Department of Computer Science & Engineering

Course Title – Software Engineering

Course Code – CSE 322

Section - A₁

Project Proposal

Laptop Recommendation system using NLP

Date of Submission – February 11, 2022

Submitted by:

Md. Hasibur Rahman (19101009)
Tanmoy Mazumder (19101013)
Shawan Das (19101020)

Submitted to:

Dr. Shah Murtaza Rashid Al Masud
Associate Professor,
University of Asia Pacific

1. Introduction

The pandemic has forced the world to go on lockdowns multiple times in the past 20 months. Each time students and a large number of office workers had to shift to their tasks from home and this created an unprecedented surge in Computer sales, especially Laptops since laptops are more mobile compared to their desktop counterparts and they come with a pre-built video calling specs.

A. Problem Statement

As the number of sales of laptops increased, so did the number of complaints from the users that they had been ripped off at the store for not understanding the specifications. This has become more and more common as there are multiple things to be considered while buying a laptop and an average person has very little to no idea of these specifications, let alone which specs would be the perfect choice for their needs.

This problem can be tackled by searching online for reviews, as user reviews reflect a true image of how the product really is. However, this method falls short because of one problem: there are literally thousands of reviews online about hundreds of such devices.

B. Objectives

This project tries to overcome this certain problem using NLP (Natural Language Processing) and some Machine learning algorithms used on data collected through text-mining from online review sites. The core goals of the project:

- Gathering user review data from the internet
- Building a system capable of understanding these human-written text reviews
- Mapping the user requirements to the suitable products based on the review, and suggesting a list of these suitable products.

C. Background Study

From research on this particular topic online, it has been concluded that no such product / service exists that are exactly similar to the goals of this project.

D. Literature Review

Sentiment analysis is the analysis of customer's opinions, expressions, likes and dislikes towards products, organizations or services. These papers primarily work on this.
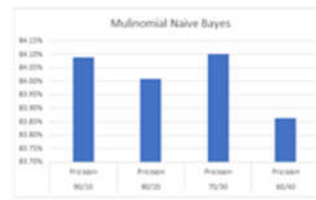
A. Product Recommendation using Sentiment Analysis of Reviews: A Random Forest Approach
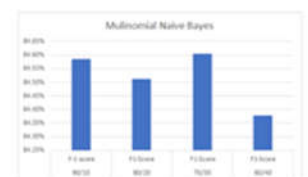*Authors:* Gayatri Khanvilkar, Prof. Deepali Vora

Ø Tries multiple approaches

Ø Based on various products

Ø Classifies the content based on polarity
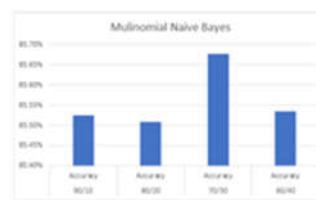
Graph 1(a) Accuracy of Multinomial NB

Graph 1(d) Recall of Multinomial NB

Graph 1(b) Precision of Multinomial NB

Graph 2(a) Accuracy of Logistic Regression

B. Data Analysis: Opinion Mining and Sentiment Analysis of Opinionated Unstructured Data
Authors: Harshi Garg, Niranjan Lal

Ø Provides a more structurally defined approach

Ø Goes deeper into the topic of sentiment analysis

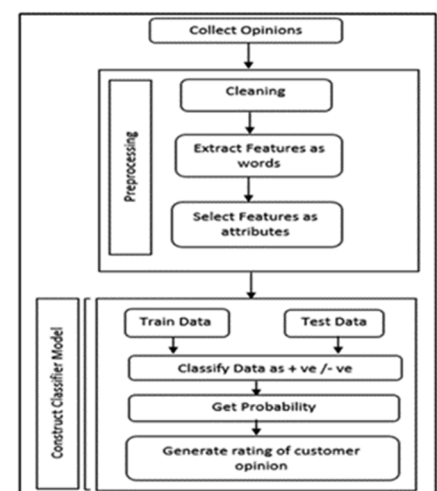Ø Discusses both approach of Machine learning and Lexicon-based

Fig. 2. Common methodology used for opinion mining

C. Comparison of Naïve Bayes, Support Vector Machine, Decision Trees and Random Forest on Sentiment Analysis *Authors: Márcio Guia, Rodrigo Rocha Silva, Jorge Bernardino*

Ø Works on a similar dataset: mobile phones

Ø Little outdated
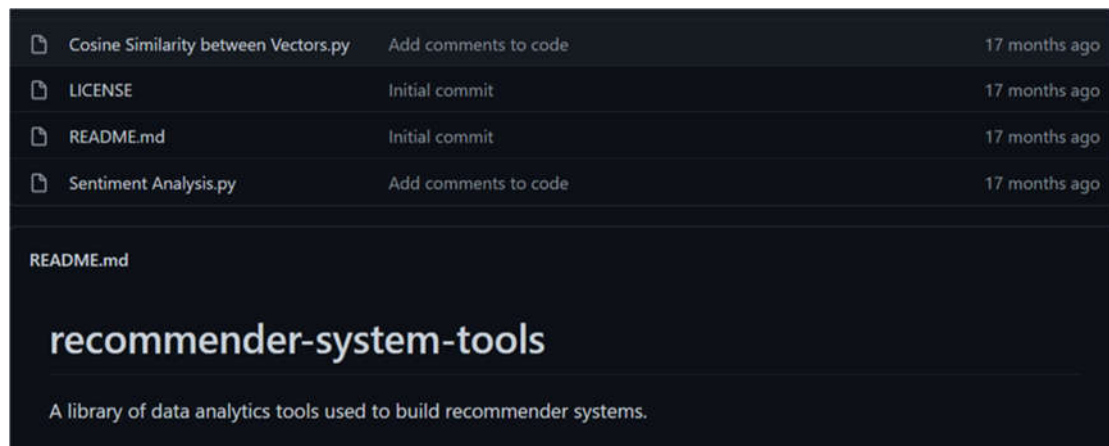
Ø Viable with adjustments

Table 5: Results for the impact of the brand on polarity review.

Brand	% of reviews	
	Positive	Negative
Samsung	79.94	20.06
Apple	77.3	22.7
Nokia	78.01	21.99
BlackBerry	74.3	25.7
Asus	77.41	22.59
LG	77.2	22.8
Sony	79.86	20.14
ZTE	82.9	17.1

D. Tell Me What You Want: A Recommender System Based on Customer Preferences and Product Reviews
 Author: Josh Barua

This is an abandoned work found on GitHub that can be used as a base for one portion of the project.

Cosine Similarity between Vectors.py	Add comments to code	17 months ago
LICENSE	Initial commit	17 months ago
README.md	Initial commit	17 months ago
Sentiment Analysis.py	Add comments to code	17 months ago

README.md

recommender-system-tools

A library of data analytics tools used to build recommender systems.

2. Complex Engineering Problem & Mapping:

❖ **How Ps are addressed through the project:**

Ps	Attribute	How Ks are addressed through the project	COs	POs
P1	Depth of Knowledge Requirement	Our project requires Machine learning (K2), rigorous study of existing projects(K8), surveys & reviews from users, monitoring new products (K3, K4), knowledge of web-development(K6), data collection & analysis(K5).	CO1 CO2 CO3 CO7	PO1 PO2 PO3 PO5
P3	Dept of Analysis Requirement	Users' requirements, device feather's data, users demand etc.	CO4 CO7	PO2 P12
P4	Familiarity of Issues	We need to Analyze & implement various programming knowledge and skills to fix many issues	CO7	PO3 PO9
P7	Interdependence	Our project involves interdependent components such as requirement analysis, designing, back-end, front-end, software testing, dataset etc.	CO2 CO7	PO2 PO10

❖ **How As are addressed through the project:**

Ps	Attribute	Ps are addressed through the project
A1	Range of Resources	Device's data is collected. Web framework is an important resource for this project. It engages diverse resources including various designing tools.
A2	Level of Interaction	Successful interaction between Web framework and NLP
A5	Familiarity	Our project deals with Internet users.

❖ **How COs are addressed through the Project:**

Ps	CO Statements	Corresponding POS
CO1	Identifying a real-life problem that can be transmitted to an engineering or computing solution through design, development and validation.	PO4 PO10 PO12
CO2	Identify, formulate and analyze a real world compels engineering problem based on requirement	PO2 PO3
CO3	Design/Develop a working solution on a complex software-intensive system and verify and validate the solution using industrial state of the practice, that indicates a high-quality software-intensive system	PO1 PO5 PO11
CO4	Use a modern/popular IDE to test complex software-intensive systems.	PO7
CO7	Work as a team and fulfill individual responsibility	PO9

CO	CO Statements:	Appendix-1	Appendix-2
CO1	**Apply** the Engineering knowledge to provide a working solution on a complex engineering problem and submit a mapping.	1	1/Apply
CO2	**Identify, formulate, and analyze a** real-world complex engineering problem based on requirement analysis.	2	1/Analyze
CO3	**Design/Develop** a working solution on a complex software intensive system and verify and validate the solution using industrial state of the practice, that indicates a high-quality software-intensive system	3	1/Apply
CO4	**Use** a modern/popular IDE to test complex software-intensive systems.	5	1/Apply

CO5	**Identify** societal, health, safety, legal and cultural issues related to the project.	6	1/Analyze
CO6	**Practice** concepts of professional ethics, confidentiality, industrial standards.	8	3/Valuing
CO7	**Work** as a team and fulfill individual responsibility.	9	1/Apply
CO8	**Communicate** effectively through presentation and write effective reports and documentations on the project.	10	1/Apply
CO9	**Apply** project management principles using Version Control System, and appraise project operating cost, financial risk analysis for complex software intensive systems.	11	1/Apply
CO10	**Recognize** the need for, and have the preparation and ability to engage in independent and life-long learning for art of project management, distributed and collaborative software development and risk analysis for developing complex software-intensive systems.	12	1/Apply

No.	PO	Differentiating Characteristic
1	Engineering Knowledge	Breadth and depth of education and type of knowledge, both theoretical and practical
2	Problem Analysis	Complexity of analysis
3	Design/ development of solutions	Breadth and uniqueness of engineering problems i.e., the extent to which problems are original and to which solutions have previously been identified or codified
4	Investigation	Breadth and depth of investigation and experimentation
5	Modern Tool Usage	Level of understanding of the appropriateness of the tool

6	The Engineer and Society	Level of knowledge and responsibility
7	Environment and Sustainability	Type of solutions.
8	Ethics	Understanding and level of practice
9	Individual and Team work	Role in and diversity of team
10	Communication	Level of communication according to type of activities performed
11	Project Management and Finance	Level of management required for differing types of activity
12	Lifelong learning	Preparation for and depth of Continuing learning.

3. Requirement:

We have two types of requirements.

i. Functional Requirements:

Here we will discuss what our software offers to the customers. On our projected website a customer needs to input their desired specifications of the laptop to see the results.

- RAM: RAM size, Specifications of the ram.
- CPU, GPU: CPU brands, GPU preference.
- Storage: Storage of the laptop.
- Weight: Weight of the laptop.

To build this website we need to acquire some hardware & software requirement:

Software Requirement:

- VS Code
- Pycharm
- Jupyter Notebook
- Discord
- GitHub

Hardware Requirement:

- GPU- Ryzen 7 3750H
- GPU- 8GB, RX470
- Ram- 16GB
- SSD- 240GB
- Stable Internet Connection

ii. Non-Functional Requirements:

Nonfunctional requirements judge the system how well it can execute the task given by the user and also how good the system works. Our non-functional requirements are:

- Accuracy
- Effectiveness
- 24/7 active
- Reliability

4. Methodology

We took the waterfall model for our project. Waterfall model is linear sequential design for software development, here progress flows in one direction downwards just like waterfall. Each phase must be completed before going to the next phase. Overlapping is not allowed in the waterfall model.
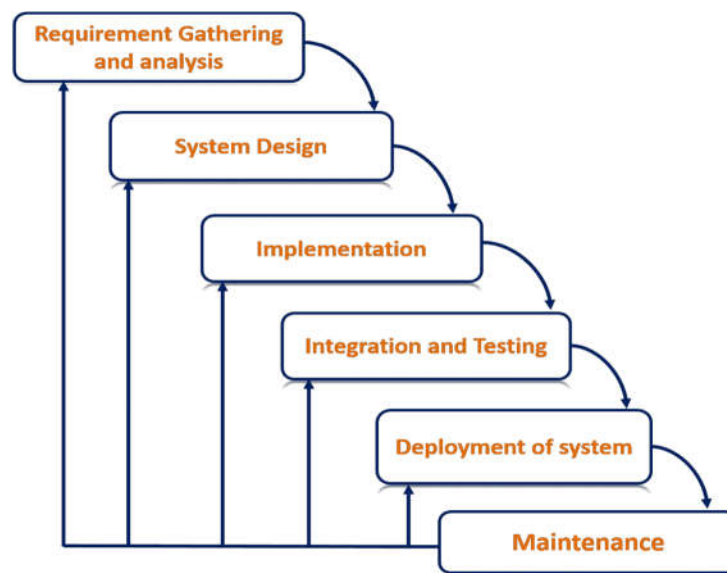
The following page contains details about each of these stages.

Requirement gathering and analysis:

In the analysis phase we gather data, analyze and validate information for the project. Define the requirements and prototype for the new system if possible. Economical and technical feasibility checked for the project in this phase.

			Year 0	Year 1	Year 2	Year 3	Year 4	Total
Benifit		Advertisement	0	100,000	155,000	198,700	252,000	705,700
		Total Benefit	0	100,000	155,000	198,700	252,000	705,700
COST	Development Costs	Site Development & Publishing	40,000	0	0	0	0	40,000
		Server Hosting	0	10,000	10,000	10,000	10,000	40,000
		Maintenance Cost	0	15,000	15,000	15,000	15,000	60,000
		Total Development cost	40,000	25,000	25,000	25,000	25,000	140,000
	Operational Costs	Advertisement	0	60,000	60,000	50,000	40,000	210,000
		Management	0	30,000	40,000	50,000	50,000	170,000
		Total Operational Cost	0	90,000	100,000	100,000	90,000	380,000
		Total Costs	40,000	115,000	125,000	125,000	115,000	520,000
Net Benefits = Total benefits - Total cost			[40,000]	[15,000]	30,000	73,700	137,000	185,700
Cumulative Net Cash Flow			[40,000]	[55,000]	[25,000]	48,700	185,700	
Return on Investment (ROI)			33.7% (185,700 / 520,000)*100%					
Break-even Point (BEP)			2.1 years [2 years + (92,800 - 87,600) / 92,800]					

We did an economic feasibility test based on our project. We can say that our system is economically feasible to work with.

System Design:

In the system design phase, we design UML, ER, DFD to represent the system as whole and how it will work in future. This design will be used in the Implementation phase as documentation.

Implementation:

In this phase we select the platform to work with. Implement the design into source code through coding. Combine all the modules together into a training environment that detects errors and defects. It is the longest phase of system development.

Integration and Testing:

In this phase we integrate the system into a real-life model and test it so that we can see if the system is working properly.

Deployment of System:

In this phase after testing the system we can deploy the system for users.

Maintenance:

In this phase technical support for any bug will be provided and undergo new changes daily, monthly for any new requirements based on customer requirement.

❖ Advantages:

Some advantages of this model:

1. It is easy to understand and use.
2. It works well with small projects where requirements are well defined.
3. It defined every stage clearly.
4. Every process is well documented.

We chose this model because we know this model very well and this model is used widely for the software development process.

Time Management:

Week	Task	Description	Start Date	End Date
1st	A	Discussion	18-Jan	18-Jan
2nd	B	Project Proposal	13-Jan	31-Jan
3rd	C	Design: UML, ER-Diagram, DFD, Gantt chart	1-Feb	15-Feb
4th	D	Project Update-1	16-Feb	14-Mar
5th	E	Project Update-1	15-Mar	28-Mar
6th	F	Project Update-1	29-Mar	11-Apr
7th	G	Final Project	16-Feb	26-Apr

Gantt chart: