

**1. Project overview:**

The growth of mobile phone users has led to a dramatic increase in SMS spam messages. Therefore, there is a need for reliable filtering techniques capable of detecting spam messages from ham (legitimate) messages. The objective of the current capstone project is to develop an automatic filtering system using natural language processing (NLP) and machine learning techniques for SMS spam detection. Potential clients for an SMS spam detection system are mobile phone service providers. Mobile service providers can use this system to automatically filter out spam messages at the first place so that their clients do not receive those spam messages which potentially leads to more customer satisfaction. Using this filtering system, the mobile service providers can trace the sources of spam messages and implement strategies to eliminate/reduce the rate of spam messages which are generated/received. To achieve the goal of this project we take the following steps:

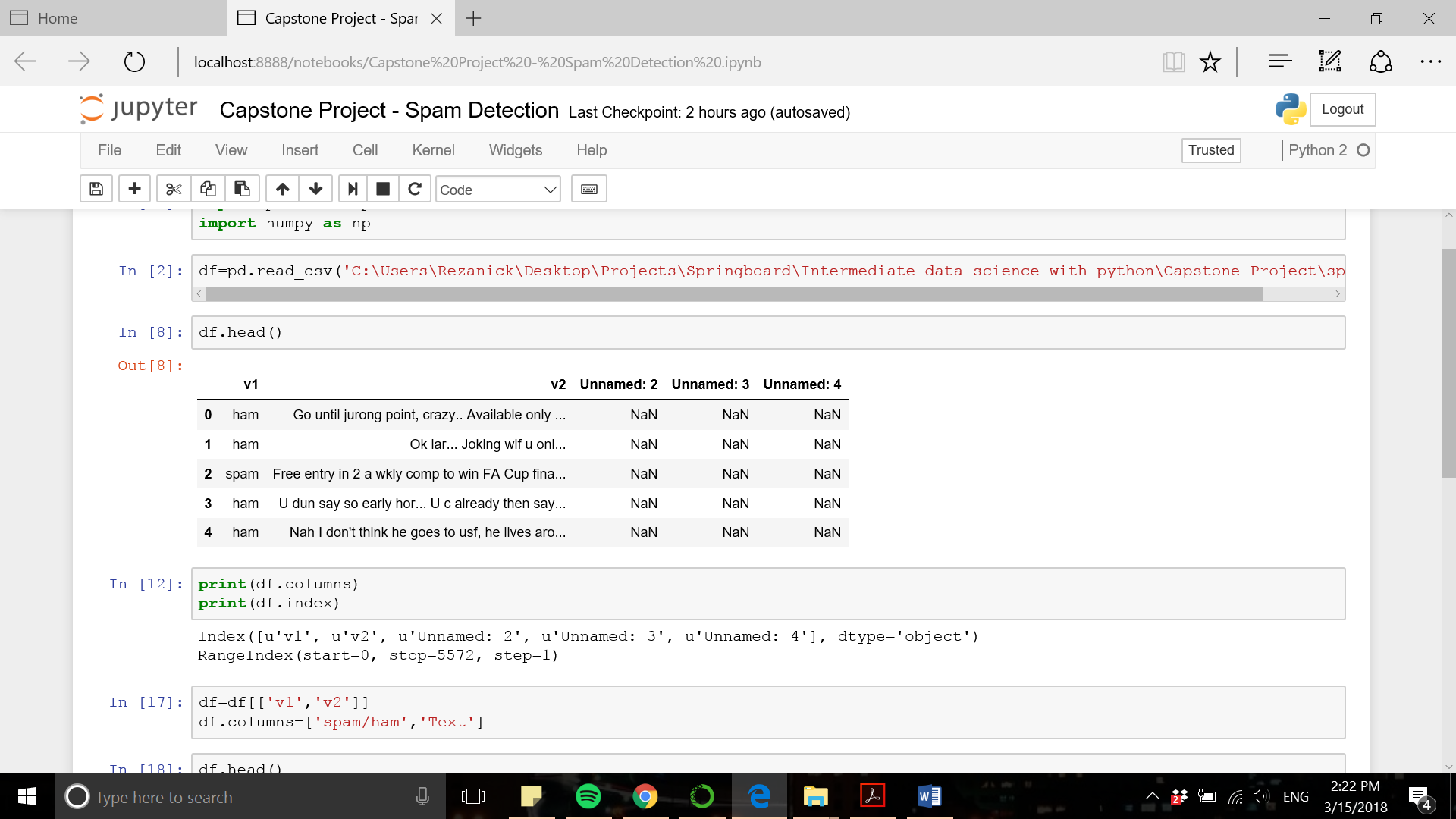
* Data exploration and cleaning
* Vectorization of text messages
* Training machine learning classifiers
* Model evaluation

**2. SMS spam collection dataset:**

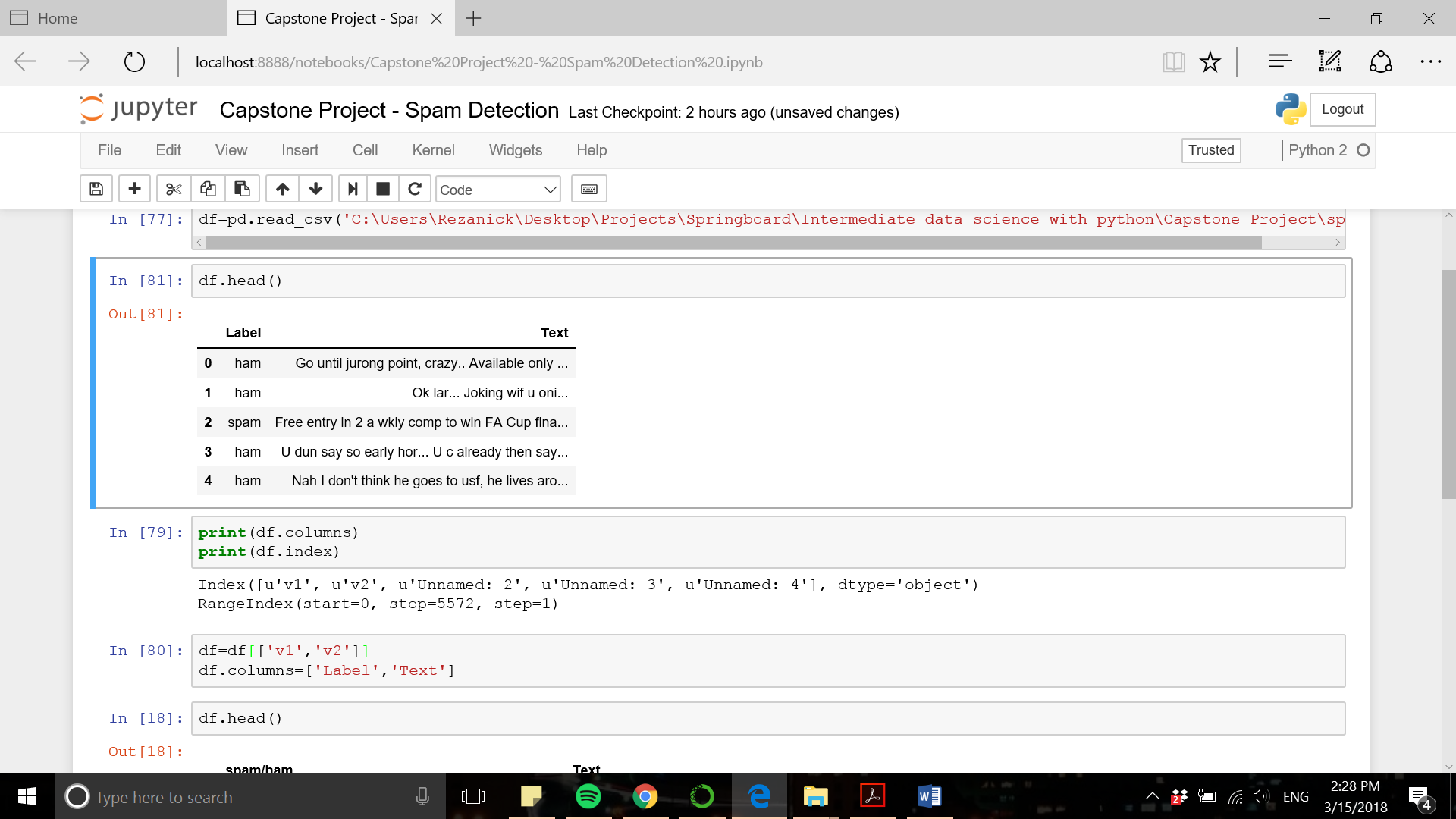
The dataset for this project originates from the UCI Machine Learning Repository which is publicly available. The SMS Spam Collection is a set of SMS tagged messages that have been collected for SMS Spam research. It contains one set of SMS messages in English of 5,574 messages, tagged according being ham (legitimate) or spam.

**3. Data wrangling**

SMS spam detection dataset consisted of 5572 entries (4825 ham messages and 747 spam messages). As shown below, the column ‘v1’ represented the tag for each message labeled as ‘ham’ or ‘spam’. The column ‘v2’ consisted of text messages in string format. There are three redundant columns (‘Unnamed: 2’, ‘Unnamed: 3’, ‘Unnamed: 4’) with NaN values.

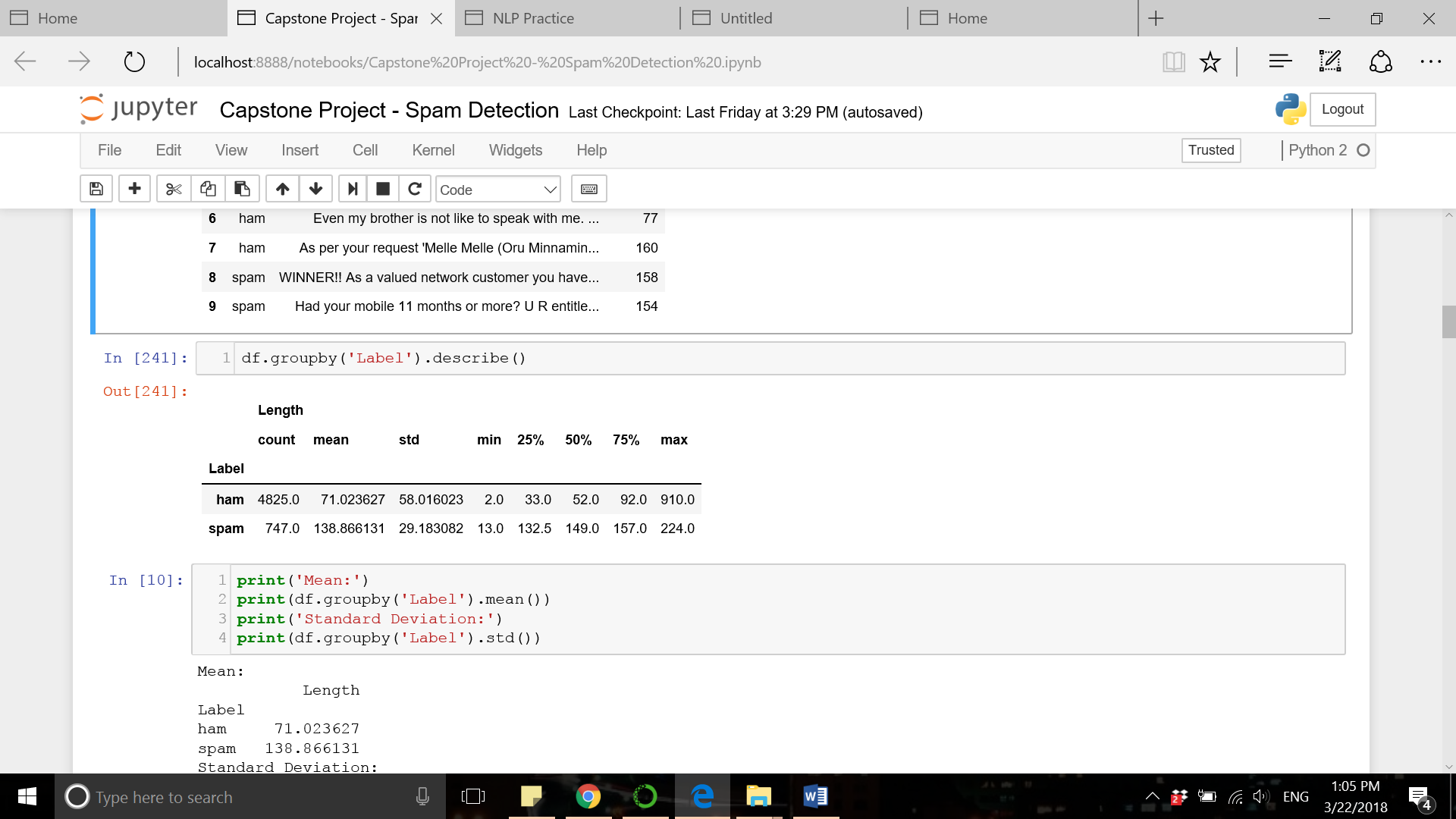


The first two columns (‘v1’ and ‘v2) are renamed into more meaningful column names (‘Label’ and ‘Text’, respectively) and the three redundant columns are dropped. The cleaned data frame is shown below and is ready for exploratory data analysis.

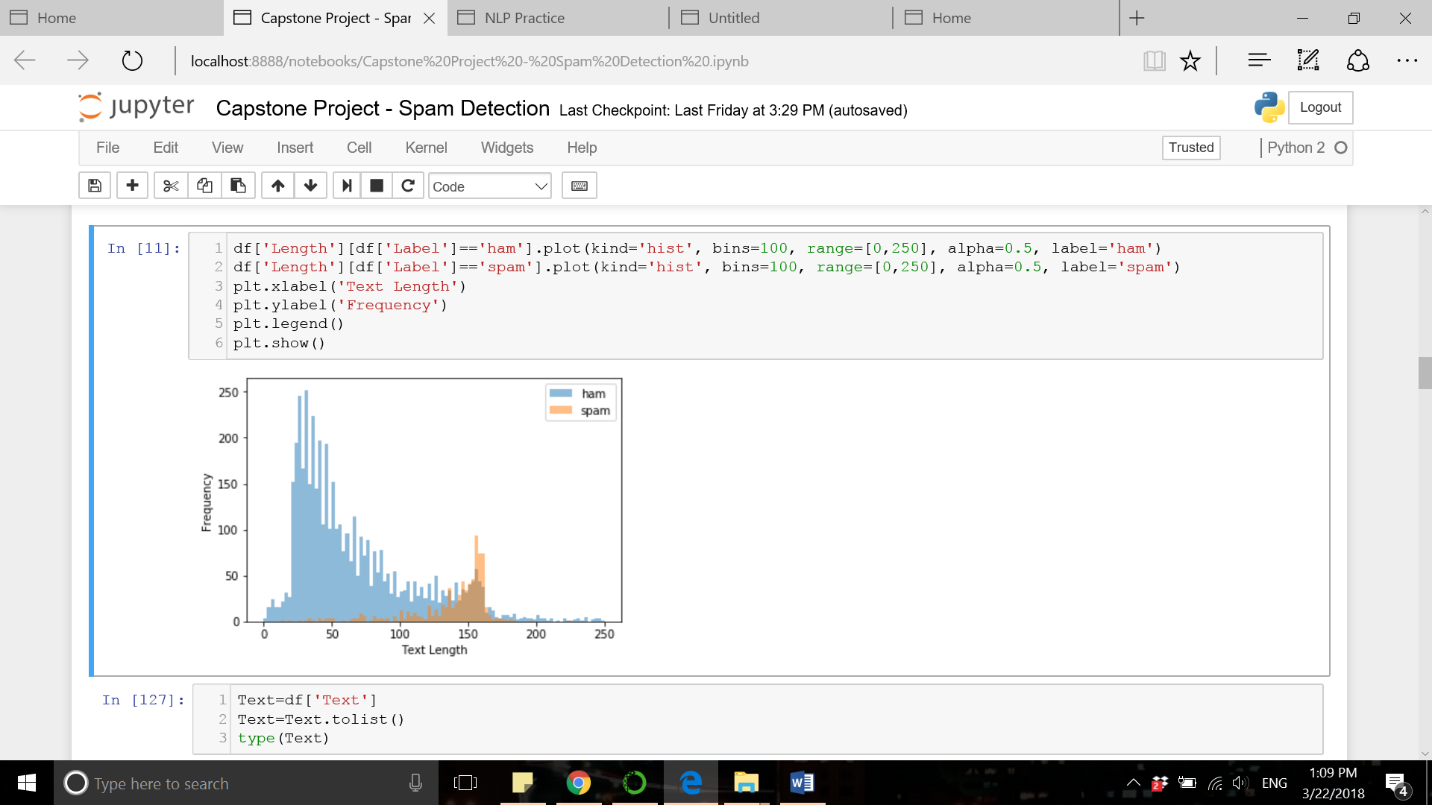


**4. Data exploration and visualization**

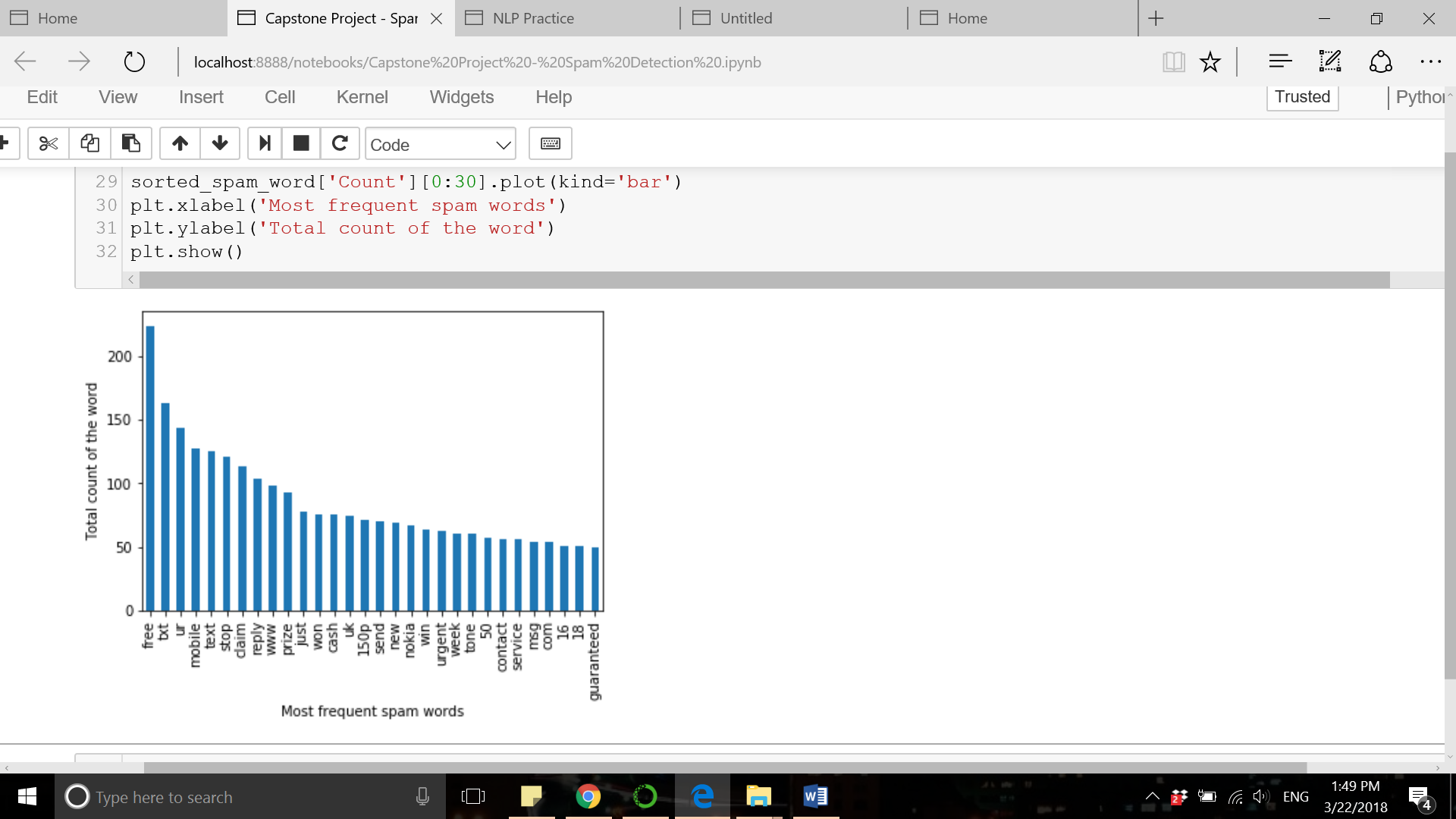
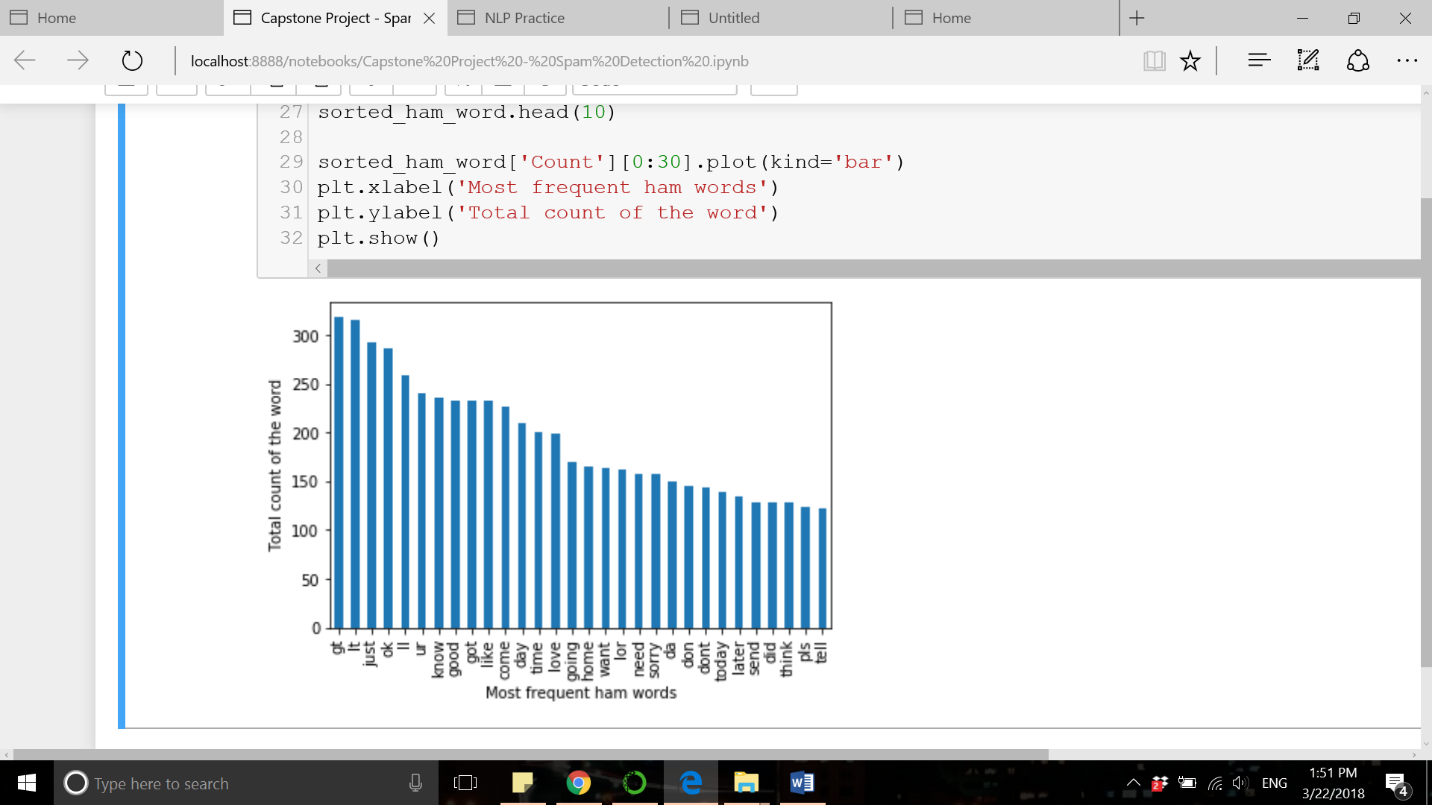
The question associated with this capstone project is which features in our data set could be used to detect spam messages from ham messages. The statistics about the length of each message clearly show that spam messages are longer than ham messages:



We visualize the distribution of message length by creating a histogram:



In addition to that, the word frequency data can provide further insights into the spam/ham messages and the features useful for detection purposes. The two charts below show the most frequent words in spam/ham messages and the total number of their occurrence at each group. It is evident that specific words are frequently found in spam messages which are used in ham messages less often. The most frequent word in spam messages is “free” which occurred 224 times while this word occurs only 60 times in ham messages (on average 1 out of 80 ham messages includes the word “free”, while 1 out of 3 spam messages includes this word). Therefore, vectorized representation of text messages created based on the vocabulary obtained from the existing corpus can provide the features for spam detection.



The bar charts below show the most frequent words in each group. Words like ‘free’, ‘claim’, ‘prize’, ‘reply’, ‘won’, ‘cash’, etc are frequent in spam messages and represent a common theme among spam messages.

**5. Vectorization of Text Messages using Tf-idf Representation**

Term-frequency inverse-document-frequency (in short: tf-idf) vectorization method is used to find the numerical representation of each text. The words in the corpus are ranked based on the frequency they appear in the document and in an inverse proportion to

Tf-idf, short for term frequency-inverse document frequency, is the numerical representation of text data which intends to reflect the importance of each word to a document in a corpus. The tf-idf value increases proportionally to the number of times a word appears in a document and is offset by the frequency of the word in the corpus, which helps to adjust for the fact that some words appear more frequently in general (useful to remove stop words).

The table below shows the vectorization of text messages. Each row represents a text message and the column names correspond to selected words.

