SMART SHOPPING APPLICATION:A mobile Application for your Shopping Needs

Aayush Patial
North Carolina State University
Raleigh, NC
apatial@ncsu.edu

Shivani Vyas
North Carolina State University
Raleigh, NC
spvyas@ncsu.edu

ABSTRACT

A lot of modern technological advances are driven by people‘s increasing need to save time and money and thereby to organize their everyday routines efficiently. Grocery shopping is one of the everyday mundane tasks performed by majority of the population around the world. The concept of supermarkets have eased this task by having variety of items at one place. With this concept of supermarkets, the task of preparing the shopping list proves very beneficial so that one buys more items that are absolutely required compared to the ones people buy out of interest. This paper aims at explaining a system that prepares such list for users which includes the items that are absolutely needed by the user. It aims at using the purchase pattern of the user and thereby generate a list of items that the user is most likely to be in need of.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications — Data mining; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—Information Filtering; D.2 [**Software Engineering**]

General Terms

Mobile application, Shopping list, Android

Keywords

Predictive Shopping, Requirement Gathering, Neural Networks

1. INTRODUCTION

Forgetfulness is a pretty common characteristic associated with grocery and regular household shopping. The reason for this is there are items that get used up very quickly like milk and bread whereas there are items like washing soap or oil that lasts for a longer time. With most of the population being students or working class makes it difficult to run errands multiple times a week which makes it useful to have some system that can give you the list of items which is needed by the user at the time he/she is planned to go shopping.

The presently implemented system aims at having more accuracy of the items that a user needs but also making sure to remove the redundancy. Re-using the items already listed once can be used only to an extent as each item will have a different purchase cycle. Groceries like vegetables and butter needs to be on the list every week but items like hand wash or shampoo can be on the list once a month or once in two months based on the usage. Over and above this factors like time of the year also needs to be considered. If the user is a student, he/she is most likely to visit relatives or go for vacation with friends in which case the usage of things will reduce or almost stop for at least a weak. But in this scenario one has to consider that perishable items like fruits or vegetables may go bad or stale whereas washing items stay the same and can be used once the user is back. On the other hand if a family is hosting guests, will lead to more consumption of all the eating and cleaning items. So, the system must be smart enough to consider the parameters like seasons and festivals.

The paper discusses how the generation of shopping list can be made easier compared to what is presently being done. Various features that are discussed in the subsequent sections of the paper are motivated to make the shopping experience for the user a more efficient and effective one. The aspects of data collection and result display can be done in a more efficient way compared to the present implementation and the same is the focus of this paper.

2. LITERATURE REVIEW

2.1 Methodology

In order to learn about the past and ongoing work on predictive shopping, we made use of Google scholar and Google Ngram Viewer. We studied various papers to study the progress of concept and used Google Ngram Viewer to study the trend of the concept.

2.2 Summary of papers

Due to immense usefulness and motivation of a predictive shopping list application, there has been a lot of methods and concepts that were devised to make the system more and more efficient and reliable. Following is the brief information of the previous work carried out in this sector.

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

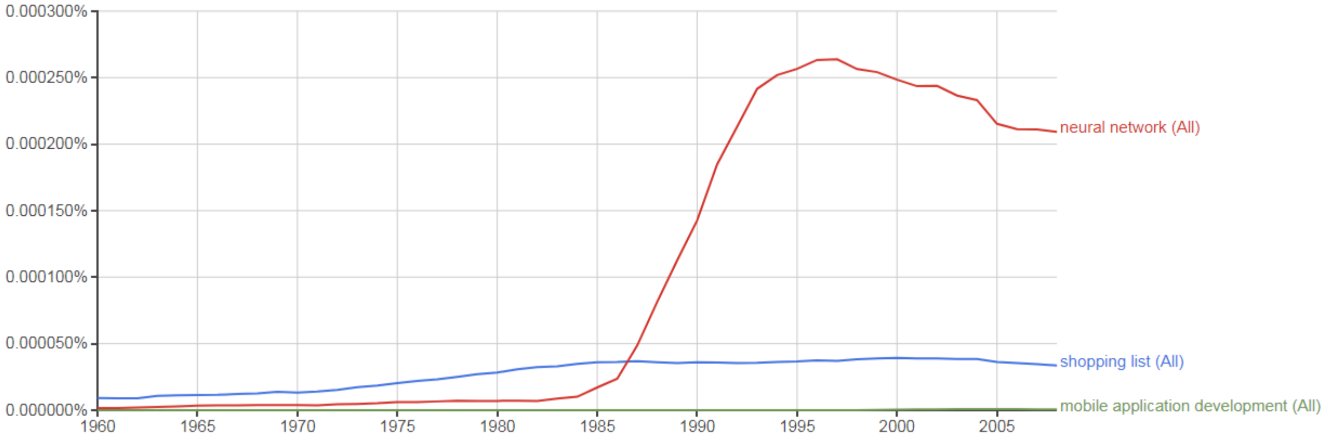


Figure 1: Results from Google Ngram Viewer.

The paper by Darin J. Dishneau, Patrick Joseph Derks[1] describes the techniques to determine a purchase pattern for a product based on a user's purchase history, share the purchase pattern with product sellers, receive discount offers for the product from those sellers, and notify the user of these offers through a mobile application. This helps the user to get cheaper deals on the stuff they buy often. Predictive text input in a mobile shopping assistant: methods and interface design by Petteri Nurmi, Andreas Forsblom, Patrik FlorÃl'en, Peter Peltonen, Petri Saarikko[2] describes the method for predictive text input technique that is based on association rules and item frequencies and also describes an interface design for integrating the predictive text input with a web-based mobile shopping assistant.

Consumers' decision-making process and their online shopping behavior: a clickstream analysis by Sylvain Senecal, Pawel J.Kalczynski, Jacques Nantel[3] investigates how different online decision-making processes that the users use, influence the complexity of their online shopping behavior. Investigating Consumers'Tendency to Combine Multipe Shopping Purposes and Destinations by Benedict G. C. Dellaert, Theo A. Arentze, Michel Bierlaire, Aloys W. J. Borgers, Harry J. P. Timmermans[4] aims at introducing a new integrated economic and experimental approach that helps to measure consumer choices for multipurpose shopping.

2.3 Results of n-gram viewer

We chose the following keywords Neural network, Shopping list and mobile application development for our study. The frequencies of the occurrence of these terms in books can be viewed in Figure 1. The time frame was chosen to suit the the period when the study about these concepts started.

On studying the graph, we realize that neural networks has been a much explored topic since its inception in 1994. Also shopping list has a constant popularity since 1970 which demonstrates the continuous work being done on it. Combining the concepts of neural networks for generation of shopping may therefore prove to have immense utility value. Also, mobile development for implementation of these predictive systems is losing its popularity which provides a future scope of implementing these systems on a different platform.

3. USER RESEARCH

3.1 Survey

In order to understand which features are most desired by a wide demography of users and will be adaptable to use, we conducted a survey including students, families and working class individuals.The questions and results for which are detailed in the sections below.

Question 1: How often do you feel the need to make a list before going grocery shopping ?

This was to understand what percentage of people actually follow the habit of preparing a shopping list as our end product will be of most use to such people.

Question 2: Do you monitor your monthly/weekly grocery or shopping expenses ?

This question was asked to understand how popular the feature of budget restrictions might be in the existing application of predictive shopping list.

Question 3: Do you keep track of items you purchase while shopping ? Follow up: If yes, how do you keep record of your purchased items ?

This was to understand the general thought process or methodology that the users instinctively apply while shopping. Taking for instance they like to explore the options available to them or they stick to certain items that they are need of. Follow up: This was to get an idea of the kind of interface most users will be comfortable with to store their purchase details.

Question 4: How would you like the mobile application to collect your purchase details?

We asked this question as the mobile application requires to collect the purchase details of the user and we wanted to understand which will be the most convenient way to do so like entering details or taking a picture.

Question 5: Would you prefer to have reminding notification about the items that you may run out of or expire before you go shopping ?

This question intended to determine if the user will prefer to be notified of the stuff they might run out of instead of them requesting a list generation on the date of their shopping or over and above it.

3.2 Results

The survey received close to 54 responses and the participants were individuals with family, who are students and working class living single. We kept the user demography so widespread in order to understand the shopping methodology of users from each of these user groups.

We can view from the Figure 2 that more 65% of people absolutely feel the need for preparing a shopping list. While 24% users may make a shopping list or may not. 4% of the people do not feel the need of making the shopping list at all.

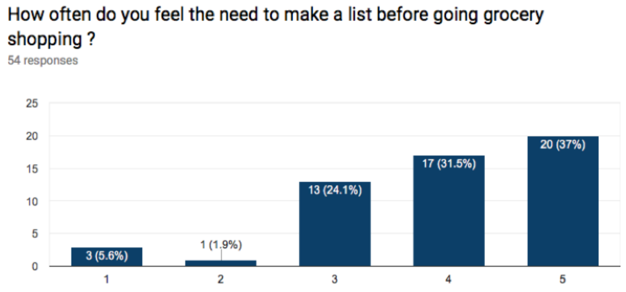


Figure 2: How often do you feel the need to make a list before going grocery shopping ?

Figure 3 shows that about 61% of users restrict their budget on a monthly or a weekly basis while grocery shopping. This tells us what part of user demography will like budget parameter to be considered while generating a shopping list.

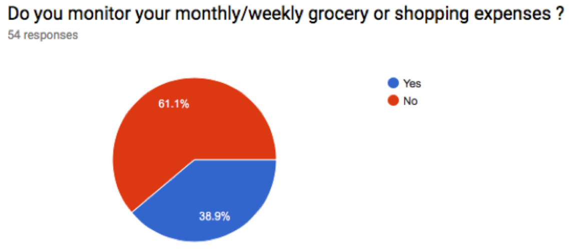


Figure 3: Do you monitor your monthly/weekly grocery or shopping expenses ?

Figure 4 shows that 70.4% users keep a track of items they purchased, in form of bills or recording the items manually. Our application requires purchase history of the user which is why it is important for us that users store their purchases in the forms provided as options.

Figure 5 gives an idea that 47% of people store their bills in order to keep a record of their purchase history while 33% of users prefer to store it on their mobile phones. 25% of users write it down after shopping. As almost half the perfect of people store their bills, it seems a better idea to capture these bills in order to collect the user's purchase history.

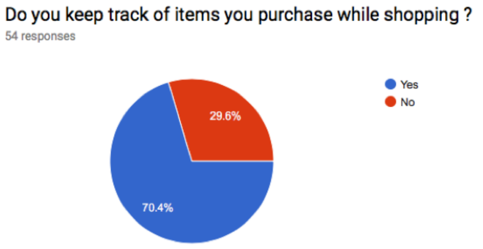


Figure 4: Do you keep track of items you purchase while shopping ?

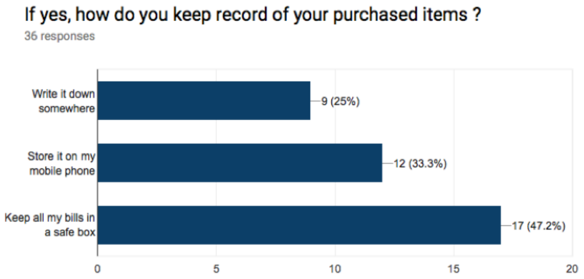


Figure 5: If yes, how do you keep record of your purchased items ?

Figure 6 shows that 72.2% of the users would like the mobile application to get their purchase history by scanning their bills in comparison to 17% people who will prefer to enter data through interactive components like text boxes and drop down menus. 1% of the users would write and strike which shows much less preference of pen/paper method of storing purchase details.

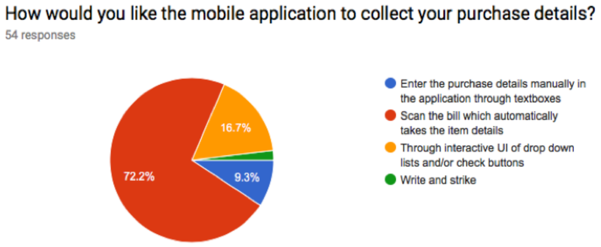


Figure 6: How would you like the mobile application to collect your purchase details?

Figure 7 shows statistical data of how open the users will be to the idea of notifications for items that they are soon to run out of or may expire and see that 70.4% of the users would like to have that feature in the present mobile application.

4. PROJECTED TIMELINE

This section shows the projected timeline of how to project is going to be worked upon:

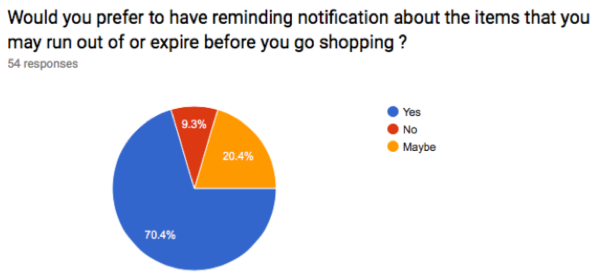


Figure 7: Would you prefer to have reminding notification about the items that you may run out of or expire before you go shopping ?

TimeLine	
Month	Progress
March 20, 2018	Finalized project idea and title
March 25, 2018	Setup the project architecture/environment and planned features to implement
March 26, 2018	Conducted user survey
March 29, 2018	Report 1 submission
April 2, 2018	Integrating OCR with the application
April 6, 2018	Increasing the database for the recommendation engine training
April 11, 2018	Connecting the android notification manager for the push notifications
April 14, 2018	Integrating features
April 16, 2018	Evaluation Survey
April 17-19, 2018	Project Presentation
May 1, 2018	Report 2 submission

5. EXISTING AND PROPOSED ARCHITECTURE

In this section we have shown the proposed architecture of our project. Proposed architecture is as shown in Figure 8.

5.1 Existing Architecture

The project uses a basic client-server architecture. Here the mobile application acts as the client. The Server here is implemented as nodeJS and server calls are made through the REST API.The server will have connection to the mongoDB database which will have details of the user purchases. The algorithm for prediction of shopping list works on the datasets derived from the database and gives out the results on user request. Here the user is asked to input maximum details possible of his/her purchase in order for the algorithm to work more efficiently.

5.2 Bugs in previous system

While understanding what the application was doing and understanding the functionality in the application we found few bugs in the application:

1. Forgot password feature not functioning
2. UI security flaw of opening directly into survey page
3. Database complications with price and item count storage
4. Application running on local server
5. Need of larger dataset to support neural network

While we were not able to fix all the issues in the application due to time constraints but, we did try to add more functionality to the project trying to enhance the experience.

5.3 New Updated Architecture

We are making additions to the existing system architecture to implement the features of Optical Character Recognition(OCR) and Reminder Notification for the user.

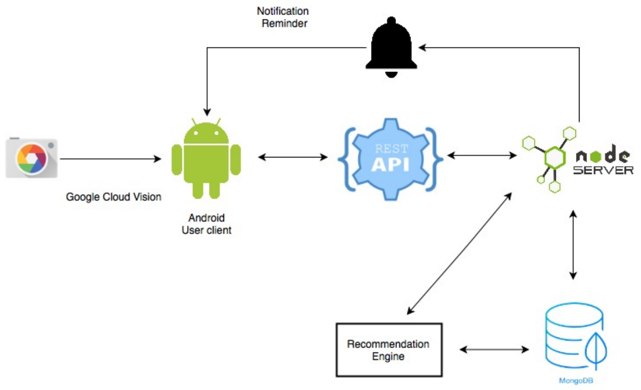


Figure 8: Updated System Architecture

The addition of the Bill Parsing feature makes use of camera application in the user's android mobile Phone. Using the cloud vision API from Google we get the content from any bill that the user wishes to enter. That string is then filtered and passed onto the server through which it is stored in the database. This process aides user to quickly go through the bill without having to enter every detail manually.

Another addition that is planned is to add the reminder notification to the user when the recommendation engine predicts how the current supplies are going to end and notify user beforehand so that user is prepared to purchase those items before the next visit to the market. This is done using the built in notification manager in the android which similar to how Rest API interacts with the server and decides when to push notification to the user application.

Further using the existing changes on the server side we intend to add the forgot password functionality to the application that for now does not exist on it. This would increase ease of use for the user. Also for the better functioning of the recommendation system we plan to add more data so as to get better results from the neural network that is already present thus refining the output.

5.4 Changes from previous system

For development of the application we decided to build on the existing framework of the previous application using the same technology used for backend and improve upon the main features it had. we made the following changes:

1. Bill Parser using google cloud vision API
2. Push notification reminder
3. UI updates

To start with the application UI seemed very bland at the beginning because user was directly greeted with the parameter survey every time he or she opened the application. This seemed very counter-intuitive. So we made some UI changes like introduced a Home Page for the user.

The previous application had a grid where user had to enter all details about the purchases before uploading the image. We thought this to be a very time consuming process and sort of a demotivating factor for the user. In case the list is very long user would not enter any detail and the application would not be able to serve its purpose then.

Also we observed that the application had a built in neural network which predicted the output but wasn't being used properly and we decided to use the output by notifying the users through notifications whenever the neural network predicted the date where a specific item would be needed again.

5.5 Application flow

The figure 9. describes the flow of our application updated from previous model from the point user opens the application to all the things user can do with it.

6. APPLICATION REVIEW

This is an Android application developed using the Android studio. The different APIs used in this application are Volley and Gson library to parse json data to string. Google cloud vision API and Notification library of Android has been used to implement OCR which helps to scan the bills and send reminders to users.

6.1 Login/Signup

Once the user installs the application,user's previous list and purchase history is synced with the account if the user has an existing account and if not, the user is asked to sign up with username and password.

This is one of the few structural UI changes we made as seen in figure10. adding a home screen to the application to avoid cluttering with the survey data in the login itself as the application tends to retain the home page in saved state and could cause security concerns allowing another user to enter data filters which could lead to faulty training of the neural network.

6.2 Data Collection

In order to run the neural network algorithm in order to predict future purchase list of the user, the user's purchase history needs to be captured. In the previous version of the application, the user was required to manually enter the details of item purchased, quantity purchased and the date of purchase. This is a tedious process if the list is too long so we implemented OCR in order to capture the bills and extract the items. While implementing the OCR we have retained the original grid format in case user is not getting current output and wants to manually enter an item. The following data is collected from the user :

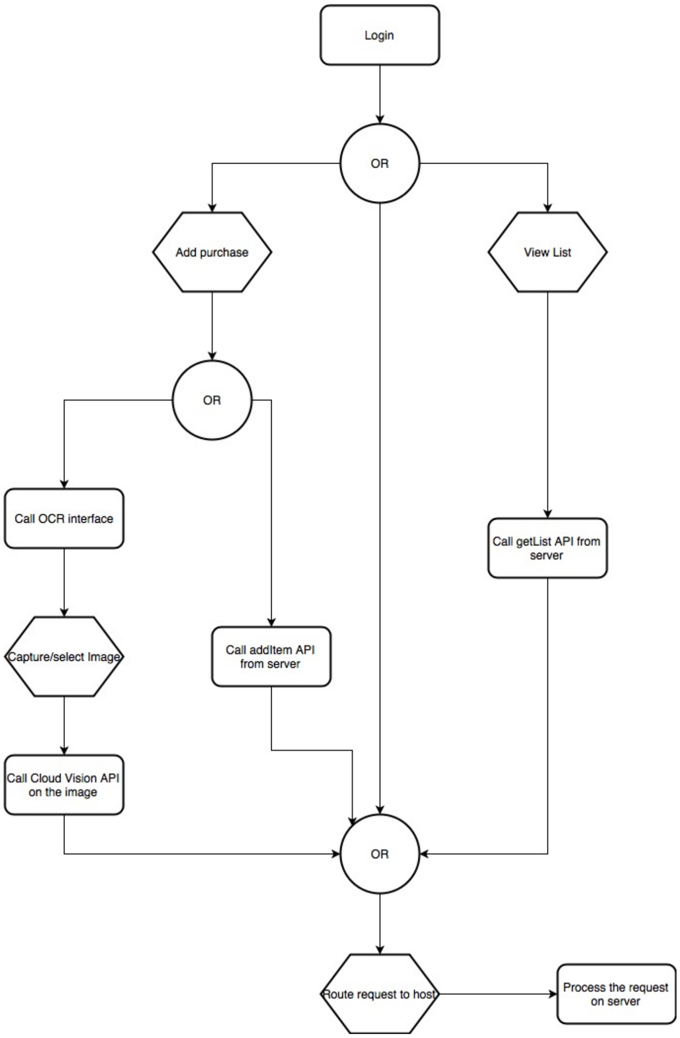

Login
OR
Add purchase
View List
OR
Call OCR interface
Call getList API from server
Capture/select Image
Call addItem API from server
Call Cloud Vision API on the image
OR
Route request to host
Process the request on server

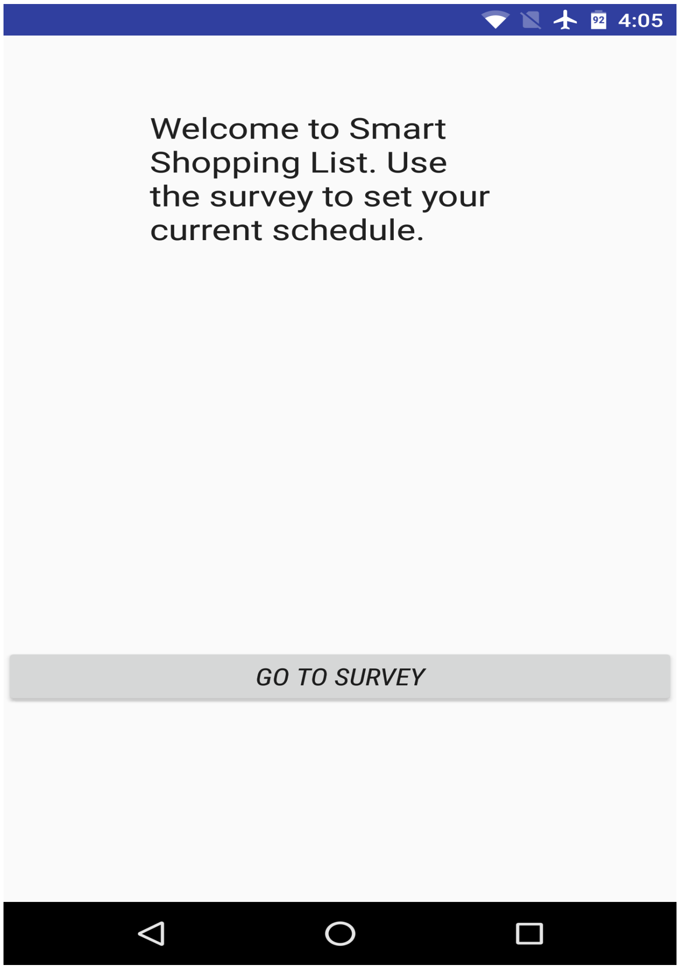
Figure 9: Flow Chart of the android application

- Item: Name of the item purchased.
- Quantity: The number/weight/amount of the purchased item.
- Price: This is the cost of item purchased which will be helpful for further enhancement of application.
- Date: This is the date of item purchased. This will help determine the average time taken by the user/ family/ group to use up the product.

In the figure 11. you can see we added the use camera option, to send user to the OCR interface where user can just use image to extract data.

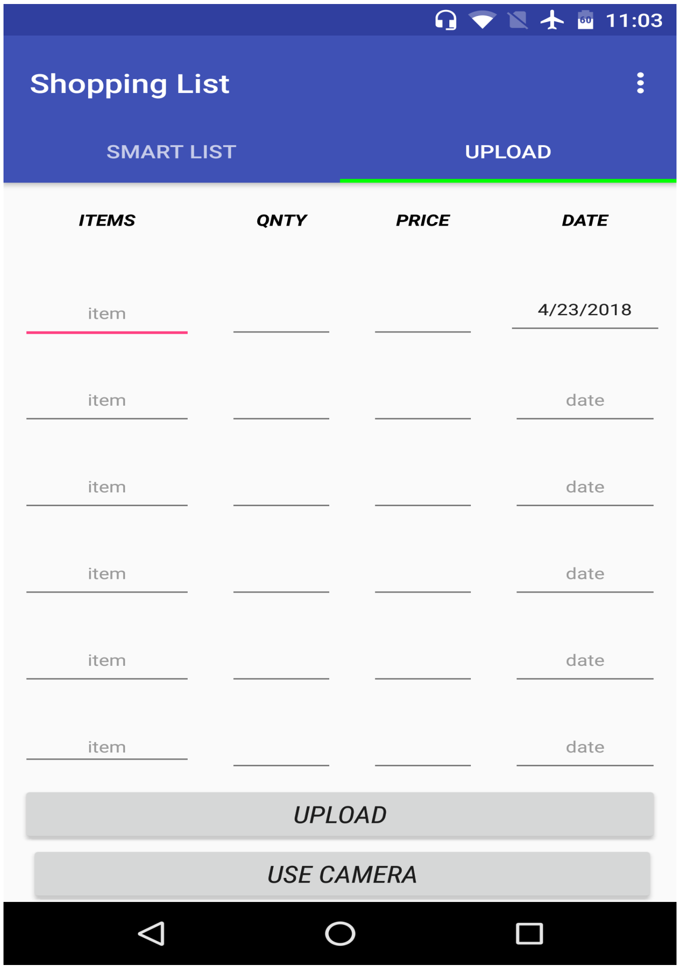
6.3 Optical Character Recognition

This is the new feature that we tried to implement. We used the Google cloud vision API to extract text out from



Welcome to Smart Shopping List. Use the survey to set your current schedule.

Figure 10: After login Home screen



ITEMS	QNTY	PRICE	DATE
item			4/23/2018
item			date
item			date
item			date
item			date
item			date

Figure 11: The upload tab with camera option

the the image and parse it to the server. We created a separate activity in the application for this where the user would be able to go and select an already existing image from the library or use camera to capture a new image. The API is used to connect to Google cloud were the image is uploaded to and process the text in the image and reflects the data back to the user before sending it to the mongoDB database of the application.

The figure 12. above shows the UI in app for using OCR after user clicks on the use camera button in the upload data tab. After selection of image the processing begins

In this figure 13. above we see the next step after the image is passed through the Goggle Cloud where the text from the image is detected and then reflected on the screen for user.

Next as you can see in figure 14., the data collected by the OCR is associated with the user account that logged in the application and added to the json object which is then send to the server for further processing for the data i.e storing database and adding to the neural network.

6.4 Notifications

This was an additional functionality added to the the version 2.0 of the application. Here the Alarm Manager triggers the service to implement the algorithm everyday and check for the things user will run out. Once the application receives response to itś request, a notifications specifying expired or used items will be sent to the users. This will help the user to keep track of grocery.

6.5 Data Storage

The data captured from the user purchase history is stored in a NoSQL database(MongoDB), this allows to store the data specific to the product as NoSQL databases have a liberal structure and allows different type of data for each entry.

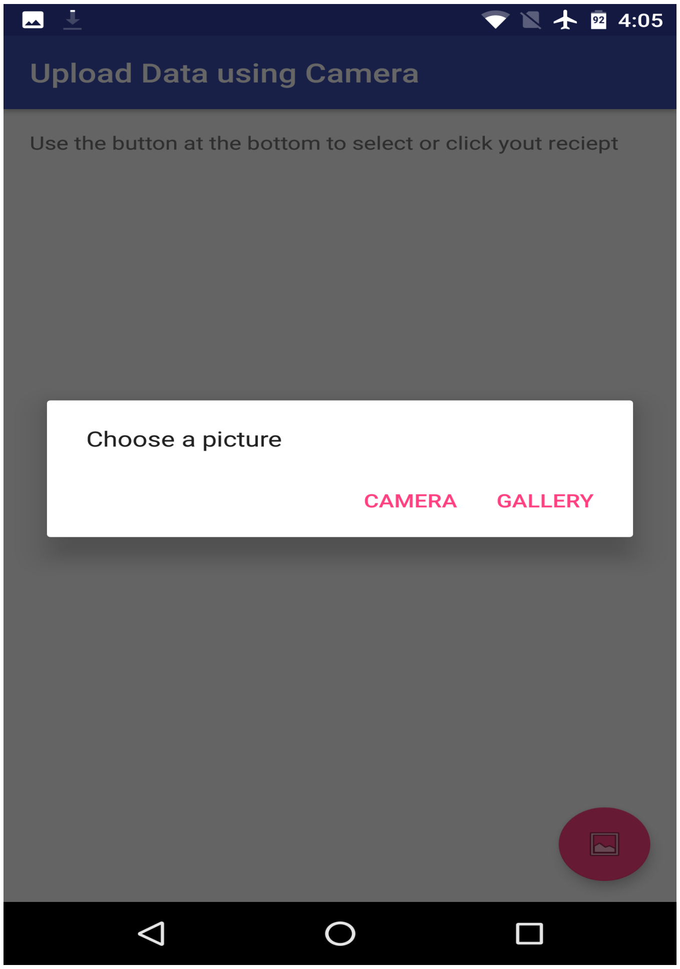
6.6 New list Generation

Once the prediction algorithm is applied to the user history captured previously, the application produces a new set of items that the user is likely to be in need of.

6.7 Continuous Algorithmic Check

Based on the the new list generated by the algorithm and the details of user's recent purchase, a comparison is drawn as to how accurately did the algorithm work. This helps to enhance the algorithm performance and provide more reliable results.

7. EVALUATION PLAN



4:05

Upload Data using Camera

Use the button at the bottom to select or click yout reciept

Choose a picture

Figure 12: The OCR UI

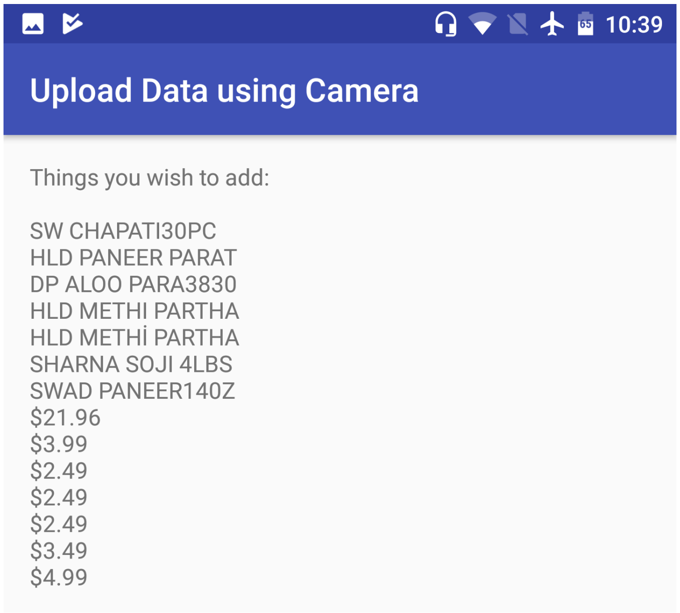
7.1 Parameters

We plan to evaluate this application by giving users a taste of the application on their phones i.e demo test the application and take in their ideas about the application on the basis of following metrics:

1. Reliability
2. Scalability
3. Security
4. Ease of Use
5. Response time of the Application

7.1.1 *Reliability*

The application needs to be reliable with the detection of the content from the camera and enter the proper data in proper fields as well as have good enough recommendation processing to give an accurate time line to the user for when the basic exhaustible supplies might end and give a proper reminder to the user accordingly.



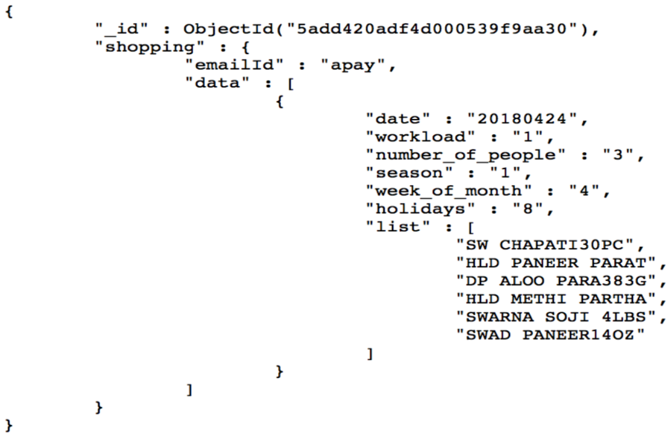
10:39

Upload Data using Camera

Things you wish to add:

SW CHAPATI30PC
HLD PANEER PARAT
DP ALOO PARA3830
HLD METHI PARTHA
HLD METHİ PARTHA
SHARNA SOJI 4LBS
SWAD PANEER14OZ
$21.96
$3.99
$2.49
$2.49
$2.49
$3.49
$4.99

Figure 13: OCR Result



```
{
        "_id" : ObjectId("5add420adf4d000539f9aa30"),
        "shopping" : {
                "emailId" : "apay",
                "data" : [
                        {
                                "date" : "20180424",
                                "workload" : "1",
                                "number_of_people" : "3",
                                "season" : "1",
                                "week_of_month" : "4",
                                "holidays" : "8",
                                "list" : [
                                        "SW CHAPATI30PC",
                                        "HLD PANEER PARAT",
                                        "DP ALOO PARA383G",
                                        "HLD METHI PARTHA",
                                        "SWARNA SOJI 4LBS",
                                        "SWAD PANEER14OZ"
                                ]
                        }
                ]
        }
}
```

Figure 14: json object

7.1.2 *Scalability*

This application is not limited in scope and can be expanded to involve many more users and functions. The more data it receives better will be the results from the neural network training. This application should be able to handle fair number of users and accept requests as well as give give proper push notification on each device at the same time.

7.1.3 *Security*

The application needs to be secure for the user and should not have any threat on the data on the server for the user. Every user has an individual unique account with independent data and needs authentication with user id and password which guarantees security of user data.

7.1.4 *Ease of Use*

The application needs to user friendly and should be easily accessible to user. User should be able to easily navigate through the features of the application. For this we plan to deploy the application which is not done currently. Results from the neural network and the user interface of the application should be up to users satisfaction.

7.1.5 *Response time of the Application*

Owing to the changes done in the Smart Shopping List, the response time of the application i.e. total time to generate a list was calculated for both version 1.0 and 2.0 and it was observed that the application performed much faster in version 2.0 due to the added feature of the OCR.

8. EVALUATION SURVEY

8.1 Evaluation

To determine if the users will be more comfortable using the application with the changes we had thought of, we made the users use our application and got feedback from them in form of suggestion and/or answers to specific questions as listed below:

Question 1: Did you receive proper notification alerts ?

This question was asked to users in order to determine if the algorithm worked correctly and the Alarm Manager generated proper triggers in order for the Broadcast receiver to call the service.

Question 2: How did you find the data output from OCR?

This was to find out if users were getting the output as desired when they scanned the bills for the application to collect data on their purchase history through OCR. This was also to understand how well the data was getting filtered from the bills for the algorithm to produce relevant results when run on the data.

Question 3: How was the app UI ?

Having made several changes to the application UI, we wanted to know if the users were more comfortable using the app now than before.

Question 3: How reliable was the overall application performance ?

The predictive algorithm used in this application is self learning meaning as the data increases, the accuracy of producing the list increases. This question was intended to understand if there was a relative improvement in the results once the data was increased.

8.2 Evaluation Results

The responses to the Evaluation questions helped us understand the user's point of view of the application. We received a total of 16 responses and the distribution of each evaluation question is explained further.

It can be seen from the figure 15 that 62.5% of the users received appropriate notifications from the application which included the list of items that the user is likely to run out of.

We understand from figure 16 that 68.8% of users received absolutely correct output from OCR implementation. 12.5% users received a partially correct output, the reason for which can be different formats that stores use on their bills. This makes the task of filtering out useful data a little tricky.

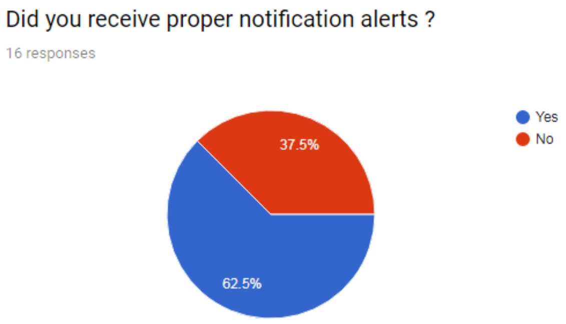


Figure 15: Did you receive proper notification alerts ?

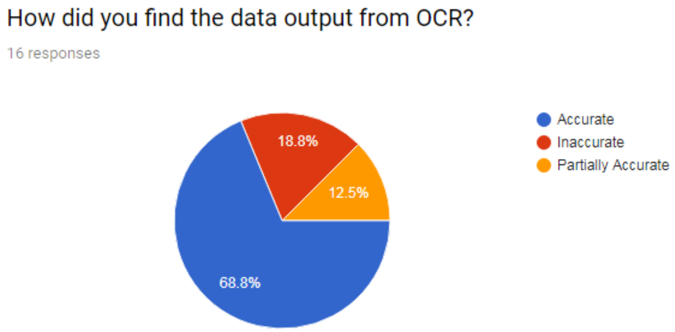


Figure 16: How did you find the data output from OCR?

It is seen from figure 17 that 68.8% users found it comfortable to use the application user interface. This was mainly to test if it was more convenient to enter the shopping details manually or have the bills scanned. And also about no cluttering the application in the application

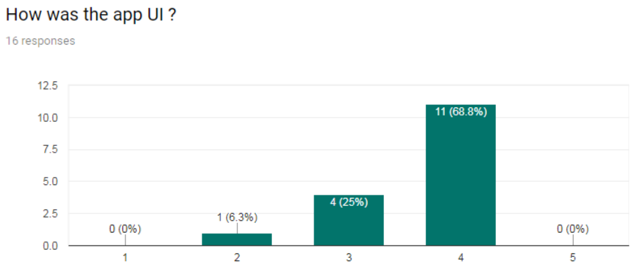


Figure 17: How was the app UI ?

A close to 88% of users who did the evaluation were of the opinion that the application gave accurate results which can be seen from the figure below.

Shown below in figure is the result of task based evaluation done by the users which show the comparison of the time taken by the application to deliver results in version 1 and version 2. We see that average time taken for list generation is around 55 seconds for version 1, while that of version 2 is 16 seconds. This clearly shows that our feature implementation was secure and worked in tandem to greatly

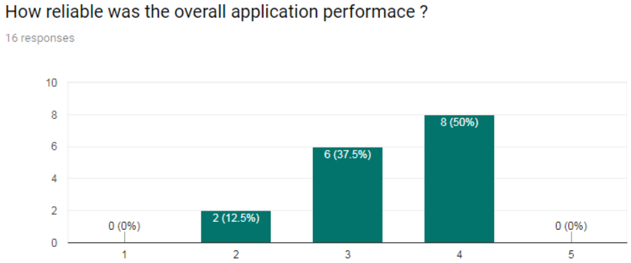


Figure 18: How reliable was the overall application performance ?

enhance the speed of data entry

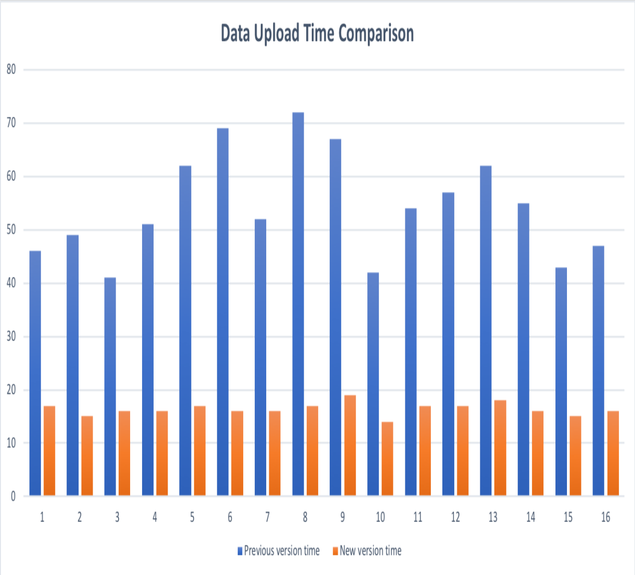


Figure 19: Task based Comparison

Apart from the above questions, there was a subjective question regarding what the user might like to see in the application(an additional feature or change an existing feature). This question was responded to great ideas like budget management for either a family or a group of single users, location based services which over and above giving the list of next shopping, will also give you suggestion on where to buy the items based on your current location. There was also a suggestion to increase the user data significantly so that the prediction accuracy can be increased.

9. FUTURE SCOPE

There is a lot of scope for addition of features in this application. Few more things that can be worked upon:

- Currently the application is restricted as an android application. This could be changed to have a wider userbase as an web application or have independent application for iOS or windows. As the server end would remain same client side can be changed to have a wider audience
- Budget Tracker feature could be added which will give user a idea about his/her expenditure on shopping. This will help user with better budget management for the future
- Bill Split between multiple users when one person pays for the bill in family/friends and the bill could be split between multiple users of the applications.
- Location based suggestions/notifications to the user whenever user enters an particular area application can suggest what generally user buys from there and suggest user about those things

Given the database this application can amass and the recommendation module scope of this application to perform multiple features is high and could be developed.

10. CONCLUSION

In conclusion our project aims at improving the lifestyle of the users by giving them options to have a smart shopping list. This could save user a lot of time and peace of mind of not forgetting anything crucial. This could relieve atleast 1 of the 99 problems a busy person faces in everyday life.The application proves to be useful for the target user base. It has great potential to expand and have more features which can greatly enhance the usability of the application.

11. ACKNOWLEDGMENTS

We would like to express our gratitude to Professor Tim Menzies and the TAs of this course for their encouragement and helpful insights. We appreciate all our friends and classmates who took the time to complete our online survey and project evaluation, enabling us to better understand the scope, requirements and drawbacks of our project.

12. REFERENCES

[1] Predictive shopping notifications by Darin J. Dishneau, Patrick Joseph Derks

[2]Predictive text input in a mobile shopping assistant: methods and interface design by Petteri Nurmi, Andreas Forsblom, Patrik FlorÃl'en, Peter Peltonen, Petri Saarikko.

[3]Consumers' decision-making process and their online shopping behavior: a clickstream analysis by Sylvain Senecal, Pawel J.Kalczynski, Jacques Nantel

[4]Consumers'Tendency to Combine Multipe Shopping Purposes and Destinations by Benedict G. C. Dellaert, Theo A. Arentze, Michel Bierlaire, Aloys W. J. Borgers, Harry J. P. Timmermans

[5] https://books.google.com/ngrams

[6] https://www.twilio.com/docs/sms/api

[7] https://developers.google.com/vision

13. CHITS USED

These were the following chits used for the evaluation:

1. IOS
2. NNS
3. CBY

4. UBE
5. XHD
6. JYP
7. FBR
8. BJA
9. VNJ
10. UZO
11. CUQ
12. YXM
13. JXX
14. EYN
15. MKT
16. VML