Exam Keeper: Detecting Questions with Easy-to-Find Answers

Ting-Lun Hsu*
Institute of Information Science, Academia Sinica
Taipei, Taiwan
hsutingl@iis.sinica.edu.tw

Shih-Chieh Dai[1]
Institute of Information Science, Academia Sinica
Taipei, Taiwan
sjdai@iis.sinica.edu.tw

Lun-Wei Ku
Institute of Information Science, Academia Sinica
Taipei, Taiwan
lwku@iis.sinica.edu.tw

ABSTRACT

We present Exam Keeper, a tool to measure the availability of answers to exam questions for ESL students. Exam Keeper targets two major sources of answers: the web, and apps. ESL teachers can use it to estimate which questions are easily answered by information on the web or by using automatic question answering systems, which should help teachers avoid such questions on their exams or homework to prevent students from misusing technology. The demo video is available at https://youtu.be/rgq0UXOkb8o [1]

CCS CONCEPTS

• **Information systems → Decision support systems**; **Data analytics**.

KEYWORDS

Question Answering, Information Retrieval, Web Application

ACM Reference Format:
Ting-Lun Hsu, Shih-Chieh Dai[1], and Lun-Wei Ku. 2019. Exam Keeper: Detecting Questions with Easy-to-Find Answers. In *Proceedings of the 2019 World Wide Web Conference (WWW '19), May 13–17, 2019, San Francisco, CA, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3308558.3314130

1 INTRODUCTION

The goal of the teacher is not always the same as that of the student, especially for short term goals. Although teachers always hope that their students really understand the knowledge they are imparting, students sometimes just want to complete their assignments with as little effort as possible. This, in the day when students can find answers with a quick web search, or use problem-solving, educational apps to find the answers. In addition, with the recent wide success of artificial intelligence, many worry that machines will soon be able to replace them in their jobs. This highlights the importance of practicing on problems that machines cannot easily solve.

However, from our observation, the advantage in this competition between teacher and student is with the latter. Many models and tools have been created to assist students to find answers, but as far as we know, there is no tool for teachers to monitor this process. Hence, we present Exam Keeper, which utilizes both retrieval-based and learning-based question answering techniques to provide a reference for teachers to know which questions have answers that are easily found on the web or using apps. Teachers can use this information to design their tests or homework problem sets accordingly to reduce the incidence of students copying answers, and hence to achieve the goal of effective practice and evaluation.

2 RELATED WORK

In the literature, there are several mobile applications which students can use to get answers, including *yuansouti*[2] and *afanti*[3]. The motivation behind such tools is to help students find answers when they are stuck on their homework assignments. However, many default to using these tools as soon as they see the questions, without even trying to solve the questions by themselves. Thus, these "time-saving" apps have become a nightmare for parents and teachers.

Researchers have also developed exam robots for high-school level exams such as AI-MATHS[4] for math exams in China and Todai[5] for the University of Tokyo entrance exam. Though these systems show the current level of AI systems for exams, they are neither open to the public nor used for educational purposes.

Previous systems for answering English exams have used a variety of different models to answer different types of questions. Examples include a state-of-the-art language model (LM)[6] trained on large datasets such as the One Billion Word Corpus [1] to answer fill-in-the-blank questions on English Exams, the reading strategies model [6] for reading comprehension, and the OpenAI transformer language model [4] for fit-the-best-sentence questions. With the invention of BERT [2], these questions can be answered more easily than before. Moreover, even for document-level questions such as SQuAD[5], BERT also achieves state-of-the-art performance. As a result, our system uses fine-tuned BERT models to predict the answer for each type of question.

3 SYSTEM

Fig. 1 illustrates the Exam Keeper framework. The first step is providing the exam questions and the answer keys. Exam Keeper provides two ways for users to input their data for examination: uploading existing files or using Exam Creator. Using the former method, teachers upload two files: one with the exam questions and the other with the answer keys. The two files should be formatted according to the instructions on the website. Teachers can also design and check exam questions one-by-one in the question design

*Equal contribution.
[1]The web page will be responsive and can be used via PCs upon demo.

WWW '19, May 13–17, 2019, San Francisco, CA, USA

ACM ISBN 978-1-4503-6674-8/19/05.
https://doi.org/10.1145/3308558.3314130

[2]http://www.yuansouti.com
[3]http://www.afanti100.com
[4]http://www.doudoushuxue.com/about/intro
[5]https://www.nii.ac.jp/userdata/results/pr_data/NII_Today/60_en/all.pdf
[6]https://github.com/tensorflow/models/tree/master/research/lm_1b

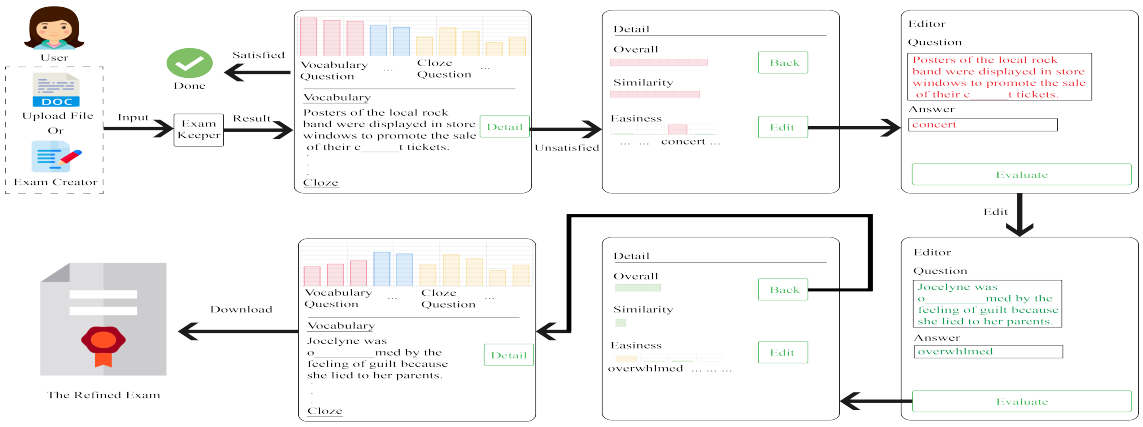

Figure 1: System flow

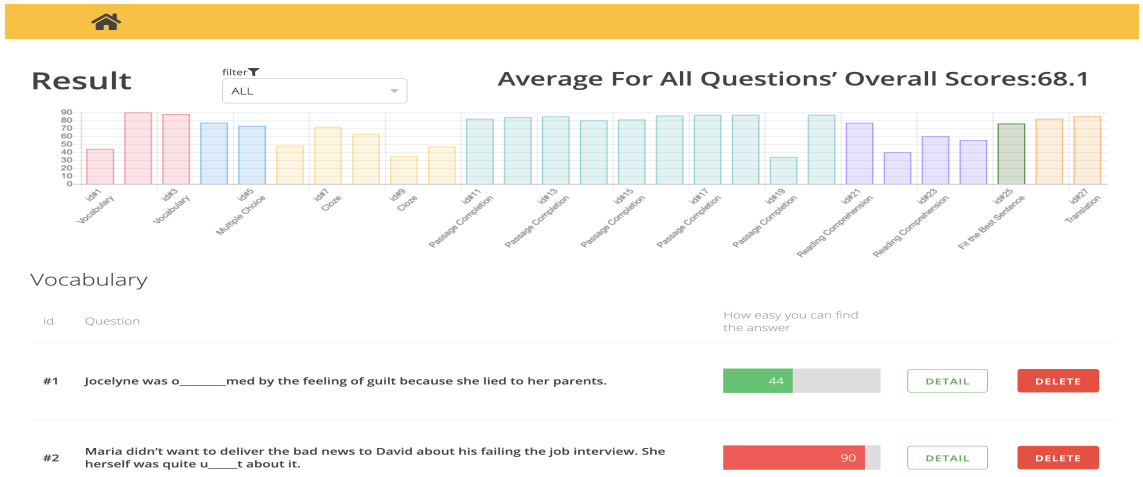

Figure 2: Result page

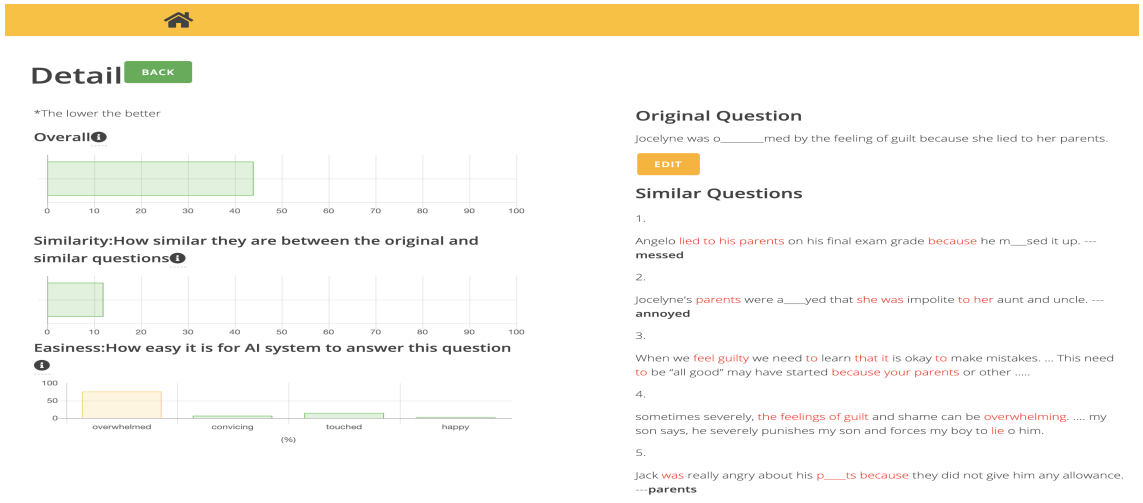

Figure 3: Question details interface

procedure by Exam Creator provided by Exam Keeper. Using Exam Creator, teachers can *test* the question and edit it accordingly until they are satisfied.

The current version of Exam Keeper supports the following question types: vocabulary, multiple-choice, cloze, passage completion, fit the best sentence, reading comprehension, and translation. Table 1 shows an example question for each type.

Exam Keeper represents the examination results at different granularities over several pages. As an overview, the top of the result page shown in Fig. 2 shows the overall accessibility of the answers in the exam and the corresponding score for each question, followed by the submitted questions with their individual scores. We mark each score with a color for straightforward visualization: red for easily-found answers (the teacher should reconsider this question), yellow for medium, and green for hard (should be good to use). The lower the score, the better the question.

The detailed calculation of the question score can be viewed clicking the *detail* button, which shows three scores: Overall, Similarity, and Easiness, as well as similar questions found on the Web and in the exam questions database. The Overall score is the average of the Similarity and Easiness. Similarity is based on how similar the original question is to the retrieved questions; it is measured in relation to answers found on the web. Easiness is measured in relation to answers found using the available QA systems or apps. Fig. 3 illustrates an example details page of a Vocabulary question, which also provides an editing function for question refinement. After going through all the questions, the teacher can download the exam sheet from Exam Creator.

4 MODELS

Exam Keeper includes several models: a retrieval-based model to retrieve similar question-answer pairs and one learning-based model for each type of question. To train these models and retrieve similar questions, we use the CLOTH [8] dataset for cloze questions, the RACE [3] dataset for reading comprehension, and the SWAG [9] dataset for fit-the-best-sentence questions. For the other types of questions, we collect raw data from free and public websites in China, Japan and Taiwan that gather exams created by English teachers.

4.1 Retrieval Based Model

We use Apache Solr[7], an extension of Apache Lucene[8], to retrieve similar questions for each input question. When a query is searched, Solr will return the documents ranked by modified Vector Space Model(VSM) score[9] with the query. VSM score of document d for query q is the Cosine Similarity of the weighted query vectors $V(q)$ and $V(d)$ whose weights are tf-idf values. Solr uses modified VSM score formula to rank documents that is not between 0 and 1. However, because we hope that the value of a score is between 0 and 1, we change it back to the original one.

The collected datasets are fed into Solr with the question type represented using the *question_type* field in Solr. For each question query, we search using Google Search and add the top ten snippets

Table 1: Example question types

Vocabulary
Maria did not want to deliver the bad news to David about his failing the job interview. She herself was quite u_____t about it.
Multiple Choice
Mangoes are a _____ fruit here in Taiwan; most of them reach their peak of sweetness in July. (A) mature (B) usual (C) seasonal (D) particular
Cloze
It is believed that taking breaks from a problem can help __1__ a moment of insight or stimulate new ideas. Unconventional solutions can also be explored. That is why some of the most successful companies in the world, such as 3M and Google, encourage their employees to __2__ all sorts of relaxing activities, such as playing pinball and wandering about the campus. 1. (A) spark (B) carve (C) drill (D) grind 2. (A) refer to (B) answer for (C) take part in (D) put up with
Passage Completion
One of these stories __1__ the Fortune cookie's origin back to 13th- and 14th-century China, which was then occupied by the Mongols. According to the legend, notes of __2__ plans for a revolution to overthrow the Mongols were hidden in mooncakes that would ordinarily have been stuffed with sweet bean paste. The revolution turned out to be __3__ and eventually led to the formation of the Ming Dynasty. (A) account (B) appeared (C) competing (D) contained (E) replaced (F) secret (G) successful (H) tastes (I) traces (J) treats
Fit the Best Sentence
On stage, a woman takes a seat at the piano. She _____ (A) sits on a bench as her sister plays with the doll. (B) smiles with someone as the music plays. (C) is in the crowd, watching the dancers. (D) nervously sets her fingers on the keys.
Reading Comprehension
For more than two hundred years, the White House has stood as a symbol of the United States Presidency, the U.S. government, and the American people. (...Skipped...) 1. What is this passage mainly about? (A) The design of the White House. (B) The location of the White House. (C) The importance of the White House. (D) The history of the White House. 2. Who initiated the construction of the White House? (A) John Adams. (B) James Hoban. (C) George Washington. (D) Thomas Jefferson.
Translation
Please translate the following sentences to English (This could be any language other than English) 1. 軍主導の暫定政権が来月予定していた総選挙の延期を示唆したことに 2. 相較於他們父母的世代，現今年輕人享受較多的自由和繁榮。 3. 成功的人都有着清晰的愿景。

[7] http://lucene.apache.org/solr/
[8] http://lucene.apache.org
[9] http://lucene.apache.org/core/7_4_0/core/org/apache/lucene/search/similarities/TFIDFSimilarity.html

into Solr according to the question type. Then we query the question in Solr and propose the top five results as similar questions. The top question's VSM score is used as the question's Similarity score in the details page.

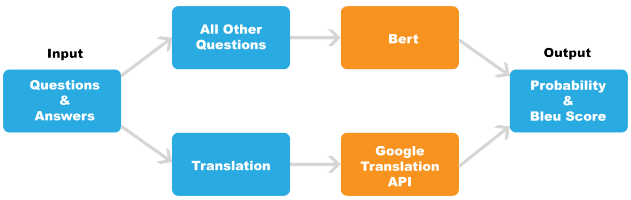

Figure 4: Model training flow

4.2 Learning Based Model

When Exam Keeper takes questions as input, based on the user-specified sections in the input file, we assign each question to one of the four categories, i.e., fill-in-the-blank, fit-the-best-sentence, reading comprehension, and translation. Models are then applied to questions by category. Fig. 4 illustrates the flow of the training process.

4.2.1 Fill-in-the-blank. This category includes four types of questions: vocabulary, multiple-choice, cloze, and passage completion. Though they appear different, they are transformed into the same format when they are input into the model: one blank masks at least one word off in a context and the model predicts the word(s). We utilize the current state-of-the-art to provide the most appropriate reference information by adopting the fine-tuned BERT [2][10] model, which is designed to solve cloze questions. After putting the context into the model, for each blank we retrieve the values for each word from BERT's dictionary. For vocabulary questions, we first search for the most likely words for the blank, and keep those words with the same first and last character given by the question. Then the top four candidates with their corresponding output values are selected. That with the maximum value is the proposed answer; the other three are also provided in the details page for teachers to use as they are similar semantically but are more difficult (for the learning models to answer). For the other questions, the softmax probability serves as the score for each answer candidate.

4.2.2 Fit-the-best-sentence. We use the BERT model designed for this type of question.

4.2.3 Reading Comprehension. During BERT's training process, in order to nail down the relationship between the contexts of two sentences, a binarized next-sentence prediction task is pretrained. This is the task we utilize to examine the reading comprehension question. In this task, the two context sentences A and B are stacked together and put into the BERT model, which predicts whether B is the next context sentence of A. In this sense, when performing downstream tasks that require information on two context sentences, we can put them into BERT as A and B and use the output of BERT's final layer for fine-tuning on each specific task.

[10]https://github.com/laiguokun/bert-cloth

Here, for reading comprehension questions, we set the article as A and each answer candidate as B, feed them to the pretrained BERT-Large model, and take the final hidden vector $C_i \in R^H$ corresponding to the first input token for each answer candidate i. The same procedure is followed for the article and question to yield C_q. In the fine-tuning stage, we use element-wise subtraction and multiplication [7] and concatenate the output of $C_q \ominus C_i$ and $C_q \otimes C_i$ to build matching representations $M_i \in R^{2H}$ for the article, question, and answer candidate. $\ominus$ and $\otimes$ are the element-wise vector subtraction and multiplication operations.

$$M_i = \begin{bmatrix} C_q \ominus C_i \\ C_q \otimes C_i \end{bmatrix} \tag{1}$$

Finally, we take the dot product of M_i and vector $V \in R^{2H}$ to get the score of i^{th} choice, where V is the only parameter needed to be trained for fine-tuning BERT on this task. Then we obtain the probability of each choice by applying softmax function over the scores of K choices.

$$Pi = \frac{e^{V \cdot M_i}}{\sum_{j=1}^{K} e^{V \cdot M_j}}. \tag{2}$$

4.2.4 Translation. We connect our application to the Google Translation API[11] and use the BLEU score between the generated translation sentence and the answer sentence to obtain the score. This score then is used as a measure of how easy it is for students to find the correct translation using Google Translate.

5 CONCLUSION

In this paper, we propose Exam Keeper, a system which calculates how easy it is for students to find exam answers. We hope this system benefits teachers in an era when technology distracts students from learning. Our future plan is to extend the application area of Exam Keeper to more subjects and to deploy it to different learning groups. In the end, we believe this is a worthwhile direction to explore for NLP techniques to improve educational environments.

ACKNOWLEDGMENTS

This research is partially supported by Ministry of Science and Technology, Taiwan, under Grant no. 105-2221-E-001-007-MY3 and MOST108-2634-F-002-008-.

REFERENCES

[1] Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, and Phillipp Koehn. 2013. One Billion Word Benchmark for Measuring Progress in Statistical Language Modeling. *CoRR* abs/1312.3005 (2013). arXiv:1312.3005 http://arxiv.org/abs/1312.3005

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805 http://arxiv.org/abs/1810.04805

[3] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard H. Hovy. 2017. RACE: Large-scale ReAding Comprehension Dataset From Examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*. 785–794. https://aclanthology.info/papers/D17-1082/d17-1082

[4] Alec Radford. 2018. Improving Language Understanding by Generative Pre-Training.

[5] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016,*

[11]https://cloud.google.com/translate/

Austin, Texas, USA, November 1-4, 2016. 2383–2392. http://aclweb.org/anthology/D/D16/D16-1264.pdf

[6] Kai Sun, Dian Yu, Dong Yu, and Claire Cardie. 2018. Improving Machine Reading Comprehension with General Reading Strategies. *CoRR* abs/1810.13441 (2018). arXiv:1810.13441 http://arxiv.org/abs/1810.13441

[7] Shuohang Wang, Mo Yu, Jing Jiang, and Shiyu Chang. 2018. A Co-Matching Model for Multi-choice Reading Comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers*. 746–751. https://aclanthology.info/papers/P18-2118/p18-2118

[8] Qizhe Xie, Guokun Lai, Zihang Dai, and Eduard H. Hovy. 2018. Large-scale Cloze Test Dataset Created by Teachers. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*. 2344–2356. https://aclanthology.info/papers/D18-1257/d18-1257

[9] Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. SWAG: A Large-Scale Adversarial Dataset for Grounded Commonsense Inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*. 93–104. https://aclanthology.info/papers/D18-1009/d18-1009