**Due Date:** Sunday, May 30th by 11:55pm.

Assignments are to be done individually. No late assignments will be accepted. Submit your assignment on the day that it is due.
**Submissions that do not comply with the specifications given in this document will not be marked.**
**Note: If your implementation matches with your classmates or with any code available on Kaggle or Github, a zero grade will be assigned.**
Write your name and e-mail id in a comment line in on top of each source file. You are required to submit a single zip file containing all code files (.py and .ipynb) an archive of your documentation and ipython notebook on Google Classroom.

# Fake News Detection

The dissemination of Fake news always beat out the truth with significant growth. Fake news and false rumors are spreading further and faster, reaching more people, and penetrating deeper into social networks. The objective of this assignment is to address the problem of detecting deceiving information in Urdu language from digital media text.

Your goal in this assignment is to perform Fake News Detection: classifying news articles as fake or real. Recall from the lecture on sentiment analysis that it can be used to extract people's opinions about all sorts of things (tweets, speeches, reviews, blogs) and at many levels of granularity (the sentence, the paragraph, the entire document). Our goal in this task is to look at a news story and classify it as $fake$ or $real$.

You will be using Naïve Bayes, following the pseudocode given in Algorithm 1 and Algorithm 2 using Laplace (add-1) smoothing. Your classifier will use words as features, add the $log-prob$ scores for each token, and make a binary decision between $fake$ and $real$.

---

**Algorithm 1** TRAINMULTINOMIALNB$(\mathcal{C}, \mathcal{T})$

---

1: **procedure** NAIVEBAYESTRAINING$(C, T)$
2:    $V \leftarrow$ EXTRACTVOCABULARY(C)
3:    $N \leftarrow$ COUNTTEXTS(T)
4:    **for each** $c \in \mathcal{C}$ **do**
5:       $N_c \leftarrow$ COUNTTEXTSINCLASS$(\mathcal{T}, c)$
6:       $N_w \leftarrow$ COUNTWORDSINALLTEXTSOFCLASS$(\mathcal{T}, c)$
7:       $prior[c] \leftarrow \frac{N_c}{N}$
8:       $doc_c \leftarrow$ CONCATENATETEXTSINCLASS$(\mathcal{T}, c)$
9:       **for each** $w_i \in \mathcal{V}$ **do**
10:          $N_i \leftarrow$ COUNTTOKENSOFWORDS$(doc_c, w_i)$
11:       **for each** $w_i \in \mathcal{V}$ **do**
12:          $condprob[w_i][c] \leftarrow \frac{N_i+1}{N_w+|V|}$
13:    **return** $V, prior, condprob$                          ▷ THE NB MODEL

---

You will also explore:

(1) The effects of stop-word filtering. This means removing common words from your train and test sets. We have provided a stopword list in the file:
`stopwords-ur.txt`

(2) The effects of Boolean Naïve Bayes. This means removing duplicate words in each document (review) before training.

**Algorithm 2** APPLYMULTINOMIALNB($\mathcal{C}, \mathcal{V}, prior, condprob, t$)

---

1: **procedure** NAIVEBAYESTEST($\mathcal{C}, \mathcal{V}, prior, condprob, t$)
2:      $\mathcal{W} \leftarrow$ EXTRACTWORDSFROMTEXT($\mathcal{V}, t$)
3:      **for each** $c \in \mathcal{C}$ **do**
4:          $score[c] \leftarrow \mathbf{log}\ prior[c]$
5:          **for each** $w_i \in \mathcal{W}$ **do**
6:              $score[c] \mathrel{+}= \mathbf{log}\ condprob[w_i][c]$
7:      **return** $argmax_{c \in \mathcal{C}} score[c]$                                      ▷ THE PREDICTED CLASS

---

# 1  Assignment Tasks

Train a Naïve Bayes classifier on the data set provided.

Your task is to implement the classifier training and testing code and evaluate them using Accuracy, Precision, Recall and $F_1$ measures. Next, evaluate your model again with the stop words removed. Show how this approach affects average accuracy (for the current given data set).

# 2  Requirements

You are required to implement:

(1) Naïve Bayes classifier.

(2) Boolean Naïve Bayes classifier.

(3) Experiment with/without using stop words.

(4) Metrics for evaluating performance of your classifier.

You are required to submit a single page report within the python notebook that analyzes your runs with/without stop word filtering, and mention if the Boolean Naive Bayes improves your performance or not.

# 3  Extra Credit

There are sometimes news that use words with negation. Preprocess the negations in the dataset and see if it improves your classification.

**Honor Policy**
This assignment is a learning opportunity that will be evaluated based on your ability to think in a group setting, work through a problem in a logical manner and write a research report on your own. You may however discuss verbally or via email the assignment with your classmates or the course instructor, but you are to write the actual report for this assignment without copying or plagiarizing the work of others. You may use the Internet to do your research, but the written work should be your own. **Plagiarized assignments will get a zero.** If in doubt, ask the course instructor.