Dataset (nltk movie_reviews):

- Number of movie reviews - 2000
- train : test split ratio - 80 : 20
- Number of sentences - ~ 65k

Vanilla Sentiment Analyzer :

- Inputs movie reviews, which are long paragraphs spanning multiple lines
- Outputs one of two labels - positive (“pos”) or negative (“neg”)
- Model used : SVM with linear kernel

Classification report :

Test Accuracy: 0.7375

Test classification Report:

	precision	recall	f1-score	support
neg	0.75	0.75	0.75	212
pos	0.72	0.72	0.72	188
accuracy			0.74	400
macro avg	0.74	0.74	0.74	400
weighted avg	0.74	0.74	0.74	400

Proposed Hypothesis :

If movie reviews are filtered out to keep only important word types {ADJECTIVE, NOUN, VERB, ADVERB}, the resulting model trained for sentiment analysis should perform better than the vanilla sentiment analyzer.

Steps :

- Implement a POS tagger using Viterbi algorithm
- Use POS tagger to tag words in all sentences of movie_reviews dataset
- Implement a processing pipeline that only keeps words having tags from {ADJECTIVE, NOUN, VERB, ADVERB} types, filtering out the remaining words
- Train a classification model like SVM on the filtered movie reviews

Instructions to run code :

Apart from common libraries like scipy, numpy etc, also install ‘tqdm’ using :

pip install tqdm

This library allows showing progress bars during execution.

Classification report :

Test Accuracy: 0.7045

Test classification Report:

	precision	recall	f1-score	support
neg	0.72	0.72	0.72	206
pos	0.68	0.68	0.68	194
accuracy			0.70	400
macro avg	0.70	0.70	0.70	400
weighted avg	0.70	0.70	0.70	400