# SENTIMENTAL ANALYSIS OF SOCIAL EXHORTATION BASED ON PRODUCT REVIEWS

- ***TEAM MEMBERS***
- **KEVIN JEAN**,Requirement Gathering
- **ASWIN CHANDER.K**,*Designer*
- **DHANDAPANI.K**,*Developer*
- **MOHAMMED NAJIM.B.A**,*Tester*


***PROJECT GUIDE***

**Dr. K.SRIDHARAN**

**Associate Professor**

# ABSTRACT


- Online shopping is a form of e-commerce which allows consumers to directly buy goods or services from a seller over the Internet using a web browser.
- Online Product reviews are valuable for upcoming buyers in helping them make decisions. To this end, different opinion mining techniques have been proposed, where judging a review sentence's orientation (e.g., positive or negative) is one of their key challenges
- Recently, **Machine learning** has emerged as an effective means for solving sentiment classification problems. A machine learning model intrinsically learns a useful representation automatically without human efforts.
- However, we propose a supervised machine learning framework for product review sentiment classification which employs prevalently available ratings as weak supervision signals.
- To evaluate the proposed framework, we construct a dataset containing **2, 00,000** weakly labeled review sentences and **15000** labeled review sentences from Amazon. Experimental results show the more accuracy compared to previous one

# EXISTING SYSTEM

- Determining a consensus opinion on a product sold online is no longer easy, because assessments have become more and more numerous on the Internet.
- To address this problem, researchers have used various approaches, such as looking for feelings expressed in the documents and exploring the appearance and syntax of reviews.
- Aspect-based evaluation is the most important aspect of opinion mining, and researchers are becoming more interested in product aspect extraction; however, more complex algorithms are needed to address this issue precisely with large data sets.
- This introduces a method to extract and summarize product aspects and corresponding opinions from a large number of product reviews in a specific domain.
- This maximize the accuracy and usefulness of the review summaries by leveraging knowledge about product aspect extraction and providing both an appropriate level of detail and rich representation capabilities. The results show that the proposed system achieves F1-scores of 0.714 for camera reviews and 0.774 for laptop reviews.

# DRAWBACKS OF EXISTING SYSTEM

- 1.Intelligent system have not been exploited.
- 2.Lack of resources of the language.
- 3.Need to address the problem of sentiments on a large scale.
- 4.Low dimensionality of dataset.
- 5.The sentiment from written language need larger dataset.

# LITERATURE SURVEY

(COMPARISION TABLE)

| s.no | PAPER TITLE AUTHOR | YEAR JOURNAL | METHODOLOGY | PROS | CONS |
|---|---|---|---|---|---|
| [1] | A survey on opinion mining and sentiment analysis: Tasks, approaches and applications **K. Ravi and V. Ravi** | Nov. 2015 | electronic Word of Mouth (eWOM) statements expressed on the web are much prevalent in business and service industry to enable customer to share his/her point of view. | leads us to extract, transform, load, and analyze very huge amount of structured and unstructured data, at a fast pace, | intelligent techniques have not been exploited exhaustively like evolutionary computation, association rule mining, fuzzy rule based systems, rule miner |
| [2] | Sentiment analysis algorithms and Apllications: A Survey **W.Medhat , A.Hassan and H.Korashy** | Dec. 2014 | Sentiment Analysis (SA) is an ongoing field of research in text mining field. SA is the computational | fields include Emotion Detection (ED), Building Resources (BR) and Transfer Learning | There is still a lack of resources for the Middle East languages including the Arabic language. |

| | | | | | |
|---|---|---|---|---|---|
| [3] | **Survey on Mining Subjective Data on the Web**<br>**M. Tsytsarau and T. Palpanas** | May.2012 | **Opinion aggregation over product reviews useful for product marketing and positioning, exposing the customers' attitude towards a product and its features such as time, geographical location, and experience.** | overall opinion of the community on some specific product, rather than the individual user opinion on that product | **need to address the problems of aggregating, manag-ing, and analyzing sentiments in a large scale, and in an ad hoc fashion.** |
| [4] | Sentiment Classification: A Combination of PMI, SentiWordNet and Fuzzy Function<br>**A.-D. Vo and C.-Y. Ock** | 2012 | unsupervised method for classifying the polarity of reviews using a combination of PMI, SentiWordNet and adjusting the phrase score in the case of modification | The algorithm achieved an average accuracy of 74.47%, ranging from 69.36% for movie reviews to 80.16% for automotive reviews | relative low dimensionality of a data set built from SentiWordNet data set - less than 100 features compared to several thousand typically seen on word vector. |

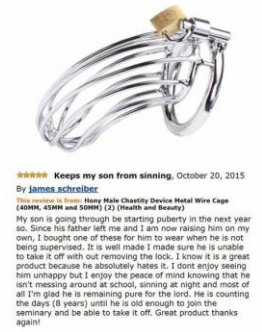
| | | | | | |
|---|---|---|---|---|---|
| [5] | Synthesis Lectures on Human Language Technologies<br>B. Liu | 2012 | sentiment analysis coincides with the growth of social media such as reviews, forum discussions, blogs, micro-blogs, Twitter, and social networks | first time in human history, we now have a huge volume of opinionated data recorded in digital form for analysis | analyzes people's opinions, sentiments, evaluations, attitudes, and emotions from written language need larger dataset |

# PROPOSED SYSTEM

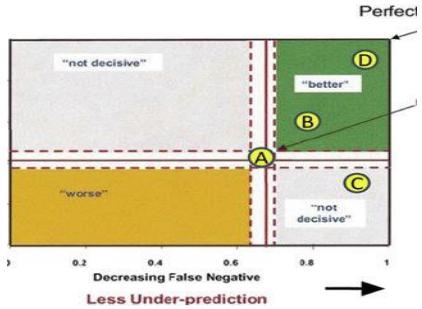
- **Opinion mining** at the document-level is the most widely used method for categorizing a whole-opinion review Sentence-level sentiment analysis focuses on finding subjective sentences. Our work thus addresses the issues of feature-based summaries of product reviews.
- In this paper, we focus on how to extract product aspects using the knowledge gained from reviews. However, before going into the details of the task, we need to define the terminology of our system.
- We propose a supervised machine learning framework for product review sentiment classification which employs prevalently available ratings as weak supervision signals using **linear regression algorithm.**
- To evaluate the proposed framework, we construct a dataset containing **2, 00,000** weakly labeled review sentences and **15000** labeled review sentences from Amazon. Experimental results show the more accuracy compared to previous one.

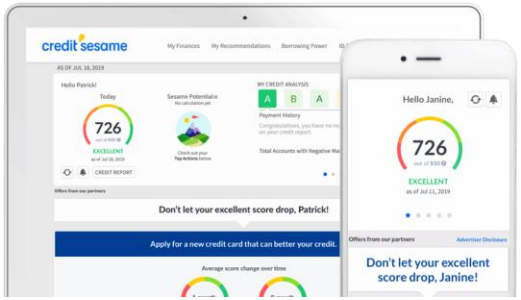
# ARCHITECTURE

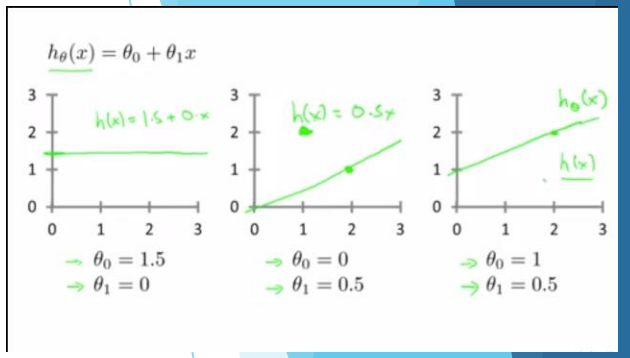
Amazon.com → Review Upload → Customer Review → Linear regression algorithm → Customer sentimental analysis function → Review sentimental data → Check product rating → Accuracy of positive or negative → Result



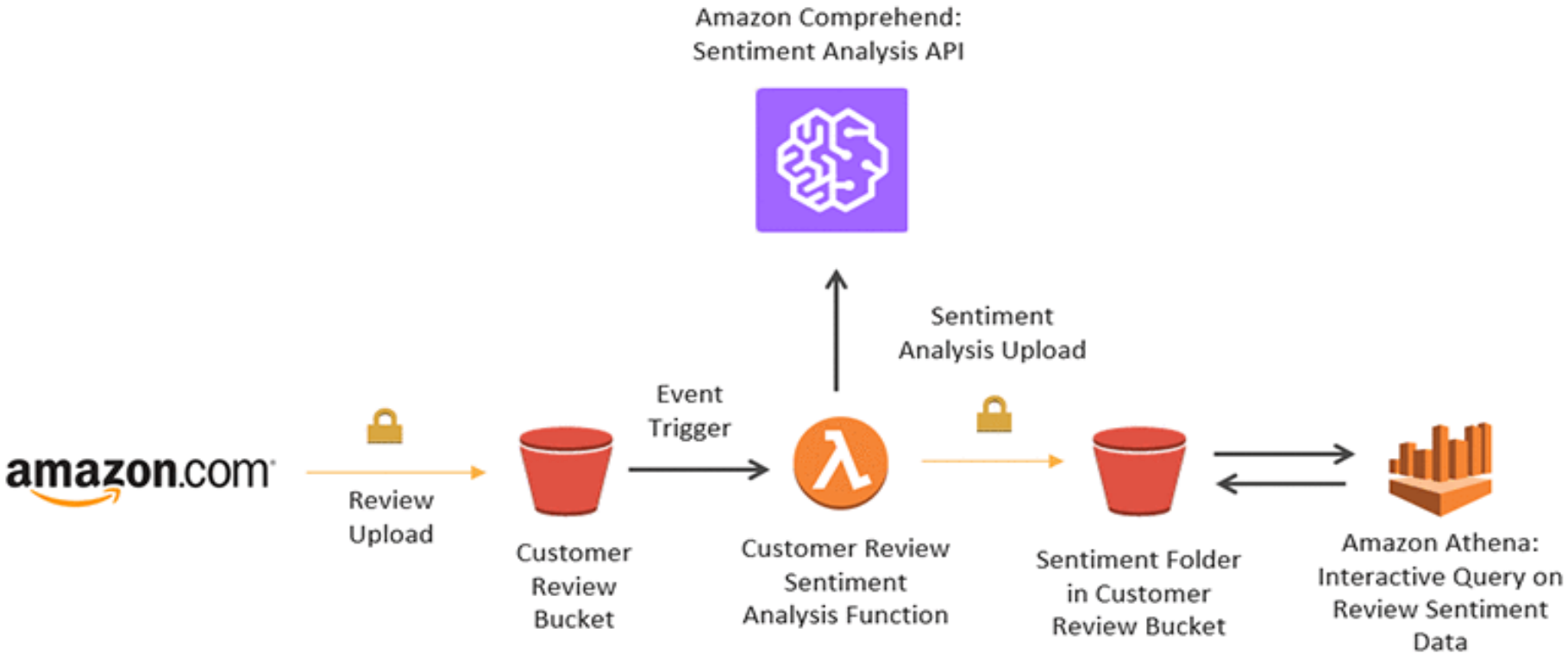





# OVERVIEW


Amazon Comprehend:
Sentiment Analysis API
Sentiment
Analysis Upload
Event
Trigger
amazon.com
Review
Upload
Customer
Review
Bucket
Customer Review
Sentiment
Analysis Function
Sentiment Folder
in Customer
Review Bucket
Amazon Athena:
Interactive Query on
Review Sentiment
Data

# SYSTEM DESIGN
# MODULES

- 1. Data collection.
- 2.Data preprocessing.
- 3.Machine learning algorithm training
- 4.Training and testing
- 5.Prediction product review

# DATA COLLECTION

- Real time data collected from Twitter ,kaggle, UCI , Data.gov.
- Collection of data is one of the major and most important tasks of any machine learning projects. Because the input we feed to the algorithms is data. So, the algorithms efficiency and accuracy depends upon the correctness and quality of data collected. So as the data same will be the output.

# DATA PREPROCESSING

- Collecting the data is one task and making that data useful is an-other vital task. Data collected from various means will be in an unorganized format and there may be lot of null values, in-valid data values and unwanted data. Cleaning all these data and replacing them with appropriate or approximate data and removing null and missing data and replacing them with some fixed alternate values are the basic steps in pre processing of data. Even data collected may contain completely garbage values. It may not be in exact format or way that is meant to be. All such cases must be verified and replaced with alternate values to make data meaning meaningful and useful for further processing. Data must be kept in a organized format.

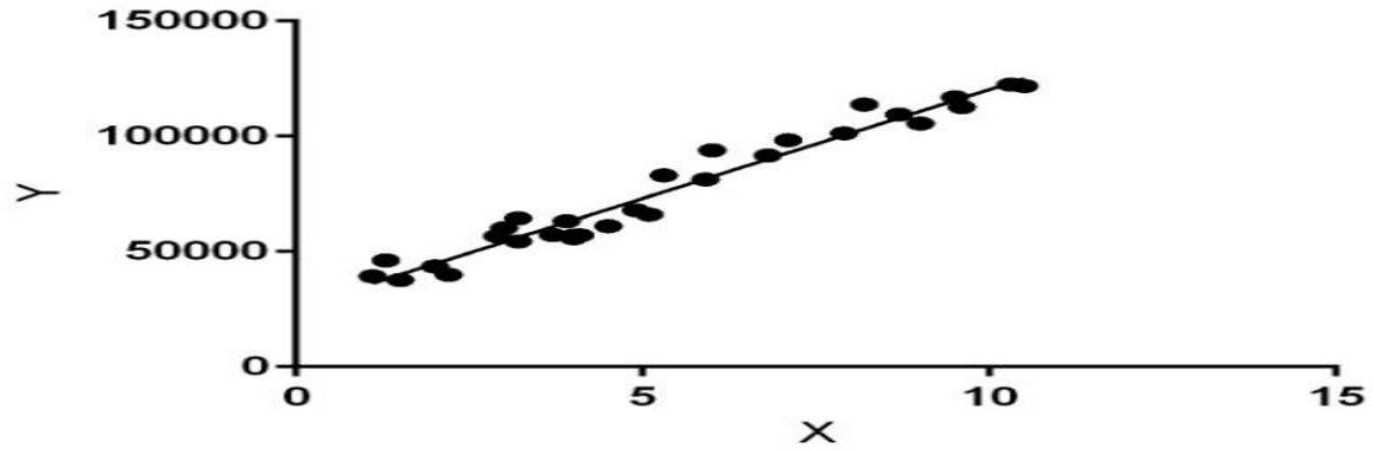
# MACHINE LEARNING ALGORITHM

- The next step is algorithms are applied to data and results are noted and observed. The linear regression algorithm applied as to improve accuracy at each stage.

# ALGORITHMS

- **LINEAR REGRESSION ALGORITHM**

- **Linear Regression** is a machine learning algorithm based on **supervised learning**. It performs a **regression task**.



$$y = \theta_1 + \theta_2.x$$

- While training the model we are given :
  **x:** input training data (univariate – one input variable(parameter))
  **y:** labels to data (supervised learning)
- When training the model – it fits the best line to predict the value of y for a given value of x. The model gets the best regression fit line by finding the best $\theta_1$ and $\theta_2$ values.
  **$\theta_1$:** intercept
  **$\theta_2$:** coefficient of x
- Once we find the best $\theta_1$ and $\theta_2$ values, we get the best fit line. So when we are finally using our model for prediction, it will predict the value of y for the input value of x.

# Cost Function (J):

- Cost function(J) of Linear Regression is the **Root Mean Squared Error (RMSE)** between predicted y value (pred) and true y value (y).

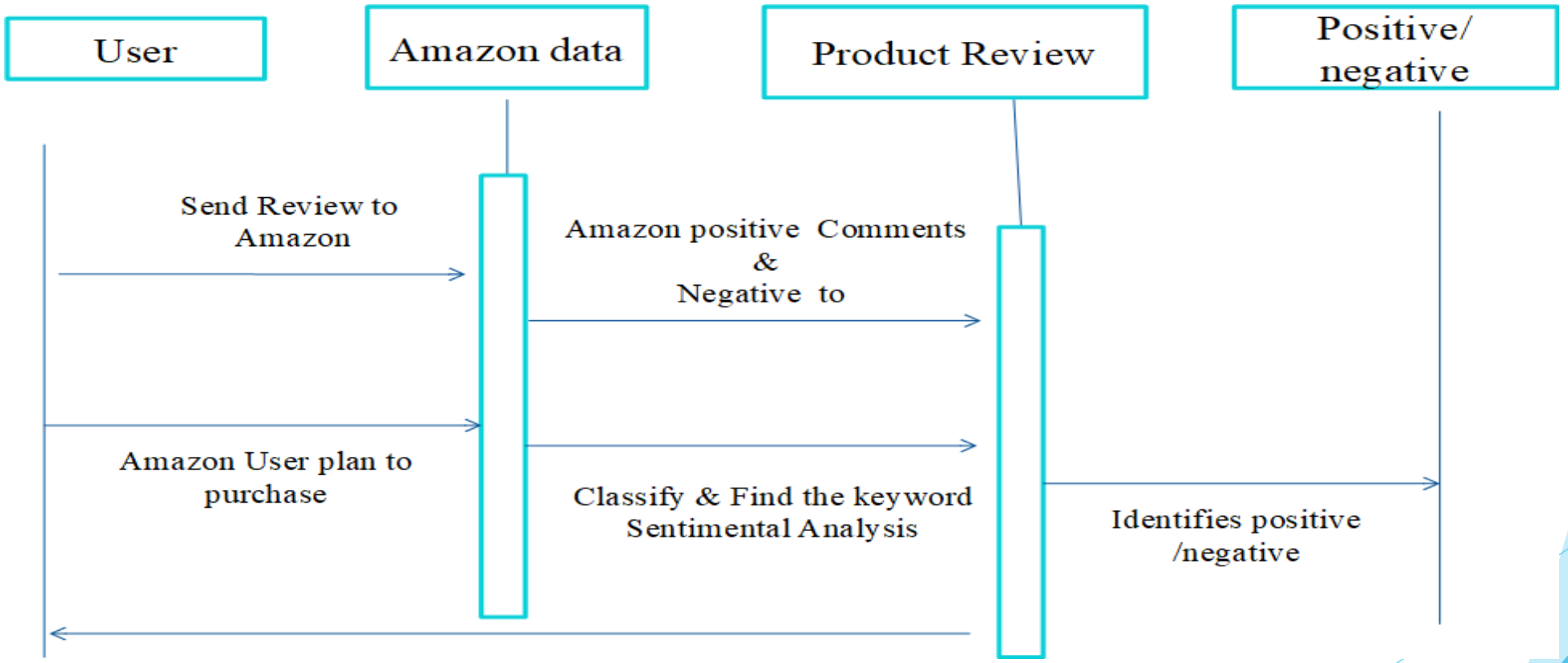
$$minimize \frac{1}{n} \sum_{i=1}^{n} (pred_i - y_i)^2$$

$$J = \frac{1}{n} \sum_{i=1}^{n} (pred_i - y_i)^2$$

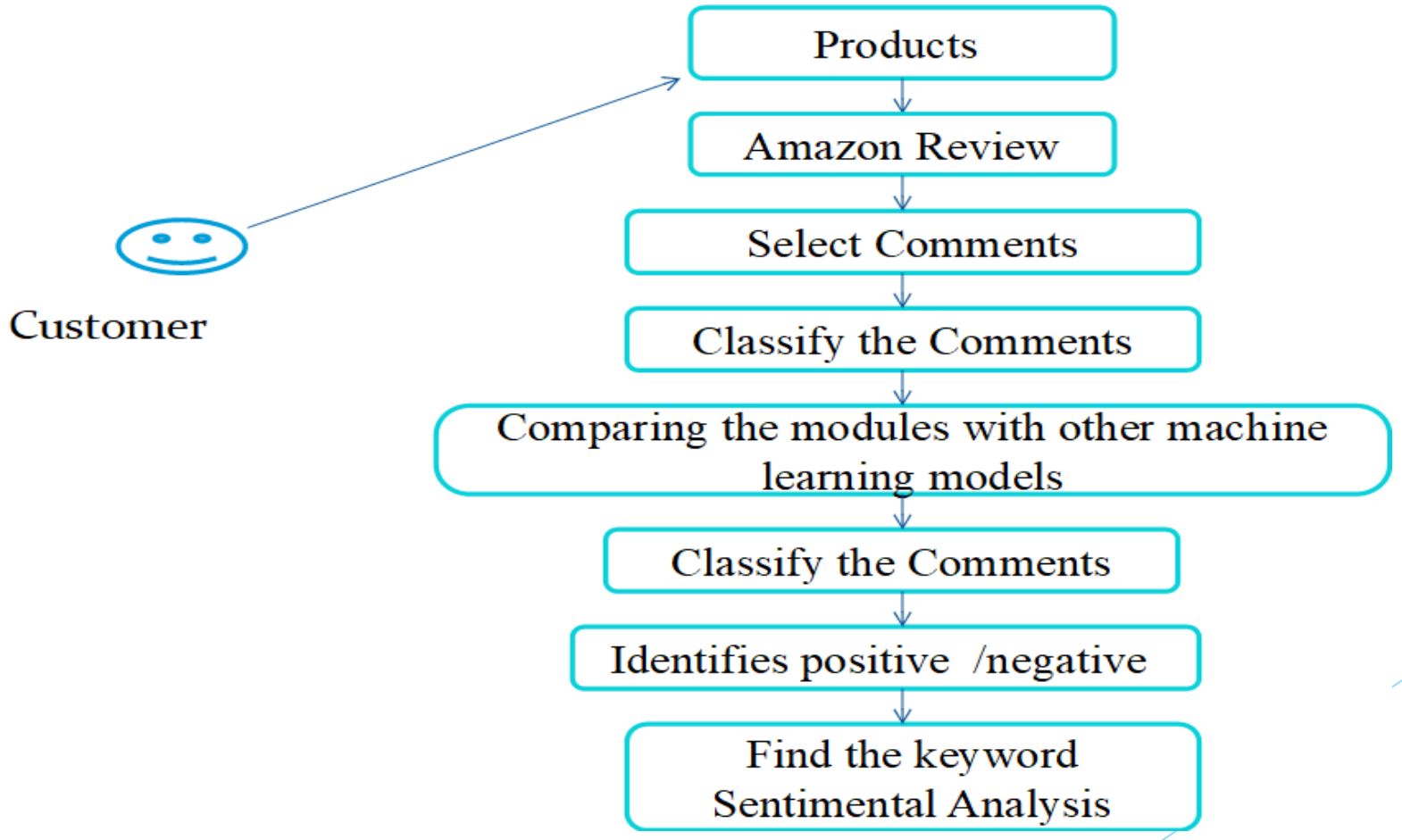
# TRAINING AND TESTING

- Finally after processing of data and training the very next task is obviously testing. This is where performance of the algorithm, quality of data, and required output all appears out. From the huge data set collected 80 percent of the data is utilized for training and 20 percent of the data is reserved for testing. Training as discussed before is the process of making the machine to learn and giving it the capability to make further predictions based on the training it took.
- Where as testing means already having a predefined data set with output also previously labelled and the model is tested whether it is working properly or not and is giving the right prediction or not. If maximum number of predictions are right then model will have a good accuracy percentage and is reliable to continue with otherwise better to change the model.

# SEQUENCE DIAGRAM FOR PROPOSED SYSTEM



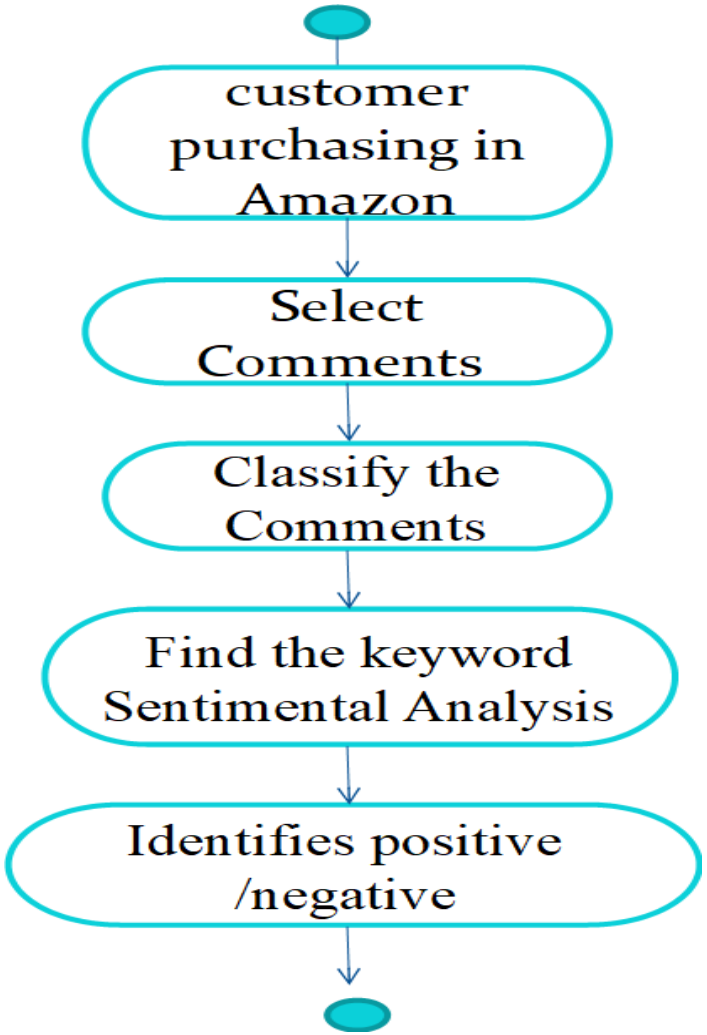
# USE CASE DIAGRAM FOR PROPOSED SYSTEM



# ATTRIBUTES

- 1-5 (stars)  reviewer
- IDReviewer  Idreviewer
- Name  Person's name (no standard format)
- Helpful  Helpfulness rating of the review
- Review Time  YYYY-MM—Dd
- unix  Review Time  Time of the review (unix time)
- pos_neg  Positive for 4-5 or 3 for  Neutral
- Negative for 1-2

# STATE CHART DIAGRAM



# TECHNOLOGIES

## Software Requirement:

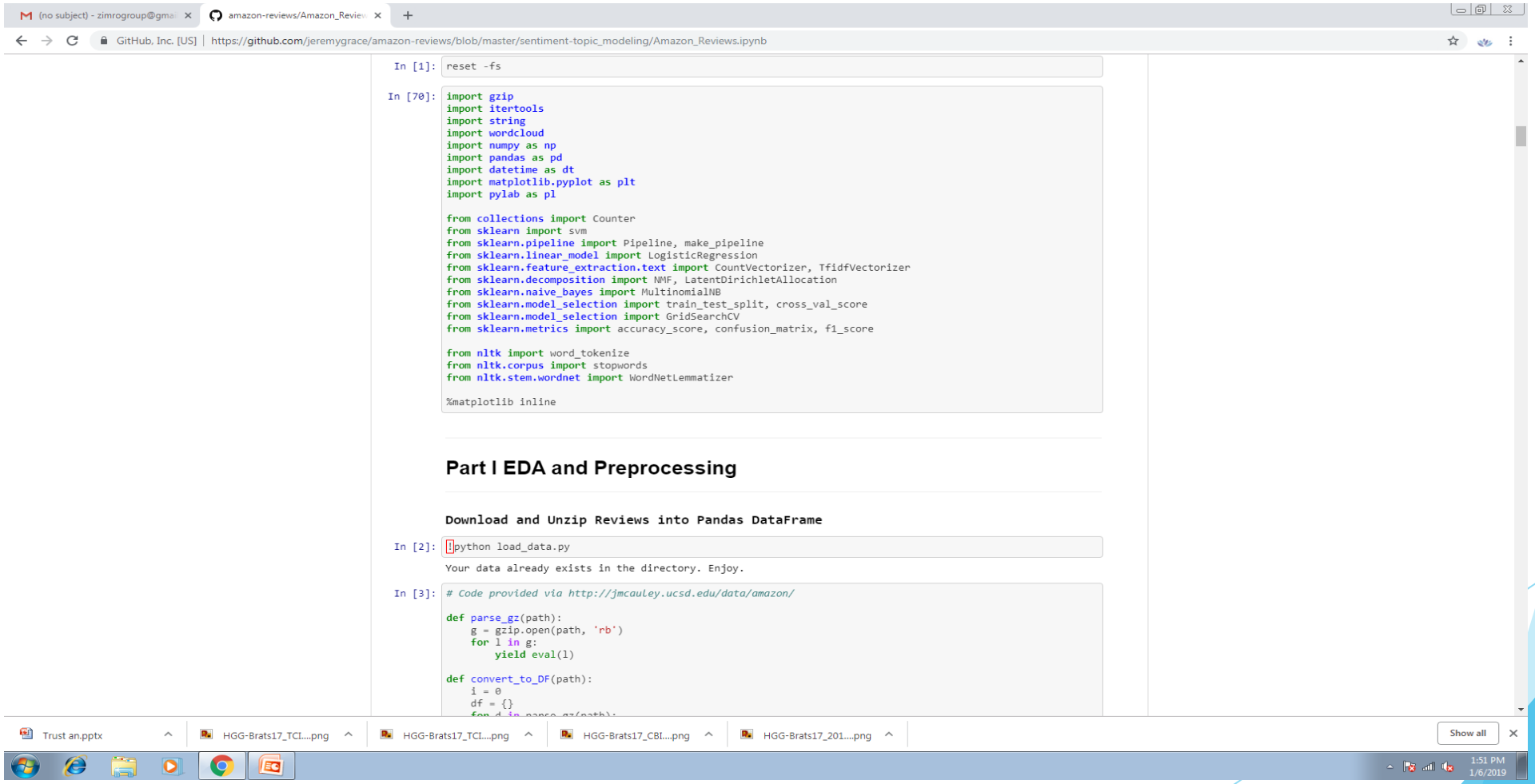
- Python 3.7
- Jupyter notebook
- Anaconda navigator
- Amazon Data sets

## Hardware Requirement:

- Processor : Dual core
- RAM:2GB
- Harddisk:500GB
- Speed:1.3Ghz

# SAMPLE RESULT WITH ITS OUTPUT SCREEN

# Sample code



(no subject) - zimrogroup@gma | amazon-reviews/Amazon_Revie

GitHub, Inc. [US] | https://github.com/jeremygrace/amazon-reviews/blob/master/sentiment-topic_modeling/Amazon_Reviews.ipynb

In [1]:
```
reset -fs
```

In [70]:
```
import gzip
import itertools
import string
import wordcloud
import numpy as np
import pandas as pd
import datetime as dt
import matplotlib.pyplot as plt
import pylab as pl

from collections import Counter
from sklearn import svm
from sklearn.pipeline import Pipeline, make_pipeline
from sklearn.linear_model import LogisticRegression
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from sklearn.decomposition import NMF, LatentDirichletAllocation
from sklearn.naive_bayes import MultinomialNB
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import accuracy_score, confusion_matrix, f1_score

from nltk import word_tokenize
from nltk.corpus import stopwords
from nltk.stem.wordnet import WordNetLemmatizer

%matplotlib inline
```

## Part I EDA and Preprocessing

### Download and Unzip Reviews into Pandas DataFrame

In [2]:
```
!python load_data.py
```

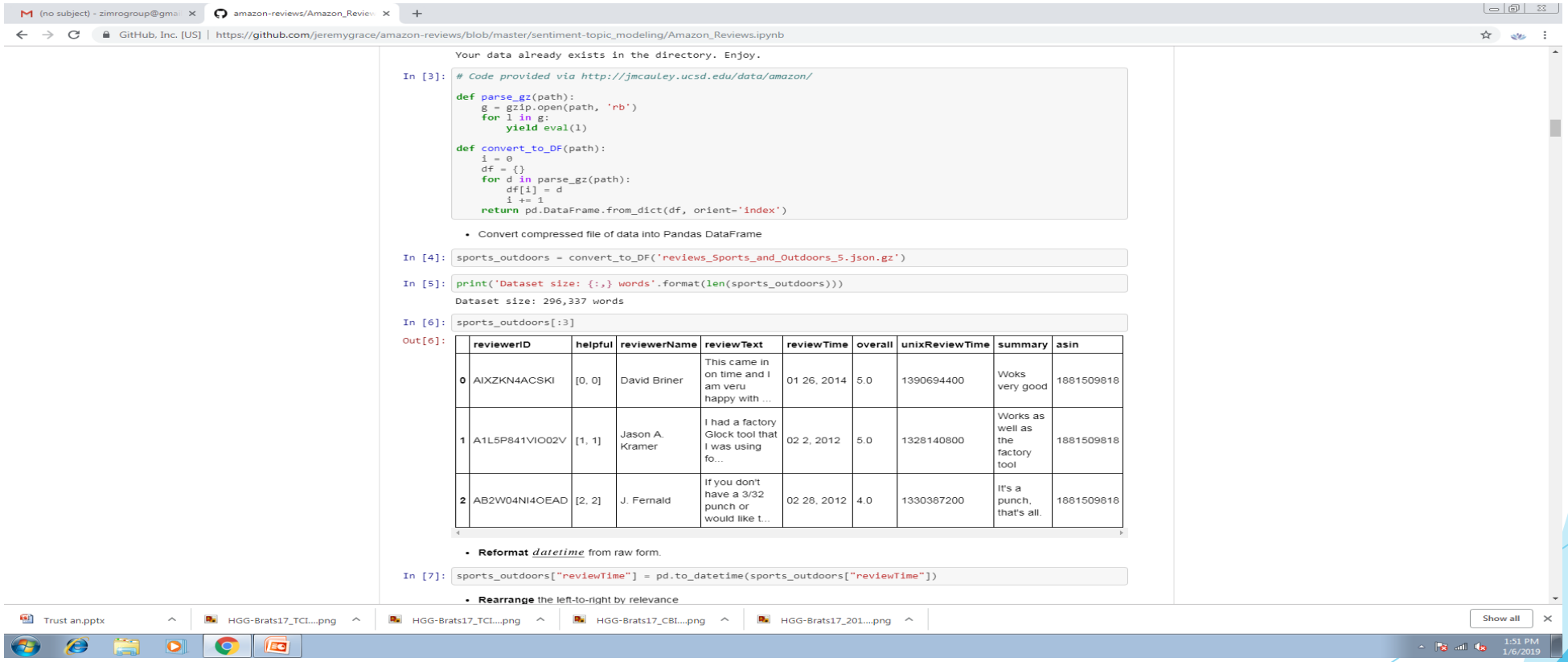
```
Your data already exists in the directory. Enjoy.
```

In [3]:
```
# Code provided via http://jmcauley.ucsd.edu/data/amazon/

def parse_gz(path):
    g = gzip.open(path, 'rb')
    for l in g:
        yield eval(l)

def convert_to_DF(path):
    i = 0
    df = {}
    for d in parse_gz(path):
```

Trust an.pptx | HGG-Brats17_TCI....png | HGG-Brats17_TCI....png | HGG-Brats17_CBI....png | HGG-Brats17_201....png | Show all

1:51 PM 1/6/2019

# ATTRIBUTE DESCRIPTION



(no subject) - zimrogroup@gma | amazon-reviews/Amazon_Revie

GitHub, Inc. [US] | https://github.com/jeremygrace/amazon-reviews/blob/master/sentiment-topic_modeling/Amazon_Reviews.ipynb

```
Your data already exists in the directory. Enjoy.
```

In [3]:
```
# Code provided via http://jmcauley.ucsd.edu/data/amazon/

def parse_gz(path):
    g = gzip.open(path, 'rb')
    for l in g:
        yield eval(l)

def convert_to_DF(path):
    i = 0
    df = {}
    for d in parse_gz(path):
        df[i] = d
        i += 1
    return pd.DataFrame.from_dict(df, orient='index')
```

- Convert compressed file of data into Pandas DataFrame

In [4]:
```
sports_outdoors = convert_to_DF('reviews_Sports_and_Outdoors_5.json.gz')
```

In [5]:
```
print('Dataset size: {:,} words'.format(len(sports_outdoors)))
```

```
Dataset size: 296,337 words
```

In [6]:
```
sports_outdoors[:3]
```

Out[6]:

| | reviewerID | helpful | reviewerName | reviewText | reviewTime | overall | unixReviewTime | summary | asin |
|---|---|---|---|---|---|---|---|---|---|
| 0 | AIXZKN4ACSKI | [0, 0] | David Briner | This came in on time and I am veru happy with ... | 01 26, 2014 | 5.0 | 1390694400 | Woks very good | 1881509818 |
| 1 | A1L5P841VIO02V | [1, 1] | Jason A. Kramer | I had a factory Glock tool that I was using fo... | 02 2, 2012 | 5.0 | 1328140800 | Works as well as the factory tool | 1881509818 |
| 2 | AB2W04NI4OEAD | [2, 2] | J. Fernald | If you don't have a 3/32 punch or would like t... | 02 28, 2012 | 4.0 | 1330387200 | It's a punch, that's all. | 1881509818 |

- **Reformat** *datetime* from raw form.

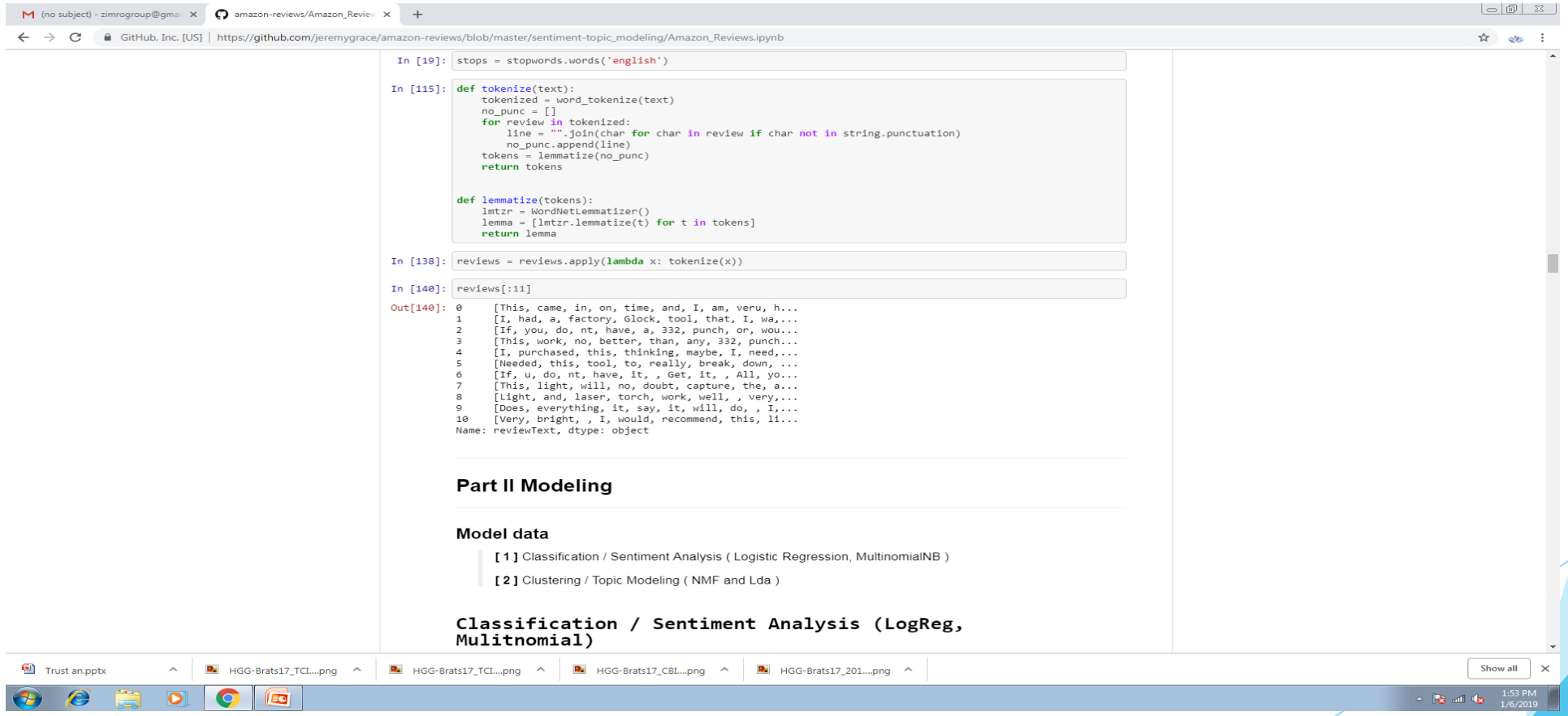
In [7]:
```
sports_outdoors["reviewTime"] = pd.to_datetime(sports_outdoors["reviewTime"])
```

- **Rearrange** the left-to-right by relevance

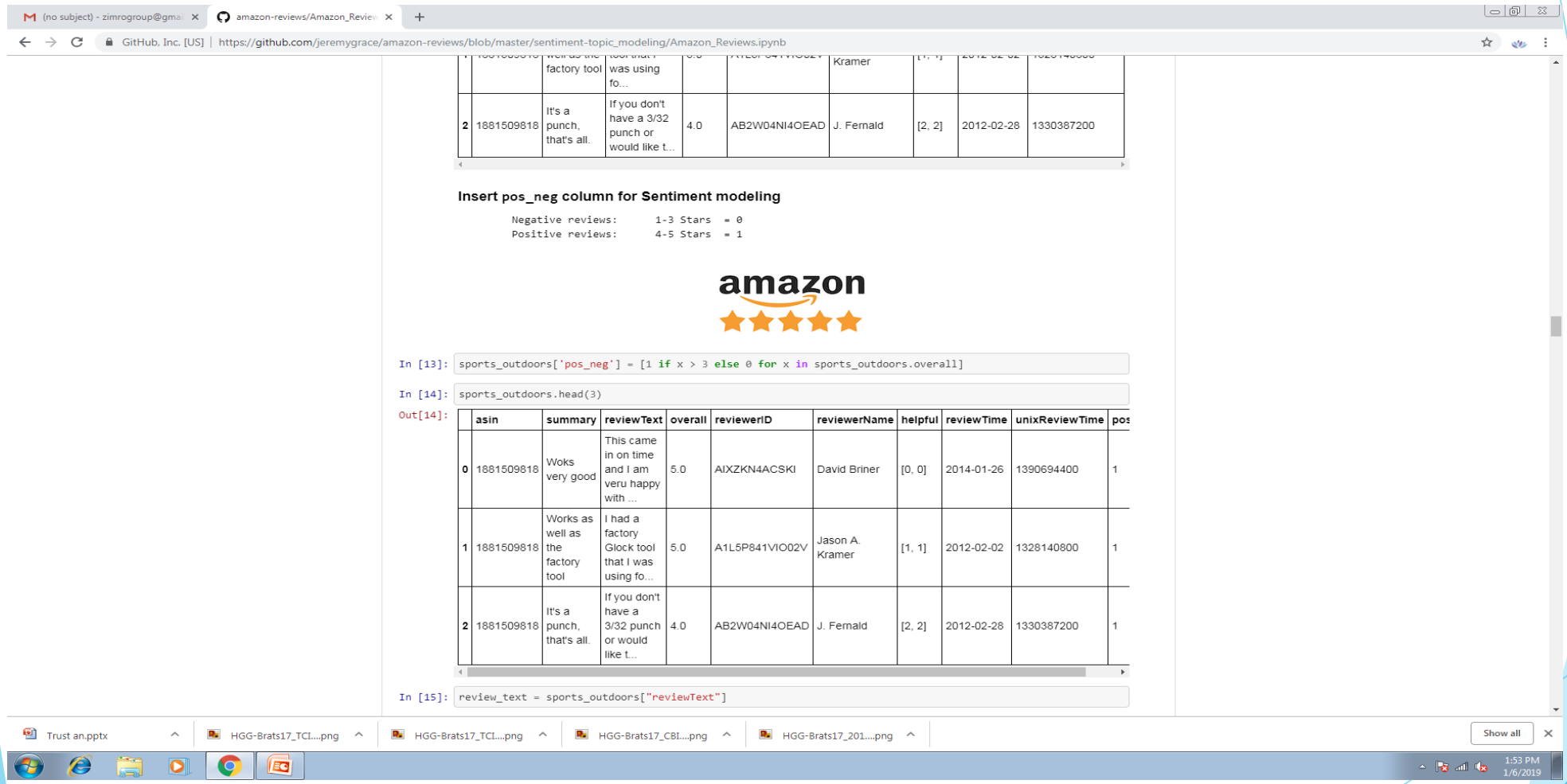
Trust an.pptx | HGG-Brats17_TCI....png | HGG-Brats17_TCI....png | HGG-Brats17_CBI....png | HGG-Brats17_201....png | Show all

1:51 PM 1/6/2019

# REVIEW MODELING



# REVIEWS(OUTPUT SCREEN)



| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | factory tool | was using fo... | | | Kramer | | | |
| 2 | 1881509818 | It's a punch, that's all. | If you don't have a 3/32 punch or would like t... | 4.0 | AB2W04NI4OEAD | J. Fernald | [2, 2] | 2012-02-28 | 1330387200 |

**Insert pos_neg column for Sentiment modeling**

```
Negative reviews:      1-3 Stars  = 0
Positive reviews:      4-5 Stars  = 1
```

amazon

In [13]:
```
sports_outdoors['pos_neg'] = [1 if x > 3 else 0 for x in sports_outdoors.overall]
```

In [14]:
```
sports_outdoors.head(3)
```

Out[14]:

| | asin | summary | reviewText | overall | reviewerID | reviewerName | helpful | reviewTime | unixReviewTime | pos |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1881509818 | Woks very good | This came in on time and I am veru happy with ... | 5.0 | AIXZKN4ACSKI | David Briner | [0, 0] | 2014-01-26 | 1390694400 | 1 |
| 1 | 1881509818 | Works as well as the factory tool | I had a factory Glock tool that I was using fo... | 5.0 | A1L5P841VIO02V | Jason A. Kramer | [1, 1] | 2012-02-02 | 1328140800 | 1 |
| 2 | 1881509818 | It's a punch, that's all. | If you don't have a 3/32 punch or would like t... | 4.0 | AB2W04NI4OEAD | J. Fernald | [2, 2] | 2012-02-28 | 1330387200 | 1 |

In [15]:
```
review_text = sports_outdoors["reviewText"]
```

# FEASIBILITY REPORT

- **Technical Feasibility**: The sentimental based product reviews is built using reliable open source python framework.
- The dataset trained help of SQL and Tensor flow.
- **Economic Feasibility**: Product reviews developed by open source frameworks so its an economic.
- **Operational Feasibility**: the end user need to know python.
- Its easy to develop and operated.

# CONCLUSION

- The Amazon review dataset should be taken from kaggle.com and the linear regression algorithm is applied on the dataset to obtain sentimental data to accurate the positive or negative results for buyer convenience.

# REFERENCES

## Reference paper


[1] K. Ravi and V. Ravi, "A survey on opinion mining and sentiment analysis: Tasks, approaches and applications," Know. Based Syst., vol. 89, pp. 14–46, Nov. 2015

[2] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," Ain Shams Eng. J., vol. 5, no. 4, pp. 1093–1113, Dec. 2014.

[3] M. Tsytsarau and T. Palpanas, "Survey on mining subjective data on the Web," Data Mining Knowl. Discovery, vol. 24, no. 3, pp. 478–514, May 2012

[4] A.-D. Vo and C.-Y. Ock, "Sentiment classification: A combination of PMI, SentiWordNet and fuzzy function," in Proc. ICCCI, Ho Chi Minh City, Vietnam, 2012, pp. 373–382

[5] B. Liu, "Sentiment analysis and opinion mining," Synth. Lectures Human Lang. Technol., vol. 5, no. 1, pp. 1–167, 2012