3 OUR APPROACH

In this section, we introduce our budget argument mining approach. We employ a pre-trained model for the budget argument mining task because pre-trained models such as transformer-based models achieved SOTA in several tasks recently. In NTCIR-16 poliinfo budget argument mining task consists of two subtasks, identifying “relatedID” and “argumentClass” of money expressions. The task of identifying the “relatedID” is to identify which budget the input money expression is related to. For example, “約28700万円” (about 287 million yen) is a money expression that appears in the utterance of assembly of Otaru City. The second task of identification of “argumentClass” is to classify input money expression into 7 classes, “Premise :過去・決定事項,” “Premise :未来（現在以降）・見積,” “Premise :その他（例示・訂正事項など),” “Claim :意見・提案・質問,” “Claim :その他” “金額表現ではない,” and “その他.” For example, the money expression “約28700万円” (about 287 million yen) is classified into the class “Premise :未来（現在以降）・見積.”

In the minutes data for this task has structured for each utterance in each meeting. Also, the utterance structured by its property such as uttered person or the money expressions mentioned above. In the supervised data, the money expression is linked to its budget data and classified into “argumentClass.”

To address these two subtasks, we assumed that the documents of utterance at the assembly and the budget summary of the same government were similar by comparing some examples extracted from the supervised data.

Therefore, we attempt to identify “relatedID” and “argumentClass” using bidirectional encoder representations from transformers (BERT) [1] and multi-task learning. In this research, we use BERT based on Wikipedia. By we conduct fine-tuning of BERT, we identify “relatedID” and “argumentClass” of input money expressions.

We create a model for identification of “relatedID” and “argumentClass” using BERT. We illustrate our model in Figure 1. Our model has two inputs, input three sentences, and the description of the target budget. In this task, we have to search budget ID related to input money expressions. Therefore, we decided to find budget ID using cosine similarity based on vectors of BERT.

From Figure 1, input three sentences consist of the sentence including the money expression and before and after the sentence including the money expression. Our model is multi-task learning that consists of identification of “relatedID” and “argumentClass.” Therefore, our model has two loss functions. One of the loss functions is cosine similarity loss related to the identification of “relatedID.”
Another one is cross-entropy loss related to the identification of “argument Class.”

We describe the flow of your model training. First, our model obtains vectors of BERT from inputs. Then, we obtain two types of the average of vectors, vectors based on the input three sentences and the description of the target budget. We calculates cosine similarity loss using obtained vectors and labeled data. Additionally, using labeled data and obtained average vectors based on input three sentences, we calculate the cross-entropy loss.

4 EXPERIMENTS

Table 1 is a official score of the formal run. In the table 1, the “Argument Class” and “Related ID” are a accuracy of each subtask, and “Score” is overall score that is calculated from these two scores.

4.1 Results & Discussion

As a result, our method was a worst because we could not find any budget (Related ID) for all money expressions. Thus our assumption that sentences appearing near money expressions will contain a similar expressions to the budget summary was declined. This means we need any kind of knowledge (e.g. a dictionary for paraphrase or any kind of knowledge base for government) to handle the utterance at assembly and the budget document at the same time.