**Mini Project Report on**



**SMART FEEDBACK/ACTIVE DOCUMENT ML**

**Submitted in partial fulfilment of the requirement for the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE & ENGINEERING**

**Submitted by:**

**LAKSHAY BANSAL University Roll No. 2018903**

***Under the Mentorship of***

**DR. JAY BHATNAGAR**

**Professor**



**Department of Computer Science and Engineering**

**Graphic Era (Deemed to be University)**

**Dehradun, Uttarakhand**

**July-2023**



**CANDIDATE’S DECLARATION**

I hereby certify that the work which is being presented in the project report entitled **“Smart feedback/Active Document ML”** in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineeringof the Graphic Era (Deemed to be University), Dehradun shall be carried out by the under the mentorship of **Dr Jay Bhatnagar, Professor**, Department of Computer Science and Engineering, Graphic Era (Deemed to be University), Dehradun.

Lakshay Bansal University Roll no. 2018903

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Description** | **Page No.** |
| Chapter 1 | Introduction | 1-2 |
| Chapter 2 | Literature Survey | 3-4 |
| Chapter 3 | Methodology | 5-6 |
| Chapter 4 | Result and Discussion | 7-11 |
| Chapter 5 | Conclusion and Future Work | 12-13 |
|  | References |  |

**Chapter 1**

**Introduction**

* 1. **Introduction**

Machine learning (ML) is a potent tool for enhancing user experience and increasing productivity in a variety of fields. Two effective ML applications are smart feedback and active documents. These technologies enable intelligent and engaging engagements with digital content by utilizing ML algorithms and approaches, offering insightful information and immediate support. This introduction will go over the ideas of intelligent feedback and active documents, emphasizing the advantages and uses of each.

**1.1.1 Smart Feedback**

To deliver insightful and individualized feedback, smart feedback refers to the automated analysis and interpretation of user input or data utilizing ML techniques. It entails recording user interactions, examining trends, and making pertinent recommendations or suggestions. Large datasets can be used to train ML algorithms, which can then alter their responses based on user behaviours to provide more intelligent and contextually aware feedback.

Smart feedback can be used to enhance user experiences across a variety of sectors. For instance, ML-powered feedback systems in e-learning platforms may analyze learners' progress, pinpoint areas for development, and offer customized recommendations for better learning results. Smart feedback can assess client attitude, feedback, and behaviour in customer care applications to produce customized responses that increase customer happiness.

**1.1.2 Active Documents**

Active documents are dynamic, interactive digital documents that use machine learning to offer intelligent functionality and real-time help. Active documents, as opposed to conventional static documents, may evaluate, respond to, and adapt to user input, making them more engaging and informative. Active documents with incorporated ML algorithms may automatically generate content, analyze data, and assist decisions.

Active documents are used in data analysis, research documentation, and collaborative settings in real life. For instance, ML algorithms can assist with auto-suggesting content, summarizing talks, or extracting important information from massive text datasets in collaborative document editing applications. Active papers in research documentation can produce tables and charts, update references automatically, and provide current information pertinent to the document's content.

* 1. **Smart Feedback and Active Documents Using ML:**

Smart feedback and active documents are made possible thanks in large part to machine learning. Large volumes of data can be used to train ML systems to identify patterns, categorize inputs, and produce precise predictions or suggestions. To deliver pertinent feedback, Natural Language Processing (NLP) models can examine textual data, sentiment, and context. For individualized suggestions, user behaviour and preferences can be understood using ML techniques like clustering, classification, and regression.

Additionally, active documents can use machine learning (ML) algorithms to process and analyze data, providing real-time insights. Active documents can be mined for information, have data transformed, and have anomalies detected using methods like text mining, picture recognition, and anomaly detection.

In conclusion, ML-powered smart feedback and dynamic documents improve user interactions with digital information significantly. These technologies enhance user experiences and boost productivity by offering tailored feedback, pertinent recommendations, and on-demand assistance. Smart feedback and active documents expand the realms of education, customer service, cooperation, research, and data analysis by utilizing the potential of ML.

* 1. **Project Outline**

I created this project by checking the sentiments and polarity of the Amazon reviews and creating a new document under which the polarity value and the sentiments are mentioned.

**Chapter 2**

**Literature Survey**

[Madhavi S. Darokar](https://ieeexplore.ieee.org/author/37088924510), [Atul D. Raut](https://ieeexplore.ieee.org/author/37088950491), [Vilas M. Thakre](https://ieeexplore.ieee.org/author/38466771600) in there paper Emotion recognition is a challenging task, but it is becoming increasingly important as social media becomes more ubiquitous. This review has discussed the different methodologies that have been used for emotion recognition on social media, including lexicon-based, machine learning-based, and deep learning-based approaches. Each approach has its own strengths and weaknesses, and the best approach for a particular application will depend on the specific requirements of that application.[1]

[Samina Kausar](https://ieeexplore.ieee.org/author/37085533836), [Xu Huahu](https://ieeexplore.ieee.org/author/37596945000), [Waqas Ahmad](https://ieeexplore.ieee.org/author/37089400728), [Muhammad Yasir Shabir](https://ieeexplore.ieee.org/author/37089547257), [Waqas Ahma](https://ieeexplore.ieee.org/author/37089400728) in their paper, they have proposed a sentiment polarity categorization technique for online product reviews. The technique uses a combination of lexicon-based and machine learning-based approaches to classify reviews into five classes: strongly negative, negative, neutral, positive, and strongly positive. The technique was evaluated on a dataset of 500,000 online reviews and achieved an accuracy of 82.2%.

The results of the evaluation show that the proposed technique is effective in classifying online product reviews. The technique is able to handle a variety of linguistic features, including negation, intensifiers, and compound words. The technique is also able to adapt to different domains and genres of text.[2]

By Sindhu Rashmi. H. R, Prof. Anisha. B. S, Dr. Ramakanth Kumar. P in their paper, they have discussed the importance of document classification and the basic steps involved in the process. They have also presented a small use case for document classification, in which they distinguish between rhyme and non-rhyme documents.

The results of our experiment show that we can achieve an accuracy of 95% by using a machine learning model to classify documents. This suggests that document classification can be a valuable tool for businesses and organizations that need to manage large amounts of text data.

In the future, they plan to explore other use cases for document classification, such as spam filtering, email routing, and language identification. They also plan to develop more robust machine learning models that can handle a wider range of documents.[3]

Ujjwal Biswas and Samit Bhattacharya of Indian Institute of Technology, Guwahati in their paper, they have proposed an ML-based intelligent real-time feedback system for blended classrooms. The system provides real-time feedback to students and teachers, and it uses students' academic performance prediction models with real-time states and dynamic feedback timings based on historic feedback statistics. The system also uses feedback scheduling algorithms, choices of peripheral devices for real-time feedback, and feedback modalities to optimize fatigue.

They conducted an empirical study to evaluate the system, and the results showed that the system was well-received by the end users. The unique design elements of the system, such as dynamic timing, choice of peripheral devices, and modalities of real-time feedback, were found to be crucial in integrating the system with blended classes. The intelligent characteristics of the system were also appreciated by a large proportion of the end-users.

They were trying to impose a higher comparative system usability scale (SUS) score with benchmarks showing real promise of the system design. The system has the potential to improve learning outcomes in blended classrooms by providing real-time feedback to students and teachers.[4]

**Chapter 3**

**Methodology**

Here, my project is based on a supervised method of Machine Learning and using the natural language toolkit (NLTK).

**3.1 Importing Libraries**

First, we import the Python pre-defined libraries which are required for my project such as numpy, pandas, ntlk, seaborn, matplotlib**.**

**3.2 Data Loading**

Then, we load our dataset in a program using the inbuilt Python library pandas then we will do all the work on that dataset for running different models to check the sentiments and polarity.

**3.3 Basic NLTK model testing**

* Plotting a graph for checking the distribution score(star) in the dataset to run model.
* Now we tokenize the review text.

**3.4 VADER Model**

VADER stands for the Valence Aware Dictionary for Sentiments Reasoning in this model we import the sentiment intensity Analyzer function to analysis the sentiment polarity of the statement and the text, first we import that and apply on the example to check it is working or not it gives the 4 parameters i.e. ‘neg’ means negative polarity value, ‘neu’ means the neutral polarity value, ‘pos’ means positive polarity value, and ‘compound’ which gives the complete polarity value of statement which lies between range of -1 to 1. If closer to negative than it shows the negative sentiment, if close to the positive value, then it shows the positive sentiment whereas if it closes to the zero that means it is a neutral sentiment.

After running on the sample test case, we run this model for the complete dataset and store the result into a new dataset and compare the result with the stars.

**3.5 Roberta Model**

This model is based on the hugging face mode, in which it uses the “transformer” library of python, which first trains an model then gives the results.

In my project first I took a python model which is twitter tweets sentiment analysis based on Roberta model to train my model then I ran the tokenization and model classification by using automated tools of transformer library on my pretrained model.

Test a Roberta model on the sample after that create model function for the whole dataset and create a list for the result and merge that result with the previous dataset.

**3.6 Comparison between VADER model and Roberta Model**

In this step I take a merge dataset of containing both models result and plot a pair-plot graph between the six (6) parameters: -

* Vader\_neg
* Vader\_neu
* Vader\_pos
* Roberta\_neg
* Roberta\_neu
* Roberta\_pos

**3.7 Pipelining Model**

In this step from the ‘transformer’ import the pipeline and in the pipeline function we mentioned the model as the sentiment analysis and the transformer library pipeline function merged with the sentiment analyzing predefined code of python and run model on my dataset.

**3.8 Exporting feedback Table**

After testing all models, I combine the results of all the models and convert it to the CSV file giving the sentiments and the value of polarity.

**Chapter 4**

**Result and Discussion**

**4.1 Importing Libraries**

**A black rectangular object with white text

Description automatically generated**

**Fig 4.1.1 showing imported libraries**

**4.2 Data Loading**

**A screenshot of a computer

Description automatically generated**

**Fig 4.2.1 showing the dataset**

**4.3 Basic NLTK model testing**

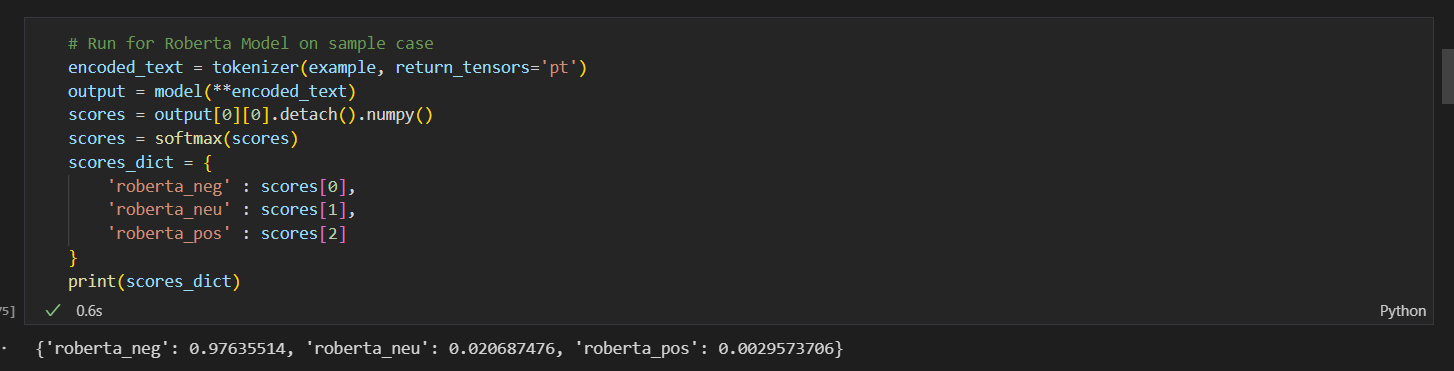
**A graph with red squares

Description automatically generated**

**Fig 4.3.1 graph for the counting reviews by star**

**4.4 VADER Model**

**4.4.1 Sample Test Case**

****

**Fig. 4.4.1 List of VADER model**

**4.4.2 Table for complete dataset**

**A screenshot of a computer

Description automatically generated**

**Table 4.4.2 Table with the VADER results**

**4.4.3 Graph for distribution of Score over polarity**

**A graph of a bar chart

Description automatically generated**

**Fig. 4.4.3 Graph between compound score and star**

**A graph of different colored bars

Description automatically generated**

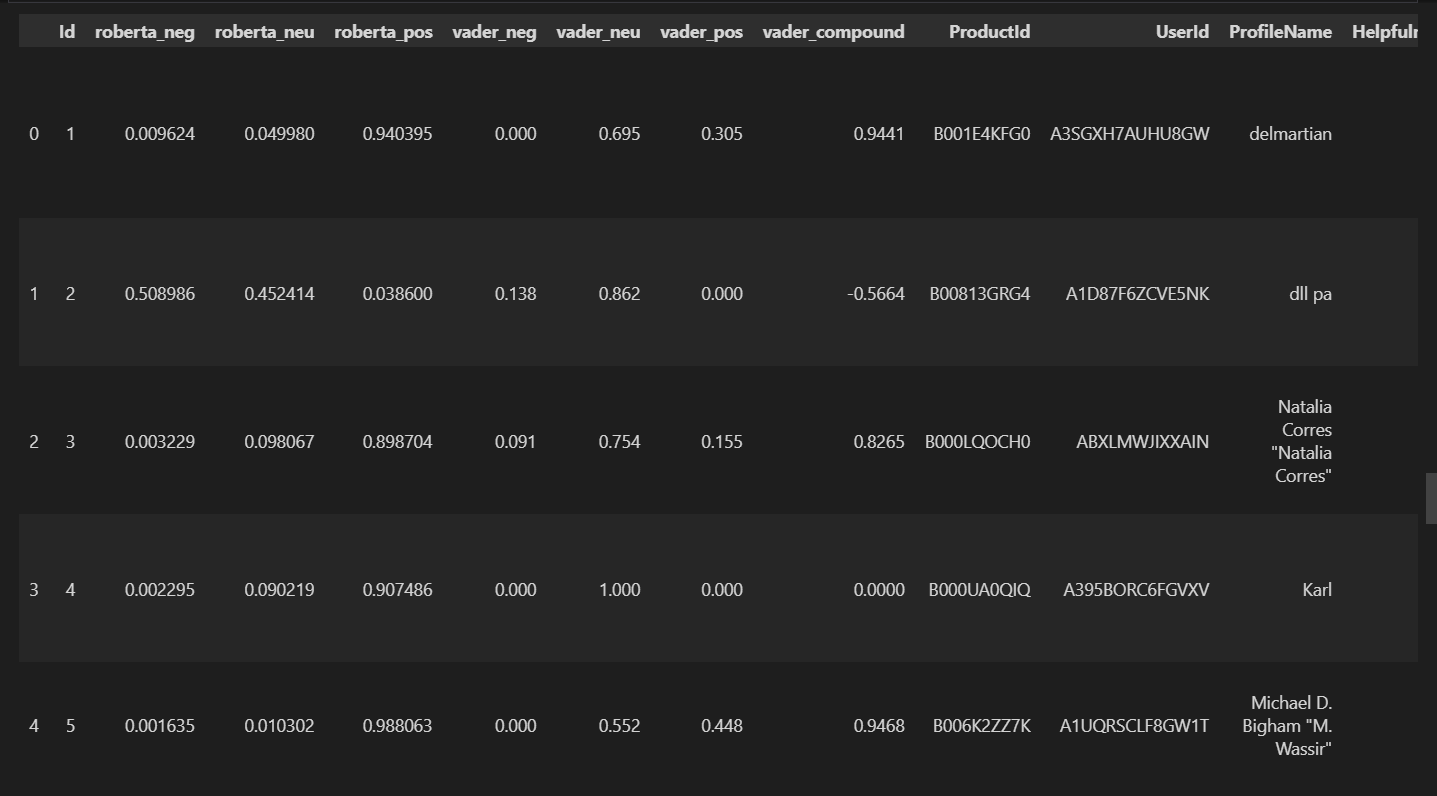
**Fig.4.4.4 Graph showing the distribution of stars over different polarity.**

**4.5 Roberta Model**

**4.5.1 Sample Test Case** **A black screen with white text

Description automatically generated**

**Fig 4.5.1 List showing the roberta result on sample case**

**4.5.2 Table for complete dataset** ****

**Table 4.5.2 table for the dataset with the roberta result**

**4.6 Comparison between VADER model and Roberta Model**

**A screenshot of a graph

Description automatically generated**

**Fig 4.6.1 Pair-plot for comparison of Vader and Roberta Model**

**4.7 Pipelining Model**

**A screenshot of a computer

Description automatically generated**

**Table 4.7.1 table showing the pipeline result of complete dataset**

**4.8 Exporting feedback Table**

**A screenshot of a computer

Description automatically generated**

**Table 4.8.1 Showing the Feedback CSV file**

**Chapter 5**

**Conclusion and Future Work**

**5.1 Conclusion**

Within the larger fields of machine learning and natural language processing, smart feedback and active document ML are both promising research and application areas. Beyond my latest update in September 2021, new methods and techniques may have developed in these fields as they continue to develop. I suggest reading the most recent research papers, publications, and resources in the NLP and ML domains to receive the most recent and accurate information.

**5.2 Future Work**

These directions for future research illustrate some of the possible avenues for improving machine learning for active documentation and smart feedback. In order to comprehend the most recent advancements and difficulties in the fields of NLP and machine learning, it is crucial to stay up to speed with the most recent literature and research.

**5.2.1 Smart Feedback**

1. Personalization and Context Adaptation: Enhancing personalized feedback systems to better comprehend user preferences, learning preferences, and context. This can entail using user profiles, previous interactions, and contextual data to deliver feedback that is more individualized and pertinent.

2. Multimodal Feedback: Combining different feedback modalities, such as text, speech, visuals, and gestures, in order to provide richer and more thorough input. The quality and accuracy of feedback in applications like language learning, virtual assistants, and online tutoring can be enhanced through multimodal techniques.

3. Explainable Feedback: Creating methods to explain the feedback that smart systems provide. This would be essential for educational applications since it would enable users to comprehend the rationale behind particular recommendations, which would improve learning and user confidence.

4. Feedback Diversity and Creativity: Investigating ways to give various feedback in order to prevent repetition and promote innovation. This can be especially important in applications for content creation or creative writing.

**5.2.2 Active Document**

1. Real-Time Document Analysis and Collaboration: Moving active documentation forward Real-time analysis and teamwork on dynamic documents will be supported by ML. This can entail keeping track of modifications, spotting document revisions, and using ML models to offer current information.

2. Dynamic Content Summarization: Creating methods for summarizing dynamic documents that change over time to assist users in understanding the most pertinent and crucial information quickly.

3. Incremental Learning for Document Analysis: Exploring incremental learning ways to change ML models continuously as new data is introduced to the texts. This would guarantee that even when the content changed, the models remained precise and pertinent.

4. Temporal Information Extraction: The study of techniques for extracting temporal information from dynamic documents, such as timelines of events, shifts in emotion through time, or historical patterns.

5. Interactive Document Exploration: The development of user interfaces that let users engage with document analysis models, allowing them to explore various features and gain particular insights.

**References**

[1] Madhavi S. Darokar, Atul D. Raut and Vilas M. Thakre, “Methodological Review of Emotion Recognition for Social Media: A Sentiment Analysis Approach”, 2021 International Conference on Computing, Communication and Green Engineering (CCGE), Pune, 23-25 September 2021, <https://ieeexplore.ieee.org/document/9776385> **(Example: Conference Paper).**

[2] Samina Kausar, Xu Huahu, Waqas Ahmad, Muhammad Yasir Shabir, Waqas Ahmad, “A Sentiment Polarity Categorization Technique for Online Product Reviews”, IEEE Access, vol. 8, no. 8, pp. 3594 – 3605, 2019.**(Example: Journal Paper)**

[3] Sindhu Rashmi. H. R, Prof. Anisha. B. S, and Dr. Ramakanth Kumar. P, “Smart Document Analysis Using AI-ML”, International Journal of Innovative Research in Computer Science & Technology (IJIRCST) ISSN: 2347-5552, Volume-7, Issue-3, May-2019 DOI: 10.21276/ijircst.2019.7.3.6. **(Example: Research Paper)**

[4] Ujjwal Biswas and Samit Bhattacharya, “ML-based intelligent real-time feedback system for blended classroom”, International Journal of Information Technology and Computer Science (IJITCS), vol. 14, Issue 2,2022, <https://doi.org/10.1007/s10639-023-11949-5> **(Example: Journal Paper)**

[5] Source code reference with the help of YouTube

<https://youtu.be/QpzMWQvxXWk> **(Example: YouTube)**

<https://youtu.be/9p1KYtYAus8> **(Example: YouTube)**

[6] Dataset reference from the Kaggle <https://www.kaggle.com/code/robikscube/sentiment-analysis-python-youtube-tutorial/input?select=Reviews.csv> **(Example: Website)**