

ASPECT BASED SENTIMENT ANALYSIS ANDROID APP FOR FINANCIAL NEWS

Saptaparna Ghosh

Assistant Professor, Institute of Engineering and Management, Kolkata
and University of Engineering and Management, Kolkata.

Dr Mitali Sengupta

Associate Professor, Institute of Engineering and Management, Kolkata and University of
Engineering and Management, Kolkata.

Dr Satyajit Chakrabarti

Professor, Institute of Engineering and Management, Kolkata and University of Engineering and
Management, Kolkata.

Abstract: This research addresses the crucial need to adjust to the rapidly changing financial environments through a mobile application developed on Android that effectively incorporates innovative machine learning algorithms. Targeting the sentiment analysis of the financial news, the app applies aspect-based sentiment analysis, predictive modeling, and also predictive eligibility assessment for loans or credit cards. The aspect-based approach provides a deeper understanding, while the predictive modeling enables round-the-clock planning. The research helps in democratizing financial information and decision-making, making advanced analytics accessible to the user no matter their level of expertise in finance. Building on a user-friendly interface, the Android app arises as an innovator leading at the crossroads of technology and finance by giving many unique insights into financial decisions.

Keywords: Android Application, Machine Learning, Sentiment Analysis, Predictive Modeling, Aspect-Based Sentiment Analysis, Financial News, Predictive Eligibility Assessment, Democratization of Financial Information.

1. INTRODUCTION:

The imperative need to address unfavorable financial setbacks becomes increasingly apparent in the dynamic landscape shaped by the ever-evolving realm of technology. This continuous transformation introduces novel solutions and concepts that redefine the intersections between finance and information analysis. The research endeavors to emphasize the significance of adapting to this change by introducing an Android mobile application that integrates cutting-edge machine learning algorithms without limitations. The primary objective is to craft a sophisticated instrument for sentiment analysis tailored specifically for financial news. In the age of instant information, users seek more than raw data—they crave insights that fortify their decision-making. This Android application strives to fulfill this demand by providing real-time and comprehensive sentiment reviews of financial data. Harnessing the power of machine learning, the app aims to

empower users with a nuanced understanding of the sentiments underlying financial news, thereby facilitating more informed investment decisions.

One noteworthy feature of the application is its predictive eligibility assessment for loans or credit cards. Going beyond traditional sentiment analysis, predictive modeling algorithms are incorporated to enable users to predict and define their eligibility for various financial products. This forward-looking feature positions the app as an all-encompassing financial companion, catering to the rapidly changing demands of users navigating personal and business finance in a fast-paced environment.

Central to the research is the utilization of aspect-based sentiment analysis. Diverging from conventional sentiment analysis that provides a general sentiment score, aspect-based sentiment analysis dissects the content to discern sentiments associated with specific aspects. In the context of financial news, this approach aids users in understanding sentiments linked to different types of financial instruments, markets, or economic indicators. This granular insight equips users with a more nuanced perspective, empowering them to make sounder decisions in their investments.

Furthermore, the integration of predictive modeling elevates the app's capability for 24-hour forward planning. Users can not only assess current sentiments but also predict potential shifts, adding a proactive dimension to their financial decision-making. This forecasting attribute proves invaluable in an environment where financial markets are influenced by numerous variables, emphasizing the importance of making precise choices that can significantly impact business outcomes.

Beyond technological boundaries, the research delves into the democratization of financial information and decision-making. The application aspires to empower individuals, irrespective of their financial expertise, by placing sophisticated sentiment analysis and predictive modeling literally in the palm of the user's hand. The user-friendly interface ensures accessibility, allowing both experienced investors and novices to navigate through the intricacies of financial sentiment analysis with ease.

Overall Android mobile application emerges as an innovative leader at the intersection of technology and finance. Seamlessly combining machine learning algorithms, aspect-based sentiment analysis, and predictive modeling, the app envisions a future where users' base financial decisions on unparalleled insights. Addressing not only current demands but also anticipating evolving user needs, this research aims to provide a robust and intelligent tool for navigating the complexities of the financial world.

1.2 Objective of The Research:

Development of a mobile application on the Android platform that efficiently applies advanced machine learning algorithms to cover the dynamic nature of financial markets.

Sentiment analysis of financial news using aspect-based sentiment analysis methods to provide users with a more thorough view of market sentiment.

To utilize predictive modeling for round-the-clock planning of financial decisions thereby empowering the user decision-making.

To integrate predictive eligibility evaluation for loans or credit cards where users are assisted in choosing the best financial option.

To democratize financial information and decision-making by providing advanced analytics to users regardless of their level of expertise in finance.

1.3 Target Audience

This application is developed for the Android environment and targets people who are passionate about financial management and decision making. This may encompass a broad range of users, including:

Everyday consumers who want to improve their financial literacy, such as budgeting, investing, and borrowing.

Small business owners seeking to streamline their financial operations and take well-informed decisions regarding loans or credit.

Financial professionals such as financial advisors or analysts that can use the app's advanced analytics and predictive modeling to give specific advice to their clients.

People new to finance and students interested in learning more about financial concepts and improving their financial literacy.

2. LITERATURE REVIEW:

Applying Deep Learning Approach to Targeted Aspect-based Sentiment Analysis for Restaurant Domain

This research proposes a deep learning approach for targeted aspect-based sentiment analysis in the restaurant domain [6]. The approach extends the traditional LSTM (Long Short-Term Memory) approach by adding external knowledge from SenticNet, a multi-attentive LSTM (MA-LSTM) model [6]. The results show that the MA-LSTM model outperforms other state-of-the-art LSTM methods, such as standard LSTM, TD-LSTM, TC-LSTM, AE-LSTM, and ATAE-LSTM.

Social Signal Processing for Evaluating Conversations Using Emotion Analysis and Sentiment Detection

This research presents a multi-modality framework for analyzing customer satisfaction levels in industries such as banking and finance [10]. The approach combines emotion analysis of speech

signals, sentiment detection from text data, and facial emotion analysis from image data. The results show that the proposed approach helps in better understanding of customers and identification of their exact perspective with respect to services provided by the industries [10].

A Deep Analysis on Aspect-based Sentiment Text Classification Approaches

This research highlights the insights of extracting the most important aspects from opinions expressed in input text using various machine learning techniques. The study focuses on feature-based sentiment analysis and can be used to identify different aspects expressed at either document or sentence level [7]. The results demonstrate that the proposed approach can be beneficial for businesses, organizations, social media, and e-commerce sites in understanding customer satisfaction.

Customer Loan Approval Classification by Supervised Learning Model

This research focuses on predicting loan repayment status using machine learning models in default loan prediction [8]. The study uses various machine learning models to classify loan approval status. The results demonstrate that machine learning models can be effective in predicting loan repayment status.

An Empirical Study on the Prediction of Farmers' Ability to Acquire Loans in the Yanliang District Based on Cloud Model

This research empirically analyzes the influencing factors of farmers' household characteristics, production characteristics, material capital characteristics, and social relations on their loan ability [9]. The study uses a cloud model to solve the fuzziness of some influencing factors and divides the ability of farmers to obtain loans into five levels [9]. The results show that the model has certain practicability and reliability, providing a method for farmers' credit qualification evaluation.

A Mobile Application for Alternative Credit Scoring

This research will address the growing smartphone penetration and also increasing desperate demands for credit-scoring assessment alternatives, especially for unbanked or under-banked individuals. This research aims to advance a mobile app based on alternative data, including the use of mobile money transactions, utility payments, and e-commerce platforms in order to determine how creditworthy an individual is. The application aims to help the underserved who do not have traditional credit scores to obtain loan approvals, improve the financial situations of the unbanked masses, and streamline lending for SMEs. A convenient, cost-effective, and time-saving method is suggested that represents a viable solution during the COVID-19 pandemic. Finally, the research aims to enable individuals to better financial management and inclusion [11].

Design and Development of a Fast Loan Management System

The research targeted designing and developing a Fast Loan Management System (FLMS) which enables efficient loan processes that include banks as well as online users. Applying HTML, CSS, Java as well as XAMPP, and MySQL the structure becomes a distributed one stored in the common centralized databases. Quick loan origination, approval, and monitoring are enabled by the ease of use. Although the system addresses the financial market requirements, it is also aware of many issues including data security, scalability, and processing time to deliver an effective experience in this networked digital lending environment [12].

Leveraging Machine Learning for Early Disease Detection and Personalized Healthcare

This study is concerned with the problems of disease prediction using just symptoms in hospital care. With the increase of diseases associated with environmental factors and lifestyles, such prediction turns out to be very important. The study offers a new method based on the patient symptoms for disease prediction. The KNN and also SVM algorithms are used because of their accuracy. The conclusion underscores the power of machine learning to transform healthcare by facilitating early diagnosis, enhanced treatment results, and individualized recommendations. Ethical considerations and the technologies such as blockchain can be a vital part of the implementation process [13].

Machine Learning-Based Mobile Application for Early Detection of Mental Health Issues

In this study, agile methodology refers specifically to the SCRUM framework in order to create a mobile application for the early identification of mental health issues. With the help of artificial intelligence, the Python programming language, SQL Server, and also Android Studio, this application adds sentiment analysis as well as machine learning in analyzing the data from Facebook and Twitter. The outcomes have given a competent machine learning application for the prevention of mental health in Peru, that has helped various citizens. The conclusion highlights the success of machine learning in monitoring mental health through social media, suggesting data enhancement for improved prediction potential and dealing with sarcasm and spelling mistakes by users [14].

Design and Implementation of a Workspace Organization System Using Machine Learning

The article studies how DeepFace, TensorFlow, and also OpenCV are pattern recognition tools that structure a workspace. It studies machine learning and also recommender systems, systematically covering the topic. The work provides a goal tree and uses the algebra of algorithms as a mathematical means. In this manner, the software system implemented the functionality by using UML to design it in Android Studio. The study pinpoints the lack of appropriate instruments in creating the workspace systems. Further research is trying to confirm the results and also widen functionality [15].

3. DATASET DESCRIPTION:

The dataset comprises 1,003,436 entries with 'Text' and 'Sentiment' columns. 'Text' contains financial content, while 'Sentiment' classifies sentiments as Positive, Negative, or Neutral. The dataset source is from Kaggle, and sentiments are distributed among the three classes. Notably, 1,000,912 'Text' entries are non-null, indicating some missing values. Understanding this structure will guide preprocessing for machine learning tasks, addressing missing values and leveraging sentiment labels for analysis.

4. METHODOLOGY:

The implemented code encapsulates a comprehensive methodology for constructing a machine learning model geared towards predicting loan approval status. This process encompasses various stages, including data preprocessing, model training, hyperparameter optimization, model evaluation, and the creation of a user-friendly interface for making predictions.

4.1 MACHINE LEARNING:

4.1.1 Libraries Used:

Pandas:

Utilizing the versatile panda's library facilitated the seamless loading and manipulation of our financial sentiment dataset. The DataFrame structure provided by pandas allowed us to explore, clean, and preprocess the data efficiently.

NLTK (Natural Language Toolkit):

The nltk library played a pivotal role in text preprocessing tasks. It provided essential tools for tokenization, stopwords removal, and stemming, enhancing the quality of the textual data for subsequent analysis.

BeautifulSoup (bs4):

To effectively handle HTML tags present in the financial text data, we employed the BeautifulSoup library. This allowed for the extraction of pure text content from potentially noisy HTML structures.

Scikit-learn:

The scikit-learn library provided essential tools for building and evaluating our sentiment analysis model. Key components, such as the `train_test_split` function, `CountVectorizer` for text vectorization, `MultinomialNB` for sentiment classification, and `accuracy_score` for model evaluation, were seamlessly integrated into our workflow [1].

Matplotlib and WordCloud:

Visualization of sentiment distribution and word clouds was made possible by the Matplotlib and WordCloud libraries, respectively. These tools allowed us to create insightful visualizations, aiding in the exploration of sentiment trends and the identification of key terms in the financial text.

Joblib:

The joblib library was instrumental in persisting our trained sentiment analysis model. By saving the model as a file ('sentiment.pkl'), we ensured its accessibility for future use without the need for retraining.

Regular Expressions (re):

The re module in Python facilitated robust text pattern matching, particularly in the identification of negation words. This was crucial for enhancing the model's capability to handle negation in user input.

Contractions:

The contractions library was employed to expand contractions in the text data. This step was essential for ensuring that the model could interpret and analyze the full meaning of contracted words.

These libraries collectively formed a powerful toolkit that streamlined our sentiment analysis research, enabling efficient data handling, preprocessing, model training, evaluation, and interactive user input processing.

4.1.2 Aspect Based Sentiment Analysis:**Dataset Loading and Exploration:**

In the initial phase of our research, we loaded the Aspect-Based Financial Sentiment Dataset using the pandas library, providing us with a glimpse of its structure and contents. The dataset's key characteristics were examined through the `data.info()` and `data.head()` functions [1].

Data Cleaning and Preprocessing:

With a focus on preparing the textual data for sentiment analysis, we embarked on a series of data cleaning and preprocessing steps. This involved addressing missing values, visualizing the distribution of sentiments with a bar plot, and implementing a comprehensive text preprocessing function. Tasks included lowercasing, HTML tag removal, accent removal, contraction expansion, punctuation removal, tokenization, and stop word removal [16].

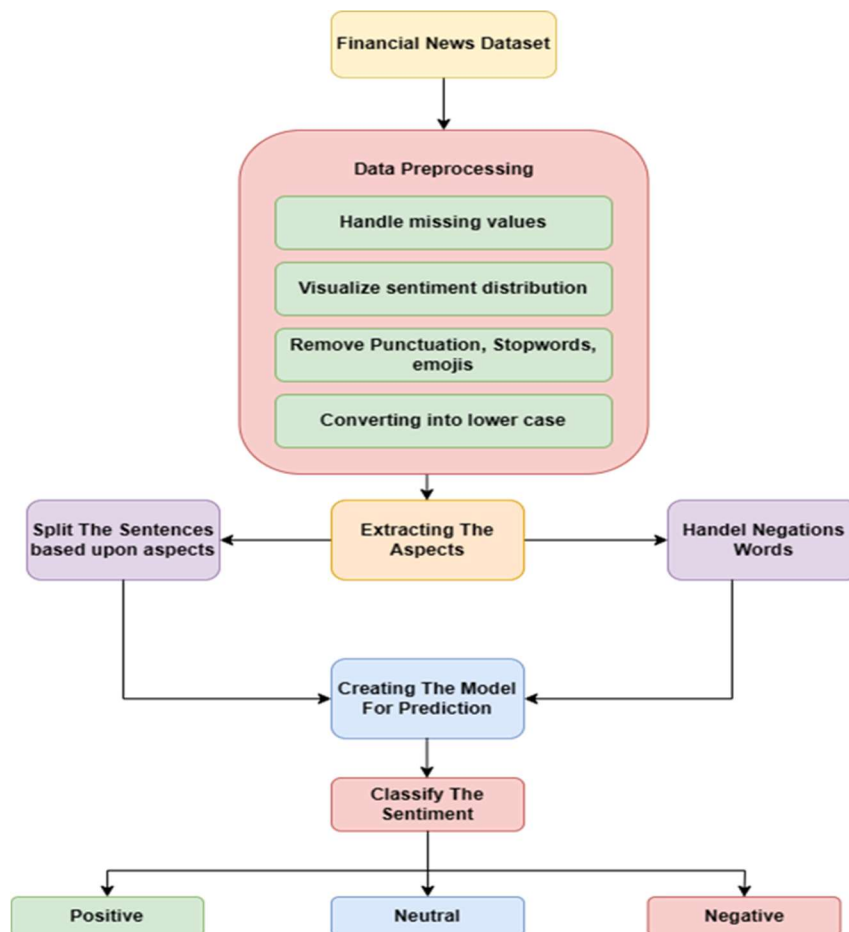


Figure 1: Flowchart of the Sentiment Analysis Model

Exploratory Data Analysis (EDA):

To gain deeper insights into the sentiment distribution within the dataset, exploratory data analysis (EDA) was conducted [1]. Sentiment counts were visualized using a bar plot, providing a clear understanding of the sentiment distribution and allowing for subsequent analysis of the sentiment trends present in the financial text data [2].

Model Building:

The core of our research involved training a sentiment analysis model using the preprocessed data. The dataset was split into training and testing sets, and a pipeline incorporating CountVectorizer for text vectorization and MultinomialNB for sentiment classification was constructed [2]. The model was then trained on the training set, paving the way for subsequent evaluation.

Model Evaluation:

Following model training, predictions were made on the test set, and the model's performance was rigorously evaluated. Accuracy, a key metric, was calculated using the accuracy_score from the

sklearn library. The evaluation phase provided valuable insights into the model's effectiveness in capturing sentiments present in the financial text data [17].

Model Persistence:

Recognizing the importance of model reusability, the trained sentiment analysis model persisted using the joblib library. This step allowed us to save the model as a file ('sentiment.pkl'), ensuring its accessibility for future applications and analyses without the need for retraining [2].

Handling Negation in User Input:

To enhance the practicality of our sentiment analysis model, a mechanism for handling negation in user input was introduced. Functions were implemented to detect negation words and split sentences based on conjunctions. The predictive model was then adapted to consider negation in its sentiment predictions, providing a more nuanced understanding of sentiment in user input [2].

User Input for Prediction:

Demonstrating the practical application of our sentiment analysis model, users were prompted to input sentences, and the model, equipped with negation handling, provided sentiment predictions for each segment of the input. This interactive example showcased the adaptability of the model to real-world user inputs and the nuanced nature of sentiment analysis in financial text [2].

4.1.3 Loan Approval Prediction:

Data Loading and Exploration:

The initial phase involves loading a dataset ('loan_prediction.csv') using the Pandas library. Subsequent exploration of the dataset is performed using various Pandas functions, including head(), tail(), shape, info(), and isnull(). These functions collectively offer a comprehensive overview of the dataset, encompassing its structure, missing values, and basic statistical characteristics.

Data Cleaning:

The data cleaning procedure focuses on addressing missing values and transforming categorical variables. Missing values are imputed using the mode for categorical variables ('Self_Employed' and 'Credit_History'). Additionally, the 'Dependents' column undergoes standardization by replacing '3+' with '4'. Categorical variables are converted to numerical representations through mapping using dictionaries.

Feature Scaling:

Numerical columns ('ApplicantIncome', 'CoapplicantIncome', 'LoanAmount', 'Loan_Amount_Term') undergo standardization using Scikit-Learn's StandardScaler. This ensures that all features contribute uniformly to the model [18].

Model Training and Evaluation:

The dataset is divided into training and testing sets via the `train_test_split` function. A dedicated function, `model_val`, is defined to facilitate the training of the model, generation of predictions, and evaluation of its performance through metrics such as accuracy and cross-validation scores. Models, including `RandomForestClassifier`, `LogisticRegression`, `SVM`, and `DecisionTreeClassifier`, are trained and assessed.

Hyperparameter Tuning:

The `RandomizedSearchCV` technique is employed for hyperparameter tuning of the `LogisticRegression`, `SVM`, and `RandomForestClassifier` models. This process aims to identify the optimal set of hyperparameters for enhancing model performance.

Model Saving and Loading:

The `RandomForestClassifier` model, equipped with the optimal hyperparameters, is saved using `Joblib`. This enables the reuse of the trained model without the need for retraining, which proves invaluable in production scenarios.

User Interface for Prediction:

A straightforward user interface is devised using Python's input functions. Users input relevant details such as gender, marital status, income, etc., and the model subsequently predicts whether the loan will be approved based on this input.

4.2 ANDROID:

The Android application aims at offering users a user-friendly environment so that they can predict from their given data set whether the loan will be approved or not. The application leverages Python scripts for machine learning models and interfaces with the `Chaquopy` library to natively connect Python in the Android environment [5].

Setting Up the Android Environment:

First, the process starts by using `Android Studio IDE` to create a new Android research. The structure of the research includes layout files for creating the user interface, a `Kotlin` file meant to implement logic, and an `assets` folder earmarked for storing Python scripts and machine learning models. With this basic structure in place, the integration between Android and Python aspects of the application is fluid [19].

Integrating Chaquopy:

Chaquopy is integrated into the Android research, so interoperability between Kotlin (Android) and Python becomes seamless. If you paste the following code into your build.gradle file and set up the research to support Python, then the development environment becomes quite good with both these languages working together [5]. This integration serves as a central node where an Android app can easily run Python scripts and take advantage of the power of such scripts, bringing the strength of Android's framework together with the flexibility that is offered by Python within one research environment.

4.2.1 Loan Analysis and Financial News:

Creating User Interface (UI):

This addresses the user interface, which is carefully designed based on XML in layout files to ensure user-friendly design. Strategic use of essential components, such as the EditText widgets are being used to collect user input for critical parameters such as gender, marital status, income etc [4]. This thoughtful design also makes the app visually appealing and sets the stage for an intuitive user experience. After initializing the EditText widgets, they are then intricately linked within the Kotlin file to facilitate smooth interaction and handling of data. This is an essential glue between the visual elements and logic driving the application's usability [4].

Initializing EditText Widgets in Kotlin:

In the predict_loan_page.kt file of Kotlin, each EditText widget is initialized. Each widget is linked with a variable through the findViewById method, setting up a direct relationship between the visual elements defined in XML and Kotlin code.

Fetching User Input:

Values are extracted from the initiated EditText widgets and user input becomes the center of attention. By courtesy of the text property, the entered data is captured as strings. These strings are then converted into the proper data types, so compatible with other operations that follow [5].

Calling Python Functions:

Integration of Chaquopy stands out as Python functions are invoked right from Kotlin without any head-scratches. A Python script (loan_status_predictor.py) is loaded and the load_model function, which includes receiving user inputs as parameters. This dynamic interplay plays a crucial bridge between the Android environment and Python functionality.

Displaying Results:

The result of the Python script, normally a loan approval prediction, is beautifully reported to the user. After making an inference, the result is passed on to a designated UI component like a TextView ending up finally being presented as something user-friendly [4].

Python Script Modifications:

The Python script is adjusted to work with the very dynamic user input (loan_status_predictor.py). The load_model function is improved for parameter acceptance, focusing specifically on the user input which determines the shape of the user_data DataFrame. This two-way communication helps smooth the flow of information between Android and Python.

4.2.2 Sentiment Analysis:

Application Components and UI Elements:

The UI of the application consists of several important components. An EditText widget enables users to enter descriptive financial text, including market insights and news articles. A TextView is assigned to show the results of the predicted sentiment analysis, and a Button initiates the sentiment analysis.

These UI components are arranged and designed within a ConstraintLayout using the XML layout file [3]. The design is simple and user-friendly, allowing a quick interaction with the functionality of sentiment analysis.

Python Integration for Sentiment Analysis:

This application also incorporates Python functionality, thanks to the Chaquopy library. This integration enables communication between the Android application and a Python script used for sentiment analysis [3].

The core sentiment analysis logic is encapsulated in the Python script 'sentiment_analysis.py'. It uses the joblib library to load a pre-trained sentiment analysis model. The model is essential for estimating the attitude related to various sections of the user-typed financial text.

Sentiment Analysis Process Flow:

When the user clicks on the "Predict" button, the Android application captures the text typed by the user. Once the text is passed to the Python script using the Chaquopy library, sentiment analysis begins. The Python script with its machine learning model breaks down the input text into segments and predicts the sentiment that follows each segment [3].

The sentiment analysis takes into account several aspects, such as the presence of negation words. This subtle approach improves the precision of sentiment predictions, especially when it concerns financial content where negations can completely change the meaning [20].

Result Presentation and User Interaction:

In the code of the Android application, there is a function named 'handleSentimentAnalysisResult' that handles the results obtained from sentiment analysis in Python. This function is responsible for formatting the results and updating the TextView to reflect a readable sentiment analysis.

The goal of the result presentation is to provide a thorough sentiment analysis for different parts of the input text. The segments are linked with their predicted sentiment, enabling users to understand the subtle variations in sentiments within financial content.

User Benefits and Engagement:

This financial mobile application is a powerful tool for users who are involved in the field of finance. It enables users to understand the sentiment of financial texts. This might involve comments about market trends, stock performance, or economic news.

With its predictive properties, sentiment analysis improves user engagement through a brief description of emotional context within financial material. This may be especially helpful for investors, analysts, or anyone who wants to understand the underlying sentiments in financial data.

5. RESULT:

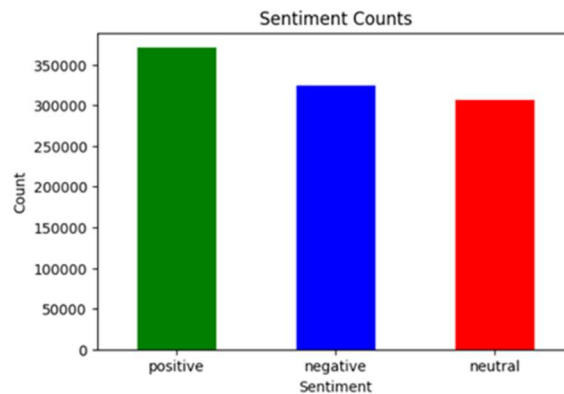


Figure 2: Count of Sentiment Present in each type

Accuracy: 0.88
 Enter a sentence: This month bank profit is very good
 This month bank profit is very good - Positive

Figure 3: Accuracy of the Aspect Based Sentiment Analysis Model



Figure 4: Homepage of the Android App

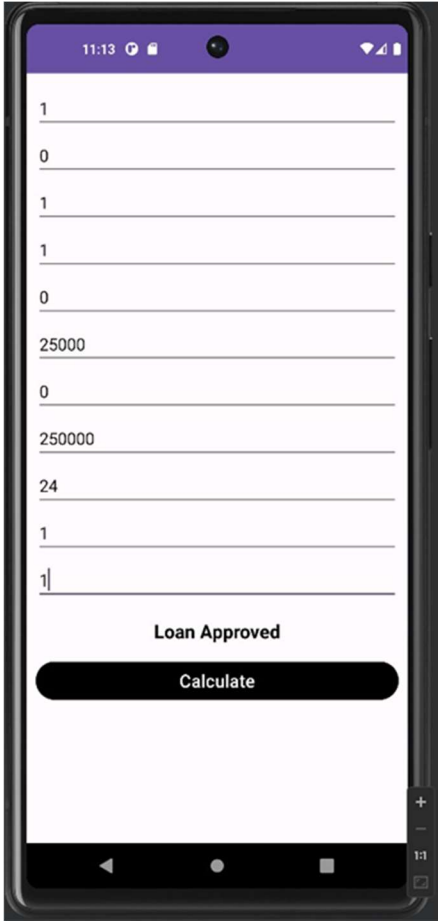


Figure 5: Loan Approval Prediction Android App

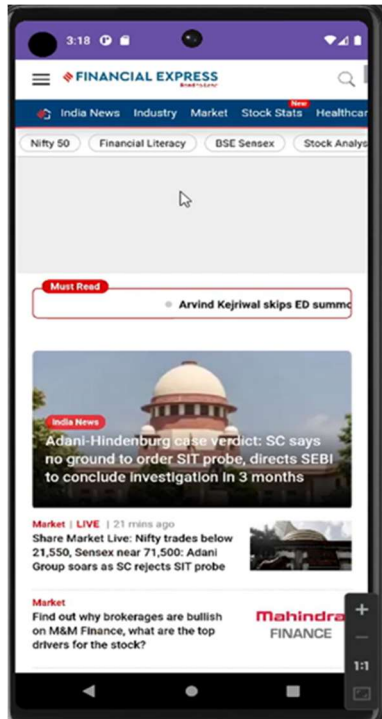


Figure 6: Financial News Displaying in the App

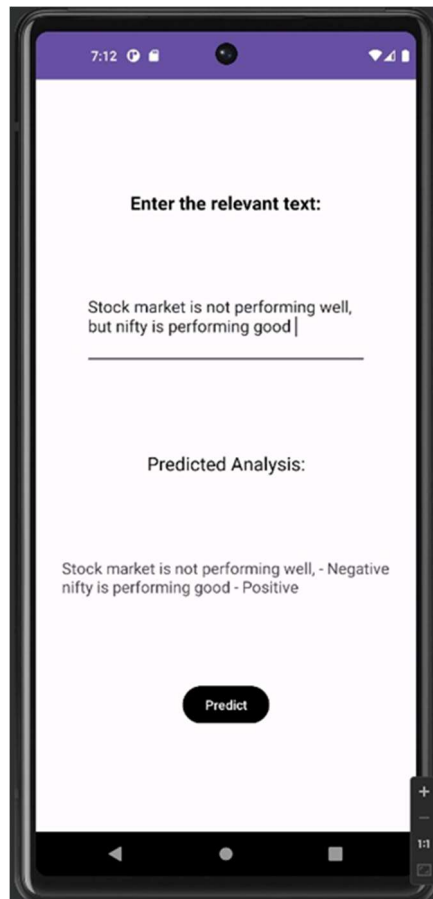


Figure 7: Predicting Aspect Based Sentiment Using the Android App

6. FUTURE WORK:

The future for the production of the application developed in Android Studio involves several significant improvements that enhance the user experience, increase functionality, and also incorporate a monetization strategy. The majority of the attention will be paid to polishing UI elements for the loan page and also sentiment analysis feature, as well as incorporating a premium version in the present application.

Enhancing UI Elements:

To make a more visually appealing and also more user-friendly interface, the UI components of the loan page and the sentiment analysis feature will be improved. This includes redesigning the layouts, enhancing the color palette, and maintaining coherence throughout the application. With an improved UI, the users will have a more captivating and fluent connection with these vital features.

Improving Sentiment Analysis:

The sentiment analysis feature will be greatly improved to ensure its accuracy and also efficiency. This may include incorporating machine learning algorithms to improve the analysis of the user input and offer more detailed sentiment results. Moreover, user feedback mechanisms will be used to regularly train and refine the sentiment analysis model to keep it relevant and functional.

Introducing Monetization:

The application will be monetized to maintain its development and also maintenance. This includes the implementation of a paid version of the application through which the users will have to pay to access the premium services or advanced features. The paid version could include a lot more financial analysis tools, personalized insights, or an ad-free environment. Consideration will be given to the pricing strategies to achieve affordability and value for the users.

Implementing Payment Gateway:

As part of the paid version integration, a payment gateway that is secure and also efficient will be incorporated. This guarantees to the user that making payments within the application is very smooth and secure. The payment gateway will handle different payment options to address a wide audience.

User Subscription Management:

For effective management of the paid version, a system for managing the user subscriptions will be integrated. This system will provide subscriptions for the users, billing cycles, and also an

interface that is very easy to use for the users to manage their subscription preferences. It will also have features like free trials and offers to help attract and retain users.

Summing up, future work on the application for Android development is directed at improving the user experience by creating a paid version that would have improved functionality. These improvements will help in ensuring the long-term sustainability of the application since it is competing with the other apps.

7. CONCLUSION:

In conclusion, the implemented code establishes a robust and all-encompassing workflow designed for the construction of a machine-learning model specialized in predicting loan approval status. Encompassing crucial stages such as data loading, exploration, cleaning, feature scaling, model training, evaluation, hyperparameter tuning, and the creation of a user-friendly interface, this code stands as a foundational asset for predictive modeling. The utilization of pivotal libraries, including Pandas, NumPy, Scikit-Learn, Joblib, and Matplotlib, significantly contributes to the efficiency and coherence of the entire process. The selection of the Random Forest Classifier model is based on its adaptability, straightforward implementation, and proficiency in managing both categorical and numerical features.

Looking ahead, the initiation of the development of a Financial Aspect-Based Sentiment Analysis App introduces a transformative dimension to the research's scope. This innovative endeavor seeks to provide a comprehensive solution for analyzing sentiments related to financial aspects. Currently in the developmental stage, the app holds the promise of delivering valuable insights into public sentiments surrounding financial topics, thereby contributing to a deeper understanding of market trends and investor sentiments. The incorporation of advanced natural language processing techniques aligns seamlessly with the evolving landscape of financial technology and sentiment analysis, opening up exciting possibilities for enhancing decision-making processes in the financial domain.

REFERENCES:

- [1] Jangid, H., Singhal, S., Shah, R. R., & Zimmermann, R. (2018, April). Aspect-based financial sentiment analysis using deep learning. In Companion Proceedings of the The Web Conference 2018 (pp. 1961-1966). <https://dl.acm.org/doi/pdf/10.1145/3184558.3191827>
- [2] Lengkeek, M., van der Knaap, F., & Frasincar, F. (2023). Leveraging hierarchical language models for aspect-based sentiment analysis on financial data. *Information Processing & Management*, 60(5), 103435. <https://www.sciencedirect.com/science/article/pii/S0306457323001723>

- [3] Consoli, S., Barbaglia, L., & Manzan, S. (2022). Fine-grained, aspect-based sentiment analysis on economic and financial lexicon. *Knowledge-Based Systems*, 247, 108781. <https://www.sciencedirect.com/science/article/pii/S0950705122003677>
- [4] Ndayisenga, T. (2021). Bank loan approval prediction using machine learning techniques (Doctoral dissertation). <http://154.68.126.42/handle/123456789/1437>
- [5] Khan, A., Bhadola, E., Kumar, A., & Singh, N. (2021). Loan approval prediction model a comparative analysis. *Advances and Applications in Mathematical Sciences*, 20(3). https://www.mililink.com/upload/article/1759044670aams_vol_203_january_2020_a10_p427-435_afrah_khan_and_nidhi_singh.pdf
- [6] Khine, W.L., & Aung, N.T. (2019). Applying Deep Learning Approach to Targeted Aspect-based Sentiment Analysis for Restaurant Domain. 2019 International Conference on Advanced Information Technologies (ICAIT), 206-211. <http://uit.edu.mm/wp-content/uploads/2020/05/NTTA-18.pdf>
- [7] Syamala, M. (2019). A Deep Analysis on Aspect based Sentiment Text Classification Approaches. *International Journal of Advanced Trends in Computer Science and Engineering*. <http://www.warse.org/IJATCSE/static/pdf/file/ijatcse01852019.pdf>
- [8] Gana, K.A., Prasad, S., Chidvilas, P.V., & Kumar, V.V. (2019). Customer Loan Approval Classification by Supervised Learning Model. <https://www.ijrte.org/wp-content/uploads/papers/v8i4/D9275118419.pdf>
- [9] Wang, J., Hu, G., & Xiang, L. (2019). An Empirical Study on the Prediction of Farmers' Ability to Acquire Loans in the Yanliang District Based on Cloud Model. *Proceedings of the Sixth Symposium of Risk Analysis and Risk Management in Western China (WRARM 2019)*. <https://download.atlantis-press.com/article/125917977.pdf>
- [10] Singh, P., Pisipati, N., Krishna, P.R., & Prasad, M.V. (2019). Social Signal Processing for Evaluating Conversations Using Emotion Analysis and Sentiment Detection. 2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP), 1-5. Social Signal Processing for Evaluating Conversations Using Emotion Analysis and Sentiment Detection | Semantic Scholar
- [11] Chang, J. S. (2022). Mobile application development of an alternative credit scoring system at a fintech company to improve personal financial planning (Doctoral dissertation, UTAR). http://eprints.utar.edu.my/4732/1/fyp_IB_2022_CJS.pdf
- [12] Shuvo, M. S. K., Tuhin, I. H., & Parvez, R. (2023). Design and Development of Fast Loan Management System (FLMS) (Doctoral dissertation, Sonargaon University (SU)). <http://202.74.246.118/bitstream/handle/123456789/700/CSE-230138.pdf?sequence=1&isAllowed=y>

- [13] Sharma, A., & Jangid, S. Disease prediction system using machine learning. <https://amity.edu/UserFiles/ajjem/269PREETIKA%20RESEARCH%20PAPER%20final%20-%20preetika%20bhardwaj.pdf>
- [14] Ponce, E. K., Cruz, M. F., & Andrade-Arenas, L. (2022). Machine Learning Applied to Prevention and Mental Health Care in Peru. *International Journal of Advanced Computer Science and Applications*, 13(1). https://www.researchgate.net/profile/Laberiano-Andrade-Arenas/publication/358332985_Machine_Learning_Applied_to_Prevention_and_Mental_Health_Care_in_Peru/links/622959523c53d31ba4b5c636/Machine-Learning-Applied-to-Prevention-and-Mental-Health-Care-in-Peru.pdf
- [15] Basyuk, T., Vasyliuka, A., & Demkiv, L. (2022). Peculiarities of organizing the workspace using machine learning methods. *Proceedings* <http://ceur-ws.org> ISSN, 1613, 0073. <https://ceur-ws.org/Vol-3312/paper19.pdf>
- [16] Jangid, H., Singhal, S., Shah, R. R., & Zimmermann, R. (2018, April). Aspect-based financial sentiment analysis using deep learning. In *Companion Proceedings of the The Web Conference 2018* (pp. 1961-1966). <https://dl.acm.org/doi/pdf/10.1145/3184558.3191827>
- [17] Lengkeek, M., van der Knaap, F., & Frasincar, F. (2023). Leveraging hierarchical language models for aspect-based sentiment analysis on financial data. *Information Processing & Management*, 60(5), 103435. <https://www.sciencedirect.com/science/article/pii/S0306457323001723>
- [18] Consoli, S., Barbaglia, L., & Manzan, S. (2022). Fine-grained, aspect-based sentiment analysis on economic and financial lexicon. *Knowledge-Based Systems*, 247, 108781. <https://www.sciencedirect.com/science/article/pii/S0950705122003677>
- [19] Levi, Y., & Benartzi, S. (2020). Mind the app: Mobile access to financial information and consumer behavior. Available at SSRN 3557689. <https://www.cafri-sif.com/2022/files/384%20Mind%20the%20App%20Mobile%20Access%20to%20Financial%20Information%20and%20Consumer%20Behavior.pdf>
- [20] Elmanda, F. A., Merdikawati, G. G., & Wahyuni, R. (2022). The application of financial recording applications towards financial report for micro, small and medium enterprises. *International Journal of Research and Applied Technology (INJURATECH)*, 2(1), 196-203. <http://ojs.unikom.ac.id/index.php/injuratech/article/download/6918/2995>