Components:

* Materials:
  + Movie dataset
* Functions:
  + Class NLP\_ natural language processing machine learning algorithm
    - preprocessing() imports, labels and splits data
      * returns numpy arrays of training data, test data, training data labels, test data label
    - label\_raw\_dat() imports input data and adds data
    - vectorizer() splits data
      * returns list of words
    - unique\_words finds all the unique words w/in input data that satisfy the given constraints of lower bound word frequency and upper bound word frequency
    - coder() maps given unique word to randomly assigned index position w/in unigram (bag of words) sparse matrix
      * Returns dictionary where key is unique word and value is index
    - elem\_freq() converts input data into unigram form due to input identified unique words given by unique\_words list
      * Return bag of words unigram sparse matrix
    - elem\_unigram() runs unique\_words(), code(), elem\_freq() methods if needed depending on if it’s the final training step, or test data conversion step which is indicated by if\_FinalTrain and if\_TestSet
      * Returns the unigram sparse matrix for given input data set
    - gridSearch() implements grid search – iterates through the possible hyperparameters to determine which hyperparameter values lead to optimal classifier performance
      * Returns and globalizes the best/optimal hyperparameter values
    - nFoldCrossValidation() iterates over the k folds and performs hold out method for the kth fold and assigns the kth fold as the development set. Runs LinearSVC classifier on all but kth fold, and the model is then tested and assessed by the development set’s predictive accuracy.
      * Returns average prediction accuracy score
    - trainModel\_best() performs the last and final training for the model and thus finalizing the ML model
    - predictionTest() performs the prediction and accuracy for the test set and its label
      * Return model’s accuracy on prediction set
    - run\_model() runs and executes all the methods required for the model training and test, executes preprocessing(), gridSearch(), trainModel\_best(), and predictionTest()

Results:

The MLA with LinearSVC classifier returns average prediction accuracy rate for final model: 0.7452250824210601 and with accuracy score for the test set is 0.7252695733708392 given parameters: c: 0.01 k: 10 n: 0 m: 4000. The MLA took considerable time and memory due to the gridsearch and unigram conversion therefore has a large space and time complexity. I think the use of SkLearn implementations seemed to be faster than my own implementation which will be kept in mind with larger data sets.