Developing a How-to Tip Machine Comprehension Dataset and its Evaluation in Machine Comprehension by BERT

Tengyang Chen[†], **Hongyu Li**[†], **Miho Kasamatsu**[†],
Takehito Utsuro[†], **Yasuhide Kawada**[‡]
[†]Graduate School of Systems and Information Engineering, University of Tsukuba, Japan
[‡]Logworks Co., Ltd., Japan

Abstract

In the field of factoid question answering (QA), it is known that the state-of-the-art technology has achieved an accuracy comparable to that of humans in a certain benchmark challenge. On the other hand, in the area of non-factoid QA, there is still a limited number of datasets for training QA models, i.e., machine comprehension models. Considering such a situation within the field of the non-factoid QA, this paper aims to develop a dataset for training Japanese how-to tip QA models. This paper applies one of the state-of-the-art machine comprehension models to the Japanese how-to tip QA dataset. The trained how-to tip QA model is also compared with a factoid QA model trained with a Japanese factoid QA dataset. Evaluation results revealed that the how-to tip machine comprehension performance was almost comparative with that of the factoid machine comprehension even with the training data size reduced to around 4% of the factoid machine comprehension. Thus, the how-to tip machine comprehension task requires much less training data compared with the factoid machine comprehension task.

1 Introduction

Recent advances in the field of QA or machine comprehension are mostly in the domain of factoid QA related to Wikipedia articles and news articles (Yi et al., 2015; Pranav et al., 2016, 2018). One of the most well-known QA datasets and benchmark tests is the Stanford Question Answering Dataset (SQuAD) (Pranav et al., 2016, 2018), which is a reading comprehension dataset, consisting of questions posed by crowdworkers on a set of Wikipedia articles, where the answer to every question is a text segment, or span, from the corresponding reading passage, or the question might be unanswerable. It is reported[1] that state-of-the-art machine comprehension models trained with SQuAD outperform humans (Devlin et al., 2019; Zhang et al., 2019).

However, apart from the issues related to developing benchmark datasets for factoid QA and improving state-of-the-art general-purpose machine comprehension models, there is a relatively limited number of published literature that handles issues, such as the development of datasets for non-factoid QA and the application of state-of-the-art general-purpose machine comprehension models to those non-factoid datasets. Typical non-factoid QA tasks include opinion QA, definition QA, reason QA, and how-to tip QA.

Among various kinds of non-factoid knowledge which are the key to developing techniques for non-factoid QA tasks, a recent study (Ohkawa et al., 2018) examined the types of Japanese websites which include various how-to tips related to job hunting, marriage, and apartment. The study (Ohkawa et al., 2018) also aims to automatically identify those how-to tip websites, which will be an important knowledge source for training how-to tip QA models. Considering such a situation, within the field of non-factoid QA, this paper studies how to develop a dataset for training Japanese how-to tip (hereafter throughout the paper, we use the simplified term "tip") QA models. As examples in this paper, we developed tip QA datasets for 'job hunting," "marriage," "apartment," "hay fever," "dentist," and "food poisoning," where "job hunting" and "marriage" tip QAs are for both training and testing, while other tip QAs are only for testing. For "job hunting", Figure 1 presents a typical example of a tuple of a context, a tip question, and an answer. Furthermore, in order to understand rough idea of

[1] `https://rajpurkar.github.io/SQuAD-explorer/`

Proceedings of the Third Workshop on Fact Extraction and VERification (FEVER), pages 26–35
July 9, 2020. ©2020 Association for Computational Linguistics

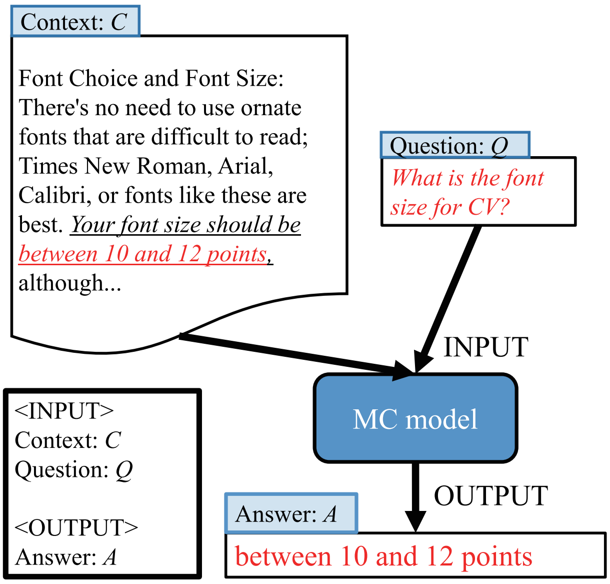
Context: C
Font Choice and Font Size: There's no need to use ornate fonts that are difficult to read; Times New Roman, Arial, Calibri, or fonts like these are best. Your font size should be between 10 and 12 points, although...
Question: Q
What is the font size for CV?
INPUT
MC model
OUTPUT
<INPUT>
Context: C
Question: Q
<OUTPUT>
Answer: A
Answer: A
between 10 and 12 points

Figure 1: An example of the machine comprehension model of tip QA for "job hunting" together with an example of a tuple of a context C, a question Q, and answer A (extracted from a column web page entitled "Formatting Tips for Your Curriculum Vitae (CV)" (https://www.thebalancecareers.com/curriculum-vitae-format-2060351) from a tip website titled "The Balance Careers" (https://www.thebalancecareers.com/))

the (how-to) tip questions we study in this paper in the broader sense, we manually classify them into five types as shown in Table 1 and also shown several examples for each of the five types and their statistics within the dataset we developed in this paper.

This paper then applies BERT (Devlin et al., 2019), one of the state-of-the-art machine comprehension models, to a Japanese tip QA dataset. The trained tip QA model is also compared with a factoid QA model which is also trained with a Japanese factoid QA dataset. Evaluation results revealed that the tip machine comprehension performance was almost comparative with that of the factoid machine comprehension even with the training data size reduced to around 4% of the factoid machine comprehension. Thus, the tip machine comprehension task requires much less training data compared with the factoid machine comprehension task.

2 Query Focuses and Collecting Web Pages

This section briefly describes the workflow of collecting web pages. First, the notion of *query focus* is a keyword used for every search request related to a specific subject. For example, whenever the aim was to collect web pages about anything related to job hunting, the word "job hunting" was always put at the beginning of the query, and all available suggested keywords provided by the search engine were collected, such as "job hunting self-promotion" and "job hunting portfolio." Using all such suggested keywords as queries (called *search engine suggests* or *suggests*), the search engine is crawled, and top 10 results for each suggest are collected.

2.1 Collecting Search Engine Suggests

Web search engine suggests are the query keywords automatically offered by a search engine when a user types part of a search query. Such suggested keywords can be seen as frequent user activities logged by the search engine, and they mostly lead to pages on trending topics. For a given query focus keyword, about 100 specified types of Japanese hiragana characters were entered into Google ® search engine from which up to 1,000 suggests were collected. These 100 types of Japanese hiragana characters include the Japanese alphabet consisting of 50 characters, voiced and semi-voiced variants of voiceless characters and Youon.

tip question type	examples	rate (%)
essential, words of caution, reminder	what is the essential of ∼ ? / what are the words of caution on ∼ ? / what should one take care of when ∼ ?	23.6
characteristics, definition, knowledge, fact, rule	what is the characteristics of ∼ ? / what is ∼ ? / which documents are required to submit to the city hall when ∼ ?	18.5
method and how-to tip (in the narrower sense)	how can I do ∼ ? / what is the tip, know-how, hack for doing ∼ ?	16.6
reason, cause, background, purpose	what is the reason for ∼ ? / why ∼ ? / what is the purpose of ∼ ?	4.3
habit, experience, recommendation (tip of any type other than the above four types)	what is the recommendation when ∼ ? / what should I use when ∼ ? / when should I start ∼ ?	37.0
total	—	100

Table 1: Statistics of the Classification of Tip Question Types

2.2 Collecting Web Pages

Google Custom Search API[2] was used to scrape web pages from the search engine. Using the web search engine suggests collected in the previous section combined with the query focus keyword as queries (in the form of AND search), the first 10 pages returned per search query are collected. The set of web pages queried by suggest s can be represented as $D(s, N)$, where N is 10 as a constant standing for top N pages. Additionally, the search engine suggests were saved for every web page. Since different search engine suggests could lead to the same web page, one web page could have multiple suggests. Let S be the set of all suggests about one query focus. Then, the set of web pages scraped using all possible suggests is represented as D.

$$D \;=\; \bigcup_{s \in S} D(s, N)$$

3 Selecting Candidates of Tip Websites

This paper employs LDA (latent Dirichlet allocation) (Blei et al., 2003) to model topic distributions among documents. Let D be a document set containing all collected web pages and K ($= 50$ in this paper) be the number of topics. When the topic model is applied, topic distribution $P(z_n \,|\, d)$ is available for every d ($d \in D$). Every document d is assigned a topic with the highest probability among all its $P(z_n \mid d)$. The net effect is that for every topic z_n, there is a group $D(z_n)$ ($n = 1, \ldots, K$) of corresponding documents that are assigned to z_n.

Then, domain names are extracted from all collected web pages based on their URLs. The domain names that have corresponding web pages reside in 10 or more sets $D(z_n)$ ($n = 1, \ldots, K$), i.e., they have their web pages under more than or equal to 10 topics which are considered as candidates for tip websites[3]. Out of those candidates whose numbers are 31 for job-hunting in this experiment, 14 domain names were randomly selected, for all of which tip QAs were successfully collected. Henceforth, the set of those 14 tips websites will be denoted as T. Similarly, for marriage, 13 domain names have their web pages under more than or equal to 10 topics and are considered as candidates for tip websites. For all of those 13 domain names, tip QAs were successfully collected. Thus, for marriage, the set of those 13 tips websites will be denoted as T.

4 Collecting Tip QAs

4.1 Collecting Web Pages of Tip Websites

From each website out of the set T of tip websites, web pages are collected as the source for collecting tip QAs. First, from each website of T, all of its web pages are collected into set $D^{\mathrm{inf}}(T)$. Then, the LDA topic model (Blei et al., 2003) $P(z_n \mid d)$ (available for every d ($d \in D$)) trained in Sec-

[2] `https://developers.google.com/custom-search/`

[3] Ohkawa et al. (2018) examined quantitative characteristics of tip websites. Furthermore, it is reported in Ohkawa et al. (2018) that domain names of candidate tip websites can be collected from those that have corresponding web pages in sets $D(z_n)$ of web pages for multiple topics z_n ($n = 1, \ldots, K$). It is observed, however, that typical tip websites actually have their web pages under far more than two topics and typically, more than or equal to 10 topics. Thus, in this paper, it was decided to select domain names which have their web pages under more than or equal to 10 topics as candidates for tip websites.

# Pairs of question and answer	1	2	3	4	5	Total
# Web pages (%)	131 (28.1)	102 (21.9)	64 (13.7)	33 (7.1)	136 (29.2)	466 (100)

Table 2: # Pairs of QA collected from a web page (for "job hunting" and "marriage")

tion 3 with the set D of web pages scraped using all possible suggests is applied to each web page d within set $D^{\mathrm{inf}}(T)$. According to the probability distribution $P(z_n|d)$ of topics z_n $(n = 1, \ldots, K)$ for each web page d, the topic z_n with the highest probability is assigned to d. Then, the set of web pages to which the topic z_n is assigned is denoted as $D^{\mathrm{inf}}(z_n, T)$:

$$D^{\mathrm{inf}}(z_n, T) = \Big\{ d \in D^{\mathrm{inf}}(T) \,\Big|\, z_n = \operatorname*{argmax}_{z_u\ (u=1,\ldots,K)} P(z_u|d) \Big\}$$

For the query focus "job hunting," out of the total $K = 50$ topics, $|D^{\mathrm{inf}}(z_n, T)| > 0$ holds, i.e., at least one web page is assigned to 42 topics for job hunting and 29 for marriage.

4.2 "Column Pages" as the Source for Collecting Tip QAs

This study analyzes the types of web pages which tend to include more and more tips compared with other types of web pages. This paper examines tip websites which include column pages containing various tips and also include other types of web pages, such as pages for commercial sale of products or pages with reviews and experiences. However, most tips are found only in column pages but not in other types of pages. The type of web pages which tend to include tips are mostly column pages.

Out of the set $D^{\mathrm{inf}}(z_n, T)$ of web pages defined in the previous section, all the column pages are extracted into a subset:

$$D^{\mathrm{inf}}_c(z_n, T)$$

In the case of the query focus "job hunting," out of 42 topics satisfying $|D^{\mathrm{inf}}(z_n, T)| > 0$, 36 topics satisfy $|D^{\mathrm{inf}}_c(z_n, T)| > 0$, i.e., include column pages. For "marriage", all the 29 topics satisfy $|D^{\mathrm{inf}}_c(z_n, T)| > 0$. For each topic z_n, all the web pages in this set are used as a source for collecting tip QAs.

4.3 Procedure for Collecting Tip QAs

This section describes the procedure for collecting tip questions and examples, such as those presented in Figure 1. From each web page within the set $D^{\mathrm{inf}}_c(z_n, T)$ constructed in the previous section, tuples of context C, question Q, and answer A are manually collected. Specifically, within each column web page, every paragraph is examined, and it is decided whether a pair of a question and an answer can be collected from the paragraph. From each column web page, at most 5 pairs of a question and an answer are collected. Figure 1 presents an example of collecting a tuple of a context, a question, and an answer from a column web page of a "job hunting" tip website. In this example, context C, the following paragraph about font choice and font size is selected:

> There's no need to use ornate fonts that are difficult to read; . . . Your font size should be between 10 and 12 points, although . . .

From this paragraph, a pair of question Q "What is the font size for CV?" and answer A "between 10 and 12 points" is extracted. Table 2 lists the distribution of the number of the pairs of a question and an answer collected from a web page for "job hunting" and "marriage".

For the query focus "job hunting," out of the overall 1,268 column web pages collected following the procedure of this paper, 352 pages were actually examined, out of which 907 pairs of tip QAs are collected. For the query focus "marriage," out of the overall 3,075 column web pages collected following the procedure of this paper, 114 pages were actually examined, out of which 432 pairs of tip QAs are collected. For "apartment" query focuses, 50 pairs of tip QAs are collected. For other query focuses "hay fever," "dentist," and "food poisoning," a total of 50 pairs of tip QAs are collected. Table 3 presents an example of Japanese tip QAs for each of "job hunting," "marriage," and "hay fever." These numbers and examples are all for SQuAD1.1 type answerable questions only.

Context C	Question Q	Answer A
履歴書に短所を書く時は前向きにまとめるようにします.「工夫して克服した」「直すように努力している」などと書けば悪いイメージの短所で好印象を与えることも可能です. 自分の短所の中で努力すれば改善しそうなものを選ぶと書きやすいでしょう.	履歴書に短所を書く時のポイントは? (What is the tip when including one's weak points into one's resume?)	前向きにまとめる (Organize them positively.)
一年の中でも結婚式の費用を抑えやすく比較的安い月といえるのが 1 月・2 月. 寒さが厳しいシーズンであるため，結婚式の施行数もそれほど多くなく，通常よりも割安なプランを用意している会場が多数あります. また，希望の日程で日取りを抑えやすいのも魅力. キャンドルの炎を使ったやさしい光のライトアップやキラキラと輝く装飾など，冬らしいコーディネートを取り入れるのもオススメです.	一年の中でも結婚式の費用を抑えやすく比較的安い月は? (In which month, is it the easiest to save money for a wedding?)	1 月・2 月 (January and February)
そのため花粉の季節は，室内の湿度を 50∼55%ほどに保てるように加湿器を使用しましょう.	花粉の季節に保つべき室内の湿度の目安は? (How much indoor humidity should be maintained in the pollen season?)	50∼55% (50∼55%)

Table 3: Examples of Japanese tip QAs selected from training and test datasets used in evaluation (tuples of Context C, Question Q, and Answer A for query focuses "job hunting," "marriage," and "hay fever", for SQuAD1.1: answerable questions)

From these tip QAs of SQuAD1.1 type with answerable questions, tip QAs of SQuAD2.0 type with unanswerable questions are manually created. From a tuple of a context C, a question Q, and an answer A of SQuAD1.1 type, which is answerable in that the context C includes the answer A to the question Q, the annotator manually created another tuple, which is an unanswerable QA, of a context C' ($\neq C$), a question Q' ($= Q$), and the answer $A' = \langle\text{null}\rangle$. Here, within exactly the same column web page of the tip website, from which the context C is extracted, the annotator searched for another paragraph other than C, which does not include any answer to the original question Q. The selected paragraph C' constitutes the context of a tip QA of SQuAD2.0 type with an unanswerable question. Note that it is quite important to search for C' within exactly the same column web page of the tip website, from which the context C is extracted. For example, in the case of the tip QA on "job hunting" in Figure 1, for the question Q "What is the font size for CV?", within the same column web page about "job hunting", another paragraph C' other than C is selected. The selected paragraph C' still presents a certain tip about job hunting and CV, while it does not include any tip about the font size for CV. We follow this strategy simply because it avoids tip QAs with unanswerable questions becoming much easier to answer compared with tip QAs with answerable questions. With this strategy, for each of almost all the tip QAs of SQuAD1.1 type answerable questions, we successfully created at least one tip QA of SQuAD2.0 type with an unanswerable question.

5 Applying BERT to Tip Machine Comprehension

5.1 Dataset for Evaluation

In this paper, we developed two types of datasets for evaluation: one for SQuAD1.1 type answerable questions only and another for SQuAD2.0 type answerable and unanswerable questions. This paper presents evaluation results with the SQuAD2.0 type dataset. For the SQuAD2.0 type dataset, Table 4 presents the statistics of training and test datasets for evaluation in this paper. Table 4 (a) presents those of the training and test datasets for Japanese factoid QAs[4]. Those Japanese factoid QAs, which are of SQuAD2.0 type, are manually collected from Japanese quiz data by automatically identifying context texts from Japanese version of Wikipedia and then manually judging whether each identified context includes the answer to the question. Table 4 (b) and Table 4 (c) present the statistics of training and test datasets for Japanese tip QAs about "job hunting" and "marriage". Similarly, Table 4 (d) presents those for test datasets for Japanese tip QAs about "apartment," "hay fever," "dentist," and "food poisoning"[5].

[4] `http://www.cl.ecei.tohoku.ac.jp/rcqa/` (in Japanese)

[5] Four annotators participated in the procedure of collecting tip QAs, where we measured inter-annotator agreement rate according to AC$_1$ (Gwet, 2008), but not to kappa (Cohen, 1960), mainly because two or more annotators tend to have high overall agreement rate, causing imbalanced class label distribution and instability of kappa. AC$_1$ inter-annotator agreement is measured through the two sub-procedures: i.e., i) manually judging whether the questions selected by two out of three annotators are semantically equivalent when exactly the same context paragraph is given to the three anno-

(a) Factoid QAs

	# tuples of a context, a question and an answer (answerable / unanswerable)	average # words within a context	average # words of a question
Train.	27,645/28,906	88.2	26.1
Test	49/51	82.8	27.1

(b) Tip QAs: "job hunting"

	# tuples of a context, a question and an answer (answerable / unanswerable)	average # words within a context	average # words of a question
Train.	755/845	63.0	10.7
Test	50/54	71.3	9.7

(c) Tip QAs: "marriage"

	# tuples of a context, a question and an answer (answerable / unanswerable)	average # words within a context	average # words of a question
Train.	382/382	44.2	11.2
Test	50/48	68.7	10.2

(d) Tip QAs: "apartment," "hay fever," "dentist," and "food poisoning"

query focus	# tuples of a context, a question and an answer (answerable / unanswerable)	average # words within a context	average # words of a question
apartment	50/49	82.0	10.3
hay fever, dentist, food poisoning	50/43	71.0	9.5

Table 4: Statistics of training and test datasets

5.2 BERT Implementation

As the version of BERT (Devlin et al., 2019) implementation which can handle a text in Japanese, the TensorFlow version[6] and the Multilingual Cased model[7] were used as the pre-trained model. Before applying BERT modules, MeCab[8] was applied with IPAdic dictionary, and the Japanese text was segmented into a morpheme sequence. Then, within the BERT fine-tuning module, the WordPiece module with 110k shared WordPiece vocabulary was applied, and the Japanese text was further segmented into a subword unit sequence. Finally, the BERT fine-tuning module for machine comprehension[9] was applied as well as the fine-tuned model. The BERT pre-trained model was fine-tuned with the following three types of train-

tators, and ii) manually judging whether the answers selected by two out of three annotators are semantically equivalent when exactly the same pair of a question and a context is given to the three annotators (their detailed procedures are omitted for space restriction). Average of AC_1 is 0.61 for the sub-procedure i) and 0.92 for the sub-procedure ii), which are sufficiently high.

[6] https://github.com/google-research/bert

[7] Trained with 104 languages, available from https://github.com/google-research/bert/blob/master/multilingual.md.

[8] http://taku910.github.io/mecab/ (in Japanese)

[9] run_squad.py, with the number of epochs as 2, batch size as 8, and learning rate as 0.00003.

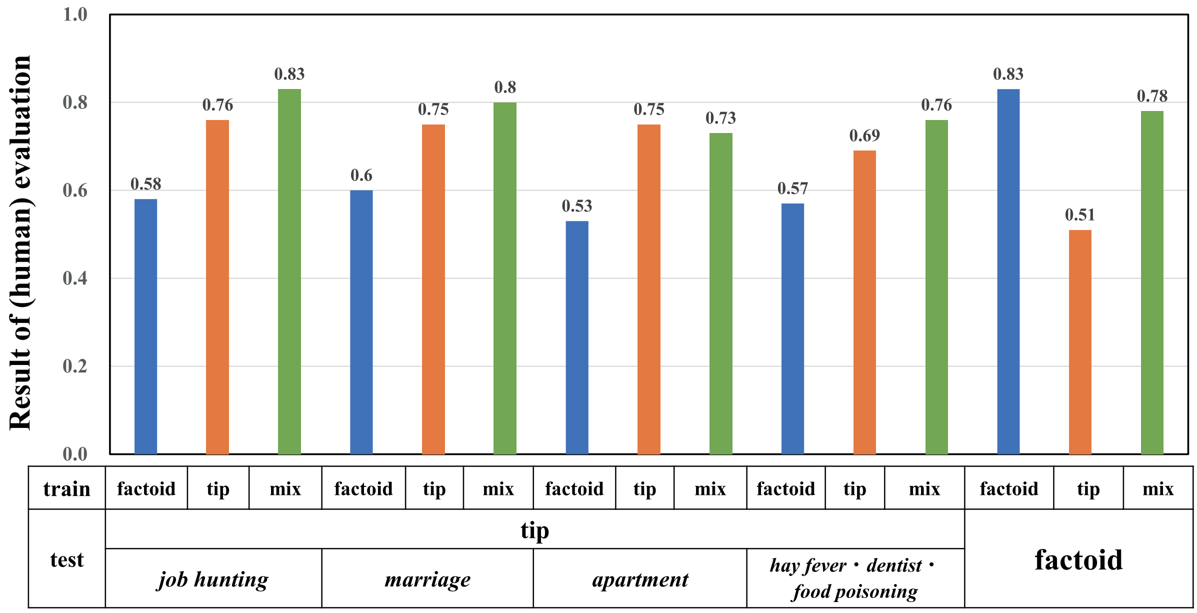

Figure 2: Evaluation results (exact match + partial match)

ing datasets:

(i) The training dataset of factoid QAs in Table 4 (a).

(ii) The training datasets of the tip QA about "job hunting" in Table 4 (b) and "marriage" in Table 4 (c).

(iii) Mix of (i) and (ii).

Here, note that we train a single model with each of these three training datasets (i)∼(iii), i.e., a single factoid machine comprehension model with (i), a single tip machine comprehension model with (ii), and a single machine comprehension model for the mixture of factoid and tip with (iii). It is especially important to note that we train a single tip machine comprehension model with the tip QA datasets about "job hunting" and "marriage", then evaluate it against the tip QA test datasets about all the query focuses, i.e., 'job hunting," "marriage," "apartment," "hay fever," "dentist," and "food poisoning."

5.3 Evaluation Result

In the evaluation, it is manually judged whether the answer predicted by the fine-tuned model and the reference answer partially match or not. We prefer manual evaluation rather than automatic evaluation, mainly because we prefer the quality of evaluation than avoiding the cost of evaluation. Figure 2 presents the evaluation results for the tip QA test datasets about "job hunting," "marriage," "apartment," and a mix of "hay fever," "dentist," and "food poisoning," as well as for the factoid QA test dataset. As clearly seen from these results, for all the tips test datasets, (ii) training only with tip QAs and (iii) training with a mix of tip QA and factoid QA training datasets outperforms and (i) training only with factoid QAs. For the factoid QA test datasets, on the other hand, (i) training only with factoid QAs and (iii) training with a mix of tip QA and factoid QA training datasets outperforms (ii) training only with tip QAs. This result supports the conclusion that the tip machine comprehension task is essentially different from the factoid machine comprehension task. But, still, for tips on "job hunting," "marriage," and the mix of "hay fever," "dentist," and "food poisoning," training with a mix of tip QA and factoid QA training datasets slightly outperforms training only with tip QAs. This result indicates that the tip machine comprehension task still to some extent benefits from a large-scale factoid QA training dataset when only small-scale tip QAs are available.

Another interesting finding is that, in tip machine comprehension, the single model fine-tuned with tip QA training datasets on "job hunting" and "marriage" performed well in tip machine comprehension of other query focuses, such as "apartment," "hay fever," "dentist," and "food poisoning." Thus, in tip comprehension, it is sufficient to collect tip QA only for one or two query fo-

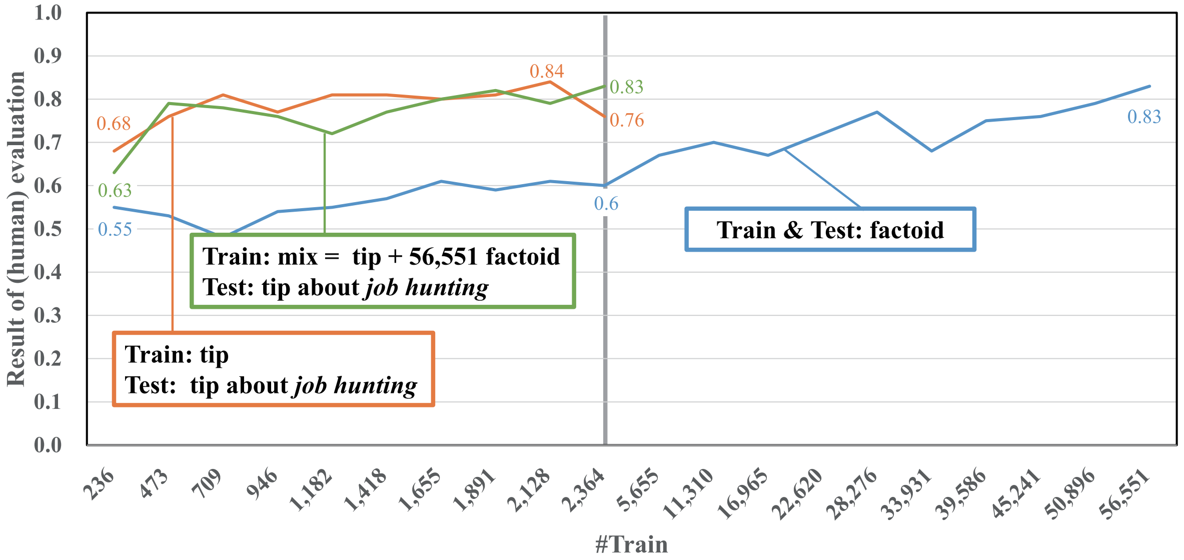

Figure 3: Comparing learning curves of factoid QAs, tip QAs, and training with a mix of factoid and tip QAs (exact match + partial match)

cuses, such as "job hunting" and "marriage," and then to fine-tune the tip machine comprehension model which is applicable to tip machine comprehension on any query focus.

Figure 3 also presents a comparison of learning curves for the following three cases:

(a) Training with 5%, 10%, . . ., 95%, and 100% of factoid QA training dataset of (i) in the previous section and testing with the factoid QA test dataset from Table 4 (a) (plotted in blue).

(b) Training with 10%, 20%, . . ., 90%, and 100% of the tip QA training datasets on "job hunting" and "marriage" of (ii) in the previous section and testing with the tip QA test dataset on "job hunting" of Table 4 (b) (plotted in orange).

(c) Training with a mix of (a) and (b), where the factoid QA training dataset of (a) is always with its 100% size, whereas the tip QA training dataset on "job hunting" of (b) has a size of 10%, 20%, . . ., 90%, and 100% sizes and testing with the tip QA test dataset on "job hunting" of Table 4 (b) (plotted in green).

As can be seen from these results, the learning curve (b) of tip QAs and that (c) of the mix of factoid and tip QAs perform comparatively well and outperform that of factoid QAs (a) in the range of around a few thousand training data size. This result indicates that, at least for tip machine comprehension of "job hunting", benefit from a large-scale factoid QA training dataset is very little. Far more important finding in this result is that the tip machine comprehension performance is almost comparative with that of the factoid machine comprehension even when trained with as little as around 4% (≒ 2,364/56,551) of the training data size of the factoid machine comprehension. Thus, it can be concluded that the tip machine comprehension task requires much less training data compared with the factoid machine comprehension task.

6 Related Work

In the field of developing QA datasets or machine comprehension datasets which may include non-factoid QAs, quite a limited number of datasets are publicly available in any language. In English, MS MARCO (Nguyen et al., 2016) has been developed using Bing's search logs and passages of retrieved web pages, which may include non-factoid QAs. Question types in MS MARCO are classified into *numeric, entity, location, person*, and *description (phrase)*. In Chinese, DuReader (He et al., 2018) has been developed using Baidu Search and Baidu Zhidao, which is a Chinese community-based QA site. DuReader's question types are classified into *entity, description*, and *yes-no* questions on *fact* or *opinion*. DuReader's QAs definitely include non-factoid ones. Another type of non-factoid QA dataset is NarrativeQA (Kočiský et al., 2018) dataset (in English), which contains questions created by editors based on summaries of movie scripts and books. In the case of the Japanese language QA dataset, there is quite a limited number of publicly available factoid QA datasets, and one of them was introduced in Section 5.1. There is no publicly available Japanese non-factoid QA dataset.

7 Conclusion

This paper explored a way to develop a dataset for training Japanese tip QA models, and it applied BERT (Devlin et al., 2019) to a Japanese tip QA dataset. Evaluation results revealed that the tip machine comprehension performance was almost comparative with that of the factoid machine comprehension even with the training data size reduced to around 4% of the factoid machine comprehension. Thus, the tip machine comprehension task requires much less training data compared with the factoid machine comprehension task.

Future direction of this work includes applying the proposed framework of tip machine comprehension to other languages, such as English and Chinese. In both languages, factoid QA datasets are publicly available (e.g., SQuAD (Pranav et al., 2016, 2018) for English and CMRC2018 (Cui et al., 2018) for Chinese), and it is quite attainable to train a factoid machine comprehension model by fine-tuning the BERT pre-trained model and then to directly apply the factoid machine comprehension model to the tip machine comprehension task. Actually, as a preliminary work, a Chinese factoid machine comprehension model is trained by fine-tuning the pre-trained Multilingual Cased model with CMRC2018 Chinese factoid QA dataset[10][11], and then applying it to 30 Chinese tip questions on “marriage” with context texts. As a result, around 50% accuracy for manual evaluation is achieved by exact and partial match, which is almost comparative to the performance achieved in the Japanese tip machine comprehension task reported in this paper. Thus, it is expected that extending the proposed framework of tip machine comprehension to other languages, such as English and Chinese, is quite straightforward.

Another future direction is to extending the proposed framework of tip machine comprehension to open domain tip machine comprehension. This extension is similar to the extension of existing factoid machine comprehension with Wikipedia texts' paragraphs as contexts to open domain machine comprehension with the whole Wikipedia articles (Chen et al., 2017). In the extended open domain tip machine comprehension framework, the document retriever module is realized based on the tip websites search and column web page collection architectures proposed in this paper. The document reader module can be easily realized by simply applying the tip machine comprehension model of this paper.

Another definitely important future direction should be to invent a technique of how to automate the procedure of collecting column web pages and generating the tuple of a context C, a question Q, and answer A. This task can be regarded as that of training a tip machine comprehension model from a noisy training dataset.

[10] https://hfl-rc.github.io/cmrc2018/english/

[11] https://github.com/ymcui/cmrc2018

References

D. M. Blei, A. Y. Ng, and M. I. Jordan. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

D. Chen, A. Fisch, J. Weston, and A. Bordes. 2017. Reading Wikipedia to answer open-domain questions. In *Proc. 55th ACL*, pages 1870–1879.

J. Cohen. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46.

Y. Cui, T. Liu, L. Xiao, Z. Chen, W. Ma, W. Che, S. Wang, and G. Hu. 2018. A span-extraction dataset for Chinese machine reading comprehension. *CoRR*, abs/1810.07366.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. NAACL-HLT*, pages 4171–4186.

K. Gwet. 2008. Computing inter-rater reliability and its variance in the presence of high agreement. *The British Journal of Mathematical and Statistical Psychology*, 61:29–48.

W. He, K. Liu, J. Liu, Y. Lyu, S. Zhao, X. Xiao, Y. Liu, Y. Wang, H. Wu, Q. She, X. Liu, T. Wu, and H. Wang. 2018. DuReader: a Chinese machine reading comprehension dataset from real-world applications. In *Proc. MRQA*, pages 37–46.

T. Kočiský, J. Schwarz, P. Blunsom, C. Dyer, K. M. Hermann, G. Melis, and E. Grefenstette. 2018. The NarrativeQA reading comprehension challenge.

Transactions of the Association for Computational Linguistics, 6:317–328.

T. Nguyen, M. Rosenberg, X. Song, J. Gao, S. Tiwary, R. Majumder, and L. Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. *CoRR*, abs/1611.09268.

Y. Ohkawa, S. Kawabata, C. Zhao, W. Niu, Y. Lin, T. Utsuro, and Y. Kawada. 2018. Identifying tips Web sites of a specific query based on search engine suggests and the topic distribution. In *Proc. 3rd ABCSS*, pages 4347–4353.

R. Pranav, Z. Jian, L. Konstantin, and L. Percy. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proc. EMNLP*, pages 2383–2392.

R. Pranav, J. Robin, and L. Percy. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proc. 56th ACL*, pages 784–789.

Y. Yi, Y. Wen-tau, and M. Christopher. 2015. WikiQA: A challenge dataset for open-domain question answering. In *Proc. EMNLP*, pages 2013–2018.

Z. Zhang, Y. Wu, J. Zhou, S. Duan, and H. Zhao. 2019. SG-Net: Syntax-guided machine reading comprehension. *CoRR*, abs/1908.05147.