Towards Machine Comprehension of Spoken Content: Initial TOEFL Listening Comprehension Test by Machine

Bo-Hsiang Tseng, Sheng-Syun Shen, Hung-Yi Lee, Lin-Shan Lee

Graduate Institute of Communication Engineering
National Taiwan University
r02942037@ntu.edu.tw, r03942071@ntu.edu.tw, tikagkb93901106@gmail.com, lsllee@gate.sinica.edu.tw

Abstract

Multimedia or spoken content presents more attractive information than plain text content, but it’s more difficult to display on a screen and be selected by a user. As a result, accessing large collections of multimedia or spoken content is difficult and time-consuming than the latter for humans. It’s highly attractive to develop a machine which can automatically understand spoken content and summarize the key information for humans to browse over. In this endeavor, we propose a new task of machine comprehension of spoken content. We define the initial goal as the listening comprehension test of TOEFL, a challenging academic English examination for English learners whose native language is not English. We further propose an Attention-based Multi-hop Recurrent Neural Network (AM-RNN) architecture for this task, achieving encouraging results in the initial tests. Initial results also have shown that word-level attention is probably more relevant than sentence-level attention for this task with ASR errors.

Index Terms: spoken question answering, TOEFL, deep learning, attention model, recurrent neural networks

1. Introduction

With the popularity of shared videos, social networks, online course, etc., the quantity of multimedia or spoken content is growing much faster than what human beings can view or listen to. Accessing large collections of multimedia or spoken content is difficult and time-consuming for humans, even if these materials are more attractive for humans than plain text information. Hence, it will be great if the machine can automatically listen to and understand the spoken content, and even visualize the key information for humans. This paper presents an initial attempt towards the above goal: machine comprehension of spoken content. In an initial task, we wish the machine can listen to and understand an audio story, and answer the questions related to that audio content. TOEFL listening comprehension test is for human English learners whose native language is not English. This paper reports how today’s machine can perform with such a test.

The listening comprehension task considered here is highly related to Spoken Question Answering (SQA) [1][2]. In SQA, when the users enter questions in either text or spoken form, the machine needs to find the answer from some audio files. SQA usually worked with ASR transcripts of the spoken content, and used information retrieval (IR) techniques [3] or relied on knowledge bases [4] to find the proper answer. Sibyl [5], a

Figure 1: An example of TOEFL listening comprehension test. The story is given in audio format, and its manual transcription is shown. The question and choices are provided in text format.

factoid SQA system, used some IR techniques and utilized several levels of linguistic information to deal with the task. Question Answering in Speech Transcripts (QAST) [6–8] has been a well-known evaluation program of SQA for years. However, most previous works on SQA mainly focused on factoid questions like “What is name of the highest mountain in Taiwan?”. Sometimes this kind of questions may be correctly answered by simply extracting the key terms from a properly chosen utterance without understanding the given spoken content. More difficult questions that cannot be answered without understanding the whole spoken content seemed rarely dealt with previously.

With the fast development of deep learning, neural networks have successfully applied to speech recognition [9][11] or NLP tasks [12][13]. A number of recent efforts have explored various ways to understand multimedia in text form [14][19]. They incorporated attention mechanisms [17] with Long Short-Term Memory based networks [20]. In Question Answering field, most of the works focused on understanding text documents [21][24]. Even though [25] tried to answer the question related to the movie, they only used the text and image in the movie for that. It seems that none of them have studied and focused on comprehension of spoken content yet.

2. Task Definition and Contributions

In this paper, we develop and propose a new task of machine comprehension of spoken content which was never mentioned before to our knowledge. We take TOEFL listening comprehension test as a corpus for this work. TOEFL is an English examination which tests the knowledge and skills of academic English for English learners whose native languages is not English. In this examination, the subjects would first listen to an audio story around five minutes and then answer several questions according to that story. The story is related to the college life such as conversation between the student and the profes-
sur or a lecture in the class. Each question has four choices where only one is correct. An real example in the TOEFL examination is shown in Fig. 1. The upper part is the manual transcription of a small part of the audio story. The questions and four choices are listed too. The correct choice to the question sequentially at a time. In Fig. 3(A), the hidden layer output of the forward GRU (green rectangle) at time index $t$ is denoted by $y_f(t)$, and that of the backward GRU (blue rectangle) is by $y_b(t)$. After looking through all the words in the question, the hidden layer output of forward GRU network at the last time index $y_f(T)$, and that of backward GRU network at the first time index $y_b(1)$, are concatenated to form the question vector representation $V_Q$, or $V_Q = [y_f(T)||y_b(1)]^T$.

### 3.2. Story Attention Module

Fig. 3(B) shows the attention mechanism which takes the question vector $V_Q$ obtained in Fig. 3(A) and the story transcriptions as the input to encode the whole story into a story vector representation $V_S$. The word vector representation of the $t$-th word $S_t$ is constructed by concatenating the hidden layer outputs of forward and backward GRU networks, that is $S_t = [y_f(t)||y_b(t)]$. Then the attention value $\alpha_t$ for each time index $t$ is the cosine similarity between the question vector $V_Q$ and the word vector representation $S_t$ of each word, $\alpha_t = S_t \cdot V_Q$. With attention values $\alpha_t$, there can be two different attention mechanisms, word-level and sentence-level, to encode the whole story into the story vector representations $V_S$.

**Word-level Attention:** We normalize all the attention values $\alpha_t$ into $\alpha_t'$ such that they sum to one over the whole story. Then all the word vector $S_t$ from the bidirectional GRU network for every word in the story are weighted with this normalized attention value $\alpha_t'$ and sum to give the story vector, that is $V_S = \sum \alpha_t' S_t$.

**Sentence-level Attention:** Sentence-level attention means the model collects the information only at the end of each sentence. Therefore, the normalization is only performed over those words at the end of the sentences to obtain $\alpha_t''$. The story vector representation is then $V_S = \sum_{t=1}^{T} \alpha_t'' \cdot S_t$, where only those words at the end of sentences (eos) contribute to the weighted sum. So $V_S = \alpha_1'' \cdot S_1 + \alpha_2'' \cdot S_2$ in the example of the Fig. 3.

---

1. The symbol $[,]$ denotes concatenation of two vectors in this paper.
2. The symbol $\cdot$ denotes cosine similarity between two vectors.
3.3. Hopping

The overall picture of the proposed model is shown in Fig. 2, in which Fig. 3(A) and (B) are component modules (labeled as Fig. 3(A) and (B)) of the complete proposed model. In the left of Fig. 2 the input question is first converted into a question vector $V_{Q_0}$ by the module in Fig. 3(A). This $V_{Q_0}$ is used to compute the attention values $\alpha_t$ to obtain the story vector $V_{S_t}$ by the module in Fig. 3(B). Then $V_{Q_0}$ and $V_{S_t}$ are summed to form a new question vector $V_{Q_1}$. This process is called the first hop (hop 1) in Fig. 2. The output of the first hop $V_{Q_1}$ can be used to compute the new attention to obtain a new story vector $V_{S_t}$. This can be considered as the machine going over the story again to re-focus the story with a new question vector. Again, $V_{Q_1}$ and $V_{S_t}$ are summed to form $V_{Q_2}$ (hop 2). After $n$ hops ($n$ should be pre-defined), the output of the last hop $V_{Q_n}$ is used for the answer selection in the Section 3.4.

3.4. Answer Selection

As in the upper part of Fig. 2 the same way previously used to encode the question into $V_Q$ in Fig. 3(A) is used here to encode four choice into choice vector representations $V_{A_1}, V_{B_1}, V_{C_1}, V_{D_1}$. Then the cosine similarity between the output of the last hop $V_{Q_n}$ and the choice vectors are computed, and the choice with highest similarity is chosen.

4. Experiments

4.1. Experimental Setup

- Dataset Collection: The collected TOEFL dataset included 963 examples in total (717 for training, 124 for validation, 122 for testing). Each example included a story, a question and 4 choices. Besides the audio recording of each story, the manual transcriptions of the stories are also available. We used a pydub library to segment the full audio recording into utterances. Each audio recording has 57.9 utterances in average. There are in average 657.7 words in a story, 12.01 words in question and 10.35 words in each choice.

- Speech Recognition: We used the CMU speech recognizer - Sphinx [30] to transcribe the audio story. The recognition word error rate (WER) was 34.32%.

- Pre-processing: We used a pre-trained 300 dimension glove vector model to obtain the vector representation for each word. Each utterance in the stories, question and each choice can be represented as a fixed length vector by adding the vectors of the all component words. Before training, we pruned the utterances in the story whose vector representation has cosine distance far from the question’s. The percentage of the pruned utterances was determined by the performance of the model on the development set. The vector representations of utterances, questions and choices were only used in this pre-processing stage and the baseline approaches in Section 4.2 not used in the proposed model.

- Training Details: The size of the hidden layer for both the forward and backward GRU networks were 128. All the bidirectional GRU networks in the proposed model shared the same set of parameters to avoid overfitting. We used RmsProp [32] with initial learning rate of 1e-5 and momentum 0.9. Dropout rate was 0.2. Batch size was 40. The number of hop was tuned from 1 to 3 by development set.

| Model | Manual | ASR |
|-------|--------|-----|
| (a) Choice length | longest | 22.95% | 25.59% |
| (b) Within choices | similar | 36.07% | 27.87% |
| (c) Question choices | different | 24.59% | 22.95% |
| (d) Sliding Window | | 33.61% | 31.15% |
| (e) Memory Network | | 39.17% | 39.17% |
| (f) Our model | word | 49.16% | 48.33% |
| sentence | | 51.67% | 46.67% |

4.2. Baselines

We compared the proposed model with some commonly used simple baselines in [25] and the memory network [17].

- Choice Length: The most naive baseline is to select the choices based on the number of words in it without listening to the stories and looking at the questions. This included: (i) selecting the longest choice, (ii) selecting the shortest choice or (iii) selecting the choice with the length most different from the rest choices.

- Within-Choices similarity: With the vector representations for the choices in pre-processing of Section 4.1, we computed the cosine distance among the four choices and selected the one which is (i) the most similar to or (ii) the most different from the others.

- Question and Choice Similarity: With the vector representations for the choices and questions in pre-processing of Section 4.1, the choice with the highest cosine similarity to the question is selected.

- Sliding Window [25, 33]: This model try to found a window of $W$ utterances in the story with the maximum similarity to the question. The similarity between a window of utterances and a question was the averaged cosine similarity of the utterances in the window and the question by their glove vector representation. After obtaining the window with the largest cosine similarity to the question, the confidence score of each choice is the average cosine similarity between the utterances in the window and the choice. The choice with the highest score is selected as the answer.

- Memory Network [17]: We implemented the memory network with some modifications for this task to find out if memory network was able to deal it. The original memory network didn’t have the embedding module for the choices, so we used the module for question in the memory network to embed the choices. Besides, in order to have the memory network select the answer out of four choices, instead of outputting a word in its original version, we computed the cosine similarity between the the output of the last hop and the choices to select the closest choice as the answer. We shared all the parameters of embedding layers in the memory network for avoiding overfitting. Without this modification, very poor results were obtained on the testing set. The embedding size of the memory network was set 128, stochastic gradient descent was used as [17] with initial learning rate of 0.01. Batch size was 40. The size of hop was tuned from 1 to 3 by development set.

4.3. Results

We used the accuracy (number of question answered correctly / total number of questions) as our evaluation metric. The results are showed in Table 1. We trained the model on the manual transcriptions of the stories, while tested the model on the test-
In this paper, we create a new task with the TOEFL corpus. TOEFL is an English examination, where the English learner is asked to listen to a story up to 5 minutes and then answer some corresponding questions. The learner needs to do deduction, logic and summarization for answering the question. We built a model which is able to deal with this challenging task. On manual transcriptions, the proposed model achieved 51.56% accuracy, while the very capable memory network got only 39.17% accuracy. Even on ASR transcriptions with WER of 34.32%, the proposed model still yielded 48.33% accuracy. We also found that although sentence-level attention achieved the best results on the manual transcription, word-level attention outperformed the sentence-level when there were ASR errors.

5. Conclusions

Figure 4: Visualization of the attention weights in sentence-level and in word-level on a small section of the manual or ASR transcriptions of an example story given a question. The darker the color, the higher the weights. The question of this story is “What is a possible origin of Venus’ clouds?” and the correct answer choice is “Gases released as a result of volcanic activity.”
6. References

[1] P. R. C. i Umbert, “Factoid question answering for spoken documents,” Ph.D. dissertation, Universitat Politècnica de Catalunya, 2012.

[2] P. R. C. i Umbert, J. T. Borras, and L. M. Villodre, “Spoken question answering.”

[3] S.-R. Shiang, H.-y. Lee, and L.-s. Lee, “Spoken question answering using tree-structured conditional random fields and two-layer random walk.” in INTERSPEECH, 2014, pp. 263–267.

[4] B. Hixon, P. Clark, and H. Hajishirzi, “Learning knowledge graphs for question answering through conversational dialog.”

[5] P. R. Comas, J. Turmo, and L. Márquez, “Sibyl, a factoid question-answering system for spoken documents,” ACM Trans. Inf. Syst., 2012.

[6] J. Turmo, P. R. Comas, S. Rosset, O. Galibert, N. Moreau, D. Mostefa, P. Rosso, and D. Buscaldi, Multilingual Information Access Evaluation I. Text Retrieval Experiments: 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfu, Greece, September 30 - October 2, 2009, Revised Selected Papers. Springer Berlin Heidelberg, 2010, ch. Overview of QAST 2009, pp. 197–211.

[7] J. Turmo, P. Comas, S. Rosset, L. Lamel, N. Moreau, and D. Mostefa, “Overview of QAST 2008;” in Working Notes for the CLEF 2008 Workshop., 2008.

[8] D. Giampiccolo, P. Forner, J. Herrera, A. Peñas, C. Ayache, C. Forascu, V. Jijkoun, P. Osenova, P. Rocha, B. Sacaleanu, and R. Sutcliffe, Advances in Multilingual and Multimodal Information Retrieval: 8th Workshop of the Cross-Language Evaluation Forum. Springer Berlin Heidelberg, 2008, ch. Overview of the CLEF 2007 Multilingual Question Answering Track, pp. 200–236.

[9] G. E. Dahl, T. N. Sainath, and G. E. Hinton, “Improving deep neural networks for lvcsr using rectified linear units and dropout.” in Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. IEEE, 2013, pp. 8609–8613.

[10] L. Deng, G. Hinton, and B. Kingsbury, “New types of deep neural network learning for speech recognition and related applications: An overview.” in Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. IEEE, 2013, pp. 8599–8603.

[11] A. Graves, A.-r. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” in Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. IEEE, 2013, pp. 6645–6649.

[12] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, “A convolutional neural network for modelling sentences,” arXiv preprint arXiv:1404.2188, 2014.

[13] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, “Natural language processing (almost) from scratch,” The Journal of Machine Learning Research, vol. 12, pp. 2493–2537, 2011.

[14] J. Weston, S. Chopra, and A. Bordes, “Memory networks,” arXiv preprint arXiv:1410.3916, 2014.

[15] K. M. Hermann, T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom, “Teaching machines to read and comprehend,” in Advances in Neural Information Processing Systems, 2015, pp. 1684–1692.

[16] A. Bordes, N. Usunier, S. Chopra, and J. Weston, “Large-scale simple question answering with memory networks,” arXiv preprint arXiv:1506.02075, 2015.

[17] S. Sukhbaatar, J. Weston, R. Fergus et al., “End-to-end memory networks,” in Advances in Neural Information Processing Systems, 2015, pp. 2431–2439.

[18] A. Kumar, O. Irosy, J. Su, J. Bradbury, R. English, B. Pierce, P. Ondruska, I. Gulrajani, and R. Socher, “Ask me anything: Dynamic memory networks for natural language processing,” arXiv preprint arXiv:1506.07285, 2015.