タイトル	Extending a Japanese Speech−to−Gesture Dataset Towards Building a Pedagogical Agent for Second Language Learning
著者	HASEGAWA, Dai; ECHIZENYA, Hiroshi
引用	北海学園大学工学部研究報告(48): 53-64
発行日	2021-01-15

Extending a Japanese Speech−to−Gesture Dataset Towards Building a Pedagogical Agent for Second Language Learning

Dai HASEGAWA* and Hiroshi ECHIZENYA*

Abstract

We created a Japanese speech−to−gesture dataset where we recorded 298 mins (trimmed into 210 mins) of speech audio data and the motion capture data of the accompanying gestures. Our aim was to tackle speech−to−gesture generation by using a data−driven approach on the dataset. Our first attempt of the speech−to−gesture generation partially succeeded. However, to improve data−driven gesture generation, we will need more informative the dataset. In this paper, to enrich the dataset, we annotated seven gesture phases (rest, preparation, pre−stroke hold, stroke, post−stroke hold, retraction) to 240 sentences out of 1047 sentences of our dataset. However, the annotation process needs efforts. Thus, to extend the annotations to all over the dataset, we tested a gesture phase estimation by using Bi−directional Long−Short Term Memory (Bi−directional LSTM). The results show that overall accuracy of seven gesture phase estimation was 0.61 in precision, 0.61 in recall, and 0.61 in f−value. The most successful phase in the estimation was rest phase scored 0.91 in precision, 0.94 in recall and 0.92 in f−value.

1 Introduction

New education paradigm, as typified by the flipped classroom model, has changed the focus of education from offering students knowledge to developing students experience. In such education process, students often required to learn basic knowledge and skills by themselves outsides of classroom activity. Therefore, the development of effective learning support tools for self−learning becomes an important issue in modern education.

Among the various learning fields, the second language learning is one of the most challenging subjects for developing self−learning support system. It requires listening and speaking practice, but it has been performed in face−to−face communication. In recent years, virtual characters with a similar body structure with humans, often referred to as virtual humans, have gained much interest by educa-

* Department of Life Science and technology, Faculty of Engineering, Hokkai−Gakuen University

tion field. In the filed of education, the virtual humans are also called as pedagogical agents. We believe that, in the second language learning, the use of pedagogical agents is promising as well as many do in other subjects.

Such pedagogical agents have to perform natural human–like non–verbal behaviors. And speech accompanying gestures also play an important role in educational interactions between the pedagogical agents with learners. However, implementing gestures with virtual human considered costly. Thus, many researches concerning the automatic gesture generation have been carried out. Data–driven gesture generation is one of the methods to generate speech accompanying gestures, aiming more simple way to implement gestures.

In human–human communication, gestures (defined as gesticulation[13]) play an important role, such as complementing or emphasizing speech. And researchers have been introducing non–verbal expressions, including gesture, into computer systems with human–like appearances, called Embodied Conversational Agents (ECAs)[3]. It has also repeatedly been verified that gestures performed by ECAs or robots have positive effects in various applications[1].

However, implementing meaningful gestures along with speech into ECAs costs time and effort. To date, several studies have been conducted to automatically generate gestures from speech or text. Early works tackled this problem using a rule–based approach[5, 4]. The rulebased approach has the advantage in that if we can prepare enough knowledge to represent a task and domain, the system will perform well. However, preparation takes a lot of effort. Thus, the domain is highly restricted.

To avoid this difficulty of the rule–based approach, a data–driven approach with machine learning has also been proposed[6, 7]. In the data–driven approach, we do not have to prepare elaborate domain knowledge. Rather, the "knowledge" is automatically acquired in the learning process, and this makes the data–driven approach be applicable to wider domains than the rule–based approach. However, generating gesture motions which are perfectly consistent with speech content still remains as a challenging problem. In [7], the authors solved the gesture generation problem as a classification task, thus the method still requires predefined gesture categories and handmade motion data. On the other hand, the work of [7] succeeded to generate the motions of beat gestures. However, this approach only used pitch and sound pressure of speech as inputs, so it is difficult to distinguish phonemes. Naturally, this leads to difficulty in speech recognition. Since the semantic content of speech is highly correlated to gestures, the difficulty in speech recognition causes a negative influence on the generated gestures.

In this paper, we will briefly introduce our data–driven approach along with our dataset. The very initial attempt of our gesture generation partially succeeded. However, we believe that enriching the

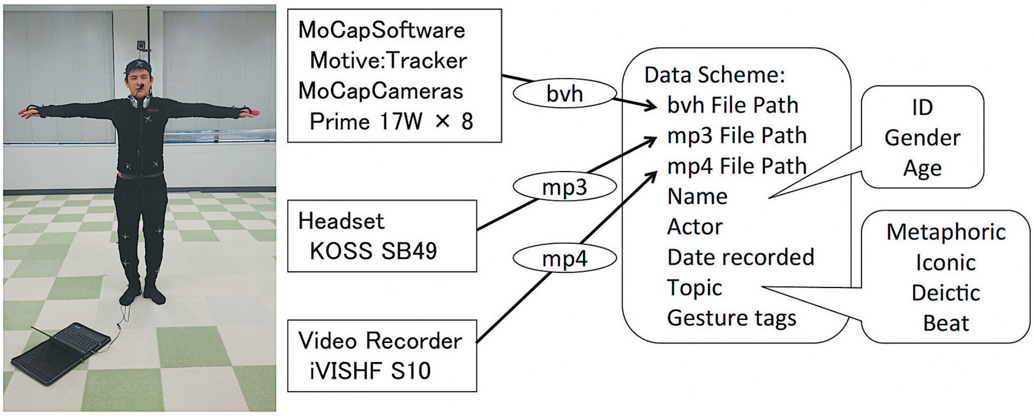


Figure 1: Recording and data format overview

dataset with annotations by hand makes our results better. However, the annotation, naturally, needs experts knowledge, time, and efforts. Thus, in the main part of this paper, we will discuss our automatic dataset annotation method by utilizing a small portion of annotated data.

2 Summary of Dataset

In our recordings, we aimed to mainly acquire metaphoric gestures and iconic gestures, as categorized by McNiel[13]. Metaphoric gestures are gestures in which an abstract meaning is visually expressed as if it had a physical form, such as showing an empty palm as to indicate one is'presenting an idea'. Iconic gestures are used to illustrate physical, concrete items or acts, like expressing how large an object is or rapidly moving one's hand up and down to indicate the action of chopping something. These gestures aid listeners in comprehending the structure and events or objects depicted in the speech, and have many potentials uses in explanation, learning, and teaching. Deictic gestures, used to indicate real/imaginary objects, people, directions, etc. around the speaker, were considered inappropriate for usage in deep learning aiming to learn the association between speech and gesture, heavily depending on the speaker's surrounding environment rather than the actual context of the speech. Also, beat gestures, used for emphasis and expressing the rhythm of conversations, have little relation to the actual context of speech and were not considered to be viable to be used in the learning.

2.1 Recording Devices

Motion data was acquired using the software Motive : Tracker by SPICE Inc., along with a motion capture suit with 49 markers and 8 OptiTrack Prime 17 cameras, placed in an 850 × 850m area (**Fig-**

ure 1). The recorded motion data was exported to bvh format, in which motion data is described as the hierarchy and initial pose of the skeleton and time sequence data of each joint's rotation angle. Speech data was acquired using a headset, as to not hinder the subject's movement. The recorded speech data was stored in mp3 format. Video data was acquired using a stationary video camera and stored in mp4 format.

2.2 Participants and Procedure

The participants were 2 male undergraduate students, both at the age of 25. The data was recorded in form of an interview, where the participant explains a topic prepared and thought about beforehand. Several other methods were attempted, but these methods were considered unsuited for the recording. First, when having the participant read a transcript out loud, valid gestures did not appear. This is thought to be because the speaker has to have a concrete enough image about the context of what they were talking about for gestures to naturally appear during speaking. Second, when having the participant make a presentation using a slide show, deictic gestures appeared with too much frequency, since the speaker tended to point at his presentation slide while explaining. Third, when having the read a transcript of easy context such as fairy tales, and instructing the participant to concentrate on using plausible gestures while speaking, the participant often used gestures too frequently, and gestures that were too exaggerated. Putting too much emphasis on doing gestures led the gesture usage to be unnatural, and having such gestures in the dataset would have a negative effect on the learning. Recording took place in a comfortably large quiet room, where only the subject of the motion/speech data and the person operating the recording devices were allowed to enter so that the recorded speech contains as less static as possible. One participant was to wear the headset and motion capture suit and make sure there are no problems with the positions and number of markers. Then, the participant was to take a T–pose so the recorder can make sure that the motion tracking was calibrated correctly. After checking, the recorder starts recording. Before proceeding to speak, the participant claps his hands once so that portion could be used to sync the speech, motion, and video data.

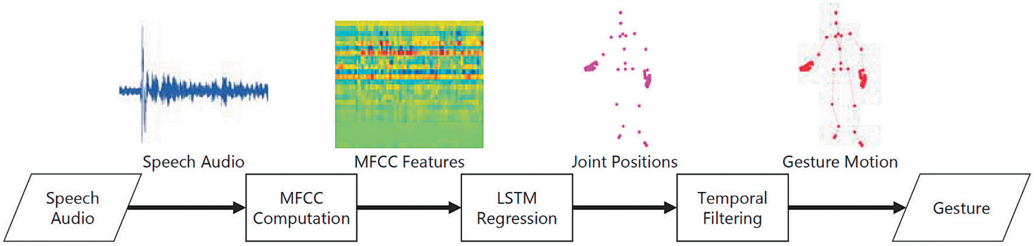


Figure 2: Outline of gesture generation

When finished, the participant goes into a T–pose once again.

2.3 Speech Associated Gesture Dataset

A total of 298 minutes (1047 sentences) of speech audio data with the motion data of the accompanying gestures were recorded. Additionally, video data of the participants was also recorded so the validity of the two data could be checked afterward. The data were trimmed one by one so 210 mins of audio and motion data in total. We will explain our annotation (later) and analysis later.

3 Summary of Speech–to–Gesture Generation

We propose a method to automatically generate gesture motions from speech audio. In this method, we will try to build a model to represent the relationship between speech content and gestures accompanying the speech by using a 5 layered Bi–Directional LSTM Network which can take into account both backward and forward consistencies over a long period of time.

Figure 2 shows an outline of our proposed method. As shown, first, a speech audio of one sentence (.wav) is converted to MFCC[8] feature vectors for each time window. MFCC is a widespread audio feature designed for speech recognition, taking into account human perceptual tendency[9, 12]. Therefore, we believe that the MFCC feature can hold sufficient information for language.

Next, the MFCC feature vector is fed into a Bi–Directional LSTM Network which has five layers. Then, 3D positions of entire body joints are predicted by regression of the network. LSTM can hold previous inputs for a relatively long duration. Hence, it should be effective for specific data modeling such as a gesture which is related to a whole sentence or sometime beyond a sentence. **Figure 2** shows the LSTM internal architecture. The network is trained by a speech and motion paired dataset we created.

The output of the network has small discontinuities between frames originating from noise in the input data or prediction errors. It is very disturbing for humans to see someone gesturing with such discontinuities. Therefore, we will address the problem in post processing by using two kinds of temporal filtering. One kind is a 1€ filter[2], and the other is a Moving Average (MA) filter.

We evaluated the results of gesture generation in a quantitative way. We used Average Position Error (APE) as the evaluation measure. APE compares the predicted positions with positions that originally accompanied the speech and it calculates Euclidean distance. **Figure 3** shows the APEs of 14 representative joints out of 64 joints in the human model. As shown, the errors were all under 10 cm, and naturally wrist joints had the biggest error because the errors were accumulated from the root of joint hierarchy to the tip.

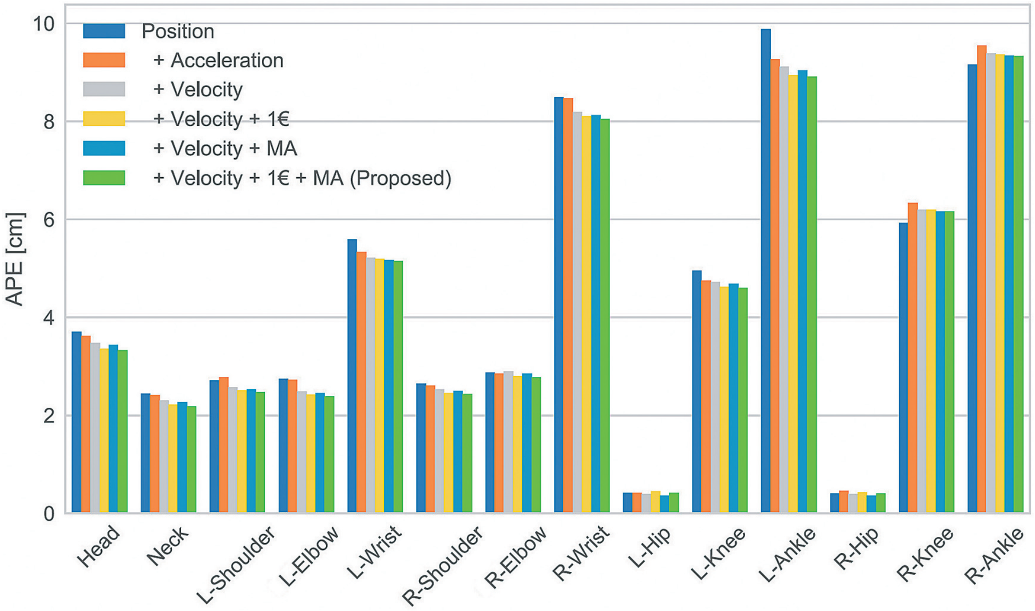
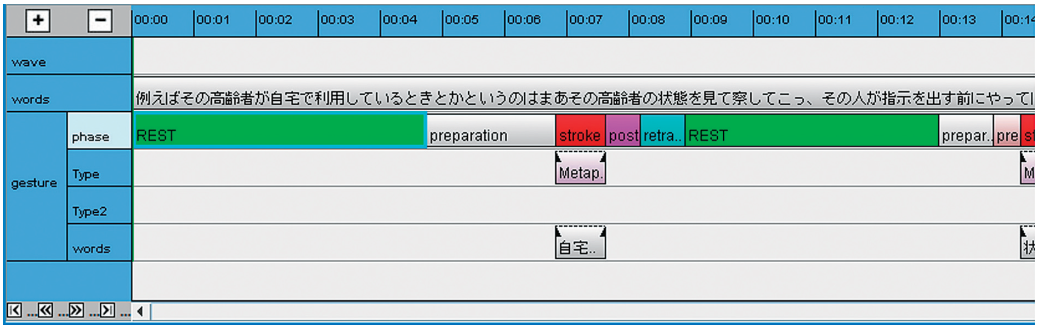


Figure 3: Results of Gesture Generation

4 Annotation

Two annotators, who preliminary learned gesture anatomy in time and kind, annotated the dataset by using an annotation tool Anvil (**Figure 4**). The target of annotation was a part of the dataset, 240 sentences out of 1047 sentences. 7 gesture phases and 4 gesture kinds were annotated.

Gesture phases are classifications in time of a gesture sequence from the start to the last. they are classified as rest, preparation, pre−stroke hold, stroke, post−stroke hold, retraction. Rest phase is the start position where arms were in relax state. Preparation is the preliminary movement for next phase, stroke. Stroke is the main phase of gesture which shows the function of the gesture along with



例えばその高齢者が自宅で利用しているときとかというのはまあその高齢者の状態を見て察してこっ、その人が指示を出す前にやってI

Figure 4: Annotation

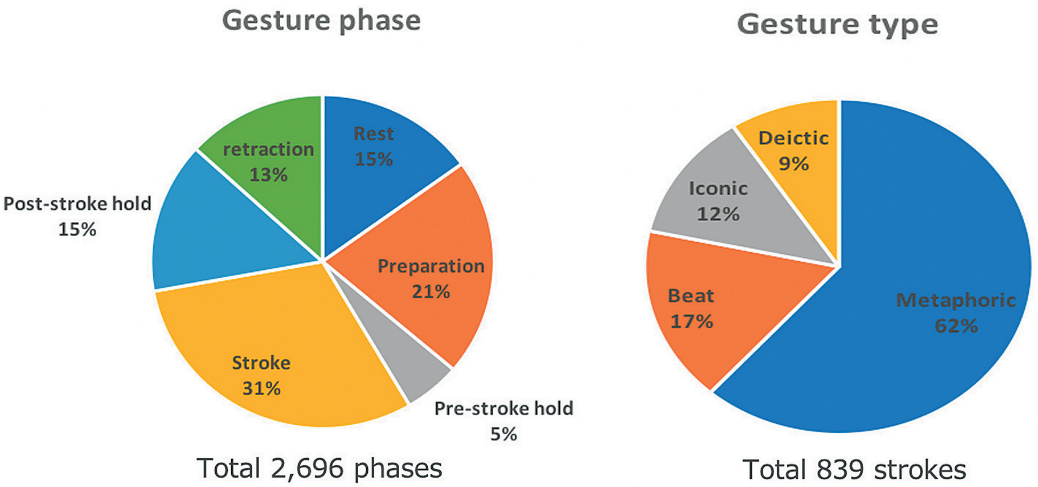


Figure 5: Results of Dataset Analysis on 240 out of 1047 sentences

speech. Pre–stroke hold and post–stroke hold is the typically small amount of stopping motion between preparation and stroke or stroke and retraction. Retraction is returning motion to the rest position.

Gesture kinds are classifications in meaning or function of gesture. Beat gesture, iconic gesture, metaphoric gesture, deictic gesture are the 4 types of gesture described in section 2.

In the annotation process, one annotator picked up a speech and perform annotation by using Anvil. Then, the other annotator checked the same speech and the annotation data again. If the two annotators had different opinion on the annotations, they determined based on discussions. **Figure 5** shows the results of annotations. In total, 2,696 phases and 839 strokes were annotated.

5 Gesture Phase Estimation

5.1 Overview of Proposed Method

To enrich the entire dataset, we conducted gesture phase estimation. As shown in the previous section, we annotated 240 sentences out of 1047 sentences in dataset. By using this annotated data, we will train a neural network model to predict a gesture phase sequence from a human pose sequence.

We used the same architecture as the one used in gesture generation except for the last layer. The last layer has 7 dimensions representing 7 gesture phase and the activation function is replaced from ReLU to soft–max function. The loss function is also replaced from mean square error to categorical cross–entropy.

We will explain the proposed gesture phase estimation method in detail. First, a human pose sequence is divided into frame by frame. We used three dimensional coordinates of all 64 joints. Thus, we will get T length time–series human pose vectors $X=\{\boldsymbol{x}_t\}$, $t=1, ..., T$. Next, $\boldsymbol{x}_t$ is fed into the network along with human pose vectors of S steps backward and forward. Thus, the actual input at time t will be the following : $\{\boldsymbol{x}_{t-S}, ..., \boldsymbol{x}_{t-1}, \boldsymbol{x}_t, \boldsymbol{x}_{t+1}, ..., \boldsymbol{x}_{t+S}\}$. In this paper, the time step t was 0.05 seconds (20 steps per second), the number of context steps S was 30 (1.5 seconds backward and forward, which is a 3.0 second time window).

By feeding the human pose vectors $\{\boldsymbol{x}_{t-S}, ..., \boldsymbol{x}_{t-1}, \boldsymbol{x}_t, \boldsymbol{x}_{t+1}, ..., \boldsymbol{x}_{t+S}\}$ into the Bi–Directional LSTM Network, that outputs probability of 7 gesture phases $\boldsymbol{P}_t=\{\boldsymbol{p}_t^i\}$, $i=1, ..., K$, where K is 7 which is the number of gesture phase categories we will predict.

5.2 Network Architecture

Figure 6 shows the Bi–Directional LSTM Network architecture we used. The network consisted of five layers. Layers $\boldsymbol{h}^1-\boldsymbol{h}^3$ and $\boldsymbol{h}^5$ were fully connected layers which are not recurrent and $\boldsymbol{h}^4$ was a Bi–Directional LSTM layer. In **Figure 6**, numbers on the left side show the number of feature dimensions for each layer. The network design was based on Deep Speech [10], which achieved one of the most successful speech recognition attempts.

$\boldsymbol{h}^1$, as described above, takes an human pose vector x_t of current step t along with context human pose vectors of S steps backward and forward. $\boldsymbol{h}^2$ and $\boldsymbol{h}^3$ take outputs from previous layers. There-

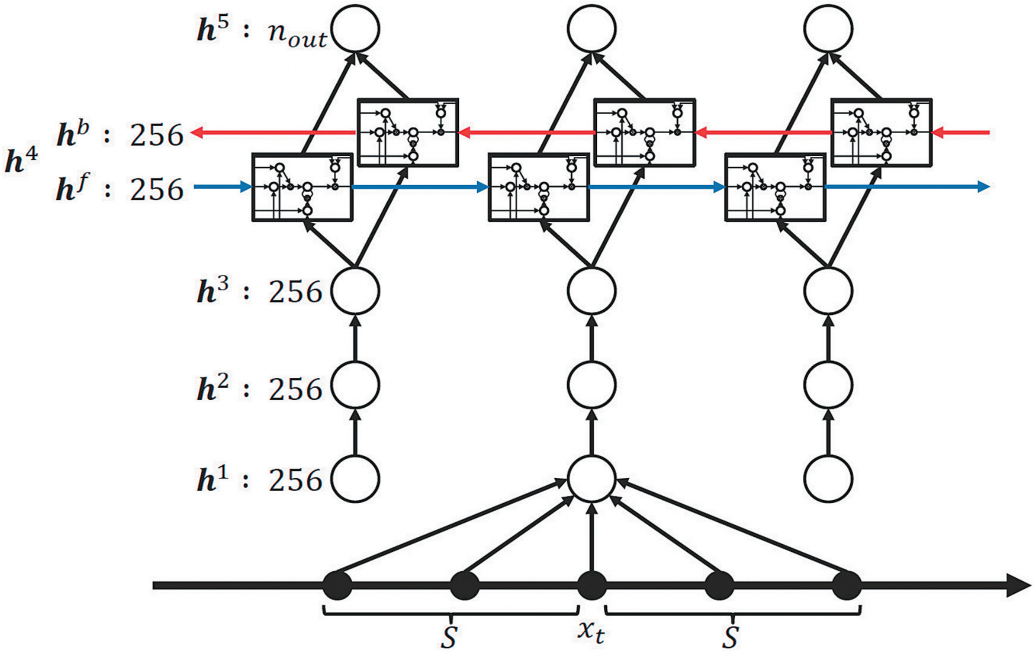


Figure 6 : Network Architecture

fore, $\boldsymbol{h}^1 - \boldsymbol{h}^3$ of step t is calculated as

$$\boldsymbol{h}_t^l = g\left(\boldsymbol{W}^l \boldsymbol{h}_t^{l-1} + \boldsymbol{b}^l\right), \tag{1}$$

where $\boldsymbol{W}^l$ and $\boldsymbol{b}^l$ are the weight and bias parameters of $\boldsymbol{h}^l$ respectively, and g is an activation by Rectified Linear Unit (ReLU) [14].

$\boldsymbol{h}^4$ is a Bi–Directional LSTM layer which consists of forward LSTM units $\boldsymbol{h}^f$ calculating from $t=1$ to $t=T$ and backward LSTM units $\boldsymbol{h}^b$ calculating backwards from step $t=T$ to $t=1$ as shown below (where M is the calculation of the LSTM unit).

$$\boldsymbol{h}_t^f = g\left(M\left(\boldsymbol{W}^4 \boldsymbol{h}_t^3 + \boldsymbol{W}^f \boldsymbol{h}_{t-1}^f + \boldsymbol{b}^4\right)\right), \tag{2}$$

$$\boldsymbol{h}_t^b = g\left(M\left(\boldsymbol{W}^4 \boldsymbol{h}_t^3 + \boldsymbol{W}^b \boldsymbol{h}_{t+1}^b + \boldsymbol{b}^4\right)\right). \tag{3}$$

Lastly, the last layer $\boldsymbol{h}^5$ is calculated as below.

$$\boldsymbol{h}_t^5 = \boldsymbol{W}^5\left(\left\{\boldsymbol{h}_t^f, \boldsymbol{h}_t^b\right\}\right) + \boldsymbol{b}^5, \tag{4}$$

In addition, to avoid overfitting and to make learning stable, Batch Normalization [11] and 10% Dropout [15] are applied to $\boldsymbol{h}^1 - \boldsymbol{h}^4$. Also, as mentioned above, for the last layer, we used soft–max functions as an activate functions.

5.3 Training

24 sentences were excluded for evaluation, and we used 80% of the rest (172 sentences) for the training, 20% (44 sentences) for the validation. We used categorical cross–entropy for the loss function. The loss function L was defined as

$$L(\boldsymbol{v}, \boldsymbol{y}) = -\sum_{t=1}^{T} \sum_{i=1}^{K} \boldsymbol{y}_t^i \log \boldsymbol{v}_t^i, \tag{5}$$

where $\boldsymbol{y}$ was the output of the Bi–Directional LSTM Network and $\boldsymbol{v}$ was the Ground Truth. We trained the network for 500 epochs which was enough to converge the learning (as shown in **Figure 7**).

5.4 Results and Discussion

We used 24 sentences for evaluation which were not use in training process. The results of gesture phase estimation for the test sentences are described in **Table 1**. As shown, the proposed method the best predict rest phase (0.93 in f–value). The reason why the rest phase was the most predictable is

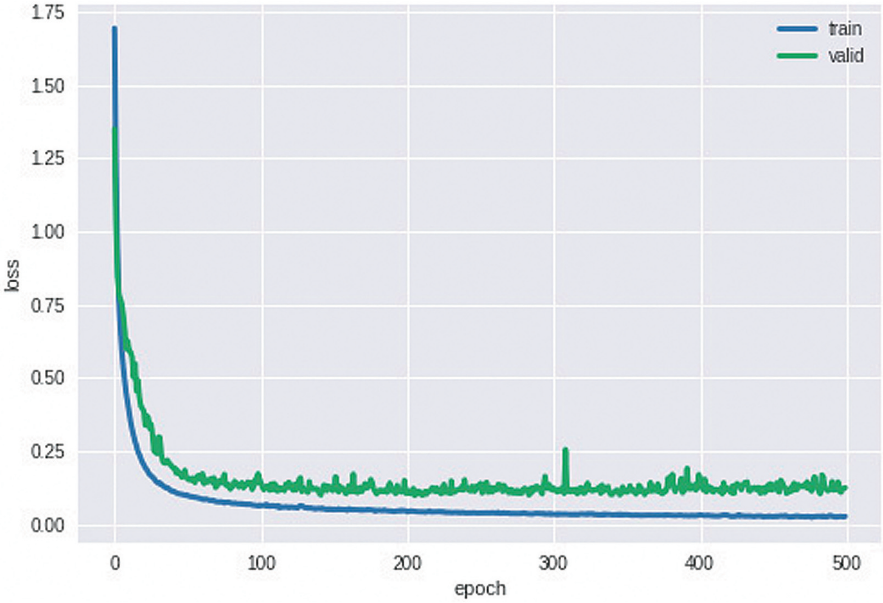


Figure 7 : Training Loss and Validation Loss

Table 1 : Results of Phase Estimation

	rest	prep	pr–hold	stroke	pst–hold	retraction
PRECISION	0.92	0.65	0.29	0.68	0.40	0.71
RECALL	0.94	0.68	0.31	0.52	0.55	0.67
F–VALUE	0.93	0.67	0.30	0.59	0.47	0.69

that the speakers' hands and arms performs the same positions for the most of time. The variety of positions are the least among the gesture phrases. At the same time, the speech signal is also salient. The rest phase motions tend to performed along with the silence.

On the other hand, pre–stroke–hold (pr–hold) and post–stroke hold (pst–hold) were the most difficult to predict (below 0.5 in f–value). These two phase were the least seen in the dataset. Although according to the annotation analysis, the proportion of pre–stroke was 5% and post–stroke was 15%, these percentage was calculated as the number of phrases. When counted by time–frame the percentage will drop significantly, because the hold states only last a moment. This worked as an disadvantage in out data–driven approach.

6 Conclusions

In our previous research, we have created a Japanese speech–to–gesture dataset where we recorded 298 mins of speech audio data and the motion capture data of the accompanying gestures (trimmed into 210 mins). We also had tackled speech–to–gesture generation by using a data–driven approach on the dataset, and the attempt had partially succeeded. To improve data–driven gesture generation, we proposed gesture phase estimation method to enrich the dataset. In this paper, We annotated seven gesture phases (stroke hold, stroke, post–stroke hold, retraction) to 240 sentences out of 1047 sentences of our previous dataset. Then, to extend the annotations to all over the dataset, we conducted a gesture phase estimation by using Bi–directional Long–Short Term Memory (Bi–directional LSTM). The network consists of five layers. The first three layers were fully connected layers, the fourth layer was Bidirectional LSTM, and the last layer was also fully connected layer. The network was trained to estimate one of seven gesture phases frame by frame, taking a time–series of motion data as an input. The result showed that overall accuracy of seven gesture phase estimation was 0.61 in precision, 0.61 in recall, and 0.61 in f–value. The most successful phase in the estimation was rest phase scored 0.92 in precision, 0.94 in recall and 0.93 in f–value.

Acknowledgements

This work was supported by Hokkai-Gakuen Kenkyu Josei (Sogo) 2019-2020.

References

[1] Timothy W Bickmore, Laura M Pfeifer, Donna Byron, Shaula Forsythe, Lori E Henault, Brian W Jack, Rebecca Silliman, and Michael K Paasche–Orlow. Usability of conversational agents by patients with inadequate health literacy: Evidence from two clinical trials. *Journal of Health Communication*, 15(S2) : 197–210, 2010.

[2] Géry Casiez, Nicolas Roussel, and Daniel Vogel. 1€ filter: A simple speed–based low–pass filter for noisy input in interactive systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 2527–2530, 2012.

[3] Justine Cassell. *Embodied conversational agents*. MIT press, 2000.

[4] Justine Cassell, Stefan Kopp, Paul Tepper, Kim Ferriman, and Kristina Striegnitz. Trading spaces: How humans and humanoids use speech and gesture to give directions. *Conversational Informatics*, pages 133–160, 2007.

[5] Justine Cassell, Hannes Högni Vilhjálmsson, and Timothy Bickmore. Beat : the behavior expression animation toolkit. In *Proceedings of the SIGGRAPH Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pages 477–486, 2001.

[6] Chung–Cheng Chiu and Stacy Marsella. How to train your avatar: A data driven approach to gesture generation. In *Proceedings of the International Conference on Intelligent Virtual Agents (IVA)*, pages 127–140, 2011.

[7] Chung–Cheng Chiu, Louis–Philippe Morency, and Stacy Marsella. Predicting co–verbal gestures: a deep and temporal modeling approach. In *Proceedings of the International Conference on Intelligent Virtual Agents* (*IVA*), pages 152–166, 2015.

[8] Steven Davis and Paul Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4) : 357–366, 1980.

[9] Moataz El Ayadi, Mohamed S Kamel, and Fakhri Karray. Survey on speech emotion recognition: Features, classi-

fication schemes, and databases. *Pattern Recognition*, 44(3) : 572–587, 2011.

[10] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. Deep speech: Scaling up end–to–end speech recognition. arXiv preprint arXiv: 1412.5567, 2014.

[11] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the International Conference on Machine Learning (ICML),* pages 448–456, 2015.

[12] A Lawson, Pavel Vabishchevich, M Huggins, P Ardis, Brandon Battles, and A Stauffer. Survey and evaluation of acoustic features for speaker recognition. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* pages 5444–5447, 2011.

[13] David McNeill. *Hand and mind: What gestures reveal about thought.* University of Chicago press, 1992.

[14] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the International Conference on Machine Learning (ICML),* pages 807–814, 2010.

[15] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout : A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1) : 1929–1958, 2014.