An Innovative Way to Study Vocabulary via App

Student : Shiyuan Zhang[1] Arnold O. Beckman High School, Irvine, CA 92620

ceciliazhang0923@gmail.com

Teacher : Yifan Du

duyifanucb@hotmail.com

ABSTRACT

As more students are required to have standardized test scores to enter higher education, developing vocabulary becomes essential for achieving ideal scores. Each individual has his or her own study style that maximizes the efficiency, and there are various approaches to memorize. However, it is difficult to find a specific learning method that fits the best to a person. This paper designs a tool to customize personal study plans based on clients' different habits including difficulty distribution, difficulty order of learning words, and the types of vocabulary. We applied our application to educational software and conducted a quantitative evaluation of the approach via IOS platform. By using data structure, filter, mapping, and reducing, the app allows users to create their own vocabulary base. The meme definition also allow young generation to have a joyful and fun way to accept vocabulary study. This is our goal: to provide a modern, acceptable, and cool way to study.

KEYWORDS

IOS Development, Xcode, Swift, Congressional App Challenge, study plan, vocabulary

1. INTRODUCTION

Vocabulary is essential in reading, writing and essay passages of standardized tests such as the Scholastic Assessment Test (SAT) and American College Test (ACT). Higherlevel vocabulary may appear in reading section since diverse variety of authors from historical to modern period wrote these reading passages. Similarly, questions of sophisticated vocabulary may appear in the writing passage; though it would not ask for the synonym of a word instead it would ask for the proper diction of the word. Building vocabulary will be efficient in improving scores, and learning new terms requires memorization skills.

Knowing the definition of complex words help students to answer the section of sentence completion and word definition correctly. However, reading, writing and essay sections cannot be completed without using memorization.[13] Since memorization is a common and vital process of developing a sophisticated level of vocabulary, learners spend many hours study to retain and recall vocabulary for their standardized test; but most of the time memorizing the definition of vocabulary can be very challenge. Although learners may

know the definition of a new complex words, they may forget the definition a month later. Thus, memorizing the definition of a vocabulary may not be a satisfactory choice.

Open Problem: Although there are many strategies to learn English vocabulary, these memorization strategies cannot bring quality result. [9][11] Learners may use books that provide the vocabulary that they must know before the standardized test. [10] However, they need to spend a lot of time memorizing the definition. Also, these books may be an old edition and contain a few words that may appear in the standardized test. Other technique is using word-search-puzzle games for improving vocabulary, but this game may not word for everybody since different learners think in different way. [12] Other strategies is using flashcards to recall the definition of words. This technique cannot be too sophisticated since learner need to memorize the definition. Furthermore, they would spend a lot of time when they need to search a word in multiple stacks of flashcard.

Solution: A pictorial method and automated personal study plan using machine learning and mobile computing. In this paper, we presented a new approach for recalling and retaining vocabulary. We have developed a mobile application that enable learners to customize their vocabulary by adding picture that assist them to remember the meaning of a vocabulary. Furthermore, this application can enable learner to input their desirable ACT/SAT score for the standardized test to obtain a personal study plan. In order to get the personal study plan learners must take an assessment. Once a learner finishes the assessment, the system would predict the group of words that is appropriate it for him/her.

Mobile Application – IOS Development

As shown in **Figure 1**, our app, Memelish, contains a home screen that provides 4 different options which users can choose during their learning process. The button in top left is a search bar, in which the user can type words and search for definitions and memes of the words. The button on top right can direct the user to the personal settings page after being clicked. It contains a banner on the top area which can tap on. After tapping it, the client will be directed to the latest unfinished study set in history. If there is no unfinished set in history, the banner will be titled with "Start a new set?" and automatically suggest a new set based according to the client's personal study plan. There are four buttons in the middle area which connect to the Learn, DIY, and Me pages. The Random button will randomly generate a list to test the user's skill level.
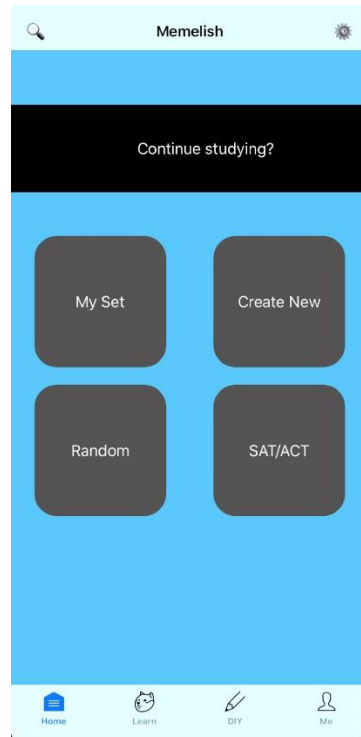

Figure 1: Home page screen

In order to create and set up the home screen buttons, we used UIBarButtonItem and UIStackView (see **Figure 2**). We used UIBarButtonItem to create the search and setting buttons. In this UIBarButtonItem, we passed icon image as a title and set the corresponded action that the button needs to execute when the button is pressed. For "My Set", "Create New", "Random" and "SAT/ACT", we used UIStackView. These buttons are created and set up in the stackview() function. In order to locate the button in the desirable position, we used 2 different lists that contains 2 buttons and passed it to the UIStackView.

```swift
let leftButton = UIBarButtonItem(title: "🔍", style: .done, target: self,
action: #selector(handleLeftButton))
navigationItem.leftBarButtonItem = leftButton

let rightButton = UIBarButtonItem(title: "⊙", style: .done, target: self,
action: #selector(handleRightButton))
navigationItem.rightBarButtonItem = rightButton

func stackview(){
    let list = [MySet, CreateNew]

    MySet.heightAnchor.constraint(equalToConstant: 150).isActive = true
    MySet.widthAnchor.constraint(equalToConstant: 150).isActive = true

    CreateNew.heightAnchor.constraint(equalToConstant: 150).isActive = true
    CreateNew.widthAnchor.constraint(equalToConstant: 150).isActive = true
```

```swift
    stackView1.topAnchor.constraint(equalTo: view.topAnchor, constant:
                              300).isActive = true
    stackView1.leadingAnchor.constraint(equalTo: view.leadingAnchor, constant:
                              30).isActive = true
    stackView1.trailingAnchor.constraint(equalTo: view.trailingAnchor,
                              constant: -30).isActive = true
    stackView1.heightAnchor.constraint(equalToConstant: 150).isActive = true

    let array = [Random, ACT]

    Random.heightAnchor.constraint(equalToConstant: 150).isActive = true
    Random.widthAnchor.constraint(equalToConstant: 150).isActive = true

    ACT.heightAnchor.constraint(equalToConstant: 150).isActive = true
    ACT.widthAnchor.constraint(equalToConstant: 150).isActive = true

    let stackView2 = UIStackView(arrangedSubviews: array)
    stackView2.axis = .horizontal
    stackView2.translatesAutoresizingMaskIntoConstraints = false
    stackView2.distribution = .equalSpacing
    stackView2.spacing = 10

    view.addSubview(stackView2)
    stackView2.topAnchor.constraint(equalTo: stackView1.bottomAnchor, constant:
                              20).isActive = true
    stackView2.leadingAnchor.constraint(equalTo: view.leadingAnchor, constant:
                              30).isActive = true
    stackView2.trailingAnchor.constraint(equalTo: view.trailingAnchor,
                              constant: -30).isActive = true
    stackView2.heightAnchor.constraint(equalToConstant: 150).isActive = true
}
```

Figure 2: Code Segment to create and set up button in homescreen

Figure 3 shows the search engine mentioned in the home screen. The user needs to click the search bar and enter a word. The algorithm can find the words in the database based on user inputs. Also, it will return a list of previewed outcomes with icons and words. The user can tap on one cell of the list for details. Details include the original images, words, and definitions. There is a button on the bottom left corner for users to add the word into personal study sets.

The core tech is using UIsearchController to capture user input in searchbar. The user input will be converted into string. The string will be processed by filter method with $0 syntax. The Boolean value of isSearchMode is a switch, which is just like people's home light switch, the default value is false. Boolean value is a useful type in control flow and we apply this feature in our search engine.
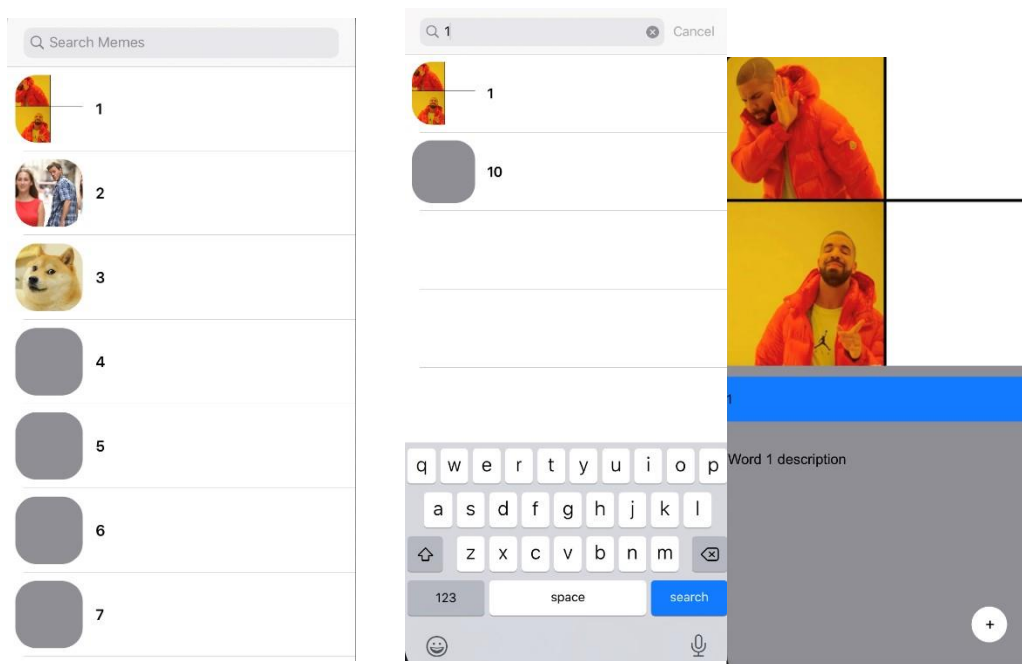

Figure 3: Searching page contains all vocabulary in alphabetic order. (left) Searching page that contains words based on user Input. (middle) Vocabulary page that contains word, definition and image. (right)

```swift
extension SearchWord{
    override func tableView(_ tableView: UITableView, didSelectRowAt
                                    indexPath: IndexPath) {
        let explanation = ExplanationVC()

        if inSearchMode{
            explanation.memeString = searchData[indexPath.row].listNameData
            explanation.imageString= UIImage(named:
            searchData[indexPath.row].listNameImage) ?? UIImage(named: "1")!
                //send image + if no image then use image1
                self.present(explanation, animated: true,  completion: nil)
        }else{
            explanation.memeString = data[indexPath.row].listNameData
            explanation.imageString = UIImage(named:
                data[indexPath.row].listNameImage) ?? UIImage(named: "1")!
                self.present(explanation, animated: true, completion: nil)
        }
    }
}
```

Figure 4: Algorithm uses for searching word in the search engine.

Figurer 5 is the editing page for users to make their own memes to help memorize the words. Users can click on the plus button to import their own images from their phones or to select templates from online sources. The text button allows users to add text on the image. Draw button can let users paint with various colors by using fingers. If users make a mistake while drawing, they can choose to undo the last step by clicking the undo button or to erase the drawing by tapping the eraser button. Below the tool buttons, there is a rectangular area that will demonstrate the word and its definition. On top right, there is a save button that allows users to save DIY memes to their study sets.
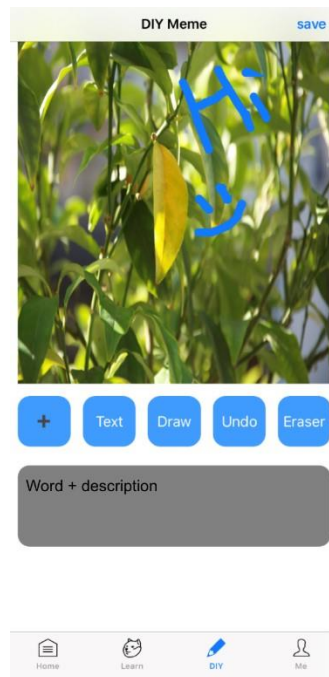

Figure 5: Editing page for adding memes to a vocabulary.

```swift
override func draw(_ rect: CGRect){
    super.draw(rect)
    guard let context = UIGraphicsGetCurrentContext() else {return}
//exception
    context.setStrokeColor(UIColor.systemBlue.cgColor) //control the color of
the pen
    context.setLineWidth(10) //control the width of the pen
    context.setLineCap(.round) // pen shape

    lines.forEach { (line) in
        for (i,j) in line.enumerated(){
            if i == 0{
                context.move(to: j)
            }else{
                context.addLine(to: j)
            }
        }
    }

    context.strokePath()
}

//2D coordinate set
fileprivate var lines = [[CGPoint]]()

override func touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?){
    lines.append([CGPoint]()) //所有点连起来成线

}

override func touchesMoved(_ touches: Set<UITouch>, with event: UIEvent?){
    guard let point = touches.first?.location(in: nil) else {return}
    guard var endPoint = lines.popLast()else{return}
    endPoint.append(point)//pen up, append the end point when finished drawing
    lines.append(endPoint) //collect the first&last points to form a line
    setNeedsDisplay()
}
```

Figure 6: Code segment for creating the editing and setting up page

Figure 7 is a list of view of study sets. Users can scroll up and down to view the entire list. This page lists out all words in the set with words and meme previews. The preview images might be adjusted to a low resolution, but users are able to view the original image by clicking on the words. After tapping, users will enter a detailed view page that contains word, image, definition, and an explanation of how the word and the image relate to each other.
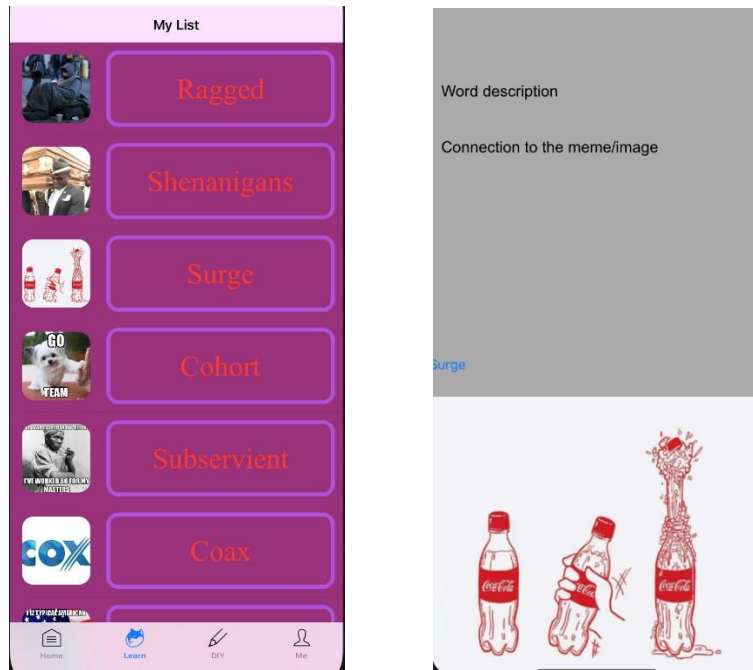

Figure 7: Study slist page contains words and memes. (left). Vocabulary page contains word, definition and image. (right)

```swift
override func tableView(_ tableView: UITableView, cellForRowAt indexPath:
IndexPath) -> UITableViewCell {

    let cell = tableView.dequeueReusableCell(withIdentifier: "cell", for:
                                              indexPath) as! ListCell

    cell.NewLabel.text = vocab[indexPath.row]
    cell.backgroundColor = CCFavColor
    cell.textLabel?.font = UIFont.boldSystemFont(ofSize: 28)
    cell.textLabel?.textColor = UIColor.systemGreen
    cell.coolerImageView.image = UIImage(named: imageList[indexPath.row])
    return cell
}

override func tableView(_ tableView: UITableView, numberOfRowsInSection
                                  section: Int) -> Int {
    return vocab.count

}

override func tableView(_ tableView: UITableView, didSelectRowAt indexPath:
IndexPath) {
        let dtvc = DetailedVC()
        dtvc.imageString = UIImage(named: imageList[indexPath.row])!
```

Figure 8: Code segment for displaying study list page

As show **Figure 9**, there is a personal settings page where users can edit their profiles. The gray block on top illustrates the user's profile picture. Users can change their profile pictures by tapping on the gray block and selecting an image from a local photo library. The username will be demonstrated below the profile picture. If users click on My Account button, their account information including name, email address and total number of memorized words will appear. My collection button will direct users to a list of the study sets they collected from others. The View History button can show the recent records of study sets users viewed. Lastly, My Plan button generates a personal study plan by utilizing machine learning.[15] [16] The user only needs to take an evaluation test and enter the goal. The program will recommend study sets corresponding to the user's skill level and record performance.
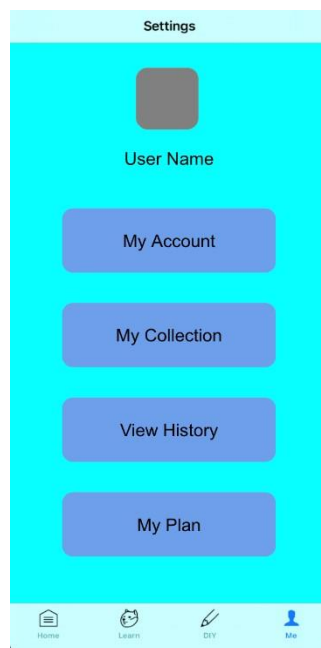

Figure 9: Personal setting page

```swift
func TapToPhotoLib() {
        let tap = UITapGestureRecognizer(target: self, action:
#selector(handleTap))
        profileImageView.addGestureRecognizer(tap)
}

@objc func handleTap()        {
        //create image picker to select image from lib or camera
        let picker = UIImagePickerController()
        picker.delegate = self
        picker.sourceType = .photoLibrary
        picker.allowsEditing = true
        self.present(picker, animated: true, completion: nil)

}
```

Figure 10: Code segment for adding image to the profile

6. CONCLUSION AND FUTURE WORK

Vocabulary is vital to communicate our thought and perform satisfactory in standardized test such as ACT and SAT. In order to be successful is standardized test, learners use appropriately study plan methods. However, different learning plan can be suitable for some learners but not for others. Some of them may use flashcards, vocabulary games, pictorial method etc. Using flashcards and memorizing the definition may seem efficient, but learners may forget the definition after a while, so all the work would be wasted.

In this project, we proposed an intelligent study method to solve memorization issue using mobile computing and machine learning; and a memorization method to recall and retain vocabulary by using picture and memes. A mobile application has been developed to design an efficient study plan. To obtain a study plan, learners must enter the desirable ACT/SAT score and take an assessment. Once a learner finished the assessment, the system predicts a study plan that is suitable for the learner and return a study plan based on his/her performance.

As a future work, we plan to gather a group of high school students to use our mobile application during the SAT study period and observe if our mobile application improved learners' memorization.