**Cleaning Dirty Data with Python & CountVectorizer**

Maya Mileva

HW 3 IST 736

A picture containing table, sitting, wooden

Description automatically generated

**Introduction**

Pandas is a popular Python library used for data science and analysis. Used in conjunction with other data science toolsets like SciPy, NumPy, and Matplotlib, a modeler can create end-to-end analytic workflows to solve business problems.

While you can do a lot of really powerful things with Python and data analysis, your analysis is only ever as good as your dataset. And many datasets have missing, malformed, or erroneous data. It is often unavoidable–anything from incomplete reporting to technical glitches can cause “dirty” data. Thankfully, Pandas provides a robust library of functions to help you clean up, sort through, and make sense of your datasets, no matter what state they are in. One of them is CountVectorizer.

Scikit-learns CountVectorizer is used to transform a corpus of text to a vector of term / token counts. It also provides the capability to preprocess your text data prior to generating the vector representation making it a highly flexible feature representation module for text.

**Analysis and Models**

**About the data**

For this project, Python and CountVectorizer were used to convert two different datasets into nice, clean, and labeled data frames.

The topic of the fist dataset (**Dataset 1**) is Restaurant and restaurant reviews and can be found here: <https://drive.google.com/file/d/11H6AbWxKsPLY3yt__OrmK0rjjYShKhig/view?usp=sharing>

The second dataset (**Dataset 2**) is about Movie reviews and can be found here: <https://drive.google.com/file/d/17nGHPsk4RXfvRoq-ndizTzc0_0PkzAqm/view?usp=sharing>

Both of these datasets are csv files and they are both reviews and dirty data. However, the first one is easier to clean and prepare. The second one is more of a challenge. The final goal is to prepare and format the datasets into dataframe, the first column in each dataframe to be called “LABEL” and to be the label got the data in that row.

The first one has a sentiment label - positive or negative and the review is in separate cells. A screenshot of a cell phone

Description automatically generated

The sentiment label is at the first column, which makes the data cleaning easier. The problem in this data set is that the columns need to be conjoint together.

The second one is a little bit different.

A screenshot of a cell phone

Description automatically generated

A screenshot of a social media post

Description automatically generatedIt is obvious that is a csv file and is made up from text. The text is all in separate column, and there is ‘text’ and ‘reviewclass’, which means that there might be labels too. With further investigation, labels are located at the end, but they are all in different columns.

**Data Cleaning**

There are a lot of different approaches for cleaning dirty data. The one used in this document will take the data and keeps it in text form  while cleaning, ultimately exporting a new, cleaner text file for a fresh import. Some challenges are present:

* Everything in each row of data should be one document
* Collapsing data into one document
* Getting the label from really dirty data

Cleaning steps:

1. Read in the file
2. Prepare new folder
3. Clean the data

* for each row in the data, clean the row
* for each word remove the label

4. Export clean data to new clean txt files from each row

5. Re-import the cleaned files

6. Turn cleaned data into a pandas df with CountVectorizer or Tfidf

**Dataset1**

In this dataset the label was in the first column, so was just spilt. After that CountVectorizer and Tfidf were used (Tfidf was added for comparison to CV), and data frames were built out of them. The next step was to clean those dataframes. The challenge here was to name the rows with the labels.

A screenshot of a computer

Description automatically generated

The first column of the dataframes was index, which was easy to make a column and labeled or could just stayed index. The two new data frames ware saved into csv files.

**Dataset2**

Dataset was cleaned following the algorithm above. Data was read into python; new folder was created. After that with loop all the rows were printed and read in together, so all the text was in one place, not in separate columns. The new challenge here was that the label was no longer at the end. After that with another loop, text from each row were cleaned, right comas are removed, so the label can be extracted. As a result, a corpus was created and the names of the files in the corpus are the sentiment labels.

A screenshot of a cell phone

Description automatically generated

From here Count Vectorizer was used to process the corpus into data frame.

A picture containing black, photo, meter, white

Description automatically generated

From here there was more work to be done. This dataframe had to be cleaned – columns that contained numbers or shorter that 3 words were removed. Still that decisions are based on the further analysis goals. The labels were reformed too.

Other approaches to handle this dirty data were performed. The file was read in and turned into panda dataframe. All the rows were merged together, to get the review text. The last characters were removed, so the label can be extracted, and the review text was cleaned.

A white sign with black text

Description automatically generated

Third way was to read the dirty file, prepare new clean file. Clean the data row by row, and word by word. Export the clean data to new clean file and after that read it in/turn into data frame.

A black sign with white text

Description automatically generated

The last two approaches require more work when we have to create predictive models. After the dataframe is created, they can be exported to csv or further processed with CV and tfidf.

**Results**

Two data frames were produces from cleaning **Dataset 1**– one from CountVectorizer and the other from Tfidf. The only difference was the second one had normalized frequencies.

A screen shot of a computer

Description automatically generated

Table 1. CV

A black sign with white text

Description automatically generated

Table 2. Tfidf

Csv file was created for comparison.

A screenshot of a cell phone

Description automatically generated

Table 3. Raw file

A screenshot of a computer

Description automatically generated

Table 5. Cleaned File

The final result from **Dataset-2** were interesting too. Again, CV and Tfidf were used.

A picture containing black, meter, white

Description automatically generatedTable 6. CV

A black sign with white text

Description automatically generated

Table 7. Tfidf

Csv file was created for comparison.

A screenshot of a cell phone

Description automatically generated

Table 8. Raw file

A picture containing filled, full, table, white

Description automatically generated

Table 9. Cleaned file

**Conclusion**

Data are characteristics or information, usually numerical, that are collected through observation, and data is everywhere. The ability to collect it, organize, analyze, and extract important information from it is crucial for a data scientist. Data can be presented in all kind of forms – images, charts, reviews, tweets, and dirty data inside of all of them.

Dirty data is costing companies millions of dollars each year. Errors and omissions in master data in particular are notorious for causing costly business interruptions. It’s helpful to understand the different types of dirty data that are commonly creeping their way into enterprise systems when considering ways to improve your data quality.

<https://medium.com/@neurodatalab/lie-detection-can-technology-uncover-deception-a5e96c498385>