Speech2Pickup : Speech embedding based Human-Robot Collaboration model for multi object robot grasping task

Minu Kim[1] Yunho Kim[2]

***Abstract*— In this paper, we propose a neural network architecture for the speech-based automatic robot grasping system, named as <*Speech2Pickup*>. It is based on *Text2Pickup* [1] model which takes command texts as an input and performs the task: picking up the proper object after interpreting the instruction. <*Speech2Pickup*>, however, takes naive speech as the model input rather then transcribing into text which is similar to the speech understanding procedure of humans: the end-to-end, one step approach. To learn meaningful representation both from speech input and state image, we used a mixture of speech embedding encoder and Stacked Hourglass Network [2] for the model architecture. We experimented the model using two speech embedding encoders(word unit embedding and sentence unit embedding) and compared the performance with two step approach, which is concatenation of ASR(i.e. Automatic Speech Recogniton) and Text2Pickup. Finally we checked the potential and efficiency of the one step approach by quantitative analysis.**

Index terms: End to end learning, speech embedding, representation learning automatic speech recognition, robotics, Human-Robot Interaction, encoder-decoder model

I. INTRODUCTION

Due to the development of machine learning, various robotics tasks, which require complicated task formulations and solutions, were solved using empirical methods. Furthermore, the robotics tasks became broader and more complex. Spotlighted tasks include HRI(i.e. Human-Robot Interaction, also called as Human-Robot Collaboration). One of the goals of HRI is to build a highly efficient pipeline and models for the situations of human-robot collaboration. Current researches in HRI try to solve variety of tasks including visual language navigation [3]–[5] and multi object grasping [1], [6], [7].

Speech is one of the effective means of communication in humans which occupies most of the everyday life. Thus, to boost up the efficiency of HRI models and make them more useful in real life, speech needs to be the main input feature of the models. However, speech, which includes acoustic features and linguistic features, has countless variables including speaker characteristics(e.g. gender, tone, pitch, speed) and environment(e.g. condition of microphone, background noise). This makes speech based systems difficult to be robust.

Due to the development of machine learning and accumulated big data, ASR(i.e. Automatic Speech Recognition) models have been highly improved and become publicly available(e.g. Google cloud Speech-to-Text API). Therefore, it is one choice to concatenate ASR with text input based models to use speech as model inputs(hereafter, we will call this two step approach). Text input based models have relatively low variables compared to speech input based models which makes it more robust.

However, these kinds of 2 step approach have several drawbacks: (1) Error accumulation will happen during the process because both the ASR system and the text input based models are not perfectly accurate. This will greatly degrade the performance of the concatenated model.(2) Time delay will happen because ASR systems, which has difficulties installing on-device, usually rely on cloud computing. This is critical in interacting scenario with humans and robots. Furthermore, 2 step approach is far from how humans interpret speech. Humans actually understand speech in itself and don't interpret them to text for understanding.

Current research that compared the accuracy between two step approach and E2E(i.e. end to end) approach on the task of NER(i.e. Named Entity Recognition) showed the result of error accumulation in two step approach and effectiveness of the jointly trained behavior [8]. These results supported our intuition of the E2E approach for the HRI model based on speech input.

In this paper, we propose *Speech2Pickup*, a speech embedding based HRI model for multi object robot grasping task which is trained in an end to end manner. We compared the model accuracy and time efficiency with two step approach and checked the effectiveness of the one step approach. In addition, rather transcribing speech into text, our model uses speech feature itself for context understanding which could be more similar to the way human use.

In summary, the key contributions of this paper are;

- We propose *Speech2Pickup* model which uses speech itself for understanding context rather than transcribed text. This model is unique because it is more similar to the way humans understand speech than the two step approach.
- We analyzed the performance of the model in two respects: (1) Prediction accuracy (2) Time efficiency, and verified the effectiveness compared to the two step approach.

II. RELATED WORK

A. HRI models for multi object robot grasping

Among various HRI models for object grasping, [1] and [7] are the most similar works done. Each works takes differ-

[1]Department of Business Administration, Seoul National University, saintminu@snu.ac.kr

[2]Department of Mechanical Engineering, Seoul Natioanl University, awesomericky@snu.ac.kr

*This work was done as an Machine Listening (Fall2020) final project.

*Codes at `github.com/awesomericky/Speech2Pickup`

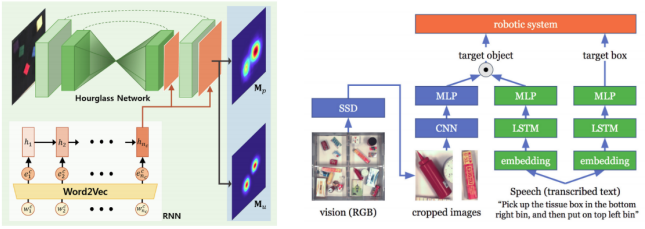

Fig. 1. Previously proposed model architectures for multi object grasping in the human-robot collaboration scenario (Left: [1], Right: [7])

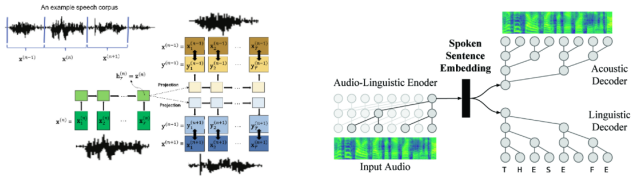

Fig. 2. Previously proposed model architectures for speech embedding (Left: word unit embedding [9], Right: sentence unit embedding [10])

ent approaches to improve model effectiveness. [1] focused on systematic reasoning of the robot behavior when grasping an object. By modeling the architecture with Stacked Hourglass Network [2] and RNN encoder, the model outputs a heat map with an attention given on the object instructed. They uses text as the model input which differs with our model. [7] focused on widening the span of the dataset for robust model performance. They used Amazon Mechanical Turk to collect unconstrained language instructions and trained the model using relatively simple architecture. In the experiment, they used speech as the model input and created realistic scenario. However, they used two step approach, using Google Chrome Web Speech API to transcribe speech into text, which differs with our model that uses speech feature itself. Model architectures for each works are shown in Fig 1.

B. Speech embedding models

To interpret the context of speech, a variety of embedding models that learn meaningful representations were proposed, each differentiated on the focusing speech unit. We mainly focus on two embedding models here.

Basically, a word unit embedding approach is a traditional method for the text-based language processing. In the NLP (i.e. Natural Language Processing) fields, Word2Vec model, which uses contexts of word as a source for word embedding, is quite powerful and highly usable. [9], thus, apply this to their research, however, they have made a little change. Instead of using word itself as a source for embedding, they used Mel frequency. So, they have first made Mel frequency files for each sentence and chunked them into word levels, which also is a preliminary work for text-based processing; the only difference here is that each sentence data is audio-recorded or text-recorded. [9] uses the RNN-based encoder-decoder to learn the model; the encoder here is Mel frequencies for each word, and decoder here is contexts for each word. The final embedding vector is the latent vector made in this architecture.

[10] is a sentence unit speech embedding model(hereafter, we will call sentence embedding model). As speech includes both acoustic features and linguistic features, the model is composed with three parts: global feature encoder, acoustic feature decoder, and linguistic feature decoder. Each component is a temporal convolution neural network using 1D causal and dilated convolutions. The model learns each feature using a multitask learning framework. Model architectures for each embedding methods are shown in Fig 2.

III. Proposed Method

In this section, we explain the architecture and the training method of *Speech2Pickup* model. For great performance of the model, three points should be considered: (1) Ways to learn meaningful speech representation from naive speech inputs (2) Ways to make the model available for systematic reasoning of the robot behavior (3) Ways to prepare wide span dataset with unconstrained speech inputs. For the first point, the encoders of speech embedding models explained in the previous section were used to project speech inputs in latent vectors. To extract features efficiently and make the training process shorter, embedding models were pretrained with the given dataset. Considering experiments in [9], [10], MFCCs (Mel Frequency Cepstral Coefficient) and log Mel Spectogram were used as initial features for the word embedding model and the sentence embedding model respectively. For the second point, the modified version of the model proposed in [1] were used as the total architecture. The pretrained word2vec model [11] and RNN encoder in the original model were replaced with speech embedding encoders to adapt the model input from text to speech. For the last point, dataset proposed in [7] could be used with several data augmentation methods applied together. However, due to the lack of time we trained model with the proposed dataset in [1] which is smaller than ones in [7]. We will leave training the model with wider dataset as a future work.

Finally, our proposed *Speech2Pickup* model is composed of speech embedding encoder and Stacked Hourglass Network [2], as in Fig 3. The model takes naive speech and a state image (RGB) as inputs and outputs a heat map with an attention given on the object instructed. In the experiment, we only tried the word unit embedding [9] and the sentence unit embedding [10] for the speech embedding encoder, however, other types of speech embedding models could also be used.

IV. Experiments

A. Dataset

For the experiment, we used the publicly available dataset proposed by [1]. The proposed dataset consists of 20349 text instructions, 478 state images, 2134 heatmaps. As the *Speech2Pickup* model requires speech as an input, we used Google cloud Speech-to-Text API to convert text into speech

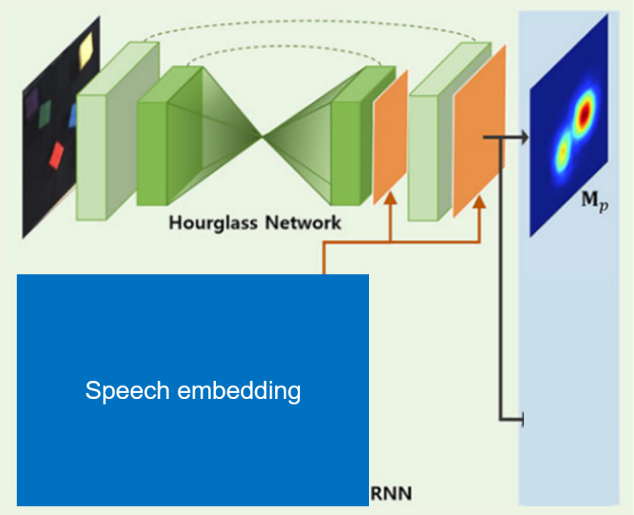

Fig. 3. The architecture of the *Speech2Pickup* model. The model is composed of speech embedding encoder and Stacked Hourglass Network [2]. Speech embedding encoder projects naive speech inputs in latent vectors. Stacked Hourglass Network outputs a heatmap conditioned on both state image and embedded speech input

inputs. As speech features greatly differ based on the gender of the speaker, texts were converted using two kinds of voice configuration, MALE and FEMALE. Language and accent were fixed to English and standard American accent. Final dataset compositions are as in Table I. 20 % of the state images are used for testing. Thus, MALE and FEMALE speech data are mixed equally both in train and test dataset. Additionally, converted speech was then aligned with text to pretrain the sentence unit embedding model. Montreal forced aligner [12] was used to align them.

B. Models

We selected two candidates, each using different speech embedding models, and one baseline for our experiment, as in Table II. The models were selected based on two purposes: (1) Compare model performance between different speech embedding models (2) Compare model performance between E2E approach and two step approach. For two step approach model, Google cloud Speech-to-Text API was used for ASR system.

C. Training and Testing

Both candidate models were trained in the same process. First, the speech embedding encoder was pretrained with the given dataset. For the candidate model using the sentence embedding encoder, we deleted the acoustic decoder to focus on linguistic features, thus pretraining the model in a single task learning framework. Then, the total model was jointly trained with the pretrained encoder. The baseline model(ASR + Text2Pickup), however, was not jointly trained. Text2Pickup network was first trained until the similar performace reported in [1]. Then, the model is just concatenated to the ASR system. All three models were trained until similar error rates. After training, testing was done on unseen state images with equally portioned MALE and FEMALE speech data.

Model performance were evaluated in two respects: (1) Prediction accuracy (2) Time efficiency. Prediction accuracy is the ratio between the number of successful prediction and the number of test data. Predicted block position corresponds to the highest value on the heat map and the prediction is defined as successful when the distance between the predicted block position and the ground truth position is less than 20 pixels, which is the half of the block size when the image size is 256. Time efficiency is evaluated by computing the time the model takes to produce an output after given a state image and speech input. For each model, model inference time is calculated for each test data and the minimum, average, and maximum time are reported in the result.

Speech	State image	Heatmap
40698	478	2134

TABLE I
COMPOSITION OF THE DATASET BEING USED

V. Results and Discussion

Model performances for all three models were evaluated based on prediction accuracy and time efficiency. The evaluated results on test dataset are reported in Table III. Examples of *Speech2Pickup* model outputs are shown in Fig 4, 5. For the prediction accuracy, the baseline model(ASR+Text2Pickup) showed the greatest performance, followed by *Speech2Pickup B* and *Speech2Pickup A*. *Speech2Pickup B* showed almost similar performance with the baseline model which meant the potential of using naive speech as model inputs rather than transcribed text, which is similar to human understanding procedure. Just considering the prediction accuracy, we could not check error accumulation in the baseline model(two step approach). This is because the speech inputs were very clear without any background noise and we will leave training the model with data augmentation as a future work.

For the time efficiency, *Speech2Pickup B* showed the greatest performance, followed by *Speech2Pickup A* and the baseline model(ASR+Text2Pickup). For the baseline model, it took almost 6 times longer than *Speech2Pickup B* for model inference. This proved that two step approach, using ASR system to convert speech into text, causes time delay due to the fact that ASR system uses cloud computing. This is very critical especially in the interacting scenario between humans and robots. The baseline model(two step approach) also showed a large gap between minimum and maximum model inference time. This is because the performance of ASR system greatly differs based on the speech length and the network(e.g. 5G, Wi-Fi) condition. Comparing time efficiency between our two *Speech2Pickup* models, the model with word unit speech embedding showed longer inference time than the model with sentence unit speech embedding

	Embedding model	Number of steps
Speech2Pickup A	Word unit speech embedding	1 step (E2E)
Speech2Pickup B	Sentence unit speech embedding	1 step (E2E)
ASR + Text2Pickup (baseline)	-	2 step

TABLE II

MODEL CANDIDATES AND BASELINE

*	**Accuracy (%)**	**Time (ms)**		
		minimum	average	maximum
Speech2Pickup A	89.36	939	953	1030
Speech2Pickup B	97.27	202	210	238
ASR + Text2Pickup (baseline)	98.59	760	1221	2443

TABLE III

MODEL EVALUATION RESULTS

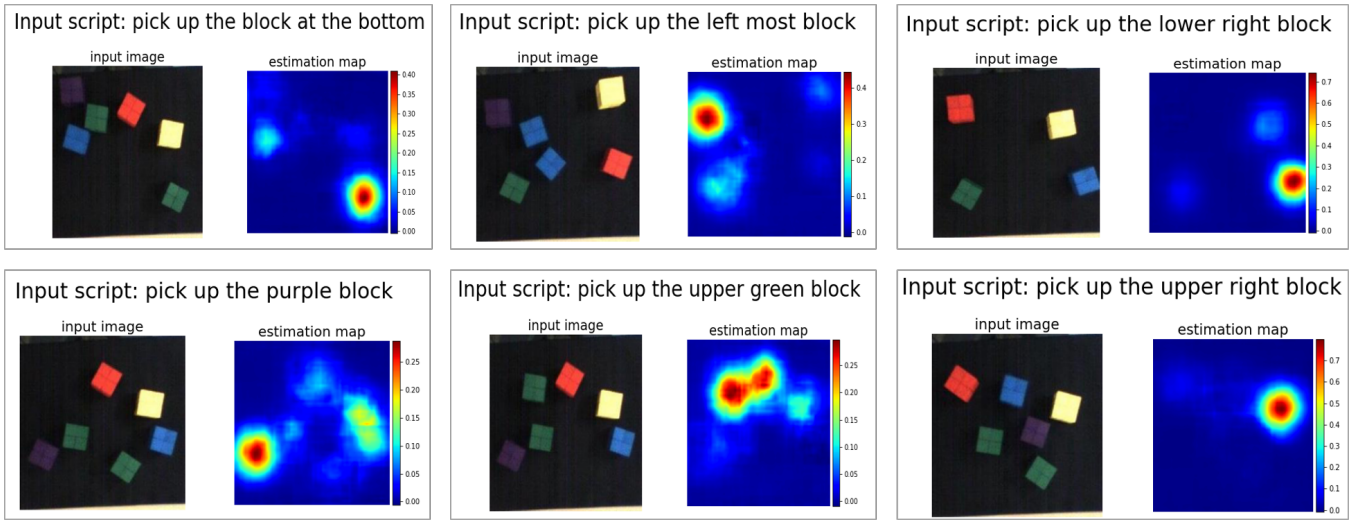

Fig. 4. Results of Speech2Pickup network with word embedding model

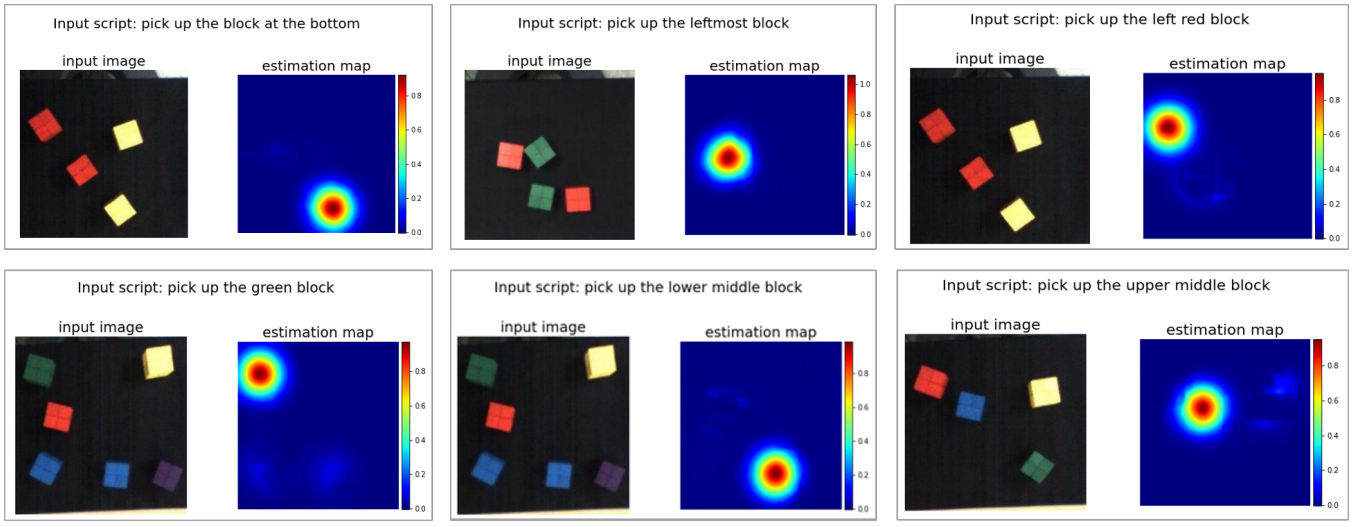

Fig. 5. Results of Speech2Pickup network with sentence embedding model

did. This is because word unit speech embedding requires speech to be parsed into word level before projecting to latent vectors. In the experiment, we used [13] algorithm to parse speech into word level. This additional process, which is required for word unit speech embedding, shows the inappropriateness in real world interacting scenario.

VI. Conclusions

Speech2Pickup extends *Text2Pickup* by taking naive speech inputs directly. Unlike the traditional two step approach, this model doesn't transcribe speech into text, which takes lower inference time than the two step model. What needs to be noted is that the direct *Speech2Pickup* model shows nearly the same performance on task accuracy with the two step model with shorter model inference time which is important in interacting scenario. The fact that speech can contain linguistic features (e.g. paralanguage expression, semantic mood information, etc.) is another powerful potential of *Speech2Pickup* model. If the input speech data are more extended, and contains wider linguistic information expressed by sound(i.e. prosody), the model will show more robust performance with short inference time.

In this paper, on the other hand, all inputs are artificially generated speech: not natural human-generated sounds. The final objective of this research, however, is a real time *Speech2Pickup* system in which inputs are generated in all different circumstances. So, to process natural voice inputs, it needs lots of naturally produced training data recorded in various environments; not just adding a noise to artificial sounds, and a more complicated speech encoder should be adopted. Thus, we leave training the model with larger span and improving the speech encoder as a future work.

Contribution

Yunho Kim worked on research proposal, data generation, ASR based *Text2Pickup* embedding, *SentenceEM* embedding and evaluation on pickup tasks. He wrote the paper on Introduction, Related Work, Proposed Method, Experiments, Results and Discussion, and References. (5/10)

Minu Kim followed Yunho's propsal and worked on *Speech2Vec* embedding and *Speech2Vec* evaluation on pickup tasks. He wrote the paper on Abstracts, Related Work, Results and Discussion, and Conclusions. (5/10)

References

[1] Hyemin Ahn, Sungjoon Choi, Nuri Kim, Geonho Cha, and Songhwai Oh. Interactive text2pickup networks for natural language-based human–robot collaboration. *IEEE Robotics and Automation Letters*, 3(4):3308–3315, 2018.

[2] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, pages 483–499. Springer, 2016.

[3] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3674–3683, 2018.

[4] Khanh Nguyen, Debadeepta Dey, Chris Brockett, and Bill Dolan. Vision-based navigation with language-based assistance via imitation learning with indirect intervention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12527–12537, 2019.

[5] Ta-Chung Chi, Minmin Shen, Mihail Eric, Seokhwan Kim, and Dilek Hakkani-tur. Just ask: An interactive learning framework for vision and language navigation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(03):2459–2466, Apr. 2020.

[6] Dipendra K Misra, Jaeyong Sung, Kevin Lee, and Ashutosh Saxena. Tell me dave: Context-sensitive grounding of natural language to manipulation instructions. *The International Journal of Robotics Research*, 35(1-3):281–300, 2016.

[7] Jun Hatori, Yuta Kikuchi, Sosuke Kobayashi, Kuniyuki Takahashi, Yuta Tsuboi, Yuya Unno, Wilson Ko, and Jethro Tan. Interactively picking real-world objects with unconstrained spoken language instructions. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3774–3781. IEEE, 2018.

[8] Hemant Yadav, Sreyan Ghosh, Yi Yu, and Rajiv Ratn Shah. End-to-end named entity recognition from english speech. *arXiv preprint arXiv:2005.11184*, 2020.

[9] Yu-An Chung and James Glass. Speech2vec: A sequence-to-sequence framework for learning word embeddings from speech. *arXiv preprint arXiv:1803.08976*, 2018.

[10] Albert Haque, Michelle Guo, Prateek Verma, and Li Fei-Fei. Audio-linguistic embeddings for spoken sentences. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7355–7359. IEEE, 2019.

[11] Google code archive https://code.google.com /archive/p/word2vec.

[12] Michael McAuliffe, Michaela Socolof, Sarah Mihuc, Michael Wagner, and Morgan Sonderegger. Montreal forced aligner: Trainable text-speech alignment using kaldi. In *Interspeech*, volume 2017, pages 498–502, 2017.

[13] James Robert, Marc Webbie, et al. Pydub. *Github*, 2011.