# Email Spam Classification

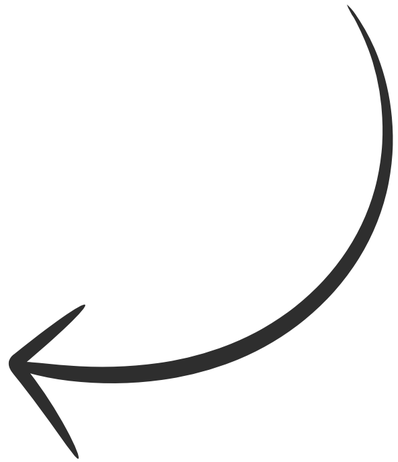
## Using semi-supervised learning and natural language processing techniques !

( Self-learning, Label propagation and Label spreading algorithms )

ESI-SBA | Artificial Intelligence and Data Science

# Agenda

- Brief Overview
- Problem Statement
- Project Objectives
- Dataset Description
- Methodology
  - Natural Language Processing
  - Semi-Supervised Learning
  - Training Results
- Best Model
- Model Deployment
- Conclusion



# Introduction

Text classification problems are among the most extensively addressed and widely applied topics in real-life applications, such as spam detection !

This is especially true with the recent breakthroughs in natural language processing techniques, which have significantly eased the handling of text data by machine learning algorithms.

In this project we will explore natural language processing and semi-supervised learning techniques to build a model that robustly distinguish spam and non spam emails.

# What is the challenge?

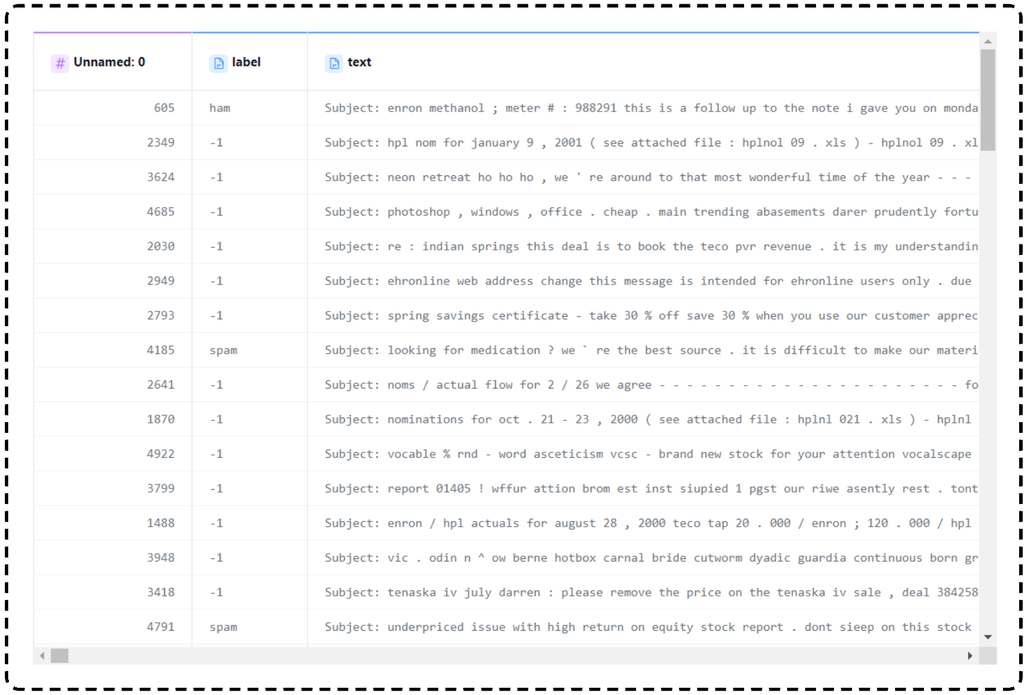
Email communication has become an integral part of our daily lives, but so has the incessant influx of spam messages.

With the increasing sophistication of spammers, traditional spam filters often fall short in accurately distinguishing between genuine and unwanted emails.

# What are the objectives?

Our primary objectives are to achieve higher accuracy in spam classification, explore the benefits of semi-supervised learning, and leverage NLP to enhance the system's ability to decipher complex language patterns in emails.

# Some information about the dataset



| Unnamed: 0 | label | text |
|---|---|---|
| 605 | ham | Subject: enron methanol ; meter # : 988291 this is a follow up to the note i gave you on monda |
| 2349 | -1 | Subject: hpl nom for january 9 , 2001 ( see attached file : hplnol 09 . xls ) - hplnol 09 . xl |
| 3624 | -1 | Subject: neon retreat ho ho ho , we ' re around to that most wonderful time of the year - - - |
| 4685 | -1 | Subject: photoshop , windows , office . cheap . main trending abasements darer prudently fortu |
| 2030 | -1 | Subject: re : indian springs this deal is to book the teco pvr revenue . it is my understandin |
| 2949 | -1 | Subject: ehronline web address change this message is intended for ehronline users only . due |
| 2793 | -1 | Subject: spring savings certificate - take 30 % off save 30 % when you use our customer apprec |
| 4185 | spam | Subject: looking for medication ? we ` re the best source . it is difficult to make our materi |
| 2641 | -1 | Subject: noms / actual flow for 2 / 26 we agree - - - - - - - - - - - - - - - - - - - - - - fo |
| 1870 | -1 | Subject: nominations for oct . 21 - 23 , 2000 ( see attached file : hplnl 021 . xls ) - hplnl |
| 4922 | -1 | Subject: vocable % rnd - word asceticism vcsc - brand new stock for your attention vocalscape |
| 3799 | -1 | Subject: report 01405 ! wffur attion brom est inst siupied 1 pgst our riwe asently rest . tont |
| 1488 | -1 | Subject: enron / hpl actuals for august 28 , 2000 teco tap 20 . 000 / enron ; 120 . 000 / hpl |
| 3948 | -1 | Subject: vic . odin n ^ ow berne hotbox carnal bride cutworm dyadic guardia continuous born gr |
| 3418 | -1 | Subject: tenaska iv july darren : please remove the price on the tenaska iv sale , deal 384258 |
| 4791 | spam | Subject: underpriced issue with high return on equity stock report . dont sieep on this stock |

The dataset consist of **5171** rows and **03** columns:

### Unnamed

An irrelevant column which will be dropped.

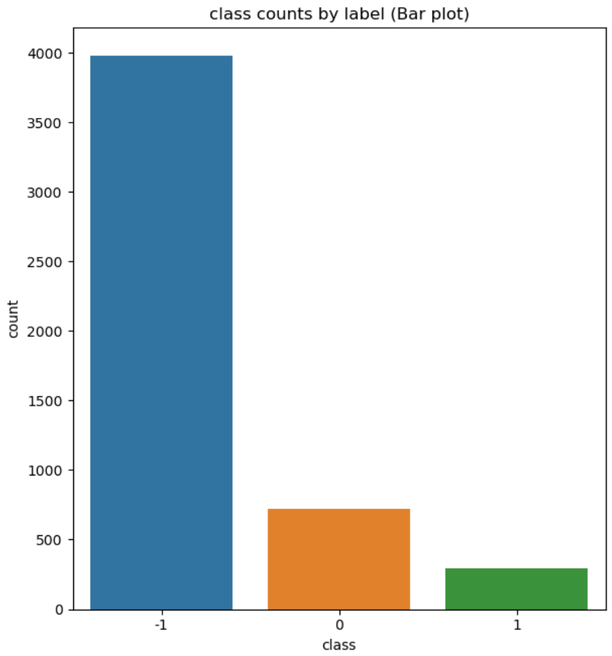
### Label

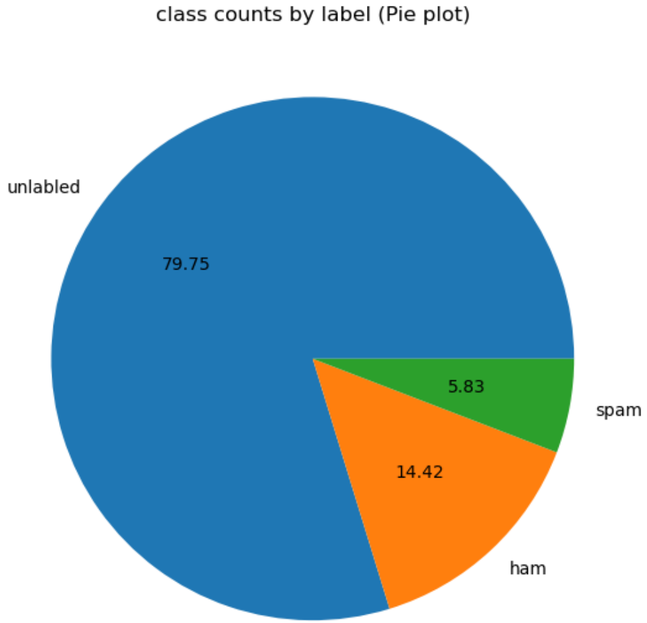
The class column for prediction. It contain 3 values: spam, ham and -1 for unlabeled rows.

### Text

The email content concatenated with the subject title.

# Class count by label type

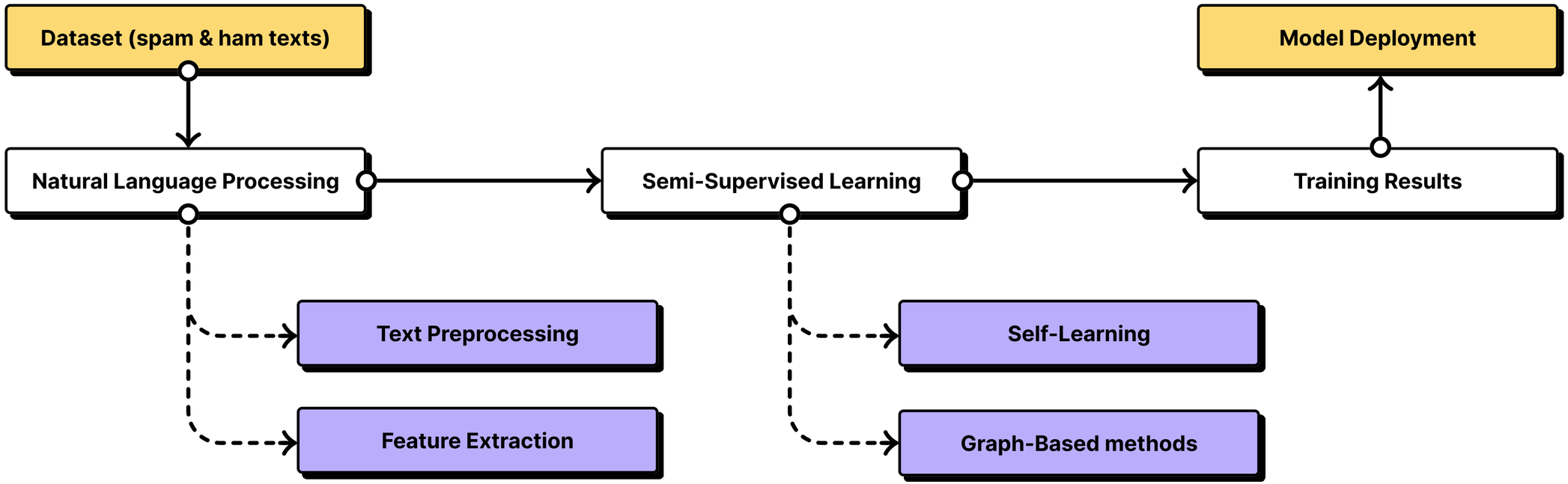



## here’s a constraint we had

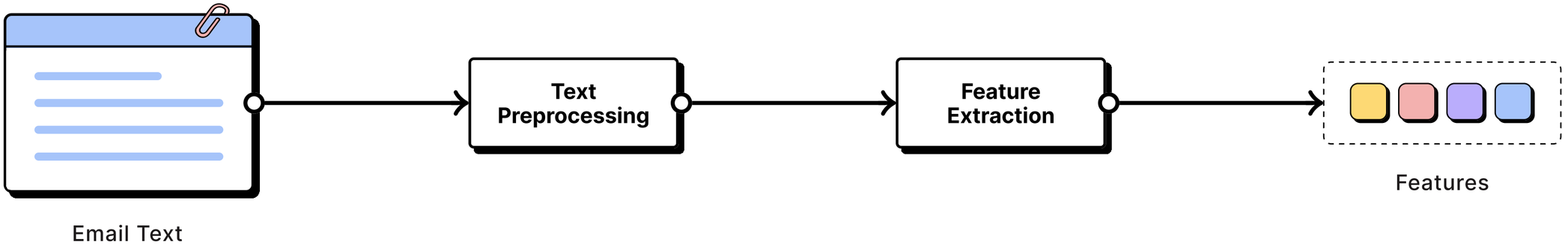
Identify any assumptions and constraints you were dealing with.  What stood in the way of your team achieving your goal? Were there any blockers? What was the timeline? What’s the scope? What’s the risk?

# A Comprehensive Approach



# Natural Language Processing

Natural Language Processing (NLP) is a subfield of Artificial Intelligence and linguistics, It is concerned with giving machines the ability to understand, interpret, and interact with human language text.



# 1. Text Preprocessing

**Sentence:** ”Embark on the journey of self-discovery with an open heart and a curious mind.”

**Tokenization:** {‘Embark’ , ‘on’ , ‘the’ , ‘journey’ , ‘of’ , ‘self-discovery’ , ‘with’ , ‘an’ , ‘open’ , ‘heart’ , ‘and’ , ‘a’ , ‘curious’ , ‘mind’ , ‘.’}

**Stop words removal:** {‘Embark’ , ‘journey’ , ‘self-discovery’ , ‘open’ , ‘heart’ , ‘curious’ , ‘mind’ , ‘.’}

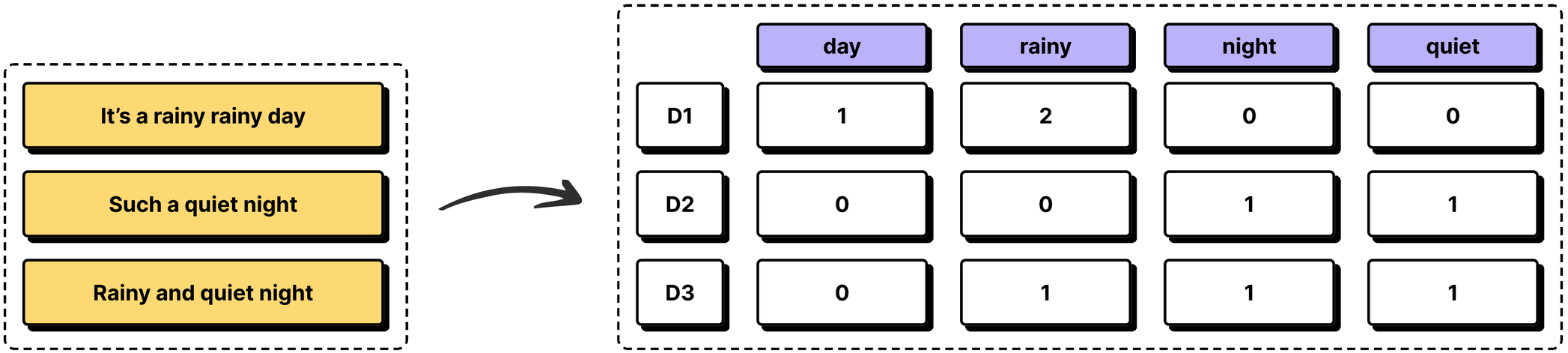
**Capitalization:** {‘embark’ , ‘journey’ , ‘self-discovery’ , ‘open’ , ‘heart’ , ‘curious’ , ‘mind’ , ‘.’}

**Noise removal:** {‘embark’ , ‘journey’ , ‘self-discovery’ , ‘open’ , ‘heart’ , ‘curious’ , ‘mind’}

**Stemming:** {‘embark’ , ‘journey’ , ‘self-discov’ , ‘open’ , ‘heart’ , ‘curious’ , ‘mind’}

# 2. Feature Extraction

**The bag of words:** is one particularly simple technique to represent documents in numerical form before we can feed it into a machine learning algorithm.




| | day | rainy | night | quiet |
|---|---|---|---|---|
| D1 | 1 | 2 | 0 | 0 |
| D2 | 0 | 0 | 1 | 1 |
| D3 | 0 | 1 | 1 | 1 |

# 2. Feature Extraction

**Term Frequency-Inverse Document Frequency (TF-IDF):** is another frequency-based feature extraction technique that assigns higher weights to words with very high or very low frequency. The value of this weight is given by the following formula:

$$w_{i,j} = tf_{i,j} \times log(\frac{N}{df_i})$$

Where:

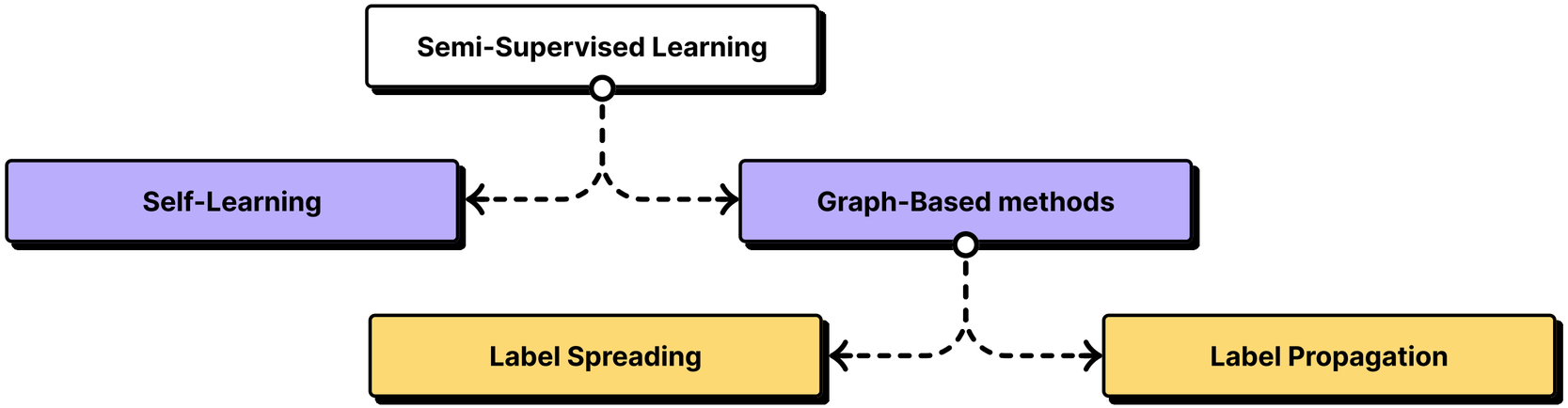
$tf_{i,j}$ : is the frequency of the term i in the document j

$df_i$ : is the number of documents that contains the term i

$N$ : is the total number of documents

# Semi-Supervised Learning

Semi-supervised learning is a broad category of machine learning techniques that utilizes both labeled and unlabeled data. In this way, as the name suggests, it is a hybrid technique between supervised and unsupervised learning.



# 1. Self-Learning Algorithm

Self-Learning is a machine learning technique where a model is iteratively trained on a combination of labeled and unlabeled data. Initially, the model is trained on a small labeled dataset, then it is used to make predictions on the unlabeled data.

**Algorithm 1** Self training algorithm

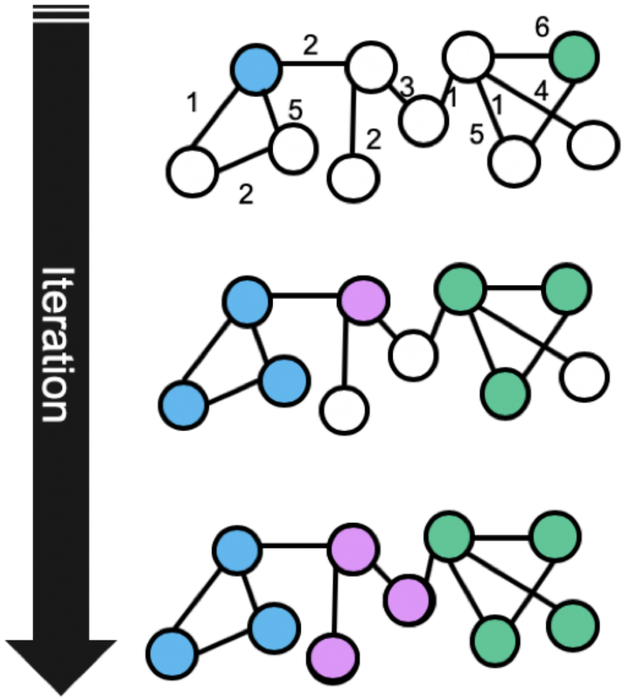
**Require:** model : a probabilistic model
**Require:** $threshold \in [0, 1]$
**Require:** $max\_iter$ a positive integer.
  $trainset \leftarrow labled\_set$
  $stop \leftarrow False$
  $n \leftarrow 0$
  **while** not $stop$ **do**
    train(model, trainset.X, trainset.y)
    $\hat{y} \leftarrow predict(model, unlabled\_set)$
    $trainset \leftarrow trainset \bigcup unlabled\_set[\text{argmax}\, \hat{y} \geq threshold]$
    $n \leftarrow n + 1$
    $stop \leftarrow unlabled\_set[\text{argmax}\, \hat{y} \geq threshold]$ is empty or $n >= max\_iter$
  **end while**

# 2. Graph-Based Methods

**Label Propagation:** is a graph-based algorithms used in semi-supervised learning. Its primary objective is to propagate labels throughout a similarity graph built based on the distance between data points. Once the labels are propagated, the algorithm uses this information to classify unlabeled data points.

**Algorithm 2** Label propagation : Zhu and Ghahramani

compute the weight matrix $w$
compute diagonal matrix $D_{i,i} = \sum_i W_{i,j}$
initialize $\hat{Y}^{(0)}$
**repeat**
$\hat{Y}^{t+1} \leftarrow D^{-1}W\hat{Y}^t$
$\hat{Y}_l^{t+1} \leftarrow Y_l^t$
**until** convergence



# 2. Graph-Based Methods

**Label Spreading:** is a graph-based semi-supervised learning method, same as Label Propagation. The key difference lies in the design of the transition matrix. Label propagation uses the graph Laplacian while Label spreading uses the normalized graph Laplacian.

**Algorithm 3** Label spreading : Zhou et al

**Require:** $\alpha \in [0, 1)$

compute the weight matrix $w$ with $w_{i,i} \leftarrow 0$

compute diagonal matrix $D_{i,i} = \sum_i W_{i,j}$

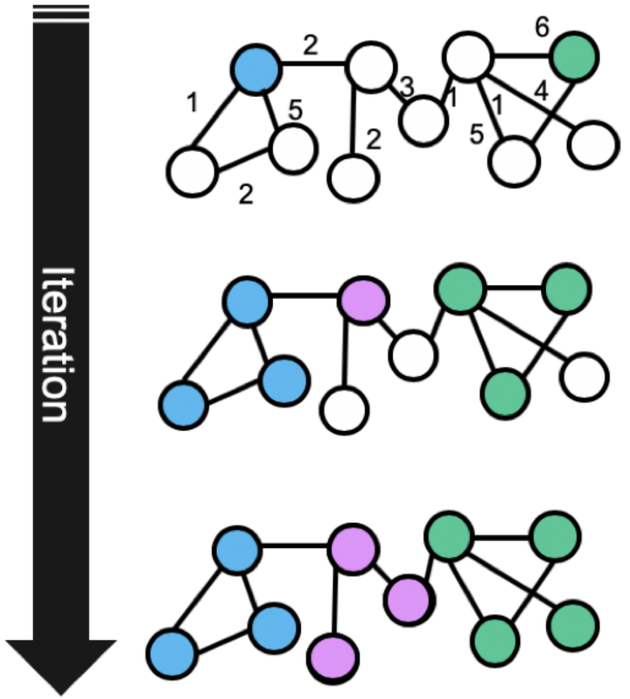
compute the matrix $L \leftarrow D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$

initialize $\hat{Y}^{(0)}$

**repeat**

$\hat{Y}^{t+1} \leftarrow \alpha L \hat{Y}^t + (1 - \alpha)\hat{Y}^t$

**until** convergence



# Training Results

To compare the different models, we've chosen the **F1-score** as the criteria for selecting the best model.
This decision is driven by the dataset's imbalance, making accuracy unreliable.

**Self-Learning algorithm:**

| Base model | accuracy | precision | recall | f1-score |
|---|---|---|---|---|
| SGD Logistic Regression | 0.976 | 0.988 | 0.936 | 0.962 |
| SVM | 0.950 | 0.976 | 0.863 | 0.916 |
| Logistic Regression | 0.980 | 0.989 | 0.947 | 0.967 |
| Complement Naive Bayes | 0.976 | 0.968 | 0.957 | 0.962 |

# Training Results

**Label Spreading algorithm:**

| Feature extraction | kernel | Dimentiality reduction | accuracy | precision | recall | f1-score |
|---|---|---|---|---|---|---|
| Bag of words | rbf | no | 0.763 | 0.632 | 0.364 | 0.46 |
| Bag of words | rbf | yes | 0.845 | 0.720 | 0.729 | 0.725 |
| Bag of words | knn | no | 0.789 | 0.576 | 0.929 | 0.711 |
| Bag of words | knn | yes | 0.861 | 0.718 | 0.847 | 0.774 |
| TF-IDF | rbf | no | 0.907 | 0.851 | 0.811 | 0.831 |
| TF-IDF | rbf | yes | 0.657 | 0.449 | 1.000 | 0.624 |
| TF-IDF | knn | no | 0.914 | 0.927 | 0.752 | 0.831 |
| TF-IDF | knn | yes | 0.805 | 0.597 | 0.941 | 0.730 |

# Training Results

**Label Propagation algorithm:**

| Feature extraction | kernel | Dimentiality reduction | accuracy | precision | recall | f1-score |
|---|---|---|---|---|---|---|
| Bag of words | rbf | no | 0.743 | 0.563 | 0.364 | 0.442 |
| Bag of words | rbf | yes | 0.819 | 0.674 | 0.682 | 0.678 |
| Bag of words | knn | no | 0.730 | 0.509 | 0.964 | 0.666 |
| Bag of words | knn | yes | 0.812 | 0.614 | 0.882 | 0.724 |
| TF-IDF | rbf | no | 0.891 | 1.000 | 0.611 | 0.759 |
| TF-IDF | rbf | yes | 0.592 | 0.406 | 1.000 | 0.578 |
| TF-IDF | knn | no | 0.917 | 0.954 | 0.741 | 0.834 |
| TF-IDF | knn | yes | 0.756 | 0.964 | 0.535 | 0.689 |

According to the chosen metric for this particular problem, which is the **F1-score** for the reasons discussed earlier, the best model is the Self-Learning classifier with Logistic Regression as the base model.

# Model Deployment & Testing

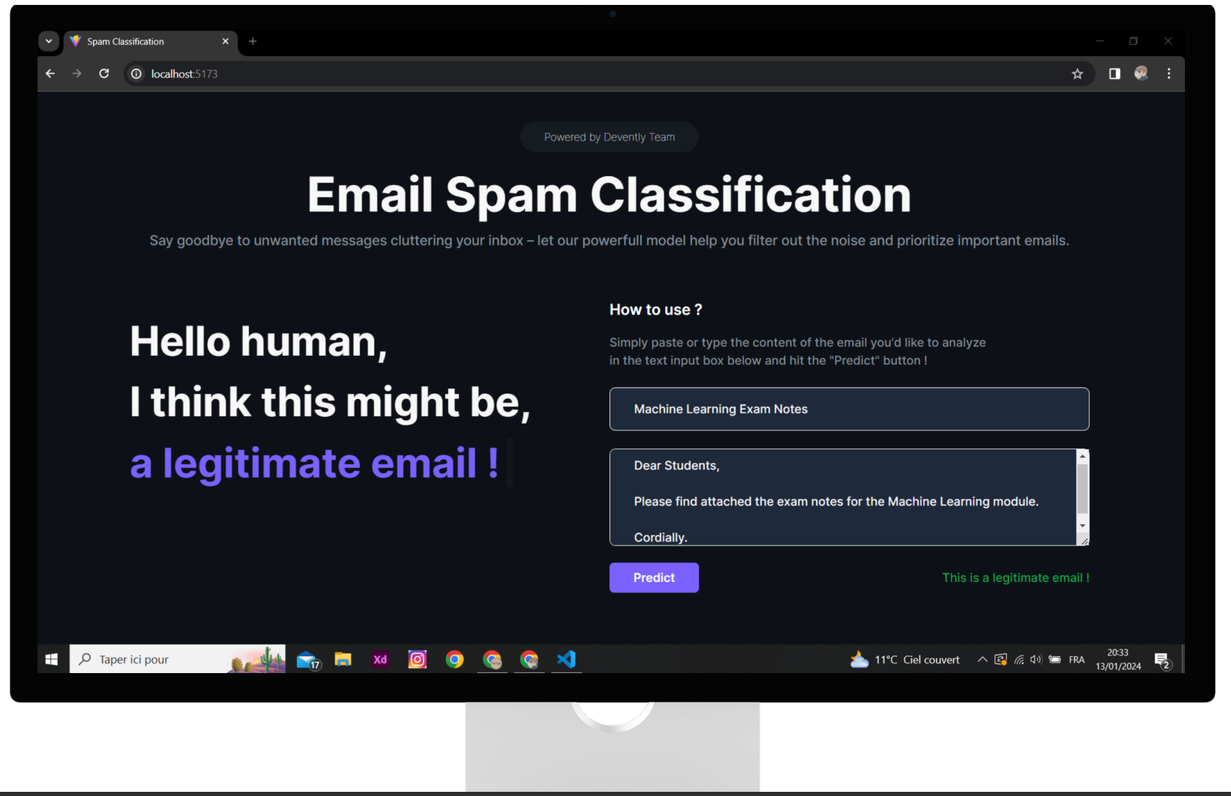
**Example 01:** ( Legitimate email )

**MACHINE LEARNING EXAM NOTES**

Dear Students,

Please find attached the exam notes for Machine Learning module.

Cordially.



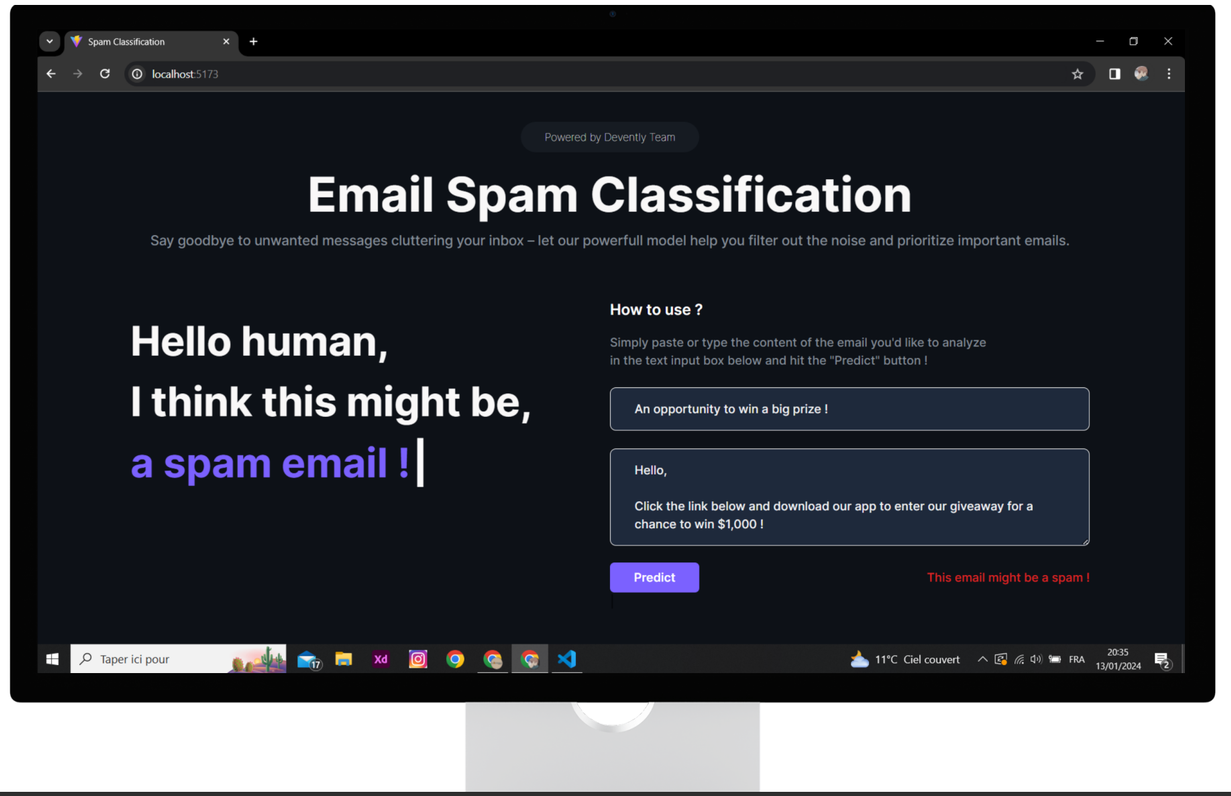
# Model Deployment & Testing

**Example 02:** ( Spam email )

**AN OPPORTUNITY TO WIN A BIG PRIZE**

Hello,

Click the link below and download our app to enter our giveaway for a chance to win $1000 !



# Summary & Conclusion

Spam is a major problem in today's world, which necessitates a strong spam filter to distinguish between genuine and unwanted emails.

In this project, we explored three different semi-supervised learning algorithms and discussed their implementation. We applied these algorithms to an email spam classification dataset, combining semi-supervised learning and NLP techniques to build a model capable of robustly identifying spam from non-spam emails.

Github Repository

You can access the full project **here** if you want to see all the details and test the model.

# Thanks for your Listening! ☺


- Abdelnour FELLAH - ( ab.fellah@esi-sba.dz )
- Adel Abdelkader MOKADEM - ( aa.mokadem@esi-sba.dz )
- Yacine Lazreg BENYAMINA - ( yl.benyamina@esi-sba.dz )
- Abderrahmene BENOUNENE - ( a.benounene@esi-sba.dz )