



Negative



Neutral



Positive

SENTIMENT ANALYSIS USING NLP AND ML



Neutral



Positive



Negative



Neutral

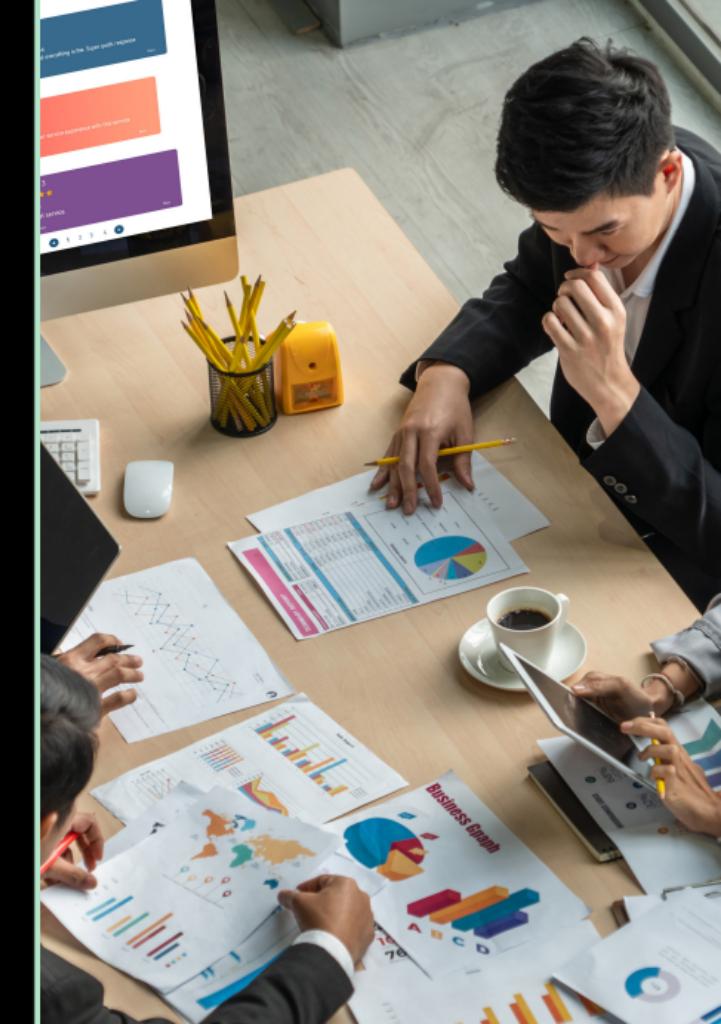
AiExploraCore presents

Shreya Sari : HU21CSEN0300148 Shravya Sari : HU21CSEN0300169 V. Siddharth: HU21CSEN0300164

Gnaeshwar Mudhiyam: HU21CSEN0300326 C.Rakesh : HU21CSEN0300162 Rohith Varma: HU21CSEN0300160

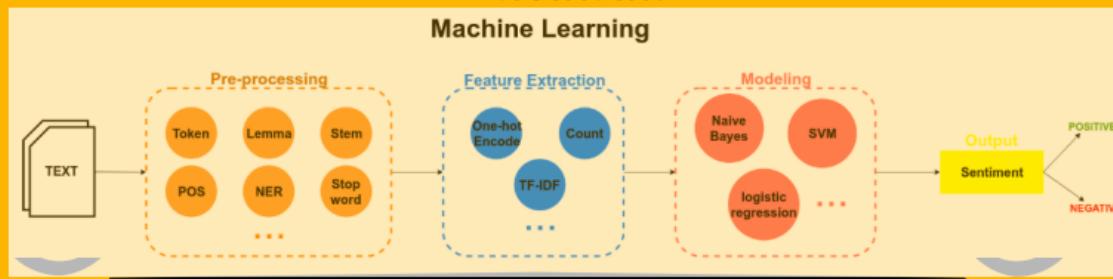
what is sentiment analysis?

Sentiment analysis is a technique through which you can analyze a piece of text to determine the sentiment behind it. It combines machine learning and natural language processing (NLP) to achieve this.



Abstract

The task of natural language processing to determine whether a piece of text contains some subjective information and what subjective information it expresses, i.e., whether the attitude behind this text is positive, negative or neutral.





How can this help?

1. *Social Media Sentiment Analysis*
2. *Brand Experience Insights*
3. *Improve Customer*
4. *News Trend Analysis*
5. *Real-Time Sentiment Insights*
6. *Real-Time Sentiment Insights*

Problem statement



To analyze the reviews from the IMDB reviews using the natural language processing and machine learning algorithms.



Techniques used are:

- *Lexicon based techniques*
- *Machine learning based techniques*
- *Hybrid techniques*

Lexicon based technique

Lexicon-based sentiment analysis is a technique that uses a dictionary of words labeled as positive, negative, or neutral to determine the sentiment of a text. There are two main approaches to lexicon-based sentiment analysis: dictionary-based and corpus-based1.

Dictionary-based methods create a database of positive and negative words from an initial set of words by including synonyms and antonyms. Corpus-based methods, on the other hand, obtain the dictionary from the initial set by usage of statistical techniques1.

Machine Learning techniques

- Machine learning-based techniques involve training a machine learning model on a dataset of text where each piece of text is labeled with its sentiment.
- The model learns to associate features of the text with the sentiment.
- When given new, unlabeled text, it can then predict the sentiment based on these learned associations.
- Naive Bayes or Support Vector Machines, or a more complex neural network model like RNN or CNN can be used.



Hybrid technique

Hybrid techniques combine both lexicon-based and machine learning-based approaches. They might use a lexicon-based method to generate features that are fed into a machine-learning model, or use a machine-learning model to predict sentiment, which is then refined using a set of rules.



There are three major types of algorithms used in sentiment analysis

Automated Systems

Automatic approaches to sentiment analysis rely on machine learning models like clustering.

Long pieces of text are fed into the classifier, and it returns the results as negative, neutral, or positive.

Hybrid Systems

Hybrid techniques are the most modern, efficient, and widely-used approach for sentiment analysis. Well-designed hybrid systems can provide the benefits of both automatic and rule-based systems.

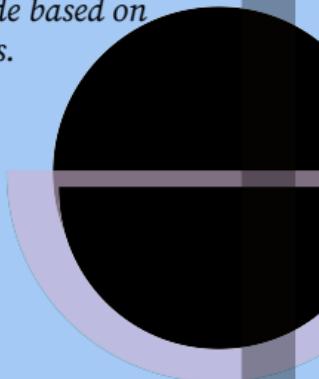
An example of a hybrid model would be a self-updating wordlist based on **Word2Vec**. You can track these wordlists and update them based on your business needs.

Rule-based Systems

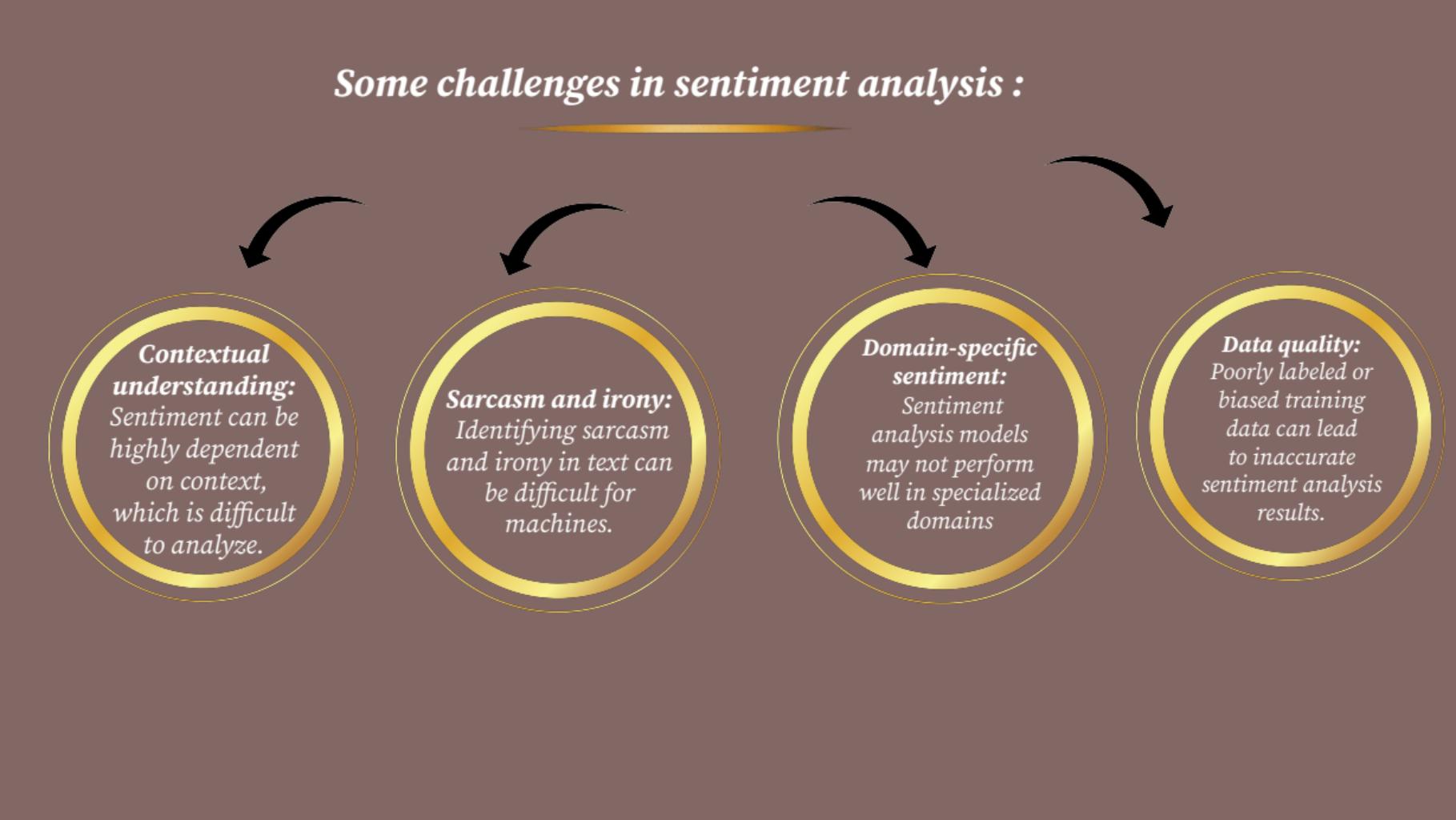
Unlike automated models, rule-based approaches are dependent on custom rules to classify data.

Popular techniques include tokenization, parsing, stemming, and a few others. You can consider the example we looked at earlier to be a rule-based approach.

These algorithms can be tailor-made based on context by developing smarter rules.



Some challenges in sentiment analysis :

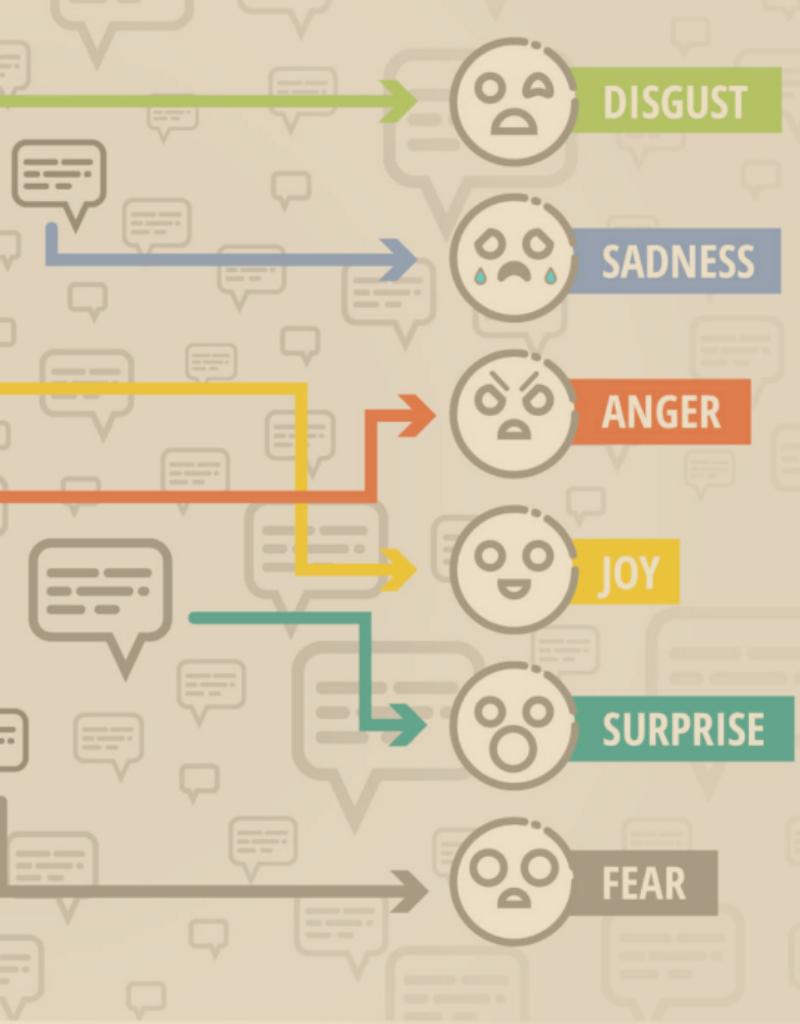


Contextual understanding:
Sentiment can be highly dependent on context, which is difficult to analyze.

Sarcasm and irony:
Identifying sarcasm and irony in text can be difficult for machines.

Domain-specific sentiment:
Sentiment analysis models may not perform well in specialized domains

Data quality:
Poorly labeled or biased training data can lead to inaccurate sentiment analysis results.



Libraries used

- *numpy as np*
- *import pandas as pd*
- *seaborn as sns*
- *matplotlib.pyplot as plt*
- *nltk*
- *CountVectorizer*
- *stopwords*
- *word_tokenize,sent_token ize*

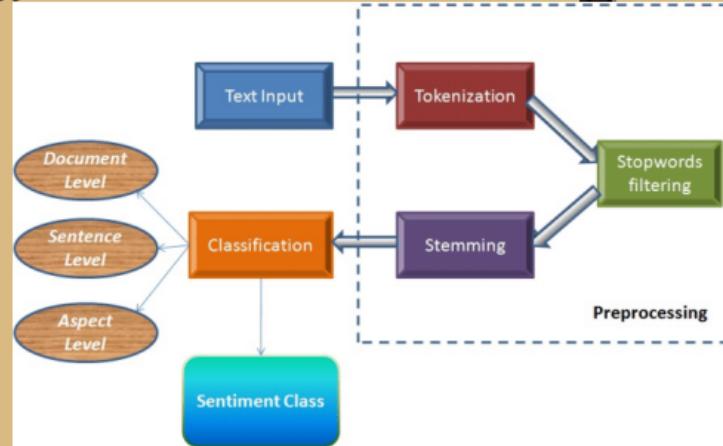
Steps for sentiment analysis

1. collect Dataset
2. Setting Up Libraries
3. Data Preprocessing
 1. Text stemming
 2. Removing stopwords and normalization
 3. Removing special characters
4. Bag of words Model
5. Labeling the sentiment text and splitting sentiment data
6. Modelling the dataset
7. Model Training
8. Model performance on test data
9. Accuracy of the model & Visualizing the classification report



Implementation of sentiment analysis

- Choose your content.
- First, you have to decide what kind of content you want to analyze.
- Gather your dataset.
- Split your dataset.
- Train a machine learning model.
- Validate your model.
- Deploy your model.
- Monitor your model's performance.

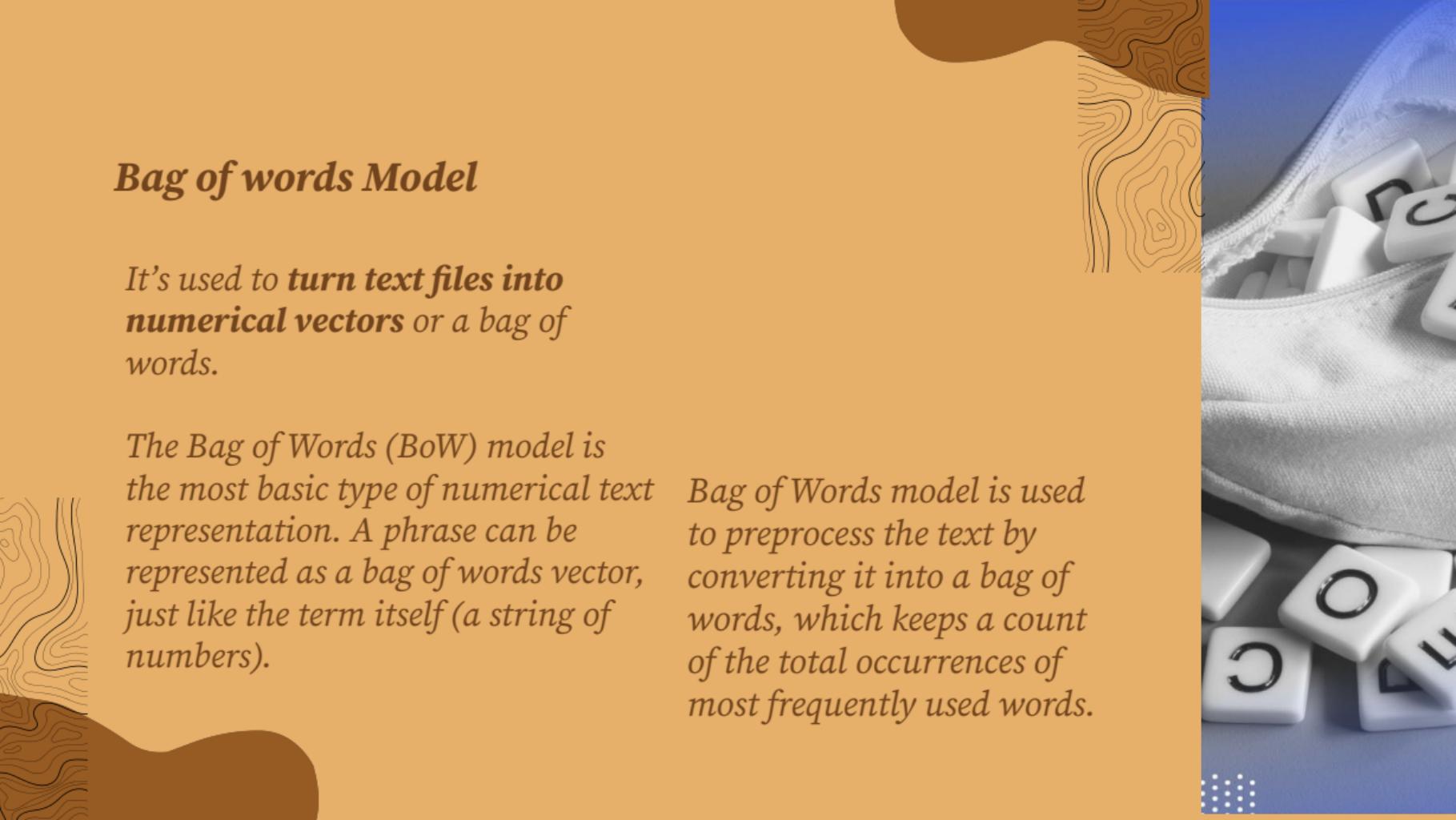


Bag of words Model

It's used to turn text files into numerical vectors or a bag of words.

The Bag of Words (BoW) model is the most basic type of numerical text representation. A phrase can be represented as a bag of words vector, just like the term itself (a string of numbers).

Bag of Words model is used to preprocess the text by converting it into a bag of words, which keeps a count of the total occurrences of most frequently used words.



Some common tools and libraries for sentiment analysis in Python



1. *NLTK (Natural Language Toolkit)*
2. *TextBlob*
3. *VADER (Valence Aware Dictionary and sEntiment Reasoner)*
4. *spaCy*
5. *Transformers (Hugging Face's library)*
6. *Stanford NLP*
7. *IBM Watson Natural Language Understanding*



To sum up.....

Sentiment analysis is a powerful tool that you can use to solve problems from brand influence to market monitoring. New tools are built around sentiment analysis to help businesses become more efficient.

Sentiment analysis is used for many buisness to analyse their customer reviews and gain the required insights which can improve their performance.

Code representation

[https://colab.research.google.com/drive/1LjI4JjnLLd2
UNwQlVrgh_9HMvdE1D-qj?usp=sharing](https://colab.research.google.com/drive/1LjI4JjnLLd2UNwQlVrgh_9HMvdE1D-qj?usp=sharing)



Thank
you!