1st Assignment:
Fake News Detection

How to solve it

- **Classification problem**
  - News Report (***document***) → ***Class***: [FAKE, REAL]
- **try text-related classifiers**
  - Naive Bayes
  - MaxEnt
  - SVM
- **NLTK+SKLearn provides you anything you need**
  - NLP Pre-processing
  - Classifiers
  - CV-evaluation

Dataset

- **fake_or_real_news_training:**
  - **ID**: ID of the tweet
  - **Title**: Title of the news report
  - **Text**: Textual content of the news report
  - **Label**: Target Variable [FAKE, REAL]
  - **X1, X2 additional fields**
- **fake_or_real_news_test:**
  - **ID, title and text**
  - **Predict Label**

Advices

- **Take a look to the data**
- Try the **pre-processing methodologies** we have seen **in class**
- **TF-IDF** seems to be better (but try it!)
- **N-grams** pay the effort
- Less than 90-92%? **Try again**

Advices/Warnings

- **Avoid ML mistakes!**
- **Explain anything you do**
- **Try different approaches and compare results**
  - **Classifiers**
  - **NLP Pipelines**
- **Analyze your results**

Submission

- **Due: 27th May**
- **Submission** (Send me **everything** please)**:**
  - **CSV with your predictions**
  - **Tweet_id (ID), prediction[FAKE, REAL]**
  - **Notebook**
- Send me something that **actually works**
- **Grading:** 50% results – 50% notebook

Resources

- **NLTK Book Chapter**
  - http://www.nltk.org/book/ch06.html
- **Examples of NLTK + SkLearn for Text Classification**
  - https://towardsdatascience.com/machine-learning-nlp-text-classification-using-scikit-learn-python-and-nltk-c52b92a7c73a
  - http://billchambers.me/tutorials/2015/01/14/python-nlp-cheatsheet-nltk-scikit-learn.html
- **Resources in the class slides**