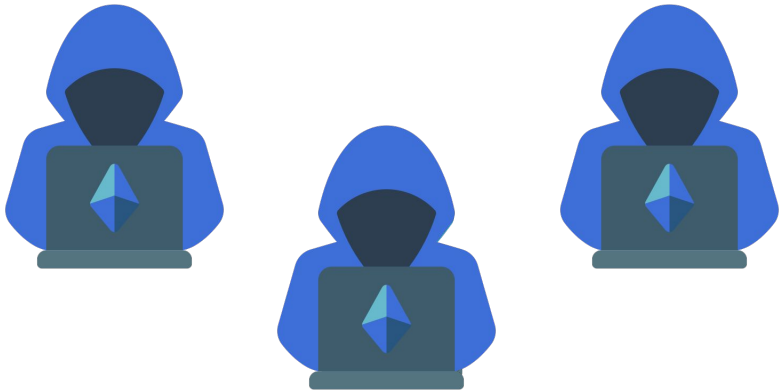
ITA DATA Hack

Team: Unisa Data Force



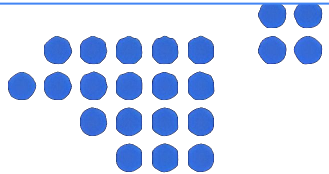
opendata PLAYGROUND

Day 1: Problem definition

Classification of chapters representing the laws in force in the European Union

There are 20 chapters to be classified, i.e. the hierarchy of standards in force in the European Union

This work is needed so that documents can be organised in a systematic way



opendata PLAYGROUND

Data Preparation

```
REPLACE_BY_SPACE_RE = re.compile('[/(){}\[\]\|@,;]')
BAD_SYMBOLS_RE = re.compile('[^0-9a-z #+_]')
STOPWORDS = set(stopwords.words('english'))

def clean_text(text):
    """
        text: a string

        return: modified initial string
    """
    text = BeautifulSoup(text, "lxml").text
    text = text.lower()
    text = REPLACE_BY_SPACE_RE.sub(' ', text)
    text = BAD_SYMBOLS_RE.sub('', text)
    text = ' '.join(word for word in text.split() if word not in STOPWORDS)
    return text
```

opendata PLAYGROUND

Day 1: Passive Aggressive

```
PA = Pipeline([('vect', CountVectorizer()),
               ('tfidf', TfidfTransformer()),
               ('rf', PassiveAggressiveClassifier(max_iter=1000,
                                                  random_state=0,tol=1e-3))])
```

If the error between the prediction and the actual value is within a certain threshold, the model remains "**passive**" and does not change the weigh

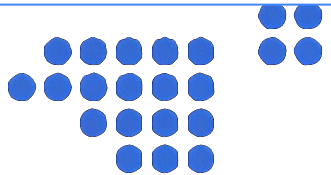
If the error exceeds the threshold, the model behaves '**aggressively**' and updates the weights

Day 2: Problem definition

Classification of chapters and sub-chapters representing the laws in force in the European Union

Each law can be uniquely attributed to a single sub-chapter within the hierarchy of laws in the European Union

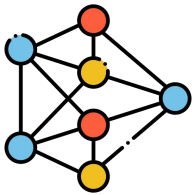
This work is needed so that documents can be organised in a systematic way



opendata PLAYGROUND

Day 2: MLPClassifier model

```
MLP = Pipeline([('vect', CountVectorizer()),
                ('tfidf', TfidfTransformer()),
                ('rf', MLPClassifier(random_state=1, max_iter=300))])
```



MLPClassifier is a machine learning algorithm for classification. Based on **artificial neural networks**, it learns from labelled data, capturing complex relationships in the data.

After training, it can make predictions on new data by assigning them a category or class. **Optimisation of parameters** is crucial to achieve the best performance

Day 3: Problem definition

Classification of chapters and sub-chapters representing the laws in force in the European Union

Each law can be associated with different directory codes, corresponding to the second level of hierarchy in the European Union classification.

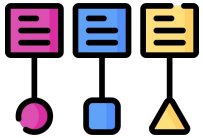
This work is needed so that documents can be organised in a systematic way

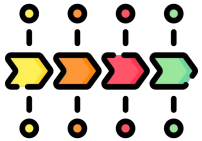
opendata PLAYGROUND

Day 3: Chain classifier and LinearSVC

```
CS = Pipeline([('vect', CountVectorizer()),
               ('tfidf', TfidfTransformer()),
               ('rf', ClassifierChain(LinearSVC(C= 3, random_state=0, tol=1e-4,
                class_weight="balanced"), order='random', random_state=0))])
```

Classifier Chain is a machine learning approach for multi-class classification that employs a **chain of binary classifiers**

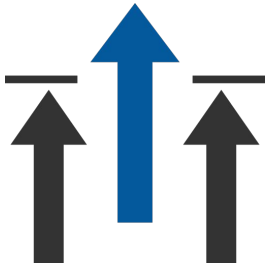




Embedded within this chain is the classifier LinearSVC, known for handling **linearly separable data** and configurable via the regularisation parameter C.

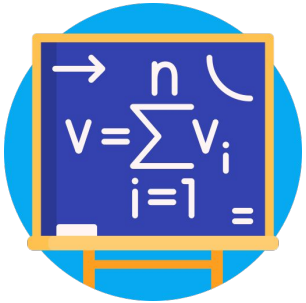
What ITADATAHACK has taught us

Overcoming problems of increasing difficulty through logical thinking



Staying on the edge: Every day pushing ourselves to do more than our best

Apply the theory studied at university in a real problem and understand our real potential



opendata
PLAYGROUND

Team UNISA

Raffaele Ferriero

Student at the University of Salerno

Rosaria Leone

Student at the University of Salerno

Giuseppe Genito

Student at the University of Salerno

opendata PLAYGROUND

Thank you all for your attention

We would like to thank OpenData Playground for organising the challenge

UNISA DATA FORCE

opendata PLAYGROUND