As far as we know, this is the first Japanese dialogue dataset associated with external knowledge. We hope our dataset facilitates further research on movie recommendation dialogue based on structured external knowledge (especially in Japanese dialogue research).

In response generation, we can observe that the utterances do not reflect the knowledge in some cases, even when the knowledge is selected correctly. There is still much room for improvement in knowledge reflection, and we leave this as future work.

7 Acknowledgments

This work was supported by NII CRIS collaborative research program operated by NII CRIS and LINE Corporation. This work was also supported by JST, CREST Grant Number JPMJCR20D2, Japan.

References

Antoine Bordes, Y-Lan Boureau, and Jason Weston. 2017. Learning end-to-end goal-oriented dialog. In *ICLR*.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174, Brussels, Belgium. Association for Computational Linguistics.

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

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1640, Berlin, Germany. Association for Computational Linguistics.

Dongyeop Kang, Anusha Balakrishnan, Pararth Shah, Paul Crook, Y-Lan Boureau, and Jason Weston. 2019. Recommendation as a communication game: Self-supervised bot-play for goal-oriented dialogue. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1951–1961, Hong Kong, China. Association for Computational Linguistics.

Diederick P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

Raymond Li, Samira Kahou, Hannes Schulz, Vincent Michalski, Laurent Charlin, and Chris Pal. 2018. Towards deep conversational recommendations. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pages 9748–9758, Red Hook, NY, USA. Curran Associates Inc.

Chia-Wei Liu, Ryan Lowe, Iulian Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2122–2132, Austin, Texas. Association for Computational Linguistics.

Zeming Liu, Haifeng Wang, Zheng-Yu Niu, Hua Wu, Wanxiang Che, and Ting Liu. 2020. Towards conversational recommendation over multi-type dialogs. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1036–1049, Online. Association for Computational Linguistics.

Chuan Meng, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tengxiao Xi, and Maarten de Rijke. 2021. Initiative-aware self-supervised learning for knowledge-grounded conversations. In *SIGIR*, pages 522–532.

Alexander Miller, Will Feng, Dhruv Batra, Antoine Bordes, Adam Fisch, Jiasen Lu, Devi Parikh, and Jason Weston. 2017. ParlAI: A dialog research software platform. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 79–84, Copenhagen, Denmark. Association for Computational Linguistics.

Nikita Moghe, Siddhartha Arora, Suman Banerjee, and Mitesh M. Khapra. 2018. Towards exploiting background knowledge for building conversation systems. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2322–2332, Brussels, Belgium. Association for Computational Linguistics.

Seungwhan Moon, Pararth Shah, Anuj Kumar, and Rajen Subba. 2019. OpenDialKG: Explainable conversational reasoning with attention-based walks over knowledge graphs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 845–854, Florence, Italy. Association for Computational Linguistics.

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

Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott,

Kurt Shuster, Eric Michael Smith, Y-Lan Boureau, and Jason Weston. 2020. Recipes for building an open-domain chatbot. abs/2004.13637.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. abs/2104.07567.

Hui Su, Xiaoyu Shen, Zhou Xiao, Zheng Zhang, Ernie Chang, Cheng Zhang, Cheng Niu, and Jie Zhou. 2020. MovieChats: Chat like humans in a closed domain. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6605–6619, Online. Association for Computational Linguistics.

Arseny Tolmachev, Daisuke Kawahara, and Sadao Kurohashi. 2020. Design and structure of the Juman++ morphological analyzer toolkit. *Journal of Natural Language Processing*, 27(1):89–132.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*, pages 5998–6008. Curran Associates Inc.

Wenquan Wu, Zhen Guo, Xiangyang Zhou, Hua Wu, Xiyuan Zhang, Rongzhong Lian, and Haifeng Wang. 2019. Proactive human-machine conversation with explicit conversation goal. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3794–3804, Florence, Italy. Association for Computational Linguistics.

Hao Zhou, Chujie Zheng, Kaili Huang, Minlie Huang, and Xiaoyan Zhu. 2020. KdConv: A Chinese multi-domain dialogue dataset towards multi-turn knowledge-driven conversation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7098–7108, Online. Association for Computational Linguistics.

Kangyan Zhou, Shrimai Prabhumoye, and Alan W Black. 2018. A dataset for document grounded conversations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 708–713, Brussels, Belgium. Association for Computational Linguistics.