

Final Year Project Report

Document Level Sentiment Classifier



Bahria University Islamabad

30th April, 2018

Supervisor

Bilal Ashraf Awan

Group Members

Sadaf Iftikhar (01-133142-199)

M. Azman Hamza Rana (01-133142-263)

Department of Software Engineering

ABSTRACT

Document Level Sentiment Classifier is a web based application which is following the concepts of data mining. This application aims to analyze the orientation of online documents i.e. reviews regarding cell phones on the basis of sentiments present in it. User will be able to search his/her desired cell phone which he/she is opting to buy. The searched results will then target various websites to extract and scrap real time data on which further operations (Such as Data cleaning, data storage etc.) are applied. Classification algorithm Naïve Bayes is used to categorize data into positive & negative sentiments. Dashboard displays visual representation of the analyzed results.

Vision of this project is to develop an application that will facilitate the users in finding opinions and sentiments of people about a cell phone. This can help individual users in making informed decision about buying a cell phone by knowing how that is perceived in the market. The application will also help phone-manufacturers in getting valuable feedback from the market and orientation of the buyers towards their product for further improvement.

Our application targets online cellphone reviews to analyze cellphone user's orientation and keeps on updating it regularly. Users can visually see the graphical representation of results enabling them to make wise and correct decision before buying such products. This web-based application is free of cost thereby relieving people from financial burden of expensive licensed applications. Users will just have to make an account once or sign in via Facebook or Gmail. Instead of spending time on going through numerous websites to get an overview of your chosen cellphone, this application will provide you the best solution about the product you are looking for.

Table of Contents

ACKNOWLEDGEMENT.....	iii
ABSTRACT.....	iv
LIST OF TABLES.....	ix
LIST OF FIGURES.....	x
CHAPTER-1 INTRODUCTION	1
1.1 Motivation.....	2
1.2 Problem Statement.....	2
1.3 Goals/Objectives.....	2
1.3.1 Data.....	2
1.3.2 Update.....	2
1.3.3 Time Saving.....	2
1.3.4 Dashboard.....	2
1.3.5 Interface.....	2
1.4 Main contributions.....	2
1.5 Thesis organization.....	3
CHAPTER-2 BACKGROUND/ LITERATURE REVIEW	4
CHAPTER-3 SYSTEM REQUIREMENTS	7
3.1 Interface Requirements.....	8
3.2 Functional Requirements.....	8
3.2.1 Functional Requirement #1: Authentication.....	8
3.2.2 Functional Requirement #2: Search keyword.....	8
3.2.3 Functional Requirement #3: Collect appropriate data.....	8
3.2.4 Functional Requirement #4: Clean Data.....	8
3.2.5 Functional Requirement #4: Analyze data.....	8
3.2.6 Functional Requirement #4: Store Data.....	8
3.2.7 Functional Requirement #4: Display results.....	8
3.3 Use Cases (Use Case Description).....	9
3.3.1 Use Case # 1: Login.....	9
3.3.2 Use Case # 2: Search by keyword.....	10

3.3.3 Use Case # 3: Gather Data	11
3.3.4 Use Case # 4: Store Data	12
3.3.5 Use Case # 5: Clean Data.....	13
3.3.6 Use Case # 6: Classify Documents.....	14
3.3.7 Use Case # 7: View Dashboard.....	15
3.3.8 Use Case # 8: Sign Up.....	16
3.3.9 Use Case # 9: View 'Contact Us'	17
3.4 Non-Functional Requirements.....	18
3.4.1 Performance.....	18
3.4.2 Reliability.....	18
3.4.3 Security	18
3.4.4 Consistency	18
3.4.5 Understandability.....	18
3.5 Resource Requirements.....	18
3.5.1 A PC	18
3.5.2 Django	18
3.5.3 Beautiful Soup.....	18
3.5.4 PostgreSQL.....	19
3.5.5 Trained Model.....	19
3.5.6 Requests.....	19
3.6 Database Requirements.....	19
3.6.1 Relevant web documents (data).....	19
3.6.2 PostgreSQL Server.....	19
3.7 Project Feasibility.....	19
3.7.1 Technical Feasibility	19
3.7.2 Operational Feasibility	19
3.7.3 Legal & Ethical Feasibility.....	20
CHAPTER-4 SYSTEM DESIGN	21
4.1 Design Approach.....	22
4.2 Design Constraints	22
4.3 Interface Design	22
4.3.1 Low Fidelity Prototype	22

4.3.2 High Fidelity Prototype	24
4.4 Data Flow Diagrams (DFD)	27
4.4.1 Data Flow Diagram: User level	27
4.4.2 Data Flow Diagram: System level	28
4.5 State-Transition Diagrams (STD)	28
4.6 Schema Diagrams	29
4.7 Entity-Relational Diagrams (ERD)	29
4.7.1 Domain Model	30
4.7.2 Design Approach	30
4.7.3 Sequence Diagram	30
4.7.4 Sequential Diagram: Login	30
4.7.5 Class Diagram	32
4.8 Functional Flows	32
4.8.1 Functional Flow	32
CHAPTER-5 SYSTEM IMPLEMENTATION	33
5.1 Strategy	34
5.2 Tools Used	34
5.2.1 Beautiful Soup	34
5.2.2 REST API (Representational State Transfer)	34
5.2.3 Django	34
5.2.4 PostgreSQL	34
5.2.5 Anaconda	34
5.2.6 Algorithms	34
5.3 Methodologies	35
5.3.1 Choose websites	35
5.3.2 Sign Up/Login	35
5.3.3 Provide Keyword option	35
5.3.4 Collect data	35
5.3.5 Clean data	35
5.3.6 Analyze Data	35
5.3.7 Store data	35
5.3.8 Display Dashboard	36

5.4 System Architecture.....	36
5.4.1 Data Layer	36
5.4.2 Processing Layer.....	36
5.4.3 Representation Layer.....	36
CHAPTER-6 SYSTEM TESTING.....	37
6.1 Test Strategy	38
6.1.1 Component Testing.....	38
6.1.2 Unit Testing.....	38
6.1.3 Integrated Testing.....	38
6.1.4 System Testing	38
6.2 Test Cases.....	38
Test cases are as follows:.....	38
6.2.1 Test Case#1: Successful Login	38
6.2.2 Test Case#2: Unsuccessful Login (Incorrect ID/Password)	39
6.2.3 Test Case#3: Unsuccessful Login (Empty field)	39
6.2.4 Test Case#4: Unsuccessful Login (Nonexistent ID)	40
6.2.5 Test Case#5: Successful sign up	40
6.2.6 Test Case#6: Unsuccessful sign up (Different Re-entered password)	41
6.2.7 Test Case#7: Unsuccessful sign up (Email ID already exists)	41
6.2.8 Test Case#8: Enter keyword (keyword matched)	42
6.2.9 Test Case#9: Enter keyword (keyword not matched with database).....	42
6.2.10 Test Case#10: Enter Invalid keyword.....	43
6.2.11 Test Case#11: View Dashboard successfully.....	43
6.2.12 Test Case#12: View "Contact us"	44
CHAPTER-7 CONCLUSION	45
7.1 Future work.....	47
References	48
APPENDIX:	49