A PROJECT REPORT ON

SOCIAL MEDIA AGGREGATOR

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)

SUBMITTED BY

PRASHANT AGRAWAL	B150054212
SHRUTI PHADKE	B150054480
SUDHANSHU BHOI	B150054487
SWAPNIL MARKHEDKAR	B150054492

UNDER THE GUIDANCE OF

Prof. R. V. Bidwe

DEPARTMENT OF COMPUTER ENGINEERING

PUNE INSTITUTE OF COMPUTER TECHNOLOGY

DHANKAWADI, PUNE – 43

CERTIFICATE

This is to certify that the project report entitled

SOCIAL MEDIA AGGREGATOR

Submitted by

PRASHANT AGRAWAL	B150054212
SHRUTI PHADKE	B150054480
SUDHANSHU BHOI	B150054487
SWAPNIL MARKHEDKAR	B150054492

are bonafide students of this institute and the work has been carried out by them under the supervision of Prof. R. V. Bidwe and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of the degree of Bachelor of Engineering (Computer Engineering).

Prof. R. V. Bidwe
Guide
Department of Computer Engineering

Prof. Mrs. M. S. Takalikar
Head
Department of Computer Engineering

Dr. R. Sreemathy
Principal,
Pune Institute of Computer Technology

Place: Pune
Date:

PROJECT APPROVAL SHEET

A Project Titled

SOCIAL MEDIA AGGREGATOR

has been successfully completed by

PRASHANT AGRAWAL	B150054212
SHRUTI PHADKE	B150054480
SUDHANSHU BHOI	B150054487
SWAPNIL MARKHEDKAR	B150054492

at

DEPARTMENT OF COMPUTER ENGINEERING

PUNE INSTITUTE OF COMPUTER TECHNOLOGY

SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE

ACADEMIC YEAR 2020-21

Prof. R. V. Bidwe
Guide
Department of Computer Engineering

Prof. Mrs. M. S. Takalikar
Head
Department of Computer Engineering

ABSTRACT

With the advent of social media, it has become more important than ever to have a social media presence. Social media is a useful tool to keep up with your friends and with what's going on around the world. However, with the increase of popularity of social media, it is easy to drown in a live feed that updates every second.

We propose a solution to categorise and group duplicates in one's feed from accounts on multiple social media platforms. Categorisation would help the user choose the topic of interest to explore posts at any given time. All the posts under a topic, including images and video, would then be grouped together. In this process, duplicate posts will also be detected and displayed together.

Keywords: social media, natural language processing, classification, duplicate detection, long short term memory, specialised information retrieval, supervised learning

ACKNOWLEDGEMENT

It gives us great pleasure in presenting the project report for our capstone project, 'Social Media Aggregator'.

We would like to take this opportunity to thank our internal guide Dr. R. V. Bidwe for giving us all the help and guidance we needed. We are really grateful to them for their kind support. Their valuable suggestions were very helpful.

We are also grateful to Prof. Mrs. M. S. Takalikar, Head of Computer Engineering Department, PICT for her indispensable support and suggestions.

Prashant Agrawal
Shruti Phadke
Sudhanshu Bhoi
Swapnil Markhedkar
B.E. Computer Engineering

Contents

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

1.1 Motivation

Social media are collaborative computer based applications that contribute to the development or sharing of content, ideas, career interests, and other methods of discourse through virtual spaces. Since the first widely popular social media network (Myspace, 2003) was released, social media networks have exponentially multiplied in number, as well as in function. The variety of stand-alone, and built-in social media services currently available makes them difficult to define.

The uniting factor, however, is that almost all social media networks are Web 2.0, Internet based applications. They enable the formation of online user social networks through the (usually service-specific) user profiles that each organization maintains. The crux of social media is typically interact-able user-generated content.

As social media has grown, both in terms of its user base and the extent of its usage by said user base, its effects, both negative and positive, on various aspects of human behaviour, as well as its use as a multi-purpose tool at both an organizational and individual level, has come under heavy scrutiny.

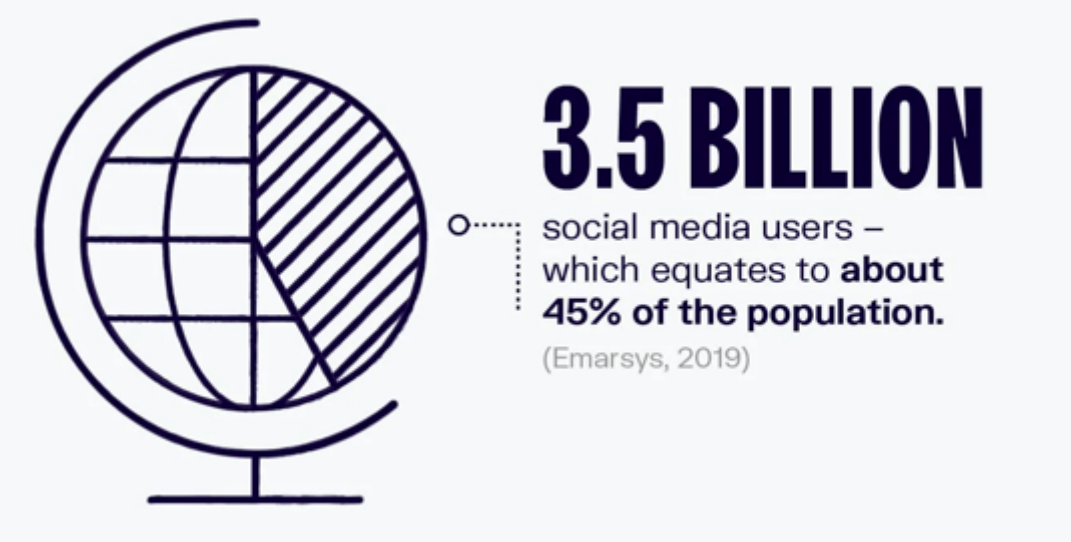


Figure 1.1.1: Social Media Users statistic infographic

While there are a multitude of positives about social media, its almost entirely negative psychologically impact cannot be ignored. Its extensive usage - with the average user spending a minimum of 2.4 hours a day on it – has a highly detrimental impact on mental health. [1]

Extensive use of social media can cross the line and turn into addiction. While problematic social media use is only a *proposed* form of a psychological dependence on social media platforms, and as of yet has not been recognized as a disorder by the World Health Organization or the Diagnostic and Statistical Manual of Mental Disorders (DSM-5), its adverse effect on its users is undeniable.

Figure 1.1.2: Social Media Usage infographic

The founding President of Facebook, Sean Parker, in an interview with Mike Allen of Axios, gave some insight into the goal of the organizations behind most social media networks, "The thought process that went into building these applications, Facebook being the first of them, was all about: 'How do we consume as much of your time and conscious attention as possible?"".[2] Social media networks like Facebook rely on interactions on their platform that result in the release of a hormone and neurotransmitter called *dopamine* in the brain.

The dopamine mechanism (homeostasis) plays a critical role in several forms of addiction. The intake of stimulants causes changes in the levels of dopamine in the brain that can last from mere minutes to hours. The chronic elevation in dopamine that comes with repetitive high-dose stimulant consumption triggers a wide-ranging set of structural changes in the brain that are responsible for the behavioral abnormalities which characterize an addiction. [3]

These structural changes, mainly the change in the brain's homeostatic balance (called allostatic overload) to accommodate the addiction, mean that the brain now requires the addictive substance or activity in order to maintain this new balance.

This is the reason treatment of stimulant-based addiction is very tough, because even if consumption is stopped, the desire that comes with psychological withdrawal does not, even when the consumption as well as desire seems to no longer exist. The craving may re-emerge when faced with stimuli once again, as association networks in the brain are greatly interlinked.

In the context of social media addiction, it means that users using a platform engineered to deliver increasingly higher doses of dopamine are predisposed to continue these harmful activities for the benefit of the platform, and to their own persisting detriment.

While the disadvantages of social media are significant, one cannot deny that man is a social animal, and that social media has enabled people to stay connected across globe. It facilitates positive as well as negative interactions, which means to overcome its negative effects, quitting social media is not a viable solution.

We propose a best-of-both-worlds solution: an aggregated social media platform, combining posts from multiple existing social media platforms, and classifying posts into categories, and leaving out certain features like comments and the infinite scroll "discover" feeds.

1.2 Problem Definition

The platform to be developed is an aggregation of at least three existing social media applications. The user should be able to log into all available social media applications and maintain the login. The user should be able to configure the categories and see posts from the people they follow within those categories.

The user should be able to switch between categories and 'like' the posts. Duplicate posts when detected, must have an appropriate indicator.

CHAPTER 2

LITERATURE SURVEY

Natural Language Processing is used to extract information from social media using text mining techniques which use summarisation, POS tagging, fact extraction, word disambiguation, sentiment analysis, etc. In this system, we use NLP techniques on the social media posts to classify them and find duplicates. First step in text classification is pre-processing which involves tokenisation, removing stop words, lower casing, noise removable, stemming, lemmatisation. This step is followed by word representation or embedding and then dimensionality reduction before applying one of the classification techniques [12]. [5] shows how Twitter-oriented toolkits perform better in social media text.

2.1 Classification

Much of the classification performed on the social media data is binary. Example to find whether the sentiment is positive or negative or to find whether the post is related to health or not [18]. Binary classification may have a covariate shift issue where not all the negative examples can be covered in the training data. [4] mitigate the problem by proposing CBS-L which uses center-based similarity instead of n-gram for transformation of document representation. [21] recommend hashtags based on words in the post content. [14] shows that lexicon based approaches perform poorly to ML based text classification on 41 datasets.

Social media data is considered to be noisy and unstructured. TagAssist [13] suggests tags to new posts by aggregating tags of similar already-tagged posts. [17] uses the source attributes, hyperlinked documents and metadata to perform classification. [10, 19] extracts the information from tweets as features, in form of vector and passes through classifier or ensemble ML methods [8]. It uses syntactic parser to extract the deep syntactic dependencies from which semantic can be calculated. [16] uses hierarchical RNNs for text classifications of mental health social media posts. It shows how the sequential consideration by the RNNs can be more beneficial than the CNNs. It also shows how the attention mechanism can distinguish words relevant for classification.

A multi-label classification is more complex task. [7] uses CNN for feature extraction before passing into dense layers for text classification. [6] proposes ensemble application of RNN and CNN to capture both local and global semantics. [22] uses ResNet into RNN to perform the classification task with lower parameters than LSTM with similar accuracy.

[11] is a comprehensive review of DL methods. It lists capsule networks, attention models, memory augmented networks, graphical neural networks, siamese neural network, transformers and pre-trained language models like BERT, autoencoders, etc. for various text classification tasks. [9] designs a Generative Explanation Framework to explain the decisions made by classification model by using fine-grained information. [20] combines the information from different modalities, like image and text, to perform emotion classification. [15] proposes ULMFiT which is a transfer learning method for NLP tasks.

2.2 Duplicate Detection

Two posts can be said to be duplicates if they are semantically similar. [28] uses 3 approaches to detect duplicate questions: first uses rule based heuristic like Jaccard coefficient, second uses ML based approaches like SVM, third uses deep CNNs. [24] converts the words into vector embeddings through Word2Vec and LSTM layers, and then calculated Manhattan distance between the 2 vectors. [25] uses a blend of three types of word embeddings, passes these to a Siamese MaLSTM neural network to predict duplicate questions in Quora dataset. [26] uses word2vec weighted by TFIDF to obtain word vector, these are then passed to Simhash to obtain characteristics, which are used to calculate hamming distance between 2 inputs. [27] proposes a method for fast near-duplicate detection from images in continuous incoming stream of social media. It uses 2 layer hierarchy: one for global feature descriptor-based similarity to reduce the search space and other LSH-based similarity for more accuracy. [23] performs classification and duplicate detection on human-loss news corpus.

CHAPTER 3

SOFTWARE REQUIREMENTS SPECIFICATION

3.1 Introduction

3.1.1 Project Scope

The project will be able to provide a platform for the following:

- Display posts from several social media handles on a single platform.
- Categorize the post, and displaying the post from the relevant category only to the end user.
- Combining duplicate post posted by same person on different platforms or by different person on the same platform.
- The user only wants to view its feed in aggregated manner, and does not want to add any post.
- Like and bookmark the project from the application itself. Bookmarked post can be viewed later also.

3.2 User classes And Characteristics

This application will be used by Users or Customers. They will use our application to check posts from various social media platforms on single application, with proper categorization and identification of duplicates. They can bookmark the post and like the post, from the application itself. These will help to save time for the user spent on surfing social medias.

3.2.1 Assumptions and Dependencies

The following are the important assumptions and dependencies for our application:

- User should have active social media accounts, to get the feed and enjoy the benefits of our application.
- The categorization will be done on text, so post should have relevant text content for categorization.
- Social media APIs are the main dependencies, and the API should be available for public use.

3.3 Functional Requirements

3.3.1 Authentication of users

The application will restrict its usage to the users who have at least one social media account that can be associated with the application. Only once the users have logged, they can use the application. Otherwise, the application will throw an HTTP 403 – indicating that it is forbidden to access the service.

3.3.2 Polling for social media feed

Each new post will be polled with the help of APIs provided by the respective social media. Following which, the post will be persisted to the database with its unique ID, and hence-after it will be pushed to the machine learning pipeline.

If the same post is fetched for a different user, then it will not be added to the pipeline again, thus reducing re-computations and the risks associated with miscalculation and duplication.

3.3.3 Categorization

Any post that enters the pipeline, will be classified based on the following categories: news, motivation, promotions, personal. The categorizations will be single-class classification, thus each post will be associated with only one category. The user can request to see post of any one category he is interested in.

3.3.4 Duplicate detection

Since a post may be posted across various social media, it adds redundancy, which is exactly what the project tries to minimize. Thus every post will be added to the pipeline to find the duplication percentage, and if it is above a certain threshold, then it'll be marked as a duplicate of one or more such posts.

3.3.5 Collation

After successful categorization and duplicate detection, similar posts will be collated and put together for ease of navigation to various social media sources for details associated with the respective post.

3.4 External Interface Requirements

3.4.1 User Interface

- The user interface would be simple and intuitive
- It would show the first show categories in big tiles, which would then contain list of subscribed topics/accounts
- For any post, the user would have the option to navigate to the source across various social media. To interact with the post, the user has the option to like or bookmark a post.

3.5 Non Functional Requirements

3.5.1 Performance Requirements

- User satisfaction: It is of utmost importance that the application not only meets and but exceeds expectations. This would rely on how effectively the application categorizes and detects duplicates across posts.
- Average response time: This is the time the user spends waiting for his feed to be populated. Ideally it should be as low as possible, and this can be achieved by pre-computing the heavy computations on the posts, and balancing load across the servers uniformly.
- Application availability: It is the extent to which the application would be functional and operational. This will be achieved by ensuring multiple instances of the server running simultaneously to ensure fail-safe mechanism.

3.5.2 Safety Requirements

- The number of requests per user could be very high, if automated by a bot. Thus it is important to limit the maximum number of requests per unit time by a user to a certain threshold. Any subsequent request would be ignored for a pre-determined cooldown period.

3.5.3 Security Requirements

- Authentication: The application does not store the plain text password in any database. Instead, the passwords are stored in local session of the browser. Thus any breach in security does not lead to loss of user passwords.
- Secure database: The database contains details of users posts and their respective categorization. The database is secured with the help of cloud service, thereby ensuring minimal tampering of leakage of data.
- Protection of integrity of responses: The response returned from the web service, to the user can be validated by using a checksum technique. If the contents of the response are manipulated or tampered with, the checksum calculated on the user's end will not match, and the response will be discarded. Subsequently, another follow up request will be sent to the server to resend the response.

3.5.4 Software Quality Attributes

The application would have the following software quality attributes:

- Correctness: The program code would agree with the specifications and independence of the actual application of the software system.
- Reliability: The application fulfils a certain maximum number of requests by the user.
- Learnability: The application learns and evolves from the data that it receives in the form of social media feed.
- Robustness: The application would handle valid or invalid inputs from the user without any unexpected behaviour or errors.

3.6 System Requirements

3.6.1 Software Requirements

- ReactJS Library for developing frontend.
- NodeJS Framework for developing backend.
- Python3 for NLP Engine.
- Natural Language Processing tools like ntlk or spacy.
- Tensorflow and scikit learn for creating classification models.

3.6.2 Hardware Requirements

- Processor which can handle intensive loads and provide efficient throughput.
- Laptop/Desktop with a minimum of 4GB RAM.
- Stable internet connection.

3.6.3 Database Requirements

- An SQL database PostgreSQL is used for storing the relevant information. The database will store the list of all users, and its credentials. It will store the post id and its category, and other relevant information. It will also maintain the list of duplicate posts.
- Database will also be required for caching, in which if the incoming post ID is already processed by the NLP Engine, the results are directly returned. This scenario occurs when the same post is sent by multiple users.

3.7 SDLC Model

SDLC stands for Software Development Life Cycle. It is a paradigm which depicts how the software can be made efficiently with complete use of resources and satisfying the user requirements to its fullest.

This project makes use of Agile Model. It is based on iterative and incremental development, where the requirements and solution evolve through collaboration between cross-functional terms. Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product.

Agile model breaks the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks.

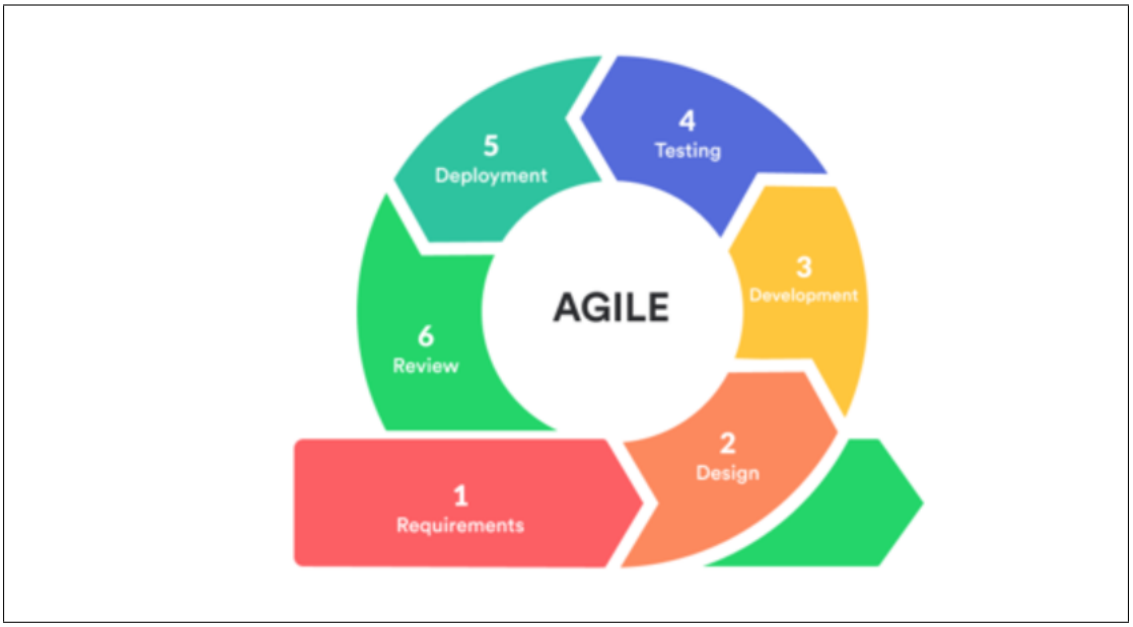


Figure 3.1: Software Development Life Cycle - Agile Model

Agile model mainly consists of six phases which are following:

I. **Planning** - This phase deals with the requirement analysis along with creating the future map of product development.

II. **Design** - This phase deals with creating of high level and low level architectures and the product workflow.

III. **Development** - This phase deals with implementing the tasks according to the designed architecture. Here the actual coding takes place.

IV. **Testing** - This phase deals with testing of the tasks and identifying the bugs if any.

V. **Release/Development** - Once the testing is completed, the product is ready and is handed over to the customer along with all the specified documents.

VI. **Maintenance** - The developed project is now maintained by performing regular updates to identify bugs and proper checks so that the project produces proper output when given proper inputs.

Following are the Agile Manifesto values:

- **Individuals and Interactions Over Process and Tools:** It is to value the individuals more highly than processes or tools, since it is the individual who responds to business needs and drives the development process. Here continuous communication with customer is given prime importance.
- **Working Software over Comprehensive Documentation:** Agile model streamlines the documentation in a form that gives the developer what is needed to do the work, without diving much into details. The agile manifesto values the working software more than its documentation.
- **Customer Collaboration Over Contract Negotiation:** The Agile Manifesto describes a customer to collaborate throughout the development process, unlike waterfall model in which the customer is delivered with final product. This makes it far easier for development to meet their needs of customer.
- **Responding to Change Over Following a Plan:** In agile, the shortness of iteration allows to shift the priorities and new features can be added into the next iteration. Agile's view is that changes always improve a project; changes provide additional value.

We are using agile model because this model helps us to know the exact status of the project at regular intervals, hence it helps to judge the efficiency of the project at very early stages. Agile methodology also helps to reduce errors further in projects as compared to other models like waterfall. It helps us to adapt to customers continuous change of plans. Agile model proves very effective with respect to incorporating changes during work.

We will be using Scrum to implement the Agile framework. Scrum is a simple framework for effective team collaboration on complex products. It is designed for teams with less than 10 members, who breaks their work into goals to be completed within a stipulated time period, called sprints. The progress is tracked by daily 15 minutes meeting, called daily scrums. At the end of sprint, team hold sprint review, to demonstrate the work done, and suggest any improvement which is required.

The advantages of using Scrum are:

- Helps to complete the project deliverables quickly and efficiently.
- Ensures effective use of time and money.
- Large projects are divided into easily manageable Sprints.
- Being agile it adopts feedback from customer and stakeholders.
- Short sprints enable changes based on feedback a lot more easily.
- The team gets clear visibility through scrum meetings.

3.8 System Implementation Plan

This sections presents an outline of how the system will be implemented efficiently in order to satisfy all the requirements. This includes the following tasks:

I. A thorough understanding of various social media APIs and NLP tools required to categorize and remove duplication of posts from the application.

II. Division of the tasks for creating the frontend, backend and NLP Engine among team members.

III. Implementation of each part or sprints by team members with continuous meetings to track progress.

IV. Testing of each module using various test cases. We will use various methods of testing to achieve a bug free product.

V. Sprint meeting with mentors after completing each goal, to receive a feedback and make changes in the next sprint if required.

CHAPTER 4

SYSTEM DESIGN

4.1 System Architecture

The backend engine fetches the posts from the various social media API endpoints. These are then passed to the NLP engine for classification and finding duplicates. The data is stored into the database. Frontend of the application requests backend for the information.

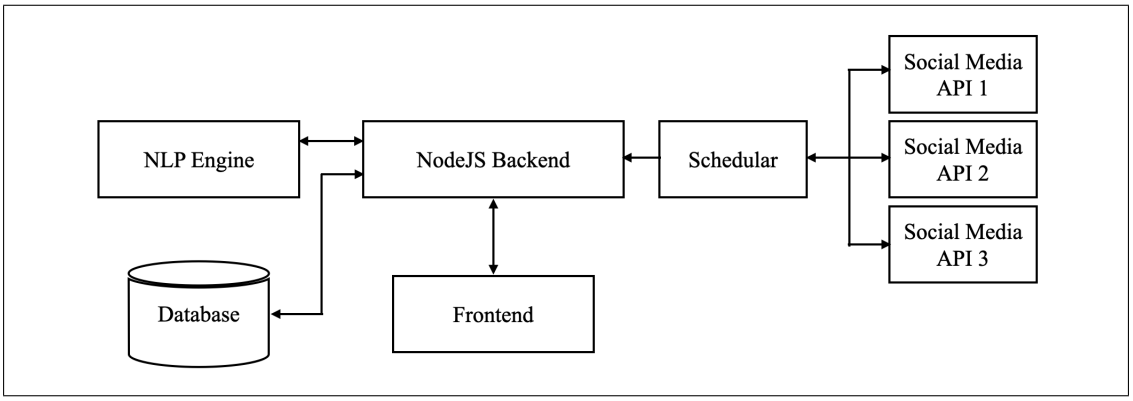


Figure 4.1: System Architecture

4.2 NLP Engine Architecture

NLP engine has 2 major functions: classification and duplicate post detection. To classify a post into one of the pre-defined categories, the classification part takes post as an input. After text preprocessing, it finds out word-embeddings and passes those to a classifier which then classifies the post. For duplicate detection, it considers a group of posts as input. It then calculates the distance between word-embeddings of post pairs to group duplicate posts together.

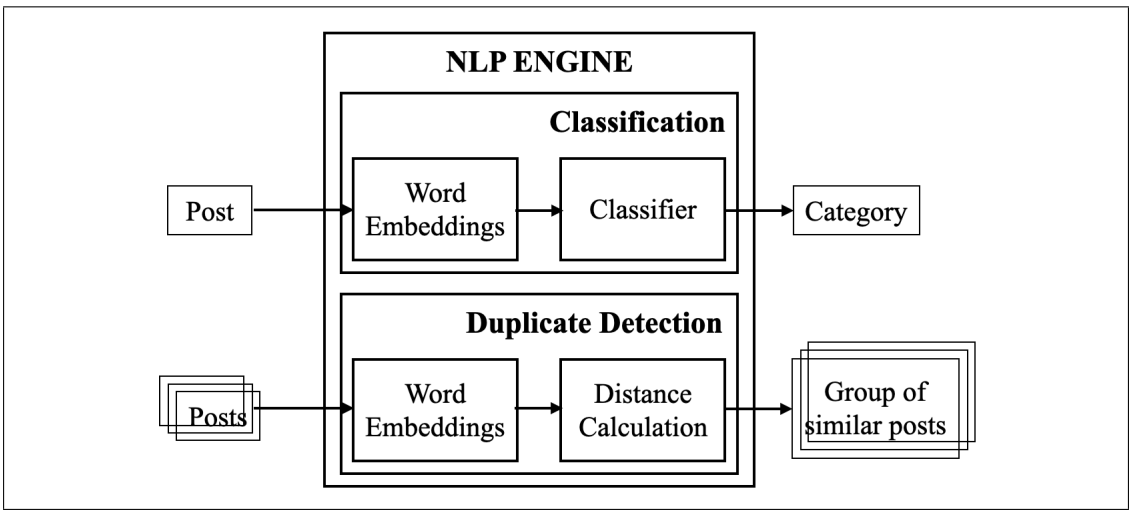


Figure 4.2: NLP Engine Architecture

4.3 State Diagram

The state digram represents the behavior of classes in reponse of other. Posts are fetched from the social media endpoints for a user. Word embedding for each are found out by the NLP engine. These embeddings are used to find duplicates and categorise the posts. Finally the posts are grouped and sent to the user.

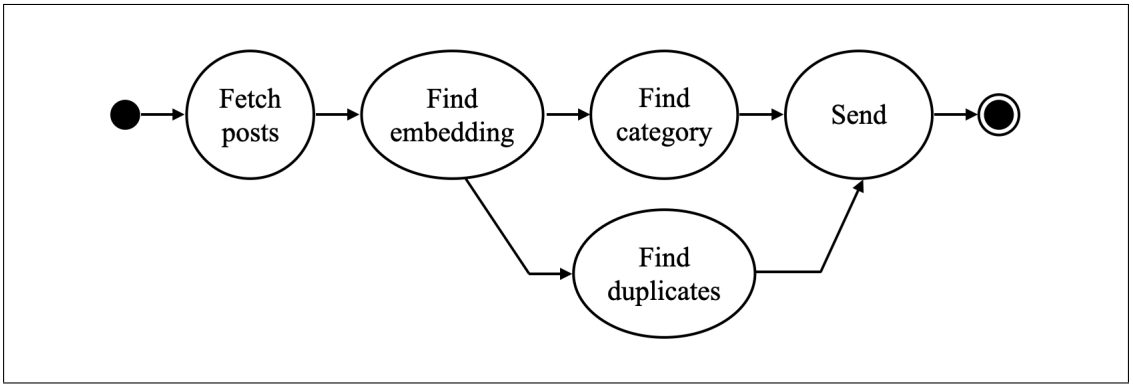


Figure 4.3: State Diagram

4.4 Use Case Diagram

The user interacts with the system to authenticate, categorise, fetch posts and find duplicates. All these operations are passed to the backend which then communicates with social media API and NLP engine.

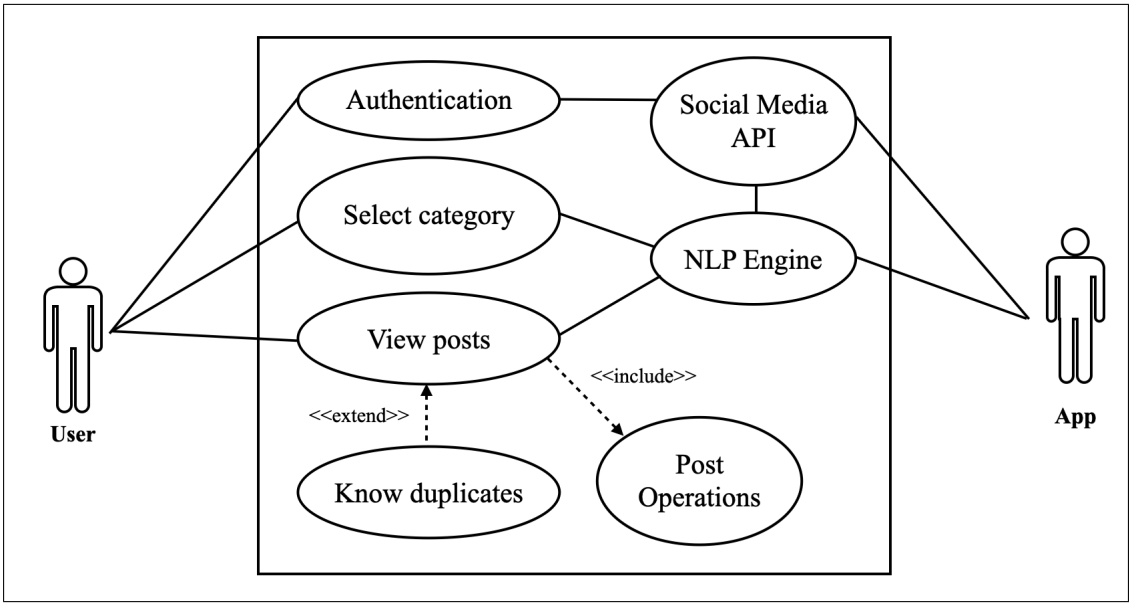


Figure 4.4: Use case Diagram

4.5 Class Diagram

The class diagram shows the various classes, its attributes and operations. Also it provides the relationship between these classes. We have two main classes User class which will be used to register new user, authenticate user, add bookmarks; and Post class which will be used to get Category of post and find the duplicates. The other classes are Category to provide the relevant Category name for the given id, SocialMedia to provide the relvant social media name based on the id provided; and Group which helps to update the Word Embedding used in finding the duplicate posts.

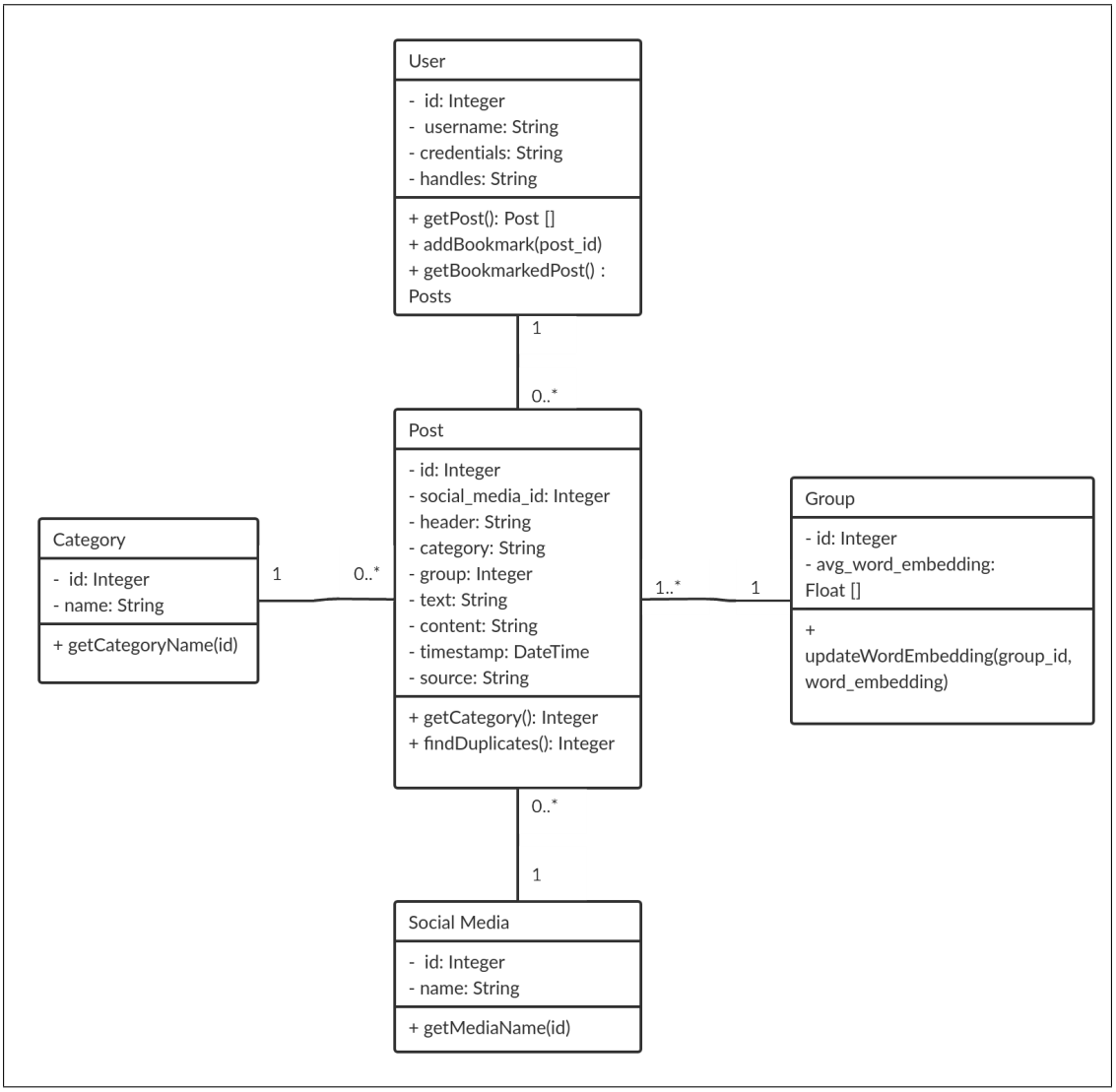


Figure 4.5: Class Diagram

4.6 Entity Relationship Diagram

System consists of users, that is the end user using the application. Each user has username, email id and password which will be used to login the next time. Every user has an entry in tokens table which is used to store the access tokens and anchor id for all the linked social media. User has multiple posts which will be classified to a particular category, and it may also be detected as a duplicate post. Posts can have many attributes as bookmark, local id, user id, group id and timestamp. The Group id is a foreign key to the group table which gives the embedding for the post, and its category, which helps in duplicate detection. Every Post have an entry in Post Details table which gives the handle name and handle post id, to which the post belongs to. All the selected categories for which the user want to see feed, is stored in the Category table.

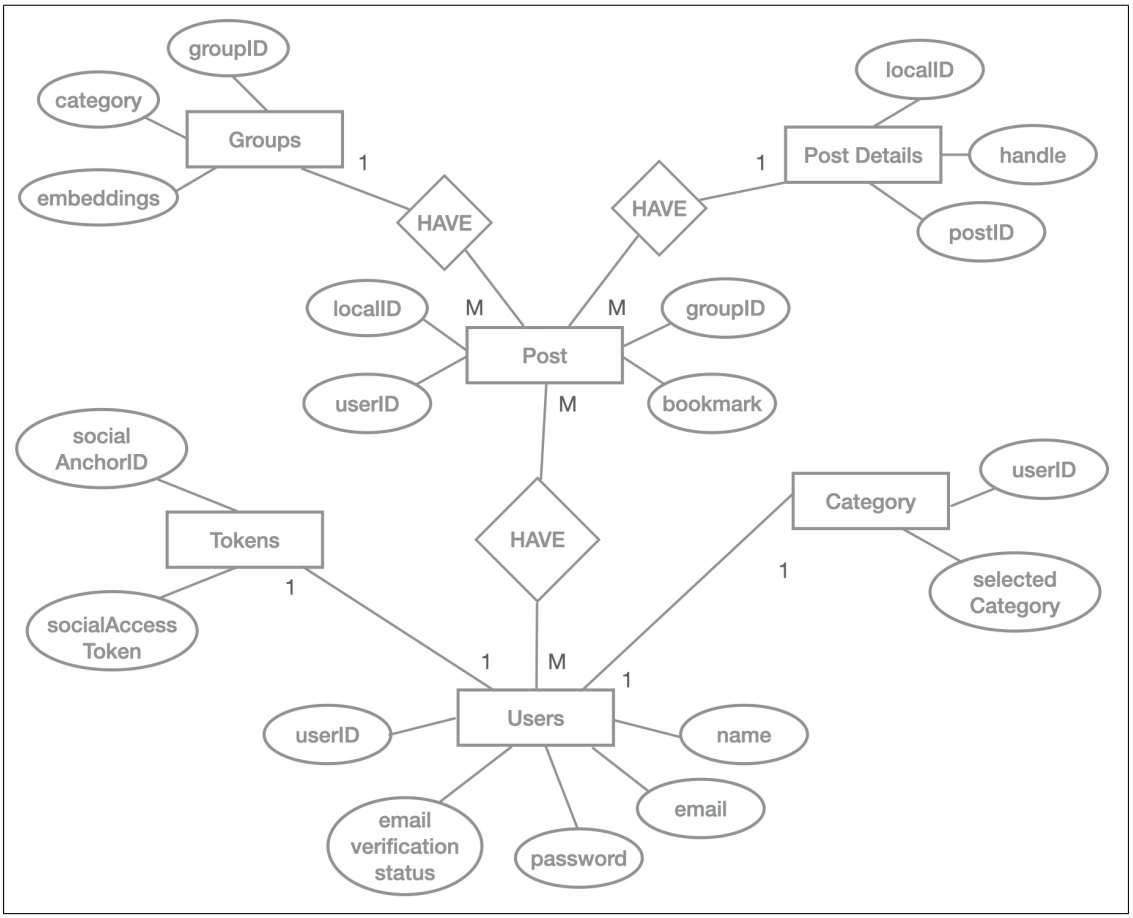


Figure 4.6: Entity Relationship Diagram

4.7 Data Flow Diagram

The data flow diagram shows how the data will move in the application. It has one main entity as the user, which can register, login or request posts for particular category. If the user registers, then the relevant information is stored in the database, and if he logins, the credentials are validated. If the credentials are matched, the users post will be fetched, categorized and checked for duplicates. When the user request posts for a particular category, it will lookup the user-post table, and find the corresponding post of the requested category, and return these to the frontend.

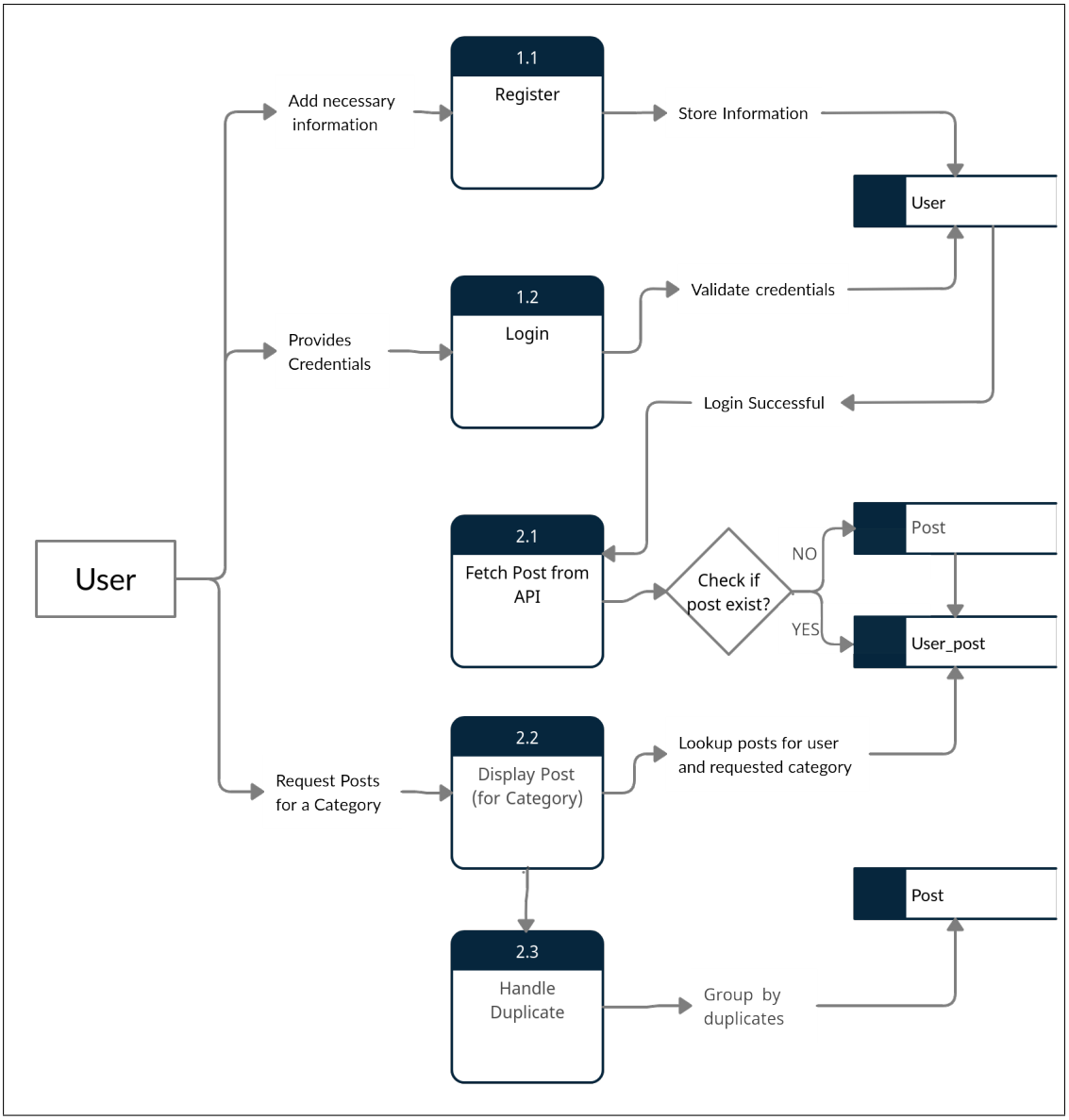


Figure 4.7: Data flow Diagram

CHAPTER 5

PROJECT PLAN

5.1 Project Estimates

5.1.1 Reconciled Estimates

- Cost Estimate: The software used for the application will incur no cost as it is all released under free open source licenses. Approximately Rs 5000 monthly will be required to host the application. As the number of users grow, we may need to scale, which may add to the cost.
- Time Estimate: 12-13 months with continuous maintenance and adding new features after release.

5.1.2 Human Resources

- Number of people : 4
- Skills: Natural Language Processing, Python, SQL(Postgres), NodeJS, ReactJS.
- Client: In our project, the users can belong to any age group, or section who have account on any one of the social media namely Twitter, Reddit or Facebook.
- Stakeholders: Stakeholders for the project will be the people who will be using the application, i.e, the clients, along with the project team and mentor.

5.1.3 Development Resources

- Data Collection: We selected 13 different categories to classify posts. To create the dataset, we sourced posts from 10-12 social media handles per category (from Reddit, Facebook and Twitter) which were determined to exclusively post content belonging to that particular category, and labelled them accordingly. Then, the labelling errors, if any, were corrected. Duplicate posts were sourced by querying multiple handles for the same content topic, both within and across the three social media networks.
- Software:
    1. Python libraries (NLTK, Tensorflow, Keras, Trax, Spacy)
    2. Flask
    3. NodeJS
    4. ReactJS libraries
    5. PostgresQL

5.2 Risk Management

5.2.1 Risk Identification

- User Data Protection: Given our aim of optimizing the product's users' social media feeds, we needed to access their social media data, and perform actions on behalf of their accounts which included retrieving posts from accounts they follow, liking and un-liking posts. We also needed to store the users' authentication information in the form of Access Tokens to efficiently use the social media API.
- Social Network API Access: The application is dependent on the APIs provided by the different online social media networks we are aggregating. Some APIs may be deprecated or revoke access in the future, which means the application will require continued maintenance.

5.2.2 Risk Probability

Category	Probability
High	$0.75 \leq x \leq 1$
Medium-High	$0.5 \leq x < 0.75$
Medium-Low	$0.25 \leq x < 0.5$
Low	$0 \leq x < 0.25$

5.2.3 Risk Analysis and Management

Risk Analysis is an important tool that provides an estimate on likelihood of a risk. It is derived from the probability of the risk and its impact.

Risk ID	1
Risk Description	User Data Protection
Category	User Privacy
Source	Software Requirements Specification
Probability	Low
Impact	Medium
Response	Mitigate
Strategy	Encrypting sensitive data before storing.
Risk Status	Identified and resolved

Risk ID	2
Risk Description	Social Network Access
Category	Application Development
Source	Feasibility Study
Probability	Low
Impact	High
Response	Mitigate
Strategy	Proper error handling and timely maintenance
Risk Status	Identified

5.3 Project Schedule

5.3.1 Task Set

Task	Start Date	End Date
Domain Selection And Literature Survey	15/07/2020	13/08/2020
Problem definition	14/08/2020	28/08/2020
Feasibility Study	29/08/2020	17/09/2020
Creation of UML Diagrams	18/09/2020	07/10/2020
Creation of Wire-frames	08/10/2020	17/10/2020
Dataset Creation	18/11/2020	01/11/2020
Implementation of NLP models and server side script	02/11/2020	09/02/2021
Implementation of client side script	10/02/2021	11/03/2021
Improvement of model	12/03/2021	10/04/2021
Research Paper	01/04/2021	05/05/2021
Testing	20/04/2021	15/05/2021

Table 4: Task Set

5.3.2 Gantt Chart

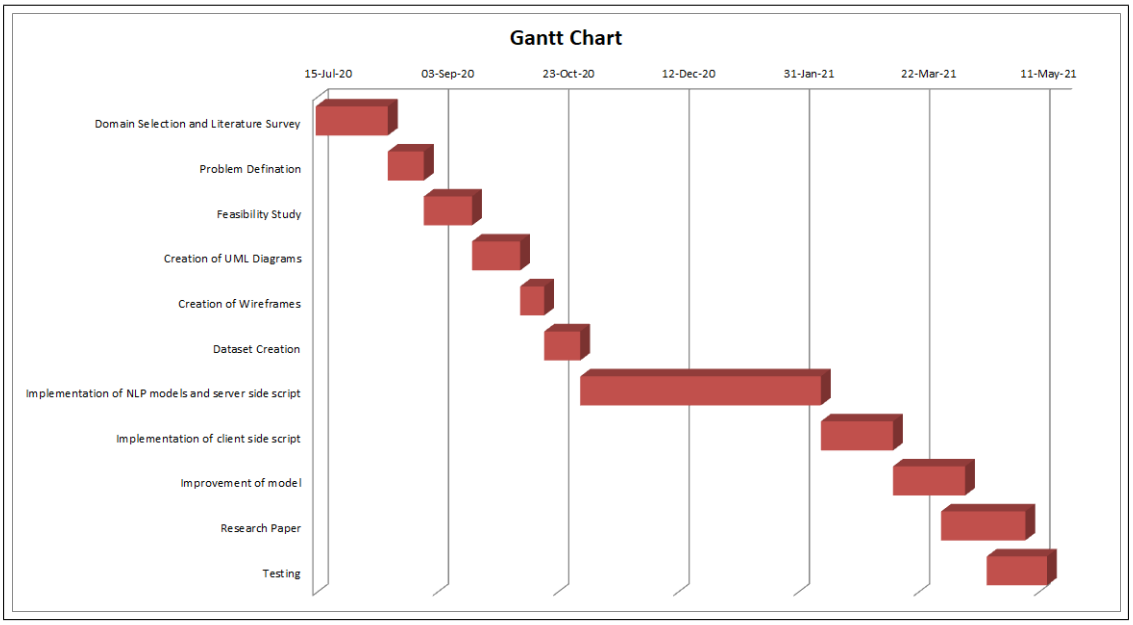


Figure 5.1: Gantt Chart to represent schedule

5.4 Team Organization

The manner in which a team is organized and the mechanisms for reporting are noted. Updates regarding the improvement of project are given to the guide. Also once or twice a month,a meet is held with the project guide regarding updates.

5.4.1 Team Structure

The team structure for the project has been identified and the following roles have been defined:

- Prashant Agrawal: Server side application & integration with NLP model and client side.
- Swapnil Markhedkar: Server side application & handling of database.
- Shruti Phadke: UX design & implementation of client-side.
- Sudhanshu Bhoi: Natural Language Processing Models experimentation & implementation

5.4.2 Management reporting and Communication

- Team is always in contact with the project guide.
- Team members also collaborate in-person and work together to improve efficiency. Also, chat groups, mails, and online meetings are used for communication.
- Used Kanban board on Github to keep track of our progress, and get list of tasks which are completed and which needs to completed in organised manner. Tasks are categorized into broad categories for better issue tracking.

CHAPTER 6

PROJECT IMPLEMENTATION

6.1 Overview of Project Modules

Main Pipeline

1. Retrieving Posts: Posts for user from various social media platforms are fetched after certain interval using APIs
2. Storing values into database (or cache)
   (a) Each User ID and a list of their Post IDs is stored into a database
   (b) Each Post and its corresponding data is queued for the NLP engine to process
3. Post data is passed onto the NLP engine for classification and finding duplicates
4. Storing NLP results into the database
   (a) Category for each Post ID processed is stored
   (b) List of duplicate posts is maintained for each user
5. Caching: If the incoming Post ID (irrespective of the user) is already processed by the NLP engine, the results are directly returned
6. Required information like list of categories for a user, list of post in each category for a user, list of duplicate posts is shared to the app using appropriate API endpoints

6.1.1 Post Categorisation

From the various social media the user gives the application access to, posts are retrieved as mentioned above. To optimize the feed, the first step is to categorise the posts received into one of predetermined 13 categories. The categorisation is performed via natural language processing of the text present in the post. The process to do the same is as follows:

1. Accept only text of a post
2. Categorises text using NLP
3. Returns the category

6.1.2 Duplicate Post Detection

To optimize the feed, the second step is to detect whether any other retrieved posts are duplicates of one another. The duplicate detection is also based on the text present in the post. The process to do the same is as follows:

1. Converts given text to embedding
2. Accepts embedding of a post and embedding of existing groups
3. Returns duplicate group id if available else -1

6.2 Tools and Technologies

6.2.1 Programming Languages

- The client and server side was written using Javascript.
- The NLP models were written in python 3.7

6.2.2 NLP Tech Stack

- TensorFlow
- Trax
- Keras
- NLTK
- sciPy
- Numpy
- Pandas
- Flask

6.2.3 Frontend Tech Stack

- ReactJS for the UI
- Material UI Framework
- React Typist

6.2.4 Backend Tech Stack

- node.js
- PostgreSQL
- Sequelize
- Nodemailer
- JsonWebToken

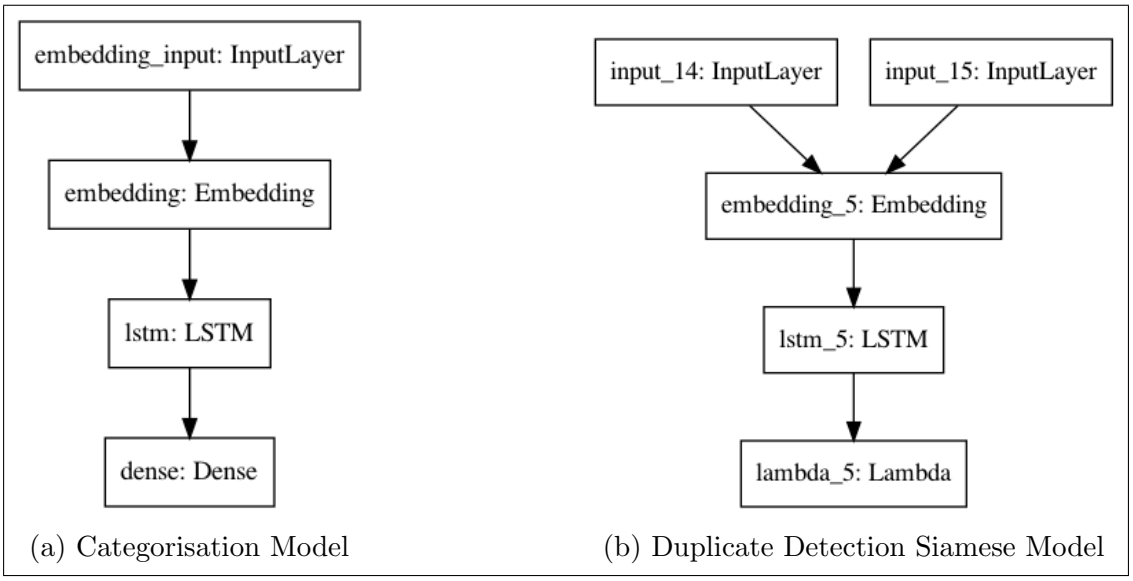
6.3 Algorithms

6.3.1 Post Categorisation

- Preparing Dataset: First we curated social media handles that specialise in a category of post. Then we fetched posts from these handles and assigned the corresponding category. Dataset thus prepared has 13 categories: Business, Celebrity, Entertainment, Finance, Gaming, Health, Motivation, News, Politics, Promotions, Sports, Technology, Travel. Each of these categories have almost 500 examples.
- Training: First the entire dataset was preprocessed by lowercasing, removing URLs, mentions, hashtags, punctuation, expanding contractions, tokenising, removing stop word and lemmatising. A label encoder was trained using the target categorises. Vocabulary was created and using this, texts were converted into vectors. These weer then split into 80:20 train-test split. Training set was then used for training a keras model consisting of embedding, LSTM, dense and softmax layers. The model was compiled with 0.001 learning rate Adam optimizer, categorical cross entropy loss and accuracy metric. The model was stopped training as training loss overtakes validation loss.
- Evaluation: Given text was preprocessed while training. Same vocabulary was used to convert them to vectors. These were then passed through the trained model. Finally, category was obtained by using reverse transform of the trained label encoder.

6.3.2 Duplicate Detection

- Dataset: We use Quora Question Pairs dataset for this task. It consists of 2 questions and a label indicating whether they are duplicate or not. For our case, we only consider the duplicate pairs for training. The training set



(a) Categorisation Model (b) Duplicate Detection Siamese Model

Figure 6.1: Model Architecture

composes of 55989 examples while the test set consists of 18664 duplicate and 18664 non-duplicate question pairs.

- Training: All the question pairs are preprocessed by only considering ASCII characters, converting to lower cases and tokenising based on 10000 word vocabulary generated. These sequences are then padded to maximum length of 300 before passing to the model. The model is a Siamese architecture with each of the parallel networks consisting of an embedding, LSTM and normalise layer. The model is compiled with 0.01 learning rate Adam optimiser and triplet loss. 0.125 of the training set is used for validation. The model is trained for 10 epochs.

- Evaluation: Given text is preprocessed as while training. After padding the sequence is passed to the model. The model outputs embeddings. Cosine similarity between these and a threshold value determines whether 2 given post texts are duplicates.

CHAPTER 7

SOFTWARE TESTING

7.1 Types of Testing

Performed UI Testing, Unit testing, and Performance testing.
The results of the UI and Unit testing are mentioned in section 7.2 below.
The results of Performance testing, performed via Lighthouse, were as follows:

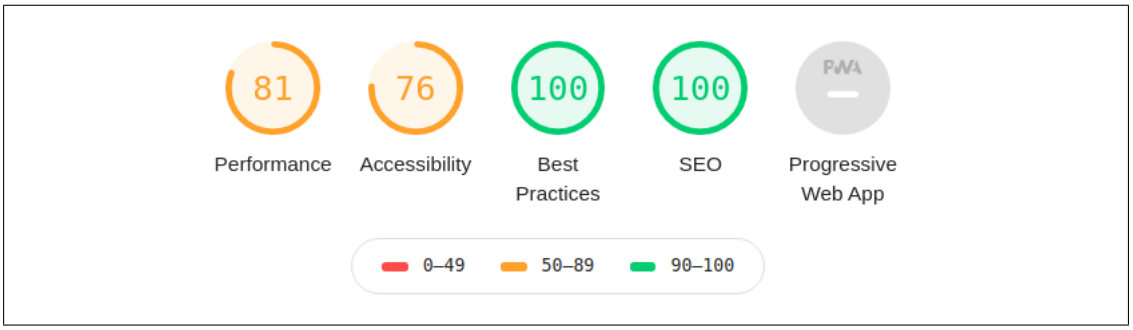


Figure 7.1: Performance Testing Results

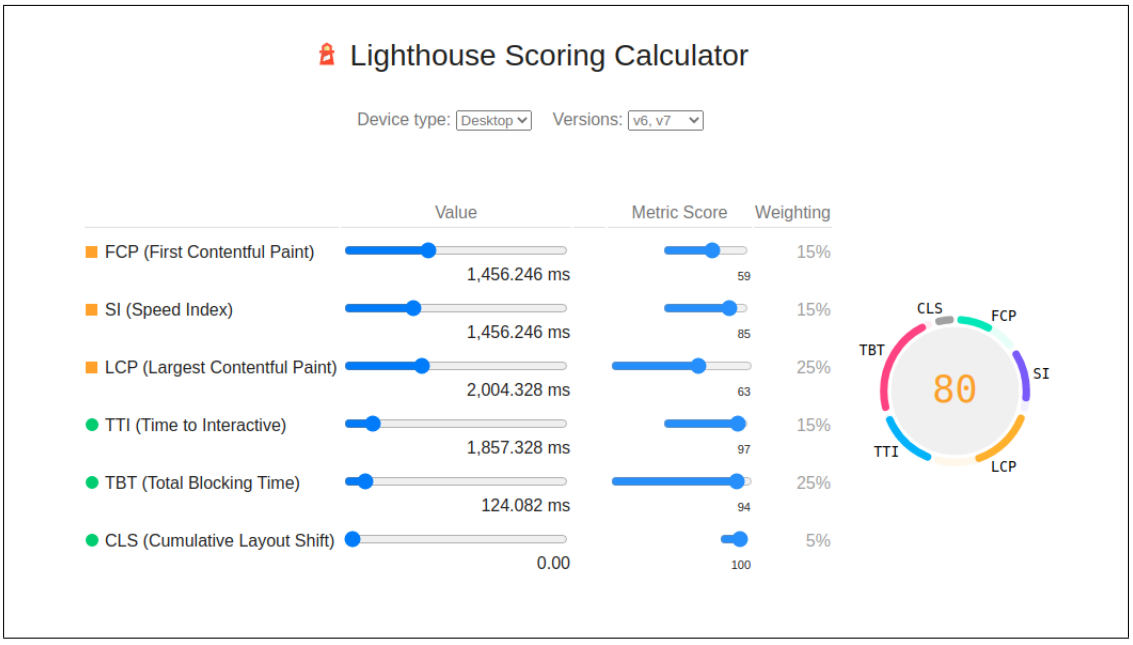


Figure 7.2: Performance Testing Metrics

7.2 Test Cases & Results

Sr. No	Test Case	Expected Output	Actual Output	Result
1	Signup form appropriately validated	Yes	Yes	Pass
2	Login form email validation	Successfully detects incorrect email formats	Successfully detected incorrect email formats	Pass
3	User tries to access feed before linking social media account	Feed does not show until at least one social media is linked	Feed not show until at least one social media is linked	Pass
4	User 'likes' a post shown on the feed	'like' should be conveyed and shown on original social media page	'like' conveyed and shown on original social media page	Pass
5	User bookmarks a post shown on feed	Post should be added to list of bookmarks	Post added to list of bookmarks	Pass
6	User opts to delete account	Information of all 3 social media networks for that user gets deleted	Information of all 3 social media networks for that user is deleted	Pass
7	User's feed contains a post that the product has detected duplicates for	Duplicate detected icon should be visible and display duplicate posts when clicked	Duplicate detected icon is visible and displays duplicate posts when clicked	Pass
8	User chooses to view posts from a different category	Feed updates according to click	Feed updated according to click	Pass
9	User chooses to revoke access from one or more social media network	User data for those social networks should be removed and feed should be updated accordingly	User data for those social networks removed and feed updated accordingly	Pass
10	User wants to add or remove categories set previously	Changes should reflect in feed	Changes are reflected in feed	Pass
11	User reaches the bottom of feed	More posts should be fetched and shown	More posts are fetched and shown	Pass

Table 5: Test Cases

CHAPTER 8

RESULTS

8.1 Web Application Screenshots

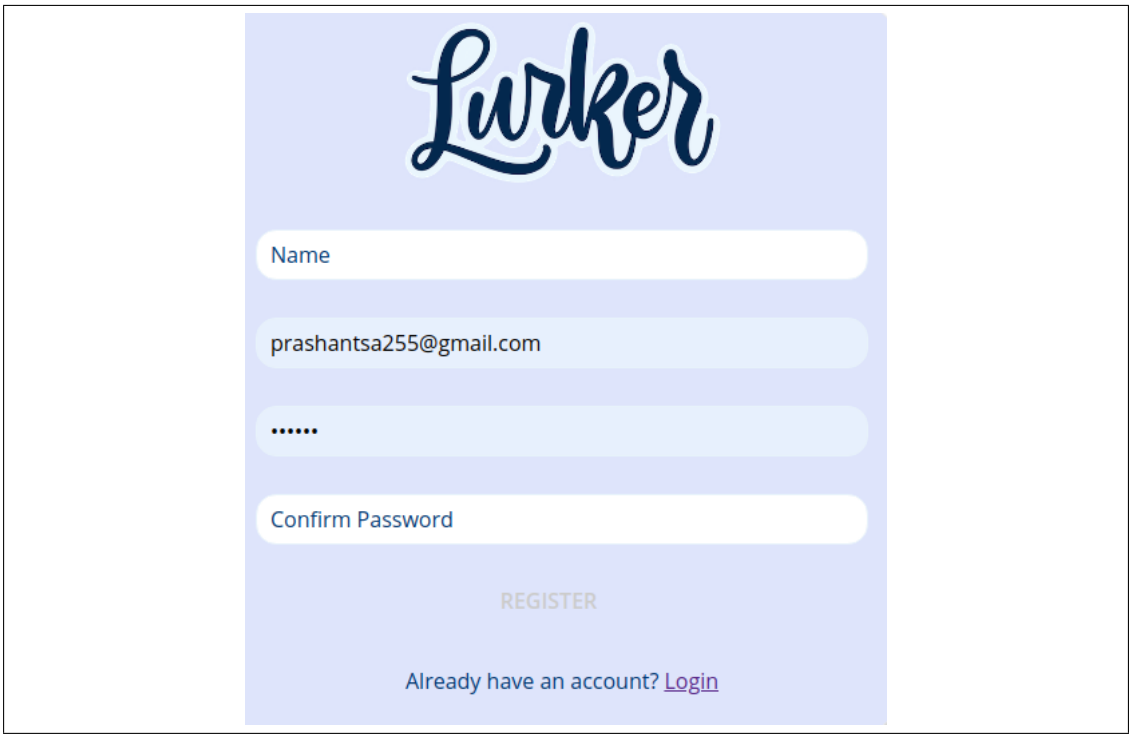


Figure 8.1: Signup Page

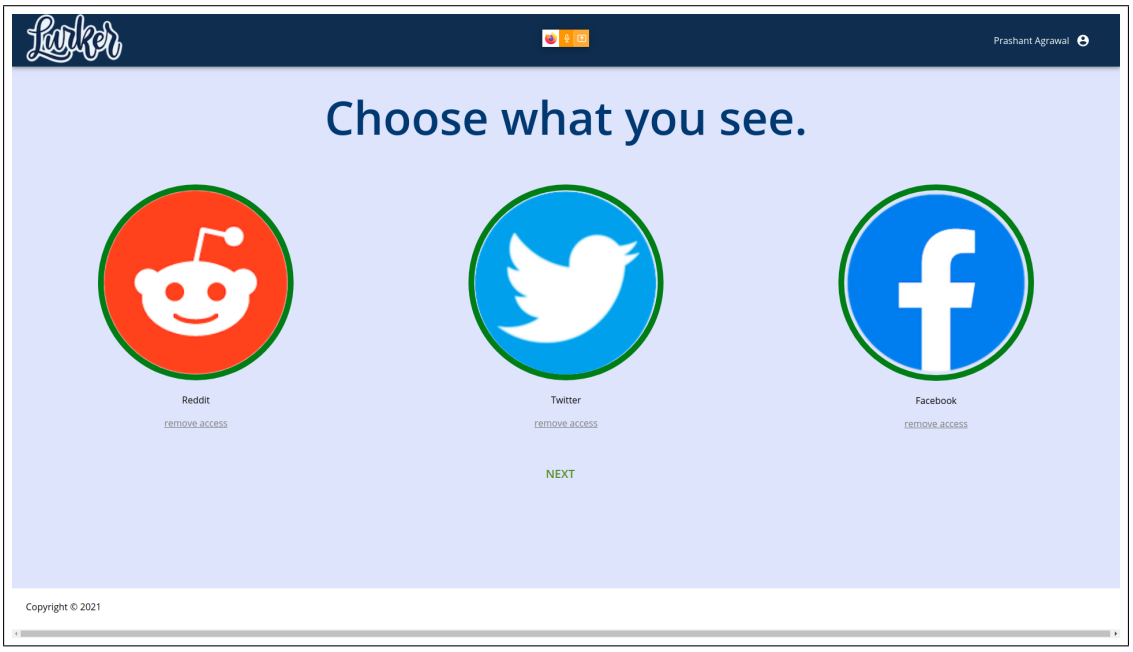


Figure 8.2: Link various social media

Figure 8.3: Select categories for feed

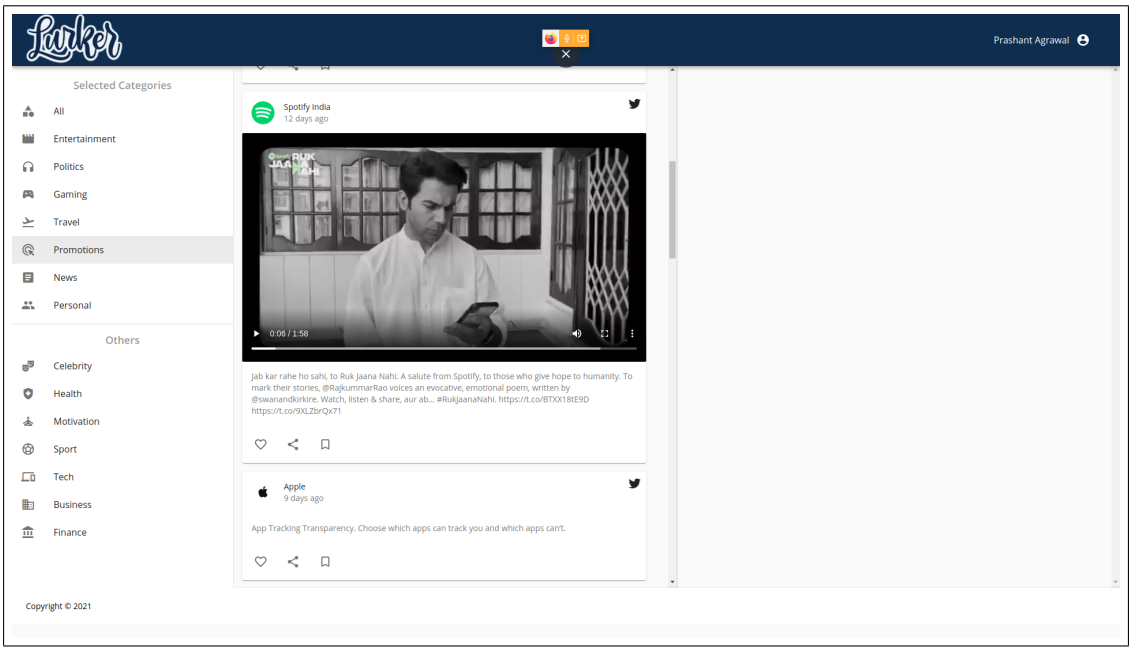


Figure 8.4: Posts classified in 'Promotions' category

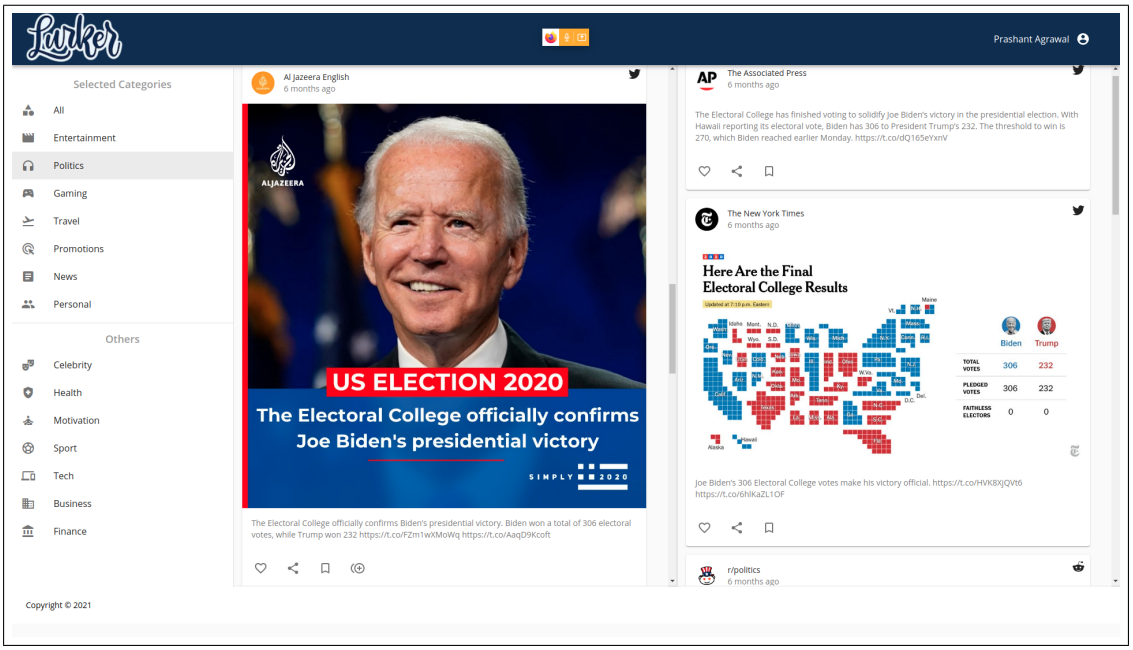


Figure 8.5: Duplicate posts aggregated together

Figure 8.6: About Us

8.2 NLP Model

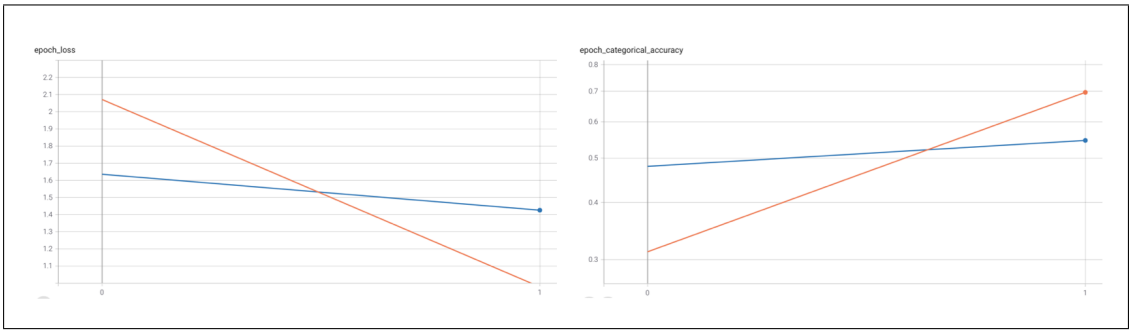


Figure 8.7: Categorical train validation accuracy and cross entropy loss

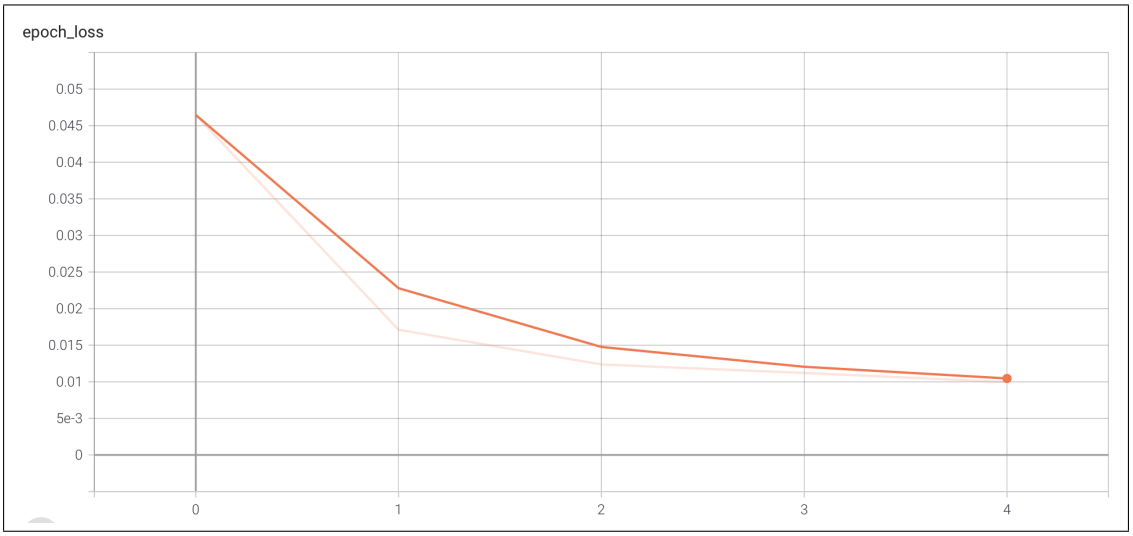


Figure 8.8: Duplicates train triplet loss

Task	Dataset	Accuracy
Categorisation	Custom	0.60
Duplicate Detection	Quora	0.73

Table 6: Model Performance

CHAPTER 9

OTHER SPECIFICATIONS

9.1 Advantages

1. Retains advantages of social media while giving the user sense of purpose.
2. Provides ability to navigate the posts based on ones interest at any given moment of time.
3. Groups duplicate posts together to avoid repetitive posts from appearing in the feed.
4. Saves time and increases productivity.

9.2 Limitations

1. Cannot categorise all the posts correctly.
2. Consists of limited number of categories.
3. Posts with same intent but different expression may not be considered as duplicates.
4. System does not capture the unique aspects of each social media platform.
5. Only text is used and no other forms of content are considered.

9.3 Applications

1. Social media for productive people.
2. Social accounts for visually impaired.
3. Navigating social feed based on interest at instance.

CHAPTER 10

CONCLUSION AND FUTURE WORK

10.1 Conclusion

Developed an application that can fetch posts for a user from multiple social media platforms. These were then grouped together into major categories and duplicates were found out. Finally, it was represented to the user in an UI.

10.2 Future Work

In future, other aspects of social media can be added to the applications. Sophisticated NLP techniques can be used to summarise the posts inside a group.These would not only increase productivity but also help visually impaired to hear posts at a glance. Along with these, better categorization can be achieved by predicting images and videos summary.

APPENDIX A

FEASIBILITY AND MATHEMATICAL MODEL

APPENDIX A

Idea Matrix

SOCIAL MEDIA PLATFORM	TYPE OF POSTS	FEASIBILITY (3: High, 2: Mod, 1: Low)
Twitter	Text, Image, Video, Stories	3
Reddit	Text threads, Image, Video, Broadcast Video	3
Linkedin	Text, Image, Video, Stories	2
Facebook	Text, Image, Video, Stories	2
Instagram	Image, 15s Video, 60s Video, Long Video, Stories	1
Quora	Q&A Text, Image	1
Tumblr	Text threads, Image threads	1

Table 7: Idea Matrix

IDEA	CRITERIA				SCORE (3: High, 2: Mod, 1: Low)
	Doable / Practical	Better than / Different from current apps	Delivers value / Solves a problem	Fits development time constraint	
Combine multiple social media apps	2	2	3	2	2
Categorize posts based on text	3	2	3	3	3
Recognize and combine duplicate posts across platforms	3	3	2	2	2
Summarize all posts for a category	1	3	2	1	1

Table 8: Social Media Feasbility Evaluation

Social Media REST API endpoints

Twitter

- Authenticating user with 3-legged OAuth flow
    1. **POST oauth/request_token**
    2. **GET oauth/authorise**
    3. **POST oauth/access_token**
- **GET /2/users/by/username/:username** to get user_id
- **GET statuses/home_timeline** to fetch the posts from user timeline
- **POST favorites/create** to like a twitter post given its id
- **GET favorites/list** to get all the liked posts
- **POST statuses/retweet/:id** to retweet a tweet from the user's timeline

Reddit

- Authenticating User
    1. **GET /api/v1/authorize**
    2. **POST /api/v1/access_token**
- **GET /api/best** to fetch posts for user
- **POST /api/vote** to upvote a reddit post for a user
- **GET /api/comments** to get comments on the post.

Facebook

- Authenticating user
    1. Configure the app
    2. **GET oauth/access_token**
- **GET /v9.0/:user-id/feed** to get user feed
- **POST /v9.0/:object-id/likes** to like a post
- **POST /:user-id/feed** to post for a user

LinkedIn

- Authenticating using 3-legged flow
    1. Configure the app
    2. **GET oauth/v2/authorization**
    3. **POST oauth/v2/access-token**
- **GET /v2/ugcPosts/:urn** to get the posts by URN
- **GET /v2/shares/:urn** to get shares of a user
- **POST /v2/socialActions/:urn/likes** to like a share

Instagram

- Authenticating user
    1. Configure the app
    2. **POST oauth/access_token** to get access token for a user
    3. **GET /access_token** to get long-lived user access token
- **GET /:user-id/media** to get user timeline
- **GET /:media-id** to get a media item

NLP Algorithms

Siamese Networks

A Siamese Neural Network is a class of neural network architectures that contain two or more identical subnetworks. 'identical' here means, they have the same configuration with the same parameters and weights. Parameter updating is mirrored across both sub-networks. It is used to find the similarity of the inputs by comparing its feature vectors, so these networks are used in many applications.

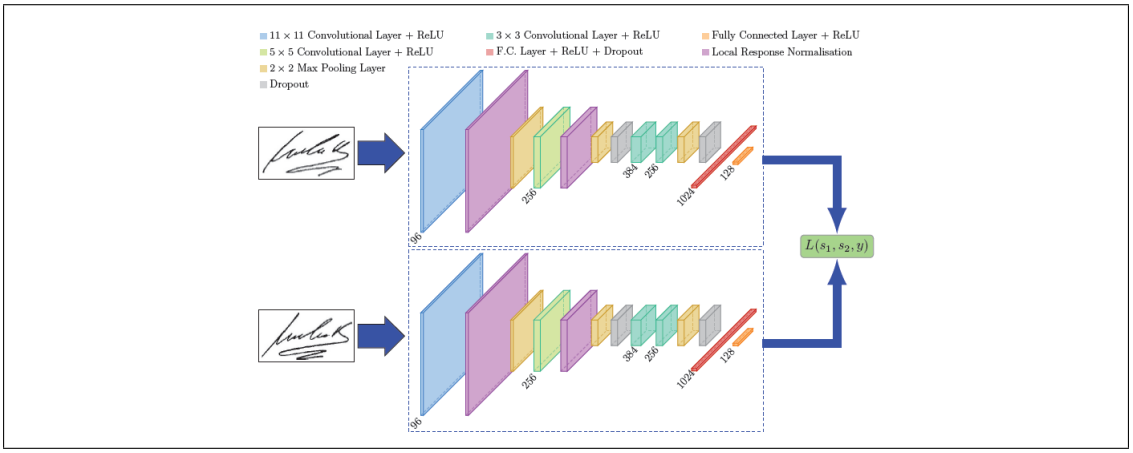


Figure 10.1: Siamese network used in Signet

LSTM

LSTM stand for Long short-term memory and is a recurrent neural network architecture. It has feedback connections to process entire sequence of data. Unlike RNN, LSTM makes a small modifications to information by multiplication and addition, this allows it to selectively remember and forget things. It consists of different memory blocks called cells. There are 2 states being transferred to the next cell: the cell state and hidden state. There are 3 different gates: forget gate, input gate and output gate.

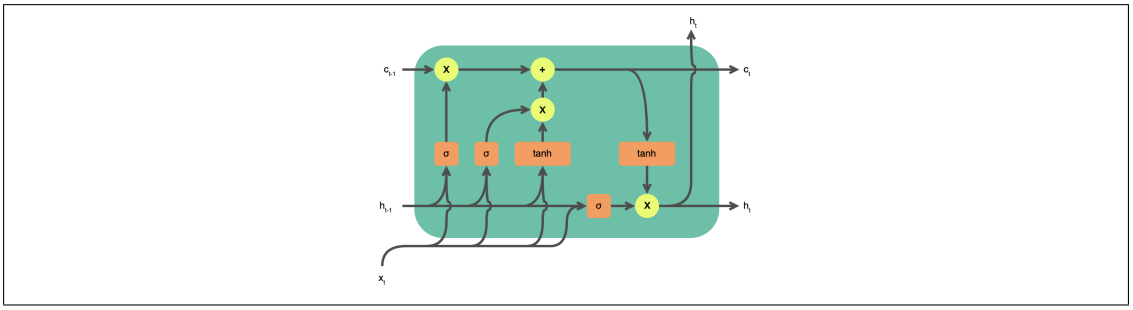


Figure 10.2: LSTM Cell

Mathematical Model

Backend Engine

Let, at any given time t,
U be the number of users
S_i be the number of social media accounts for ith user
P_{ij} be the number of post of ith user and his/her jth social media account
G_i be the number of groups of duplicate posts for ith user
D_{ik} be the number of duplicate posts for ith user and kth group of duplicates

Total number of API requests made at t:
$A = \sum_{i=1}^{U} S_i$
Assumption: Only one API request is made to fetch all the required posts from a social media account for a user.

Total number of posts received and stored in the database at time t:
$P = \sum_{i=1}^{U} \sum_{j=1}^{S_i} P_{ij}$

Total number of duplicate posts at time t:
$D = \sum_{i=1}^{U} \sum_{k=1}^{G_i} D_{ik}$
where D $\leq$ P

Let, for a post,
f be the time required to fetch
q be the time required to start processing from the queue
w be the time required to find word-embeddings
d be the time required to compare with P_i posts of ith user
c be the time required to classify
r be the time required to combine all the results

Total time required for a post to reach the user:
$T = f + q + w + d + c + r$

NLP Engine

Word2Vec

$$argmax \prod_{w \epsilon T} [\prod_{c \epsilon c(w)} p(c|w;\theta)] \tag{1}$$

where T refers to Text, and θ is parameter of $p(c|w;\theta)$.

LSTM

$$i_t = \sigma(w_i[h_{t-1}, x_t] + b_i) \tag{2}$$

$$f_t = \sigma(w_f[h_{t-1}, x_t] + b_f) \tag{3}$$

$$o_t = \sigma(w_o[h_{t-1}, x_t] + b_o) \tag{4}$$

where i_t, f_t, o_t are input, forget and output gates, σ is sigmoid function, w_x represent weight for respective gate(x) neurons, h_{t-1} is output of previous lstm block, x_t is input at current timestamp and b_x is biases for the respective gate(x).

Cross-Entropy Loss

$$L_{CE} = -\sum_{i=1}^{n} t_i log(p_i) \tag{5}$$

for n classes where t_i is truth label and p_i is the Softmax probability for i^{th} class.

Triplet Loss

$$cost_i^1 = max(-cos(f(x_i^a), f(x_i^p)) + mean\_neg + \alpha, 0) \tag{6}$$

$$cost_i^2 = max(-cos(f(x_i^a), f(x_i^p)) + closest\_neg + \alpha, 0) \tag{7}$$

$$cost_i = cost_i^1 + cost_i^2 \tag{8}$$

$$TripletLoss = \sum_{i}^{N} cost_i \tag{9}$$

where x_i^a, x_i^p, x_i^n are anchor, positive, negative inputs respectively. $f(x)$ is the embedding from the network. α is the margin between positive and negative pairs. N is the batch-size.

APPENDIX C

PLAGIARISM REPORT

Plagiarism Report

Curiginal

Document Information

Analyzed document	Group_55_Final_Report (1).pdf (D108523077)
Submitted	6/10/2021 5:56:00 PM
Submitted by	Ranjeet Bidwe
Submitter email	rvbidwe@pict.edu
Similarity	0%
Analysis address	rvbidwe.pict@analysis.urkund.com

Sources included in the report

REFERENCES

References

[1] Clement, J. *"Worldwide Daily Social Media Usage by Region 2019."* Statista, 12 Aug. 2019, www.statista.com/statistics/1031948/global-usage-duration-of-social-networks-by-region/.

[2] Allen, Mike. *"Sean Parker Unloads on Facebook: 'God Only Knows What It's Doing to Our Children's Brains.'"* Axios, 15 Dec. 2017, www.axios.com/sean-parker-unloads-on-facebook-2508036343.html.

[3] Nestler, Eric J. *"Transcriptional Mechanisms of Drug Addiction."* Clinical Psychopharmacology and Neuroscience : the Official Scientific Journal of the Korean College of Neuropsychopharmacology, Korean College of Neuropsychopharmacology, Dec. 2012, www.ncbi.nlm.nih.gov/pmc/articles/PMC3569166/.

[4] Fei, Geli and Liu, Bing. (2015). Social Media Text Classification under Negative Covariate Shift. 2347-2356. 10.18653/v1/D15-1282.

[5] Pinto, Alexandre and Gonçalo Oliveira, Hugo and Alves, Ana. (2018). Comparing the Performance of Different NLP Toolkits in Formal and Social Media Text. 51. 3:1-. 10.4230/OASIcs.SLATE.2016.3.

[6] G. Chen, D. Ye, Z. Xing, J. Chen and E. Cambria, "Ensemble application of convolutional and recurrent neural networks for multi-label text categorization," 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, 2017, pp. 2377-2383, doi: 10.1109/IJCNN.2017.7966144.

[7] Gargiulo, Francesco and Silvestri, Stefano and Ciampi, Mario. (2018). Deep Convolution Neural Network for Extreme Multi-label Text Classification. 641-650. 10.5220/0006730506410650.

[8] Suppawong Tuarob, Conrad S. Tucker, Marcel Salathe, Nilam Ram, An ensemble heterogeneous classification methodology for discovering health-related knowledge in social media messages, Journal of Biomedical Informatics, Volume 49, 2014, Pages 255-268, ISSN 1532-0464, https://doi.org/10.1016/j.jbi.2014.03.005.

[9] Liu, H., Yin, Q., and Wang, W. Y., "Towards Explainable NLP: A Generative Explanation Framework for Text Classification", arXiv e-prints, 2018.

[10] Anna Stavrianou, Caroline Brun, Tomi Silander, and Claude Roux. 2014. NLP-based feature extraction for automated tweet classification. In Proceedings of the 1st International Conference on Interactions between Data Mining and Natural Language Processing - Volume 1202 (DMNLP'14). CEUR-WS.org, Aachen, DEU, 145–146.

[11] Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., and Gao, J., "Deep Learning Based Text Classification: A Comprehensive Review", arXiv e-prints, 2020.

[12] Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L. E., and Brown, D. E., "Text Classification Algorithms: A Survey", arXiv e-prints, 2019.

[13] Sood, Sanjay and Hammond, Kristian and Owsley, Sara and Birnbaum, Larry. (2007). TagAssist: Automatic tag suggestion for blog posts.

[14] Jochen Hartmann, Juliana Huppertz, Christina Schamp, Mark Heitmann, Comparing automated text classification methods, International Journal of Research in Marketing, Volume 36, Issue 1, 2019, Pages 20-38, ISSN 0167-8116, https://doi.org/10.1016/j.ijresmar.2018.09.009.

[15] Howard, J. and Ruder, S., "Universal Language Model Fine-tuning for Text Classification", arXiv e-prints, 2018.

[16] Ive, Julia and Gkotsis, George and Dutta, Rina and Stewart, Robert and Velupillai, Sumithra. (2018). Hierarchical neural model with attention mechanisms for the classification of social media text related to mental health. 69-77. 10.18653/v1/W18-0607.

[17] Kinsella S., Wang M., Breslin J.G., Hayes C. (2011) Improving Categorisation in Social Media Using Hyperlinks to Structured Data Sources. In: Antoniou G. et al. (eds) The Semanic Web: Research and Applications. ESWC 2011. Lecture Notes in Computer Science, vol 6644. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-21064-827

[18] Nivedha, R. and Sairam, N.. (2015). A Machine Learning based Classification for Social Media Messages. Indian Journal of Science and Technology. 8. 10.17485/ijst/2015/v8i16/63640.

[19] Dönicke, T., Florian Lux and Matthias Damaschk. "Multiclass Text Classification on Unbalanced, Sparse and Noisy Data." (2019).

[20] Duong, Chi and Lebret, Rémi and Aberer, Karl. (2017). Multimodal Classification for Analysing Social Media.

[21] Ghaly, Rehab and Elabd, Emad. (2016). Tweets classification, hashtags suggestion and tweets linking in social semantic web. 1140-1146. 10.1109/SAI.2016.7556121.

[22] Wang, Yiren and Tian, Fei. (2016). Recurrent Residual Learning for Sequence Classification. 938-943. 10.18653/v1/D16-1093.

[23] A. Abid, W. Ali, M. S. Farooq, U. Farooq, N. S. Khan and K. Abid, "Semi-Automatic Classification and Duplicate Detection From Human Loss News Corpus," in IEEE Access, vol. 8, pp. 97737-97747, 2020, doi: 10.1109/ACCESS.2020.2995789.

[24] Mamata Shrestha, Mohit Dhungana, Subin Panta, Ujjwal Pudasaini, Santosh Giri. Duplicate Question Detection (DQD) using Siamese LSTM. NCE Journal of Science and Engineering, Volume I, Issue I, February 2020.

[25] Z. Imtiaz, M. Umer, M. Ahmad, S. Ullah, G. S. Choi and A. Mehmood, "Duplicate Questions Pair Detection Using Siamese MaLSTM," in IEEE Access, vol. 8, pp. 21932-21942, 2020, doi: 10.1109/ACCESS.2020.2969041.

[26] J. Gao, Y. He, X. Zhang and Y. Xia, "Duplicate short text detection based on Word2vec," 2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS), Beijing, 2017, pp. 33-37, doi: 10.1109/ICSESS.2017.8342858.

[27] A. K. Layek, A. Gupta, S. Ghosh and S. Mandal, "Fast near-duplicate detection from image streams on online social media during disaster events," 2016 IEEE Annual India Conference (INDICON), Bangalore, 2016, pp. 1-6, doi: 10.1109/INDICON.2016.7839137.

[28] C. Saedi, J. Rodrigues, J. Silva, V. Maraev, "Learning profiles in duplicate question detection," in 2017 IEEE International Conference on Information Reuse and Integration (IRI). IEEE, 2017, pp. 544– 550.