Flipkart

GRiD 4.0

2022 Campus Challenge

Extract Trends from social media data

Team Name: CetITchamp

Institute Name: Odisha University Of Technology And Research

Team members details

Team Name	CetITchamp		
Institute Name	Odisha University Of Technology And Research		
Team Members >	**1 (Leader)**	**2**	**3**
Name	Siddhant Sekhar Purohit	BiswaKiran Das	Aniket Mishra
Batch	Information Technology	Information Technology	Information Technology

Deliverables/Expectations for Level 2 (Idea + Code Submission)

Deliverable 1:

Identification of trends from social media

1. Identify trends on social media based on category. Can restrict to Fashion as a category for the project. Ex: Polka dots dresses are trending on twitter.
2. Ranking/scoring logic for trends extracted.
3. Outcome format:
    a. Option1: List of trending keyword(s) along with list of sample images and respective links from which the trend is derived with most trending first:
    Example: Trends:[{Polka dot dresses, <list of links/images>,trending score}, {Bellbottom Jeans, <list of links/images>,trending score}..]
    b. Option 2: structured data according to flipkart category, sub category, vertical and product attributes
    Example: {category: Fashion, Sub-category: Women Western, vertical: Women dresses, trending attribute type: Pattern, trending attribute value: Polka Print, list of sample images and links from which the trend is derived}.
    Outcome with Option 2 format will be given bonus points.

Deliverable 2:

Mapping trends with Flipkart products:

1. Create mapping of extracted trending keyword(s) with Flipkart category, sub category, vertical and product attribute(s), search page links.
   Example:{category: Fashion, Sub-category: Women Western, vertical: Women dresses, trending attribute type: Pattern, trending attribute value: Polka Print}
   Note: Use category, Subcategory combination from the Flipkart Website
2. From a trending keyword, creating a corresponding searchable term on Flipkart which will lead to matching products.
   Example: Tropical Tops keywords will not give right results directly on Flipkart but we can construct search query for it using some intelligence.
3. Points will be given based on similarity between sample images for trends and product results on Flipkart.

Use-cases

P0-> We can get current fashion related trends from Twitter which is one of the biggest social-media platform of current time.

.

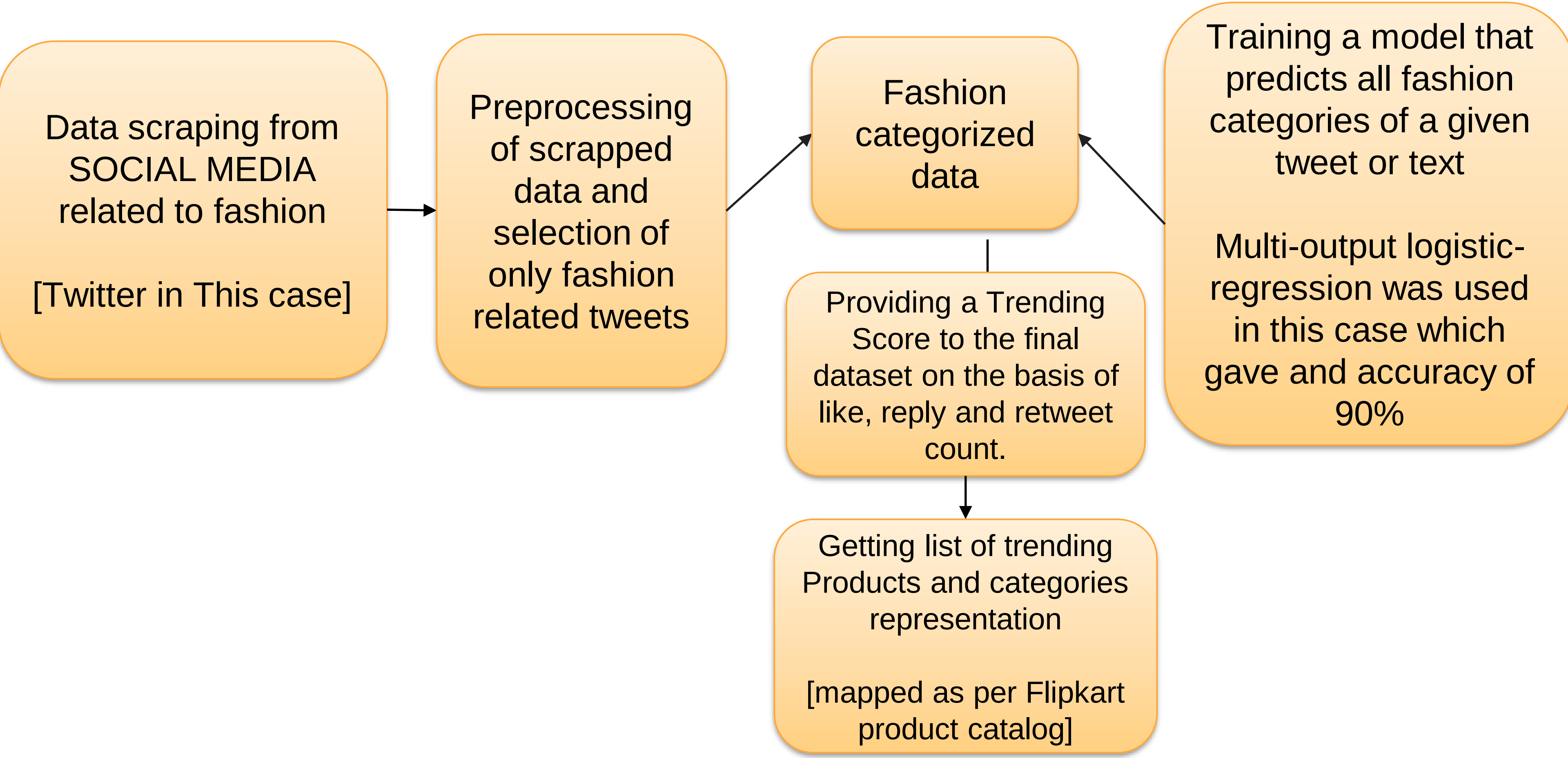
P1-> Preprocessing any data or text-files in order to get only fashion related sentences and further classify those to all different fashion categories

[Eg:

Given sentence:	`text:"RT @harzior: These menâ€™s &amp; polka print dresses for girls are perfect for this weather.\n\nShop NowðŸ”¥\nhttpâ€¦"}`
Sentence after preprocessing:	`polka print dresses girls`
Categories predicted:	`'Girls' 'Apparel' 'Dress' 'Dresses' 'Casual' 'Polka Dot‘`**]**

P2-> extracted images of Instagram models can be processed and classified into fashion categories which can further be used to get trends from the social media app Instagram.

SOLUTION 1 [GETTING FASHION TRENDS FROM TWITTER]



Data scraping from SOCIAL MEDIA related to fashion

[Twitter in This case]

Preprocessing of scrapped data and selection of only fashion related tweets

Fashion categorized data

Training a model that predicts all fashion categories of a given tweet or text

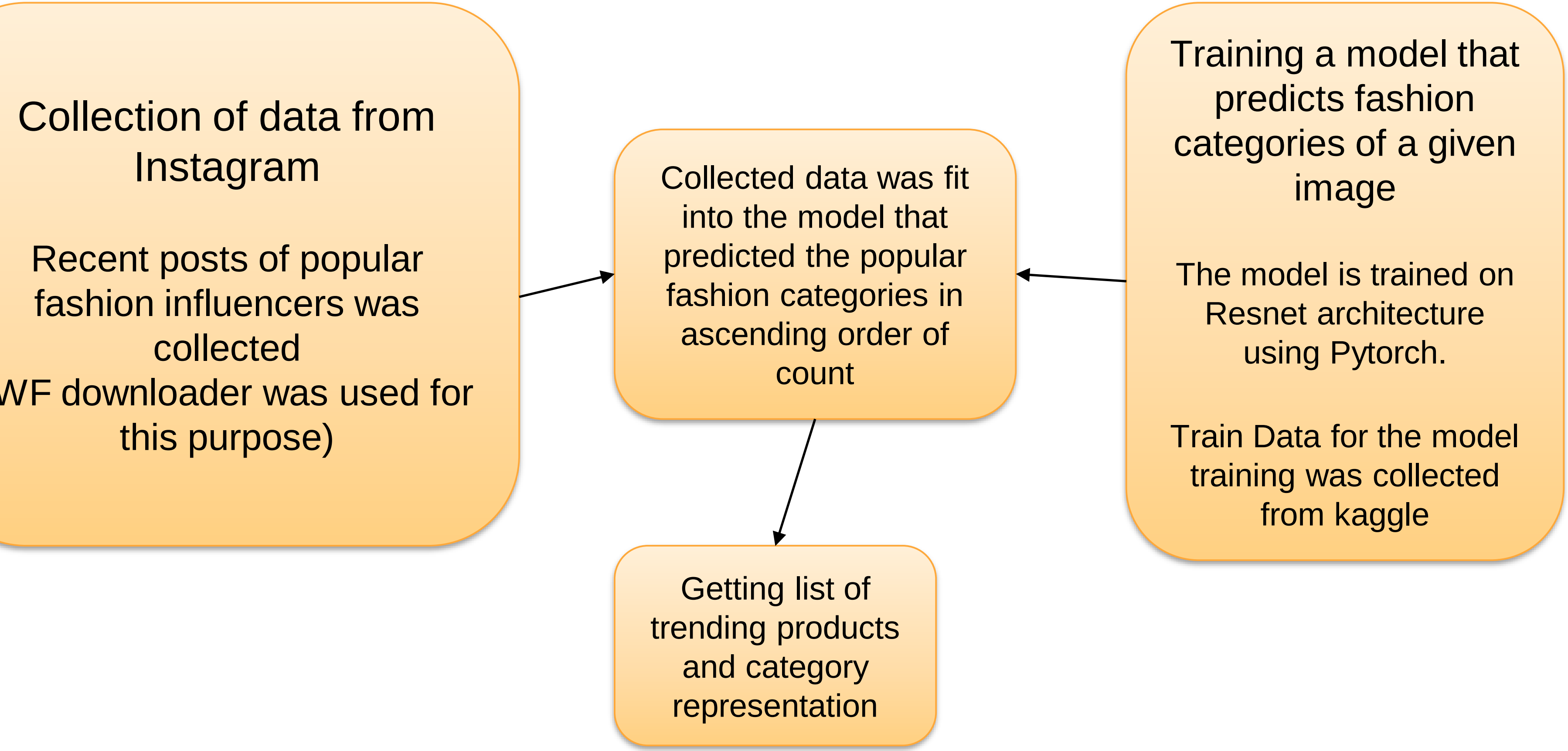
Multi-output logistic-regression was used in this case which gave and accuracy of 90%

Providing a Trending Score to the final dataset on the basis of like, reply and retweet count.

Getting list of trending Products and categories representation

[mapped as per Flipkart product catalog]

SOLUTION PART-2 [GETTING TRENDS FROM INSTAGRAM



Collection of data from Instagram

Recent posts of popular fashion influencers was collected

WF downloader was used for this purpose)

Collected data was fit into the model that predicted the popular fashion categories in ascending order of count

Training a model that predicts fashion categories of a given image

The model is trained on Resnet architecture using Pytorch.

Train Data for the model training was collected from kaggle

Getting list of trending products and category representation

STEPS INVOLVED

1. Collection of Data:

.Data was scraped using Twitter’s Stream API.
.Popular fashion hashtags like #fashion #OOTD #style #streetfashion #modelling was used in order to extract popular/trendy fashion data.
.Extracted data was stored in separate CSV files which was finally merged to get one big dataset.

2.Cleaning of data

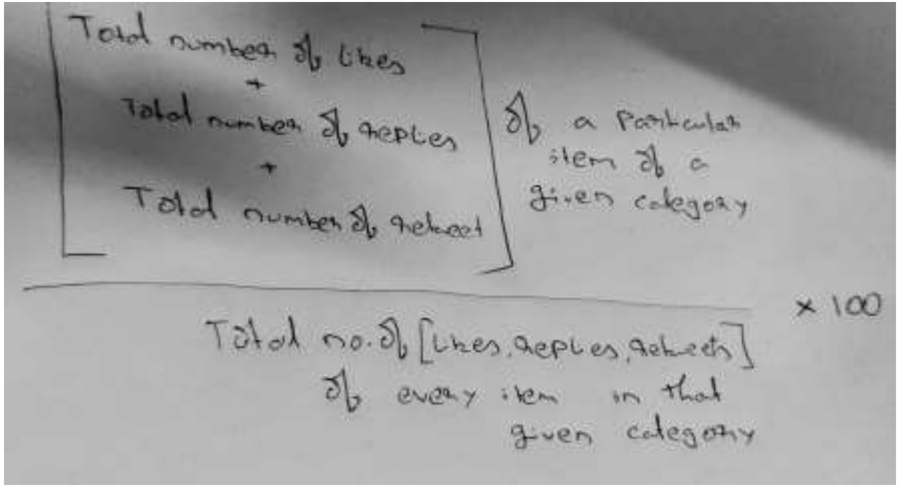
.Given dataset was preprocessed and cleaned in order to get a proper balanced dataset which was further preprocessed to extract only the fashion related data.
.NLP was used in this case and a custom list of only fashion related words was used to identify the fashion sentences.
Final dataset consisted only of preprocessed fashion related tweets along with the like, retweets and reply count.

3.Creation and training of Model

.A custom model was trained which classifies a fashion related sentence to its subcategories, (['Gender', 'masterCategory', 'subCategory', 'articleType', 'usage‘, ‘patternType’])
.Model was trained from a dataset found on the internet (Myntra product images catalog)
.Multi-class Logistic regression was used which gave an accuracy of 90%.
.The final cleaned dataset was fit on the model.

4.Giving Trending score and mapping

. A trending score logic was given and a list of trending products was obtained.
. Trending score for each item was given based on the like, reply and retweet count.

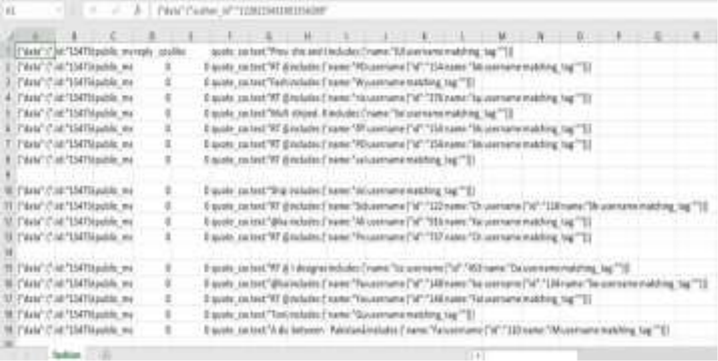


$$\frac{\left[\begin{array}{c}\text{Total number of likes} \\ + \\ \text{Total number of replies} \\ + \\ \text{Total number of retweet}\end{array}\right] \begin{array}{l}\text{of a particular} \\ \text{item of a} \\ \text{given category}\end{array}}{\begin{array}{c}\text{Total no. of [likes, replies, retweets]} \\ \text{of every item in that} \\ \text{given category}\end{array}} \times 100$$

<-**LOGIC given in picture**

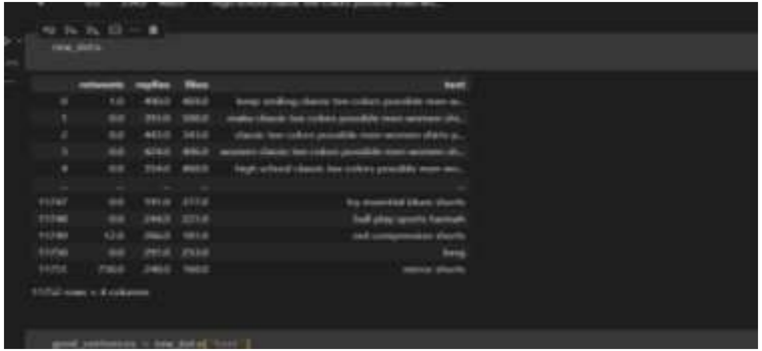
.Hence, trendy fashion products and categories was obtained.
We can use the logic and the data to get any sort of desired results.
Given trendy items can be mapped via the model itself. Giving a particular product to the model, gives its mapped representation.

1.Raw data extracted from twitter.



```
{"data":[{"author_id":"1228[illegible]"
{"data":[{"id":"154T[illegible] public_me reply_co likes   quote_co text":"[illegible] includes":{"name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"RT @ includes":{"name":"[illegible] username":{"id":"154 name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"Fash includes":{"name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"RT @ includes":{"name":"[illegible] username":{"id":"378 name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"Multi striped. [illegible] includes":{"name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"RT @ includes":{"name":"[illegible] username":{"id":"154 name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"RT @ includes":{"name":"[illegible] username":{"id":"154 name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"RT @ includes":{"name":"[illegible] username matching_tag"}]}

{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"Ship includes":{"name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"RT @ includes":{"name":"[illegible] username":{"id":"122 name":"[illegible] username":{"id":"118 name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"@[illegible] includes":{"name":"[illegible] username":{"id":"316 name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"RT @ includes":{"name":"[illegible] username":{"id":"157 name":"[illegible] username matching_tag"}]}

{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"RT @ [illegible] designe includes":{"name":"[illegible] username":{"id":"453 name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"@[illegible] includes":{"name":"[illegible] username":{"id":"148 name":"[illegible] username":{"id":"134 name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"RT @ includes":{"name":"[illegible] username":{"id":"148 name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"Toni includes":{"name":"[illegible] username matching_tag"}]}
{"data":[{"id":"154T[illegible] public_me   0   0 quote_co text":"A du between - Pakistan& includes":{"name":"[illegible] username":{"id":"110 name":"[illegible] username matching_tag"}]}
```

2.Preprocessed data containing only fashion tweets



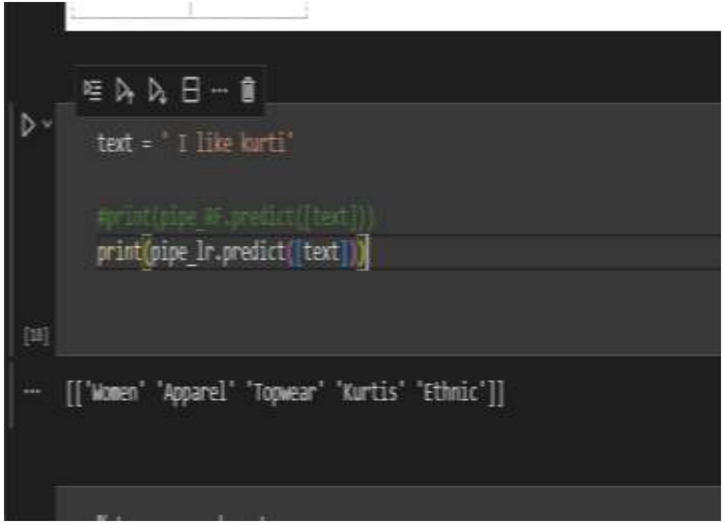
```
new_data
```

	retweets	replies	likes	text
0	1.0	490.0	469.0	keep smiling classic tee colors possible men w...
1	0.0	391.0	500.0	make classic tee colors possible men women shi...
2	0.0	443.0	343.0	classic tee colors possible men women shirts p...
3	0.0	424.0	406.0	women classic tee colors possible men women sh...
4	0.0	354.0	460.0	high school classic tee colors possible men wo...
...	...	...	...	...
11747	0.0	191.0	277.0	try essential blues shorts
11748	0.0	244.0	227.0	ball play sports hannah
11749	12.0	266.0	181.0	red compression shorts
11750	0.0	291.0	253.0	long
11751	758.0	248.0	160.0	mirror shorts

11752 rows × 4 columns

```
good_sentences = new_df['text']
```

3.Trained model with an example classification

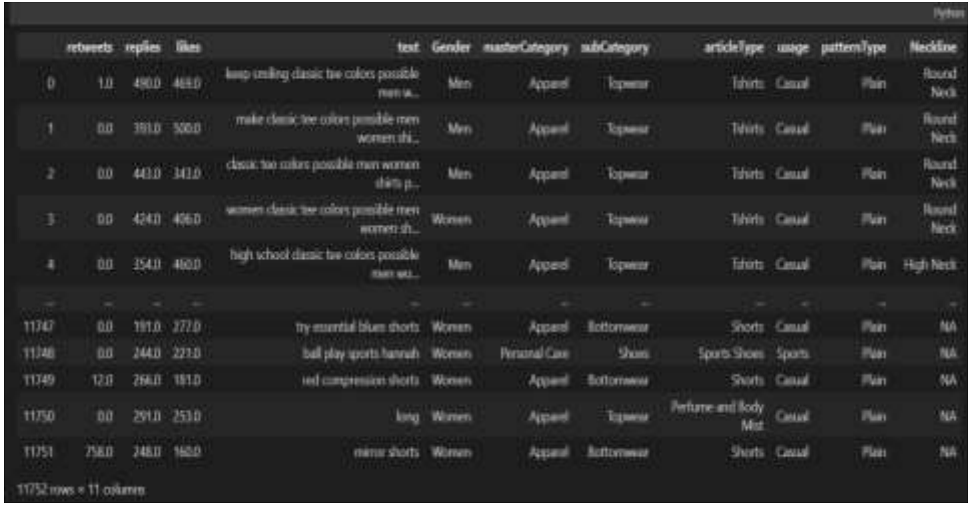


```
text = ' I like kurti'

#print(pipe_RF.predict([text]))
print(pipe_lr.predict([text]))
```

[10]

```
[['Women' 'Apparel' 'Topwear' 'Kurtis' 'Ethnic']]
```

4.Final dataset with all predicted categories



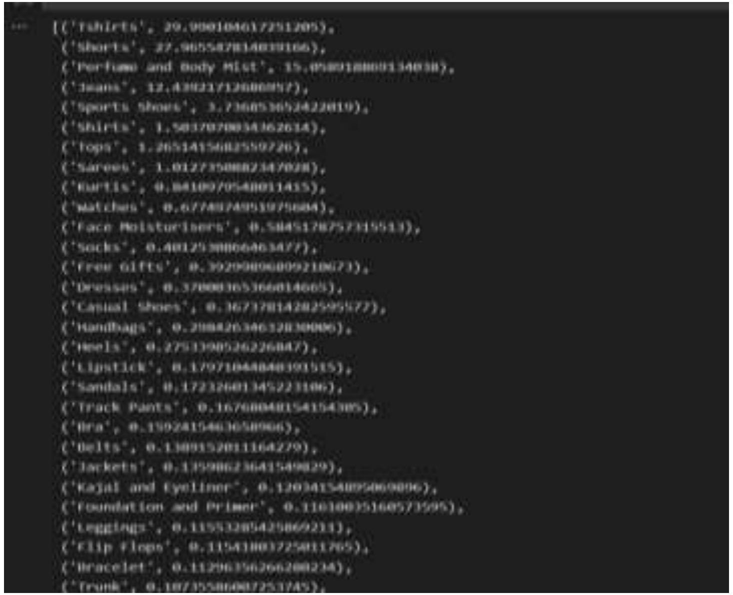
Python

	retweets	replies	likes	text	Gender	masterCategory	subCategory	articleType	usage	patternType	Neckline
0	1.0	490.0	469.0	keep smiling classic tee colors possible men w...	Men	Apparel	Topwear	Tshirts	Casual	Plain	Round Neck
1	0.0	391.0	500.0	make classic tee colors possible men women shi...	Men	Apparel	Topwear	Tshirts	Casual	Plain	Round Neck
2	0.0	443.0	343.0	classic tee colors possible men women shirts p...	Men	Apparel	Topwear	Tshirts	Casual	Plain	Round Neck
3	0.0	424.0	406.0	women classic tee colors possible men women sh...	Women	Apparel	Topwear	Tshirts	Casual	Plain	Round Neck
4	0.0	354.0	460.0	high school classic tee colors possible men wo...	Men	Apparel	Topwear	Tshirts	Casual	Plain	High Neck
...	...	...	...	...	...	...	...	...	...	...	...
11747	0.0	191.0	277.0	try essential blues shorts	Women	Apparel	Bottomwear	Shorts	Casual	Plain	NA
11748	0.0	244.0	227.0	ball play sports hannah	Women	Personal Care	Shoes	Sports Shoes	Sports	Plain	NA
11749	12.0	266.0	181.0	red compression shorts	Women	Apparel	Bottomwear	Shorts	Casual	Plain	NA
11750	0.0	291.0	253.0	long	Women	Apparel	Topwear	Perfume and Body Mist	Casual	Plain	NA
11751	758.0	248.0	160.0	mirror shorts	Women	Apparel	Bottomwear	Shorts	Casual	Plain	NA

11752 rows × 11 columns

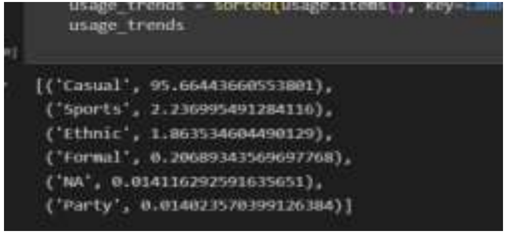
TRENDS EXTRACTED FROM TWITTER

Trending score of top fashion items



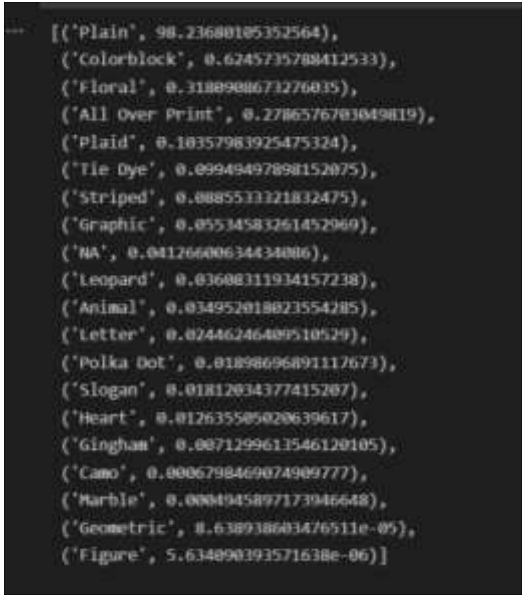
```
[('Tshirts', 29.990104617251205),
 ('Shorts', 27.965547814039166),
 ('Perfume and Body Mist', 15.058918809134038),
 ('Jeans', 12.438217126808857),
 ('Sports Shoes', 3.736853652422019),
 ('Shirts', 1.583707003436263),
 ('Tops', 1.265141568255972),
 ('Sarees', 1.0127350882347028),
 ('Kurtis', 0.8410970548011415),
 ('Watches', 0.677487495197560),
 ('Face Moisturisers', 0.5845178757315513),
 ('Socks', 0.4012538066463477),
 ('Free Gifts', 0.3929989600921867),
 ('Dresses', 0.3700036536601466),
 ('Casual Shoes', 0.3673781428259557),
 ('Handbags', 0.2984263463283000),
 ('Heels', 0.2753390526222684),
 ('Lipstick', 0.1797104484039151),
 ('Sandals', 0.1723260134522310),
 ('Track Pants', 0.1676004815415438),
 ('Bra', 0.1592415463650906),
 ('Belts', 0.1389152011164279),
 ('Jackets', 0.1359862364154982),
 ('Kajal and Eyeliner', 0.1203415489506989),
 ('Foundation and Primer', 0.1161003516057359),
 ('Leggings', 0.1155328542586921),
 ('Flip Flops', 0.1154180372580117),
 ('Bracelet', 0.1129635626620823),
 ('Trunk', 0.1073558600725374),
```

Trending score of fashion usage categories



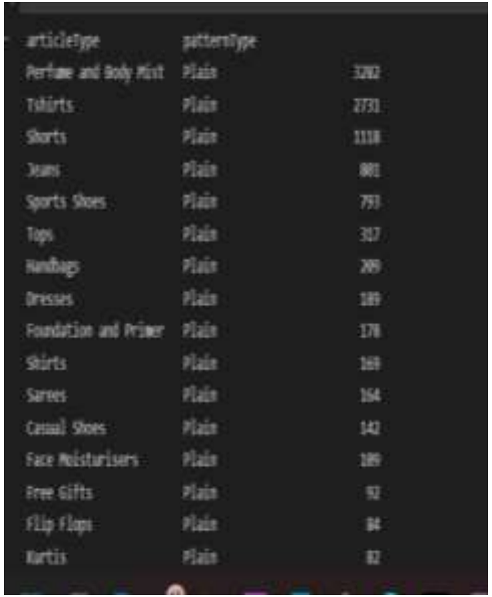
```
        usage_trends = sorted(usage.items(), key=
        usage_trends

[('Casual', 95.66443660553801),
 ('Sports', 2.236995491284116),
 ('Ethnic', 1.863534604490129),
 ('Formal', 0.20689343569697768),
 ('NA', 0.014116292591635651),
 ('Party', 0.014023570399126384)]
```

Trending score of fashion patterns



```
[('Plain', 98.23680105352564),
 ('Colorblock', 0.6245735788412533),
 ('Floral', 0.3188908867327603 5),
 ('All Over Print', 0.27865767030498 19),
 ('Plaid', 0.10357983925475324),
 ('Tie Dye', 0.09949497898152075),
 ('Striped', 0.08855333218324 75),
 ('Graphic', 0.05534583261452969),
 ('NA', 0.04126600634434086),
 ('Leopard', 0.03608311934157238),
 ('Animal', 0.03495201802355428 5),
 ('Letter', 0.02446246409510529),
 ('Polka Dot', 0.01898696891117673),
 ('Slogan', 0.01812034377415207),
 ('Heart', 0.01263550502063961 7),
 ('Gingham', 0.007129961354612010 5),
 ('Camo', 0.0006798469074909777),
 ('Marble', 0.0004945897173946648),
 ('Geometric', 8.638938603476511e-05),
 ('Figure', 5.634090393571638e-06)]
```

Trending score of fashion item along with pattern



```
articleType            patternType
Perfume and Body Mist  Plain          3282
Tshirts                Plain          2731
Shorts                 Plain          1118
Jeans                  Plain           801
Sports Shoes           Plain           793
Tops                   Plain           317
Handbags               Plain           209
Dresses                Plain           189
Foundation and Primer  Plain           178
Shirts                 Plain           169
Sarees                 Plain           164
Casual Shoes           Plain           142
Face Moisturisers      Plain           109
Free Gifts             Plain            92
Flip Flops             Plain            84
Kurtis                 Plain            82
```

PART-2 GETTING TRENDS FROM INSTAGRAM

COLLECTION OF DATA-

.Recent posts of popular fashion influencers was collected (WF downloader was used for this purpose)

.Due to time constraint we have used only 4 influencers (2 male and 2 female) for this project.

.Around 600 images were collected.

CREATION AND TRAINING OF MODEL-

.Model was trained on the dataset found on kaggle (named- Fashion Product Images Small).

.The model is trained on Resnet architecture using Pytorch.

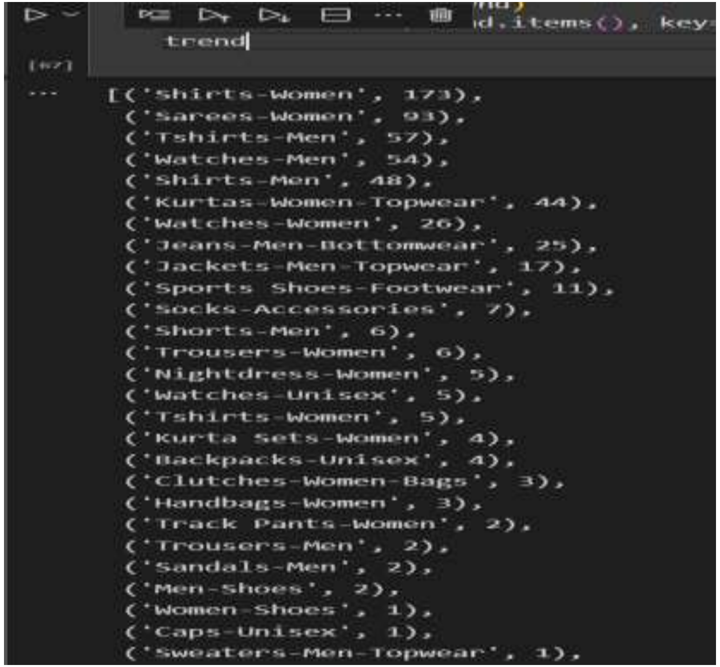
.Model can classify top 65 categories present on the dataset used.

GETTING TRENDY FASHION ITEMS AND CATEGORIES

.Extracted images are fit into the model to get fashion category/item in each image.
.All categories are stored in a list and sorted in ascending order.

.Final output produces list of fashion categories and items with their count number in sorted format, which we assume is the trendiness order.

[All the images for inference are downsized to 60x60px and normalized before sending to model. So a high pixel images doesn't make difference.]



```
                                        d.items(), key=
    trend
```

```
[('Shirts-Women', 173),
 ('Sarees-Women', 93),
 ('Tshirts-Men', 57),
 ('Watches-Men', 54),
 ('Shirts-Men', 48),
 ('Kurtas-Women-Topwear', 44),
 ('Watches-Women', 26),
 ('Jeans-Men-Bottomwear', 25),
 ('Jackets-Men-Topwear', 17),
 ('Sports Shoes-Footwear', 11),
 ('Socks-Accessories', 7),
 ('Shorts-Men', 6),
 ('Trousers-Women', 6),
 ('Nightdress-Women', 5),
 ('Watches-Unisex', 5),
 ('Tshirts-Women', 5),
 ('Kurta Sets-Women', 4),
 ('Backpacks-Unisex', 4),
 ('Clutches-Women-Bags', 3),
 ('Handbags-Women', 3),
 ('Track Pants-Women', 2),
 ('Trousers-Men', 2),
 ('Sandals-Men', 2),
 ('Men-Shoes', 2),
 ('Women-Shoes', 1),
 ('Caps-Unisex', 1),
 ('Sweaters-Men-Topwear', 1),
```

Future Scope

1. Improving the models further can be useful to predict more accurate results and getting trends from a more wide range of categories.

2. We have collected textual data only from Twitter as the idea was to extract trends from social media, but we can also get data from more websites and fashion magazines which when fed to the model will produce more accurate results.

3. This project can be extended to predict region wise trending products for targeted audience by collecting region specific data

4. This project focused mainly on extraction of fashion trends from twitter and Instagram using past 3-5 months data. But we can also work on real time data which will produce more recent and accurate trends.

Limitations

1. Giving a trendy score to images scrapped from Instagram was difficult as only the image was being scrapped so other factors which could be important in getting a trend score was missing. So we have only stick to the count of each category to determine trendiness as for this project.

2. We have scraped nearly 600 number of images from 4 popular fashion influencers from Instagram for our project because of time constraint. Although we can add more images and more wide range of models and influencers for a more accurate result.

3. For textual data our project only focuses on Twitter as we were more focused on the social media aspect of the assignment. But giving more accurate data to our model from fashion blogs, magazines and websites will produce better results [current results is only based on the data collected from Twitter].

4. The image classification is not very extensive as getting data from Instagram was difficult and because of the time constraint, which we can surely improve in the future.