HagiwaraLab at the NTCIR-13 QALab-3 Task

Yuanzhi Ke
Keio University, Japan
enshika8811.a6@keio.jp

Masafumi Hagiwara
Keio University, Japan
hagiwara@keio.jp

ABSTRACT

The objective of the term question subtask of QALab-3 in NTCIR-13 is to answer some historical questions by several words, instead of choosing the right answers from several options like the QALab-3 multi-choice subtask. We used recurrent neural networks (RNNs) to extract the answer. However, we encountered memory issues when we tried to input the retrieved documents from Wikipedia into the neural network. To solve the memory issues, we input the automatically summarized summaries of the documents instead of the original ones. They were summarized by the relevance scores based on the tf-idf weighted word embeddings. In this paper, we introduce our system for the term question subtask. We discussed the effects of the summarization technology, the length of the summary and the issue of multi-document summarization. The system can be improved by more carefully specified knowledge base, a better algorithm for summarization or more powerful machines.

Team Name

HagiwaraLab

Subtasks

Term Questions (English, Phase2)

Keywords

question answering, summarization, recurrent neural network, attention mechanism

1. INTRODUCTION

The subtask of term questions is new in QALab-3 [1]. The objective is to answer the question with some words, instead of choosing the right answers from options like the QALab-3 multi-choice tasks.

Though there have been datasets of term questions (e.g., the Stanford Question Answering Dataset (SQuAD) [2]), the subtask of term questions of QALab-3 has its own challenging points: there are various types of questions and the document of the correct answer for each question is not given. Hence we need to make our systems able to understand the requirements of different types of questions, and to obtain the answers in external data.

In the task, we encountered memory issues when we tried to use recurrent neural networks (RNNs) to extract the answer from the retrieved documents from Wikipedia. Henceforth, we summarized the documents before they were input into the neural networks, in order to shorten the input reasonably. Relevance based on the tf-idf weighted word embeddings is employed to rank sentences for summarization. In this paper, we would like to introduce our system for the subtask of term questions of QALab-3, discuss its performance and outlook.

2. THE CHALLENGES OF THE TERM QUESTIONS SUBTASK

2.1 The Challenges of Question Analysis

The subtask includes the following types of questions:

- The questions asking for some historical knowledge about the grand question and begin with “what”, “where”, “who”, etc. They are usually followed with an instruction. For example: “What is the treaty referred to in the underlined section (2)? Write the name of the treaty” (B792W10-3).
- The questions that ask for some historical knowledge about the grand question but do not begin with “what”, “where”, “who”, etc. For example, “Write the name of the accord in underlined section (6)” (B792W10-7).
- A small story related to the grand question and a question about the story. For example, “India, which advocated peaceful foreign diplomacy, took a leadership role in the Afro Asian Conference. However, India had violent disagreements, leading to war, with neighboring Pakistan immediately following its independence regarding a certain territory. Write the name of this territory” (B792W10-9)

It is challenging to recognize the type and extract the corresponding keywords by rule-based methods. Besides, in some sections, several questions are in the same sentence. It is also a challenge to recognize the boundaries of each question. Hence, we followed the RNN-based approach [4], took advantage of attention technology of RNNs to automatically fit different types of questions.

2.2 The Challenges of Answer Retrieval

Unlike SQuAD, almost no answer in the subtask is contained by the grand question, the question instruction or any other materials in the dataset. Therefore it is necessary to retrieve the external knowledge base. Besides, in the English track, only a dump of Wikipedia was provided as the knowledge base, which is large and not quite specified for the questions, contains redundant contents.

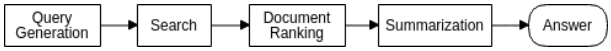

Figure 1: The process flow of our system. At first, the system generates a query for the question, and then searches the knowledge base. After that, it ranks the retrieved documents and summarizes them. Finally, an RNN is employed to find the answer from the summary.

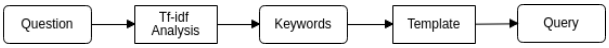

Figure 2: The procedure to generate the query. At first, the words with the highest tf-idf are selected as the keywords. Then a template is employed to comprise the query of the keywords.

Moreover, the length of each answer is not fixed. We do not know how long the answer should be. The answers in SQuAD are also of variable length and have been well answered by RNNs [3, 4]. The RNN-based approach searches the start and end position of the answer, hence the length of the answer can be automatically fitted.

2.3 Memory Issue

Most of the documents in the Wikipedia are long and our machine encountered memory issues when the total input length of the RNNs was longer than 1600 tokens at the task, which was shorter than the length of many Wikipedia documents. We had to input only a piece of text that most probably contains the correct answer instead of the whole document.

3. METHODOLOGY

The process flow of our system is as shown in Fig. 1. The system firstly generates a query from the question and searches the knowledge base for the related documents. After that, the top-N documents are summarized in a short text for the memory issues. Finally, the summary is input into the RNN to find a span in the summary which is the most probable to be the answer.

3.1 Query Generation, Search

For the less computational expense, we did not use neural networks for query generation. The procedure is as shown in Fig. 2.

We extract the keywords according to the tf-idf [5] of each entity. At first, we use Spacy [1] to extract the entites. Then the tf-idf of each entity in our system is defined as the following:

$$tfidf(s,w) = tf(s,w) \times \left(log \frac{1+n_S}{1+df(S,w)} + 1 \right) \quad (1)$$

The s, w here refer to one of the sentences and one of the entities, respectively. $tf(s,w)$ refers to the raw count of word w in sentence s. n_S refers to the total number of sentences. $df(S,w)$ refers to the count that how many sentences contain w.

[1]https://spacy.io/

The entities with the highest tf-idf are used as the keywords. Our submitted results are from the documents queried with the top-3 keywords.

After we got the keywords, we put them in a template of the query format of Apache Solr [2], a full-text search engine. Then, we input the query to Apach Solr for the related documents.

3.2 Document Ranking

We reranked the retrieved documents for the question by the averaged value of tf-idf weighted word embeddings in each document as the following:

$$Score_d = \frac{\sum_{w \in d} (tfidf(d,w) \circ v_w)}{\sum_{w_d \in d} (tfidf(d,w)} \quad (2)$$

Here, $Socre_d$ refers to the score to rank document d, v_w is the word embedding of word w, $tfidf(d,w)$ is computed similarly to $tfidf(s,w)$ in Section 3.1.

3.3 Summarization

We summarized the documents for the given question by choosing the sentences whose embeddings are closest to the question's. The method is simple for the cheaper computational cost. At first, we calculated the tf-idf weighted sum of the embeddings of the words in each question and each sentence in the retrieved documents as its sentence embedding. Formally, the sentence embeddings are defined as the following:

$$v_s = \sum_{w \in s} (tfidf(s,w) \circ v_w) \quad (3)$$

Here, v_s refers to the embedding of sentence s, v_w refers to the pre-trained word embedding of word w. $\circ$ refers to the element-wise production. The word embeddings are pre-trained with Wikipedia 2014 [3] and Gigaword 5 [4], opened by Jeffrey Pennington [5] [6].

Then, the system ranks all of the sentences in the retrieved documents by the cosine similarities of the embeddings of the question and each sentence. After that, the most similar sentences are concatenated as the summary.

Here is an example of the summaries extracted by our system at QALab-3:

> *confusing semantics of English Christendom equalling German Christenheit, French chr0̆0e9tient0̆0e9 vs. English Christianity equalling German Christentum, French christianisme. 900 (from Preslav, Bulgaria).]] Abrahamic Christian apocryphal gospels. A very few illuminated manuscript fragments survive on papyrus.*

3.4 Answer Extraction

We followed the Bi-directional Attention Flow [4] to extract the answer from the summarized text. The architecture is as shown in Fig. 3.

[2]http://lucene.apache.org/solr/
[3]https://dumps.wikimedia.org/enwiki/20140102/
[4]https://catalog.ldc.upenn.edu/LDC2011T07
[5]https://nlp.stanford.edu/projects/glove/

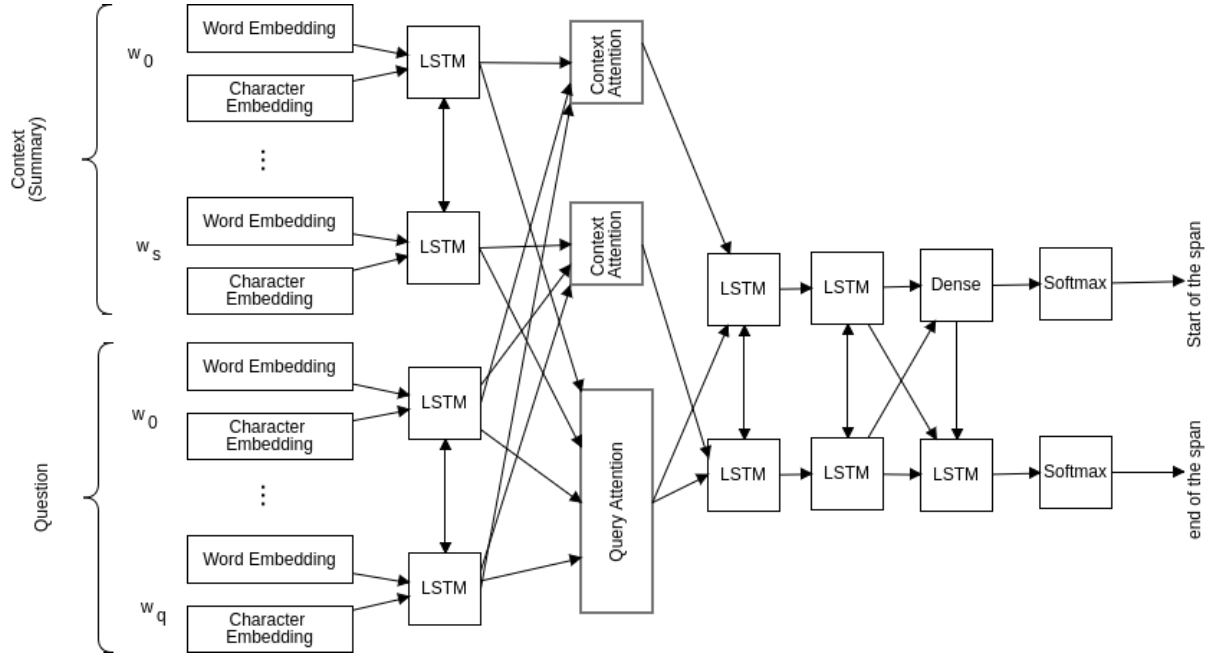

Figure 3: The architecture of the model to extract the answer span in the summary. Here, w_s refers to a word of the summarized sentences, w_q refers to a word of the question.

We concatenate the word embedding and the character embeddings for each word as the inputs. We used an RNN comprised of Long Short-term Memory(LSTM) [7] to extract the feature of each word in the summary, and the feature of each word in the question.

After that, the output of each lstm unit for the word in the summary was multiplied by an attention. Here, attention is the result of a softmax that models the relevance between the word and the question, whose input is the output of the corresponding lstm unit of the word, and all of the outputs of the lstm units of the question.

The output of the rnn for the question was also timed by another attention, whose input is all of the outputs of the lstm units of the words in the summary and the question, as a question feature weighted by the summary.

Then we input the corresponding weighted output of the words in the summary and the question into a bidirectional rnn [8] to get the start and the end position of the answer span. More detailedly, for the start position, the outputs of all the LSTMs were input into a full connected layer (i.e., dense layer) and then a softmax function to "classify" the input to one of all the possible start positions; for the end position, the start position is input into another LSTM, together with the outputs of the LSTMs for the end position.

4. VALIDATION

4.1 Setup

We validated our system on the term questions of Phase 1 and Phase 2 of QALab-3. As described before, Apache Solr was used to read the subset of world history of Wikipedia[6] as the knowledge base. This subset was used by the CMU Multiple-choice Question Answering System at NTCIR-11 QA-Lab [9, 10].

We did not perform any further preprocessing of the wiki subset. For all of the questions, three keywords are extracted to comprise the query, because we found that it brings a balance of coverage and accuracy. Besides, the input of the RNN in our system was limited to 1600 tokens at the QALab3 subtask. The memory was not enough for a longer input. Only the first 1600 tokens of the top-ranked documents were used. As we now have got a more powerful machine that can compute the neural network with 3200-token input. Hence we also compared with the performance when we input 3200 tokens of the top-ranked documents into the neural network. All the runs shared the same queries.

For training, because the training set provided by QALab-3 was not enough to train the RNN of our system, we used the training dataset of SQuAD to train it.

In the validation experiment described in this paper, we considered an answer was correct if it contained the gold standard.

4.2 Results

We compared the performance when we only used the top-1 retrieved documents and when we use the summary of the top-5 documents. The numbers of correct answers by different settings are shown in Table 1.

The performance on the summarized short texts is similar to the longer original documents, better than the original documents of the same length.

The group that summarizes the top-1 document in 15 sentences obtained the best performance for Phase 1. And the answers we submitted for Phase 2 were achieved by it. How-

[6]https://github.com/oaqa/ntcir-qalab-cmu-baseline/wiki/Solr-Instance-with-Indexed-Wikipedia-Subset

Table 1: The number of the correct answers of each runs in the validation.

	#Document	Length Limit	#Correct Answer(P1)	#Correct Answer(P1)
No Summary	1	No limit	Out of Memory	Out of Memory
No Summary	1	3200	8/69	**15/78**
No Summary	1	1600	4/69	13/78
Summary(5 Sentences)	1	1600	6/69	6/78
Summary(10 Sentences)	1	1600	5/69	7/78
Summary(15 Sentences, the submitted one)	1	1600	**9/69**	12/78
Summary(5 Sentences)	5	1600	2/69	5/78
Summary(10 Sentences)	5	1600	3/69	9/78
Summary(15 Sentences)	5	1600	2/69	11/78

Table 2: Comparison of the number of correct answers when different method was used for summarization. All the summary or the document were padded to 1600 tokens before they were input into the neural network. The top-1 document is used for summarization.

	#Document	#Correct Answer (P1)	#Correct Answer (P2)
Ours	1	**9/69**	**12/78**
TextRank	1	6/69	11/78
LSA	1	5/69	**12/78**
Ours	5	2/69	5/78
TextRank	5	5/69	10/78
LSA	5	6/69	10/78

ever, after we got the gold standard of Phase 2, now we have found that when we input the original document that is cut to 1600 tokens to the RNN, the system can get better performance for Phase 2.

Secondly, we noticed that the case of multi-document summarization deteriorates the performance. Our method for summarization is not suitable to generate the summary from multiple documents from Wikipedia. Meanwhile, for phase 1, when we summarize the top-5 documents, the summaries of 10 sentences surprisedly slightly outperformed those of 15 sentences.

5. DISCUSSION

5.1 Discussion on Summarization Method

We compared our summarization method with the following conventional ones: TextRank [11] and Latent Semantic Analysis (LSA) [13].

We compared the correct rates when different methods was used for summarization. We used the vanilla TextRank and LSA, which do not consider the question but extracted the main contents of the documents. The result is as shown in Table 2. Our method failed to outperform the others on the questions of Phase 2. It suggests that our method to involve the question for summarization is not obviously effective. We need to improve it in the future.

Besides, in the case of multi-document summarization, the performance of TextRank and LSA did not deteriorate. We found that they do not tend to choose the meta information so much like our method. An example is shown in Table 3. They are the summaries extracted from the top-5 documents by the different algorithms for the question A792W10-2, which is asking who organized the independence movement of Mongolia. We can see that the summaries by TextRank and LSA contain fewer meta information such as the ISBN and the subtitles in the documents. Meanwhile, it is interesting that the 3 algorithms obtained totally different summaries. Unfortunately, all of the summaries are not obviously related to the question. It indicates that probably none of them are suitable for the task, which suggests that we should explore other methods in the future.

5.2 Discussion on the Length of Summaries from Multiple Documents

In the case that a single document is used, the longer summaries bring better performance. However, in the case of multiple documents, the summaries of 10 sentences slightly outperform those of 15 sentences for Phase 1. On the other hand, it also surprised us that the performance with multiple documents tends to be worse than that with a single document. To find the reason, we checked what the multi-document summaries were like. An example of the multi-document summary of different length by our system is shown in Fig. 4.

We found that the documents were not stripped well and our method wrongly ranked some meta information (e.g. the ISBN number in the example) from each document as highly related sentences. Hence, when the most rated sentences from multiple documents are extracted to be a summary, the summary tends to contain more useless contents than the summaries extracted from a single document. On the one hand, too short summaries are fulfilled with the meta information. On the other hand, too long summaries contain too much lower ranked sentences. It led to the accident that 10-sentence summaries from top-5 documents were best for Phase 1. The issue is how to avoid ranking non-story sentences (e.g. ISBN number). We believe a cleaner knowledge base or more careful preprocessing would improve the performance for multiple documents.

5.3 Other Issues

The results also indicate that the queries were not generated carefully enough because the groups without summary in our experiments are also low (all below 20%).

Besides, another popular modern method for summarization is the Sequence-to-sequence Model [14]. However, our machine was not powerful enough to train such a model in reasonable time. For the same reason, we neither compared our system with the Coarse-to-Fine model [15]. We are sorry for them but would like to explore them when we get more

ISBN 0-19-826463-1 ==Reform Judaism in the UK== UK Reform Progressive movements in the UK. migration from the villages to cities by the countrys younger demographic. The government first selected 33,267 villages and provided 335 sacks of cement. ==Reform Judaism in North America== Reform Judaism denomination of American Jews today. Official bodies of the Reform Movement in North America include the Union for Reform Judaism, the Central Conference of American Rabbis, and Hebrew Union College-Jewish Institute of Religion. with a fellow miner]] Soviet history Taylorist efficiencies to over-achieve at work. The Stakhanovite movement was named after Aleksei Stakhanov, who had mined 102 tons of coal in less than 6 hours (14 times his quota). 16,600 villages that demonstrated success were then granted additional resources of 500 sacks of cement and a ton of iron bars. ==Reform movement in Judaism== North American Reform, , from The Jewish Religion: A Companion, Oxford University Press, 1995. Elements of Orthodoxy developed their cohesive identity in reaction to the Reform movement in Judaism. Also, the government-led centralized system brought corruption problems such as misuse of funding. Raising Seed Money 1: Through sample cooperative projects #Raising Seed Money 2: By cooperative works Step 2: Operation of the Project #Principles and standards for selecting projects #Planning a project #Persuading villagers {{Infobox Korean name |img= |caption=logo of the New Village Movement |hangul= |hanja=\uc0c8\ub9c8\uc744 |rr=Saemaeul Undong | mr=Saema\u016dl Undong }} The New Community Movement, also known as the New Village Movement, Saemaeul Movement or Saemaul Movement, was a political initiative launched on April 22, 1970 by South Korean president Park Chung Hee(\ubc15\uc815\ud76c,\u6734\u6b63\u7199) to modernize the rural South Korean economy. Although the Saemaul Movement has achieved a great success in reducing poverty and improving living conditions in rural areas, rapid industralization of South Korea changed the environment and diminished the momentum of the movement.

The end of the summary of 5 sentences

The end of the summary of 10 sentence

The end of the summary of 15 sentences

Figure 4: The results of summarization of the top 5 documents for question A792W10-2 in different length. The sentences in the documents were ranked by the metric introduced in Section 3.3. The top-ranked sentences were used to comprise the summary.

powerful machines.

6. RELATED WORKS

A framework for question answering on long documents called Coarse-to-Fine is recently reported [15]. The idea of Coarse-to-Fine looks similar to our system, however, the summarization is implicit, done by the attention mechanism. They input all the sentences into their model, and employ soft and hard attentions to implicitly rank and select the sentences. In their experiments, their model is on par with the baselines. In our cases, because we faced memory issues, we avoided using more complex neural networks like that. Hence we used explicit summarization method, although it may perform worse than the end-to-end approaches like Coarse-to-Fine.

7. CONCLUSIONS

In this paper, we introduced our system at NTCIR-13 QALab-3. We use an RNN with two kinds of attentions to extract the answers from the summaries of related documents. The reason why we used summaries instead of the original texts is the memory issues for long documents. With the proposed method, our system is able to achieve the similar correct rates with shorter inputs. However, the results for the task were disappointing. We believe it is because of the lack of careful addressing for the noise in the knowledge base, and unsuitable method for summarization. The system may be improved by a more carefully addressed knowledge base with noise removed, or a better summarization method of high noise tolerance. More powerful machines that allow larger and more complex neural networks may also bring better answers.

8. REFERENCES

[1] Overview of the NTCIR-13 qalab-3 task. In *the Proceedings of NTCIR-13*, pages 1–100, 2017.

[2] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. In *the Proceedings of EMNLP 2016*, 2016.

[3] Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. Gated self-matching networks for reading comprehension and question answering. In *the Proceedings of ACL 2017*, 2017.

[4] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. In *the Proceedings of ICLR 2017*, 2017.

[5] Hans Peter Luhn. A statistical approach to mechanized encoding and searching of literary

Table 3: The summary extracted by the different algorithms for question A792W10-2 as an example.

Ours	TextRank [11]	LSA [12]
ISBN 0-19-826463-1 ==Reform Judaism in the UK== UK Reform Progressive movements in the UK. migration from the villages to cities by the countrys younger demographic. The government first selected 33,267 villages and provided 335 sacks of cement. ==Reform Judaism in North America== Reform Judaism denomination of American Jews today. Official bodies of the Reform Movement in North America include the Union for Reform Judaism, the Central Conference of American Rabbis, and Hebrew Union College-Jewish Institute of Religion. with a fellow miner]] Soviet history Taylorist efficiencies to over-achieve at work. The Stakhanovite movement was named after Aleksei Stakhanov, who had mined 102 tons of coal in less than 6 hours (14 times his quota). 16,600 villages that demonstrated success were then granted additional resources of 500 sacks of cement and a ton of iron bars. ==Reform movement in Judaism== North American Reform, , from The Jewish Religion: A Companion, Oxford University Press, 1995.	==Basic steps== The basic steps of the Saemaul Movement Step 1: Basic Arrangements #Three arrangements for the start: People, Seed Money, Basic Principles #Forming a Core Group 1: Leaders #Forming a Core Group 2: Working Groups #Forming a Core Group 3: Applying the principles to existing organizations #Forming a Core Group 4: Sectional organizations #Raising Seed Money 1: Through sample cooperative projects #Raising Seed Money 2: By cooperative works Step 2: Operation of the Project #Principles and standards for selecting projects #Planning a project #Persuading villagers 1 - Setting a model to villagers #Persuading villagers 2 - Encouraging you can do it spirit #Collecting consensus 1- Small group meetings #Collecting consensus 2- General meeting of villagers #Let everybody play a their part #Preparing and managing the public property #Establishing the local Saemaul Movement center #Encouraging we are the one spirit #Cooperating with other communities and the government Step 3: Main Stage of the Project #Project 1 for living environment improvement: Improving the houses.	Other major figures influenced by the movement who became Roman Catholics included: *Thomas William Allies, Church historian and former Anglican priest *Edward Lowth Badeley, ecclesiastical lawyer *Robert Hugh Benson, son of the Archbishop of Canterbury, novelist and monsignor patristic scholar and Roman Catholic priest Dominican prioress *Frederick William Faber, theologian, hymn writer, Oratorian and Roman Catholic priest *Gerard Manley Hopkins, poet and Jesuit priest *Robert Stephen Hawker, poet and Anglican priest, received Catholicism on his deathbed *James Hope-Scott, barrister and Tractarian, received with Manning *Ronald Knox, Biblical texts translator and formerly an Anglican priest *Henry Edward Manning, later Cardinal Archbishop of Westminster *George Jackson Mivart, biologist, later excommunicated by Cardinal Herbert Vaughan *John Brande Morris, Orientalist, eccentric and Roman Catholic priest *Augustus Pugin, architect *William George Ward, theologian *Benjamin Williams Whitcher, American Episcopal priest.

information. *IBM Journal of research and development*, 1(4):309–317, 1957.

[6] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *the Proceedings of EMNLP 2014*, pages 1532–1543, 2014.

[7] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, November 1997.

[8] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.

[9] Di Wang, Leonid Boytsov, Jun Araki, Alkesh Patel, Jeff Gee, Zhengzhong Liu, Eric Nyberg, and Teruko Mitamura. Cmu multiple-choice question answering system at ntcir-11 qa-lab. In *the Proceedings of NTCIR-11*, 2014.

[10] Hideyuki Shibuki, Kotaro Sakamoto, Yoshinobu Kano, Teruko Mitamura, Madoka Ishioroshi, Kelly Y Itakura, Di Wang, Tatsunori Mori, and Noriko Kando. Overview of the ntcir-11 qa-lab task. In *the Proceedings of NTCIR-11*, 2014.

[11] Rada Mihalcea and Paul Tarau. Textrank: Bringing order into text. In *the Proceddings of EMNLP 2004*, volume 4, pages 404–411, 2004.

[12] Josef Steinberger and Karel Jezek. Using latent semantic analysis in text summarization and summary evaluation. In *the Proceddings of ISIM 2004*, pages 93–100, 2004.

[13] Yihong Gong and Xin Liu. Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of SIGIR '01*, pages 19–25, New York, New York, USA, 2001. ACM Press.

[14] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *the Proceedings of EMNLP 2014*, 2014.

[15] Eunsol Choi, Daniel Hewlett, Jakob Uszkoreit, Illia Polosukhin, Alexandre Lacoste, and Jonathan Berant. Coarse-to-fine question answering for long documents. In *the Proceedings of ACL 2017*, volume 1, pages 209–220, 2017.