Flipkart GRiD 2.0

# Fashion Intelligence Systems

Team Name : IITKianDoge

Institute Name: IIT Kanpur

# Team members details

| Team Name | IITKianDoge | | |
|---|---|---|---|
| Institute Name | Indian Institute of Technology Kanpur | | |
| Team Members | 1 (Leader) | 2 | 3 |
| Name | Shiven Tripathi | Rishabh Dugaye | Atharv Singh Patlan |
| Batch | 2023 | 2023 | 2023 |

**Experience**

Our team members are working in Team AUV-IITK and Team ERA-IITK. We have previous experience in making various machine learning models like deep generative models, natural language processing models,various sequence models etc.We also have a lot of experience with software development.We believe that with our expertise we can make an amazing product to solve the problem statement.

# Glossary

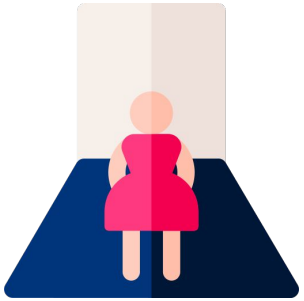
- **Attributes:** Distinguishing traits in clothes like fabric, colour, type, etcf.
- **Embedding:** A translation of a high dimensional vector to a lower dimensional space, here we generate embeddings composed of attributes from the higher dimensional images
- **Model:** An ML/DL agent which after training returns predictions on unseen data, here we have used CNN Models and ML Parametric Model
- **Data Augmentation:** Adding new features or transformations of existing features to training data to increase its size and train a better generalised model.
- **Time series Data:** Dataset indexed with time points to observe relations across time dimension
- **Fashion Trend:** A Fashion Style which has been observed to be more popular, or used by influential individuals like celebrities or Fashion Magazines
- **Fashion Styles:** A collection of attributes, forming a distinctly observable dress piece. For eg., yellow + half sleeve + shirt is a fashion style.
- **Corpus:** A collection of data, here, it is the collection of images taken daily from a list of fashion sites
- **StreetStyle 27k:** A manually annotated dataset which has been used for identifying attributes on our corpus.
- **Portfolio:** A collection of fashion styles, ranked in order of trendiness
- **API:** Application Program Interface, here we'll create a convenient dashboard as a frontend to a responsive API which serves to return trends.
- **CNN:** A type of Neural Network which has been very successful in solving Computer Vision tasks like Localisation, Recognition etc.

# Use-cases

## Identify current trends

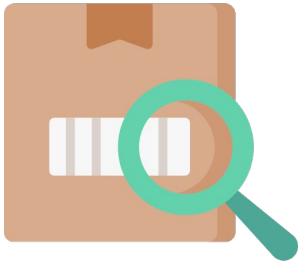
We can design a winning portfolio of our own, without human intervention from these trends.

## Identify upcoming trends



Using trends showcased in Fashion Magazines, we can build predictions on which trends would emerge.

## Predict future inventory



Knowing what trends we need to keep in stock, we can generate a dynamic predictive inventory for our portfolio.

# Solution statement/ Proposed approach

Fast Fashion retailers depend on runway trends for inspiration. Increasing competition in the industry demands for automated approaches for quickly identifying trends and implementing them in next lineup.
Our solution works in a **3 step approach**:

| | | |
|---|---|---|
| 1. | We use a Deep Learning based multi task Classifier to **recognise the attributes** present in any cloth sample. | Identifying these attributes gives a **representation for each cloth sample**. Eg.: (white, shirt, half sleeve, patterned, ….) |
| 2. | To identify the current trends, we **generate a dataset** which contains images of all products found on E Commerce sites. Likewise for upcoming trends, we maintain a separate database for images from Fashion Sites. | We would observe **winning trends to stay on all sites for longer duration** and losing trends to get taken down. We use this approach because most of the sites offer no useful features like ratings to identify trendiness. |
| 3. | We then use an ML model on this data and can finally **predict trends using parametric fitting**. | Using **observed average probability of an attribute** for a week, we can **model its variation** to a parametric model and then identify the winning trends. |

# Design/ solution choices

We did an extensive survey of existing literature on Fashion and observed that majority of it deals with very clean annotated datasets. This presented a unique problem to us wherein we had to identify best ways to utilise the limited data available from websites.

| Objective | Challenge | Solution | Why this works? |
|---|---|---|---|
| Getting useful features from sites | Most sites only have images with no time or rating data | Build a combined corpus of all images on a site, updated daily. | A combined corpus across all sites generalizes well to trends and gives time-series data. |
| Making a machine ‘understand’ fashion | Model has to identify attributes in images with varying proportions, labels. | We build a multi task CNN model based on GoogleNet architecture | We get an ‘embedding’ made of ‘attributes’ which our model identified for each cloth sample. |
| Identify Current Trends | Most E Commerce sites do not give ratings for each product, using which we could have identified trends. | Use the popularity of attributes across the corpus to build a parametric model for variation across time. | We get style clusters which are combinations of most popular attributes, thus, identifying a trend. |
| Identify Future Trends | None of the Fashion Magazines give ratings or suggestions for styles. | Make a similar model, but exclusively on corpus of runway images, magazines etc. | By modelling on trends yet to hit the markets, we can get indication of future trends and predict them. |

# Limitations

| What are the limitations? | How can these limitations be addressed? |
|---|---|
| Data from StreetStyle dataset is biased towards western fashion attributes. | Including more data from Indian websites and social media sites so that the model can learn attributes of Indian dresses. |
| Due to inherent sparsity of data, we would need to wait for some time before being able to observe any trends | This problem can be addressed by using real time series data over a wider time interval and for a larger number of websites spanning across product categories. |
| A lot of sites only give image data, which by itself is not a very strong indicator of trendiness | This limitation can be addressed by augmenting the image data with the sales data which can give us another feature indicating for popularity of a style. |

# Future Scope

01

Linking our solution with the **Inventory management system.**

- Saves time and labour.
- Saves a lot of money as stock of only trendy goods needs to be increased and lagging products are easily identified.

02

Combining the model with a **generative adversarial network.**

- Reduces the cost of hiring fashion designers.
- Model can automatically generate new designs which have attributes of trendy designs.
- Fashion designers can use it to gain inspiration for their new designs.

03

Linking the model with **social media data.**

- Social media is a great source of current fashion data.
- This data can be highly useful in determining current as well as upcoming trends.
- Data of what celebrities and the masses are wearing can be easily accessed.

# Milestones and Timeline for Completion

| During Stage 1 | Before stage 2 results | During stage 2 | During Grand Finale |
|---|---|---|---|

| Step 1 | Step 2 | Step 3 | Step 4 | Step 5 |
|---|---|---|---|---|
| • Train a CNN on the StreetStyle 27k dataset.<br>• We are now ready to use attribute embeddings | • Build an automated scraper using Selenium and BeautifulSoup to get image data.<br>• Since we are relying only on image data, this would be very efficient for us and allows us to use all websites. | • For each day, we create data dumps of images.<br>• We use Corpus from E Com. sites for current trends.<br>• We use Corpus from Fashion Sites for upcoming trends. | • Over these corpus, we generate attribute based embeddings for each sample.<br>• We then train an ML model over this temporal distribution to identify popularity trends. | The results will be served to the end-user via a highly responsive API, wherein they can view the present trends for the vertical they desire, as well as view the upcoming trends within a timescale. |

# Features for the end-user

### Two problems, one solution

Our API will give fashion designers and retailers alike the ability to know what is trending now, and what will in the future, at the touch of a button

### Focus on what matters the most

With our API, you get the ability to choose what matters the most to you! Whether you want to focus on the image data, the rating, or the review count, just wish, and our metrics will obey!

### Perfect Harmony with top e-com portals

Daily retrieval of data from top E-Com. websites, which passes through our custom metric, which utilizes all the important parts of the product description

### Predict the future using top Fashion Magazines

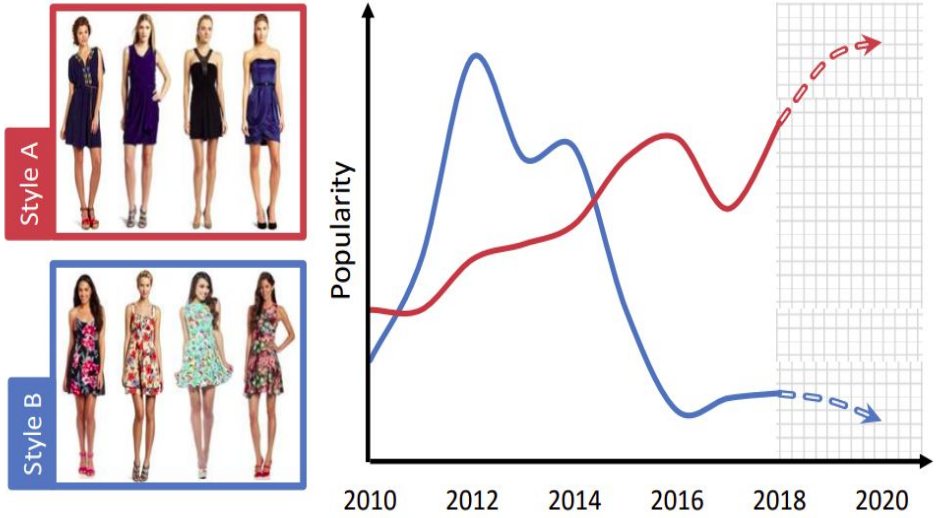
Using the superb details provided in the fashion magazines, our metric will determine if the product will really reach the sky, and help you grow in the way!

**“The customer is the final filter. What survives the whole process is what people wear.”**

While structured data from vendors is relevant to fashion forecasting, we believe that it is not enough. Fashion is visual, and comprehensive fashion forecasting demands actually looking at the product, just as a real human will do before deciding if he/she will buy the product

# Architecture


Sample Multi Task Model Architecture for our Deep Attribute CNN Model.




Visualisation for the predicted trends for different product verticals.

# Tech Stack and Research

## Research:

GeoStyle: Discovering Fashion Trends and Events, Utkarsh Mall *et al.*

Fashion Forward: Forecasting Visual Style in Fashion, Ziad Al-Halah *et al.*

## Tech Stack:

- **API:**
  - React JS - For UI/UX design in the API
  - Node JS - For backend features, and calling the Machine Learning stack
- **ML:**
  - Tensorflow 2.x - To build scalable Machine Learning models
  - GoogLeNet - A CNN which we are using as the base for our Attribute Classifier Model.
  - StreetStyle 27k- A dataset containing 27,000 images, which we use to finetune our GoogLeNet
- **Web Scraping:**
  - Beautiful Soup 4: A Python library made specifically for web scraping.
  - Selenium: For automating web scraping process.
  - Using these, we will retrieve the data from the relevant elements of e-com websites/ fashion websites, and extract images for initial prototype, and star rating, review data, etc. when available at a later stage.


Flipkart
GRID 2.0