CLOTH RECOMMENDATION SYSTEM

A PROJECT REPORT

submitted by

Kush Purohit, Priya Patel, Neha Rana
(21BCE238, 21BCE239, 21BCE246)

to

Institute Of Technology, Nirma University

Department of Computer Science and Engineering
Ahmedabad

April 2024

CONTENTS

0.1 DATASET DESCRIPTION

Context of the dataset:

We have selected a fashion dataset openly available on hugging face. The dataset consists of various features that describe a cloth. The clothes mentioned in the dataset are from different brands that have different master, sub-categories, and colors. They are worn in different seasons and were launched in different years. All these along with some more features make this dataset a useful source for building a content-based recommendation system.
Source URL.

Metadata of the dataset :

Number of Columns: 11
Number of rows : 44072

Data Types:

- **id** - int64
- **gender** - object
- **masterCategory** - object
- **subCategory** - object
- **articleType** - object
- **baseColour** - object
- **season** - object

- **year** - float64
- **usage** - object
- **productDisplayName** - object
- **image** - object

Number of Null Values in Each Column:

id 0
gender 0
masterCategory 0
subCategory 0
articleType 0
baseColour 0
season 0
year 0
usage 0
productDisplayName 0
image 0
dtype: int64

The Specific Distribution Graphs :

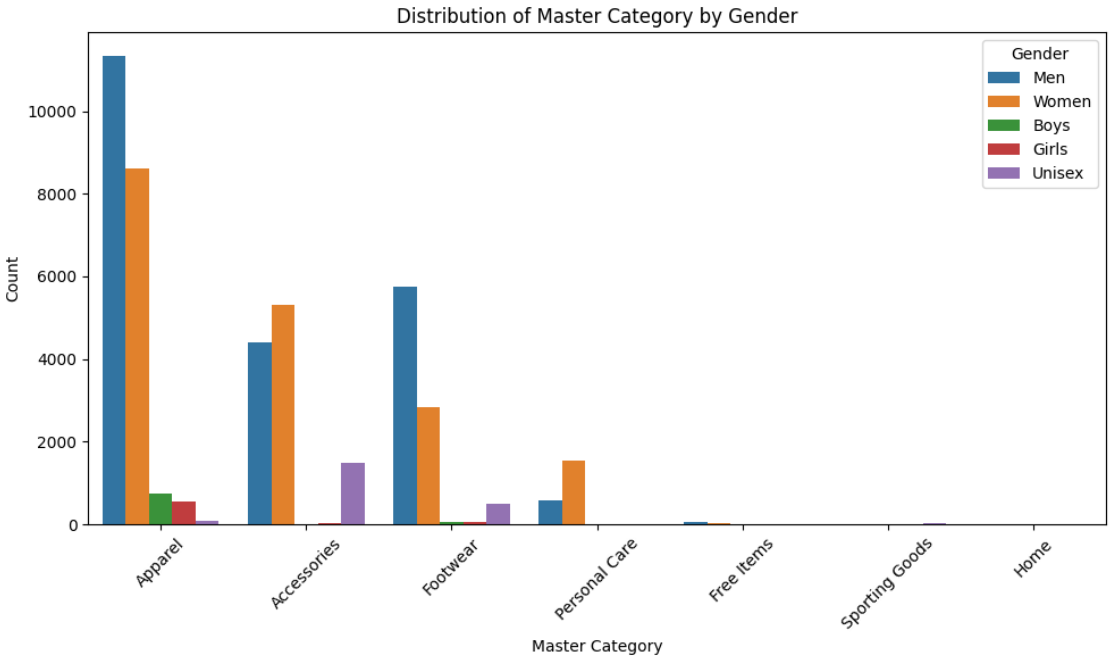

Figure 1: Distribution of Master Category by Gender

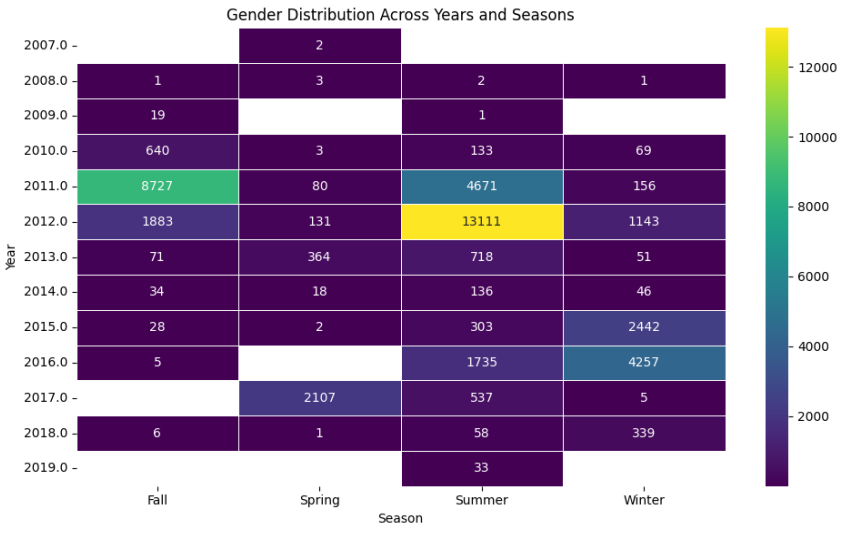

Figure 2: Distribution based on season and year

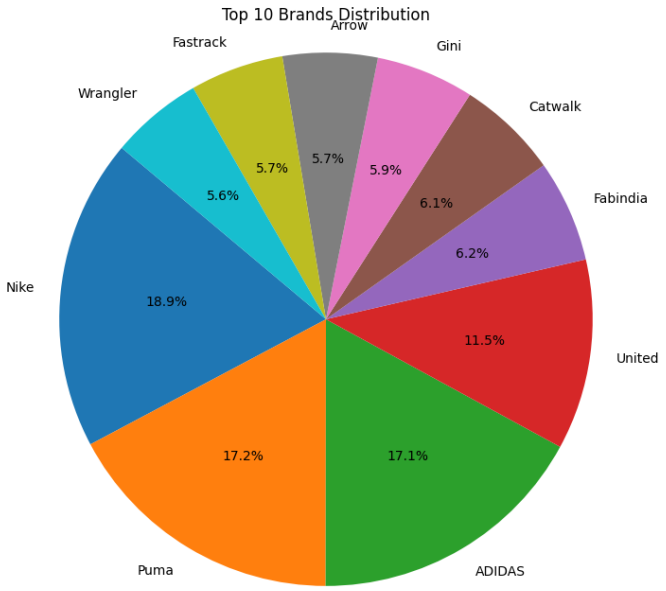

Figure 3: Top 10 Brands Distribution

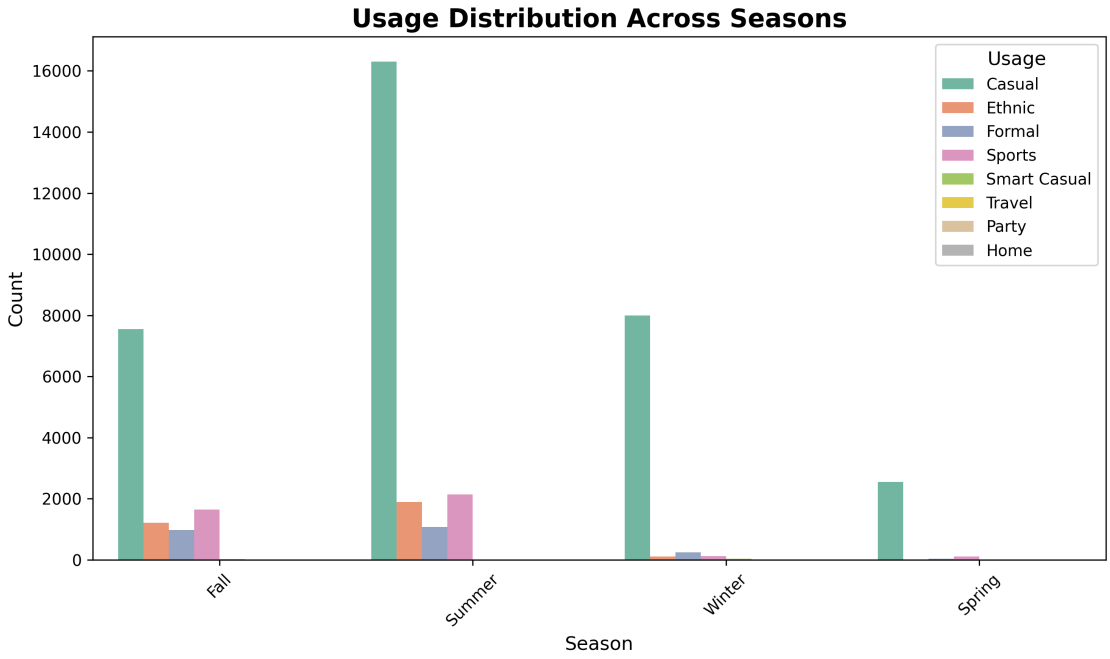

Figure 4: Usage Based On season

0.2 NOVELTY

The recommendation systems are a well-liked and extensively employed method for giving customers customized recommendations. There are different types of recommendation systems algorithms based on data available. The one that we have used in this project is :

Content-Based Filtering Recommendation System - It recommends items similar to those a user has liked or interacted with in the past. It analyzes the attributes or features of items and recommends items with similar characteristics.

We have compared the results of the recommendation made by using two different approaches to vectorizing the features.

This first approach is the simple Bag Of Word (BOW) method. The algorithm to calculate BOW is explained in the Proposed Algorithm section.

The second approach **weighted Word2vec** is a novel one and involves the use of two vectorization methods.

Here we have first vectorized the Tag column using TFIDF vectorization. The algorithm to calculate TFIDF is explained in the Proposed Algorithm section.

Then we used the Word2Vec method to represent every unique word with a 159-size vector. Hyperparameters like vector size and window are tuned to improve the closeness of the vectors with similar words. Then, we use the TFIDF value of a word in the tag of a particular cloth as the weight to measure the weighted average Word2Vec vector representation of the Tag.

This vector is **159 X 1** which is much less as compared to the BOW vector

which was **1958 X 1** (a total of **1958 unique words** in the Tag column). Also, the vector formed using weighted Word2Vec holds semantic meaning.

Also different from the normal way, we didn't pre-calculate the cosine similarity between every two pairs of vectors, instead we kept it real-time. That is, once the user has entered the cloth ID, we will calculate the cosine similarity of the vector of that cloth with every other cloth, sort them, and then find the top 10 similar cloths.

The first approach has an amortized cost of $O(n^3)$. $O(n)$ to calculate cosine similarity between a pair of vectors, and we have n^2 possible pairs of vectors. After the user has entered the ID, it will take $O(n \log n)$ time to find the top 10 clothes. Here we require an extra space of $O(n^2)$, which is very costly. The matrix size will be 44072×44072, which will occupy around $14GB$, and that is why we selected the second method. It was not possible to store such a huge matrix on Kaggle or Colab; also, it requires a lot of computation time and high-end processors.

In the second method, it will take $O(n^2)$ to find the top 10 clothes and **no extra space**.

One more optimization that we applied here is the use of an adjacency list while calculating the cosine similarity using a BOW vector. We know that the vector formed using BOW is a **sparse** one and has only 10 useful values so using an **adjacency list** saved much time and space.

The BOW method is found to be more optimized in terms of space and time. Since we are using an adjacency list the approximate vector size will be **10X1** as compared to **159X1** in the weighted wrd2vec approach. Also, if the user enters the

tags that are **not** present in the current dataset then it will be easy to vectorize that tag using BOW. Here, there is no need to change the vector representation of the other tags present in the dataset whereas if we use TFIDF or wrd2vec then entering a new tag will change the vector representation of all the other tags.

0.3 PROPOSED ALGORITHM

Figure 5: Process Diagram

Step :1 Loading the dataset from kaggle

```
[23]: dataset = load_dataset("ashraq/fashion-product-images-small")
```

```
[24]: print(dataset)
```

```
DatasetDict({
    train: Dataset({
        features: ['id', 'gender', 'masterCategory', 'subCategory', 'articleT
ype', 'baseColour', 'season', 'year', 'usage', 'productDisplayName', 'imag
e'],
        num_rows: 44072
    })
})
```

Figure 6: Loading dataset using load dataset

[136]:
```
data.head()
```

[136…

	id	gender	masterCategory	subCategory	articleType	baseColour	season	year	usage	productDisplayName	image
0	15970	Men	Apparel	Topwear	Shirts	Navy Blue	Fall	2011.0	Casual	Turtle Check Men Navy Blue Shirt	<PIL.JpegImagePlugin.JpegImageFile image mode=...
1	39386	Men	Apparel	Bottomwear	Jeans	Blue	Summer	2012.0	Casual	Peter England Men Party Blue Jeans	<PIL.JpegImagePlugin.JpegImageFile image mode=...
2	59263	Women	Accessories	Watches	Watches	Silver	Winter	2016.0	Casual	Titan Women Silver Watch	<PIL.Image.Image image mode=L size=60x80 at 0x...
3	21379	Men	Apparel	Bottomwear	Track Pants	Black	Fall	2011.0	Casual	Manchester United Men Solid Black Track Pants	<PIL.JpegImagePlugin.JpegImageFile image mode=...
4	53759	Men	Apparel	Topwear	Tshirts	Grey	Summer	2012.0	Casual	Puma Men Grey T-shirt	<PIL.Image.Image image mode=RGB size=60x80 at ...

Figure 7: Dataset

Step :2

1)Data Cleaning :

The values of the year were given in the float datatype. To achieve proper format, we first convert each value to an integer and then to a string, utilizing the apply function in combination with a lambda function. This process ensures homogeneity in data type.

[140]:
```
# Data cleaning
data['year'] = data['year'].apply(lambda x : str(int(x)))
```

[141]:
```
data.head()
```

[141…

	id	gender	masterCategory	subCategory	articleType	baseColour	season	year	usage	productDisplayName	image
0	15970	Men	Apparel	Topwear	Shirts	Navy Blue	Fall	2011	Casual	Turtle Check Men Navy Blue Shirt	<PIL.JpegImagePlugin.JpegImageFile image mode=...
1	39386	Men	Apparel	Bottomwear	Jeans	Blue	Summer	2012	Casual	Peter England Men Party Blue Jeans	<PIL.JpegImagePlugin.JpegImageFile image mode=...
2	59263	Women	Accessories	Watches	Watches	Silver	Winter	2016	Casual	Titan Women Silver Watch	<PIL.Image.Image image mode=L size=60x80 at 0x...
3	21379	Men	Apparel	Bottomwear	Track Pants	Black	Fall	2011	Casual	Manchester United Men Solid Black Track Pants	<PIL.JpegImagePlugin.JpegImageFile image mode=...
4	53759	Men	Apparel	Topwear	Tshirts	Grey	Summer	2012	Casual	Puma Men Grey T-shirt	<PIL.Image.Image image mode=RGB size=60x80 at ...

Figure 8: Data Cleaning

2)Data Reduction :

We then remove the 'image' column from the DataFrame. This effectively reduces the dimensionality of the data frame, thus making it easier to analyze or model.

[142]:
```
# data reduction remoiving unsual columns
data = data.drop(columns=['image'])
```

[143]:
```
data.head()
```

[143…

	id	gender	masterCategory	subCategory	articleType	baseColour	season	year	usage	productDisplayName
0	15970	Men	Apparel	Topwear	Shirts	Navy Blue	Fall	2011	Casual	Turtle Check Men Navy Blue Shirt
1	39386	Men	Apparel	Bottomwear	Jeans	Blue	Summer	2012	Casual	Peter England Men Party Blue Jeans
2	59263	Women	Accessories	Watches	Watches	Silver	Winter	2016	Casual	Titan Women Silver Watch
3	21379	Men	Apparel	Bottomwear	Track Pants	Black	Fall	2011	Casual	Manchester United Men Solid Black Track Pants
4	53759	Men	Apparel	Topwear	Tshirts	Grey	Summer	2012	Casual	Puma Men Grey T-shirt

Figure 9: Data Reduction

3)Checking for null values :

The next step is to identify null values in each column of the dataframe. This is done by utilizing the 'isna()' method to detect null values and then applying the 'sum()' method to calculate the total count of null values in each column. This approach is useful in assessing the dataset's completeness and identifying columns that have missing data that may require further handling or imputation.

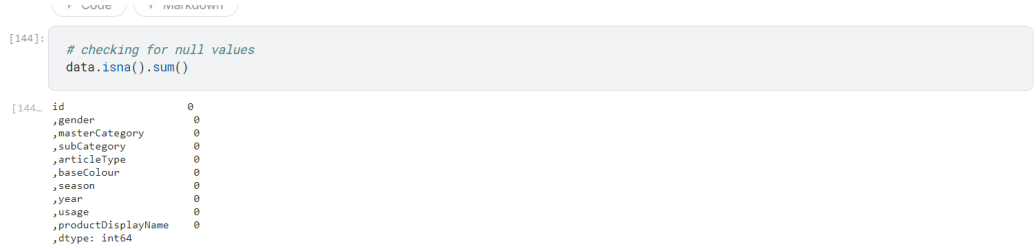

Figure 10: Checking for NULL values

4)Data Extraction :

This line of code extracts the brand name from the 'productDisplayName' column in the DataFrame and stores it in a new column called 'BrandName'. It uses a lambda function with the apply() method to split each value in the 'productDisplayName' column by whitespace (split()), then it takes the first one or two elements (depending on the length of the split) and joins them together to form the brand name. This extraction process helps isolate and categorize brand information for further analysis or organization within the dataset.

[148]:
```
# Data extraction - extraction Brand name form Product Display Name
data['BrandName'] = data['productDisplayName'].apply(lambda x: "".join(x.split()[0:min(2,len(x.split()))]))
```

[149]:
```
data.head()
```

[149…

	id	gender	masterCategory	subCategory	articleType	baseColour	season	year	usage	productDisplayName	BrandName
0	15970	Men	Apparel	Topwear	Shirts	Navy Blue	Fall	2011	Casual	Turtle Check Men Navy Blue Shirt	TurtleCheck
1	39386	Men	Apparel	Bottomwear	Jeans	Blue	Summer	2012	Casual	Peter England Men Party Blue Jeans	PeterEngland
2	59263	Women	Accessories	Watches	Watches	Silver	Winter	2016	Casual	Titan Women Silver Watch	TitanWomen
3	21379	Men	Apparel	Bottomwear	Track Pants	Black	Fall	2011	Casual	Manchester United Men Solid Black Track Pants	ManchesterUnited
4	53759	Men	Apparel	Topwear	Tshirts	Grey	Summer	2012	Casual	Puma Men Grey T-shirt	PumaMen

Figure 11: Data Extraction

5) Preparing data to fit for vectorizer

This loop iterates through each column in the DataFrame, excluding the 'id' column. For each column, it applies a lambda function that removes whitespace from each value by replacing it with an empty string. The result is then enclosed within a list. This preprocessing step is likely performed to prepare the data for vectorization, ensuring consistency in formatting across different columns and facilitating subsequent analysis.

[151]:

```
# Data preprocessing to make it fit for vectorization
for col in data:
  if(col=='id'): continue
  data[col] = data[col].apply(lambda x : [x.replace(" ","")])
```

```
data.head()
```

[152…

	id	gender	masterCategory	subCategory	articleType	baseColour	season	year	usage	BrandName
0	15970	[Men]	[Apparel]	[Topwear]	[Shirts]	[NavyBlue]	[Fall]	[2011]	[Casual]	[TurtleCheck]
1	39386	[Men]	[Apparel]	[Bottomwear]	[Jeans]	[Blue]	[Summer]	[2012]	[Casual]	[PeterEngland]
2	59263	[Women]	[Accessories]	[Watches]	[Watches]	[Silver]	[Winter]	[2016]	[Casual]	[TitanWomen]
3	21379	[Men]	[Apparel]	[Bottomwear]	[TrackPants]	[Black]	[Fall]	[2011]	[Casual]	[ManchesterUnited]
4	53759	[Men]	[Apparel]	[Topwear]	[Tshirts]	[Grey]	[Summer]	[2012]	[Casual]	[PumaMen]

Figure 12: Final data format

Step :3

Applying Vectorizer :

1)BOW(Bag Of Word)

```
[167]: processed_data.head()
```

	id	Tag	BOW
0	15970	men apparel topwear shirts navyblue fall 2011 ...	{1148: 2, 112: 2, 1825: 2, 1648: 2, 1215: 2, 7...
1	39386	men apparel bottomwear jeans blue summer 2012 ...	{1148: 2, 112: 2, 296: 2, 223: 2, 939: 2, 216:...
2	59263	women accessories watches watches silver winte...	{296: 2, 1922: 2, 26: 2, 1901: 4, 1657: 2, 192...
3	21379	men apparel bottomwear trackpants black fall 2...	{1148: 2, 112: 2, 716: 2, 9: 2, 296: 2, 223: 2...
4	53759	men apparel topwear tshirts grey summer 2012 c...	{1148: 2, 112: 2, 1825: 2, 296: 2, 1755: 2, 10...

Figure 13: BOW

2)weighted Word2Vec

```
[179]: processed_data.head()
```

	id	Tag	BOW	wrd2vec
0	15970	men apparel topwear shirts navyblue fall 2011 ...	{1148: 2, 112: 2, 1825: 2, 1648: 2, 1215: 2, 7...	[0.05704307701024744, -0.04891449109547668, 0....
1	39386	men apparel bottomwear jeans blue summer 2012 ...	{1148: 2, 112: 2, 296: 2, 223: 2, 939: 2, 216:...	[-0.0538736575593551, 0.08224543266826206, 0.1...
2	59263	women accessories watches watches silver winte...	{296: 2, 1922: 2, 26: 2, 1901: 4, 1657: 2, 192...	[-0.11195660041024287, 0.1914643837759892, 0.4...
3	21379	men apparel bottomwear trackpants black fall 2...	{1148: 2, 112: 2, 716: 2, 9: 2, 296: 2, 223: 2...	[0.01708908234205511, 0.04222696812616454, 0.1...
4	53759	men apparel topwear tshirts grey summer 2012 c...	{1148: 2, 112: 2, 1825: 2, 296: 2, 1755: 2, 10...	[-0.01710337053777443, 0.03846833854913715, 0....

Figure 14: 2 vectorizer results for each ids

Step 4:

Measuring Cosine Similarity

The function computes the cosine similarity between two vectors represented as dictionaries. It converts the dictionaries into arrays for numerical operations, calculates the magnitudes of the vectors, and then computes their dot product. If either vector has zero magnitude, it indicates an error. Finally, it returns the cosine similarity, a measure ranging from -1 to 1, where values closer to 1 imply higher similarity. This function is valuable for tasks such as measuring similarity between documents or analyzing relationships in high-dimensional data.

```python
def calc_cosine(x,y):

    mod_a=np.array([i for i in x.values()])
    mod_b=np.array([i for i in y.values()])
    mod_a=(np.sum(mod_a**2))**0.5
    mod_b=(np.sum(mod_b**2))**0.5

    mod_ab=0
    for i in y:
        if i in x:
            mod_ab+=(y[i]*x[i])
    if(mod_a==0 or mod_b==0):
        print("something went wrong")
        return 0
    return (mod_ab/(mod_a*mod_b))
```

Figure 15: Cosine Measuring

Step 5: This code segment retrieves recommendations for clothes based on cosine similarity with the input cloth ID. It first prompts the user to input a cloth ID, and then maps it to the corresponding index in the dataset. After obtaining the relevant tag for the input cloth, it computes the cosine similarity between the input cloth and all other clothes in the dataset. The top five unique most similar clothes are selected and displayed as recommendations. Finally, it prints the IDs and tags of the recommended clothes. This functionality provides users with personalized recommendations based on the similarity of clothing tags.

```
Enter clothe ID : 47359
women accessories bags handbags brown summer 2012 casual baggitwomen women accessories bags handbags brown summer 2012 casual baggitwomen
Recommended Clothes :
-----------------------------------------------------------------
38441 women accessories bags handbags brown summer 2012 casual baggitwomen women accessories bags handbags brown summer 2012 casual baggitwomen
33633 women accessories bags handbags brown summer 2012 casual linoperros women accessories bags handbags brown summer 2012 casual linoperros
38446 women accessories bags handbags beige summer 2012 casual baggitwomen women accessories bags handbags beige summer 2012 casual baggitwomen
33406 women accessories bags handbags olive summer 2012 casual baggitwomen women accessories bags handbags olive summer 2012 casual baggitwomen
33430 women accessories bags handbags blue summer 2012 casual baggitwomen women accessories bags handbags blue summer 2012 casual baggitwomen
```

Figure 16: Top 5 Recommendation

0.4 RESULTS AND DISCUSSION

```
Enter clothe ID :  29114
men accessories socks socks navyblue summer 2012 casual pumamen men accessories socks socks navy
blue summer 2012 casual pumamen

Recommended Clothes :
----------------------------------------------------------------

29116 men accessories socks socks black summer 2012 casual pumamen men accessories socks socks b
lack summer 2012 casual pumamen

43561 men accessories socks socks white summer 2012 casual pumamen men accessories socks socks w
hite summer 2012 casual pumamen

18585 men accessories socks socks navyblue summer 2010 casual pumamen men accessories socks sock
s navyblue summer 2010 casual pumamen

46868 men accessories socks socks black summer 2012 casual linoperros men accessories socks sock
s black summer 2012 casual linoperros

14614 men accessories socks socks navyblue summer 2011 casual unitedcolors men accessories socks
socks navyblue summer 2011 casual unitedcolors
```

Figure 17: BOW Output 1

```
Enter clothe ID :  29114
men accessories socks socks navyblue summer 2012 casual pumamen men accessories socks socks navy
blue summer 2012 casual pumamen

Recommended Clothes :
----------------------------------------------------------------

29116 men accessories socks socks black summer 2012 casual pumamen men accessories socks socks b
lack summer 2012 casual pumamen

18579 men accessories socks socks navyblue summer 2011 sports pumamen men accessories socks sock
s navyblue summer 2011 sports pumamen

43561 men accessories socks socks white summer 2012 casual pumamen men accessories socks socks w
hite summer 2012 casual pumamen

18581 men accessories socks socks black summer 2011 casual pumamen men accessories socks socks b
lack summer 2011 casual pumamen

18586 men accessories socks socks olive summer 2011 casual pumamen men accessories socks socks o
live summer 2011 casual pumamen
```

Figure 18: Weighted Wrd2Vec Output 1

```
Enter clothe ID :  48311
women accessories jewellery bracelet bronze winter 2012 casual pitaraawomen women accessories je
wellery bracelet bronze winter 2012 casual pitaraawomen

Recommended Clothes :
----------------------------------------------------------------

48313 women accessories jewellery bracelet gold winter 2012 casual pitaraagold women accessories
jewellery bracelet gold winter 2012 casual pitaraagold

48572 women accessories jewellery bracelet silver winter 2012 casual lucerawomen women accessori
es jewellery bracelet silver winter 2012 casual lucerawomen

35133 women accessories jewellery bracelet cream winter 2012 casual allensolly women accessories
jewellery bracelet cream winter 2012 casual allensolly

48319 women accessories jewellery bracelet gold winter 2012 casual pitaraagolden women accessori
es jewellery bracelet gold winter 2012 casual pitaraagolden

48318 women accessories jewellery bracelet silver winter 2012 casual pitaraasilver women accesso
ries jewellery bracelet silver winter 2012 casual pitaraasilver
```

Figure 19: BOW Output 2

```
Enter clothe ID :  48311
women accessories jewellery bracelet bronze winter 2012 casual pitaraawomen women accessories je
wellery bracelet bronze winter 2012 casual pitaraawomen

Recommended Clothes :
----------------------------------------------------------------

48313 women accessories jewellery bracelet gold winter 2012 casual pitaraagold women accessories
jewellery bracelet gold winter 2012 casual pitaraagold

48319 women accessories jewellery bracelet gold winter 2012 casual pitaraagolden women accessori
es jewellery bracelet gold winter 2012 casual pitaraagolden

58058 women accessories jewellery earrings gold winter 2012 casual rreveriewomen women accessori
es jewellery earrings gold winter 2012 casual rreveriewomen

46031 women accessories jewellery pendant gold winter 2016 casual estellependant women accessori
es jewellery pendant gold winter 2016 casual estellependant

42620 women accessories jewellery earrings gold winter 2015 casual estellegold women accessories
jewellery earrings gold winter 2015 casual estellegold
```

Figure 20: Weighted Wrd2Vec Output 2

The above output shows that the result produced by BOW and Weighted Wrd2Vec does not differ much.

To evaluate the performance of this system either we need to have labeled data or we can use any of the below methods :

- User Feedback Collection: Gather feedback from users through surveys or studies to understand their perception of the recommended items' relevance and utility. This qualitative input helps assess the system's effectiveness.
- A/B Testing Method: Conduct A/B testing by dividing users into different groups to compare the performance of the recommendation system against alternatives or no recommendations at all. Analyze user engagement metrics like click-through rates to measure impact.
- Implicit Signals Analysis: Utilize implicit feedback cues such as user interactions (e.g., clicks, purchases) to infer user preferences and evaluate how well the recommendation system aligns with user behavior.

To conclude, the choice of the approach depends on Tag size and the computation power available. If the tag size is large such that the weighted wrd2vec vector is smaller than the BOW vector then we can go for weighted wrd2vec, or else with BOW.