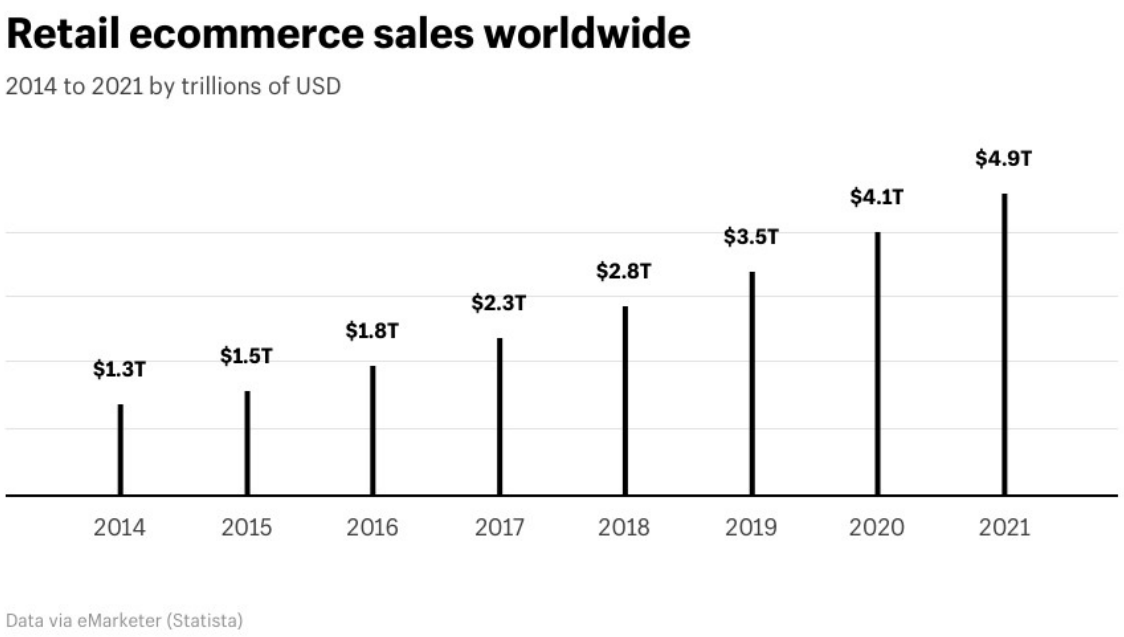
Predict Rating from Review for E-Commerce Sites

Introduction

Ecommerce was introduced about 40 years ago in its earliest form. Since then, electronic commerce has helped countless businesses grow with the help of new technologies, improvements in internet connectivity, added security with payment gateways, and widespread consumer and business adoption.



Amid slowing economic activity, COVID-19 has led to a surge in e-commerce and accelerated digital transformation. As lockdowns became the new normal, businesses and consumers increasingly "went digital", providing and purchasing more goods and services online, raising e-commerce's share of global retail trade from 14% in 2019 to about 17% in 2020.

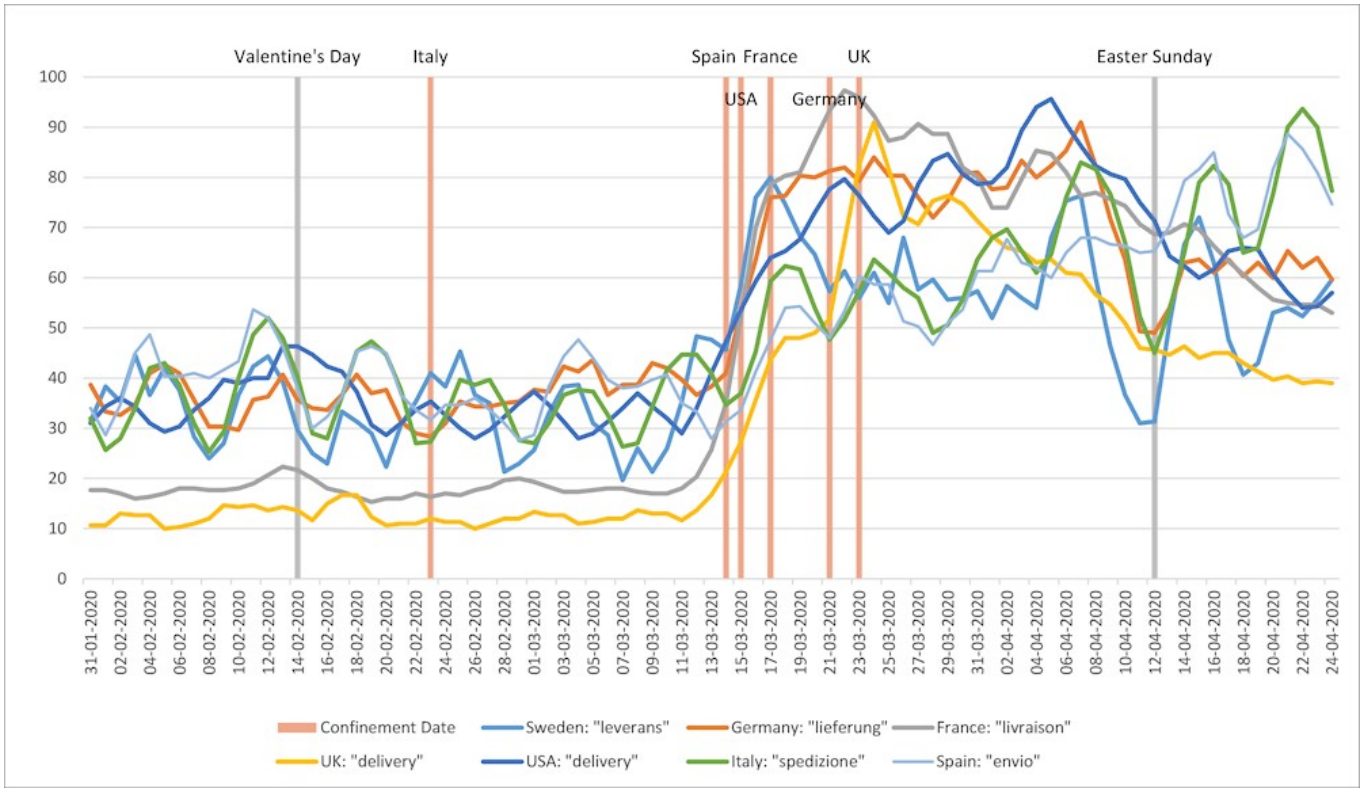
These and other findings are showcased in a new report, COVID-19 and E-Commerce: A Global Review, by UNCTAD and eTrade for all partners, reflecting on the powerful global and regional industry transformations recorded throughout 2020. At an event to release the report, UN General Assembly President Volkan Bozkir said the trend towards e-commerce is likely to continue throughout the recovery from COVID-19.

The COVID-19 crisis accelerated an expansion of e-commerce towards new firms, customers and types of products. It has provided customers with access to a significant variety of products from the convenience and safety of their homes, and has enabled firms to continue operation in spite of contact restrictions and other confinement measures.

Despite persistent cross-country differences, the COVID-19 crisis has enhanced dynamism in the e-commerce landscape across countries and has expanded the scope of e-commerce, including through new firms, consumer segments (e.g. elderly) and products (e.g. groceries). Meanwhile, e-commerce transactions in many countries have partly shifted from luxury goods and services towards everyday necessities, relevant to a large number of individuals.

Some of these changes in the e-commerce landscape will likely be of a long-term nature, in light of the possibility of new waves of the epidemic, the convenience of the new purchasing habits, learning costs and the incentive for firms to capitalise on investments in new sales channels

Google search interest in "delivery", selected OECD countries (February to April 2020)



E-Commerce

E-commerce (electronic commerce) is the buying and selling of goods and services, or the transmitting of funds or data, over an electronic network, primarily the internet. These business transactions occur either as business-to-business (B2B), business-to-consumer (B2C), consumer-to-consumer or consumer-to-business. The terms e-commerce and e-business are often used interchangeably. The term e-tail is also sometimes used in reference to the transactional processes that make up online retail shopping.

In the last decade, widespread use of e-commerce platforms such as Amazon and eBay has contributed to substantial growth in online retail. In 2007, e-commerce accounted for 5.1% of total retail sales; in 2019, e-commerce made up 16.0%.

How does e-commerce work?

E-commerce is powered by the internet, where customers can access an online store to browse through, and place orders for products or services via their own devices.

As the order is placed, the customer's web browser will communicate back and forth with the server hosting the online store website. Data pertaining to the order will then be relayed to a central computer known as the order manager -- then forwarded to databases that manage inventory levels, a merchant system that manages payment information (using applications such as PayPal), and a bank computer -- before circling back to the order manager. This is to make sure that store inventory and customer funds are sufficient for the order to be processed. After the order is validated, the order manager will notify the store's web server, which will then display a message notifying the customer that their order has been successfully processed. The order manager will then send order data to the warehouse or fulfillment department, in order for the product or service to be successfully dispatched to the customer. At this point tangible and/or digital products may be shipped to a customer, or access to a service may be granted.

Platforms that host e-commerce transactions may include online marketplaces that sellers simply sign up for, such as Amazon.com; software as a service (SaaS) tools that allow customers to 'rent' online store infrastructures; or open source tools for companies to use in-house development to manage.

Types of Ecommerce

Generally, there are six main models of ecommerce that businesses can be categorized into:

1. *Business-to-Consumer (B2C)*
   B2C ecommerce encompasses transactions made between a business and a consumer. B2C is one of the most popular sales models in the ecommerce context. For example, when you buy shoes from an online shoe retailer, it's a business-to-consumer transaction.
2. *Business-to-Business (B2B)*
   Unlike B2C, B2B ecommerce encompasses sales made between businesses, such as a manufacturer and a wholesaler or retailer. B2B is not consumer-facing and happens only between businesses.
   Business-to-business sales often focus on raw materials or products that are repackaged before being sold to customers.
3. *Consumer-to-Consumer (C2C)*
   C2C is one of the earliest forms of ecommerce. Customer-to-customer relates to the sale of products or services between customers. This includes C2C selling relationships, such as those seen on eBay or Amazon.
4. *Consumer-to-Business (C2B)*
   C2B reverses the traditional ecommerce model, meaning individual consumers make their products or services available for business buyers.
   For example, the iStockPhoto business model in which stock photos are available online for purchase directly from different photographers.
5. *Business-to-Administration (B2A)*
   B2A covers the transactions made between online businesses and administrations. An example would be the products and services related to legal documents, social security, etc.
6. *Consumer-to-Administration (C2A)*
   C2A is similar to B2A, but consumers sell online products or services to an administration. C2A might include online consulting for education, online tax preparation, etc.
   B2A and C2A are focused on increased efficiency within the government via the support of information technology.

Recently, Indian government introduced various reforms to help the nation's e-commerce industry grow. The onset of lockdown and a push towards digitization proved to be a blessing in disguise for the sector. As consumers resorted to online shopping, e-commerce became the backbone for supplying essentials to the more than 1.3 billion people of India.

The e-commerce sector has registered a growth of 36% year over year in the last quarter of 2020. The biggest beneficiaries of this rise are the personal care, beauty, and wellness categories which grew by 95% year over year. Meanwhile, th electronics category saw a rise of 27% annual growth .

These figures were mainly driven by Tier 2 and Tier 3 cities, which accounted for a 90% year over year increase in volume and value growth in particular. In fact, these cities registered a growth of 14% and value share growth of 43% in the last quarter of 2020.

Tier 1 cities were dominated by the FMCG and healthcare categories with more than 150% growth. The e-commerce platforms that contributed to these figures were led by Amazon and Flipkart in addition to upcoming blooming companies like IndiaMart Nykaa,

Review Analysis

What is reviewing?

Reviewing is learning from experience - or enabling others to do so. Reviewing helps you get more from work, life and recreation -

especially if you have the reviewing skills to match your ambitions. A Definition of Reviewing

Reviewing is any process that helps you to make use of personal experience for your learning and development. These reviewing processes can include:

- reflecting on experience
- analysing experience
- making sense of experience
- communicating experience
- reframing experience
- learning from experience

Alternative terms for reviewing are 'processing', 'debriefing' and 'reflection'.
According to Dictionary.com imply careful examination of something, formulation of a judgment, and statement of the judgment, usually in written form. A review is a survey over a whole subject or division of it, or especially an article making a critical reconsideration and summary of something written.

Importance of Reviews in E-Commerce

When we go to make an online purchase, what's the first thing you do? In an ecommerce-driven world where customers can't physically experience products before purchasing, many consumers turn to online product reviews.

As online review sites such as Yelp! and Facebook have expanded, finding an opinion on just about anything is only a few clicks away. The proliferation of reviews has even gone so far as to shape how businesses are perceived online.

Who is Reading Online Reviews?

In today's web-based world, virtually everyone is reading online reviews. In fact, 91% of people read them and 84% trust them as much as they would a personal recommendation. The effects of reviews are measurable, too. The average customer is willing to spend 31% more on a retailer that has excellent reviews.

Negative reviews can carry as much weight as positive ones. One study found that 82% of those who read online reviews specifically seek out negative reviews. That may sound alarming — this stat only emphasizes that negative reviews aren't going unnoticed — but there are some benefits: Research indicates that users spend five times as long on sites when interacting with negative reviews, with an 85% increase in conversion rate.

Customers like to see lots of reviews. A single review with a few positive words makes up an opinion, but a few dozen that say the same thing make a consensus. The more reviews, the better, and one study found that consumers want to see at least 40 reviews to justify trusting an average star rating. However, a few reviews are still better than no reviews. One study found that, on average, products are 270% more likely to sell with as few as five reviews.

With the vast array of review sites and the level of trust most consumers have in reviews, it's a safe assumption that virtually everyone considering your products, no matter your target demographic, industry, or market, is reading online reviews before making a purchase.

Online Reviews are Essential for an Online Store

Online reviews can reveal a lot about an online store. A wealth of positive words can have a measurable impact on your sales, driving purchases and creating a base of consumers who will stand behind you and your product. These key points outline the benefits that make online reviews are essential for your online store.

1. Drive sales.
   Social proof refers to the psychological phenomenon in which people make judgments and decisions based on the collective actions of others. In this case, reading positive reviews from other people who made similar purchases drives confidence that buying a well-reviewed item is a good choice.
   In essence, people want proof from other consumers that a product or service is worthwhile, not just biased advertising from brands. Reviews are trusted 12 times more than other marketing materials, demonstrating that social proof is a powerful force.
2. Build trust.
   The global ecommerce market reached nearly $3.5 trillion in 2019. There are countless brands in every category, but without a way to verify quality and reliability, it's hard to know who to trust.
   While handling a product is the best way to gauge quality, reviews can be the next best thing for businesses that exist solely in the ecommerce space. Reading dozens of reviews that indicate good quality and services create an online reputation that customers can trust. In fact, customers are 63% more likely to trust and buy from a company with reviews.

3. Contribute to SEO efforts.
   Ranking high in the SERPs is a goal for most businesses. However, building an SEO-friendly web presence can take a lot of time and energy. Fortunately, customer reviews can further your mission without you lifting a finger.
   Most customers use keywords, like the name of the products, in their reviews, adding more content on the internet associated with you. That can benefit you twofold: your name is more likely to appear when web users search keywords related to your store, and they're most likely to see your positive reviews.
4. Aid customer decision-making.
   When purchasing online, the customer decision-making process becomes a lot more complicated. As such, most shoppers put a lot more time and energy into evaluating products, reading reviews, and comparing items with one another before pulling the trigger.
   Reviews are key to the decision-making process, helping customers to get a better idea about the product, including material, size, and shape. For example, a product may look too small to meet consumer needs in a picture, but customer reviews that address size more accurately can put a wary shopper at ease.
5. Enable problem-solving.
   Not all reviews are positive and, believe it or not, that's okay. No business is perfect, and reviews can help you identify pain points in need of improvement. Some negative reviews misinterpreted a situation or have been written by an angry customer. Still, if you see multiple negative reviews with similar complaints, you may have a problem worth addressing. If 15 different reviews praise your products but disparage your clunky checkout process, for example, it may be time to invest in creating a smoother, more efficient purchase process.

An astounding 94% of online consumers have been dissuaded from shopping based on negative reviews, so remedying the problematic trends you see can definitely be beneficial

Reasons why customer reviews are important

1. Better Understand your Customers & Improve Customer Service
   Analyzing reviews left by your customers, helps your company understand overall customer satisfaction, as they can provide your business with feedback regarding what your customers truly want.
   By using this insightful information as input, you will be able to improve customer service by quickly and efficiently resolve the issues that consumers faced, thereby creating a positive experience for the consumer and keeping your focus on their needs.
2. Credibility & Social Proof
   No doubt, we are social creatures since the moment we come to this world and we are interested in knowing what other say before we make our buying decisions. Much like we would ask friends and family for recommendations, review sites allow us to do this online with just some clicks.
3. Fight with experience to save margins
   Reviews enable new businesses to stand shoulder to shoulder with more established competition, and potentially gain a positive niche in people's estimation and expectations. Look at it this way…which company would you rather buy from: one with 50 3-star reviews or one with 5 5-star reviews? Voila! You just took the discussion away from the discount and price!
4. Allow Consumers to Have a Voice and Create Customer Loyalty
   Consumers that take the time to leave an online review for your business are far more likely to feel a certain loyalty to you and keep coming back. Through the act of leaving a review and establishing a relationship with your business, it allows consumers to feel like they have a voice even behind a desktop and/or mobile and/ or tablet screen and are able to provide feedback in a positive and meaningful way.
5. Improve Rankings
   Reviews appear to be the most prominent ranking factor in local search. It helps businesses rank well, even if they have low quality link profiles.
   According to SEJ, "pages with reviews which mention a keyword and/or the name of a city, were found to have higher rankings in Google's local pack. At a high level, having a keyword you are trying to rank for, and a mention of a city you are working to rank in, in reviews has a high correlation with high ranking Google My Business results".
6. Consumers are Doing your Marketing for You
   Positive online business reviews are worth a great deal and can offer your business benefits that a simple marketing campaign can' t. In a nutshell, they are like micro – marketing campaigns that keep working long after the online review has been posted, providing, thus, a constant positive image to potential customers and creating a continual brand awareness that benefits the business for both the short and the long term.
7. Reviews Generate More Reviews
   When a business has already received online reviews, it encourages other visitors to leave their own feedback. Just the appearance of several reviews seems to be enough to give new customers the incentive and confidence to submit their own opinion on a particular product or service

Target of the Project work

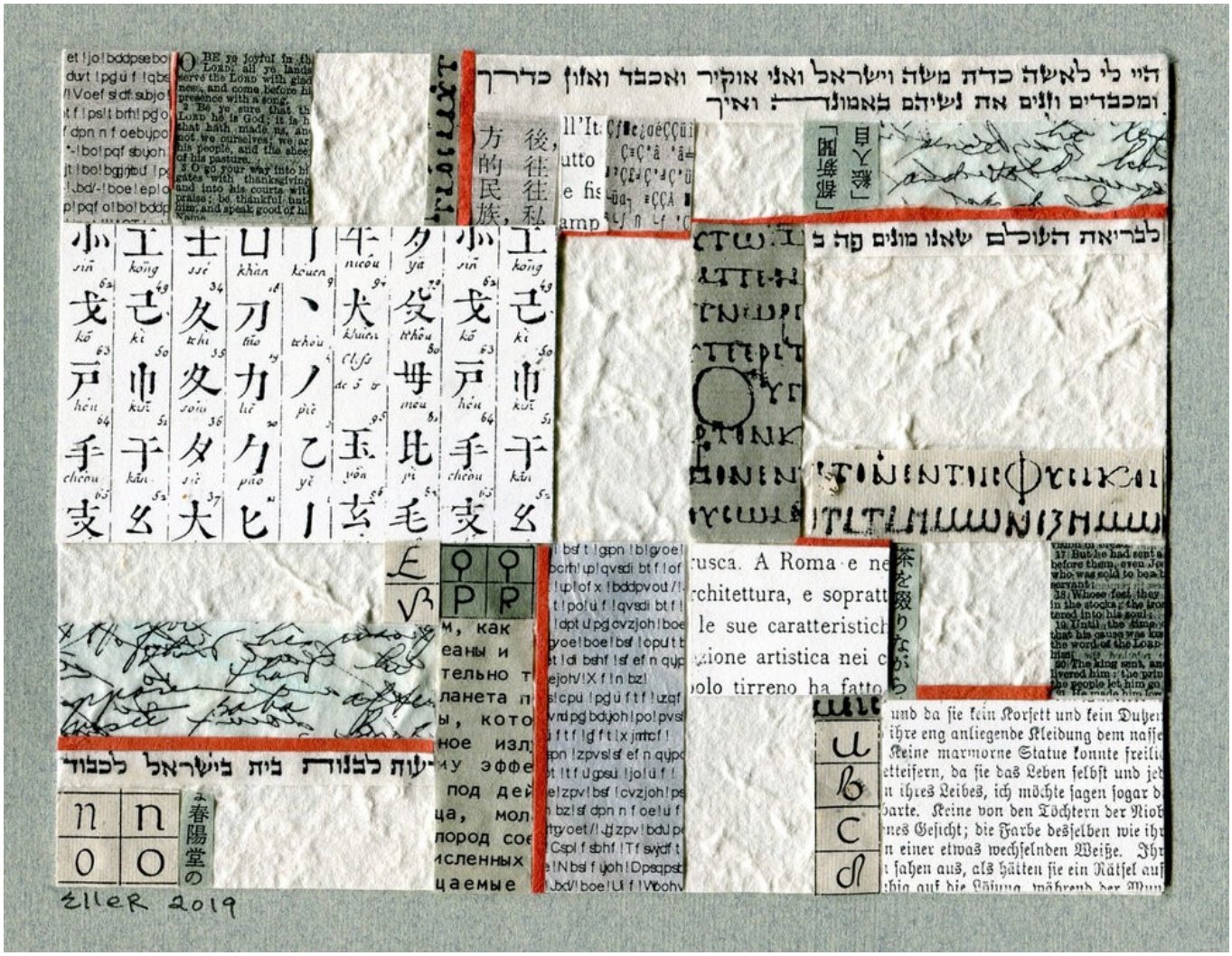
With the advent of technology placing a small window to the worlds of each people expcept for the ones living a hobbit's life under a dark cave or communities of tribes cutoff from the modern world completely, its no doubt that E-Commerce is a booming industry and with their large turnover every year in comparison to the various industries besides them having decades of history in shaping mankind.

Our project is primarily a machine learning work and in tandem with its title "Predict Rating from Reviews for E-Commerce Sites" is primarily tasked with generating a machine algorithm that understands the reviews posted by users and determine the aggregate rating of a product bought by them, whether the product is recommendable to any future buyers or be scrapped from the listing due to poor performance, ultimately understand the sentiment of the the buyer conveyed in the reviews. Thus we need to employ NLP(Natural Language Processing) to understand the words of the review and ultimately make the system understand and make out the ideas conveyed.

Besides learning the sentiment of a review we need to as per the preset provisions made for this project, hit the following roadblocks while we are understanding sentiments

1. Load the dataset
2. Clear the dataset
3. Explore the dataset
4. Create the visualization
5. Train on all the Models
6. Create a summary of Model which model is best and why

Natural Language Processing



What Is Natural Language Processing (NLP)?

Natural Language Processing (NLP) is a subfield of artificial intelligence (AI). It helps machines process and understand the human language so that they can automatically perform repetitive tasks. Examples include machine translation, summarization, ticket classification, and spell check.

Why Is Natural Language Processing Important?

One of the main reasons natural language processing is so critical to businesses is that it can be used to analyze large volumes of text data, like social media comments, customer support tickets, online reviews, news reports, and more.

All this business data contains a wealth of valuable insights, and NLP can quickly help businesses discover what those insights are. It does this by helping machines make sense of human language in a faster, more accurate, and more consistent way than human agents. NLP tools process data in real time, 24/7, and apply the same criteria to all your data, so you can ensure the results you receive are accurate – and not riddled with inconsistencies.

Once NLP tools can understand what a piece of text is about, and even measure things like sentiment, businesses can start to prioritize and organize their data in a way that suits their needs.

Challenges of NLP

While there are many challenges in natural language processing, the benefits of NLP for businesses are huge making NLP a worthwhile investment. However, it's important to know what those challenges are before getting started with NLP. Human language is complex, ambiguous, disorganized, and diverse. There are more than 6,500 languages in the world, all of them with their own syntactic and semantic rules.Even humans struggle to make sense of language.

So for machines to understand natural language, it first needs to be transformed into something that they can interpret. In NLP, syntax and semantic analysis are key to understanding the grammatical structure of a text and identifying how words relate to each other in a given context. But, transforming text into something machines can process is complicated.

Data scientists need to teach NLP tools to look beyond definitions and word order, to understand context, word ambiguities, and other complex concepts connected to human language.

How Does Natural Language Processing Work?

In natural language processing, human language is separated into fragments so that the grammatical structure of sentences and the meaning of words can be analyzed and understood in context. This helps computers read and understand spoken or written text in the same way as humans.
Here are a few fundamental NLP pre-processing tasks data scientists need to perform before NLP tools can make sense of human language:

- Tokenization: breaks down text into smaller semantic units or single clauses Part-of-speech-tagging: marking up words as nouns, verbs, adjectives, adverbs, pronouns, etc
- Stemming and lemmatization: standardizing words by reducing them to their root forms
- Stop word removal: filtering out common words that add little or no unique information, for example, prepositions and articles (at, to, a, the).

Only then can NLP tools transform text into something a machine can understand.

Sentiment analysis

A contextual mining of text which identifies and extracts subjective information in source material, and helping a business to understand the social sentiment of their brand, product or service while monitoring online conversations. However, analysis of social media streams is usually restricted to just basic sentiment analysis and count based metrics. This is akin to just scratching the surface and missing out on those high value insights that are waiting to be discovered.
With the recent advances in deep learning, the ability of algorithms to analyse text has improved considerably. Creative use of advanced artificial intelligence techniques can be an effective tool for doing in-depth research. We believe it is important to classify incoming customer conversation about a brand based on following lines:

1. Key aspects of a brand's product and service that customers care about.
2. Users' underlying intentions and reactions concerning those aspects.

These basic concepts when used in combination, become a very important tool for analyzing millions of brand conversations with human level accuracy.

Library Imports

Let us now import the libraries required for understanding the sentiments of a buyer.

In [34]:
```
# imports
import nltk
import numpy as np
import pandas as pd
import  matplotlib.pyplot as plt
import seaborn as sns
import warnings as warns
import time
from copy import deepcopy
from nltk.corpus import stopwords
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
import re
from sklearn.metrics import precision_recall_fscore_support
from sklearn.metrics import confusion_matrix,ConfusionMatrixDisplay
from wordcloud import WordCloud, STOPWORDS
from sklearn.metrics import accuracy_score, classification_report
```

Primaries

In [35]:
```
nltk.download('stopwords')
nltk.download('punkt')
pd.set_option('display.max_colwidth', -1)
warns.filterwarnings("ignore")
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
```

Data Acquirement

The data on which is project is worked on is obtained from Kaggle.com provided by the user Nick Brooks (Kaggle ID: nicapotato) link to which is as given below

https://www.kaggle.com/nicapotato/womens-ecommerce-clothing-reviews

The dataset contains 23486 lines of records segmented into 10 fields of information

- Clothing ID: Integer Categorical variable that refers to the specific piece being reviewed.
- Age: Positive Integer variable of the reviewers age.
- Title: String variable for the title of the review.
- Review Text: String variable for the review body.
- Rating: Positive Ordinal Integer variable for the product score granted by the customer from 1 Worst, to 5 Best.
- Recommended IND: Binary variable stating where the customer recommends the product where 1 is recommended, 0 is not recommended.
- Positive Feedback Count: Positive Integer documenting the number of other customers who found this review positive.
- Division Name: Categorical name of the product high level division.
- Department Name: Categorical name of the product department name.
- Class Name: Categorical name of the product class name.

Now as per the provisions of this project work we will be accessing the same data stored as a copy in Github made exclusively for this project

https://github.com/WolfDev8675/RepoSJX7/tree/Assign5_3/Data

Imports

In [36]:
```
#import data
data_full=pd.read_csv("https://raw.githubusercontent.com/WolfDev8675/RepoSJX7/Assign5_3/Data/Womens%20Clo
```

Confirmation Display

In [37]:
```
data_full.head()
```

Out[37]:

	Unnamed: 0	Clothing ID	Age	Title	Review Text	Rating	Recommended IND	Positive Feedback Count	Division Name	Department Name	Class Name

	Unnamed: 0	Clothing ID	Age	Title	Review Text	Rating	Recommended IND	Positive Feedback Count	Division Name	Department Name	Class Name
0	0	767	33	NaN	Absolutely wonderful - silky and sexy and comfortable	4	1	0	Initmates	Intimate	Intimates
1	1	1080	34	NaN	Love this dress! it's sooo pretty. i happened to find it in a store, and i'm glad i did bc i never would have ordered it online bc it's petite. i bought a petite and am 5'8". i love the length on me- hits just a little below the knee. would definitely be a true midi on someone who is truly petite.	5	1	4	General	Dresses	Dresses
2	2	1077	60	Some major design flaws	I had such high hopes for this dress and really wanted it to work for me. i initially ordered the petite small (my usual size) but i found this to be outrageously small. so small in fact that i could not zip it up! i reordered it in petite medium, which was just ok. overall, the top half was comfortable and fit nicely, but the bottom half had a very tight under layer and several somewhat cheap (net) over layers. imo, a major design flaw was the net over layer sewn directly into the zipper - it c	3	0	0	General	Dresses	Dresses
3	3	1049	50	My favorite buy!	I love, love, love this jumpsuit. it's fun, flirty, and fabulous! every time i wear it, i get nothing but great compliments!	5	1	0	General Petite	Bottoms	Pants
4	4	847	47	Flattering shirt	This shirt is very flattering to all due to the adjustable front tie. it is the perfect length to wear with leggings and it	5	1	6	General	Tops	Blouses

Exploration of the Data

Datatype Information

In [38]:
```
#info function
data_full.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 23486 entries, 0 to 23485
Data columns (total 11 columns):
 #   Column                   Non-Null Count  Dtype 
---  ------                   --------------  ----- 
 0   Unnamed: 0               23486 non-null  int64 
 1   Clothing ID              23486 non-null  int64 
 2   Age                      23486 non-null  int64 
 3   Title                    19676 non-null  object
 4   Review Text              22641 non-null  object
 5   Rating                   23486 non-null  int64 
 6   Recommended IND          23486 non-null  int64 
 7   Positive Feedback Count  23486 non-null  int64 
 8   Division Name            23472 non-null  object
 9   Department Name          23472 non-null  object
 10  Class Name               23472 non-null  object
dtypes: int64(6), object(5)
memory usage: 2.0+ MB
```

Dataset Description

In [39]:
```
#description
data_full.describe()
```

Out[39]:

	Unnamed: 0	Clothing ID	Age	Rating	Recommended IND	Positive Feedback Count
count	23486.000000	23486.000000	23486.000000	23486.000000	23486.000000	23486.000000
mean	11742.500000	918.118709	43.198544	4.196032	0.822362	2.535936
std	6779.968547	203.298980	12.279544	1.110031	0.382216	5.702202
min	0.000000	0.000000	18.000000	1.000000	0.000000	0.000000
25%	5871.250000	861.000000	34.000000	4.000000	1.000000	0.000000
50%	11742.500000	936.000000	41.000000	5.000000	1.000000	1.000000
75%	17613.750000	1078.000000	52.000000	5.000000	1.000000	3.000000
max	23485.000000	1205.000000	99.000000	5.000000	1.000000	122.000000

Dataset Size Exploration

In [40]:
```
#Size explore
print(" Size of the dataset: ",data_full.shape)
```

```
 Size of the dataset:  (23486, 11)
```

Missing Informations

In [41]:
```
data_full.isna().sum()
```

Out[41]:
```
Unnamed: 0                  0
Clothing ID                 0
Age                         0
Title                    3810
Review Text               845
Rating                      0
Recommended IND             0
Positive Feedback Count     0
Division Name              14
Department Name            14
Class Name                 14
dtype: int64
```

Understanding findings

With respect to the sections on Datatype information, Dataset Size, and Missing informations and with a general idea on a customer's tendency to review we can conclude a few things

1. Many a times a customer will just review with the rating and no physical words excanged.
2. A lot larger times a customer doesn't give a title to their review
3. Some products aren't listed to segmentation by class, these are just off the shelf products for an online store, often marketed as products from new sellers/recent startups
4. Some sellers do not mark their product to specific department or classes, owning to the factor that to put a product to a sepcific section there must be a few guidelines that need to be followed as set by the E-commerce website controlling the platform.

Thus we need to

1. Set a new review field where we merge the review title with the total review so that we don't have to
   - handle extra fields
   - manage missing titles with the present review text.
2. Get rid of, or in other words filter out essential fields and records that has valuable data for us to work on.

According to our study the fields we consider important are

1. Clothing ID
2. Review - Title and Review Text to be merged into one
3. Rating
4. Recommended IND - to be renamed to just "**Recomended**" for ease of understanding and use.
5. Positive Feedback Count

Since of all the important fields selected here only the review sections has infractions as well as being the star of the show we will only consider those data records where we get a review others may not be a candidate to this current experiment.

Filtered Data

In [42]:
```
d_set_Filtered=pd.DataFrame(columns=["Clothing ID","Review","Rating","Recommended","Positive Feedback Cou
d_set_Filtered["Clothing ID"]=data_full["Clothing ID"]
d_set_Filtered["Review"]=data_full["Title"].fillna("")+" "+data_full["Review Text"].dropna()
d_set_Filtered["Rating"]=data_full["Rating"]
d_set_Filtered["Recommended"]=data_full["Recommended IND"]
d_set_Filtered["Positive Feedback Count"]=data_full["Positive Feedback Count"]
d_set_Filtered.head()
```

Out[42]:

	Clothing ID	Review	Rating	Recommended	Positive Feedback Count
0	767	Absolutely wonderful - silky and sexy and comfortable	4	1	0
1	1080	Love this dress! it's sooo pretty. i happened to find it in a store, and i'm glad i did bc i never would have ordered it online bc it's petite. i bought a petite and am 5'8". i love the length on me- hits just a little below the knee. would definitely be a true midi on someone who is truly petite.	5	1	4
2	1077	Some major design flaws I had such high hopes for this dress and really wanted it to work for me. i initially ordered the petite small (my usual size) but i found this to be outrageously small. so small in fact that i could not zip it up! i reordered it in petite medium, which was just ok. overall, the top half was comfortable and fit nicely, but the bottom half had a very tight under layer and several somewhat cheap (net) over layers. imo, a major design flaw was the net over layer sewn directly into the zipper - it c	3	0	0
3	1049	My favorite buy! I love, love, love this jumpsuit. it's fun, flirty, and fabulous! every time i wear it, i get nothing but great compliments!	5	1	0
4	847	Flattering shirt This shirt is very flattering to all due to the adjustable front tie. it is the perfect length to wear with leggings and it is sleeveless so it pairs well with any cardigan. love this shirt!!!	5	1	6

Missing Information Reassessment

Availability

In [43]:
```
# finding NANs
d_set_Filtered.isna().sum()
```

Out[43]:
```
Clothing ID                0
Review                   845
Rating                     0
Recommended                0
Positive Feedback Count    0
dtype: int64
```

Observation : we have the same amount of missing review texts as was in the _"**data*full***" dataframe

Removal

In [45]:
```
# Removing NaN from Reviews
d_set_Filtered.dropna(subset=['Review'],inplace=True)
```

Checking

Missings

In [46]:
```
# Rechecking NAN
d_set_Filtered.isna().sum()
```

Out[46]:
```
Clothing ID                0
Review                     0
Rating                     0
Recommended                0
Positive Feedback Count    0
dtype: int64
```

Datatype Finals

In [47]:
```
#field information
d_set_Filtered.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 22641 entries, 0 to 23485
Data columns (total 5 columns):
 #   Column                   Non-Null Count  Dtype 
---  ------                   --------------  ----- 
 0   Clothing ID              22641 non-null  int64 
 1   Review                   22641 non-null  object
 2   Rating                   22641 non-null  int64 
 3   Recommended              22641 non-null  int64 
 4   Positive Feedback Count  22641 non-null  int64 
dtypes: int64(4), object(1)
memory usage: 1.0+ MB
```

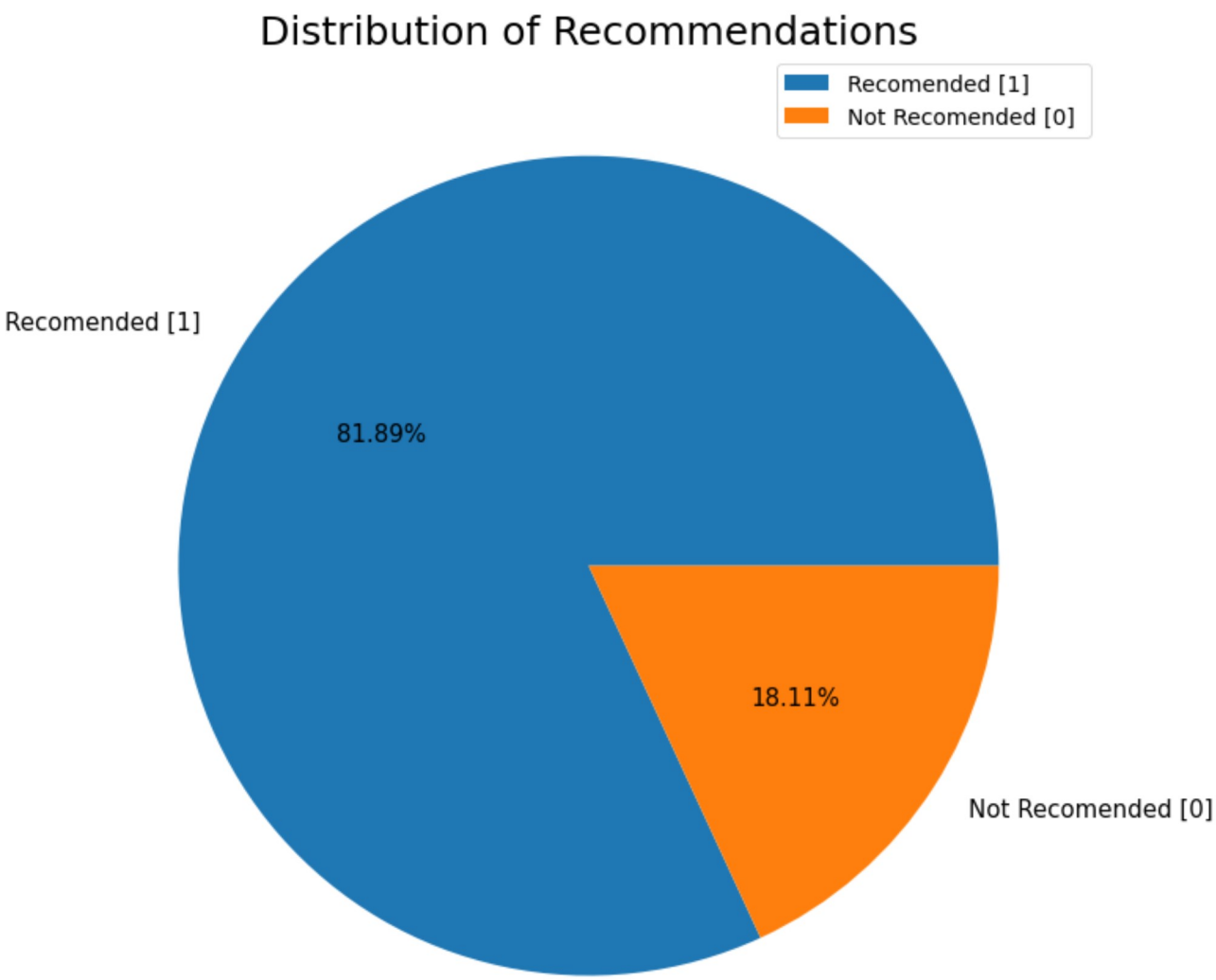
Size

In [48]:
```
#Size
print(" Size of the filtered: ",d_set_Filtered.shape)
```

```
 Size of the filtered:  (22641, 5)
```

Distribution of Recommendations

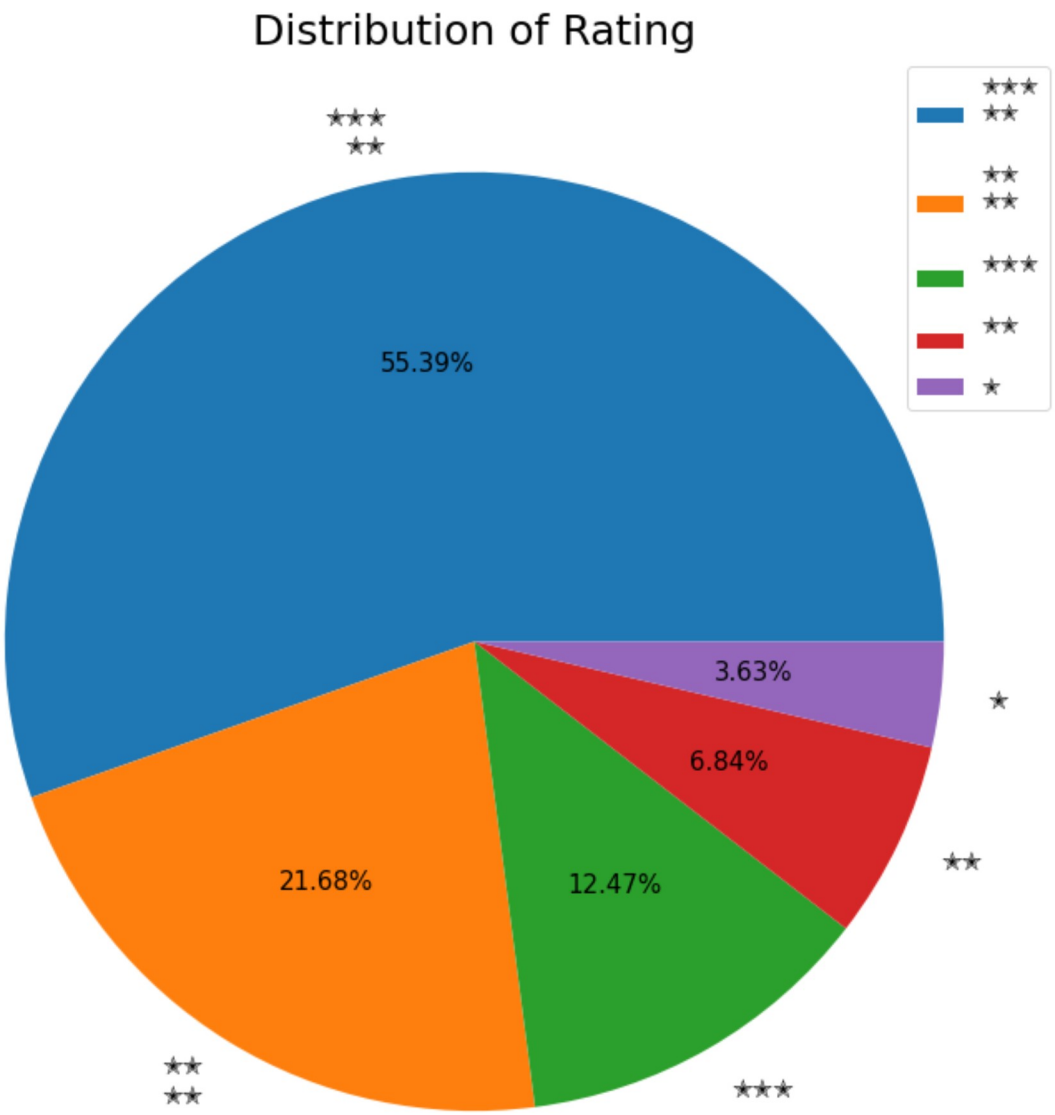
In [75]:
```
# Recommendation label distribution
percs=d_set_Filtered.Recommended.value_counts()*100/len(d_set_Filtered.Recommended)
idfs=d_set_Filtered.Recommended.value_counts().index.values
#print(idfs)
fig1=plt.figure(figsize=(13,12));ax1=fig1.add_subplot(111)
ax1.pie(percs,labels=['Recomended [1] ','Not Recomended [0] '],
        autopct='%1.2f%%', textprops={'fontsize': 15});
ax1.legend(['Recomended [1] ','Not Recomended [0] '],fontsize=14);
plt.title("Distribution of Recommendations",fontdict = {'fontsize' : 25})
plt.show()
```



Thus we have approximately a $80\%:20\%$ distribution in ***Recommendation*** to ***Non Recommendation*** items in the products listing something we shouldn't right now need to be too concerned with.

Distributions of Ratings

In [74]:
```
# Ratings label distribution
percs=d_set_Filtered.Rating.value_counts()*100/len(d_set_Filtered.Rating)
idfs=d_set_Filtered.Rating.value_counts().index.values
#print(idfs)
fig2=plt.figure(figsize=(14,13));ax2=fig2.add_subplot(111)
ax1.pie(percs,labels=[f'\u272D\u272D\u272D\n\u272D\u272D',f'\u272D\u272D\n\u272D\u272D',f'\u272D\u272D\u2
        ,autopct='%1.2f%%', textprops={'fontsize': 15});
ax1.legend([f'\u272D\u272D\u272D\n\u272D\u272D\n',f'\u272D\u272D\n\u272D\u272D\n',f'\u272D\u272D\u272D\n'
plt.title("Distribution of Rating",fontdict = {'fontsize' : 25})
plt.show()
```



visible from this plot is that around more than 50% of the the ratings is in the 5☆ category, around approximately half of the remaining half is in the 4☆ category (22%). Hence the 5☆ and 4☆ make up 75% of the data available to us. Thus making it apparent that either our customer base is quite linient or the products of this E-commerce (Amazon as per the data source) website is quite satisfactory and meets the needs of the customer base(Women as per the data source)

Word Clouds

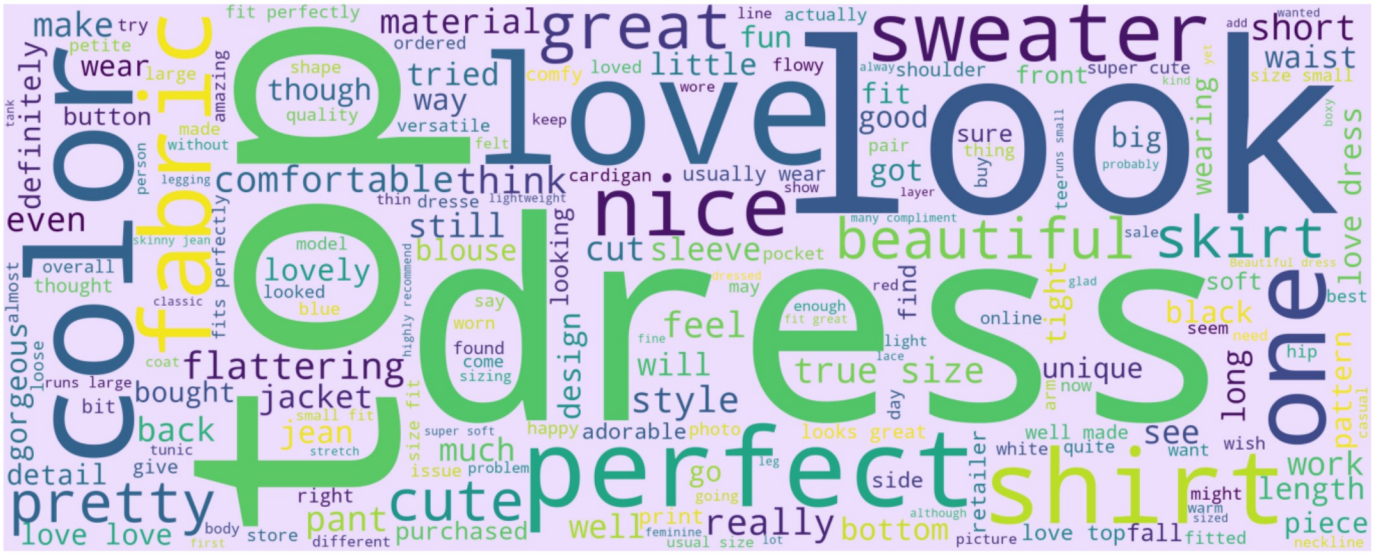
Words that Recommends

In [76]:

```
# Recommended Products Reviews
temp_df = d_set_Filtered[d_set_Filtered.Recommended==1]
words = " ".join(temp_df.Review)
review_words = " ".join([w for w in words.split()
                                if 'http' not in w
                                  and not w.startswith('@')
                                  and w!='RT'])

wrdcld = WordCloud(stopwords=STOPWORDS,
                   background_color='#EEDEFD',
                   width=2500,
                   height=1000).generate(review_words)
fig3=plt.figure(figsize=(25,10));ax3=fig3.add_subplot(111);
plt.imshow(wrdcld)
plt.axis('off')
plt.title(" Words in Recommended Product Reviews ",fontdict = {'fontsize' : 25})
plt.show()
```

Words in Recommended Product Reviews



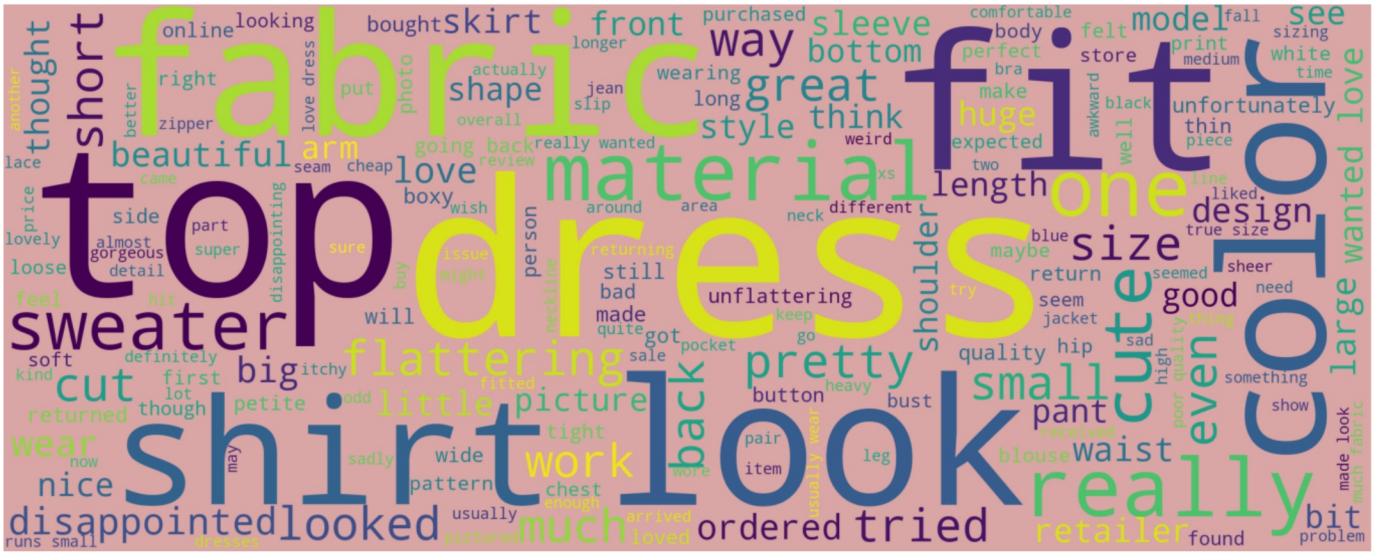
Words that Recommends Against

In [77]:

```
# Not-Recommended Products Reviews
temp_df = d_set_Filtered[d_set_Filtered.Recommended==0]
words = " ".join(temp_df.Review)
review_words = " ".join([w for w in words.split()
                                if 'http' not in w
                                  and not w.startswith('@')
                                  and w!='RT'])

wrdcld = WordCloud(stopwords=STOPWORDS,
                   background_color='#d9a6a6',
                   width=2500,
                   height=1000).generate(review_words)
fig4=plt.figure(figsize=(25,10));ax4=fig4.add_subplot(111);
plt.imshow(wrdcld)
plt.axis('off')
plt.title(" Words in Non-Recommended Product Reviews ",fontdict = {'fontsize' : 25})
plt.show()
```

Words in Non-Recommended Product Reviews



the words as visible to us, it is noticed that there are many words depicting that these are Women's apparels (according to the data source). Most words describe that we are dealing with with Women's Clothing rather than pointing good or bad response to the deal. Pointer words that describe if a product is good or bad is quite low in population, this may disturb the learning process of machine learning algorithms.

Data Preparation

Presets

In [18]:
```
stops_EN=set(stopwords.words('english'))
```

Cleaning Review Texts

In [19]:
```
d_set_Filtered.insert(2,"Cleaned Review",range(d_set_Filtered.shape[0]))
for idx in d_set_Filtered.index:
  Review=d_set_Filtered.Review[idx]
  alph_0=re.sub("[^a-zA-Z]"," ",Review)
  words=alph_0.lower().split();
  cleaned=[word for word in words if word not in stops_EN]
  d_set_Filtered['Cleaned Review'][idx]=' '.join(cleaned)
```

Result of Cleaning

In [20]:
```
d_set_Filtered.head()
```

Out[20]:

	Clothing ID	Review	Cleaned Review	Rating	Recommended	Positive Feedback Count
0	767	Absolutely wonderful - silky and sexy and comfortable	absolutely wonderful silky sexy comfortable	4	1	0
1	1080	Love this dress! it's sooo pretty. i happened to find it in a store, and i'm glad i did bc i never would have ordered it online bc it's petite. i bought a petite and am 5'8". i love the length on me- hits just a little below the knee. would definitely be a true midi on someone who is truly petite.	love dress sooo pretty happened find store glad bc never would ordered online bc petite bought petite love length hits little knee would definitely true midi someone truly petite	5	1	4
2	1077	Some major design flaws I had such high hopes for this dress and really wanted it to work for me. i initially ordered the petite small (my usual size) but i found this to be outrageously small. so small in fact that i could not zip it up! i reordered it in petite medium, which was just ok. overall, the top half was comfortable and fit nicely, but the bottom half had a very tight under layer and several somewhat cheap (net) over layers. imo, a major design flaw was the net over layer sewn directly into the zipper - it c	major design flaws high hopes dress really wanted work initially ordered petite small usual size found outrageously small small fact could zip reordered petite medium ok overall top half comfortable fit nicely bottom half tight layer several somewhat cheap net layers imo major design flaw net layer sewn directly zipper c	3	0	0
3	1049	My favorite buy! I love, love, love this jumpsuit. it's fun, flirty, and fabulous! every time i wear it, i get nothing but great compliments!	favorite buy love love love jumpsuit fun flirty fabulous every time wear get nothing great compliments	5	1	0

Clothing ID	Review	Cleaned Review	Rating	Recommended	Positive Feedback Count
	Flattering shirt This shirt is very flattering to all due to	flattering shirt shirt flattering due			

Model Creation

Dataset Fixation

Abscissa and Ordinate

In [21]:
```
# Abscissa and Ordinates
abscissa_primary=d_set_Filtered[['Clothing ID','Cleaned Review','Recommended','Positive Feedback Count']]
ordinate=d_set_Filtered['Rating']
```

Vectorizer

In [22]:
```
# Vectorizer object
vector=CountVectorizer(analyzer = "word",tokenizer = None,preprocessor = None,stop_words = None,max_featu
```

Review Vectorization

In [23]:
```
ReviewFeatures=vector.fit_transform(abscissa_primary['Cleaned Review'])
VectoredReview=pd.DataFrame(ReviewFeatures.toarray())
```

Splitter

In [24]:
```
# splitting
abscissa=deepcopy(abscissa_primary);abscissa.join(VectoredReview);abscissa.drop(labels="Cleaned Review",i
trainer_X,tester_X,trainer_Y,tester_Y=train_test_split(abscissa, ordinate, test_size=0.2)
```

Classification Models

Our task in this project is to identify ***the Rating of a product***, hence we need to classify one review from another or in other words we have a problem of classifying reviews into either of the five groups. Classification can be performed on structured or unstructured data. Classification is a technique where we categorize data into a given number of classes. The main goal of a classification problem is to identify the category/class to which a new data will fall under. As per the terms of Classification there are many algorithms but for logicality and feasibility of this project work we will employ $``7"$ of the most common Classification Algorithms.

1. *Logistic Regression*
   Definition: Logistic regression is a machine learning algorithm for classification. In this algorithm, the probabilities describing the possible outcomes of a single trial are modelled using a logistic function.
   Advantages: Logistic regression is designed for this purpose (classification), and is most useful for understanding the influence of several independent variables on a single outcome variable.
   Disadvantages: Works only when the predicted variable is binary, assumes all predictors are independent of each other and assumes data is free of missing values.

2. *Naïve Bayes*
   Definition: Naive Bayes algorithm based on Bayes' theorem with the assumption of independence between every pair of features. Naive Bayes classifiers work well in many real-world situations such as document classification and spam filtering.
   Advantages: This algorithm requires a small amount of training data to estimate the necessary parameters. Naive Bayes classifiers are extremely fast compared to more sophisticated methods.
   Disadvantages: Naive Bayes is is known to be a bad estimator.

3. *Stochastic Gradient Descent*
   Definition: Stochastic gradient descent is a simple and very efficient approach to fit linear models. It is particularly useful when the number of samples is very large. It supports different loss functions and penalties for classification.
   Advantages: Efficiency and ease of implementation.
   Disadvantages: Requires a number of hyper-parameters and it is sensitive to feature scaling.

4. *K-Nearest Neighbours*
   Definition: Neighbours based classification is a type of lazy learning as it does not attempt to construct a general internal

model, but simply stores instances of the training data. Classification is computed from a simple majority vote of the k nearest neighbours of each point.
Advantages: This algorithm is simple to implement, robust to noisy training data, and effective if training data is large.
Disadvantages: Need to determine the value of K and the computation cost is high as it needs to compute the distance of each instance to all the training samples.

5. *Decision Tree*
Definition: Given a data of attributes together with its classes, a decision tree produces a sequence of rules that can be used to classify the data.
Advantages: Decision Tree is simple to understand and visualise, requires little data preparation, and can handle both numerical and categorical data.
Disadvantages: Decision tree can create complex trees that do not generalise well, and decision trees can be unstable because small variations in the data might result in a completely different tree being generated.

6. *Random Forest*
Definition: Random forest classifier is a meta-estimator that fits a number of decision trees on various sub-samples of datasets and uses average to improve the predictive accuracy of the model and controls over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement.
Advantages: Reduction in over-fitting and random forest classifier is more accurate than decision trees in most cases.
Disadvantages: Slow real time prediction, difficult to implement, and complex algorithm.

7. *Support Vector Machine*
Definition: Support vector machine is a representation of the training data as points in space separated into categories by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.
Advantages: Effective in high dimensional spaces and uses a subset of training points in the decision function so it is also memory efficient.
Disadvantages: The algorithm does not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation

Model Class Imports (Multiple Models)

In [25]:
```
from sklearn.linear_model import LogisticRegression as LGR
from sklearn.naive_bayes import MultinomialNB as NBS
from sklearn.linear_model import SGDClassifier as SGD
from sklearn.neighbors import KNeighborsClassifier as KNN
from sklearn.tree import DecisionTreeClassifier as DTC
from sklearn.ensemble import RandomForestClassifier as RFC
from sklearn.svm import SVC
```

Model Creation (Multiple Models)

In [26]:
```
#Model Creation (Multiple models)
model_list={'LGR':'Logistic Regression',
            'NBS':'Naïve Bayes',
            'SGD':'Stochastic Gradient Descent',
            'KNN':'K-Nearest Neighbours',
            'DTC':'Decision Tree',
            'RFC':'Random Forest',
            'SVM':'Support Vector Machine'}

print('Creating models for : \n')
for key in model_list: print('\t',model_list[key].ljust(30,' '),' : ',key,sep='  ');
print('_'.center(80,'_'));
models={'LGR':LGR(solver='lbfgs', max_iter=10000),
        'NBS':NBS(),
        'SGD':SGD(),
        'KNN':KNN(n_neighbors = 5),
        'DTC':DTC(),
        'RFC':RFC(n_estimators=200),
        'SVM':SVC(kernel='linear',C=1.0)};
```

```
Creating models for :

         Logistic Regression              :   LGR
         Naïve Bayes                      :   NBS
         Stochastic Gradient Descent      :   SGD
         K-Nearest Neighbours             :   KNN
         Decision Tree                    :   DTC
         Random Forest                    :   RFC
         Support Vector Machine           :   SVM
```

Model Training (Multiple Models)

In [27]:
```
#Training Data on Models (Multiple models)
print(' Model Training Status : ')
for key in models:
    tic=time.perf_counter();
    models[key].fit(trainer_X,trainer_Y)
    toc=time.perf_counter()
    print('\tTrained Model : ',key,f"\t time taken: {toc - tic:0.4f} seconds");
print('_'.center(80,'_'));
```

```
 Model Training Status :
        Trained Model :  LGR     time taken: 31.2147 seconds
        Trained Model :  NBS     time taken: 0.0082 seconds
        Trained Model :  SGD     time taken: 1.2039 seconds
        Trained Model :  KNN     time taken: 0.0171 seconds
        Trained Model :  DTC     time taken: 0.0287 seconds
        Trained Model :  RFC     time taken: 3.1460 seconds
        Trained Model :  SVM     time taken: 219.5525 seconds
```

Model Testing (Multiple Models)

Generating Predictions for Tests

In [28]:
```
#Prediction tests from the Trained Models
predicts=dict.fromkeys(models.keys())
print(' Prediction Generation Status : ')
for key in models:
    tic=time.perf_counter()
    predicts[key]=models[key].predict(tester_X)
    toc=time.perf_counter()
    print('Prediction generated for : ',key,f"\t time taken: {toc - tic:0.4f} seconds");
print('_'.center(80,'_'));
```

```
 Prediction Generation Status :
Prediction generated for :  LGR          time taken: 0.0025 seconds
Prediction generated for :  NBS          time taken: 0.0014 seconds
Prediction generated for :  SGD          time taken: 0.0013 seconds
Prediction generated for :  KNN          time taken: 0.1650 seconds
Prediction generated for :  DTC          time taken: 0.0030 seconds
Prediction generated for :  RFC          time taken: 0.2295 seconds
Prediction generated for :  SVM          time taken: 0.7028 seconds
```

Studying Accuracy and Reports of the Model

In [29]:
```
accr=dict.fromkeys(models.keys())
for key in models:
    accr[key]=100*accuracy_score(predicts[key],tester_Y)
    print('Metrics for : ',model_list[key])
    print('Accuracy : {:10.4f}%'.format(accr[key]))
    print(classification_report(predicts[key],tester_Y))
    print('_'.center(80,'_'));print('\n\n');
print('*'.center(80,'-'));
```

```
Metrics for :  Logistic Regression
Accuracy :    61.9784%
              precision    recall  f1-score   support

           1       0.00      0.00      0.00         0
           2       0.32      0.33      0.32       284
           3       0.35      0.39      0.37       507
           4       0.00      0.00      0.00         0
           5       1.00      0.67      0.80      3738

    accuracy                           0.62      4529
   macro avg       0.33      0.28      0.30      4529
weighted avg       0.88      0.62      0.72      4529
```

```
Metrics for :  Naïve Bayes
Accuracy :    52.9477%
              precision    recall  f1-score   support

           1       0.03      0.19      0.05        26
           2       0.04      0.24      0.07        45
           3       0.07      0.13      0.09       300
```

4	0.00	0.00	0.00	0
5	0.93	0.56	0.70	4158
accuracy			0.53	4529
macro avg	0.21	0.23	0.18	4529
weighted avg	0.86	0.53	0.65	4529

Metrics for : Stochastic Gradient Descent
Accuracy : 55.6414%

	precision	recall	f1-score	support
1	0.00	0.00	0.00	1
2	0.00	0.00	0.00	0
3	0.00	1.00	0.01	2
4	0.00	0.00	0.00	0
5	1.00	0.56	0.71	4526
accuracy			0.56	4529
macro avg	0.20	0.31	0.14	4529
weighted avg	1.00	0.56	0.71	4529

Metrics for : K-Nearest Neighbours
Accuracy : 50.8059%

	precision	recall	f1-score	support
1	0.19	0.22	0.20	144
2	0.28	0.33	0.31	248
3	0.17	0.29	0.21	339
4	0.24	0.26	0.25	924
5	0.74	0.64	0.69	2874
accuracy			0.51	4529
macro avg	0.32	0.35	0.33	4529
weighted avg	0.55	0.51	0.53	4529

Metrics for : Decision Tree
Accuracy : 56.6129%

	precision	recall	f1-score	support
1	0.18	0.19	0.19	155
2	0.38	0.36	0.37	304
3	0.25	0.33	0.28	431
4	0.13	0.28	0.18	443
5	0.86	0.67	0.75	3196
accuracy			0.57	4529
macro avg	0.36	0.37	0.35	4529
weighted avg	0.67	0.57	0.61	4529

Metrics for : Random Forest
Accuracy : 58.2248%

	precision	recall	f1-score	support
1	0.13	0.23	0.17	96
2	0.34	0.35	0.35	284
3	0.30	0.38	0.33	456
4	0.09	0.26	0.13	334
5	0.90	0.67	0.77	3359
accuracy			0.58	4529
macro avg	0.35	0.38	0.35	4529
weighted avg	0.73	0.58	0.64	4529

Metrics for : Support Vector Machine
Accuracy : 62.7070%

	precision	recall	f1-score	support
1	0.00	0.00	0.00	0
2	0.00	0.00	0.00	0
3	0.56	0.41	0.48	791

```
              4       0.00      0.00      0.00         0
              5       1.00      0.67      0.80      3738

       accuracy                           0.63      4529
      macro avg       0.31      0.22      0.26      4529
   weighted avg       0.92      0.63      0.75      4529
```

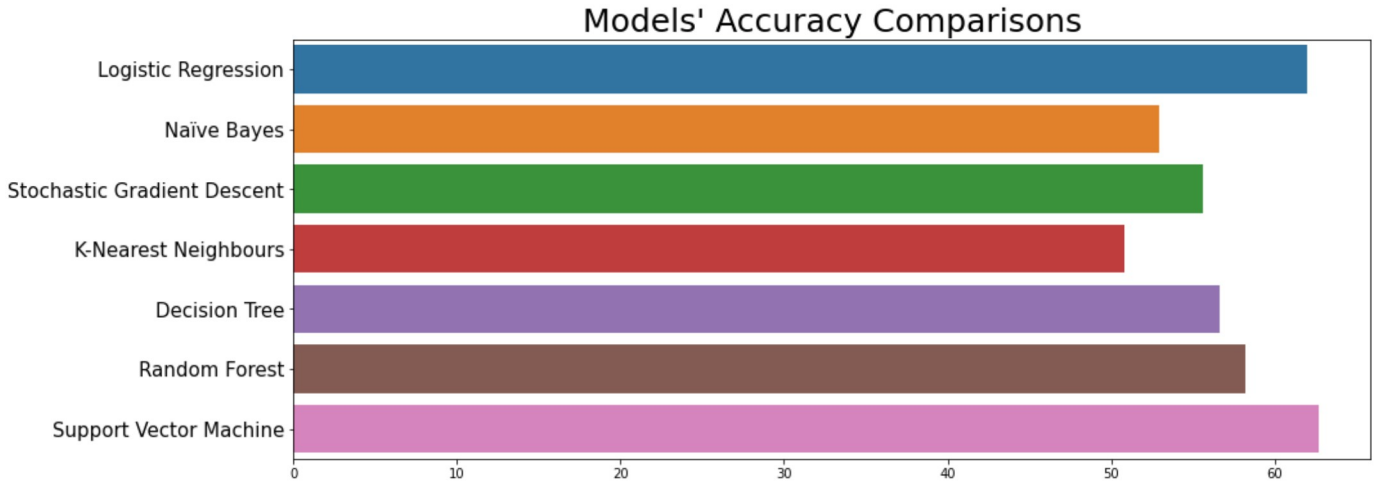
Comparative Study

As per the status we stand we have trained and obtained metics of 7 classification machine learning models for removing hate-speech. Now, we need to make a comparison on which model is more feasible to handle the problem of hate-speech

Accuracy Comparisons

In [30]:

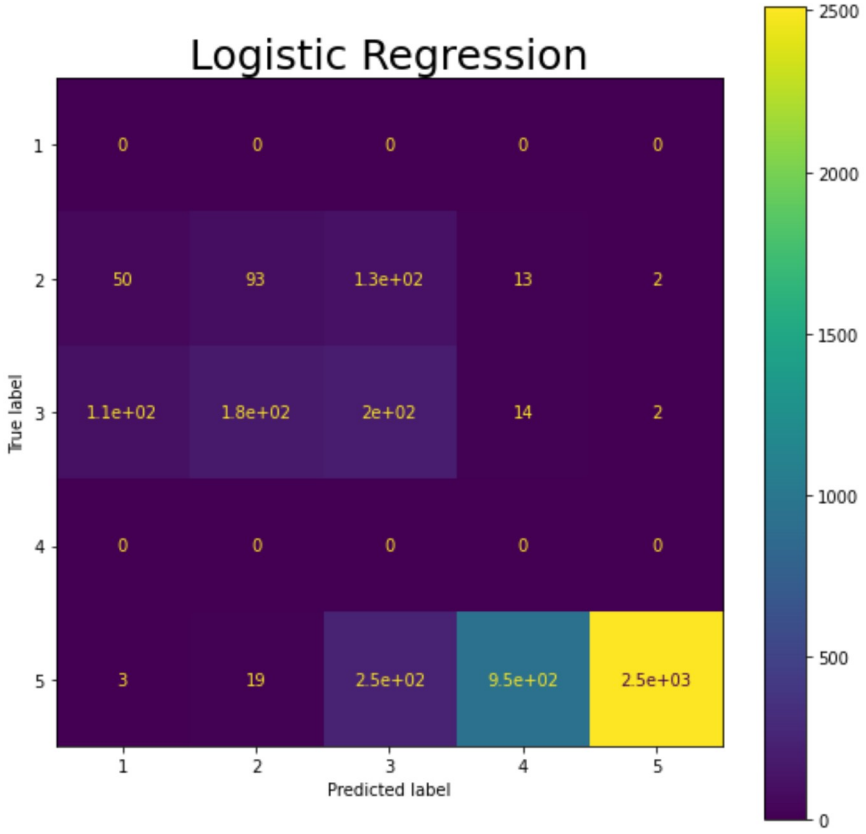
```
#Comparison of Accuracies of Models
fig4=plt.figure(figsize=(15,6));ax4=fig4.add_subplot(111)
sns.barplot(list(accr.values()),list(range(len(accr))),orient='h');
plt.yticks(range(len(accr)), list(model_list.values()),rotation=0,ha='right',fontsize=15)
plt.title(" Models' Accuracy Comparisons ",fontdict = {'fontsize' : 25});
plt.show()
```

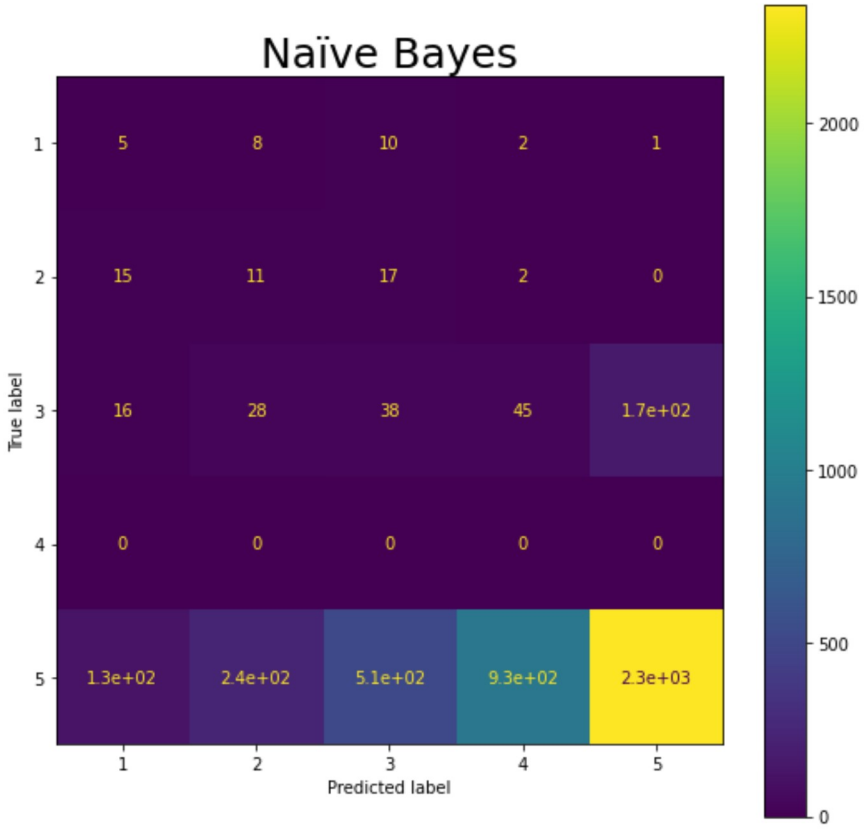


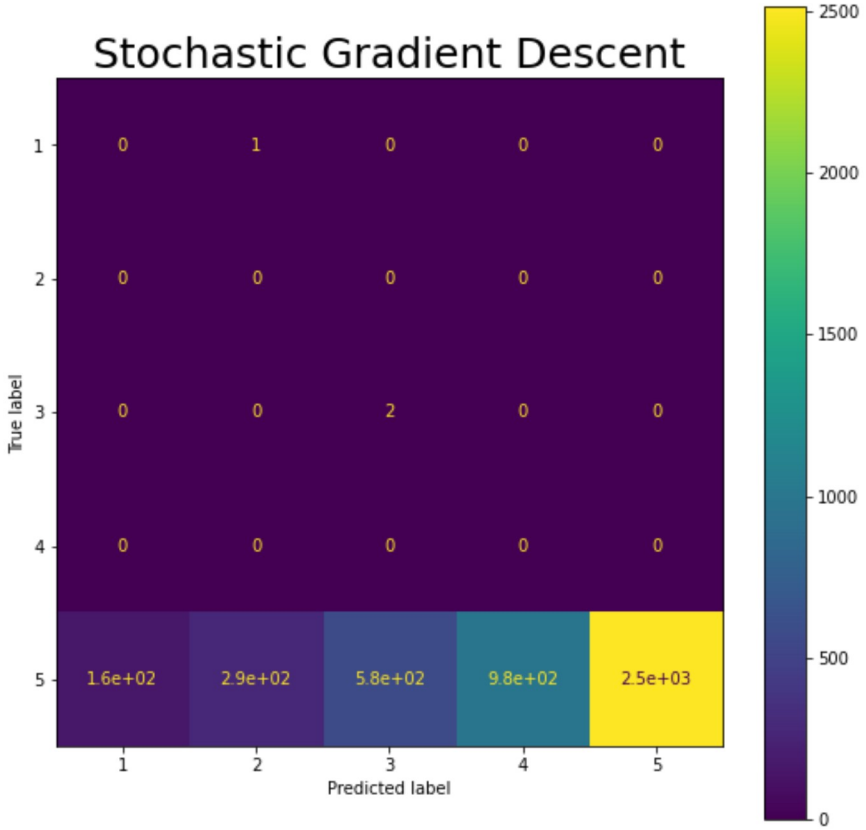
Confusion Matrices

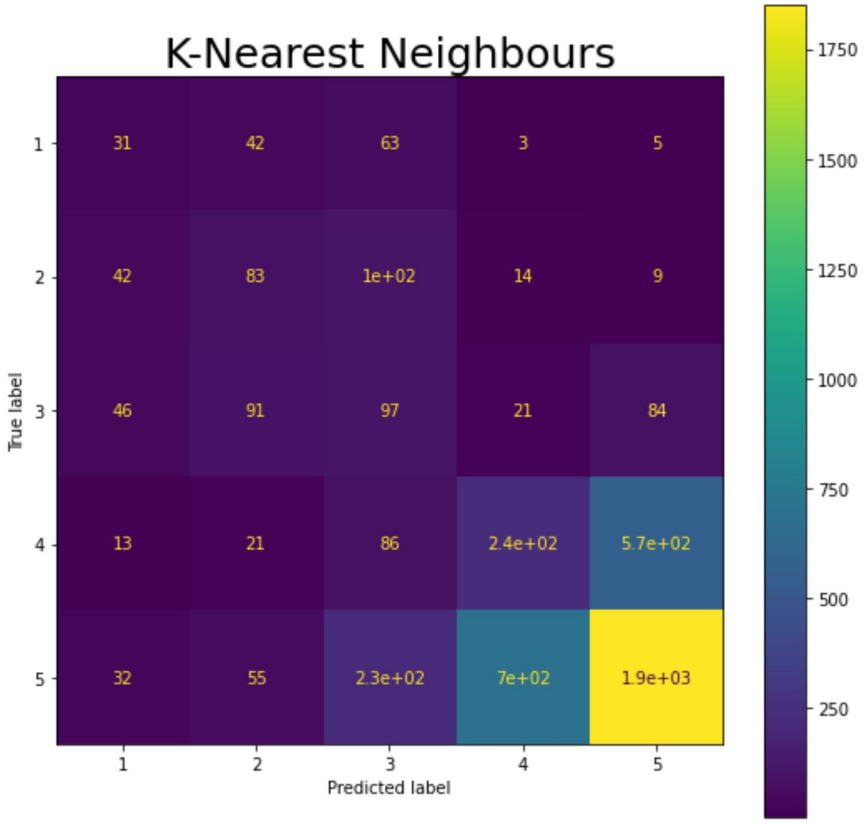
In [33]:

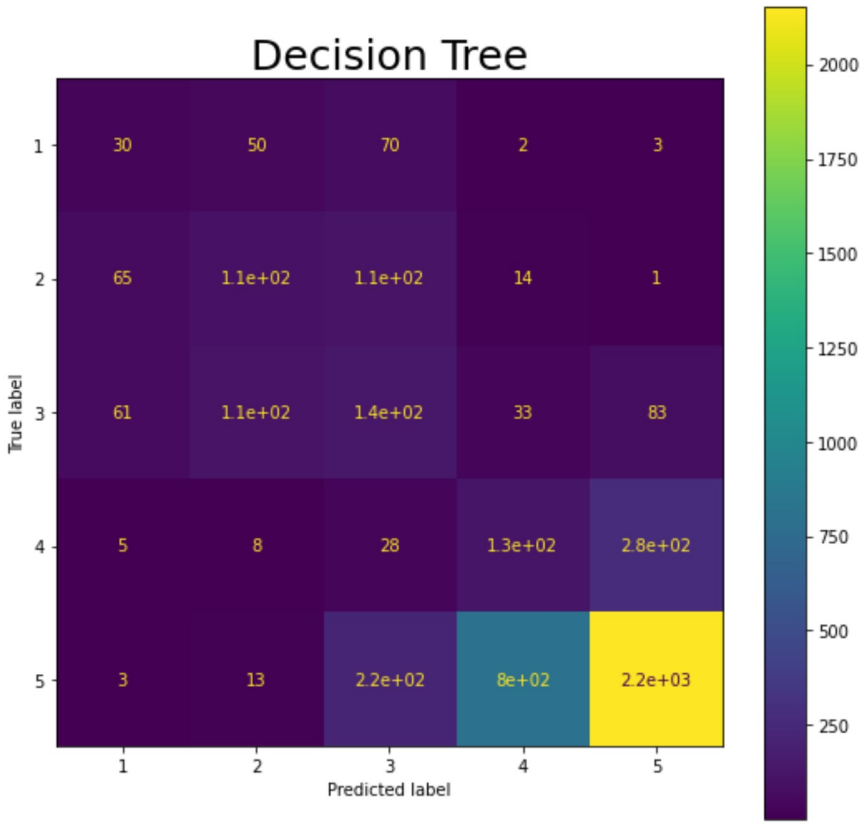
```
# Confusion Matrices for the models
for key in models:
  fig5=plt.figure(figsize=(9,9));ax5=fig5.add_subplot(111);
  confMat=confusion_matrix(predicts[key],tester_Y)
  viewer=ConfusionMatrixDisplay(confMat,display_labels=[1,2,3,4,5]);
  viewer.plot(ax=ax5);
  plt.title(model_list[key],fontdict = {'fontsize' : 25});
  plt.show()
```

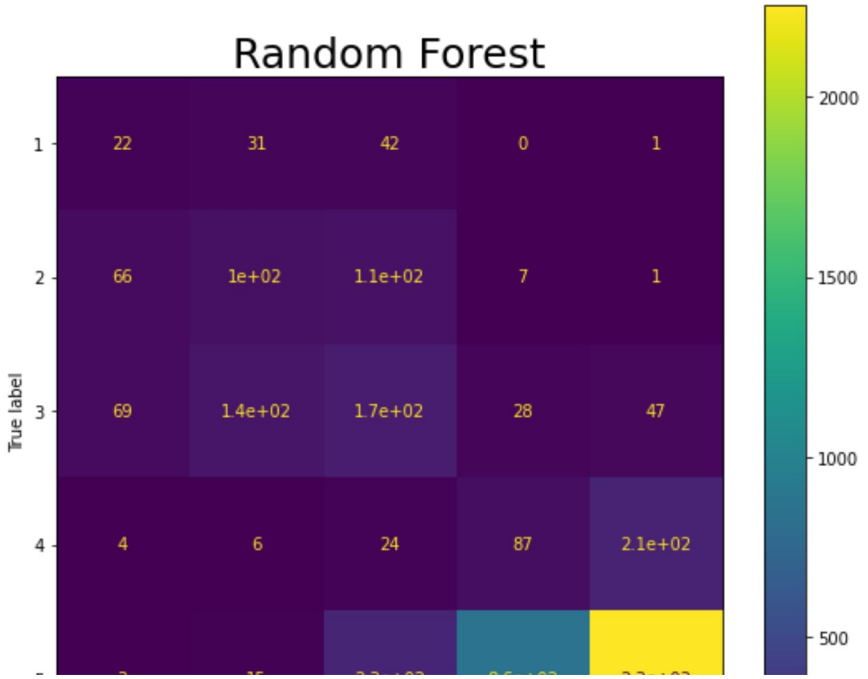

Logistic Regression
0 0 0 0 0
50 93 1.3e+02 13 2
1.1e+02 1.8e+02 2e+02 14 2
0 0 0 0 0
3 19 2.5e+02 9.5e+02 2.5e+03
True label
Predicted label
1 2 3 4 5
2500 2000 1500 1000 500 0


Naïve Bayes
5 8 10 2 1
15 11 17 2 0
16 28 38 45 1.7e+02
0 0 0 0 0
1.3e+02 2.4e+02 5.1e+02 9.3e+02 2.3e+03
True label
Predicted label
1 2 3 4 5
2000 1500 1000 500 0


Stochastic Gradient Descent
0 1 0 0 0
0 0 0 0 0
0 0 2 0 0
0 0 0 0 0
1.6e+02 2.9e+02 5.8e+02 9.8e+02 2.5e+03
True label
Predicted label
1 2 3 4 5
2500 2000 1500 1000 500 0


K-Nearest Neighbours
31 42 63 3 5
42 83 1e+02 14 9
46 91 97 21 84
13 21 86 2.4e+02 5.7e+02
32 55 2.3e+02 7e+02 1.9e+03
True label
Predicted label
1 2 3 4 5
1750 1500 1250 1000 750 500 250





from the work accomplished till now we can see that the the models in descending order of their accuracies are

1. The Support Vector Machines at 62.7070%
2. Logistic regression at 61.9784%
3. Random Forest 58.2248%
4. Decision Trees at 56.6129%
5. Stocastic Gradient Descent at 55.6414%
6. Naïve Bayes at 52.9477%
7. K - Nearest Neighbours at 50.8059%

Now, keeping the Logistic Regression as a base model we can see that almost all models performed sub par with respect to as far as accuracy is concerned. It was said in the Model definitions that Naïve Bayes is an underperforming model and so is what we found here also but outperformed the K-Nearest Neighbours.

As per computation time cost the Support Vector is costiliest at a time requirement of 3 minutes and 40 seconds followed by Random Forest at 3 seconds the others took time varying from fractions of a second to just above 1 second to train completely. Conversely the Logistic Regression took a lot comparatively at just above 31 seconds. Thus on superficial terms as related to time consumption none of the models are are too code heavy for the processor and in fact all could be considered almost equally to be time efficient.

If, we now consider the efficiency of a model in terms of how it predicts a particular product over with respect to sales then we can in very simple terms check the diagonals of the **Confusion matrices** we obtained. Disregarding the the calculations of precisions and recalls we can use this diagonal to see how far the accurate model is actually efficient. This is very much visible in the Confusion matrices for the Naïve Bayes and Stocastic Gradient Descent methods, where the model tried to predict the $``Rating=5"$ true label into all the variations of ratings. Comparatively on efficiency terms the K -Nearest Neighbour, Decision Trees and Random Forests seem to be more efficient than the top most accurate model of Support Vector machines.

Now, if given a choice to choose the best model our best bet is for the Random Forest algorithm followed by Support Vector Machines. But at this point we are sceptical and better not place our bets too fast as all models can be improved by Gradient Descent, Cross Validation, Hold-off Validation or Hyper-parameter boosting. We can also go out of our way to select a Deep Learning methods to suffice any shortcomings in conventional methods of machine learning. More improvement could also be done if instead of superficial cleaning of words using the Natural Language Toolkit, we used the Sentiment Intensity Analyser facility of the same toolkit, but as per as the requirements of the project guideline bindings goes we have successfully accomplished ourselves and infact built a benchmark on which further development is possible.

$\dagger$ $NOTE:$ All information and metrics discussed above is subject to change on every execution. The study made here is purely on the basis of a solo complete execution. The results so obtained from that execution cycle is used for study and the conclusions drawn. Any values spoken here may not be taken for face value in future executions but may be set as comparison.

End Notes

After getting success in speech recognition and vision research, natural language processing is the most targeted research area in artificial intelligence.

Although it is started decades ago, most people lack the NLP experience. Because it's hard to teach a machine with the challenges listed below:

1. Sarcasm
2. Ambiguity (Syntactical and Lexical)
3. Syntax
4. Co-reference
5. Typos
6. Normalization
7. Puns

For a machine running on numbers it is behemoth of a task to make it understand which even normal people fail to cope up with sometimes owning to large variations languages among us, the sense in which it is spoken, tone of voice, etc., although for tweets a large amount of this problem is overcome due to the size of the tweets and scanability of tweets still a challege is a challenge and as long as people will be there, there will be communication, there will be sentiments involved in communication.

As far as technology goes an E-Commerce platform will always try to bank on positive reviews for pushing a product to a customer and reviews play a big role in changing the course of the business game of a company especially if it holds an online store .

Thank You

Bibliography

- https://unctad.org/news/global-e-commerce-jumps-267-trillion-covid-19-boosts-online-sales
- https://www.oecd.org/coronavirus/policy-responses/e-commerce-in-the-time-of-covid-19-3a2b78e8/
- https://www.bigcommerce.com/articles/ecommerce/
- https://techtimez.com/business/growth-of-ecommerce/
- https://www.gartner.com/en/marketing/insights/daily-insights/the-rise-of-e-commerce-in-india
- https://searchcio.techtarget.com/definition/e-commerce
- https://developers.google.com/machine-learning/recommendation/overview
- https://www.bigcommerce.com/blog/online-reviews/#who-is-reading-online-reviews
- https://www.e-satisfaction.com/7-reasons-why-customer-reviews-are-important/
- https://reviewing.co.uk/_review.htm
- https://www.dictionary.com/browse/review
- https://www.merriam-webster.com/dictionary/review
- https://towardsdatascience.com/sentiment-analysis-concept-analysis-and-applications-6c94d6f58c17

- https://amueller.github.io/word_cloud/generated/wordcloud.WordCloud.html
- https://www.colorbook.io/colorschemes
- https://imagecolorpicker.com/
- https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html
- https://pandas.pydata.org/

_(b.biswas*94587@ieee.org)*