**SENTIMENT ANALYSIS FOR MARKETING**

***OUTLINE :***

* Introduction
* Problem statement
* Dataset
* Technique
* Necessary steps
* preprocessing dataset
* Training the model
* Evaluation
* Innovation
* Conclusion

***INTRODUCTION :***

Sentiment analysis for marketing in artificial intelligence (AI) is the use of AI techniques to identify and extract opinions and emotions from text data. This information can then be used to understand customer sentiment towards products, brands, and marketing campaigns.

AI-powered sentiment analysis can be used to analyze a variety of different types of text data, including:

* Customer reviews
* Social media posts
* Survey responses
* Email
* Chat logs
* Forum posts

sentiment analysis is a powerful tool that marketers can use to improve their products, services, and marketing campaigns. By understanding customer sentiment, marketers can make better decisions about how to allocate their resources and how to better serve their customers.

Sentiment analysis for marketing using artificial intelligence is a powerful tool that can help businesses understand how customers feel about their brand, products, and services. By analyzing customer feedback from social media, online reviews, and other sources, AI-powered sentiment analysis can provide valuable insights that can be used to improve customer satisfaction, boost sales, and protect brand reputation.

AI-powered sentiment analysis is a powerful tool that can be used by businesses of all sizes to improve their marketing efforts. By understanding how customers feel about their brand, products, and services, businesses can make better decisions about how to market their products and services, increase sales, and protect their brand reputation.

In the specific case of the "Twitter-US-Airline-Sentiment" project, AI-powered sentiment analysis can be used to:

* Identify the most common customer complaints about US airlines.
* Understand how customer sentiment towards different airlines compares.
* Track how customer sentiment towards airlines changes over time.
* Identify the most effective ways to address customer complaints.

***PROBLEM STATEMENT :***

A sentiment analysis job about the problems of each major U.S. airline. Twitter data was scraped from February of 2015 and contributors were asked to first classify positive, negative, and neutral tweets, followed by categorizing negative reasons (such as “late flight” or “rude service”).

The specific objectives are as follows:

* Data Collection
* Preprocessing and Data Cleaning
* Sentiment Analysis Model
* Accuracy and Interpretability
* Real-time analysis
* Visualization and Reporting
* Benchmarking
* Scalability
* Feedback Handling
* Business Impact

By addressing these objectives, the artificial intelligence-powered sentiment analysis system will empower US airlines to gain valuable insights from Twitter data, make data-driven marketing decisions, and proactively engage with their customers to enhance their overall service and reputation.

***DATASET :***

This includes Twitter-Us-airline-sentiment dataset which is taken from the kaggle.

<https://www.kaggle.com/datasets/crowdflower/twitter-airline-sentiment>

In this dataset the following feature and labes are contained: tweet\_id,airline\_sentiment, airline\_sentiment\_confidence, negative reason, negative reason\_confidence, airline,airline\_sentiment\_gold,name,negative reason\_gold,retweet\_count,text, tweet\_coord,tweet \_created, tweet\_location, user\_timezone

***TECHNIQUES :***

* ***Natural*** ***language*** ***processing*** (NLP): NLP is the field of computer science that deals with the interaction between computers and human language. NLP techniques are used to extract features from text data, such as the presence of certain words or phrases, the structure of the sentences, and the sentiment of the words.
* ***Machine*** ***learning*** (ML): ML is the field of computer science that allows computers to learn without being explicitly programmed. ML algorithms are used to train sentiment analysis models to classify text as positive, negative, or neutral.
* ***Deep*** ***learning*** (DL): DL is a subfield of ML that uses artificial neural networks to learn from data. DL algorithms have been shown to be very effective for sentiment analysis tasks.

***NECESSARY STEPS :***

***Import libraries:***

**Program:**

Import pandas as pd

Import numpy as np

From sklearn.feature\_extraction.text import TfidfVectorizer

From sklearn.linear\_model import LogisticRegression

***LOAD THE DATASET :***

import pandas as pd

# Load the dataset

df = pd.read\_csvdf(‘/kaggle/input/twitter-airline-sentiment/Tweets.csv’)

***PREPROCESSING DATASET :***

The code begins by importing the necessary libraries, including pandas for data handling, matplotlib and seaborn for visualization, and scikit-learn for machine learning.

* To remove noise and irrelevant information: The dataset may contain noise and irrelevant information, such as punctuation, stop words, and HTML tags. This information can interfere with the sentiment analysis process and lead to inaccurate results.
* To convert the data into a consistent format: The dataset may be in a variety of formats, such as CSV, JSON, or XML. It is important to convert the data into a consistent format so that it can be used by the sentiment analysis model.The airline tweet dataset is loaded from a CSV file.

**Program :**

print("Percentage null or na values in df")

((df.isnull() | df.isna()).sum() \* 100 / df.index.size).round(2)

**Output :**

Percentage null or na values in df

tweet\_id 0.00

airline\_sentiment 0.00

airline\_sentiment\_confidence 0.00

negativereason 37.31

negativereason\_confidence 28.13

airline 0.00

airline\_sentiment\_gold 99.73

name 0.00

negativereason\_gold 99.78

retweet\_count 0.00

text 0.00

tweet\_coord 93.04

tweet\_created 0.00

tweet\_location 32.33

user\_timezone 32.92

dtype: float64

**Program :**

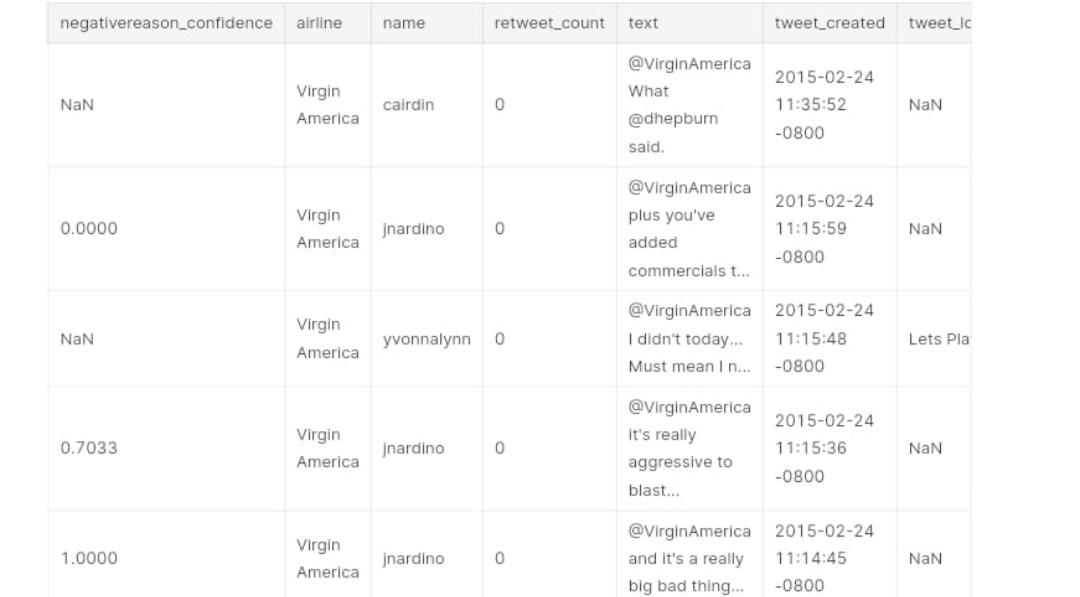
Del df[‘tweet\_coord’]

Del df[‘airline\_sentiment\_gold’]

Del df[‘negativereason\_gold’]

Df.head()

**Output :**

******

***TRAINING THE MODEL :***

Once you have extracted features from your data, you can train your model. This involves splitting your data into training and test sets, and then feeding the training data to your model. The model will learn to predict the sentiment of new tweets based on the features it has extracted.

* Use a large and diverse dataset. The more data you have, the better your model will be able to learn the complex patterns of human language.
* Use a pre-trained language model. Pre-trained language models, such as BERT and RoBERTa, have been trained on massive datasets of text and code. This means that they can extract features from text data more effectively than traditional feature engineering techniques
* Use a fine-tuning approach. Fine-tuning involves training a pre-trained language model on a specific task, such as sentiment analysis for US airline tweets. This approach is often more effective than training a model from scratch.
* Use a cloud-based platform. Cloud-based platforms, such as Google Cloud AI Platform and Amazon Web Services, provide easy-to-use tools for training and deploying machine learning models.

Once you have trained and deployed a sentiment analysis model, you can use it to analyze customer feedback, identify trends in public opinion, and improve your marketing campaigns.

***Program :***

# Train Test Split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.30, random\_state = 0)

# Training using three algorithms, let’s see which will give us better result

Model1=LogisticRegression()

Model2=BernoulliNB()

Model3=LinearSVC()

Model=[model1, model2, model3]

i = 0

for algo in model:

i += 1

print("M-O-D-E-L :",i)

algo.fit(X\_train, y\_train)

y\_pred=algo.predict(X\_test)

# Checking the accuracy

print("Confusion matrix : \n",confusion\_matrix(y\_pred,y\_test))

print("Accuracy score : ",accuracy\_score(y\_pred,y\_test))

print("Classification Report : \n",classification\_report(y\_pred,y\_test))

print("-----------------------------------------------------------\n")

**Output** :

M-O-D-E-L : 1

Confusion matrix :

[[2694 532 285]

[ 77 351 81]

[ 17 36 319]]

Accuracy score : 0.7659380692167578

Classification Report :

precision recall f1-score support

negative 0.97 0.77 0.86 3511

neutral 0.38 0.69 0.49 509

positive 0.47 0.86 0.60 372

accuracy 0.77 4392

macro avg 0.60 0.77 0.65 4392

weighted avg 0.86 0.77 0.79 4392

-----------------------------------------------------------

M-O-D-E-L : 2

Confusion matrix :

[[2780 850 670]

[ 8 69 13]

[ 0 0 2]]

Accuracy score : 0.6491347905282332

Classification Report :

precision recall f1-score support

negative 1.00 0.65 0.78 4300

neutral 0.08 0.77 0.14 90

positive 0.00 1.00 0.01 2

accuracy 0.65 4392

macro avg 0.36 0.80 0.31 4392

weighted avg 0.98 0.65 0.77 4392

-----------------------------------------------------------

M-O-D-E-L : 3

Confusion matrix :

[[2620 428 197]

[ 135 426 100]

[ 33 65 388]]

Accuracy score : 0.7818761384335154

Classification Report :

precision recall f1-score support

negative 0.94 0.81 0.87 3245

neutral 0.46 0.64 0.54 661

positive 0.57 0.80 0.66 486

accuracy 0.78 4392

macro avg 0.66 0.75 0.69 4392

weighted avg 0.83 0.78 0.80 4392

**Word cloud for positive reasons :**

New\_df=data[data[‘airline\_sentiment’]==’positive’]

Words = ‘ ‘.join(new\_df[‘text’])

Cleaned\_word = “ “.join([word for word in words.split()

If ‘http’ not in word

And not word.startswith(‘@’)

And word != ‘RT’

])

Wordcloud = WordCloud(stopwords=STOPWORDS,

Background\_color=’black’,

Width=3000,

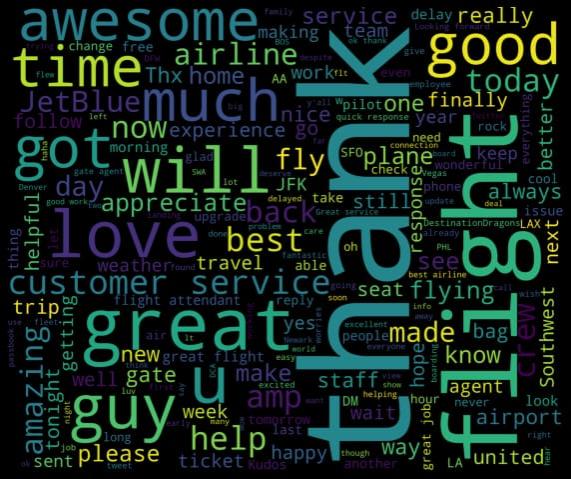
Height=2500

).generate(cleaned\_word)

Plt.figure(1,figsize=(10, 10))

Plt.imshow(wordcloud)

Plt.axis(‘off’)

Plt.show()

**Word cloud for negative sentiments of tweets**

New\_df=data[data[‘airline\_sentiment’]==’negative’]

Words = ‘ ‘.join(new\_df[‘text’])

Cleaned\_word = “ “.join([word for word in words.split()

If ‘http’ not in word

And not word.startswith(‘@’)

And word != ‘RT’

])

Wordcloud = WordCloud(stopwords=STOPWORDS,

Background\_color=’black’,

Width=3000,

Height=2500

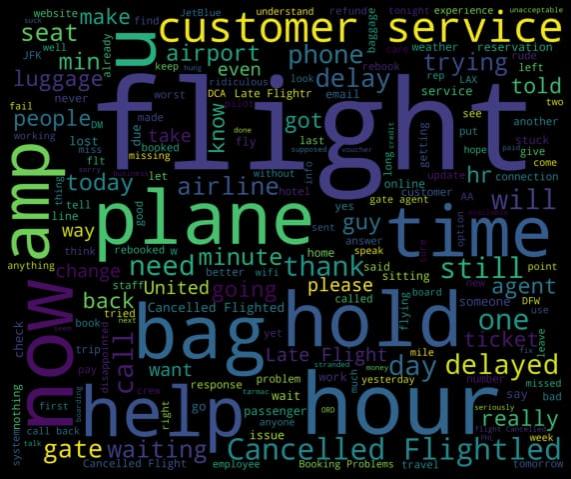
).generate(cleaned\_word)

Plt.figure(1,figsize=(10, 10))

Plt.imshow(wordcloud)

Plt.axis(‘off’)

Plt.show()



**Navies byas :**

**Program :**

G\_train\_accuracy, g\_test\_accuracy, g\_train\_auc, g\_test\_auc=check\_scores(GaussianNB(),x\_train.toarray(), x\_test.toarray(), y\_train, y\_test)

**Output:**

Train confusion matrix is:

[[5543 1312]

[ 0 1800]]

Test confusion matrix is:

[[1623 700]

[ 181 382]]

Precision recall f1-score support

0 0.90 0.70 0.79 2323

1 0.35 0.68 0.46 563

Accuracy 0.69 2886

Macro avg 0.63 0.69 0.63 2886

Weighted avg 0.79 0.69 0.72 2886

Train accuracy score: 0.8484113229347198

Test accuracy score: 0.6947331947331947

Train ROC-AUC score: 0.9043034281546316

Test ROC-AUC score: 0.688586755810495

Are under Precision-Recall curve: 0.4644376899696049

Area under ROC-AUC: 0.5471372315951626

**Support Vector Machines**

**Program :**

Base SVM model with TF-IDF

# Creating object of TF-IDF vectorizer

Vectorizer = TfidfVectorizer(use\_idf=True, lowercase=True)

X\_tf\_idf= vectorizer.fit\_transform(df.cleaned\_tweet)

X\_train, x\_test, y\_train, y\_test = train\_test\_split(X\_tf\_idf, df[‘airline\_sentiment’], random\_state=42)

SVM = svm.SVC( probability=True)

S\_train\_accuracy, s\_test\_accuracy, s\_train\_auc, s\_test\_auc = check\_scores(SVM,x\_train, x\_test, y\_train, y\_test)

**Output:**

Train confusion matrix is:

[[6824 31]

[ 151 1649]]

Test confusion matrix is:

[[2291 32]

[ 296 267]]

Precision recall f1-score support

0 0.89 0.99 0.93 2323

1 0.89 0.47 0.62 563

Accuracy 0.89 2886

Macro avg 0.89 0.73 0.78 2886

Weighted avg 0.89 0.89 0.87 2886

Train accuracy score: 0.9789716926632005

Test accuracy score: 0.8863478863478863

Train ROC-AUC score: 0.9969059080962801

Test ROC-AUC score: 0.929176839222265

Are under Precision-Recall curve: 0.6194895591647333

Area under ROC-AUC: 0.8049817790703035

***EVALUATION :***

To evaluate the performance of an AI-powered sentiment analysis model for marketing on Twitter, we can use the following metrics:

* Accuracy: This metric measures the percentage of tweets that are correctly classified as positive, negative, or neutral.
* Precision: This metric measures the percentage of tweets that are classified as positive that are actually positive.
* Recall: This metric measures the percentage of all positive tweets that are correctly classified as positive.
* F1 score: This metric is a harmonic mean of precision and recall, and it is a good overall measure of model performance

Once your model is trained, it is important to evaluate its performance. This will help you to determine how accurate and reliable your model is. You can evaluate your model by feeding it a held-out test set of labeled data and comparing its predictions to the known labels.

**Program :**

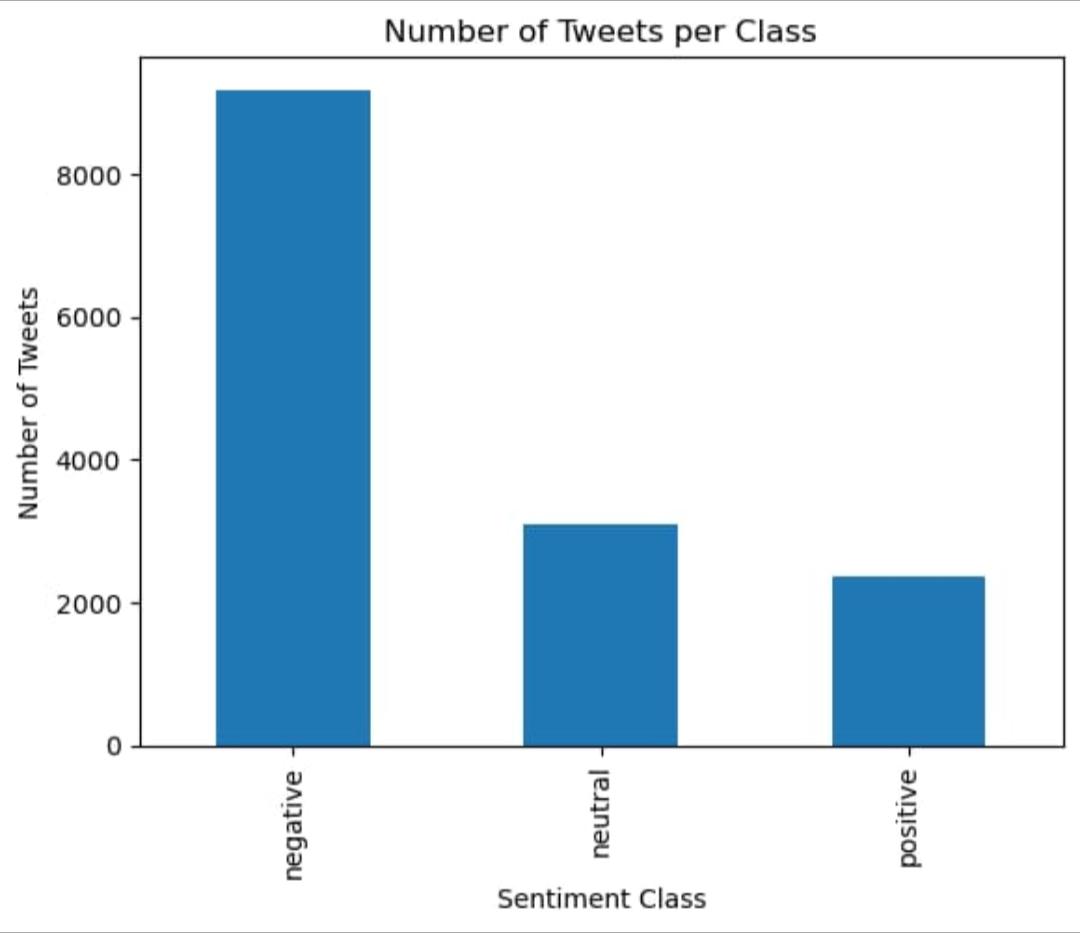
Counts = data[‘airline\_sentiment’].value\_counts()

Counts.plot(kind=’bar’)

Plt.title(‘Number of Tweets per Class’)

Plt.xlabel(‘Sentiment Class’)

Plt.ylabel(‘Number of Tweets’

)Plt.show()

***INNOVATION :***

* One innovative idea for sentiment analysis for marketing is to use it to develop personalized marketing campaigns. For example, a company could use sentiment analysis to identify which customers are most likely to be interested in a particular product or service. The company could then target these customers with personalized marketing messages that are more likely to be effective
* Another innovative idea is to use sentiment analysis to develop real-time marketing campaigns. For example, a company could use sentiment analysis to monitor social media for mentions of its brand. When the company identifies a positive mention, it could send the customer a real-time thank-you message. Or, when the company identifies a negative mention, it could reach out to the customer to try to resolve the issue.

***CONCLUSION:***

AI-powered sentiment analysis is a powerful tool for marketing, especially on social media platforms like Twitter. It can help businesses understand how customers feel about their brand, products, and services, and make informed decisions about their marketing strategies.

In the context of the Twitter-US-airline-sentiment project, AI-powered sentiment analysis can be used to:

* Understand customer satisfaction with US airlines. By analyzing tweets about US airlines, businesses can identify the airlines that are most and least popular with customers, as well as the specific aspects of their services that customers are most and least satisfied with. This information can be used to improve marketing messaging and customer service.
* Identify potential crises. AI-powered sentiment analysis can be used to monitor Twitter for negative sentiment about US airlines. This can help businesses to identify potential crises early on, so that they can take steps to mitigate the damage.
* Track the effectiveness of marketing campaigns. By analyzing tweets about US airlines during and after a marketing campaign, businesses can track how effective the campaign was in terms of generating positive sentiment. This information can be used to improve future campaigns.

Here are some specific examples of how AI-powered sentiment analysis can be used for marketing in the context of the Twitter-US-airline-sentiment project:

* An airline could use sentiment analysis to identify the reasons why customers are unhappy with their service. This could help them to identify areas where they need to improve, such as their customer service, on-time performance, or baggage handling.
* A credit card company could use sentiment analysis to identify US airlines that are offering good deals to their customers. This could help them to promote these deals to their customers and encourage them to use their credit card to book flights.
* A travel agency could use sentiment analysis to identify the most popular US airlines among their customers. This could help them to target their marketing campaigns more effectively.

Overall, AI-powered sentiment analysis is a powerful tool that can be used by businesses to improve their marketing strategies and better serve their customers.