V for Vocab: An Intelligent Flashcard Application

Nihal V. Nayak[1], Tanmay Chinchore[1], Aishwarya Hanumanth Rao[1]
Shane Michael Martin[1], Sagar Nagaraj Simha[2], G.M. Lingaraju[1] and H.S. Jamadagni[2]
[1]M S Ramaiah Institute of Technology, Department of Information Science and Engineering
[2]Indian Institute of Science, Department of Electronic Systems Engineering
{nihalnayak,tanmayrc,aishwarya.hrao,shanemartin1995}@gmail.com

Abstract

Students choose to use flashcard applications available on the Internet to help memorize word-meaning pairs. This is helpful for tests such as GRE, TOEFL or IELTS, which emphasize on verbal skills. However, monotonous nature of flashcard applications can be diminished with the help of Cognitive Science through Testing Effect. Experimental evidences have shown that memory tests are an important tool for long term retention (Roediger and Karpicke, 2006). Based on these evidences, we developed a novel flashcard application called "V for Vocab" that implements short answer based tests for learning new words. Furthermore, we aid this by implementing our short answer grading algorithm which automatically scores the user's answer. The algorithm makes use of an alternate thesaurus instead of traditional Wordnet and delivers state-of-the-art performance on popular word similarity datasets. We also look to lay the foundation for analysis based on implicit data collected from our application.

1 Introduction

In recent times, we have seen how Internet has revolutionized the field of education through Massive Open Online Courses (MOOCs). Universities are incorporating MOOCs as a part of their regular coursework. Since most of these courses are in English, the students are expected to know the language before they are admitted to the university. In order to provide proof of English proficiency, students take up exams such as TOEFL (Test Of English as a Foreign Language), IELTS (International English Language Testing System),etc. In addition, students are required to take up GRE (Graduate Record Examination) in some universities. All these tests require the students to expand their vocabulary.

Students use several materials and applications in order to prepare for these tests. Amongst several techniques that have known to be effective for acquiring vocabulary, flashcard applications are the most popular. We believe the benefits of flashcard applications can be further amplified by incorporating techniques from Cognitive Science. One such technique that has been supported by experimental results is the Testing Effect, also referred to as Test Enhanced Learning. This phenomenon suggests that taking a memory test not only assesses what one knows, but also enhances later retention (Roediger and Karpicke, 2006).

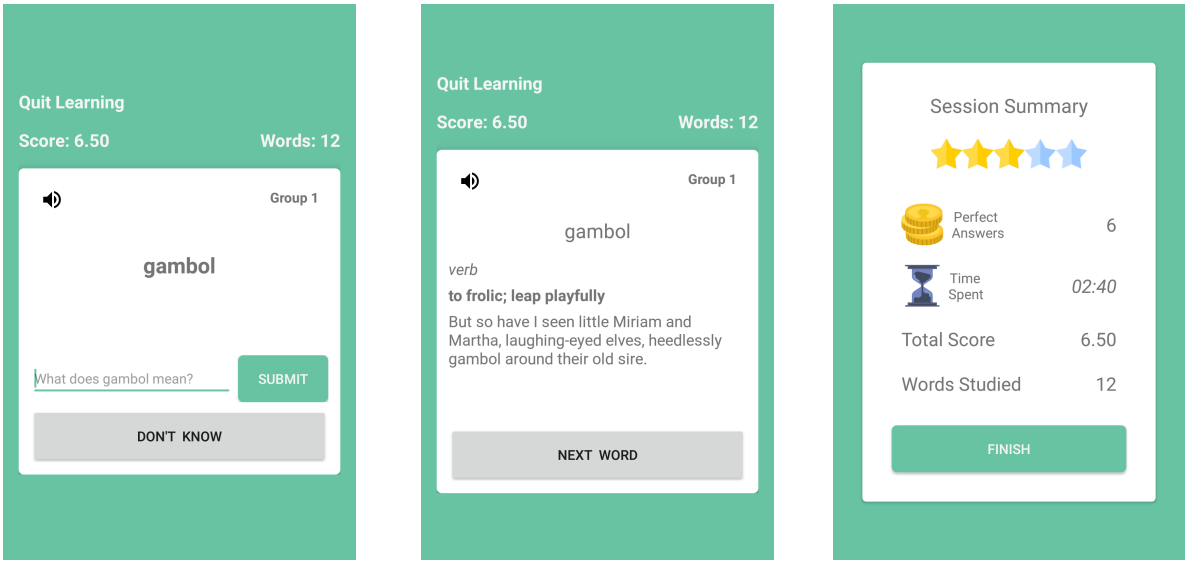
In this paper, we start by briefly discussing Testing Effect and other key works that influenced the development of the automatic short answer grading algorithm, implemented in V for Vocab[1] for acquiring vocabulary. Next, we have an overview of the application along with the methodology we use to collect data. In the later section, we describe our automatic short answer grading algorithm and present the evaluation results for variants of this algorithm on popular word similarity datasets such as RG65, WS353, SimLex-999 and SimVerb 3500. To conclude, we present a discussion that provides fodder for future work in this application.

2 Background

We have seen that flashcards have gained a lot of popularity among language learners. Students extensively use electronic flashcards while preparing for tests such as TOEFL, GRE and IELTS. Wissman et al. (2012) surveyed the use of flashcards among students and established that they are mostly used for memorization. To understand the

[1]https://goo.gl/1BBWN4

Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics- Student Research Workshop, pages 24–29
Vancouver, Canada, July 30 - August 4, 2017.
https://doi.org/10.18653/v1/P17-3005



(a) Front Flip (b) Back Flip (c) Score Card

Figure 1: a) Front of the card showing a textbox b) Back of the card giving feedback to the user c) Session Scorecard

decay of memory in humans, we delve into the concept of forgetting curve. Hermann Ebbinghaus was the first to investigate this concept way back in the 19th century. Since then, researchers have studied the benefits of several strategies which improve long term memory retention in an attempt to combat the forgetting curve. One such strategy is Testing Effect.

Our application is an amalgamation of the regular flashcard concept and Testing Effect. Roediger and Karpicke (2006) showed that repeated testing facilitates long term retention when compared to repeated studying. Further investigation revealed that short answer based tests are more effective in comparison to multiple choice question tests (Larsen and Butler, 2013). Experimental evidence also suggested that providing feedback to test takers improved their performance (Mcdaniel and Fisher, 1991; Pashler et al., 2005). This motivated us to incorporate short answer based tests with feedback in V for Vocab. To automate the process of scoring these tests, we developed a grading algorithm.

Since production tests allow the users to be more expressive, we had to develop an algorithm to grade answers that range from a single word to several words. The task of grading anywhere between a fill-in-the-gap and an essay is known as Automatic Short Answer Grading (ASAG) (Burrows et al., 2015). Thomas (2003) used a boolean pattern matching system to score answers which makes use of a thesauri and uses a boolean function OR to check with alternate options. FreeText Author (Jordan and Mitchell, 2009) provides an interface for teachers to give templates for the answer along with mandatory keywords. Different permutations for these templates are generated using synonyms obtained from thesauri. On similar lines, we developed an algorithm which employs an online thesaurus as a knowledge base.

3 Our Application

V for Vocab is an electronic version of the flashcard concept for learning new words. On these flashcards, we populate words from a popular wordlist[2] supplemented with sentences from an online dictionary[3]. These words have been divided into 25 groups and are saved in a database. The word, meaning and sentence combinations present in the data were verified by a qualified person. The interface we provide for our users is an Android Application. The application is designed to be simple and intuitive and is modelled based on other popular flashcard softwares.

On signing up, the user is prompted with a survey. The survey asks basic profile questions such as Name, Gender, Date of Birth, Occupation and

[2]https://quizlet.com/6876275/barrons-800-high-frequency-gre-word-list-flash-cards/

[3]http://sentence.yourdictionary.com

Number of Words	Raw Answers(%)	Bag of words of answers(%)
1	58.507	67.408
2	18.128	23.298
3	10.994	5.562
4 to N	12.369	2.356

Table 1: Statistical information regarding the data collected from our application where users had typed a meaning. The first column indicates the number of tokens or words in user's answers. N refers to the highest number of words typed by the user. The second column represents the percentage of raw answers or unprocessed responses, N = 16. The third column represents the percentage of answers after processing its bag of words, N = 8. However, after computing bag of words we saw of loss of 1.37% where the user's meaning was reduced to 0 words. In that case, the user's answer would not be graded.

Place of Origin. Apart from this, we ask whether the user is a voracious reader, whether the user is preparing for GRE and the background of the user. This background has been described by Educational Testing Service (ETS) [4], the organization that conducts tests such as TOEFL and GRE.

As mentioned earlier, the user can study from any of the 25 groups. Flashcards from the selected group are shown to the user one at a time in random order. On the front of the card, we provide a text field where the user may type his/her understanding of the word (Refer to Figure 1a). Regardless of whether the user submits an answer, the back of the card shows the word, its part-of-speech, dictionary meaning and a sample sentence (Refer to Figure 1b). This serves as feedback to the user as they review the meaning of the word. Before going to the next flashcard, we send implicit data to the server. If the user has submitted an answer, our algorithm scores it and returns back a score. On quitting, the user is prompted with a learning summary (Refer to Figure 1c).

3.1 Data Collection

During each flip of the card, V for Vocab collects implicit data from the phone in order to facilitate future analysis. The following data points are collected -

- Time spent on the front of the card in milliseconds
- Time spent on the back of the card in milliseconds
- Ambient Sensor value data in SI lux units

The ambient sensor value data is calculated by tapping into the mobile phone's light sensor. These values are found to be dependent on the manufacturer of the light sensor. They are only retrieved when there is a change in the sensor value data and stored in an array.

[4] www.ets.org/s/gre/pdf/snapshot_test_taker_data_2014.pdf

4 Short Answer Grading Algorithm

Algorithm 1: Grading Algorithm

Input: B1 & B2, the sets of Bag of Words for Text1 & Text2

Output: Similarity Score between Text1 & Text2

```
score, match_count, total_count ← 0
for w_i in B_1 do
    total_count ← total_count + 1
    for w_j in B_2 do
        flag ← 0
        for s_i in synonym(w_i) do
            for s_j in synonym(w_j) do
                if lemma(s_i) == lemma(s_j)
                 then
                    match_count ←
                     match_count + 1
                    flag ← 1
                    break
                shortend
            if flag == 1 then
                break
        if flag == 1 then
            break
score ← match_count/total_count
return score
```

In order to build a grading algorithm that suited V for Vocab, we first needed to understand the variation in the answers provided by our users. For

Dataset	S.L.	W.L.
RG65	0.632 / 0.617	**0.752 / 0.727**
WS353	0.286 / 0.313	**0.316 / 0.346**
Simlex-999	0.443 / 0.440	**0.523 / 0.521**
SimVerb-3500	0.278 / 0.276	**0.369 / 0.367**

Table 2: Pearson and Spearman rank correlation coefficients (separated by /; first one is Pearson correlation) computed between the human-annotated similarity score and the score given by our algorithm for a given pair of words from each dataset (S.L : Spacy Lemmatizer and W.L. : Wordnet Lemmatizer)

our analysis, we used 3027 data points collected over 2 months from different users. We found that in 1528 data points users had typed an answer. Based on statistical evidence, we observed that 58.507% of the answers were one word response. After performing bag of words computation on these answers, 67.408% of them were reduced to one word (See Table 1). This meant that our algorithm had to be tailored to grade one word answers, yet be versatile enough so as to grade answers which contained more words.

The answers from the users included a mix of synonyms for the main word or a paraphrase for the definition of the word. Therefore, in order to grade, we first compute the textual similarity of the answer with the word itself and then with the meaning from our database. These are considered as answer templates against which we compare the user's answers to compute the score. Our grader resembles the algorithm described in (Pilehvar et al., 2013) with minuscule changes in similarity measure, which is defined by the ratio of the total number of synonym overlap between word pairs in the answer templates and the user's answers to the total number of words in the answer template (See Algorithm 1). It should be noted that the bag of words is passed to the algorithm for computing the score. The algorithm scores the answers and returns a decimal score in the range [0,1] with a score of 1 being the highest.

Traditionally, people have used Wordnet (Miller, 1995) as a thesauri to find synonyms for a given word. Majority of the words in our wordlist being adjectives, Wordnet posed a disadvantage as it does not work well with adjectives. We also looked into word2vec (Mikolov et al., 2013), but we decided to not go with that approach as we got a high similarity score between a word and its antonym. Therefore, we preferred to retrieve the synonyms using a python module called PyDictionary [5]. This web scrapes synonyms from 21st Century Roget's Thesaurus [6].

We preprocess the user's answers with the help of a lemmatizer and stopwords list in order to compute the bag of words. The resulting bag of words is passed to the algorithm and it computes the strict synonym overlap between the user's answers and answer templates to calculate the score. Table 3 shows an example of the scores generated by our algorithm [7].

We developed this algorithm using lemmatizers from two popular NLP libraries - NLTK and Spacy, independently. Table 2 shows our evaluation results on popular datasets. We noticed that the algorithm produced higher correlation with NLTK's Wordnet lemmatizer, even though no explicit POS information was passed to the lemmatizer.

In case of an error caused due to absence of synonyms while web scraping, our algorithm returned a score of 0 which we have included during evaluation with the datasets.

User's Answers	Score
Trustworthy	0
Providing	0.33
Providing for the future	0.67
Frugal	1

Table 3: The table shows the evaluation of user's short answer for the word - provident, with the meaning - providing for future needs; frugal. Multiple meanings are separated by a ;(semicolon).

5 Discussion and Future Work

With trends showing that many applications curate their business model around data, we believe that the data collected from our application is valuable. We have the unique opportunity of performing analytics on an individual user and on all users as a whole. By analyzing the individual's data, we can personalize the application to each user. One way would be to observe the user's scores on the words studied and subsequently categorize them

[5] https://pypi.python.org/pypi/PyDictionary/1.5.2

[6] http://www.thesaurus.com

[7] The answers in Table 3 are compiled from the actual data we have collected from our users

into easy, medium and hard. We also have the potential to carry out exploratory analysis and bring out interesting patterns from our data. For example, we are hoping to discover an optimal duration to study words in a day so that the user can study effectively. Similarly, light sensor values could be used to understand how a user's learning would vary in a well lit environment versus a darker environment.

Spacing Effect is the robust phenomenon which states that spacing the learning events results in long term retention (Kornell, 2009). Anki, a popular flashcard application incorporates a scheduling algorithm in order to implement spacing effect. More recently we have seen Duolingo, a language learning application implement a machine learning based spacing effect called Half-Life-Regression (Settles and Meeder, 2016). With Testing Effect in place, it would be beneficial to incorporate spacing effect as it has shown great promise in practical applications . A thorough juxtaposition of Testing Effect versus the combination of Testing Effect with Spacing Effect, in terms of data, will help us better evaluate these memory techniques.

We can further improve the system through a mechanical turk. The turk could be any linguist or a person well versed with the language. The mechanical turk compares the answer templates with the user's answer and provides a score that represents how close the two are according to the turk's intuition. With labelled data, we can apply supervised learning and improve the algorithm.

When learning a new language, people often try to remember a word and its translation in a language they already know. For example, a person well versed in English who is trying to learn German will try to recollect word-translation pairs. With a bit of content curation for German-English word pairs, our grading algorithm will work seamlessly, as our algorithm is tailored to grade short answers in English. We believe that in future, V for Vocab can be ported to other languages as well.

Therefore, with the help of this application we are able to improve upon existing flashcard applications and lay groundwork for intelligent flashcard systems.

Acknowledgments

We thank the anonymous reviewers for their insightful comments. We also thank Dr. Vijaya Kumar B P, Professor at M S Ramaiah Institute of Technology, Bangalore for his valuable suggestions. This research was supported by Department of Electronic Systems Engineering (formerly CEDT), Indian Institute of Science.

References

Steven Burrows, Iryna Gurevych, and Benno Stein. 2015. The eras and trends of automatic short answer grading. *International Journal of Artificial Intelligence in Education* 25(1):60–117. https://doi.org/10.1007/s40593-014-0026-8.

Sally Jordan and Tom Mitchell. 2009. e-assessment for learning? the potential of short-answer free-text questions with tailored feedback. *British Journal of Educational Technology* 40(2):371–385. http://oro.open.ac.uk/15270/.

Nate Kornell. 2009. Optimising learning using flashcards: Spacing is more effective than cramming. *Applied Cognitive Psychology* 23(9):1297–1317. https://doi.org/10.1002/acp.1537.

Douglas P. Larsen and Andrew C. Butler. 2013. Test-enhanced learning. *Oxford Textbook of Medical Education* pages 443–452.

Mark A. Mcdaniel and Ronald P. Fisher. 1991. Tests and test feedback as learning sources. *Contemporary Educational Psychology* 16(2):192–201. https://doi.org/10.1016/0361-476x(91)90037-l.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, Curran Associates, Inc., pages 3111–3119. http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM* 38(11):39–41. https://doi.org/10.1145/219717.219748.

Harold Pashler, Nicholas J. Cepeda, John T. Wixted, and Doug Rohrer. 2005. When does feedback facilitate learning of words? *Journal of Experimental Psychology: Learning, Memory, and Cognition* 31(1):3–8. https://doi.org/10.1037/0278-7393.31.1.3.

Mohammad Taher Pilehvar, David Jurgens, and Roberto Navigli. 2013. Align, disambiguate and walk: A unified approach for measuring semantic similarity. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Sofia, Bulgaria, pages 1341–1351. http://www.aclweb.org/anthology/P13-1132.

Henry L. Roediger and Jeffrey D. Karpicke. 2006. Test-enhanced learning. *Psychological Science* 17(3):249–255. https://doi.org/10.1111/j.1467-9280.2006.01693.x.

Burr Settles and Brendan Meeder. 2016. A trainable spaced repetition model for language learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Berlin, Germany, pages 1848–1858. http://www.aclweb.org/anthology/P16-1174.

Pete Thomas. 2003. The evaluation of electronic marking of examinations. *SIGCSE Bull.* 35(3):50–54. https://doi.org/10.1145/961290.961528.

Kathryn T. Wissman, Katherine A. Rawson, and Mary A. Pyc. 2012. How and when do students use flashcards? *Memory* 20(6):568–579. https://doi.org/10.1080/09658211.2012.687052.