TWEET PREDICTION CHALLENGE INF554

ALHUSSEIN JAMIL – EMMANUEL GNABEYEU – YOUSSOUFA TALBA

PROJECT ORGANIZATION

- Brainstorming
  - Review of state-of-the-art bibliography
  - Look for related problems and how they are solved, the relevant libraries
  - Intuitions at that stage

PROJECT ORGANIZATION

- Main steps:
  - Preprocessing
  - Text processing
  - ML model choice
- Tasks distribution

FEATURES SELECTION

- Data Cleaning:

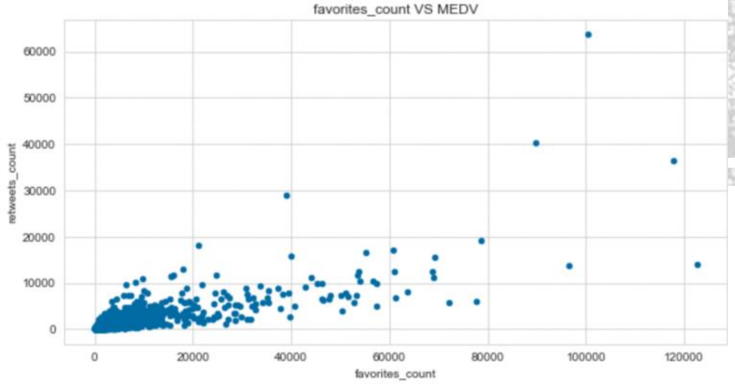
We dropped unusable columns like ’Urls’ and ’mention’

- Statistical correlation analysis

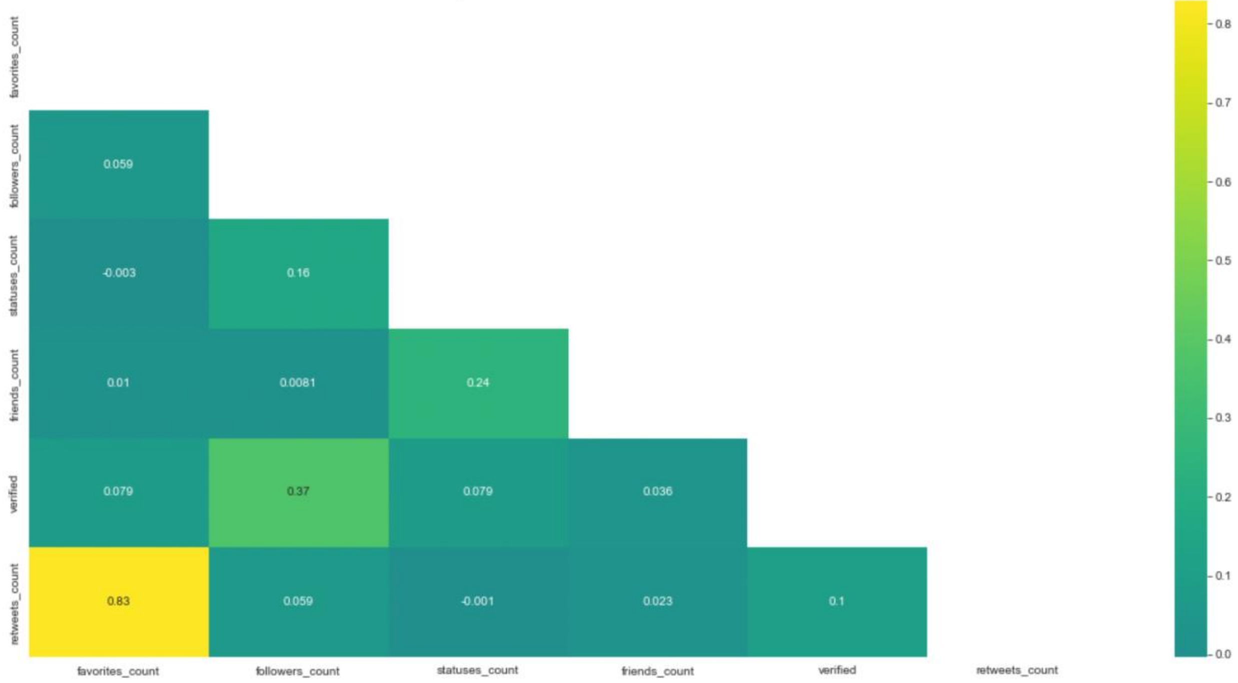
over numerical features:

to Choose the most important variables:

  - Correlation matrix
  - Scatter Plot
  - Feature Importance Scikit-Learn

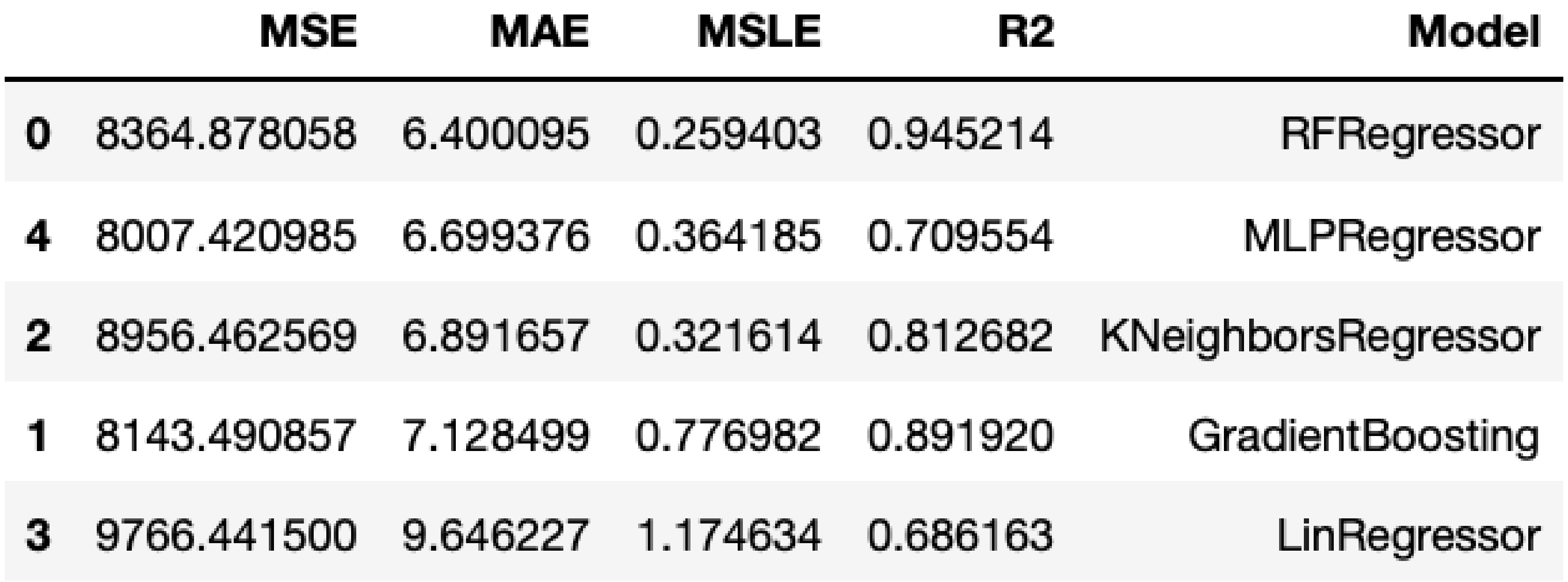


Heatmap of co-relation between variables



FEATURES SELECTION

- Simple models testing (LinearRegressors, KNNRegressor)



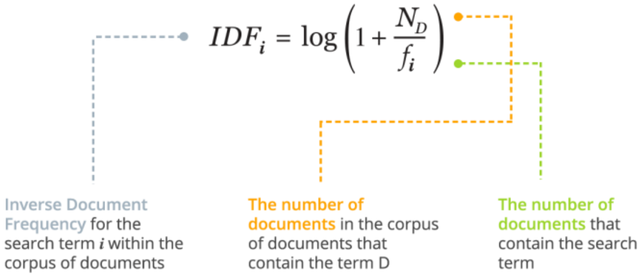
	MSE	MAE	MSLE	R2	Model
0	8364.878058	6.400095	0.259403	0.945214	RFRegressor
4	8007.420985	6.699376	0.364185	0.709554	MLPRegressor
2	8956.462569	6.891657	0.321614	0.812682	KNeighborsRegressor
1	8143.490857	7.128499	0.776982	0.891920	GradientBoosting
3	9766.441500	9.646227	1.174634	0.686163	LinRegressor

TEXT PROCESSING

1) Sentiment Analysis:
- Interesting, why ?
- Categorical value
- Textblob package

2) TF-IDF Analysis
- Useful 1st approach
- 100 words

$$TF = \left(\frac{\textit{Number of times keyword is found in document}}{\textit{Number of words in document}} \right)$$



$$IDF_i = \log\left(1 + \frac{N_D}{f_i}\right)$$

Inverse Document Frequency for the search term *i* within the corpus of documents

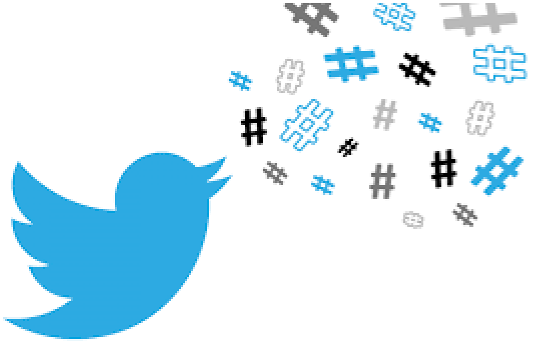
The number of documents in the corpus of documents that contain the term D

The number of documents that contain the search term

TEXT PROCESSING

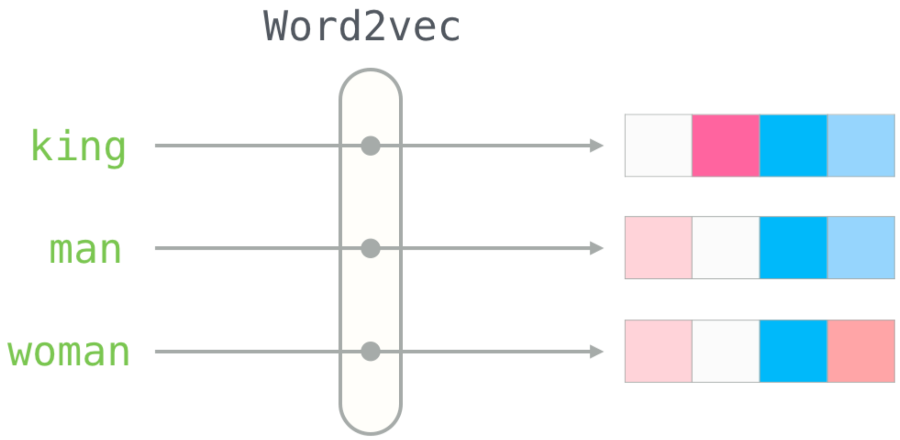
3) Hashtags processing :

- Carry interesting information
- 18% of data
- Crammed



4) Word2Vec

- Used in NLP, uses NN
- Pretrained by Google
- Captures word similarity
- Implemented using mean vector over tweet



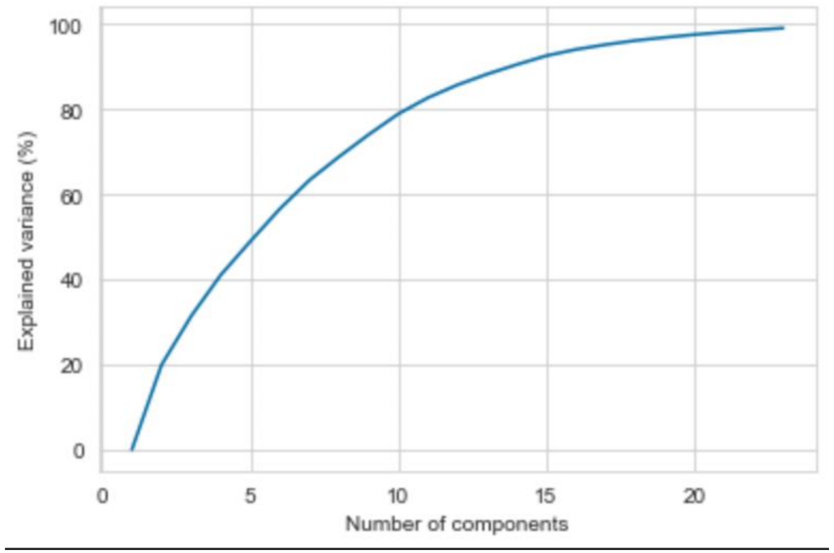
TEXT PROCESSING

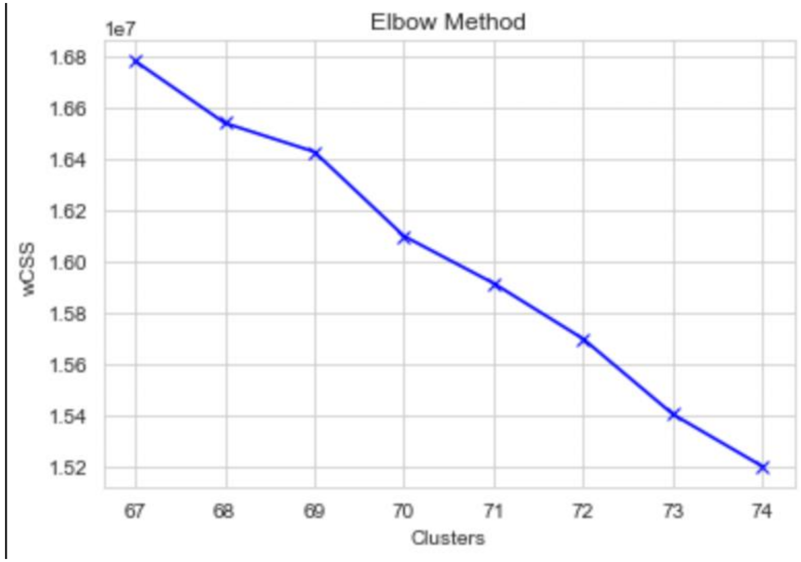
5) Candidate/Name Extraction:

- Maybe tweets about Macron are more appreciated ?
- Maybe news about the Ukraine are interesting to spread?
- Extract using Flair-NER package
- Find similarity to already found names
- Classify or add new names
- Gives back a number as class e.g. Macron = 3.

DIMENSIONALITY REDUCTION

- PCA (explained variance ratio curve)
- KMeans Clustering (Unsupervised Learning, creates categorical feature)





ML

1) Regression Models ++ :

After:

- Data Pre-processing
- Selection of the final predictors
- Data Transformation (Standardization/Normalization for distance-based algorithms)

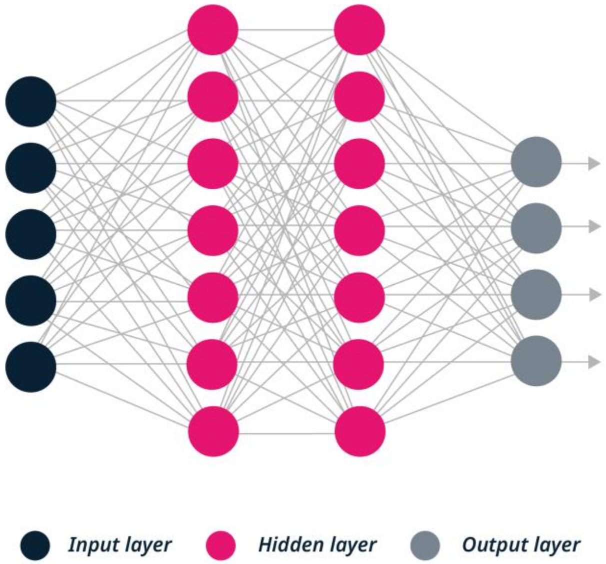
We retested regression models :

- We even tried gradient boosting regressor

Worse performance!!

Probably because : + features needs more data.

ML



2) Deep learning:

- Keras

- Architecture
  - ❖ Infinite possibilities
  - ❖ Sticking to the basics (ReLu)
  - ❖ FCNN

- Optimizing GD using SGD and Adam

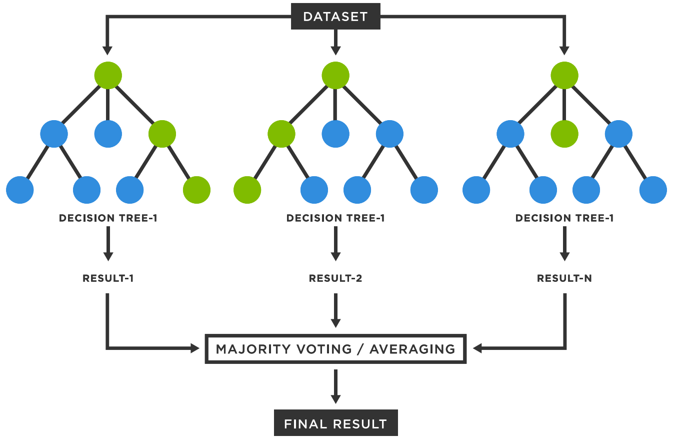
- Extremely good results by Keras on the training set MAE ≈ **1.9**

FINAL MODEL AND DEPLOYMENT

- 1) Predictors:
  - Clusters(tweets similarity after applying KMeans)
  - The number of users that clicked the (like) button for this tweet.
  - The number of followers the user has

- 2) Model: Random Forest
  - Reasons :
    - Categorical Data(Clusters, Sentiment, Candidate,Verified)
    - Aggregates the result of many decision trees and then outputs the most unbiased result (uses averaging).
    - A random forest produces good predictions that can be understood easily.

  - Hyperparameter tuning (Grid Search using Scikit Learn)



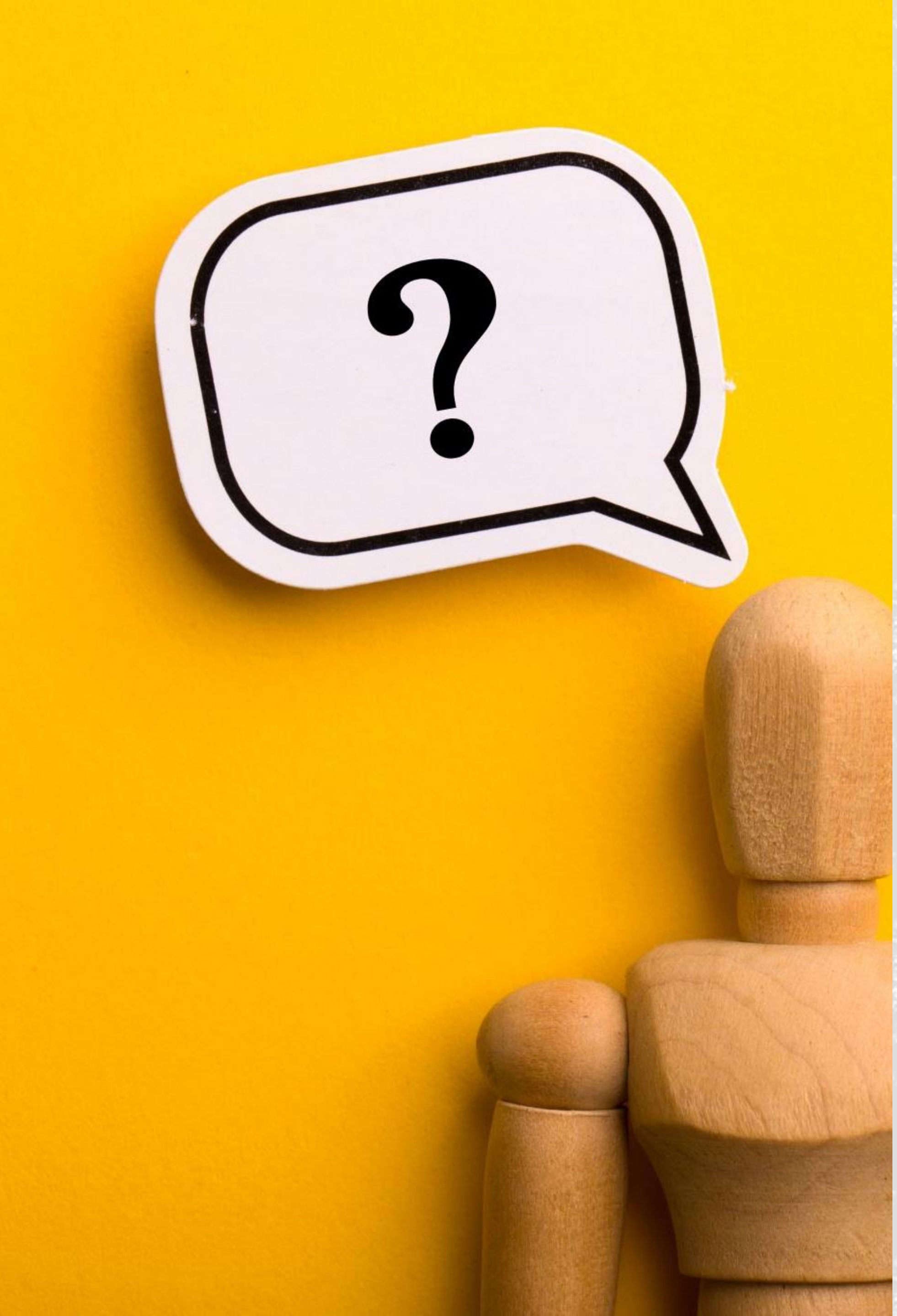
CONCLUSION

Challenges

- Execution Time
  - ○ Text Processing
  - ○ grid search
- Language - NLK

Potential Improvements

- More TFIDF Words
- Smaller Testing Batches
- Using a raw word2vec implementation (untrained)
- Dropout for NN



THANK YOU FOR YOUR ATTENTION