# KSU Team's Dialogue System at the NTCIR-13 Short Text Conversation Task 2

Yoichi Ishibashi
Kyoto Sangyo University, Japan
g1445539@cc.kyoto-su.ac.jp

Sho Sugimoto
Kyoto Sangyo University, Japan
g1444693@cc.kyoto-su.ac.jp

Hisashi Miyamori
Kyoto Sangyo University, Japan
miya@cc.kyoto-su.ac.jp

## ABSTRACT

In this paper, the methods and results by the team KSU for STC-2 task at NTCIR-13 are described. We implemented both retrieval-based methods and a generation-based method. In the retrieval-based methods, a comment text with high similarity with the given utterance text is obtained from Yahoo! News comments data, and the reply text to the comment text is returned as the response to the input. Two methods were implemented with different information used for retrieval. It was confirmed that the precision of response selection was improved by selectively using some information on news articles on which the dialogue was based. In the generation-based method, we propose the Associative Conversation Model that generates visual information from textual information and uses it for generating sentences in order to utilize visual information in a dialogue system without image input. In research on Neural Machine Translation, there are studies that generate translated sentences using both images and sentences, and these studies show that visual information improves translation performance. However, it is not possible to use sentence generation algorithms using images for the dialogue systems since many text-based dialogue systems only accept text input. Our approach generates (associates) visual information from input text and generates response text using context vector fusing associative visual information and textual information. As a preliminary result, it was confirmed that visual information seemed to work effectively in several examples.

## Team Name

KSU

## Subtasks

Japanese Subtask

## Keywords

BM25, encoder-decoder model, multimodal learning

## 1. INTRODUCTION

In NTCIR-13[1] Short Text Conversation task 2 [14] (hereafter referred to as "STC-2"), a non-task-oriented dialogue system is required that can return response sentences with high evaluation in some viewpoints for a given utterance text. We submitted three runs: two retrieval-based methods in Run 1 and Run 3, and a generation-based method in Run 2.

In the retrieval-based methods, we propose two methods with different information used for retrieval. Although the retrieval-based methods have been shown to be able to select the responses accurately to some degree in [3] and [8], the setting in STC-2 task is somewhat different from the previous studies, because the Yahoo! news article on which the dialogue is based is given in addition to the utterance text and the response text. Therefore, in the proposed methods, a comment text with high similarity with the given utterance text is searched from Yahoo! News[2] comments data, and the reply text to the comment text is returned as the response to the input. Okapi BM25[10] was used as the similarity measure. The title and the theme of a news article were also used in the similarity search.

Next, we describe the generation-based method. In human conversation, communication is carried out on the premise that each other has common knowledge. This is also true when humans use the dialogue system, and we conduct dialogue on the premise that the system has common sense. Therefore, it is essential that the dialogue system has common sense. However, it is nearly impossible for humans to organize and describe a huge amount of common sense, when the necessary time and cost are considered. In the case of a human being, it is possible to acquire the knowledge through conversation with a person. Hence, it is desirable for dialogue systems to acquire knowledge through conversation as well. This has the advantage that human beings need not explicitly give knowledge to the system.

As a model that can extract knowledge from conversations, the encoder-decoder model has been proposed [16] [18]. It consists of an encoder that encodes the input information into a context vector and a decoder that generates sentences using the context. [18] showed that it is possible to extract knowledge and to conduct conversation by learning pairs of dialogues with the model. For example, [18] reported that when asked who is Skywalker, their conversation model (NCM) responded "*he is a hero.*"

NCM has a problem that it is not possible to respond properly to the input texts that require visual information. For example, [18] reported that when asked how many legs a spider have, NCM responded "*three, i think.*" Further, the image or video may contain more detailed information than texts. Consider, for example, a scene in a news program including a closed caption "*one player won the figure skating*

[1]http://research.nii.ac.jp/ntcir/ntcir-13/index-en.html

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

*match* " and showing an image with the skating player with the gold medal. Here, in the video, more detailed information such as the gold medal that does not exist directly in the text is presented. We thought that if such detailed visual information could be extracted from the image, more specific and useful texts could be generated, including "*gold medals*" which can not be obtained with text alone. In recent years, studies have been reported in which translated sentences are generated by adding image features to the context vector encoded by the encoder-decoder model [2] [5] [9] [12] [17]. These studies showed that visual information works effectively for generating translation.

Meanwhile, visual information is not considered in many text-based dialogue systems, because what is given to the input is only the utterance text. How can the visual information be used without accepting visual information as the input to the dialogue system?

Based on the discussion above, we propose an Associative Conversation Model that associates the input text with the visual information and generates the response using both the text and the associated visual information. In our proposed method, we attempted to generate response texts using visual information without inputting images.

# 2. RETRIEVAL-BASED METHOD

## 2.1 Overview of Architecture

Run 1 and Run 3 are retrieval-based methods. Figure 1 shows the overview of the methods. Queries generated from the utterance text are used for document retrieval to obtain candidate responses, followed by selecting the final response. Apache Solr[3] was used to implement the system. The title in Yahoo! Topics and the article theme were used as well as the comment text, in the similarity search. The difference between Run 1 and Run 3 is queries for similarity search. All themes were used in Run 3, whereas they were selectively used in Run 1 (see Sec 2.2).

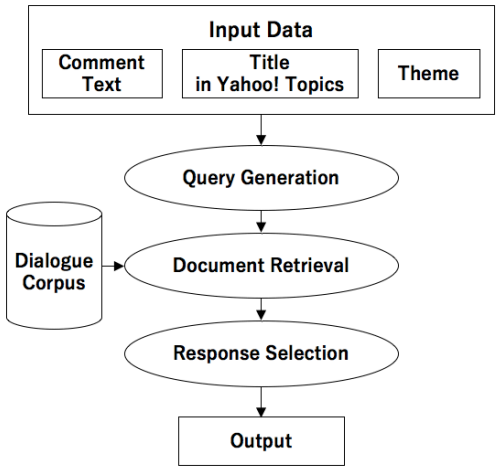


**Figure 1: Overview of processing in Run 1 and Run 3**

[3]http://lucene.apache.org/solr/

## 2.2 Query Generation

First, a query used for similarity search is generated from the given input data. Then, the comment text and the title in Yahoo! topics in the input data are morphologically analyzed and filtered by using the standard analyzers provided in Solr. Kuromoji was used for morphological analysis. As a result of preliminary investigation, we observed that the article theme in the input data often contained words which were not directly related to the current topic in the dialogue. For example, in the case of comments on weather, place names that just happened to appear in the news articles are often included in the article theme. Also, in case of comments on sports, various names such as those of related players are included. Place names and person names important in dialogue are often already included in the comment texts and titles in Yahoo! Topics. Therefore, the place names and person names were removed from the words in the article theme as the search query.

## 2.3 Document Retrieval

Similarity search is conducted for only comment texts in the Yahoo! news comments data, which is composed of pairs of a comment text and the corresponding reply text. The final output is the reply text corresponding to the obtained comment text.

Okapi BM25 was used for weighting words when performing similarity search. In Apache Solr, BM 25 shown in equations 1 to 3 is implemented.

$$score(t,f) = idf(t) \times \frac{(k_1+1) \times tf(t,f)}{k_1 \times (termLenPenalty(f)) + tf(t,f)} \times Boost \tag{1}$$

$$termLenPenalty(f) = 1.0 - b + b \times \frac{fieldLength(f)}{avgfieldLength} \tag{2}$$

$$idf(t) = log\left(\frac{N}{df(t)}\right) \tag{3}$$

Here, $t$ represents a word, and $f$ represents a field (a comment text in the Yahoo! news comments data). Parameters $k_1$ and $b$ of BM25 were set to $k_1 = 1.2$, and $b = 0.75$, respectively, which are default values. $df(t)$ is the number of comment texts including $t$. For *Boost* parameter, the optimum value obtained in the preliminary experiment was set according to the type of words in the search query, which are either comment text, title in Yahoo! topics, or theme. Table 1 shows the optimum values used in the experiment.

**Table 1: Boost value for each query**

| Query | *Boost* |
|---|---|
| Comment text | 1.0 |
| Title in Yahoo! Topics | 9.1 |
| Theme | 3.5 |

# 3. GENERATION-BASED METHOD

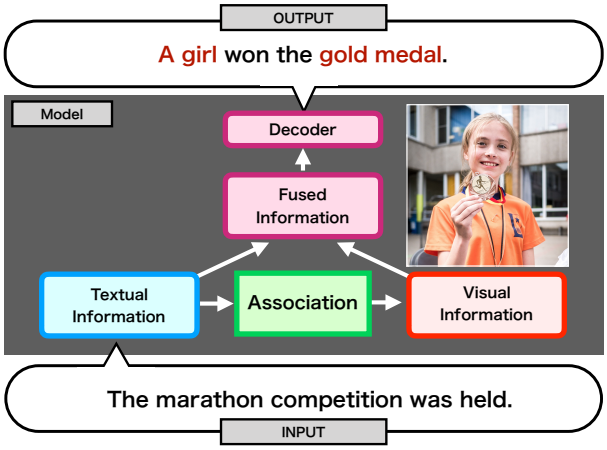


**Figure 2: Generating a response by visual association. The textual information is used to estimate the corresponding visual information, and a response text is generated using the vector obtained by fusing the textual and visual information.**

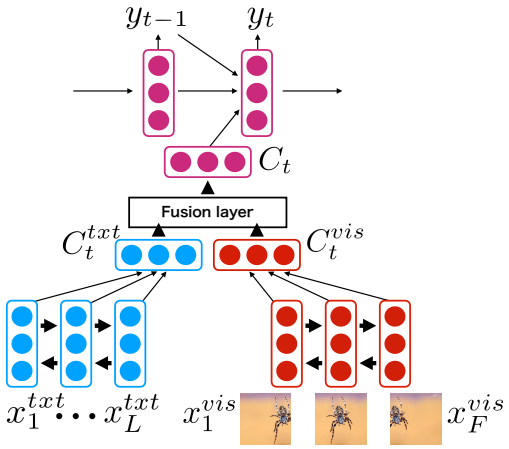


**Figure 3: Step 1: A model that performs prior learning for extracting context vectors.**

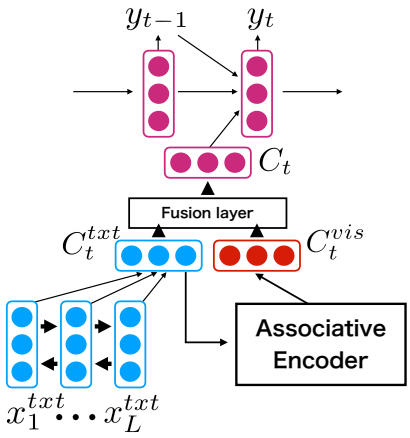


**Figure 4: Step 3: Associative Conversation Model.**

## 3.1 Overview of Architecture

Figure 2 shows the overview of our generation-based model. The objective of this model is to enable for the decoder to generate a more appropriate response text which cannot be obtained only by textual information, by adaptively referring to the context vector based on visual information. However, in this task, only the textual description is given as input. Therefore, we adopted the following approach so that visual information can be utilized from the textual information.

- First, visual association is performed from the input text, and the visual information corresponding to the text is generated.
  (For example, figure 2 shows that the input text "*The marathon competition was held*" is used to generate the visual information corresponding to a scene where a girl runner in marathon competition won the gold medal.)
- Next, by fusing the textual information and the associated visual information is obtained the information reflecting either or both as necessary.
  (Figure 2 shows that the fused information is generated from the textual information "*marathon competition*" and the visual information "*gold medal*", "*girl*".)
- Finally, a response text is generated based on the fused information.
  (Figure 2 shows that the fused information was used to decode the final response "*A girl won the gold medal*".)

Our approach is simple. When generating sentences from the input texts and videos, an encoder for text and another for video can be used to acquire context vectors for text and for video, respectively. In our task, however, the videos cannot be directly obtained from the input because only the textual information is given as input. Therefore, our idea is to replace the encoder for video with a mechanism for generating the visual context vectors from texts. The configuration of this associative conversation model is simple, and is composed of the encoder for text and the decoder equipped with attention mechanism, as well as the visual association encoder from texts. Associative encoder is an RNN which generates visual context vectors from textual context vectors (see Sec 3.2.2). We used LSTMs for the textual encoder, the decoder, and the associative encoder, respectively [6]. The input to the associative conversation model is textual descriptions only, and the output is a response text to the input. This model generates the response text by inputting to the decoder the fused context vector obtained by fusing the textual context vector and the visual context vector generated from the input text. The associative conversation model is able to learn the pairs of dialogue texts by end-to-end, like a normal encoder-decoder model. Actually, however, two prior learnings are performed before training the associative conversation model. One is the training of multimodal encoder-decoder model which generates a response text from the video and text. The other is the training of the associative encoder which predicts the visual context vector from the textual context vector obtained by the multimodal encoder-decoder model.

## 3.2 Learning Method

Learning consists of the following three steps. **In order to learn the Associative Conversation Model in the final step 3, it is necessary to learn two models in step 1 and step 2, in advance.**

**Step 1: Extraction of context vectors between textual and visual information**

Figure 3 shows a network model used in step 1. End-to-end learning is performed on the model that inputs the utterance text $X_{txt}$ and the video $X_{vis}$ corresponding to the utterance and outputs the response sentence $Y$. In this learning, the following four components are trained simultaneously.

- Textual encoder inputting $X_{txt}$ and outputting $C_{txt}$ which is the textual context vector
- Visual encoder inputting $X_{vis}$ and outputting $C_{vis}$ which is the visual context vector
- Fusion layer inputting $C_{txt}$ and $C_{vis}$ and outputting the fused context vector $C$
- Decoder with attention mechanism, inputting $C$ and outputting the response text $Y$

The trained model is used to extract the correspondence between the textual and visual information (see Fig. 3).

**Step 2: Learning for visual association**

In this step, the associative encoder is trained, which inputs the textual context vector $C_{txt}$ extracted in step 1 and outputs the visual context vector $C_{vis}$ corresponding to the input utterance text.

**Step 3: Generation of response text via association**

In step 3, learning is performed in the network where the visual encoder in step 1 is replaced with the associative encoder trained in step 2. This model is a generation-based model which inputs $X_{txt}$, and outputs a response text $Y$, by using the fused context vector $C$ obtained from $C_{txt}$ and $C_{vis}$ in the fusion layer. That is, the structure of the Associative Conversation Model is the same as the network of step 1, except that it uses an associative encoder instead of the visual encoder. In the fusion layer, the decoder with attention is trained again after the weights learned in step 1 are initialized. In textual and video encoders, the weights trained in step 1 are left unchanged, and are not updated. It should be noted that what is trained in this model are only the decoder, the attention, and the fusion layer.

### 3.2.1 *Step 1: Extraction of Context Vectors between Textual and Visual Information*

The training data used in this step are the utterance text, the video corresponding to the utterance text, and the response text for the utterance. We used videos rather than images because the texts often include the expression of actions. We created the dataset based on TV programs as they contain the utterance texts, their corresponding videos, and their response texts. First, the closed caption texts were extracted from TV news programs as the utterance texts $X_{txt}$. Also, the scenes corresponding to the temporal intervals where the utterances occurred were extracted as the sequence of images $X_{vis}$ from the TV news programs. For more information on the dataset, see Sec 5.1. Here, the utterance $X_{txt}$ is represented by the sequence of the corresponding word $x^{txt}$. Similarly, the video $X_{vis}$ corresponding to the utterance text is represented by the sequence of image features $x^{vis}$. In this work, the image features acquired by the already learned CNN were used instead of learning the image features directly from the input video. Therefore, the input to the prior learning models are the sequence of word vectors $X_{txt} = (x_1^{txt}, ....., x_L^{txt})$ and the sequence of image feature vectors $X_{vis} = (x_1^{vis}, ....., x_F^{vis})$. Also, the output is the sequence of words vectors $Y = (y_1, ....., y_T)$. Here, $L$, $F$, and $T$ represent the length of each input text, the number of the input images, and the length of the output text, respectively. The model used in step 1 is the multimodal encoder-decoder model consisting of a textual encoder, a visual encoder, and a decoder with attention [1] . In step 1, the textual and visual encoders encode the input text and video, respectively, to obtain the context vectors $C_t^{txt}$ and $C_t^{vis}$. Then, $C_t$ which is the fused context vector is input to the decoder to generate a response text. For extracting context vector $C_t$, the attention mechanism was used.

$$s_t = LSTM(C_t, s_{t-1}, y_{t-1}) \tag{4}$$

$$P(y|s_{t-1}, C_t) = softmax(W_s s_{t-1} + W_c C_t + b_s) \tag{5}$$

Attention mechanism can extract a context vector that strongly reflects some parts of the sequence corresponding to the words of the response text. For example, when the word "*spider*" of the text and the image of the spider were paid attention to, the textual and visual context vectors of the spider are generated, respectively. In this case, there is a correspondence relations between the textual and visual context vectors. So, in step 2, the associative encoder which predicts the visual context vectors from the textual context vectors is trained. The purpose of step 1 is to extract the textual and visual context vectors having correspondence relations to be used for learning in step 2. The usual attention-based encoder-decoder model calculates the context vectors by Eq. (6), (7), and (8). $h_i$ is the intermediate vector of the bidirectional LSTM computed from the input sequence $h_i = [\overrightarrow{h_i}; \overleftarrow{h_i}]$ [13].

$$C_t^{att} = \sum_{i=0}^{T} \alpha_{t,i} h_i \tag{6}$$

$$\alpha_{t,i} = softmax(e_{t,i}) \tag{7}$$

$$e_{t,i} = w_e^{\mathrm{T}} tanh(W_e s_{t-1} + V_e h_i + b_e) \tag{8}$$

Meanwhile, the multimodal encoder-decoder model acquires two context vectors $C_t^{txt}$ and $C_t^{vis}$. Thus, the multimodal encoder-decoder model uses the following equations instead of Eq. (6) (7) and (8).

$$C_t^{txt} = \sum_{i=0}^{L} \alpha_{t,i}^{txt} h_i^{txt} \tag{9}$$

$$C_t^{vis} = \sum_{j=0}^{F} \alpha_{t,j}^{vis} h_j^{vis} \tag{10}$$

$$\alpha_{t,i}^{txt} = softmax(e_{t,i}^{txt}) \quad (11)$$

$$\alpha_{t,j}^{vis} = softmax(e_{t,j}^{vis}) \quad (12)$$

$$e_{t,i}^{txt} = w_{e_{txt}}^{\mathrm{T}} tanh(W_{e_{txt}} s_{t-1} + V_{e_{txt}} h_i^{txt} + b_{e_{txt}}) \quad (13)$$

$$e_{t,j}^{vis} = w_{e_{vis}}^{\mathrm{T}} tanh(W_{e_{vis}} s_{t-1} + V_{e_{vis}} h_j^{vis} + b_{e_{vis}}) \quad (14)$$

Also, there are some responses that do not always require visual information in the dialogue. Therefore, the decoder needs to be able to measure which of the textual and visual information should be paid attention to and how strongly it should be focused. The degree to which the textual and visual context vectors $C_t^{txt}$ and $C_t^{vis}$ should be referred to can be learnt by introducing the weight matrices and by multiplying them to $C_t^{txt}$ and $C_t^{vis}$, respectively. Therefore, the context vector eventually passed to the decoder is the fused context vector having how strongly the textual and visual information should be referred to.

$$C_t = W_{txt} C_t^{txt} + W_{vis} C_t^{vis} + b_c \quad (15)$$

This fused context vector $C_t$ is used to predict the next word in the decoder. We call this layer the fusion layer. We learn the parameters $W_{txt}$, $W_{vis}$, and the biases $b_c$. After training the multi-modal encoder-decoder model, the training data is again input to the model to save the textual and visual context vectors.

#### 3.2.2 Step 2: Learning for Visual Association

In step 2, the visual associative encoder is trained. The associative encoder is trained so that it can predict the visual context vector from the textual context vector. In addition, the associative encoder decides the next visual context vector by referring to the past textual context vectors. Let us consider the following example. The visual information (visual context vector) associated with the word "*squeeze*" is diverse. For example, the phrases "*squeeze a lemon*", "*squeeze a person's hand*", and "*squeeze toothpaste out*" would make us imagine the different visual situations representing "*squeeze* ". In this example, the phrases "*lemon*", "*person's hand*", and "*toothpaste*" make the situation of "*squeeze*" more concrete. In this work, the association is to generate the corresponding visual information from the textual information. Namely, there is a demand that the model can imagine what to "*squeeze* ". Here, suppose that a context vector "*squeeze*" is obtained by the attention mechanism. If the context vector "*lemon*" was obtained before this point, it will be possible to generate the visual information of "*squeeze a lemon*" from the context vectors of "*lemon*" and "*squeeze*". This indicates the utilization of temporal features. Therefore, RNN is used for the associative encoder. Accordingly, the associative encoder can be seen as a regression model with the input sequence of context vectors of words $C^{txt} = (C_1^{txt}, ..., C_t^{txt}, ..., C_T^{txt})$ and the output sequence of context vectors of images $C^{vis} = (C_1^{vis}, ....., C_t^{vis}, ....., C_T^{vis})$ (Eq. (16)). Here, $T$ is the length of the output text. We used LSTM which is capable of learning long-term dependence.

$$C_t^{vis} = AssociativeEncoder(C_t^{txt}) \quad (16)$$

#### 3.2.3 Step 3: Generation of Response Text via Association

In step 3, the associative conversation model learns response texts using the associative visual information instead of visual information directly obtained from video (see Fig. 4). The input to the associative conversation model is the sequence of words $X_{txt} = (x_1^{txt}, ..., x_L^{txt})$. The output is also the sequence of words $Y = (y_1, ....., y_T)$.

In the associative conversation model, the visual encoder obtained in the prior learning of step 1 is replaced with the associative encoder. Therefore, the architecture of the associative conversation model can be obtained by replacing Eq. (10) in step 1 with Eq. (16). In other words, the mechanism of blending the textual and visual information trained in step 1 is left unchanged, and the visual context vector is predicted from the textual context vector by the associative encoder. The visual context vectors as well as the textual context vectors are given to the decoder, and the decoder, the attention, and the fusion layer (Eq. (15)) are trained. The weights of the decoder, the attention and the fusion layer are initialized, and the weights obtained by prior learning of step 1 is not used. The weights in the textual and associative encoders are not updated, and the same parameters used in the prior learning of step 1 are given. The reason for re-training the decoder, the attention and the fusion layer is because the associative visual context has different property from the visual context vector generated in step 1. The associative visual context is different from the visual context vector obtained in step 1. We thought that re-training parts other than encoding of context vector enables to generate response texts using the associative visual elements in common knowledge.

## 4. EXPERIMENTS

In NTCIR-13 STC-2 Japanese Subtask, a participant needs to implement a dialogue system that returns top ten responses for each of the given 100 utterances. These responses are evaluated manually to qualify the accuracy of the dialogue system[11].

### 4.1 Dataset

The test data of STC-2 are 100 comments randomly sampled from Yahoo! news articles from December 7, 2016 to February 7, 2017. In the retrieval-based methods, the following information was used: comment texts, titles in Yahoo! topics, and themes. The repository used for search is composed of pairs of comment texts and their corresponding reply texts from Yahoo! News comments data.

In the generation-based method, our model was learned using the following data collected independently: subtitles in TV drama and the corresponding video where the subtitles were displayed. The visual encoder of the model of step 1 encodes a sequence of images (video).

Note that the subtitles were delimited by "!" , "?", or Japanese period to acquire sentences, and a pair of one sentence and the following sentence was regarded as one dialogue. Therefore, the response text of a certain utterance becomes the next input utterance. Also, the video was cut out as a frame sequence with a frame rate of 5 fps. Each frame was input to the prior learned convolutional neural network, and the output of the last pooling layer was used as the image features. VGG16 was used for the convolutional neural network [15]. The recorded programs were 150 Japanese TV dramas broadcast in May 2017, having 98k dialogues and 23k vocabulary words.

### 4.2 Model Configuration

**Table 2: Result of formal run**

| | Rule-1 | | | | | |
|---|---|---|---|---|---|---|
| Run | Mean $nG$@1 | Mean $nEER$@2 | Mean $Acc_{L2}$@1 | Mean $Acc_{L2}$@2 | Mean $Acc_{L1L2}$@1 | Mean $Acc_{L1L2}$@2 |
| Run1 | 0.4150 | 0.4538 | 0.1560 | 0.1350 | 0.6680 | 0.6320 |
| Run2 | 0.2541 | 0.2858 | 0.0020 | 0.0030 | 0.5840 | 0.5290 |
| Run3 | 0.3303 | 0.3737 | 0.1220 | 0.1130 | 0.5300 | 0.5210 |
| | | | | | | |
| | Rule-2 | | | | | |
| Run | Mean $nG$@1 | Mean $nEER$@2 | Mean $Acc_{L2}$@1 | Mean $Acc_{L2}$@2 | Mean $Acc_{L1L2}$@1 | Mean $Acc_{L1L2}$@2 |
| Run1 | 0.3762 | 0.4101 | 0.1560 | 0.1350 | 0.5760 | 0.5360 |
| Run2 | 0.0177 | 0.0230 | 0.0020 | 0.0030 | 0.0360 | 0.0370 |
| Run3 | 0.2912 | 0.3317 | 0.1220 | 0.1130 | 0.4420 | 0.4330 |

**Table 3: Example of output from Run1 and Run3**

| | |
|---|---|
| Input | <Comment text>I stopped going out by bicycle because the forecast said it would rain or snow at 12 o'clock, but there is nothing falling yet.<br><Title in Yahoo! Topics>Snow falls in downtown Tokyo. Snow likely to pile up in the Kanto flat land<br><Theme>Tochigi Prefecture, Weather forecast, Gunma prefecture, Ibaraki prefecture, Snow damage and its measures |

| | **Run1** | **Run3** |
|---|---|---|
| Rank1 | It would be fun if the snow occasionally piled up in downtown area. Because even though it is the Kanto local news, the news program of the key station makes a fuss like a nationwide incident. | It's amazing that as many as two schools will participate from Kochi prefecture, including a "21st-century selection" slot. |
| Rank2 | Unusually, there was no snow last month, wasn't it? | It's also heavy snow in the southern part of Kanagawa prefecture. |
| Rank3 | It'd be easier to objectively understand the situation with the expressions like possibility or probability. Ridiculous to describe it with the word "worry" ... It's quite an emotional expression. | In the electricity industry, Niigata prefecture is within the jurisdiction of Tohoku Electric Power Co. |

In Run 2, the hidden layer dimension of the text encoder was set to 512, the hidden layer dimension of the visual encoder to 512, the dimension of the fused context vector to 512, and the hidden layer dimension of the decoder to 512. Adadelta was used for optimization. The associative encoder is composed of two-layer LSTMs with the hidden layer dimension being 512.

## 4.3 Evaluation

Table 2 summarizes the evaluation results for each run submitted to the formal run.

For Run 1 and Run 3, which are retrieval-based methods, Run 1 gained better evaluation than Run 3 in any evaluation measure. From this result, it can be said that unnecessary words were successfully removed from the article themes. As can be seen from table 3, Run 1 obtained more relevant dialogues from the repository, by excluding words that are not very important to determine responses, such as "Tochigi prefecture" and "Gunma prefecture".

For Run2, which is the generation-based method, the results are summarized as follows;

- In Rule 1, Mean $Acc_{L2}$@2 and Mean $Acc_{L2}$@1, which are the accuracy when only L2 is correct, were more than 0.5, indicating that Run2 had many L1 and few L2.
- In Rule 2, Mean $Acc_{L2}$@2, and Mean $Acc_{L2}$@1 got worse than those in Rule 1.

From the above results, it is expected that the proposed method does not contain sufficiently useful information in the response text, and/or it is not context dependent on the input utterance. The following factors can be considered as their causes;

- insufficient amount of learning data,
- lack of tuning,
- many unknown words (proper nouns) in the test data.

As a result of checking the generated responses, we could not confirm the case where the visual information provided good results for generating the response texts this time. The most probable cause is that there were few comment texts that require visual information in the test data. Although there were a few comments that required visual information, there is a possibility that the keyword (e.g., Osprey) used for the visual information happened to be an unknown word and that the proposed method did not work well. Another possible reason why the system could not answer a useful question requiring visual information is the possibility that the number of correspondence between subtitles and videos was not sufficient in the learning data.

# 5. ADDITIONAL EXPERIMENTS

In this section, the additional experiments are described, where our generation-based model is trained with TV news data instead of TV drama data. Then, the result of comparison is shown, between the Associative Conversation Model

and a model without association. An encoder-decoder model with attention mechanism for generating dialogue responses were used as the baseline model. The baseline model generates a sentence $Y$ from the input utterance text $X_{txt}$. Both the proposed model and the baseline model were trained with the same dialogue sentences, and we investigated the effect of association. We also analyzed whether the associative conversation model acquired effective visual information for response generation by visualizing the associated objects. As a preliminary result, it was confirmed that visual information seemed to work effectively in several examples. We also found that our proposed model associates visual information related to input texts.

## 5.1 Dataset

The model was learned using the following data collected independently: subtitles in TV news and the corresponding video where the subtitles were displayed. The recorded programs were 163 Japanese TV news broadcast from December 2016 to March 2017, having 38K dialogues and 19K vocabulary words. Data was divided into 34K dialogue pairs for training data and 4K for test data.

## 5.2 Model Configuration

In step 1, we used single layer LSTMs as both the encoders and the decoder. The hidden layer dimension of the textual encoder was set to 512, the hidden layer dimension of the visual encoder to 512, the dimension of the fused context vector to 1024, and the hidden layer dimension of the decoder to 1024, the dimensionality of word embedding to 512, the dimensionality of image feature to 512, and the batch size to 64. Adagrad was used for optimization [4].

In step 2, the associative encoder is composed of 4-layer LSTM with the hidden layers dimension being 1024, and the batch size was set to 64. Adam was used for optimization [7].

In step 3, the model is composed of the textual encoder, the decoder, and the associative encoder instead of the visual encoder. The hidden layer dimension of the textual encoder was set to 512, the dimension of the fused context vector to 1024, the hidden layer dimension of the decoder to 1024, the dimensionality of word embedding to 512, and the batch size to 64. Adagrad was used for optimization.

Also, in step 3, the same weights and biases of the decoder and attention as in step 1 were used and not updated. The weights of associative encoder were not updated either. The weights of the decoder were initialized in step 3 (See Eq. (4), (5), (13), (14), and (15)) .

The baseline model is composed of single layer LSTMs as the encoder and the decoder. The hidden layer dimension of the textual encoder was set to 512, the dimension of the fused context vector to 1024, the hidden layer dimension of the decoder to 1024, the dimensionality of word embedding to 512, and the batch size to 64. Adagrad was used for optimization.

## 5.3 Results

Figure 5 shows the texts generated from the test data by our model or by the baseline, and the images that are similar to the associated visual information. The Japanese sentences generated by the models were translated into English. There results show that our model generated texts with more useful information than the baseline. For example, in the example on the left of the figure 5, the proposed model generated a specific weather forecast with the word "*snowy*" for the input sentence "*The University Entrance exam will be held on 14th and 15th.*" Note that it is a fact that the snow actually fell on the day of the exam, and that the images showing that it was snowing at the venue of the exam were included in the training data. The important point here is that the word "*snowy*" cannot be easily generated from the input sentences alone, but is a word that can be generated for the first time in association with the image of snow.

On the other hand, the result generated by the baseline is "*There will be a large-scale fire that is also in western Japan and eastern Japan* " and contains erroneous information such as "*fire*". In the example on the right of figure 5,


**Figure 5: Example of comparison results on validity of sentence generation by visual association[3]. It indicates that the visual information associated from the input text by the proposed model exhibits some correspondence with the input text. In addition, it shows that the proposed model can generate response texts including useful information compared with the model without association. For example, the result on the left shows that our model associates the snow scene with the utterance text and generated the word "*snowy*". Note that it is a fact that it actually snowed on the day of the exam. The important point here is that the word "*snowy*" cannot be easily generated from the input sentences alone, but is a word that can be generated for the first time in association with the image of snow.**

[3]Image on the left: "NHK ニュース 7", NHK, 11 January 2017, Image on the right: "NHK ニュース 7", NHK, 23 February 2017

the proposed model generated the text including the word "*gold medal*" for the input sentence "*Well, today is All Japan Figure Skating Championships.*" In addition to this example, it was confirmed that the proposed model with the associative function generated useful sentences with more specific information than the baseline without it. These results suggest that association works effectively to generate sentences with more specific information. In order to verify this, we analyzed what kind of visual information was generated by the associative encoder.

## 5.4 Analysis of Visual Association

Figure 5 shows the analysis result of the association. These results show that our model successfully associated visual information related to the input sentence. The upper image in figure 5 is the image with the highest similarity to the associative visual context generated by association from the input sentence by the proposed model. We calculated the cosine similarity between the visual context vector $C_{vis}$ obtained in step 1 and the associative visual context vector $C'_{vis}$ generated by the associative encoder. Also, we assumed that the images that were paid attention to at the time indicated by the value of $\alpha^{vis}$ which is derived when the most similar context vector $C_{vis}$ was generated, are associative images. In other words, the image in figure 5 was obtained by visualizing the associative visual information from texts using images in the training data. From the result of figure 5, it is confirmed that the visual information matching the contents of the sentence can be associated. For example, the example to the right of figure 5 shows that a scene where a skating player acquired a gold medal was obtained by visual association on the topic of "*Skating Championships*", although it misunderstood figure skating as speed skating. Also, the word "*gold medal*" was generated from its association result. The associated image includes a player holding the gold medal and other players skating behind her. Multiple examples were found, in which the images matching the input sentence were successfully associated like this. Some of these examples are found in the appendix. Also, these examples show that useful words were generated which are difficult to generate easily without any association. For example, it is difficult to generate the word "*snowy*" from the sentence "*the University Entrance exam will be held on 14th and 15th*", without the visual association that it actually snowed around the venue at that time.

However, there remain several problems. In some examples, it was confirmed that the visual information was associated with a topic different from the input sentence. For example, in the example above, a scene where a speed skating player won the gold medal was associated with the topic about figure skating. It is expected that this can be improved by increasing the amount of the training data. Also, the proposed model could not generate a good reply to the utterances in a colloquial tone, because the learning was performed using the training data composed of sentences of news programs. A solution to this is to learn by training data composed of general dialogue sentences instead of news sentences when re-learning the model using the fused context vectors in step 3. That is, using the knowledge extracted by the encoders in steps 1 and 2, relearning is performed in step 3 using general dialogue sentences. Once the knowledge can be extracted, the model can be trained more efficiently by performing only step 3 according to the task (e.g., general conversation).

# 6. CONCLUSIONS

This paper summarized the three dialogue systems proposed by team KSU for STC-2 task at NTCIR-13.

With the retrieval-based methods, we were able to select response texts with some degree of accuracy. It was also found that the accuracy is improved by excluding place names and person names from article themes. In the future, instead of simply excluding the place name and the person's name from themes, we will consider a method to judge their necessity as dialogue information and use only more appropriate themes as search queries. Furthermore, although similarity search was performed only for comment texts in the repository this time, we plan to consider methods to search for other elements as well to acquire more adequate responses.

In the generation-based method, we could not find enough evidence from the results using the evaluation data of STC to show that the associated visual information would accelerate to generate more appropriate responses. For this reason, we learned our model using TV news data instead of TV drama data. As a preliminary result, it was confirmed that visual information seemed to work effectively in several examples.

# 7. ACKNOWLEDGEMENTS

The authors would like to express our appreciation to Tasuku Kimura and Ryo Tagami for their valuable comments and suggestions throughout the work.

# 8. REFERENCES

[1] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014.

[2] I. Calixto, Q. Liu, and N. Campbell. Doubly-attentive decoder for multi-modal neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1913–1924, 2017.

[3] K. Chikai and Y. Arase. Analysis of similarity measures between short text for the ntcir-12 short text conversation task. In *NTCIR*, 2016.

[4] J. C. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.

[5] D. Elliott and Á. Kádár. Imagination improves multimodal translation. *CoRR*, abs/1705.04350, 2017.

[6] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[7] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[8] S. Matsumoto and M. Araki. Scoring of response based on suitability of dialogue-act and content similarity.

[9] H. Nakayama and N. Nishida. Zero-resource machine translation by multimodal encoder-decoder network with multimedia pivot. *Machine Translation*, 31(1-2):49–64, 2017.

[10] S. Robertson, H. Zaragoza, et al. The probabilistic relevance framework: Bm25 and beyond. *Foundations*

*and Trends® in Information Retrieval*, 3(4):333–389, 2009.

[11] H. Ryuichiro. Overview of the ntcir-13 short text conversation task 2 (japanese subtask). In *NTCIR*, 2017.

[12] A. Saha, M. M. Khapra, S. Chandar, J. Rajendran, and K. Cho. A correlational encoder decoder architecture for pivot based sequence generation. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 109–118, 2016.

[13] M. Schuster and K. K. Paliwal. Bidirectional recurrent neural networks. *IEEE Trans. Signal Processing*, 45(11):2673–2681, 1997.

[14] L. Shang, T. Sakai, H. Li, R. Higashinaka, Y. Miyao, Y. Arase, and M. Nomoto. Overview of the NTCIR-13 short text conversation task. In *Proceedings of NTCIR-13*, 2017.

[15] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[16] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112, 2014.

[17] J. Toyama, M. Misono, M. Suzuki, K. Nakayama, and Y. Matsuo. Neural machine translation with latent semantic of image and text. *CoRR*, abs/1611.08459, 2016.

[18] O. Vinyals and Q. V. Le. A neural conversational model. *CoRR*, abs/1506.05869, 2015.

## 9. APPENDIX

Additional examples of comparison results on validity of sentence generation by visual association are shown in figure 6 and 7.

In the example on the left of figure 6, "*Yokozuna*" which represents the highest rank in sumo, was visually associated from the input, but the name "*Hakuho*" who is also Yokozuna and is different from the person in the associated image, was output. In the example on the right of figure 6, a pitcher was visually associated, but he is a different person from "*Otani*" who is another pitcher described in the input.

In the example on the left of figure 7, the proposed model associated an image showing the placement of high pressures from the input phrase "*high pressure*", and the word "*sunny*" was generated from the associated image. In the example on the right of figure 7, the proposed model associated an image of a person riding a bicycle likely to fall over due to a strong wind, from the input phrase of low pressure. The words "*traffic*" and "*windstorm*" were generated from the associated image

[4]Image on the left: "NHK ニュース 7", NHK, 13 January 2017, Image on the right: "ニュースウォッチ 9", NHK, 17 February 2017

[5]Image on the left: "ニュースウォッチ 9", NHK, 27 February 2017, Image on the right: "ニュースチェック 11", NHK, 20 February 2017

| | | |
|---|---|---|
| **Input** | The Grand Sumo Tournament is in the second day. | As for pitchers, three players including Otani have been selected from Nippon Ham. |
| | 大相撲初場所は2日目です。 | 投手では、<br>日本ハムから大谷投手を含む3人が<br>選ばれています。 |
| **Output by Baseline** | No. 1 is No. 1 in 1 meter. | Baseball is<br>out in the professional this season. |
| | 1位は1メートルでの<br>1位の1位です。 | 今シーズンはプロに野球<br>が出ています。 |
| **Output by ACM** | Today, **Yokozuna** Hakuho will aim for the first victory. | This is an **athlete**. |
| | きょうは初優勝を目指す<br>**横綱**・白鵬です。 | こちら の スポーツ **選手** 。 |
| **Image associated from the input** |  |  |
| **Words generated mainly from the associated image** | **Yokozuna**<br>(The highest rank in sumo) | **Athlete** |
| | **横綱** | **選手** |

**Figure 6: Sports**[4]

| | | |
|---|---|---|
| **Input** | The vicinity of Honshu is expected to be covered widely by mobile high pressure. | In tomorrow morning, it will be isolated snowstorms around the Japan Sea side of western Japan and eastern Japan, due to the developing low pressure. |
| | 本州付近は、移動性高気圧に広く<br>覆われる見込みです。 | 発達中の低気圧の影響で、<br>あすの朝にかけては、西日本や東日本の<br>日本海側を中心に、<br>所によって吹雪となる見込みです。 |
| **Output by Baseline** | Let's move on to the weather around the country on tomorrow. | Vigilance is necessary for windstorms and high waves. |
| | では、あすの各地の天気です。 | 気象庁は、暴風や高波に警戒<br>が必要です。 |
| **Output by ACM** | Tomorrow morning, it will be **sunny** in many places from western Japan to eastern Japan, and the side on the Japan Sea of western Japan. | Please also be aware of the influence on **traffic** caused by **windstorms**, heavy blizzards, and snowdrift. |
| | あすの朝には、西日本、東日本から<br>西日本の日本海側では、<br>**晴れ**ている所が多くなりそう | 気象庁は、**暴風**や猛吹雪、<br>吹きだまりによる**交通**への<br>影響も注意してください。 |
| **Image associated from the input** |  |  |
| **Words generated mainly from the associated image** | **Sunny** | **Traffic, Windstorms** |
| | **晴れ** | **交通, 暴風** |

**Figure 7: Weather**[5]