The goal of this project is to predict sentiments in twitter data using logistic regression, closely following the approach outlined in Chapter 5. The input data can be found at <https://github.com/cardiffnlp/tweeteval/tree/main/datasets/sentiment>. The dataset is already split into training, validation and test. For each case the input and the labels are in separate txt files.

In order to use logistic regression, we need to come up with a set of features. One approach suggested in the book is to use positive and negative lexicons. You can find such lexicons at <https://nlp.stanford.edu/projects/socialsent/>. For your convenience I have already downloaded them and have included along with this document.

The file socialsent\_hist\_adj.zip has historical adjectives for each decade from 1850 to 2000 in the form of tab separated values. For example, 2000.tsv has a list of 5000 negative and positive lexicons. Each lexicon has an individual score. Use 2000.tsv.

The file socialsent\_hist\_freq.zip has historically frequently used words for each decade from 1850 to 2000 in the form of tab separated values. For example, 2000.tsv has a list of 5000 negative and positive lexicons. Each lexicon has an individual score. Use 2000.tsv.

The file socialsent\_subreddits.zip has more interesting data. It has a huge collection of subreddit groups and for each group it has a list of up to 5000 positive and up to 5000 negative lexicons. Instead of considering all the tsv files, let’s consider the following (the first file for each alphabet and all ones starting with a number).

3DS.tsv

4chan.tsv

2007scape.tsv

ACTrade.tsv

amiugly.tsv

BabyBumps.tsv

baseball.tsv

canada.tsv

CasualConversation.tsv

DarknetMarkets.tsv

darksouls.tsv

elderscrollsonline.tsv

Eve.tsv

Fallout.tsv

fantasyfootball.tsv

GameDeals.tsv

gamegrumps.tsv

halo.tsv

Homebrewing.tsv

IAmA.tsv

india.tsv

jailbreak.tsv

Jokes.tsv

KerbalSpaceProgram.tsv

Keto.tsv

leagueoflegends.tsv

Libertarian.tsv

magicTCG.tsv

MakeupAddiction.tsv

Naruto.tsv

nba.tsv

occulus.tsv

OkCupid.tsv

Parenting.tsv

pathofexile.tsv

raisedbynarcissists.tsv

Random\_Acts\_Of\_Amazon.tsv

science.tsv

Seattle.tsv

TalesFromRetail.tsv

talesfromtechsupport.tsv

ultrahardcore.tsv

videos.tsv

Warthunder.tsv

whowouldwin.tsv

xboxone.tsv

yugioh.tsv

Using these file design 9 features. Let the 10th feature be the log of the word count of the tweet. The 11th feature be the log of the length of the longest word. The 12th feature be the log of the count of words with 5 or more characters.

So, for each tweet you would have 12 features.

Create a Jupyter notebook and read all the tweets and lexicon files. Use Pandas to organize the data. Use either numpy or PyTorch to create the input and output datasets for training and testing.

Implement logistic regression using the algorithm provided in the book. Train your model with the training dataset once. The apply the model to the test data set. You will implement cross validation later. Calculate F1 score and accuracy and submit your ipynb file.