FULL NAME:**ZERVAS MICHAEL**
REGISTRATION NUMBER:**ics20015**

Programming Project 2

The entire dataset was used for the implementation of the work.

Preprocessing (train and test set):

Preprocessing removed lines starting with specific words from the heading (such as “From”), removed words that do not contribute to the content (such as “Subject”), and removed numbers. Python's Natural Language Toolkit (NLTK) library was used to remove stop words, punctuation and stemming.
Then each word of each document was weighted with tf-idf and based on this the 15 most important words of each document were kept.

Correlation rules (train set):

Here, using the mlxtend library, apriori was used to find the frequent sets and extract from them, the association rules. Various support and confidence values were tested. It was observed that the smallest value the support could take without filling the ram was 0.003. The value finally used was support = 0.0031 and 816 frequent datasets were produced. Confidence did not affect the results for small changes in its price to the same extent as support. For value 0.15 and 0.2 it produced 98 correlation rules. In the end I kept the value 0.2.

Text categorization (train set):

Frequent sets and association rules were recomputed, except that the category it belongs to was also added to the words representing each document. The association rules were filtered so that only some of the text categories were contained in the right part of the rule. So now the rules became word1, word2
→ category1. With the same support and confidence values, a total of 1009 frequent sets and 168 association rules of the format we want were used in the valuation.

Valuation:

The implementation of the prediction function was based on the category to which most (out of 15) words of each document belong. Specifically, a check is made every time if each word of the document belongs to the right part of any association rule and at the end we keep the category that represents the most words. Because there were several cases where the words were not on the right-hand side of a rule (in the code and in the runtimes they are represented by the x variable), then the document is not categorized. For the evaluation, 5 dictionaries were created: one consists of the number of documents from the test set belonging to each category, the second consists of the number of predictions for each category and the other 3 concern True Positives, False Positives and False Negatives . The results of the metrics with macro averaging are: Precision = 0.6701 and Recall = 0.3709. The category with the best Precision and Recall was sci.electronics:
Precision of category: 0.9583
Recall of category: 0.05852