# H1 – Extracting Sentiment and Emotion features

**Introduction**

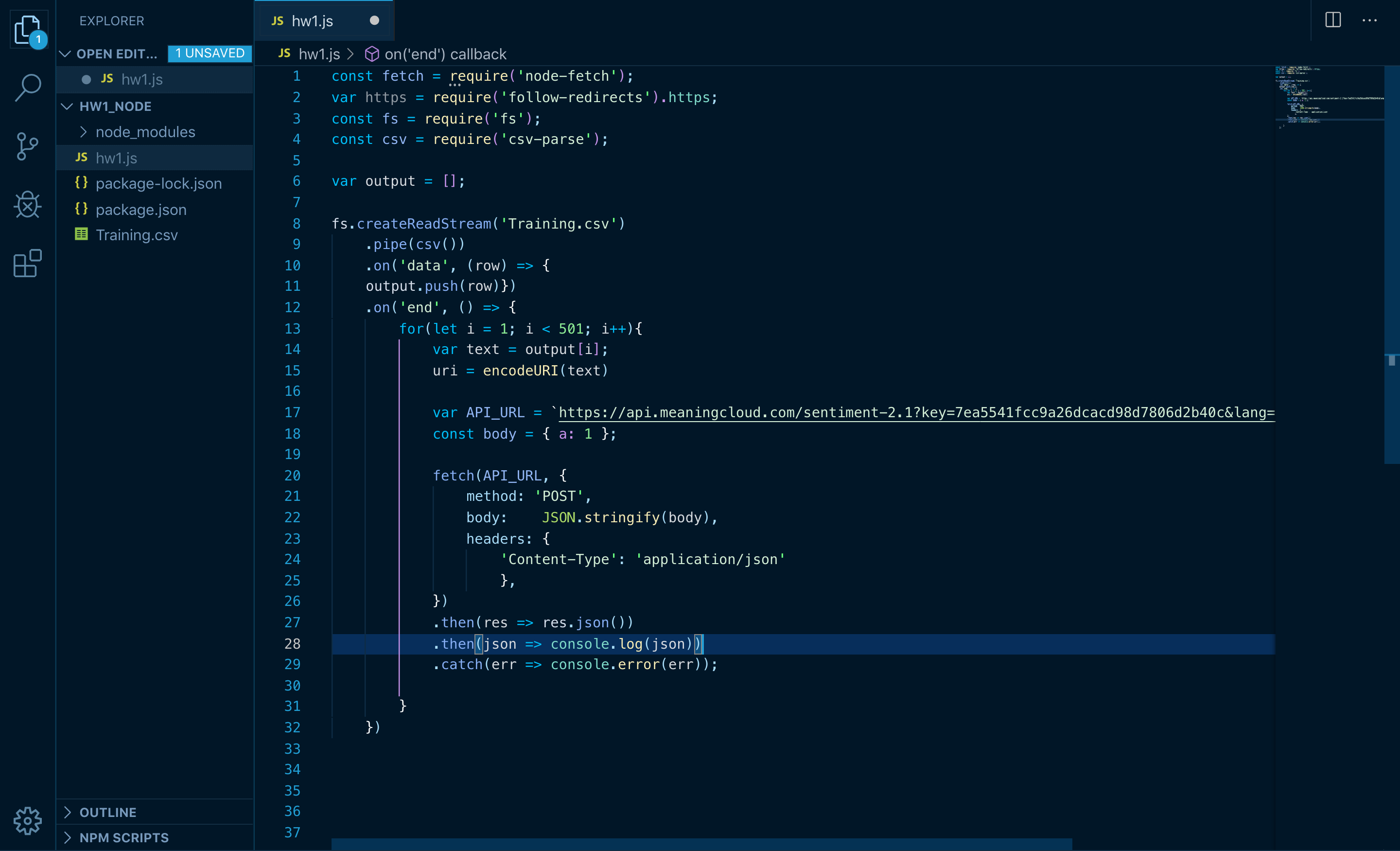
Creating machine learning algorithms for analyzing sentiments and emotions can be quite a challenge. Fortunately, there are organizations who have already created such algorithms, so you don’t have to create everything from scratch. In this homework, we were given a training dataset with three original features and required to add 10 new features (5 regarding to sentiment analysis and 5 to emotion extraction) by using the frameworks created by [meanincloud.com](http://meanincloud.com/) and [paralleldots.com](http://paralleldots.com/).

**Development**

At the beginning of this homework, I decided to work with NodeJS, but several problems were presented because the fetch API function couldn’t parse correctly the spaces from the text’s pararaphs, hence, I switched to rewrite my program with Python.

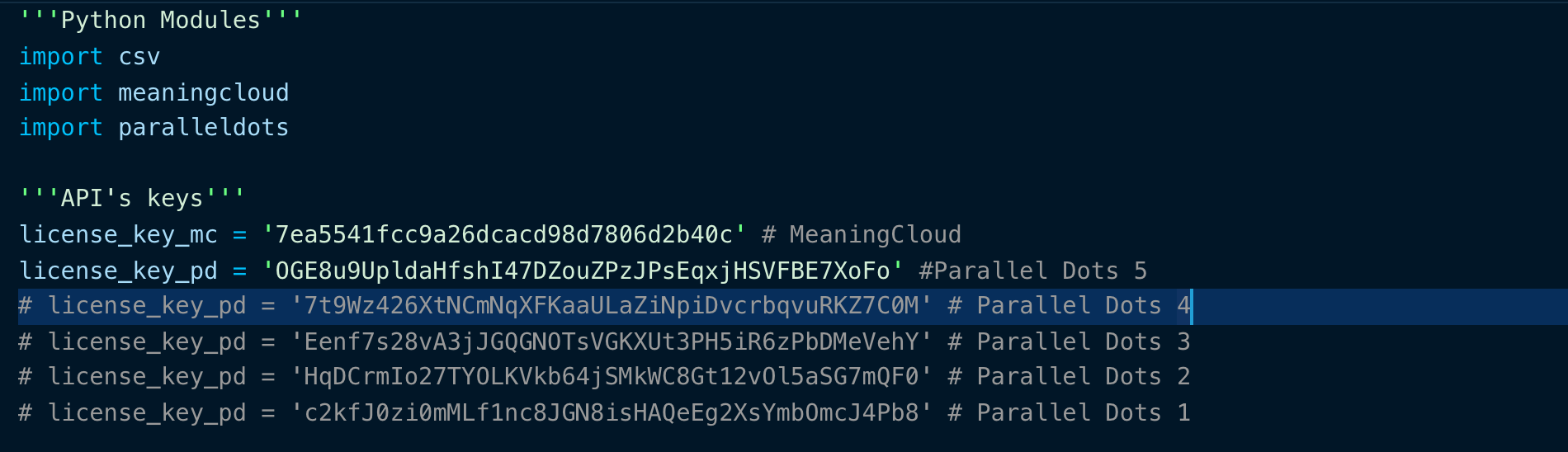


*Figure 1. Responses from the Meaning Cloud’s API made with the function fetch of NodeJS.*

**

*Figure 2. Code snippet for getting the text from the Training.csv and calling the Meaning Cloud’s API written in Javascript.*

In python, it wasn’t necessary to manually call both APIs because both Meaning Cloud and Parallel Dots provided SDKs to access their frameworks. The CSV module was pretty relevant because it was required to read and write CSV files such as the Training.csv. Five Parallel Dots accounts were created due to the limitations of the quantity of API calls that could be made with the free versions of both frameworks and also because of time limitations.



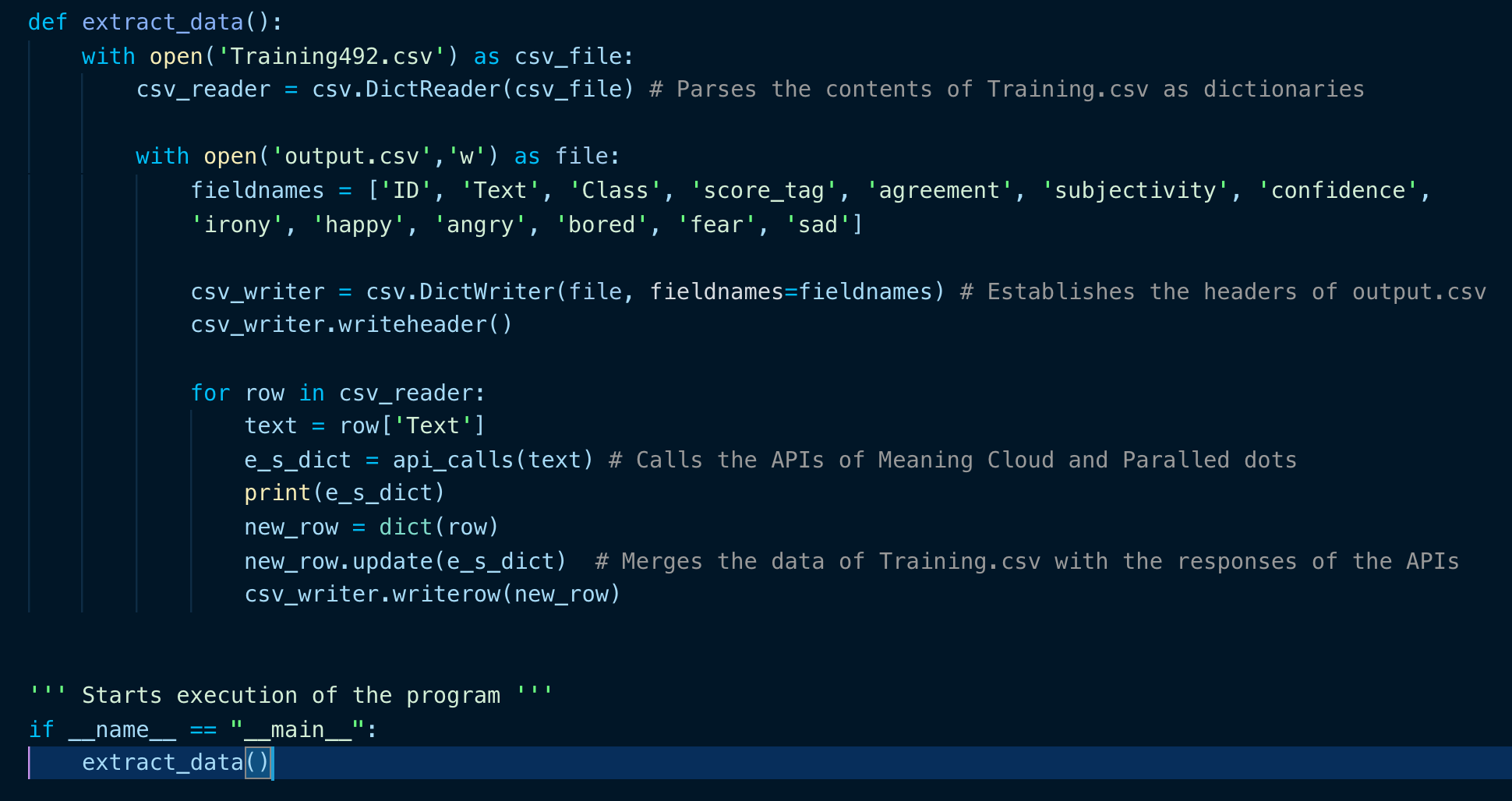
*Figure 3. Python Modules and API’s keys required for the execution of the program.*

Also because of the limitations of the free versions of the APIs, the original Training.csv file was divided into 4 smaller CSV files, which are: Training114, Training234, Training372, Training492. The first time the program was executed, the first Parallel Dots license key and the original Training.csv were used, program execution finished after the response to the 113th row’s data was received because the quantity of free trials of Parallel Dots’ API was reached. So then, The CSV file to open was changed to Training114.CSV, which started with the row 114 of the original file, and the Parallel Dots’ license key was changed, and so on with the following Training CSV files.



*Figure 4. The api\_calls function received the text paragraph extracted from the current CSV file and returned the merge of both APIs responses.*

A new CSV file named output was created each time the responses of a Training CSV file were received. The output.csv file has the following headers: ID, Text, Class, score\_tag, agreement, subjectivity, confidence, irony, happy, angry, bored, fear and sad. After all the rows from a Training CSV file were written into the output.csv file including the new extracted features, the name of output.csv was renamed with the following format: “output[first row]-[last row]”.



*Figure 5. The extract\_data function extracted the data of a Training CSV file, called the extract\_data function with the extracted data and wrote it into a new Output CSV file.*

**Results**

After all the output CSV files were obtained, all the features regarding to sentiment analysis and emotion extraction were manually copied and pasted into the original Training.csv using Excel. Therefore, the original objective of adding 10 new features regarding to sentiment analysis and emotion extraction was accomplished.

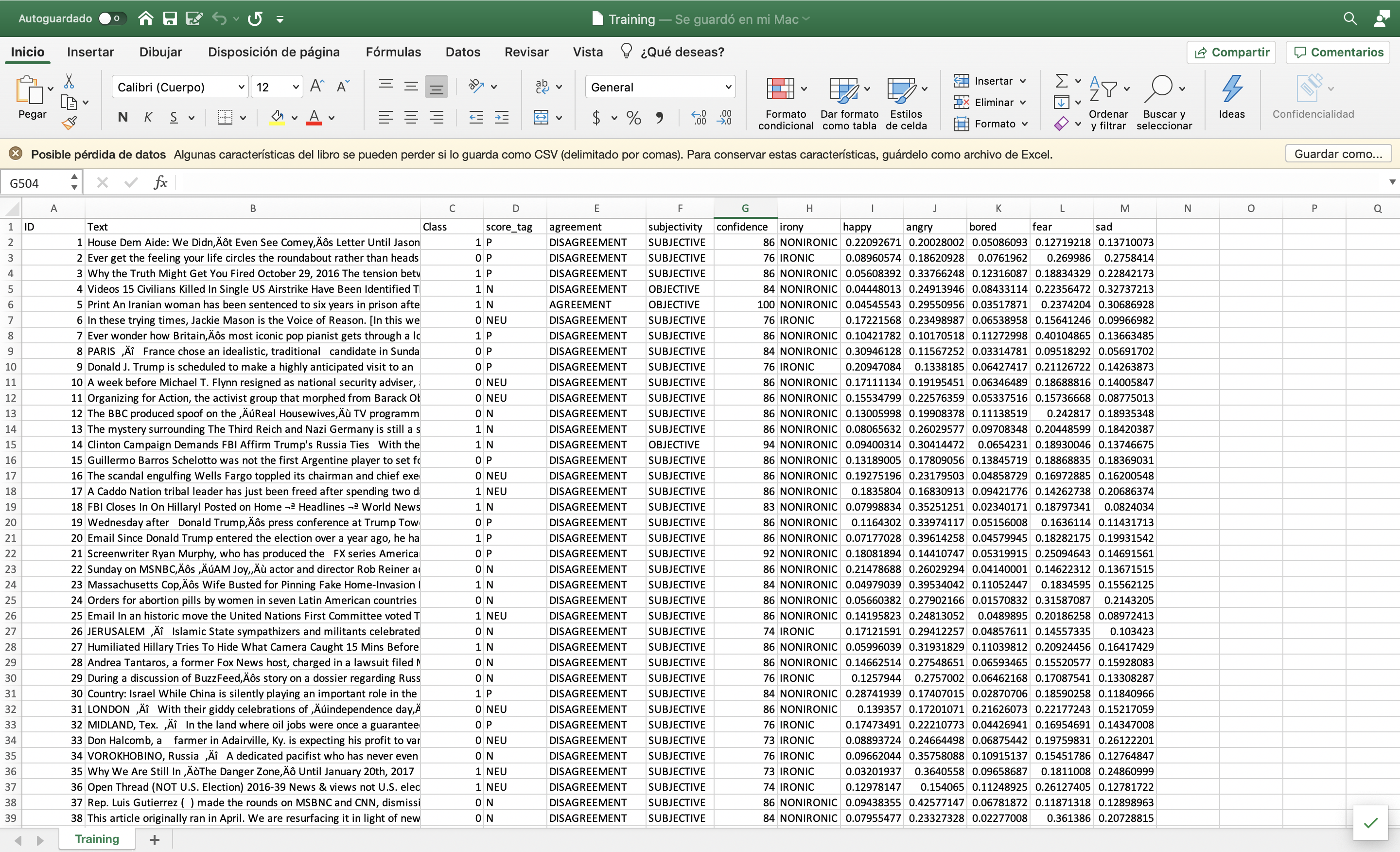


Figure 6. Original Trainig.csv but with the 10 new features added.

**Conclusion**

It would have been easier and less time-consuming if from the beginning I would have chosen Python as the programming language for this homework, considering that it is the *state-of-the-art* programming language for Data Analysis, instead of NodeJS.