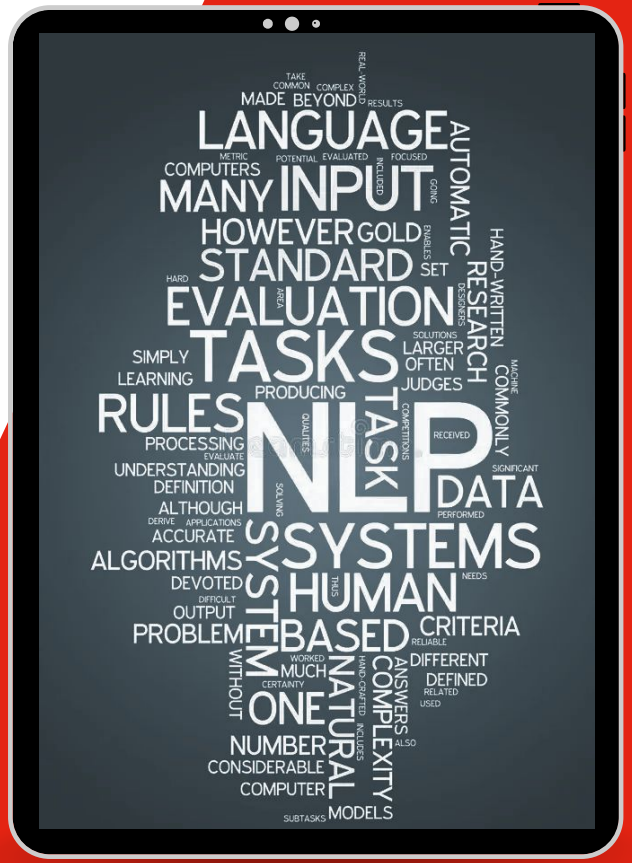
Text Classification

Muhajir Akbar Hasibuan

Class Meeting 5

2023



Muhajir Akbar Hasibuan

Data Scientist

muhajirakbarhsb@gmail.com
+62 85297023234
Jakarta, Indonesia
linkedin.com/in/muhajirakbarhsb
github.com/muhajirakbarhsb
medium.com/@muhajirakbarhsb

Experience

Telkom Indonesia *Sept 2020 - present*

- Develop internal and external use cases
- Provide data understanding in making a model
- Provide Preparation and data engineering according to the use case implemented
- Provide data validation so that the analysis results are as expected
- Building modeling for the development or improvement of internal and external programs
- Provide descriptive and diagnostic insight into data processing
- Recommend and define new growth hacking strategy for digital marketing team

Project

- Pioneered a Robust Big Data Solution for MyIndihome, Revolutionizing Customer Experiences
- Orchestrated a High-Performing ML Team, Elevating myIndihome TV's Personalized Content Impact by 25%
- Envisioned and Executed a Cutting-Edge Big Data Solution for Elevated Customer Engagement on myIndihomeTV
- Fueled Business Insights via Dynamic Data Profiling, Performance Dashboards, and Insights for Langit Musik, RBT, and Upoint
- Engineered Innovative Big Data Solutions that Propelled Growth for PadiUmkm (E-Commerce)
- Powered Success for GameQoo through Strategic Big Data Solutions
- Architected and Established MLOps Framework, Elevating Telkom Indonesia's Digital Business Products
- Crafted Visionary Video Analytics Solutions for Telkom Indonesia's Revolutionary Digital Business IoT Product

MEMBER OF DATA SCIENTIST TASK FORCE | NOVEMBER 2021 - PRESENT
MEMBER OF AI TASK FORCE | NOVEMBER 2022- PRESENT

A Pool of data scientists and AI Engineer Expert in Telkom Indonesia. It was established to leverage data-driven culture for decision-making within the organization. (Applied Research, Standardization, Consultation)

Achievement

BEST Talent of the Year at Telkom Indonesia, Digital Business and Technology Division - Digital Technology and Platform -2022

Codex by Telkom Indonesia *Apr 2020- Sept 2020*

- Building Data Pipeline for Langit Musik Recommender System
- Business Analytic for Langit Musik
- Build Recommender System for Langit Musik

Universitas Syiah Kuala *2015-2020*

- Bachelor of Science, Statistics

Skills

Hard Skills

- Data Analytics
- Statistics
- Machine Learning
- Deep Learning
- MlOps
- Business Intelligence
- Data Engineering
- Cloud Computing
- Time Series & Demand Forecasting
- Natural Language Processing
- Fraud & Anomaly Detection
- Computer Vision

Tools Skills

- Python
- Pyspark
- Sql
- Apache Airflow
- GCP
- Docker
- MLflow
- Prometheus
- Evidently
- Grafana
- Redash, Metabase, Superset, Looker Studio
- Pytorch

Class Meeting 1: Introduction to NLP (1 session)
Introduction to NLP
Basics of text data and its characteristics
The importance of NLP in today's world
Overview of the course structure and objectives

Class Meeting 2: Text Preprocessing (1 session)
Understanding the text preprocessing pipeline
Tokenization, stemming, and lemmatization
Stop words removal
Hands-on exercises with Python for text preprocessing

Class Meeting 3: Text Representation (1 session)
Introduction to text representation techniques
Bag of Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF)
n-gram
Word embeddings (Word2Vec, GloVe)**just an introduction
Practical exercises on text representation

Class Meeting 4: Introduction to ML, DL Supervised and Unsupervised (1 session)
Introduction to deep learning (ML)
Introduction to deep learning (DL)
DL vs. traditional machine learning (ML)
Machine Learning and their applications
Neural networks and their applications
Basic ML DL concepts and terminology

Class Meeting 5: Sentiment Analysis (1 session)
What is sentiment analysis?
Data collection and labeling for sentiment analysis
Building a sentiment analysis model
Practical sentiment analysis examples

Class Meeting 6: Text Classification (1 session)
Introduction to text classification
Binary and multi-class classification
Building a text classification model
Real-world text classification examples

Class Meeting 6 Introduction to Topic Modeling(1 session)
Introduction to topic Modelingg
Topic Modeling Implementation
LDA(Latent dirichlet allocation)
Hands-on exercises with Topic Modeling

mid term

Class Meeting 8: **Sesi Khusus deep Learning**

Class Meeting 9: Introduction to Word Embeddings (1 session)
Fundamentals of word embeddings
Word2Vec, GloVe, and FastText
Word embedding applications
Hands-on exercises with word embedding

Class Meeting 10: Attention Mechanisms (1 session)
Introduction to attention mechanisms
Self-attention and multi-head attention
Transformers architecture
Practical examples of attention in NLP

Class Meeting 11: Transformer Models (1 session)
In-depth study of the Transformer model
Pre-trained transformer models (BERT, GPT-2)
Fine-tuning transformers for NLP tasks
Transformer-based applications

Class Meeting 12: Advanced NLP Topics (1 session)
Advanced NLP topics such as BERT, XLNet, and RoBERTa
Transfer learning in NLP
Ethics in NLP
Recent developments and trends in NLP

Class Meeting 13: Advanced NLP Topics and Deployment Process in Industry (1 session)
Introduction to LLM
Introduction how industry utilize NLP to generate revenue
Introduction MLOps for NLP(bonuses from practitioners)

Class Meeting 14-15: Project Work (2 sessions)
Dedicated sessions for students to work on NLP projects with guidance and assistance.

Class Meeting 16: Project Presentations and Conclusion (1 session)
Students present their NLP projects
Recap of key takeaways from the course
Discuss further resources for NLP enthusiasts
Course conclusion and feedback

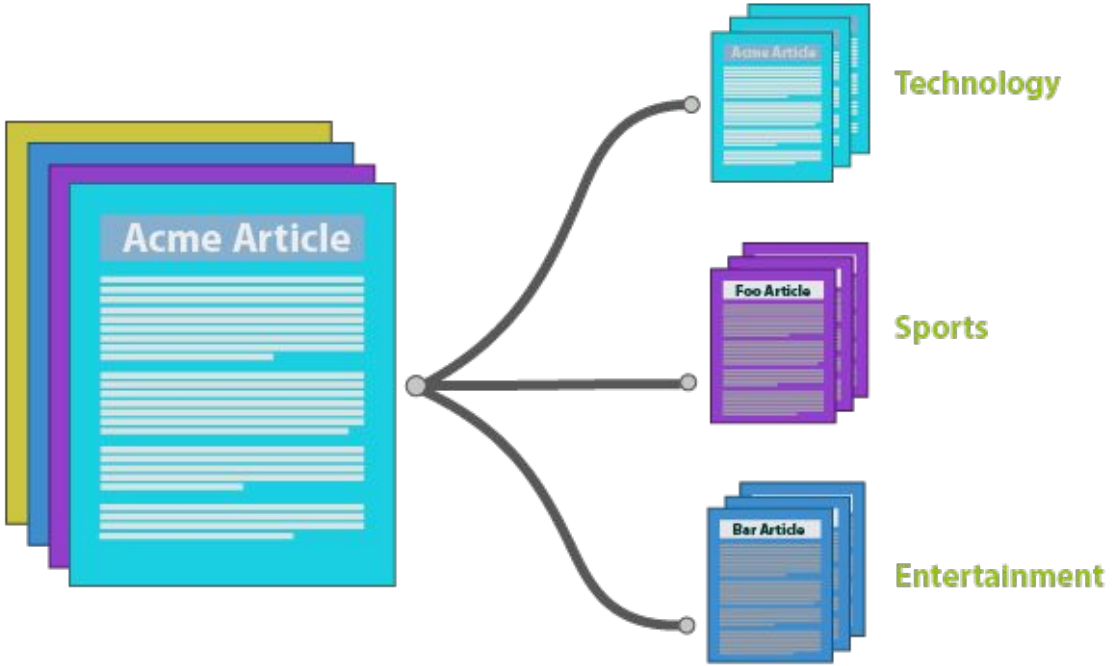
Agenda

- Introduction to text classification?
- Data collection and labeling for Text Classification
- Binary and multi-class classification
- Building a text classification model

Introduction to Text Classification

Text Classfication

Text classification is a natural language processing task that involves c**ategorizing text documents into predefined categories or labels**, such as spam detection or sentiment analysis.



Machine Learning-Based Text Classification

With advance growth of machine learning, nowadays it is just easier to create model using machine learning and feed data to the model and wait until the model is complete. With the machine learning model, it's much easier and f**aster to classify category from input text**. One important step to use machine learning is **feature extraction**. We transform text to **numeral representation in from of vector, one way of doing it is using bags of word, or basically, we count every words in a text, or using tfidf (term frequency inverse document frequency)** etc.

Data Collection and Labeling

Machine Learning-Based Text Classification

+ Code + Text

[4]
```
df['sentiment_first_scheme'] = df['score'].apply(lambda x: 'negative' if x in [1, 2] else 'positive')

# Sentiment analysis based on the second scheme
df['sentiment_second_scheme'] = df['score'].apply(lambda x: 'negative' if x in [1, 2] else ('neutral' if x == 3 else 'positive'))
```

[7]
```
df.sentiment_second_scheme.value_counts()
```

```
negative    1717
positive    1243
neutral      228
Name: sentiment_second_scheme, dtype: int64
```

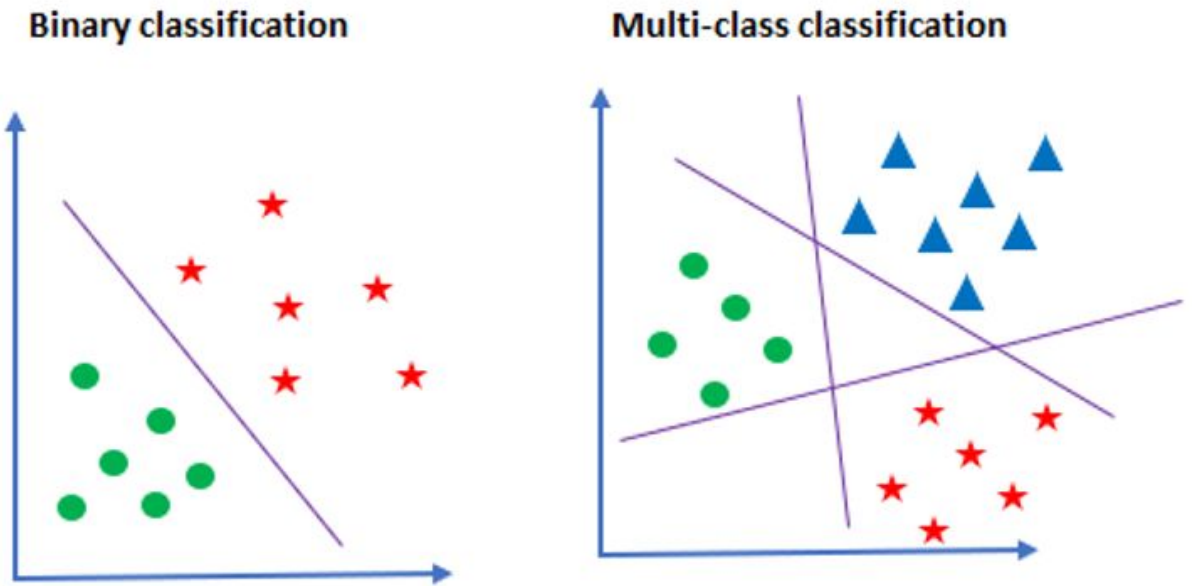
```
df.sentiment_first_scheme.value_counts()
```

```
negative    1717
positive    1471
Name: sentiment_first_scheme, dtype: int64
```

Binary and multi-class classification

Binary vs Multiclass

Binary classification involves categorizing data into **two distinct classes**, while multi-class classification extends this task to categorize data into **three or more classes**, making it suitable for a broader range of real-world applications.



Binary vs Multiclass

Binary Classification:

- Binary classification focuses on distinguishing between two exclusive classes or categories, typically referred to as the **positive class and the negative class**.
- Techniques such as logistic regression, support vector machines, and decision trees can be directly applied to binary classification.
- Evaluation metrics often include **accuracy, precision, recall, F1-score, ROC-AUC, and others tailored to two-class scenarios**.

Multi-class Classification:

- Multi-class classification involves categorizing data into three or more classes or categories, each mutually exclusive.
- Common techniques for multi-class classification include **one-vs-all (OvA or OvR)**, **softmax regression (logistic regression extension)**, and neural network architectures designed for multiple output classes.
- Evaluation metrics differ and can include **macro/micro-average F1-score, cross-entropy loss, confusion matrices, and accuracy calculated across all classes**.

Binary vs Multiclass

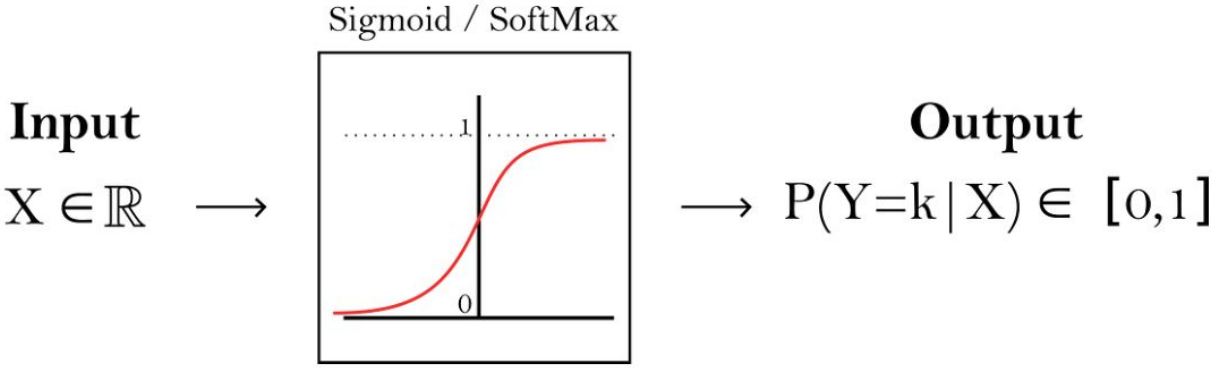


Figure 1. Illustration of Sigmoid and SoftMax function. The output is read as "The probability of Y being the class k given the input X". Image by author

But if both functions map the same transformation (i.e. do the same thing), what is the difference between them?

Sigmoid is used for **binary classification** methods where we only have 2 classes, while SoftMax applies to **multiclass problems.** In fact, the SoftMax function is an extension of the Sigmoid function.

Binary vs Multiclass

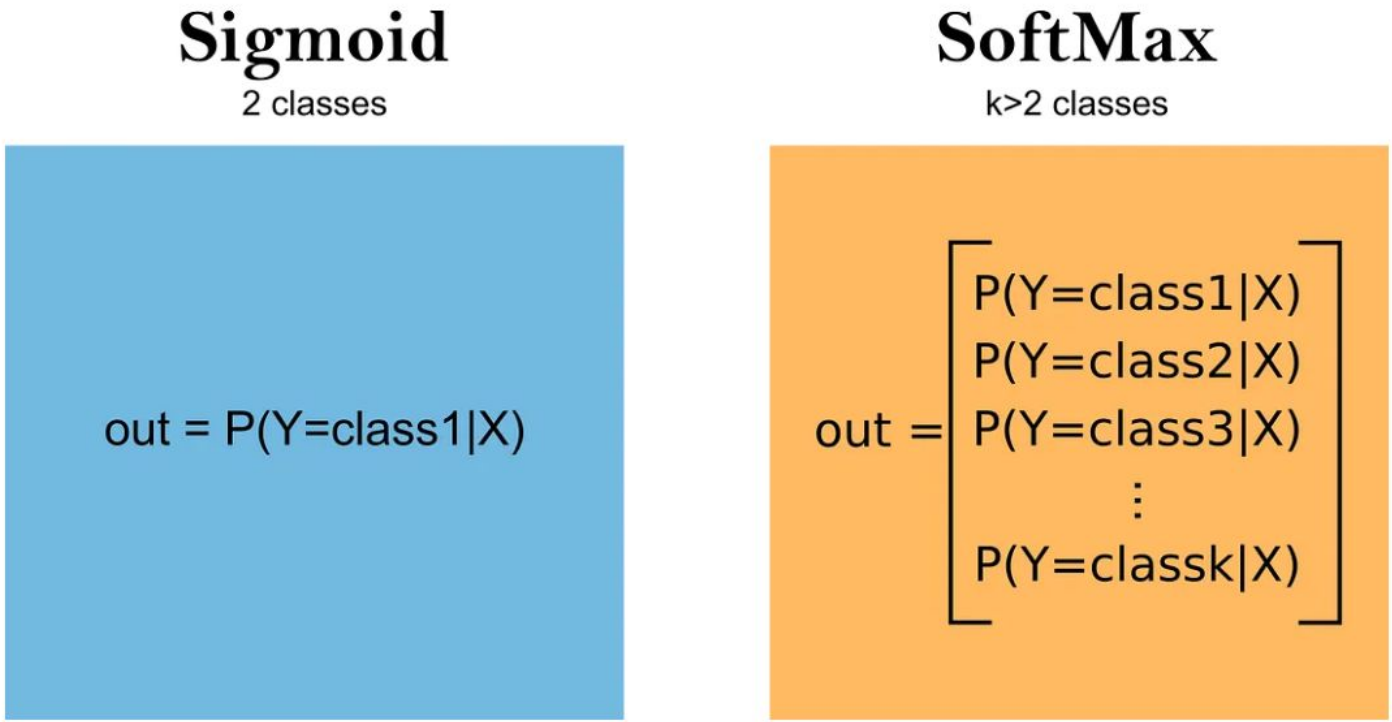


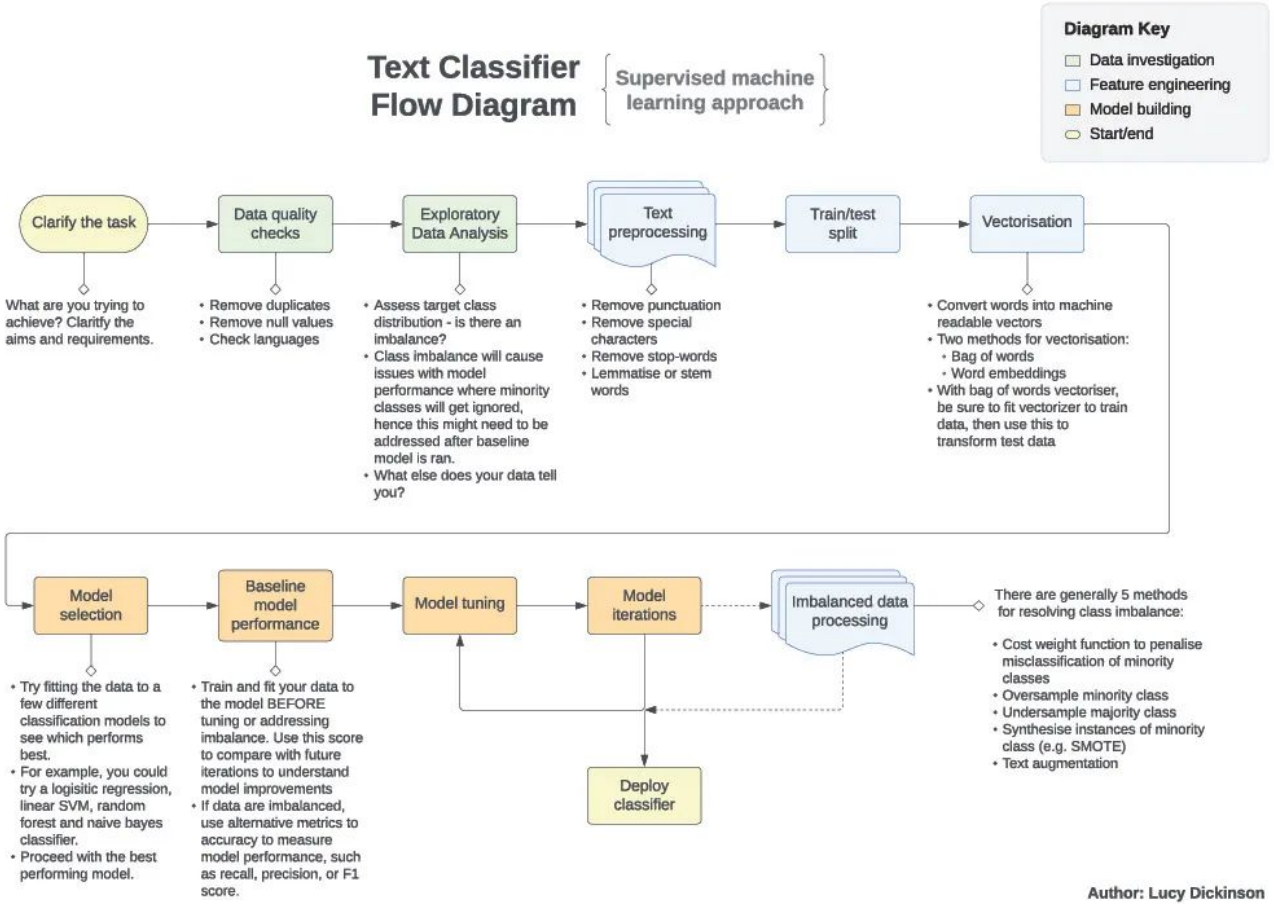
Figure 2. Illustration of the outputs of each function. An important property is the addition of all the probabilities for each Sigmoid class and SoftMax should be equal to 1. In the case of Sigmoid we obtain P(Y=class2|X) = 1 - P(Y=class1|X). Image by author

Modeling Process

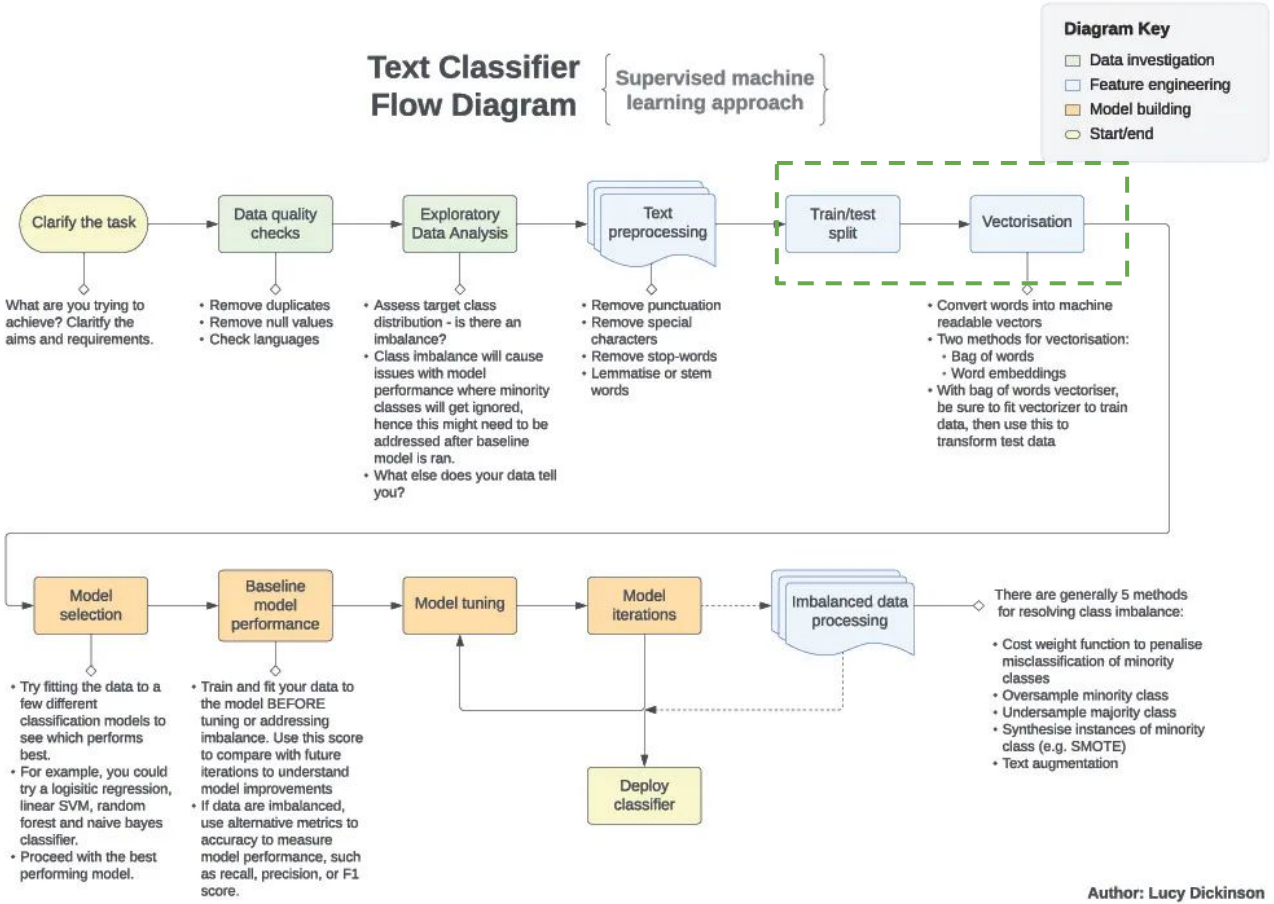
Modeling Process

In a machine learning pipeline, what should be the **first step**: splitting the data into training and testing sets or preprocessing steps like imputation, SMOTE (Synthetic Minority Over-sampling Technique), or scaling data?

Modeling Process

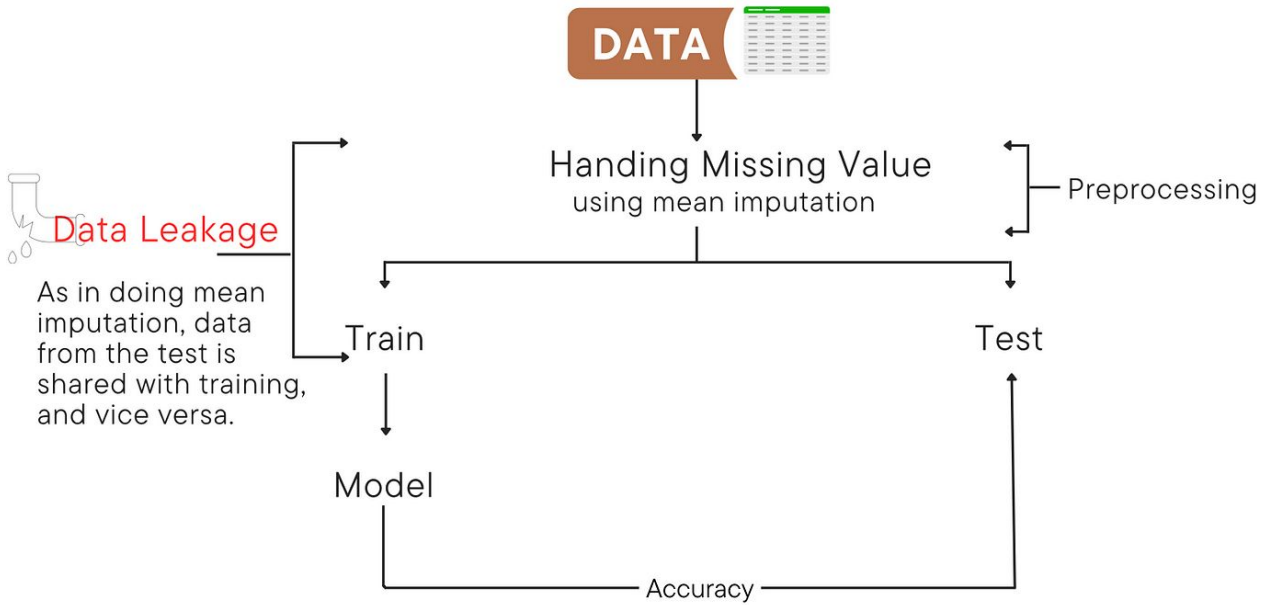


Modeling Process

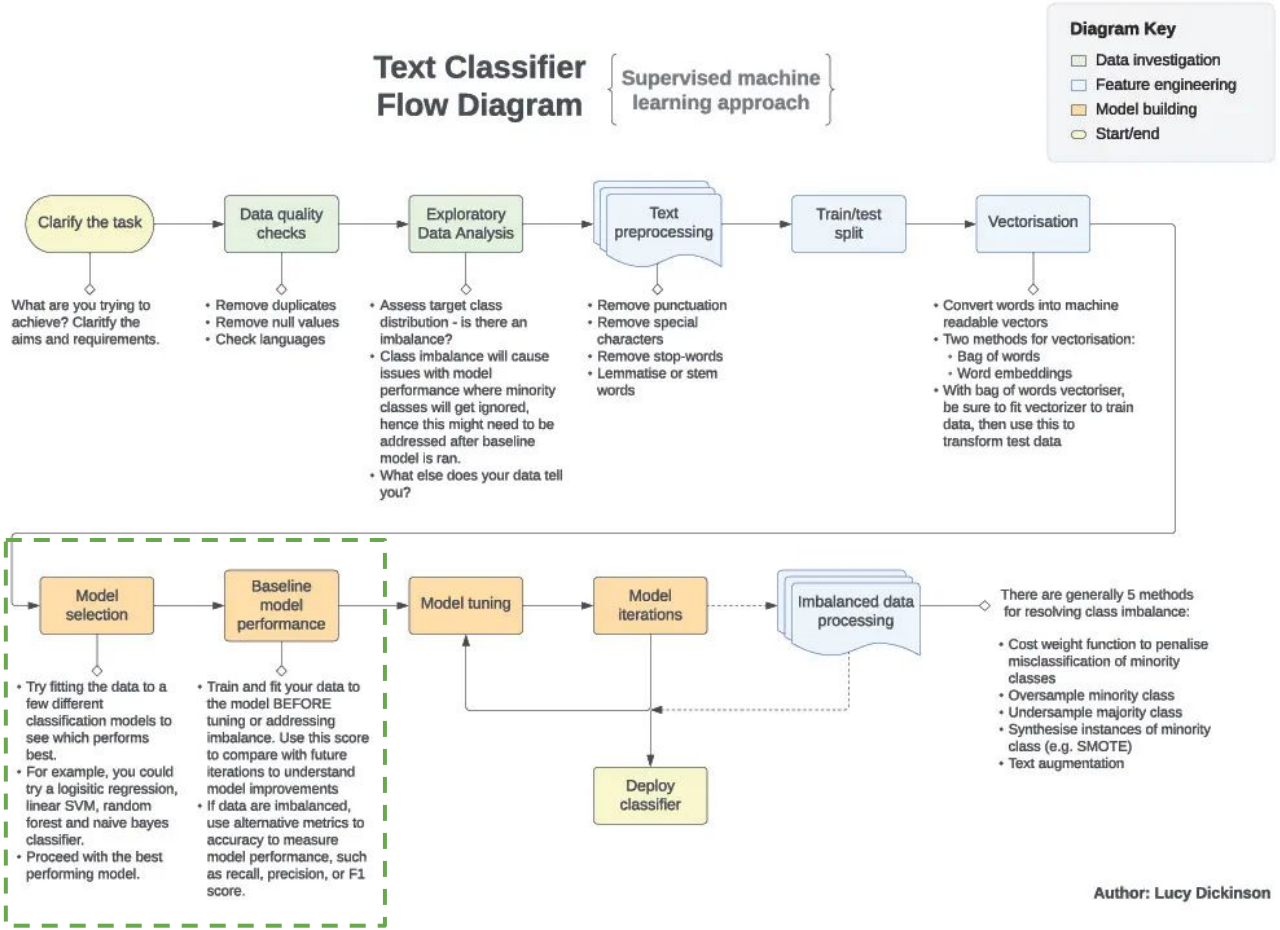


Data Leakage Problem

This is getting its own sub heading because it is so important to do this step **BEFORE** your start fiddling with the features. Split your data using sklearn's **train_test_split() function** and then leave the **test data alone** so there's no risk of **data leakage.**



Modeling Process



Classification Evaluation Metrics

Classification evaluation metrics

In contrast to regression, classification metrics have more variations and must be evaluated carefully. Some metrics are useful for binary and multiclass classification. Some metrics may be chosen when the proportion among classes severely differs.

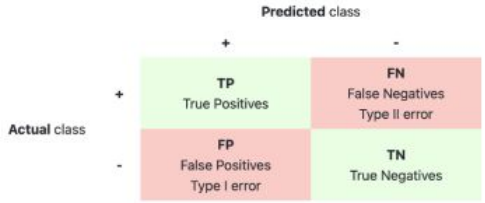
Often looking at accuracy can be misleading, basically when the data is unbalanced (do you know why? Hint try to calculate the formula when you have 1:90 proportion). There are examples of when to choose other evaluation metrics when accuracy alone is not enough.

Metric	Formula	Interpretation
Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$	Overall performance of model
Precision	$\frac{TP}{TP + FP}$	How accurate the positive predictions are
Recall Sensitivity	$\frac{TP}{TP + FN}$	Coverage of actual positive sample
Specificity	$\frac{TN}{TN + FP}$	Coverage of actual negative sample
F1 score	$\frac{2TP}{2TP + FP + FN}$	Hybrid metric useful for unbalanced classes

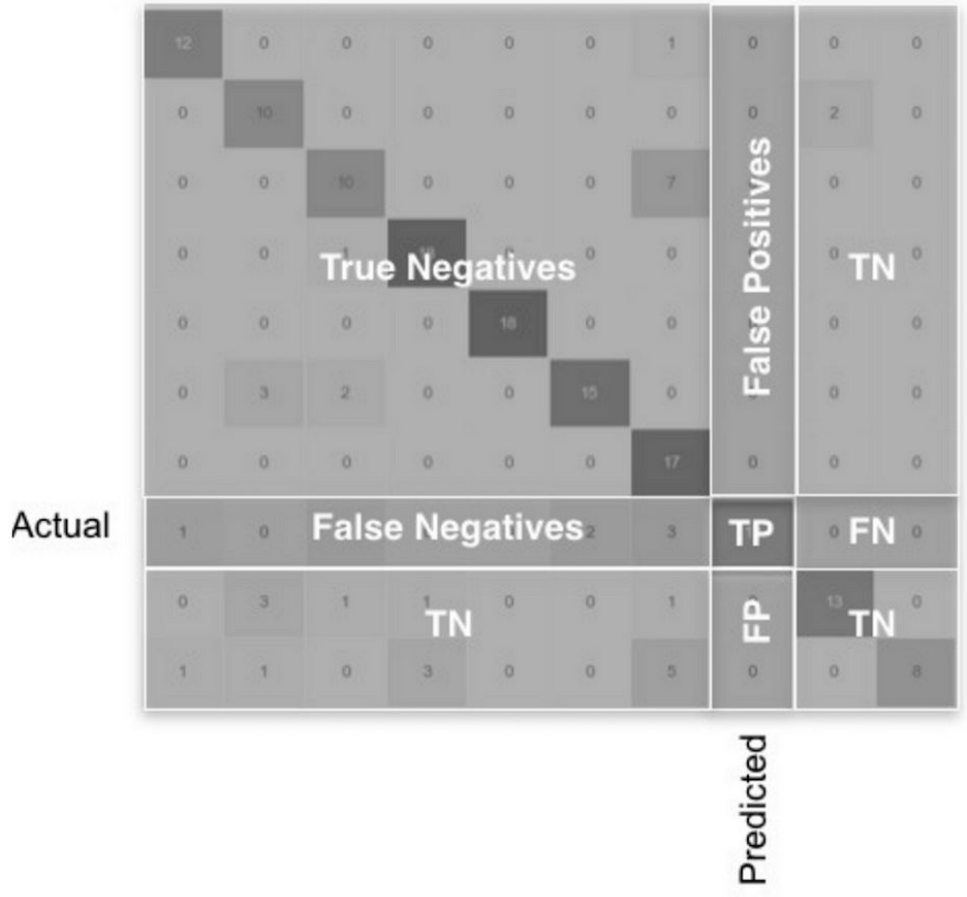
Choose Recall if the idea of false positives is far better than false negatives, in other words, **if the occurrence of false negatives is unaccepted/intolerable**, that you'd rather get some extra false positives(false alarms) over saving some false negatives, like in our diabetes example.
You'd rather get some healthy people labeled diabetic over leaving a diabetic person labeled healthy.

Choose precision if you want to be more confident of your true positives. for example, Spam emails. You'd rather have some spam emails in your inbox rather than some regular emails in your spam box. So, the email company wants to be extra sure that email *Y* is spam before they put it in the spam box and you never get to see it.

Choose Specificity if you want to cover all true negatives, meaning you don't want any false alarms, you don't want any false positives. for example, you're running a drug test in which all people who test positive will immediately go to jail, you don't want anyone drug-free going to jail. False positives here are intolerable.



Confession Matrix Multiclass



Python Time

https://github.com/muhajirakbarhsb/NLP_class_2023/blob/main/Week_Class_5.ipynb