LSTM vs. BM25 for Open-domain QA: A Hands-on Comparison of Effectiveness and Efficiency

Sosuke Kato
Waseda University
sow@suou.waseda.jp

Riku Togashi
Yahoo Japan Corporation
rtogashi@yahoo-corp.jp

Hideyuki Maeda
Yahoo Japan Corporation
hidmaeda@yahoo-corp.jp

Sumio Fujita
Yahoo Japan Corporation
sufujita@yahoo-corp.jp

Tetsuya Sakai
Waseda University
tetsuyasakai@acm.org

ABSTRACT

Recent advances in neural networks, along with the growth of rich and diverse community question answering (cQA) data, have enabled researchers to construct robust open-domain question answering (QA) systems. It is often claimed that such state-of-the-art QA systems far outperform traditional IR baselines such as BM25. However, most such studies rely on relatively small data sets, e.g., those extracted from the old TREC QA tracks. Given *massive* training data plus a separate corpus of Q&A pairs as the target knowledge source, how well would such a system *really* perform? How fast would it respond? In this demonstration, we provide the attendees of SIGIR 2017 an opportunity to experience a live comparison of two open-domain QA systems, one based on a long short-term memory (LSTM) architecture with over 11 million Yahoo! Chiebukuro (i.e., Japanese Yahoo! Answers) questions and over 27.4 million answers for training, and the other based on BM25. Both systems use the same Q&A knowledge source for answer retrieval. Our core demonstration system is a pair of Japanese monolingual QA systems, but we leverage machine translation for letting the SIGIR attendees enter English questions and compare the Japanese responses from the two systems after translating them into English.

CCS CONCEPTS

• **Information systems → Question answering**;

KEYWORDS

community question answering; long short-term memory; question answering

SIGIR'17, August 7–11, 2017, Shinjuku, Tokyo, Japan.

ACM ISBN 978-1-4503-5022-8/17/08…$15.00
http://dx.doi.org/10.1145/3077136.3084147

1 INTRODUCTION

Recent advances in neural networks, along with the growth of rich and diverse *community question answering* (cQA) data, have enabled researchers to construct robust open-domain question answering (QA) systems. It is often claimed that such state-of-the-art QA systems far outperform traditional IR baselines such as BM25 [8] (See Section 2). However, most such studies rely on relatively small data sets, e.g., those extracted from the old TREC QA tracks. Given *massive* training data plus a separate corpus of Q&A pairs as the target knowledge source, how well would such a system *really* perform? How fast would it respond?

In this demonstration, we provide the attendees of SIGIR 2017 an opportunity to experience a live comparison of two open-domain QA systems, one based on a *long short-term memory* (LSTM) [10] architecture with over 11 million *Yahoo! Chiebukuro* (i.e., Japanese Yahoo! Answers) questions and over 27.4 million answers for training, and the other based on BM25. Both systems use the same Q&A knowledge source for answer retrieval. Our core demonstration system is a pair of Japanese monolingual QA systems, but we leverage machine translation for letting the SIGIR attendees enter English questions and compare the Japanese responses from the two systems after translating them into English.

Yahoo! Chiebukuro is the major cQA service in Japan, which claimed 714 million pageviews in October 2014. As of December 2014, it had over 84 million questions. While we only have 100,000 questions with 241,994 answers in the target Q&A knowledge source at the time of this writing, we plan to deliver a demonstration with a much larger data set for answer retrieval.

2 RELATED WORK

TREC (Text Retrieval Conference) ran the QA Track from 1999 to 2007; over the years, the task evolved from the relatively simple *factoid QA* to *question series* ending with *others* questions, which meant "*Tell me other interesting things about this target I don't know enough to ask directly*" [2] and therefore solicited complex answers. Other evaluation tasks have also tackled the open-domain QA problem: examples include the QAC (Question Answering Challenge) [3] and the ACLIA (Advanced Crosslingual Information Access) [7] tasks of NTCIR, which were run from 2002 to 2010 when taken together.

While the above efforts in QA aim at automatic extraction of good answers from corpora such as the web and news, leveraging cQA as the knowledge source has also become a promising way to

tackle the QA problem, as both questions and answers in cQA data are diverse and rich in content. In 2012, Liu et al. reported that unsuccessful web searchers often turn to cQA [5]. The NTCIR-8 cQA task [4] tackled the problem of ranking cQA answers given a question, by leveraging the *best answers* data available in the Yahoo! Chiebukuro data. However, the entire Yahoo! Chiebukuro data set from that task contained only about 3.1 million resolved questions from 2004 and 2005, which is substantially smaller compared to the data used in our demonstration.

More recently, the TREC LiveQA track was launched [1], where unresolved cQA questions are sampled in real time and the systems are expected to respond effectively to them. Wang and Nyberg [10], the top performer in that track, first searched the web and Yahoo! Answers to obtain *answer clues*, which are texts similar to the question string, and *answer passages*, which are likely to contain sentences that serve as the answer to the question. Then they ranked candidate sentences based on answer clue scores that leverage BM25 as well as answer passage scores that leverage a multilayer stacked bidirectional LSTM (BLSTM) where the input words are pre-trained by skip-gram-based word embedding [6].

Wang and Nyberg [11] also tackled the problem of *answer selection* for QA using a combination of a BLSTM approach and BM25. Based on an experiment that utilised topics from the QA tracks of TREC-8 through 13, they reported that their combination of BLSTM and BM25 outperformed BM25 as well as other previously reported methods that were evaluated on the same data. However, it remains to be seen whether their conclusion will generalise to more realistic situations where we have millions of questions and answers. Tan *et al.* [9] proposed QA-LSTM, Convolutional-pooling and Convolution-based LSTM, and Attentive LSTM, and reported further improvements on the aforementioned data set.

3 LSTM-BASED QA

This section describes the LSTM-based algorithm that we use for our QA demonstration system. LSTM-based answer selection consists of two phases: training and retrieval. In the training phase, given a question and an answer, the trainer converts each of them into a representation vector as described below and calculates the distance between these vectors. Figure 1 provides an overview of how we calculate the distance between a question and an answer .

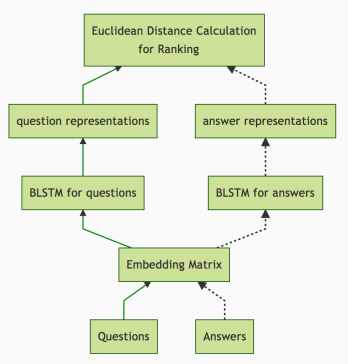

Figure 1: Computation of the distance between questions and answers with LSTM.

Using Mecab[1], a widely-used Japanese morphological analysis tool, we tokenise both the given question and the answer, and then convert them into a word embedding vector pre-trained by a skip-gram method with the Japanese Wikipedia data[2]. Similar to the system of Tan *et al.* [9], our BLSTM takes a sequence of word embedding vectors as input and yields an encoded vector as output. Given a sequence of word embedding vectors $X = x_1, \ldots, x_t, \ldots, x_n$, a hidden vector h_t at time step t is updated as follows:

$$\begin{aligned} i_t &= \sigma(\mathbf{W}_i x_t + \mathbf{U}_i h_{t-1} + \mathbf{b}_i) \\ f_t &= \sigma(\mathbf{W}_f x_t + \mathbf{U}_f h_{t-1} + \mathbf{b}_f) \\ o_t &= \sigma(\mathbf{W}_o x_t + \mathbf{U}_o h_{t-1} + \mathbf{b}_o) \\ j_t &= \tanh(\mathbf{W}_j x_t + \mathbf{U}_j h_{t-1} + \mathbf{b}_j) \\ C_t &= i_t * j_t + f_t * C_{t-1} \\ h_t &= o_t \tanh(C_t), \end{aligned}$$

where an LSTM has three gates, namely input i, forget f and output o, and a cell memory vector C_t. σ is the sigmoid function. $\mathbf{W} \in R^{H \times E}$, $\mathbf{U} \in R^{H \times H}$ and $\mathbf{b} \in R^{H \times 1}$ are the network parameters. E is the dimension of word embedding vectors, and H is that of hidden vectors; we let $E = 600$ and $H = 600$. A BLSTM processes the sequence from forward and backward directions, generating two sequences of output vectors, which are then concatenated. We use 1,200-dimensional output vectors at each time step. Given the concatenated output vectors from BLSTM, max pooling over the concatenated vectors is performed in order to generate fixed-sized distributed vector representations of a sequence of tokens. Given token sequences of a question-answer pair, the question and answer BLSTMs generate a vector representation of the question and the answer, respectively. The distance between the question and the answer is calculated as the euclidean distance between thus generated vectors.

We train the parameters of two BLSTMs and a embedding matrix to minimize triplet loss, which is formulated as follows:

$$L = \max\{0, M + d(q, a_+) - d(q, a_-)\}$$

where d denotes a euclidean distance function and q denotes a question representation vector. a_+ and a_- denote positive and negative example answers, respectively. M denotes the margin of the separation; we let $M = 0.2$.

In the retrieval phase, given a question, we rank answers by euclidean distance by the same mechanism depicted in Figure 1.

4 DEMONSTRATION SYSTEM

4.1 Overview

Figure 2 shows an overview of our demonstration system. We let SIGIR 2017 attendees enter arbitrary questions in either Japanese or English; English questions are automatically translated into Japanese. As can be seen, the Japanese question is fed to the LSTM component and the BM25 component simultaneously, so that retrieved answers obtained with each method can be viewed side by side. The retrieved Japanese answers are also translated into English.

[1] http://taku910.github.io/mecab/
[2] https://dumps.wikimedia.org/jawiki/

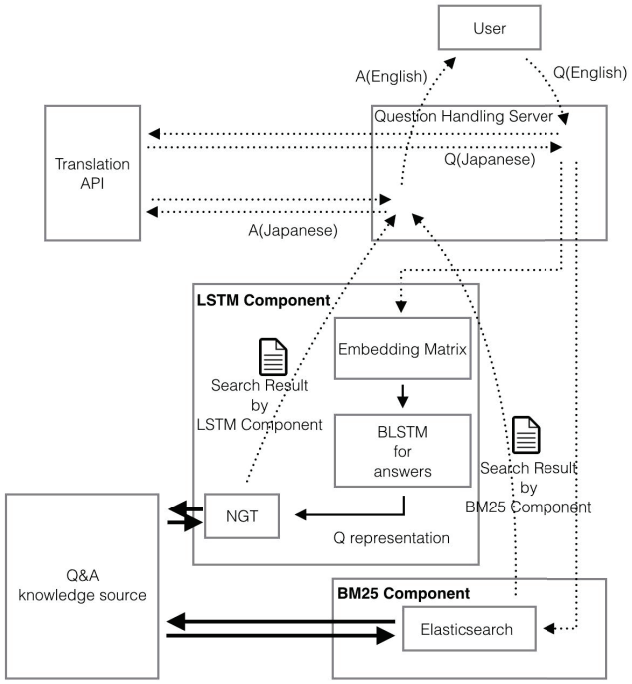

Figure 2: Overview of the demonstration system.

4.2 Yahoo! Chiebukuro Data

Table 1: Number of questions and answers in our test and training dataset.

	(a) training data	(b) Q&A knowledge source
# of questions	11,074,960	100,000
# of answers	27,429,741	241,994

Table 1 shows the statistics of the Yahoo! Chiebukuro data we use. Column (a) shows the number of questions and the number of corresponding answers for training our LSTM component. Column (b) shows the scale of the target knowledge source used by both LSTM and BM25; for the actual demonstration, we plan to expand this data set so that we can cover as diverse questions as possible with the answers that we have for retrieval.

4.3 Ranking Components

Our LSTM component trains the parameters of BLSTM and an embedding matrix using the aforementioned training data, where we use only posts with 50 tokens or less. After the training is done, our LSTM-component indexes answers in the Q&A knowledge source, by converting answer sentences into representation vectors using the BLSTM for answers, where the dimension of representation vectors is 1,200, which are then indexed with NGT[3], a spatial indexing tool. Due to the memory limitation of GPU boards, we only indexed answers shorter than 50 tokens. Table 2 shows the elapsed time of the LSTM component for retrieving the top 10 and 1,000 answers from 155,648 candidates.

Table 2: Elapsed time of the LSTM component to retrieve the top n answers ($n = 10, 1000$).

retrieving top n	$n = 10$	$n = 1000$
average elapsed time [msec]	5.76	41.31

On the other hand, our BM25 component uses Elasticsearch (v5.2.1) [4] with default settings. We indexed all answers in the Q&A knowledge source for retrieval. Search queries are formulated by extracting only nouns, verbs and adjectives from the input question after performing morphological analysis with Mecab.

4.4 Examples

Figure 3 shows an example question from our target Q&A knowledge source, and the answers retrieved by LSTM and BM25, with rough English translations. Note that in the actual demonstration, attendees can enter arbitrary questions, whose topics may be outside the target Q&A knowledge source. How will LSTM and BM25 compare in terms of effectiveness and efficiency?

5 SUMMARY

Given the abundance of data, are neural network-based QA systems going to completely replace traditional IR-based systems in the near future? Please come to the demo, enter a question, compare the answers, and let us know what you think.

REFERENCES

[1] Eugene Agichtein, David Carmel, Dan Pelleg, Yuval Pinter, and Donna Harman. 2016. Overview of the TREC 2015 LiveQA Track. In *Proceedings of TREC 2015.*
[2] Hoa Trang Dang, Diane Kelly, and Jimmy Lin. 2008. Overview of the TREC 2007 Question Answering Track. In *Proceedings of TREC 2007.*
[3] Junichi Fukumoto, Tsuneaki Kato, Fumito Masui, and Tatsunori Mori. 2007. An Overview of the 4th Question Answering Challenge (QAC-4) at NTCIR Workshop 6. In *Proceedings of NTCIR-6.*
[4] Daisuke Ishikawa, Tetsuya Sakai, and Noriko Kando. 2010. Overview of the NTCIR-8 Community QA Pilot Task (Part I): The Test Collection and the Task. In *Proceedings of NTCIR-8.*
[5] Qiaoling Liu, Eugene Agichtein, Gideon Dror, Yoelle Maarek, and Idan Szpektor. 2012. When Web Search Fails, Searchers Become Askers: Understanding the Transition. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval.* 801–810.
[6] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of NIPS 2013.* 3111–3119.
[7] Teruko Mitamura, Hideki Shima, Tetsuya Sakai, Noriko Kando, Tatsunori Mori, Koichi Takeda, Chin-Yew Lin, Ruihua Song, Chuan-Jie Lin, and Cheng-Wei Lee. 2010. Overview of the NTCIR-8 ACLIA Tasks: Advanced Cross-Lingual Information Access. In *Proceedings of NTCIR-3.*
[8] Stephen E. Robertson, Steve Walker, Micheline Hancock-Beaulieu, Mike Gatford, and A. Payne. 1996. Okapi at TREC-3. In *Proceedings of TREC-3.*
[9] Ming Tan, Cícero Nogueira dos Santos, Bing Xiang, and Bowen Zhou. 2016. Improved Representation Learning for Question Answer Matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics.* 464–473.
[10] Di Wang and Eric Nyberg. 2015. CMU OAQA at TREC 2015 LiveQA: Discovering the Right Answer with Clues. In *Proceedings of TREC 2015.*
[11] Di Wang and Eric Nyberg. 2015. A Long Short-Term Memory Model for Answer Sentence Selection in Question Answering. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers).* 707–712.

[3] https://github.com/yahoojapan/NGT

[4] https://www.elastic.co/jp/products/elasticsearch

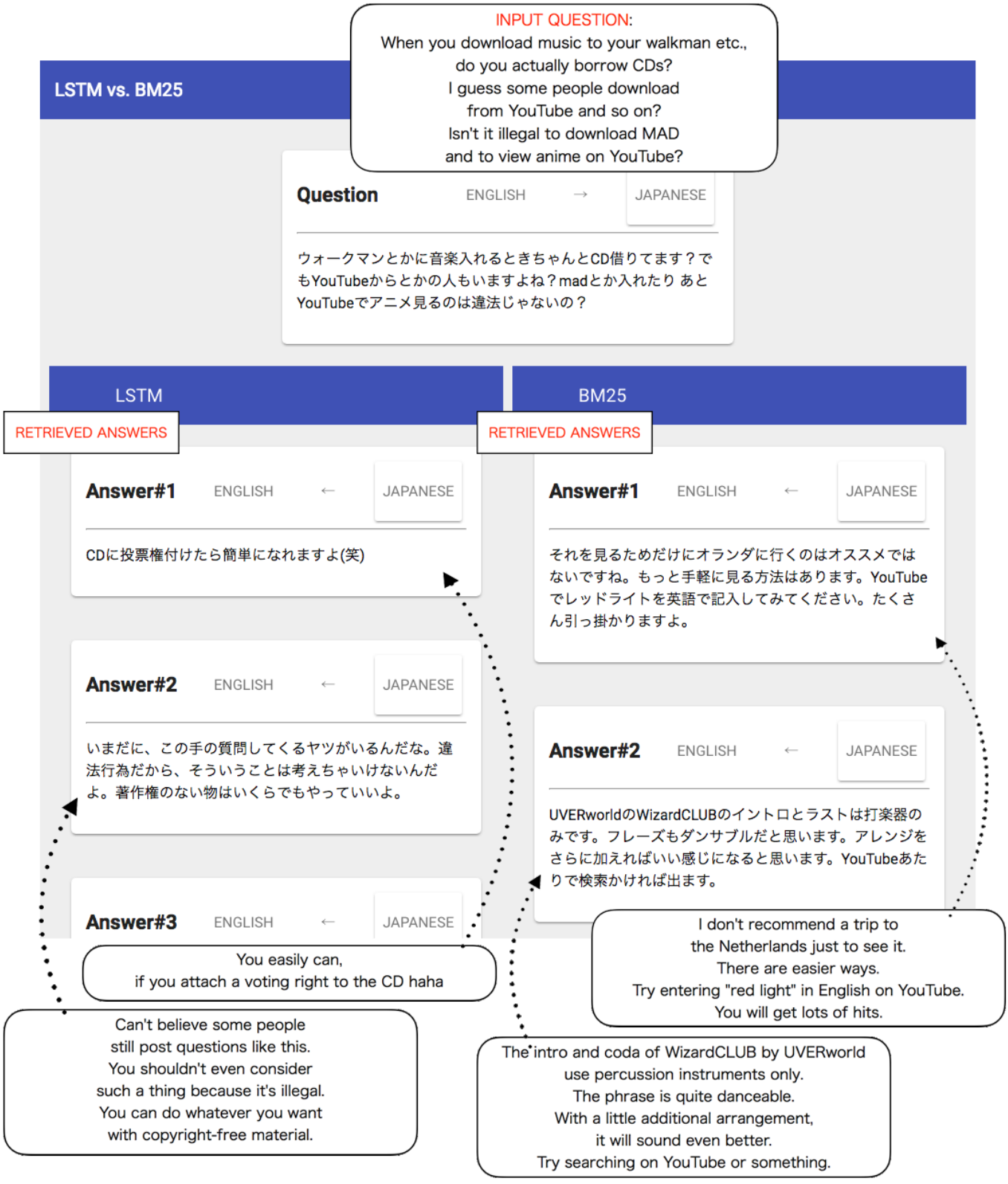

Figure 3: An example question, with answers returned by the LSTM-based system (left) and the BM25-based system (right). Rough English translations are provided by one of the authors; in the actual demonstrations, machine translation will be used.