Chapter 1: Introduction

1.1 Motivation

The Internet is booming and the state-of-the-art e-commerce has now spread over every field. This boost has a major contribution from the rise of smartphones and their availability for the masses. It has essentially brought the user close to the market because of the facility of providing instant reviews and ratings. There is a multitude of products and services available on various ecommerce websites. Choosing the best among the better products available on the Internet is observed to be an arduous task.

We are proposing an advanced recommendation system which performs natural language processing on the user reviews and gains a sentiment score for various aspects of the product/service. We shall refer to 'product/service' as 'product' hereon. The sentiment score based on user reviews provides a more fine grained result for the aspects rather than coarse grained approaches like collaborative filtering which is purely based on product features. Such classified recommendation can be used to recommend products to users based on their profiles and also gather the negative sentiments about certain aspects to report areas of improvement to the businesses selling the products.

1.2 Problem Definition

Most of the work done in the field of leveraging user reviews to provide personalized recommendations to users has focused on improving the estimations of user ratings of items based on user reviews and other relevant information as a whole. It is often seen that review content is not given much weightage and the focus is on the rating. The rating gives the score for a product as a whole and not the various aspects of the product.

Our proposed method identifies the most valuable aspects of possible user experiences and recommends items together with suggestions to consume the most valuable user-controlled aspects that have been identified as beneficial to the user. When the reviews are studied, a deeper analysis of the aspects of the product is possible. The score is more specific and not binary; the aspects are rated on a wider scale which will provide the system with more accurate insights about the product. Natural Language Processing plays a major role in extracting the sentiments from the review content with respect to various aspects of a product. The aspects are essentially the features or specifications of the product as mentioned on the source of sale.

The Aspect and Review based recommendation system essentially learns the aspects from the review content and builds user profile and product profile for next applications.

1.3 Relevance of Project

The aspect based recommendation system is still a research topic, and there have been various approaches researchers have undertaken to try and improve the accuracy and thoroughness of the system. Our system is made with a rounded perspective that has the primary motive of giving the most relevant recommendations to the user. The subject is very intriguing, mainly because the instinctive medium that humans use for communication is of words and not numbers. Ratings and reviews are the most common form of feedback that customers of a product or service can provide on an online platform. While ratings are quantitative, reviews are expressive. Extracting the users' true sentiments from their review with respect to each aspect is highly insightful.

These insights are leveraged to provide the businesses with the overall sentiment about a product category in the market. The businesses get a holistic view of the demand of the most preferred aspects with respect to a particular product. Temporal patterns can be observed by businesses with breakthroughs in technology. For example when the octa-core processor came into picture, the *processor* aspect of smart phones certainly grabbed the most attention. Similar is the current trend, where the *camera* aspect is turning heads after the dual camera technology for portraits has been adopted as the signature selling point by a maximum of top smartphone brands.

The facts gathered through our system can lead to some fine grained inferences and implications. Each user, with each purchase or review, makes a contribution in building her profile. It helps the user choose, not the best product in the market with a general viewpoint, but the product that is right for her.

1.4 Methodology Used

- The first step is to extract the aspect-sentiment pairs using an ‘Opinion Parser’.
- The Opinion parser does two things viz. aspect extraction and aspect-sentiment classification.
- The aspect extraction step uses Double Propagation Algorithm and the aspect sentiment classification is based on set of sentiment expression, also called sentiment lexicons, and context analysis.
- The method is focused on the user-controlled aspects.
- Then builds item and user profiles.
- The product is recommended on the basis of aspects which can be categorized as popular aspects, most positive aspects and Most Negative aspects.

- Thus, in a nutshell, we take user reviews and ratings, extracts aspects and classify sentiments on the aspects in the user reviews and recommend items together with the most important aspects that may enhance the user experience with the items.

Chapter 2: Literature Survey

The literature survey involved looking for relevant IEEE papers published by students or well renowned authors. These papers played a role in giving us a guideline for our approach toward the problem and an overall depth of the problem and its extent. It also helped us to get an idea about the extent to which the problem has been perused and dealt with. We were also able to define the scope up to which our proposed system will take care of the issue. The challenges faced by the authors of these papers gave insights and will help us to plan mitigation strategies if we face similar challenges in the implementation. The key words or phrases we used to find the most relevant research papers are Natural Language Processing, Sentiment analysis, Opinion parsing, Aspect Extraction, Sentiment Extraction.

2.1 Papers

1. ***Author:*** *Li Chen, Guanliang Chen and Feng Wang*
   `***Title:*** *Recommender systems based on user reviews: the state of the art*
   Published in: *June 2015, Volume 25, Issue 2, pp 99–154*

   a. This survey classifies state-of-the-art studies into two principal branches: review-based user profile building and review-based product profile building.
   b. In the user profile sub-branch, the reviews are not only used to create term-based profiles, but also to infer or enhance ratings. Multi-faceted opinions can further be exploited to derive the weight/value preferences that users place on particular features.

2. ***Author*****:** *Yu Zhang, RuiFang Liu, AoDong Li*
   Title: *A Novel Approach to Recommender System Based on Aspect-level Sentiment Analysis*
   Published in: *4th National Conference on Electrical, Electronics and Computer Engineering (NCEECE 2015)*

   a. In this paper, a novel approach to introduce aspect-based sentiment analysis into recommender systems is proposed. The aspect of the product using the topic model is extracted and then the aspect-specific sentiment words are identified using the SentiWordNet (a sentiment lexicon).
   b. The use the result of sentiment analysis is then used to make user interests model and the product model. By comparing two models of each user-product pair, we obtain the similarity of the user's interest and the product.

3. ***Author:*** *Deepali Virmani, Vikrant Malhotra and Ridhi Tyagi*
   Title: *Aspect Based Sentiment Analysis to Extract Meticulous Opinion Value*

Published in: *IJCSIT, MAY 2014*

  a. This paper proposes an algorithm to implement aspect level sentiment analysis. The algorithm takes input from the remarks submitted by various teachers of a student.
  b. A lower aspect value signifies a general remark whereas a higher aspect value represents specific remarks about an aspect. The polarity of remarks is given by the opinion value.

4. ***Author:*** *Niharika Purbey, Aishwarya Iyer*
   Title: *Categorization and Analysis of Yelp restaurant*
   Published in: *arxiv 1709.08698*
   a. This paper focuses on the yelp dataset, which is a general and broad dataset for reviews of restaurant, bars, hair salons, stores, etc that suit the users' taste and preferences.
   b. Sentiment analysis is applied to determine the sentiment associated with each topic to obtain a granular breakdown of a business' strong and weak suits.

5. ***Author*** *:Warih Maharani, Dwi H. Widyantoro, Masayu Leylia Khodra*
   Title*: Syntactic pattern based aspect extraction*
   Published in *: Procedia Computer Science Volume 59 , 2015*
   a. There are two approaches for opinion mining, traditional approach and aspect based approach. This paper focuses on the use of aspect based opinion mining. It states the stages in aspect based opinion mining viz. Extraction of aspects and opinions and Polarity of classification.
   b. This research focuses aspect extraction by comparing some patterns that represent an aspect.

6. ***Author*** *: Konstantin Bauman, Bing Liu et al*
   Title *: Aspect Based Recommendations: Recommending Items with the Most Valuable Aspects Based on User Reviews*
   Published in *: KDD '17 Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining , August 2017*
   a. The sentiment analysis system proposed in this paper performs two key functions, aspect extraction and aspect sentiment classification. Aspect extraction aims to extract sentiment targets on which some sentiments have been expressed.
   b. These targets are usually different aspects of entities (e.g., products or services), which are items in our context. Aspect sentiment classification classifies whether the sentiment expressed on an aspect is positive, neutral, or negative.

c. The main advantage of the proposed SULM model is the new additional functionality of providing not only recommendations of items to users, but also recommendations of the most valuable aspects that may enhance user experiences with items.

7. ***Author:*** *S.K. Bharti, B. Vachha, R.K. Pradhan, K.S. Babu, S.K. Jena*
   Title: *Sarcastic sentiment detection in tweets streamed in real time: a big data approach*
   Published in: *DCN Volume 2, Issue 3, August 2016, Pages 108-121*
   a. Sarcasm sentiment analysis is a rapidly growing area of NLP with research ranging from word, phrase and sentence level classification to document and concept level classification.
   b. Research is progressing in finding ways for efficient analysis of sentiments with better accuracy in written text as well as analyzing irony, humor and sarcasm within social media data.

8. ***Author:*** *Chinsha T C and Shibily Joseph*
   Title:*Aspect-Based Opinion Mining and Recommendation System for Restaurant Reviews*
   Published in: *ACM RecSys '14*
   a. A product/service that has a higher average review or rating usually gets picked against a similar product/service with less favourable reviews. Reviews usually have an overall rating, but most of the times there are subtexts in the review body that describe certain features/aspects of the product.
   b. This demonstration presents a system that extracts aspect-specific ratings from reviews and also recommends reviews to users based on their and other users' rating patterns.

9. ***Title:*** *Marie-Catherine de Marneffe and Christopher D. Manning*
   Author: *Stanford typed dependencies manual*
   a. Stanford typed dependencies represent grammatical relationships, in a given sentence, in a simple manner.
   b. Definitions of various typed dependencies were used by us in the project to form the rules for aspect-sentiment extraction process.

Chapter 3: Requirement Analysis

The process of the project is important, but unless the requirements of a system are not well defined, the project can hardly move ahead. Even though it is often observed that the requirements of a project may change over the course of the project, the functioning of the project and the team is much smoother if the objective of the project is clear. The requirements are essentially categorized in functional and nonfunctional requirements. The functional requirements provide details about the outputs of a system and all features and functionalities which the system implements. The nonfunctional requirements are more general which consist of the criteria based on which a system can be judged. The nonfunctional requirements often give an idea about the system's performance after deployment and these can be used to ensure that the project is up to the mark with the current systems and trends.

3.1 Functional Requirements

A. Customers

- Recommendation of products based on customer interests.
- Preference based product ranking.
- Notifications related to new developments based on user interests.
- User/product profile building.

B. Businesses

- Overall sentiment towards product in market.
- Suggested areas of improvement for product.
- Current trends in the market.

3.2 Non-functional Requirements

- **Accuracy:** The system aims at achieving more accuracy in recommendations as compared to the existing, purely feature based recommendation systems.
- **Robustness:** Ability to handle large datasets with minimal slowdown or crash.
- **Usability:** Easy to understand user interface, with minimalistic and well spread-out features and a responsive layout.
- **Scalability:** Ability to extend the system to a distributed environment.

3.3 Constraints

- It is necessary that the users write reviews using proper grammar.
- The punctuation and spacing should be perfect in the review.
- The reviewer must write only correct spellings of words to make sure that they are considered as aspects.
- Lack of multi-lingual support.

3.4 Software Requirements

- Python version >= 2.7.12
- Stanford CoreNLP Parser for POS tagging, Dependency Parsing and Coreference Resolution.
- pycorenlp library to access CoreNLP Server APIs from Python.
- textblob and vaderSentiment libraries to calculate intensity of sentiments words on the scale of -1(most negative) to +1(most positive).
- Natural Language Toolkit (NLTK) to use wordnet for finding anonym of given word.
- VaderSentiment to parse sentiment intensity of aspects.

Chapter 4: Proposed Design

4.1 Block Diagram

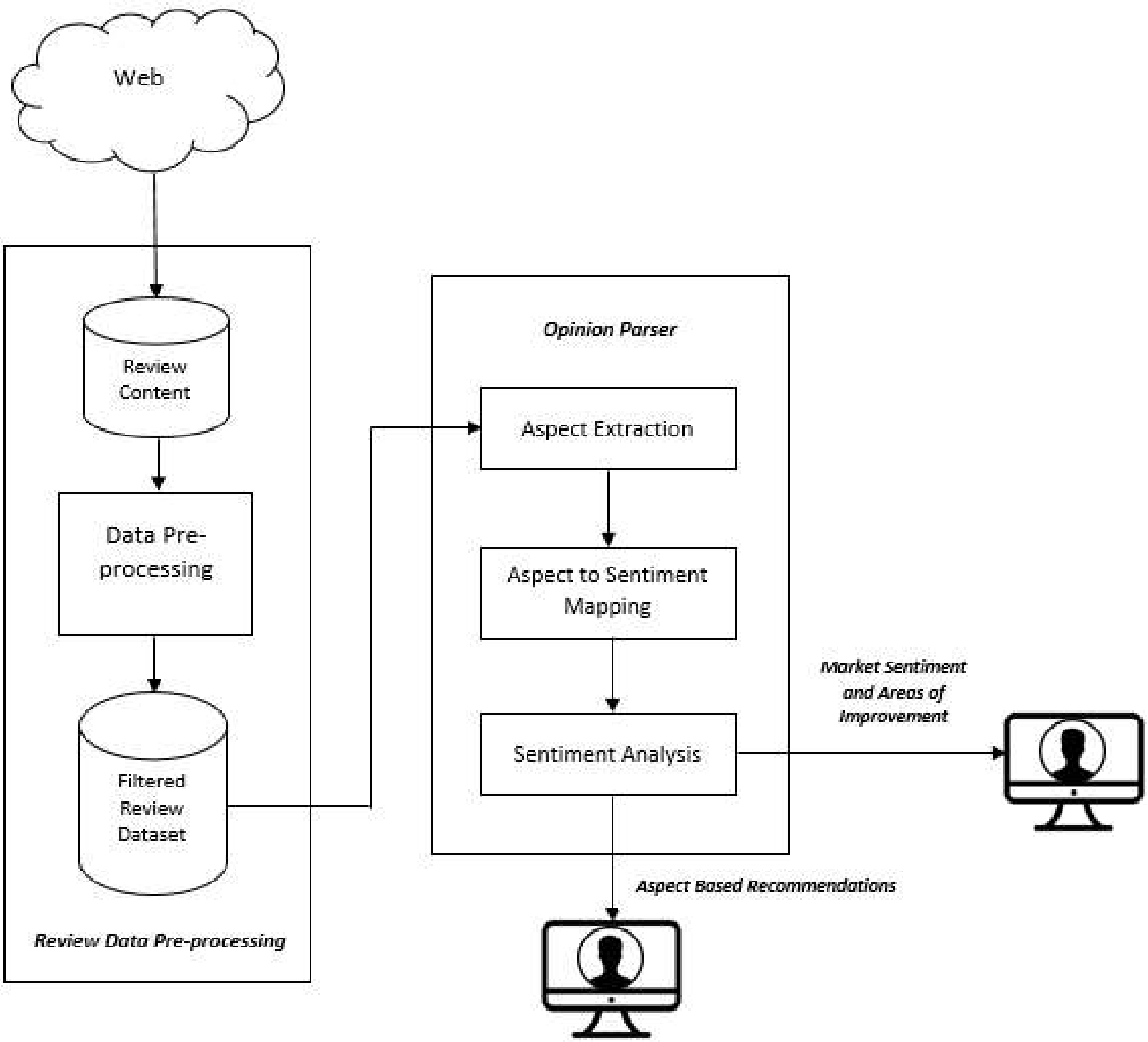


Fig 1. Block Diagram of Aspect and Review Based Recommendation System

4.2 Modular Design

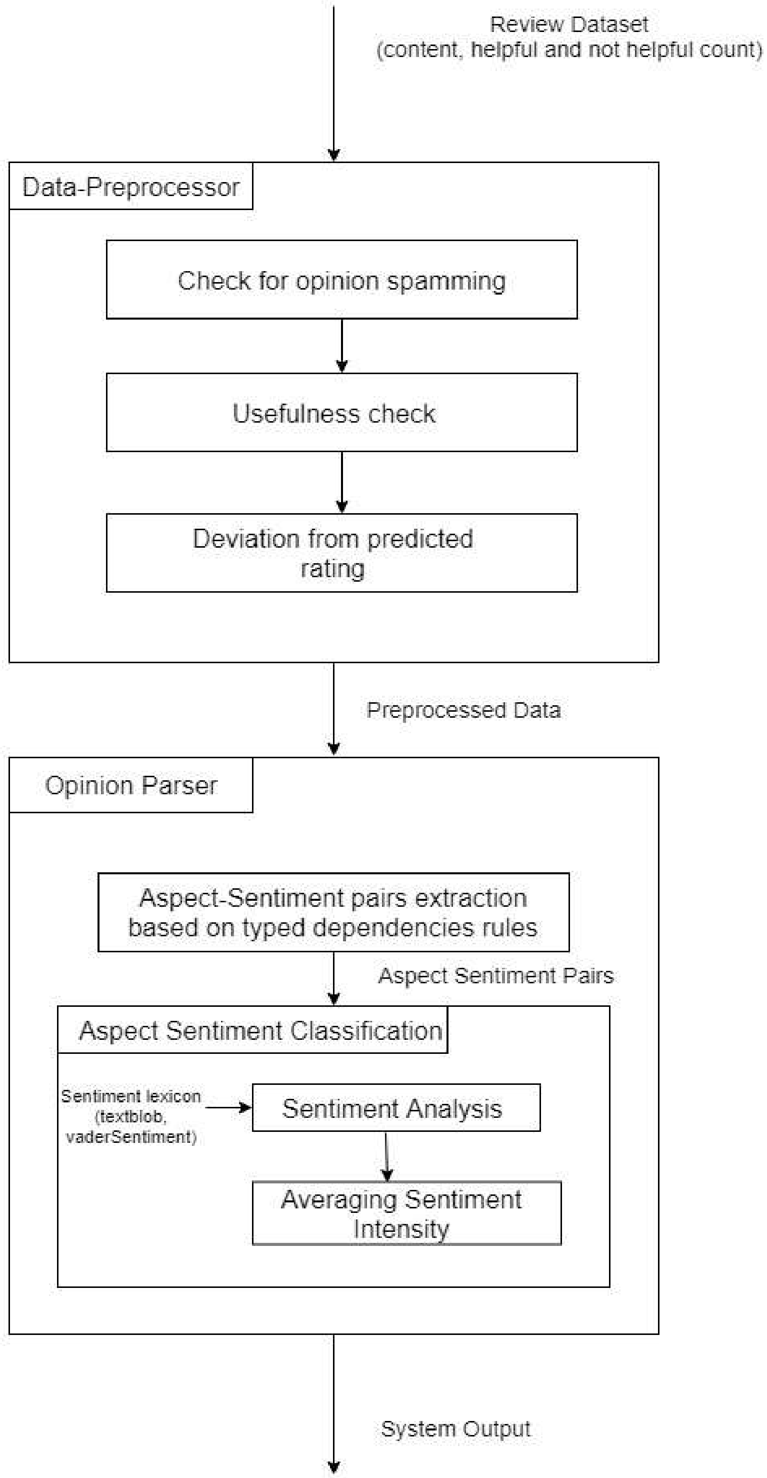


Fig 2. Modular Design

Data preprocessor:

1. ***Check for opinion spamming:***
    a. Parse the content of all reviews for the same product.
    b. Parse the content of reviews for similar selective products.
    c. If any reviews are found to have exactly matching content, mark all such reviews as irrelevant.
    d. Set their weightage for the recommendation system to 0.
    e. Blacklist the spam reviewers.
    f. *Example: If the review has the following content:* ***"This phone is really good. Please buy this phone! It's has the best touch screen and the camera is awesome!"****, this content is compared with reviews on the same product and with the content of reviews on similar products. If the content with other reviews matches, all such reviews are marked irrelevant and their weightage is set to 0, also, the users who wrote these reviews will be added to a blacklist and other reviews by these users will be marked unimportant.*

2. ***Helpfulness check:***
    a. Obtain the number of people who have viewed the review, call it 'v'.
    b. Obtain the number of people who have found the review useful, call it 'u'.
    c. Calculate the ratio of u to v.
    d. Approximate the weight according to this ratio to one decimal place.
    e. *Example: If the review has a high votes-to-views ratio, say a review has been marked helpful by* ***200 users*** *and not helpful by* ***150 users****, the ratio is* ***1.33****. This number is compared to a threshold number,* ***0.6****, which, if the number fails to exceed, the review is discarded.*

3. ***Deviation from predicted rating:***
    a. The adjectives extracted from the review are scored on a scale of 1 to 5.
    b. An overall expected rating associated with the review is calculated.
    c. This calculated rating is compared to the actual rating.
    d. The weight of the review is adjusted on the basis of the deviation of the calculated rating from the actual rating.
    e. *Example: If a review has all positive sentiments, like* ***"This is the best phone in this price range. The screen is very smooth and responsive. The clarity and vividity of the camera is amazing and the processor can handle heavy apps."****,*

and the entered rating of the phone is 2.5/5, the rating is not in sync with the review content. Such reviews are discarded.

Opinion Parser:

Opinion Parser is a heart of the system. It takes preprocessed data from data preprocessor and produces output which contain recommendations (for users) and overall product sentiment(for businesses).

1. Bootstrapping process for aspect extraction

a. Bootstrapping is an opinion words and opinion target (aspect) extraction process.
b. In this process a set of opinion words like "*good*", "*bad*", "*amazing",*called as Opinion lexicon, is given as an input to the bootstrapper.
c. This Opinion lexicon is used by the bootstrapper to identify opinion words from the reviews.
d. Initially bootstrapper uses initially provided Opinion Lexicon to identify opinion words in the reviews. It then extracts corresponding aspects and forms <aspect, sentiment> pairs.
e. Known Opinion lexicon and extracted opinion words and target (aspects) are then used together to further extract opinion words and targets. Subtasks included in this process are:
    1. extracting targets using opinion words
    2. extracting targets using extracted targets
    3. extracting opinion words using extracted targets
    4. extracting opinion words using both the given and the extracted opinion words
f. This process goes on till no opinion words and target are left to be extracted.

2. Aspect Sentiment Classification

a. In this step, intensity of each sentiment of each aspect is calculated using TextBlob and vaderSentimentAnalyazer. Intensity is in the range of (-1,+1) with -1 being most negative and +1 being most positive.
b. Each aspect is assigned a sentiment score which is average of all sentiment values of sentiments of that aspect.
c. Then rating is scaled for each aspect from (-1,+1) to (0,5).

Bootstrapping using double propagation algorithm:

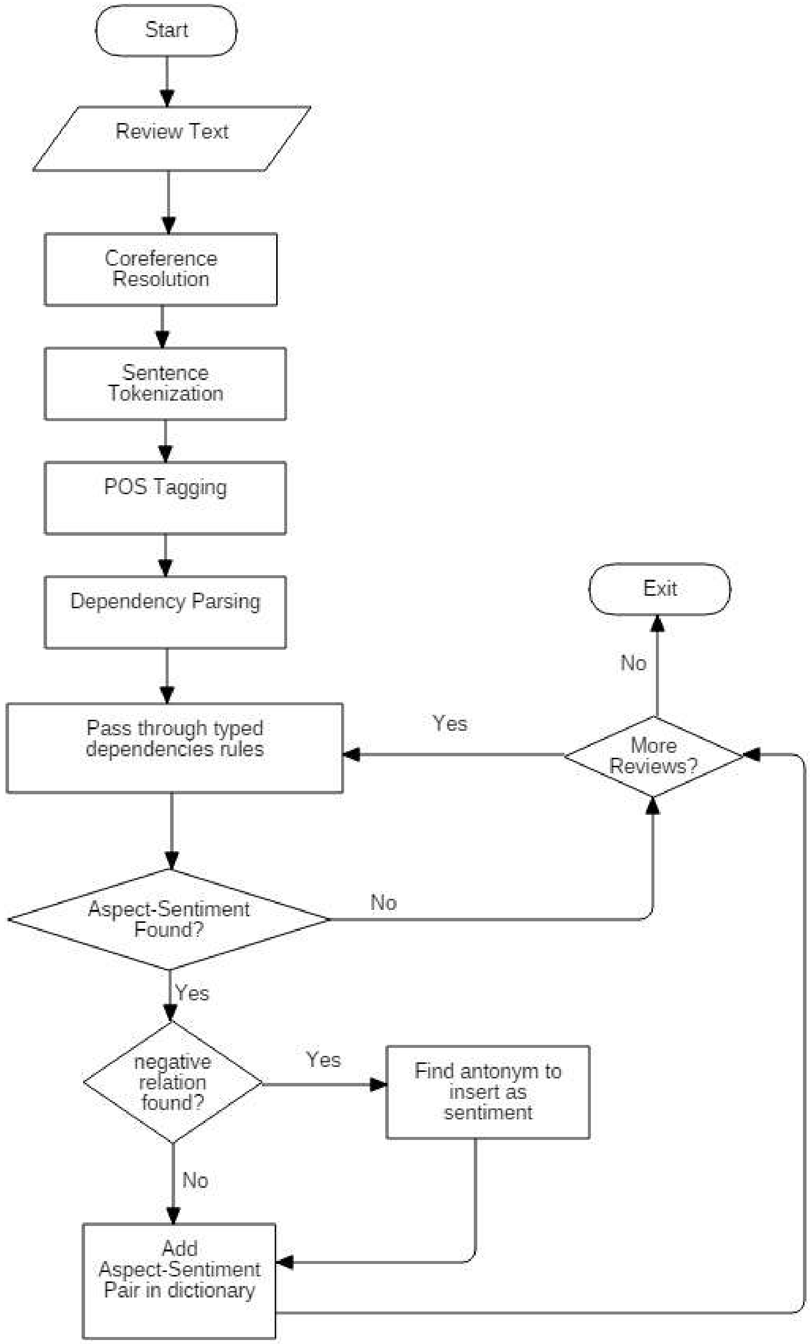


Fig 3. Flowchart of Bootstrapping

4.3 Detailed Design(DFD)

Level 0

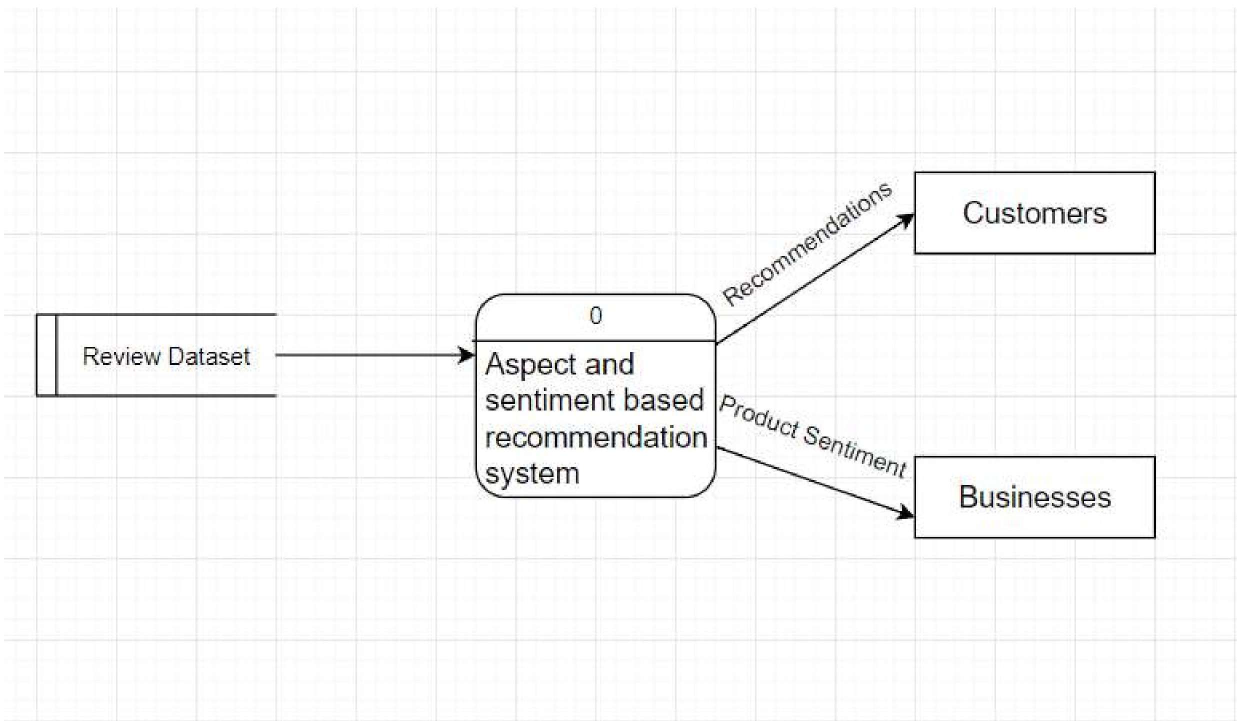


Fig 4. Level 0 DFD

Level 1

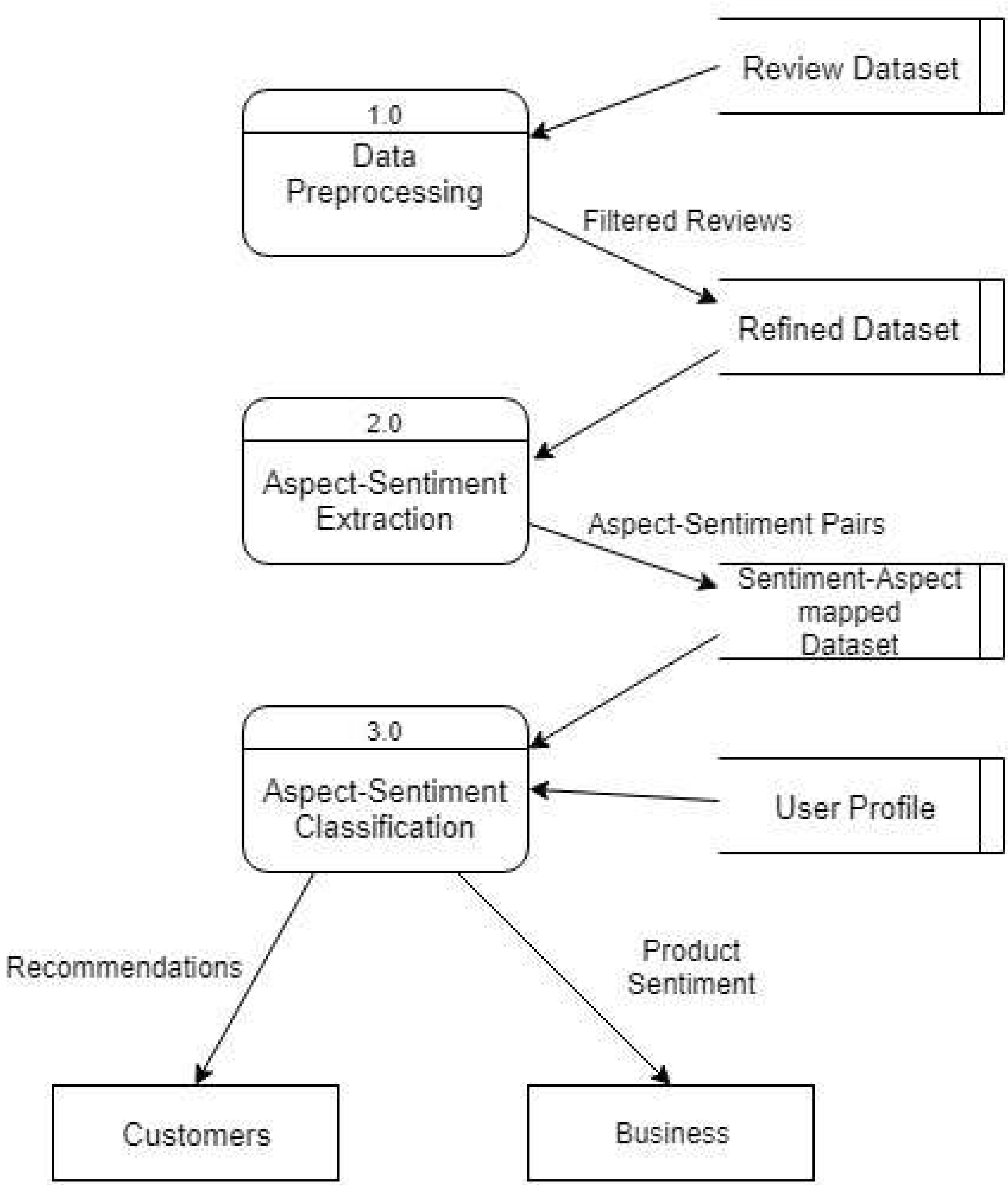


Fig 5. Level 1 DFD

Level 2

A) Data Pre-processing

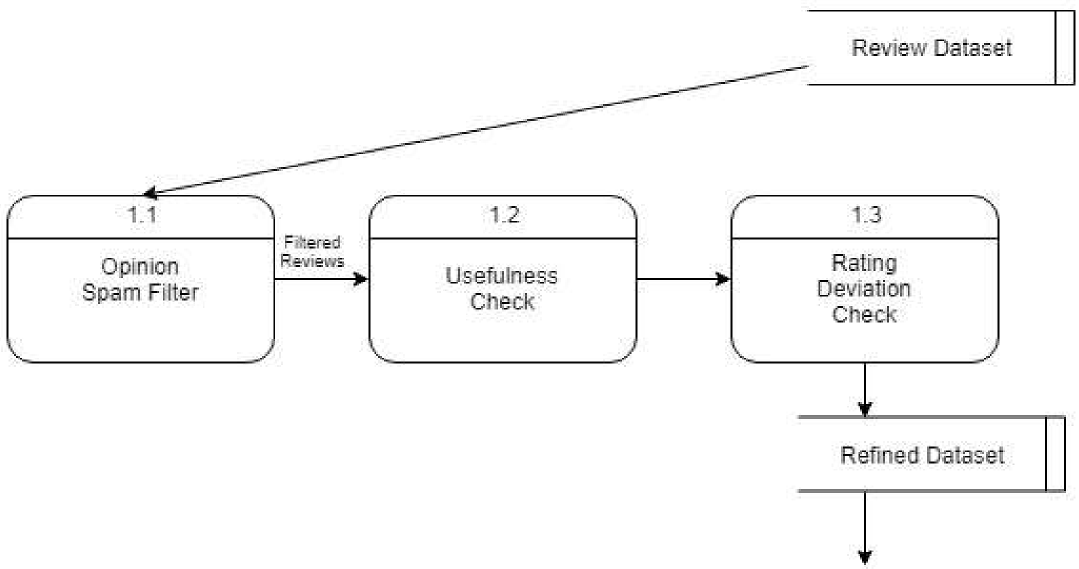


Fig 6.1: Level 2 DFD - Data Preprocessing

B) Aspect Sentiment Extraction

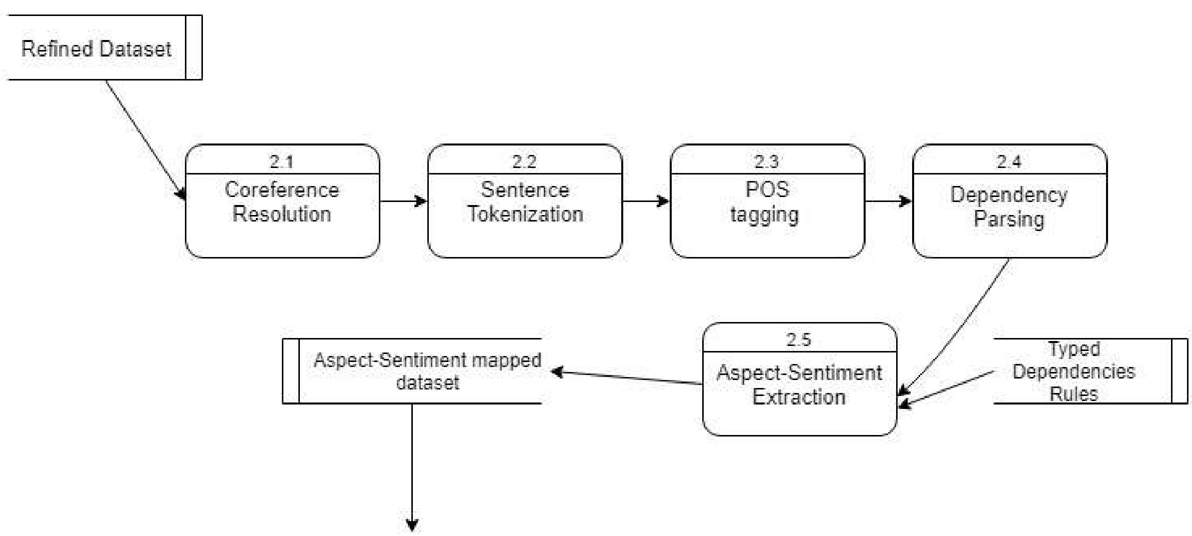


Fig 6.2: Level 2 DFD - Bootstrapping

C) Aspect sentiment classification

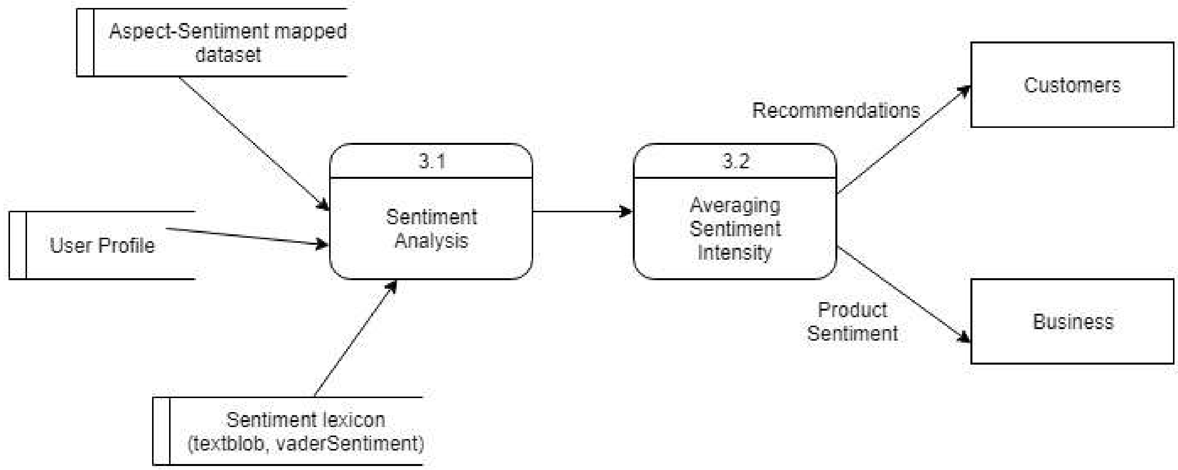
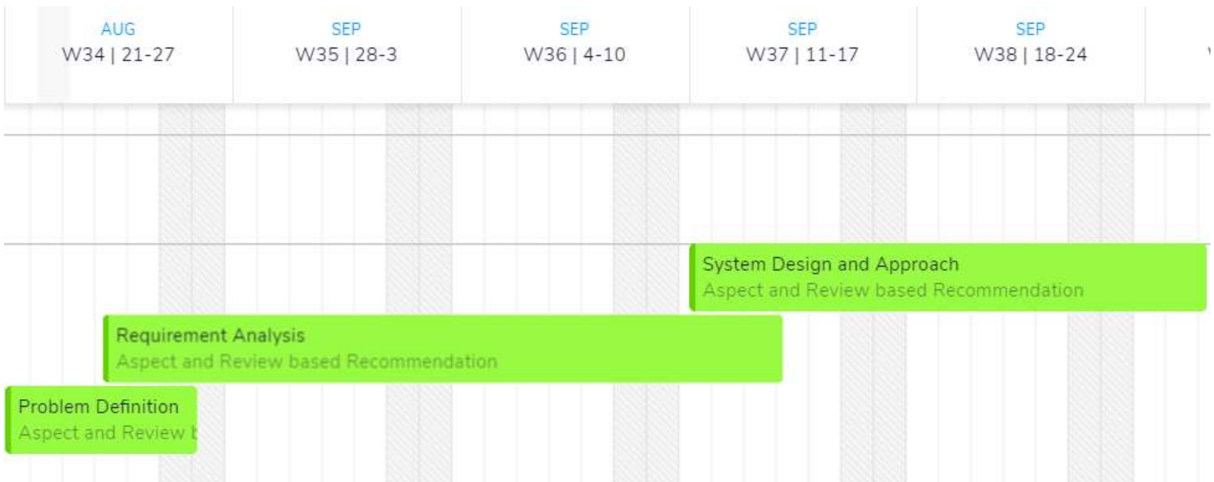


Fig 6.3: Level 2 DFD - Aspect Sentiment Classification

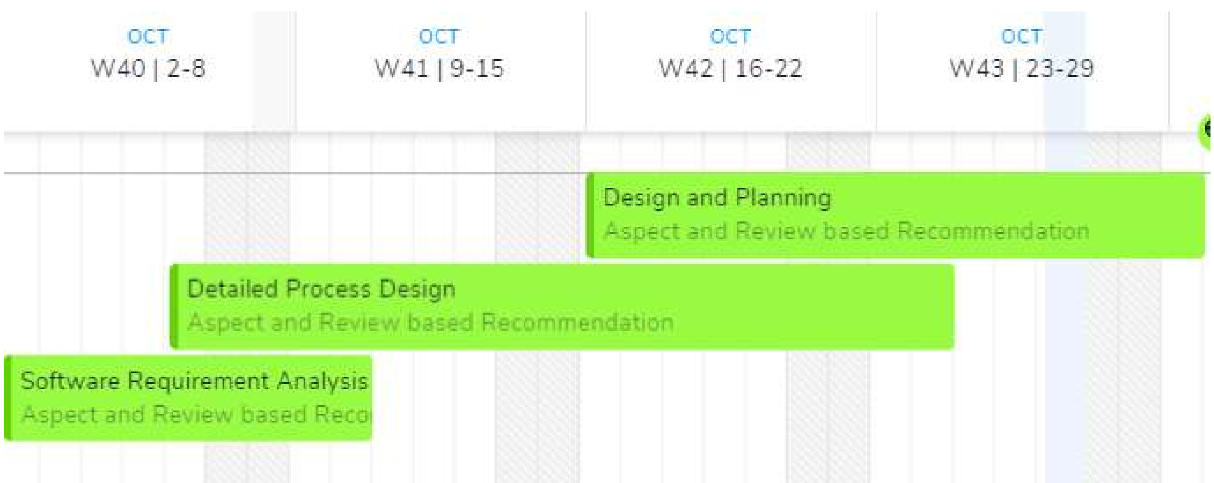
4.4 Project Scheduling and Tracking

PHASE I:

August 2017-September 2017:



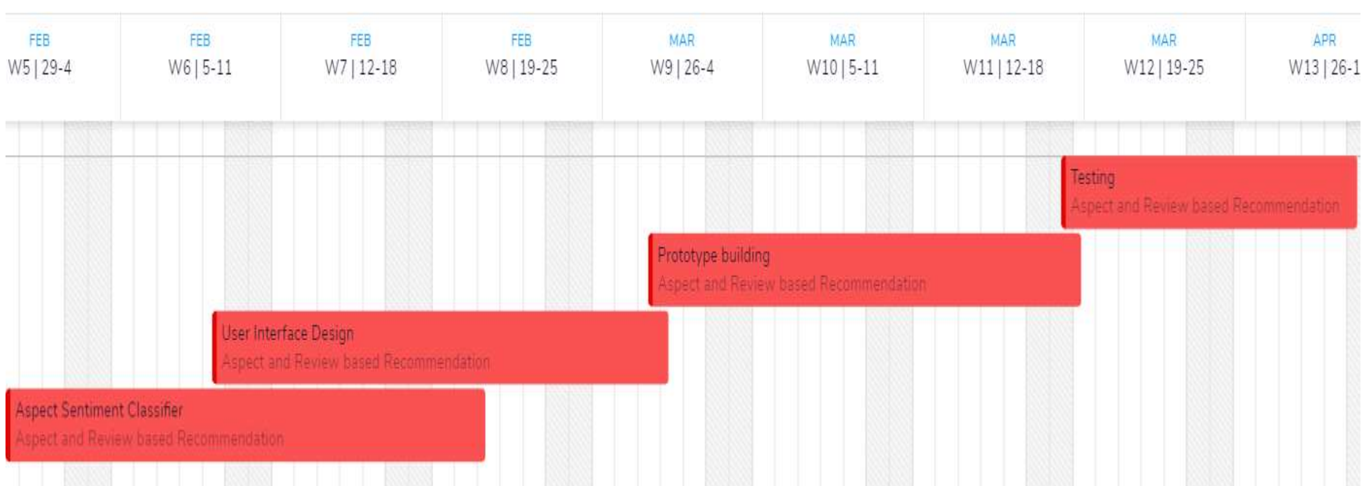
October 2017:



PHASE II:

December 2017 - January 2018:

February 2018 - March 2018



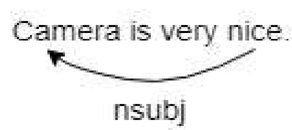
Chapter 5: Implementation Details

5.1 Algorithms for the respective modules developed

The Stanford typed dependencies representation was designed to provide a simple description of the grammatical relationships in a sentence that can easily be understood and effectively used by people without linguistic expertise who want to extract textual relations. In particular, rather than the phrase structure representations that have long dominated in the computational linguistic community, it represents all sentence relationships uniformly as typed dependency relations. That is, as triples of a relation between pairs of words, such as "the subject of distributes is Bell." Our experience is that this simple, uniform representation is quite accessible to non-linguists thinking about tasks involving information extraction from text and is effective in relation extraction applications.

5.1.1 Typed Dependencies Rules:

1. ***Nominal Subject (nsubj) Rule*:**

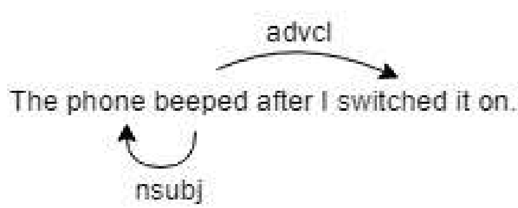


When the sentence has a copular verb, the **nsubj** dependency has an adjective **governor** and noun **dependent.** Hence, here the aspect is the dependent and the sentiment is the governor.

2. ***Adjectival Modifier (amod) Rule:***

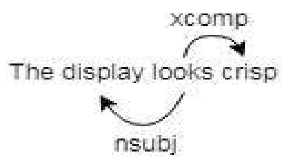
When the sentence has a copular verb, the **amod** dependency has an noun **governor** and adjective **dependent.** Hence here the aspect is the governor and the sentiment is dependent.

3. ***Adverbial Complement (advcl) Rule:***



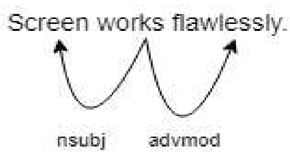
When the **governor** is a non-copular verb for **nsubj** dependency as well as **advcl** dependency, the **dependent of nsubj** is the **aspect** and the **dependent of advcl** is the **sentiment.**

4. ***Nominal Subject + complement (acomp/xcomp) Rule:***



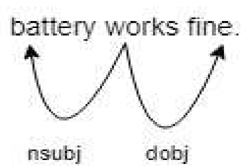
When the **governor** is a non-copular verb for **nsubj** dependency as well as **xcomp/acomp** dependency, the **dependent of nsubj** is the **aspect** and the **dependent of xcomp/acomp** is the **sentiment.**

5. ***Adverbial Modifier (advmod) Rule:***



When the **governor** is a non-copular verb for **nsubj** dependency as well as **advmod** dependency, the **dependent of nsubj** is the **aspect** and the **dependent of advmod** is the **sentiment.**

6. ***Direct Object (dobj) Rule:***

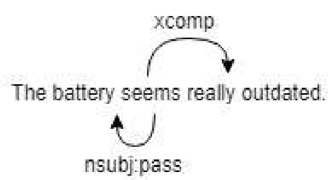


When the **governor** is a non-copular verb for **nsubj** dependency as well as **dobj** dependency, the **dependent of nsubj** is the **aspect** and the **dependent of dobj** is the **sentiment.**

7. ***Passive Nominal Subject (nsubjpass) Rule:***

When the sentence has a copular verb, the **nsubj** dependency has an adjective **governor** and noun **dependent.** Hence, here the aspect is the dependent and the sentiment is the governor.

8. ***Passive Nominal Subject (nsubjpass) + complement (acomp/xcomp) Rule:***



When the **governor** is a non-copular verb for **nsubj:pass** dependency as well as **xcomp/acomp** dependency, the **dependent of nsubj:pass** is the **aspect** and the **dependent of xcomp/acomp** is the **sentiment.**

9. ***Relative Clause Modifier (acl:relcl) + complement (acomp/xcomp) Rule:***

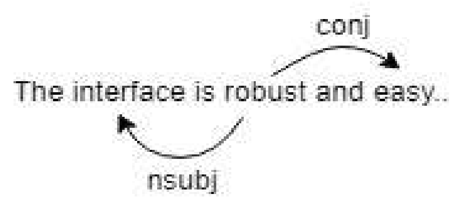
When the **dependent** is a non-copular verb for **acl:relcl** dependency as well as **xcomp/acomp** dependency, the **governor of acl:relcl** is the **aspect** and the **dependent of xcomp/acomp** is the **sentiment.**

5.1.2 Universal Rules:

Some rules are such that they need to be associated with each independent rule as they are the most general rules. We have implemented three such rules, namely the *conjunction* rule, the *compound* rule for bigrams and the *negation* rule. These rules are common across the primary rules and their module needs to be implemented as a function only once. This function is called regardless of which rule is triggered by the conditions satisfied. The description of the rules is as follows:

1. ***Conjunction Rule:***

The conjunction rule is applicable when a review text contains multiple sentiment words associated with a single aspect.



In the above example, the conjunction rule is associated with the simple *nsubj* rule for copular verbs. The basic nsubj rule will only associate the adjective ***robust*** with the aspect ***interface***. It will not understand that the adjective ***easy*** is also associated with the same aspect. This is where the *conj* dependency plays a pivotal role. Every word following ***robust*** which is separated by commas or conjunctions has a *conj* dependency with the word ***robust***. In this case, the word ***easy*** is associated with ***robust*** with the *conj* dependency. Hence, ***easy*** is also associated with the aspect ***interface*** as its sentiment.

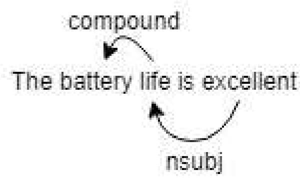
2. ***Negation Rule:***

The negation rule is necessary because sometimes inverting terms are associated with the sentiment words. In such a case, a word of the opposite meaning of the negated sentiment word is added to the sentiment for the aspect.

In the above example, the negation rule is associated with the simple nsubj rule. Here, the aspect ***processor*** is associated with the sentiment ***good***. Actually, the reviewer has written an inverting term ***not***, before the sentiment word. If it was not for the *neg* dependency, the aspect-sentiment pair would be ***processor-good*** which is completely the opposite of what the reviewer means. The negation rule will help extract the true meaning of what the user has written. It looks for a word closest to the opposite polarity of the extracted sentiment word. This is done by using the getAntonyms() function of the textblob package. In this example, the word having the closest of the opposite polarity of ***good*** is ***bad***. Hence, the word ***bad*** will be added as the sentiment of the aspect ***processor***.

3. Compound Rule:

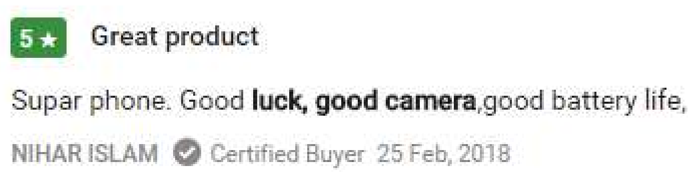
If multiple words are playing a role in naming the aspect or sentiment, both the words play a role in constructing the aspect-sentiment lexicon.



Here, the Compound rule is associated with the simple nsubj rule. The simple nsubj rule extracts the noun ***life*** as an aspect and the adjective ***excellent*** as the sentiment. The aspect ***life*** is completely irrelevant to the product since the reviewer was actually talking about the ***battery life***. Now, the Compound rule checks whether the extracted aspect is the governor for a *compound* dependency. If such a dependency is found, the dependent of this dependency is appended with the governor, and the bigram is now considered to be an aspect as a whole. Hence, the sentiment is ***excellent*** and the aspect is ***battery life***.

5.2 Comparison Analysis with existing systems

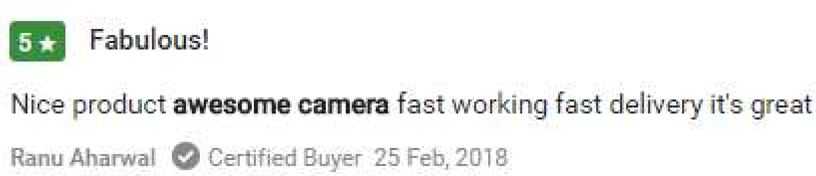
FLIPKART



ASPECT BASED RECOMMENDATION SYSTEM

Camera:[“good”]
Battery:[“good”]

FLIPKART



ASPECT BASED RECOMMENDATION SYSTEM

Camera:[“awesome”]

FLIPKART

5★ Mind-blowing purchase

Nice mobile
Good camera portrait **photography**
Battery also good
I prefer to buy this

Ñaveen Tej Certified Buyer 3 Mar, 2018

ASPECT BASED RECOMMENDATION SYSTEM

Camera:[“good”]
Overall:[“nice”]
Battery:[“also good”]

FLIPKART

5★ Just wow!

I've fallen in love!
R- Raging Speed
E- Excellent Performance
D- Dynamic Display
M- Metal Build
I- **Incredible Battery Life**

N- Noteworthy
O- Outstanding
T- Top Notch Specs
E- Ergonomical

5- Fabulous, Fantastic, Furious, Futuristic, Fantabulous

akshat Kumar Certified Buyer 3 Mar, 2018

ASPECT BASED RECOMMENDATION SYSTEM

Processor:["raging","excellent"]
Display:["dynamic"]
Battery:["incredible"]
Overall:["top notch"]

FLIPKART

5 ★ Classy product

Good performance, Battery back up is good, Excellent front camera, Best features at this Price Range

Flipkart Customer Certified Buyer 26 Feb, 2018

ASPECT BASED RECOMMENDATION SYSTEM

Processor:["good"]
Camera:["excellent","front"]
Overall:["best"]

FLIPKART

5 ★ Awesome

Really a good mobile under this budget. No other mobile can beat this. **Good performance, good processor, battery** losts a full day. But the only disappointment is, this runs on Nougat & available in flash sale only.

YADIDYA Certified Buyer 16 Mar, 2018

ASPECT BASED RECOMMENDATION SYSTEM

Processor:["good"]

FLIPKART

5 ★ Highly recommended

Best Camera,Best Processor,**Beast Battery**,what else u want??

Nishant Certified Buyer 24 Feb, 2018

ASPECT BASED RECOMMENDATION SYSTEM

Camera:[“best”]
Processor:[“best”]

FLIPKART

5 ★ Really Nice

Pros: 1 : good camera quality in this price segment and front camera is amazing,
2: good bettery life ,
3: good **crispy display**,
4: front side design is pretty good but back side is traditional mi style,
5: powerful processor no lags .
6: face unlock available in OTA update.
Cons: 1: hybrid SIM slot.
2: Ram management was not so good.
Overall this phone is amazing in this price segment.

UDAYAN MONDAL Certified Buyer 24 Feb, 2018

ASPECT BASED RECOMMENDATION SYSTEM

Camera:[“good”]
Display:[“good”,”crispy”]
Processor:[“powerful”]

FLIPKART

5 ★ Great product

awesome phone.best camera,ultymate **design.super display.totaly** fully packed devise.

Dinesh P Certified Buyer 23 Mar, 2018

ASPECT BASED RECOMMENDATION SYSTEM

Overall:[“awesome”]
Camera:[“best”]
Display:[“super”]

5.3 Evaluation of the developed system (Accuracy, Effectiveness, Efficiency)

We parsed 50 reviews to measure the **accuracy**. The accuracy is calculated by *aspects extracted/aspects available in review*. There were a total of 108 aspects which were relevant to our defined aspects. But our system extracted 92 aspects in all.

Therefore, the accuracy is given by

(92/108)*100 = 82.14%

The **effectiveness** of the system is evident by the fact that it uses an all-rounded parser like the StanfordCoreNLP and a well-defined rule set which abides by the typed dependencies of the StanfordCoreNLP parser. We used a dictionary based approach to fetch only the relevant aspects. When the dictionary based approach extracts aspects only based on the dictionary, it covers almost all the relevant aspects, but there is a certain amount of limitation to which the aspects are extracted. Our system works like an expert system and not an intelligent system and hence needs a Knowledge Engineer to make sure that new words or phrases are regularly appended to the dictionary. This is necessary because we have not integrated neural networks or learning algorithms with the system.

The **efficiency** of our system lies in the fact that the primary focus of our system is on text data. The data structure that we have used primarily is dictionary. Time complexity of finding an element in a dictionary is O(1). We store aspect as the key of the dictionary so all the frequent searches of aspects in aspect-sentiment dictionary are of O(1). Plus, in the algorithm of finding aspect-sentiment pairs we have made a dictionary of dependencies which has dependency name as a key and list of dictionary of governor and dependents as value. It reduces the complexity of finding dependency (to apply appropriate rule) from O(n) to O(1) where ‘n’ is number of dependency relations found for the given sentence(review).

Chapter 6: Testing

In the initial development stages, all the rules required for the aspect sentiment extraction were developed on the PyCharm IDE. We tested our rules using some sample sentences like the one given below :

```
The display looks crisp. screen works flawlessly. battery works fine. camera is nice. phone has lovely camera.
```

Corresponding aspect-sentiment dictionary is received as an output

```
C:\Users\abk16\AppData\Local\Programs\Python\Python36-32\python.exe C:/Users/abk16/PycharmProjects/untitled1/test_new.py
{'display': ['crisp'], 'screen': ['flawlessly'], 'battery': ['fine'], 'phone': ['camera'], 'camera': ['lovely']}
{'display': 0.25, 'screen': 1.0, 'battery': 0.4166666666666667, 'phone': 0.0, 'camera': 0.5}

Process finished with exit code 0
```

When integrated with Django framework it produced the outputs on browser page.

B01LWZ14Z2{"battery": 2.5, "camera": 2.5, "overall": 3.3, "processor": 2.5, "display": 2.5}

B01LWZ14Z2{"battery": 2.5, "camera": 3.05, "overall": 2.56, "processor": 2.5, "display": 2.5}

B01LWZ14Z2{"battery": 3.12, "camera": 2.5, "overall": 2.5, "processor": 2.5, "display": 2.5}

B01LWZ14Z2{"battery": 2.5, "camera": 2.5, "overall": 2.5, "processor": 2.5, "display": 2.5}

These Outputs were integrated in UI to produce results in well presented manner like pie charts, column charts and stacked charts.

Chapter 7: Result Analysis

7.1 Screenshots of User Interface (UI)

1. Home Page

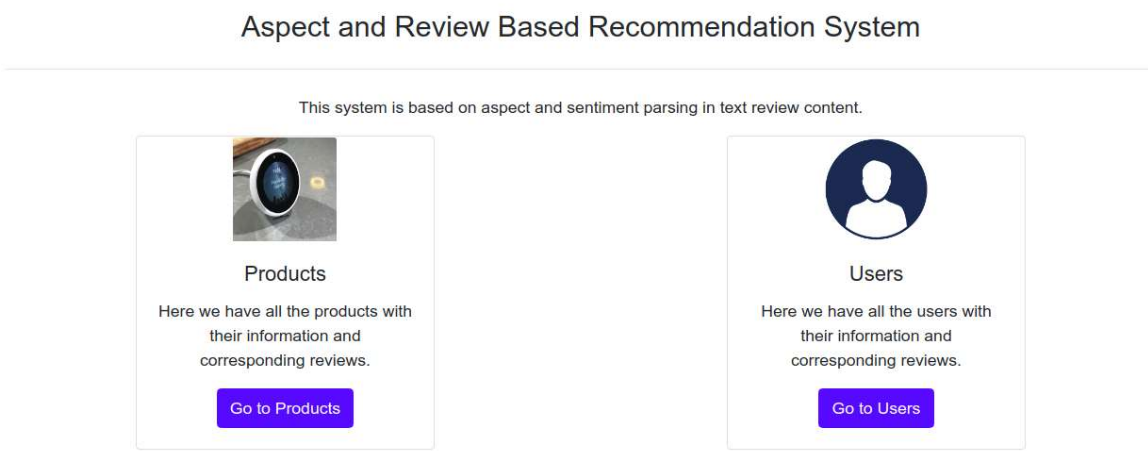


Fig 7.1.1 Home Page

2. Product Page

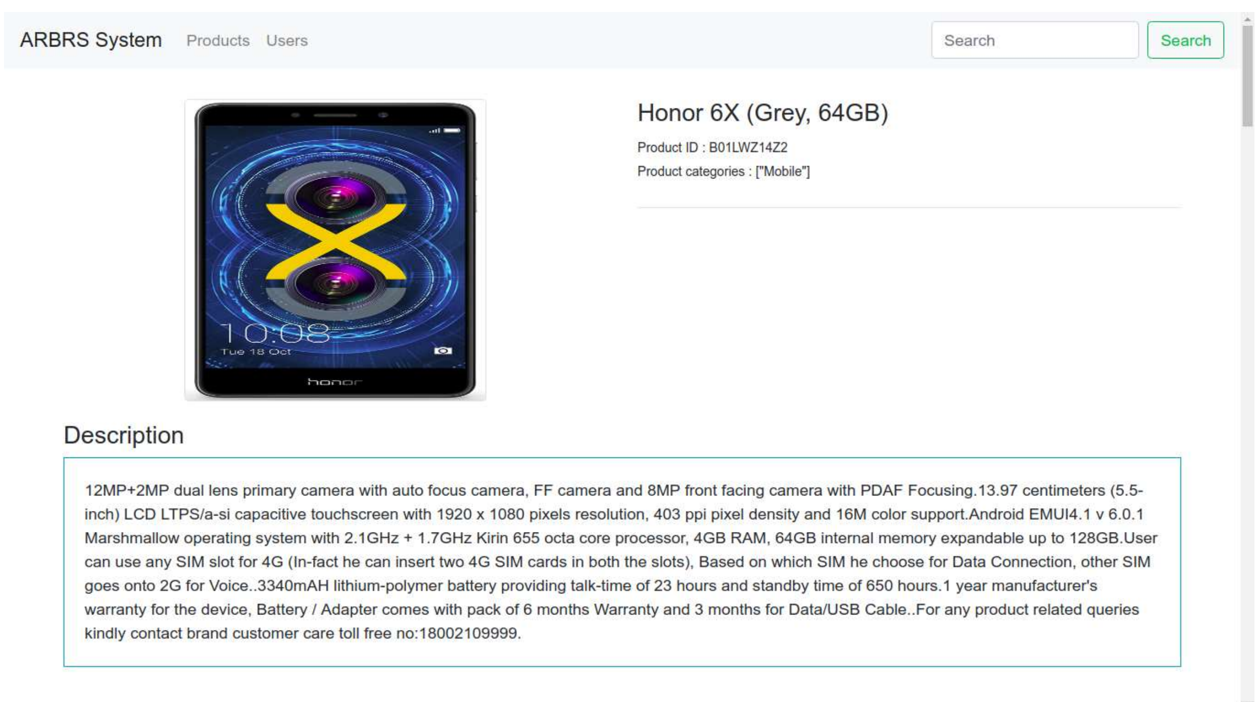


Fig 7.1.2 Product page

3. Reviews on Product

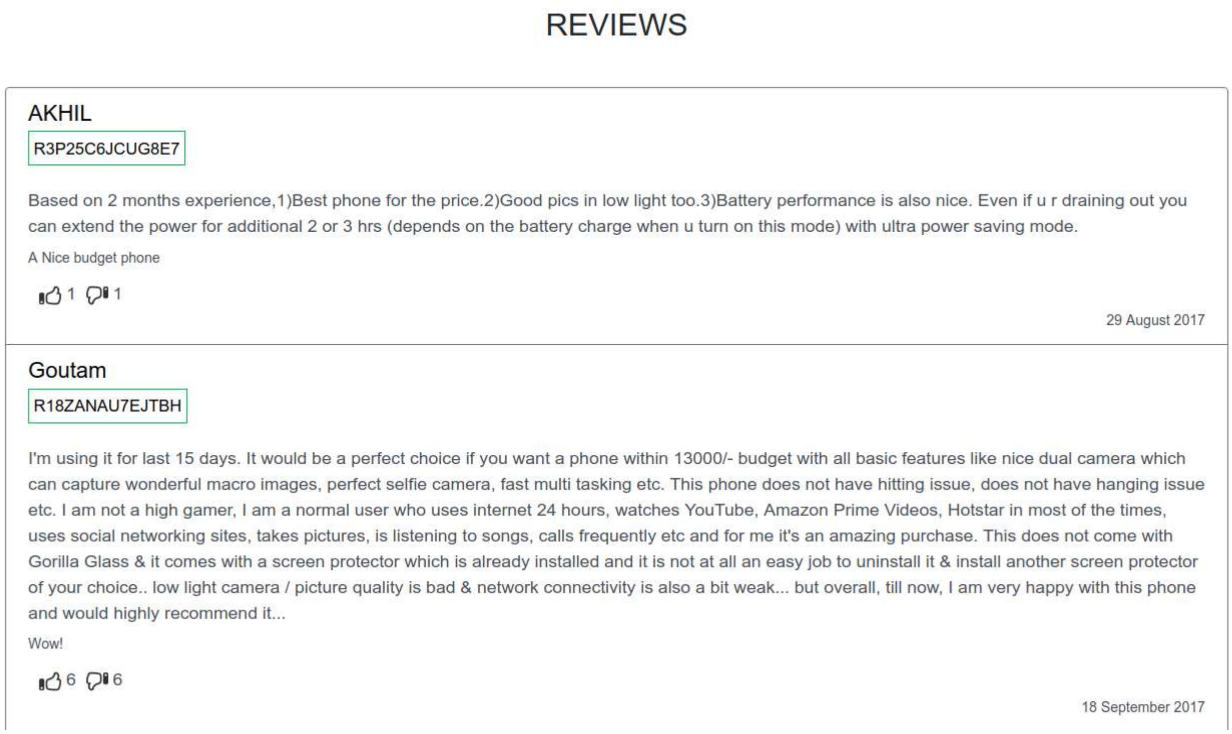


Fig 7.1.3 Reviews

4. Write a Review on product

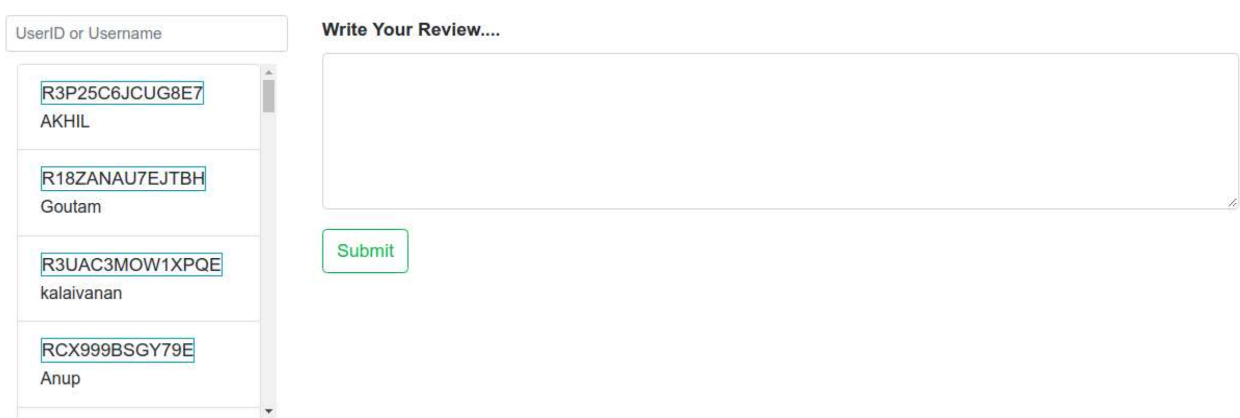


Fig 7.1.4 Submit Review

5. Product Navigation

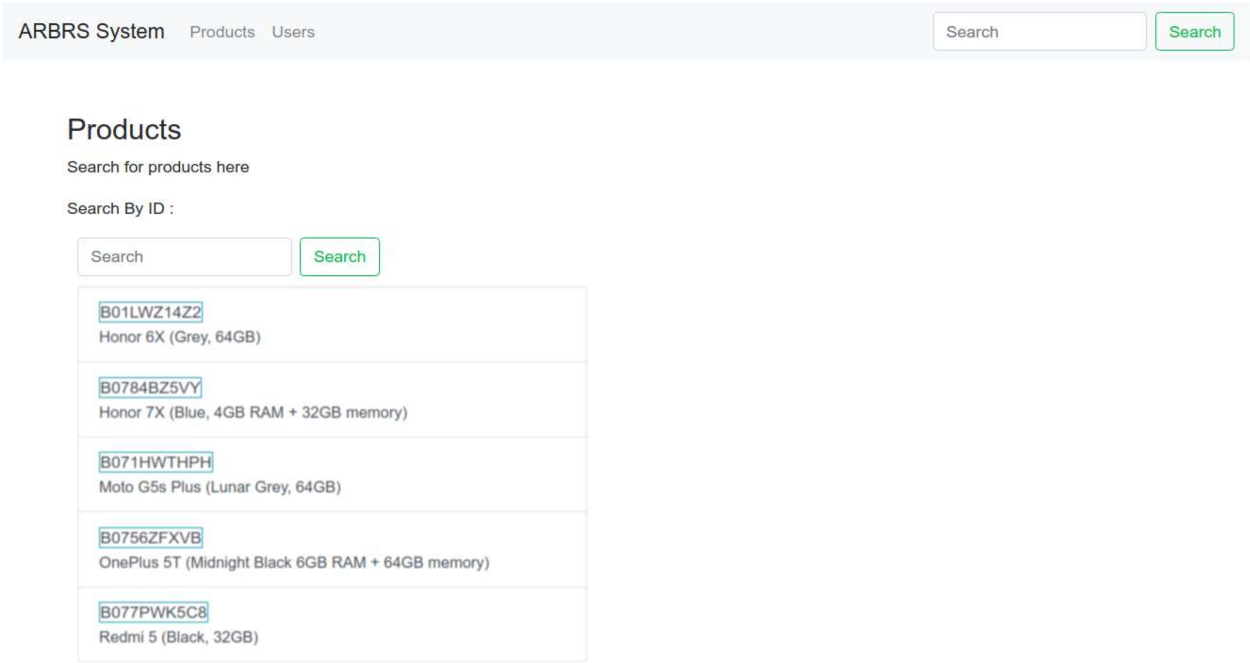


Fig 7.1.5 Products List

6. User Page

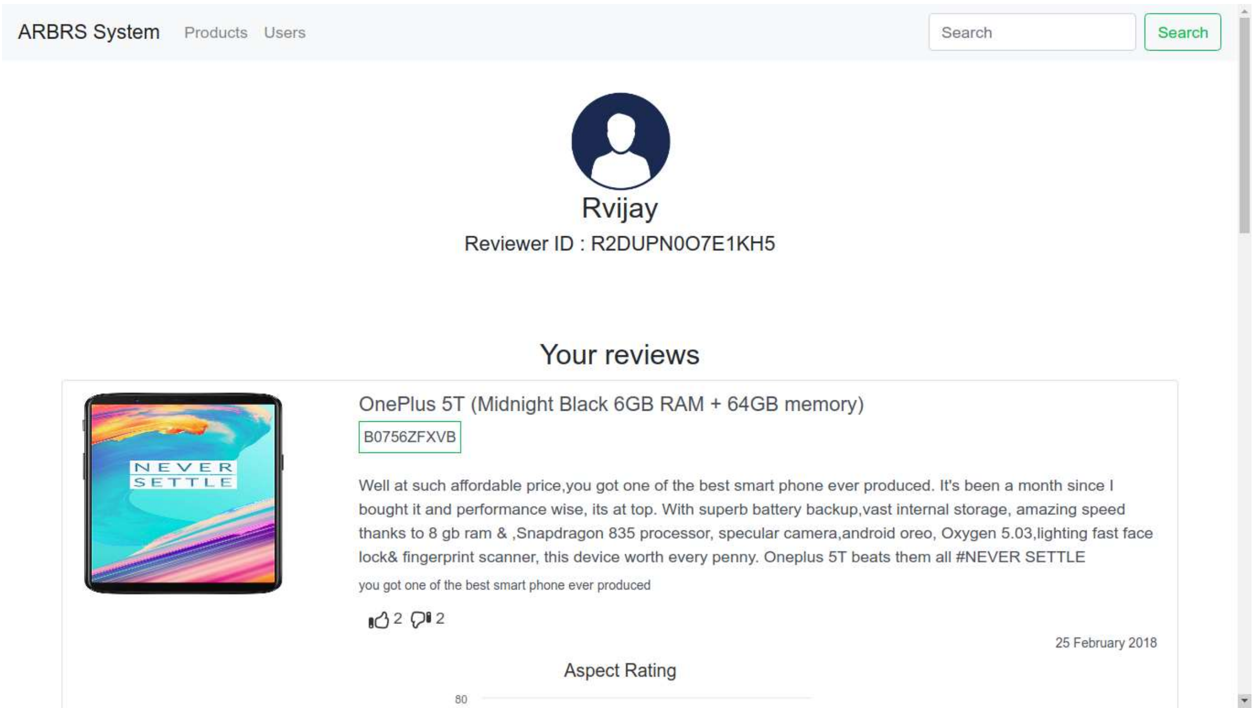


Fig 7.1.6 User Page

7. Recommendations

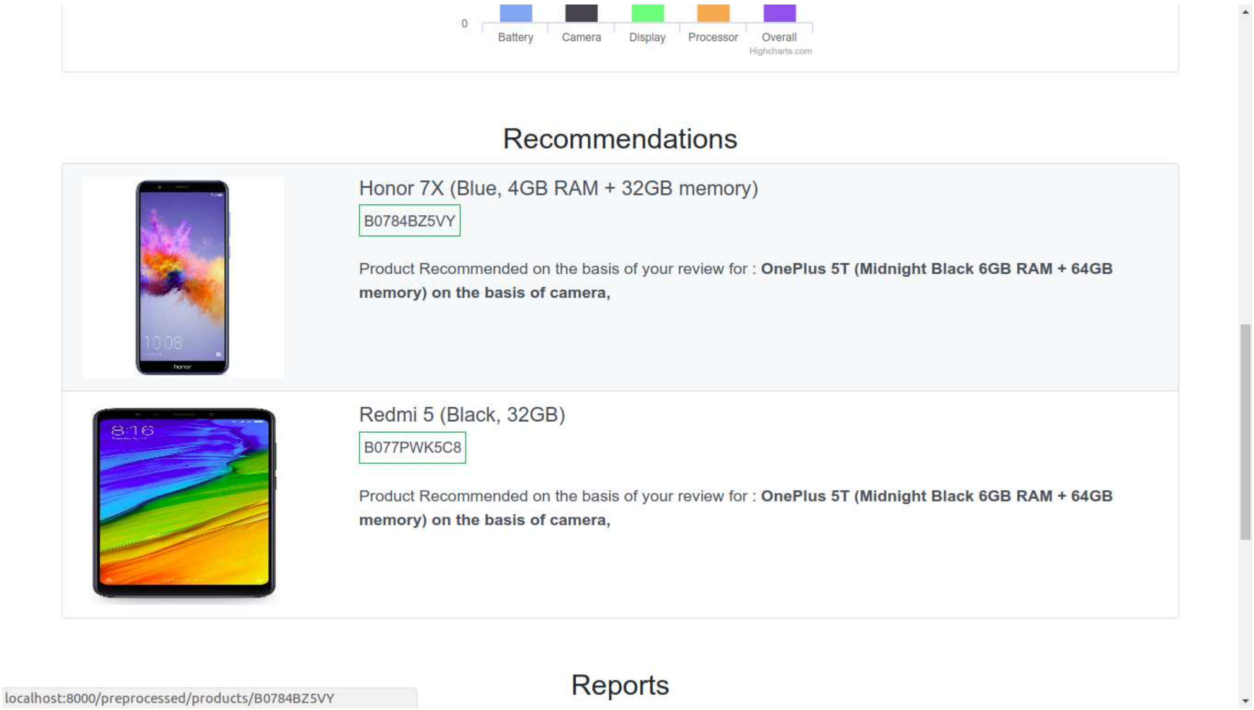


Fig 7.1.7 Recommendations to the user

8. User Navigation

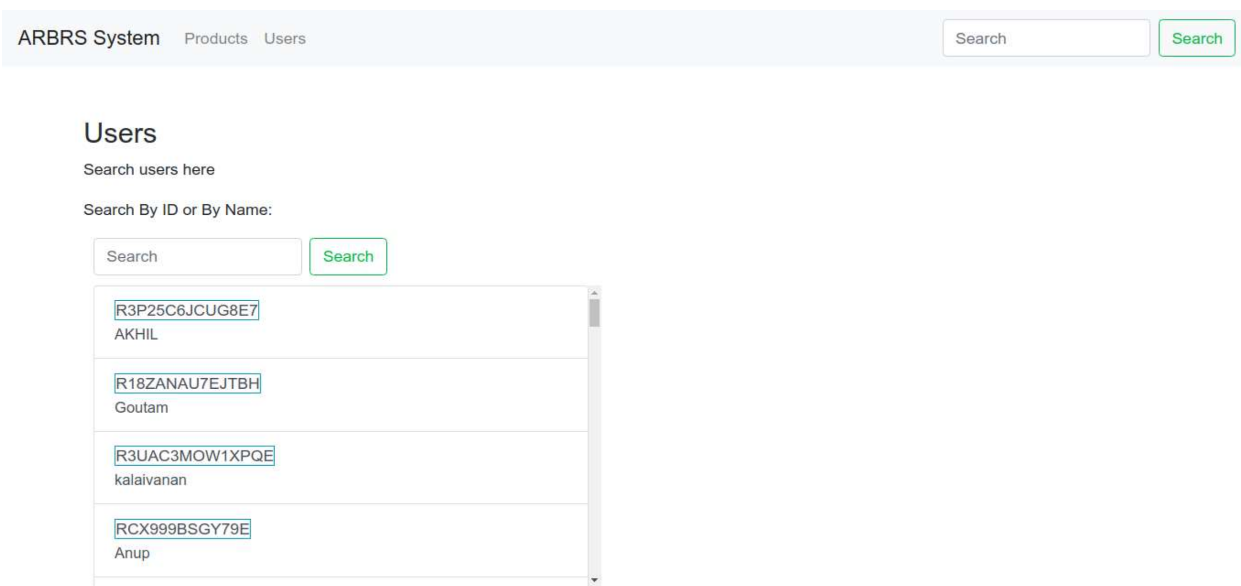


Fig 7.1.8 Users list

7.2 Graphical outputs

1. User Recommendation Graph

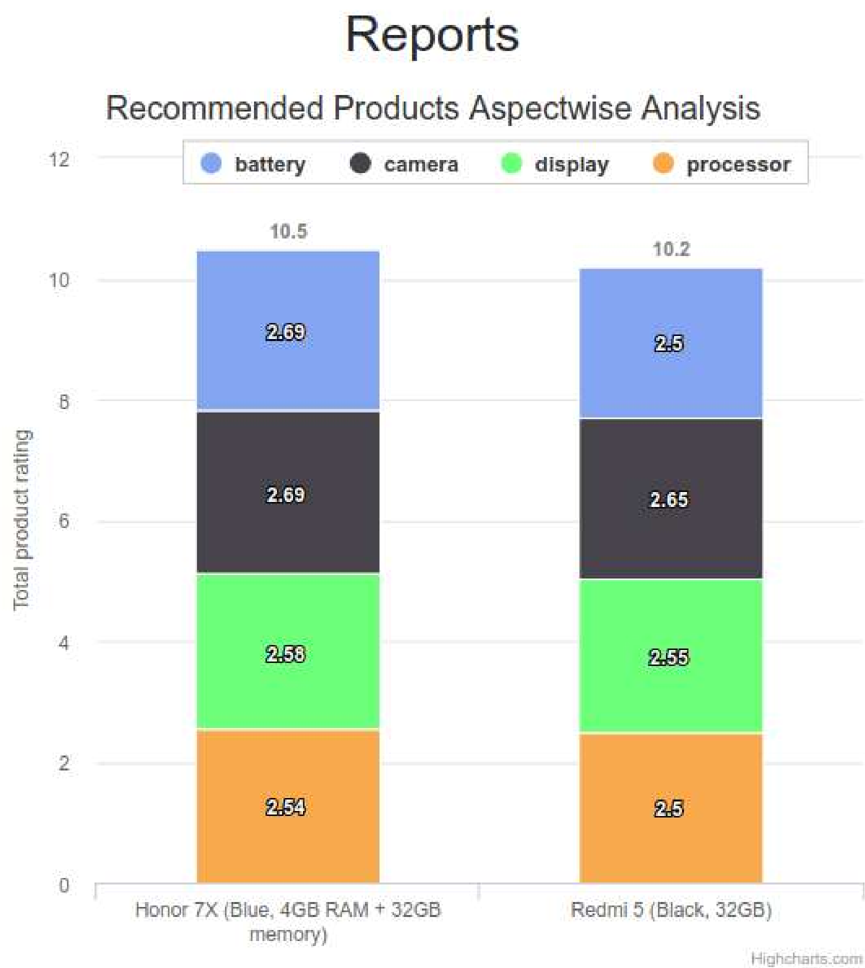


Fig 7.2.1 Sentiment Graph for Product recommended to the user

This graph in its true essence defines 'better of the best' perfectly. It contains the products recommended to a user placed next to each other and each colour in the bar represents an aspect of the product. This can be used as a direct comparison between the best products fetched by the system and then leave it up to the user which product to actually buy.

2. User Recommendations (one aspect disabled)

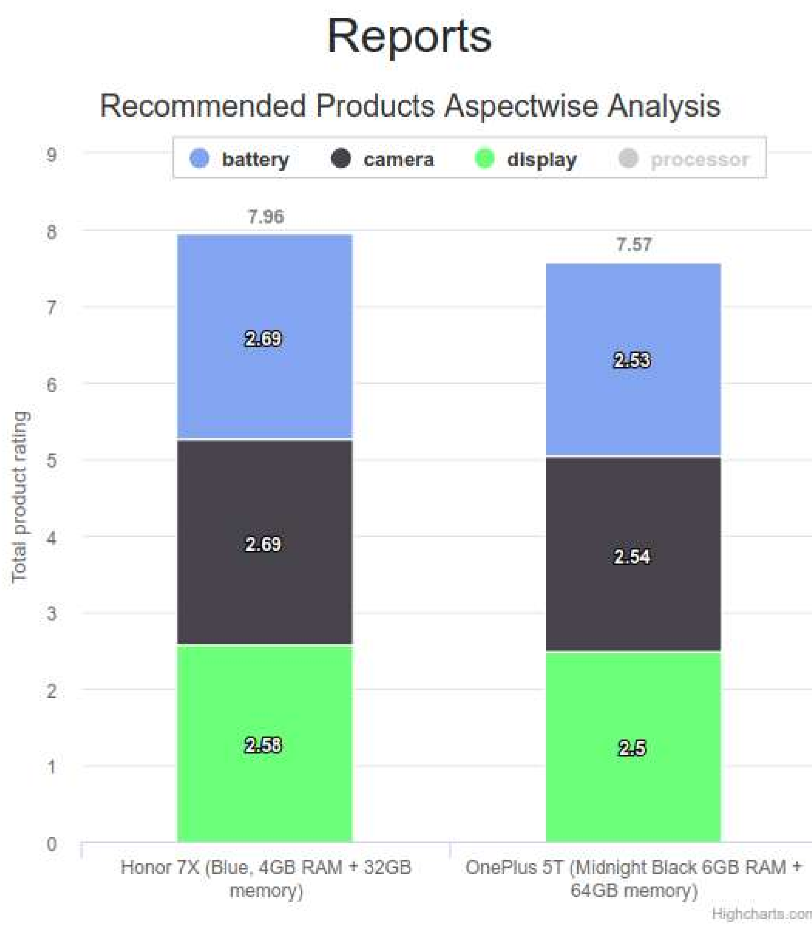


Fig 7.2.2 Sentiment Graph for Product recommended to the user

This graph is the same as the graph above, with one aspect disabled if the user is not concerned with that aspect.

3. **Review Sentiment**

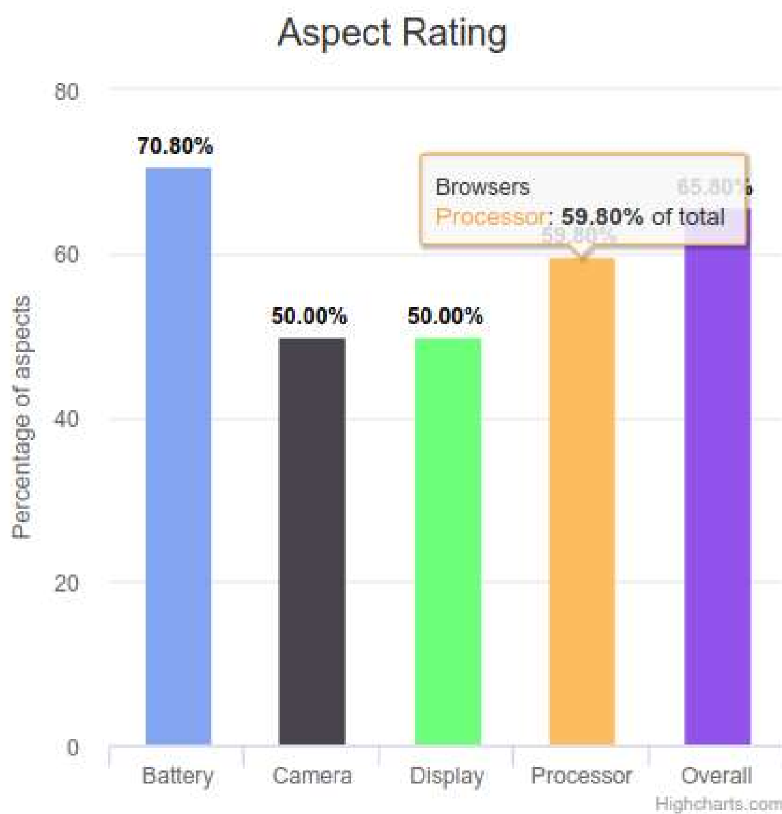


Fig 7.2.3 Sentiment Graph according to each user-review

This graph represents various aspects mentioned in a certain set of reviews. This graph is user specific and the aspects extracted from all reviews of one user to describe the likes and dislikes of a particular user. It describes what aspects are more important to the user and also what aspects the user is most critical about. This will show the user preferences and help recommendations in the system and can also be provided to the vendors for targeted advertisement.

4. **Product Sentiment**

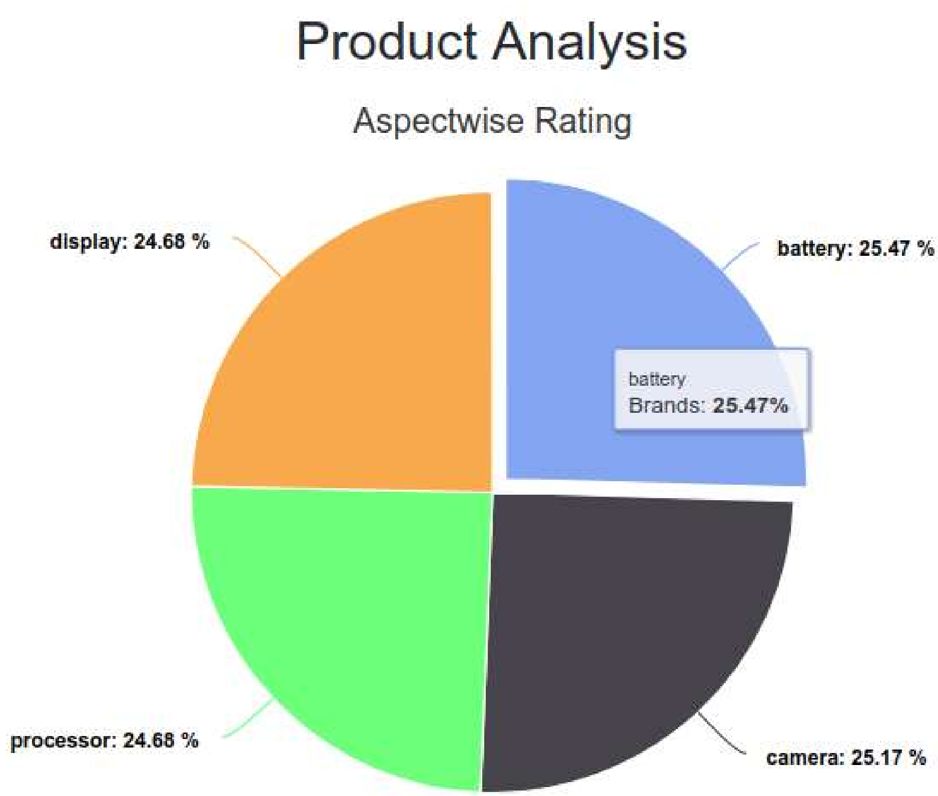


Fig 7.2.4 Sentiment graph for each product

This pie-chart represents the overall sentiment of the product. The graph can be used by vendors, to see how each and every aspect of a product fares in the eyes of the user. If a particular aspect has a low rating, the vendor will make sure that the company makes up for the same with its new product. This basically provides a rounded feedback about the product to the vendor. If there is an aspect which is highly rated, the vendors will use this aspect as a foundation for advertisement.

Chapter 8: Conclusion

8.1 Limitations:

- It is necessary that the users write reviews using proper grammar.
- The punctuation and spacing should be perfect in the review.
- The reviewer must write only correct spellings of words to make sure that they are considered as aspects.
- The dictionary is constructed manually, hence it is necessary that a Knowledge Engineer updates the dictionary frequently whenever he/she comes across certain words which are relevant to a specific aspect.
- This system is currently limited to the English language and no other.
- The system's Natural Language Processing depends entirely on the Stanford CoreNLP Dependency Parser. The parser produces incorrect POS tags resulting in anomalous typed dependencies.

8.2 Conclusion

The project aims at implementing a novel approach in recommendation systems to make them more precise and specific. The system essentially considers a broader set of perspectives which can form the basis for judging whether a person will like a certain product or not and if he/she will, to what extent. If the users of this system who follow the recommendations will rate the product the same or better, it is evident that these users conformed to the recommendations the system provided. The businesses will have a clear picture of the opinion not only about their products, but about specific aspects of their products in the state-of-the-art e-market. This will assist the business in improving the very particular areas where the customers are unhappy. It makes the overall business process more efficient and objective. This system is a state-of-the-art system wherein the recommendations are personalized which add to their value.

The rule set is constructed by exploiting the typed dependencies of the Stanford CoreNLP Dependency Parser. For doing this and gain the extent of accuracy that we did, we required to study the Stanford Dependency Manual in depth. The meaning of each typed dependency is explained in this document along with the variety of conditions which trigger for that dependency to show up on the output of the parser. As we constructed direct rules one by one, which were applicable only when certain criteria were met with respect to POS tags and dependencies, we discovered some anomalies, which were relevant to all direct rules. Thus we made the universal rules and associated them with each direct dependency rule. Some other improvements were made like replacing all pronouns with the relevant nouns using the coreference resolver.

This system provides a variety of data in graphical format, which gives different insights to both, buyers and vendors. The buyers are given an output which shows them an aspect-wise comparison for the products best-suited for them. The vendors get an overview of the sentiment of various aspects relevant to a certain product so as to make business plans based on the approach or inclination of the users to a particular aspect. The recommendation system uses the buyer's reviews on various products and which feature of the product the user most frequently mentions in all his/her reviews. This gives the vendors a perspective to focus on targeted advertising and recommendations. The overall sentiment of aspects over a variety of products is also useful so as to give a view of which aspect is considered the most while an average user buying a product and giving the industry a perspective about the universal sentiment.

8.3 Future Scope

- Inclusion of more number of languages.
- Usage of an unsupervised learning approach to cluster semantically similar aspects as opposed to the present knowledge engineer based approach.
- Inclusion of more domains.

Chapter 9.Appendix

A. List of figures