#TrendingFashion

Grace Park
Caltech '16
gehpark@gmail.com

JD Co-Reyes
Caltech '16
jdcoreyes@gmail.com

Ella Mathews
Caltech '15
emathews@caltech.edu

Aditya Karan
Caltech '16
akaran@caltech.edu

ABSTRACT

Jumpsuits, maroon ankle boots, and colorful prints are "in." Plain black leggings, pastel colors, and Ugg boots are "so last season." Finding out what's currently in style is not always easy. The #TrendingFashion project endeavors to solve that problem. Our site leverages social media to provide users with up-to-date information about current fashion trends.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Fashion Detection

Keywords

Fashion, Data crawling, Classification

1. INTRODUCTION

Fashion has always played a large role in culture. It can represent anything from a nationality to an individual's sense of self. Despite its importance, fashion is not always easy to follow. In the past, the fashion industry defined what was trending. Designers and clothing stores predicted the trends in fashion years before the clothes actually hit the market. These creators were the major sources of new styles, and strongly influenced the direction of the trends. Consumers were limited to following celebrity fashion, runway models, and their close friends' styles. Trends and sources were well defined. Sites and magazines like Vogue and Elle acted as the authority, directly telling people what was "in" without any way for the user to provide feedback.

The rise of social media has changed the face of fashion. Sites such as Pinterest, Twitter, and Instagram provide platforms for users to interact with each other and form tightly connected networks. These networks allow for a bidirectional spread of trends between consumers and producers of fashion. Social media offers instant feedback for the designers and clothing lines, which then affects the future production and direction of the trends to come. With the influence of social media, trends may pop back up as fast as in a few weeks or elongate the period of time it takes to resurface. Reposting and widely spreading a certain trend through social media might drastically extend the duration of popularity - Van's shoes or denim jackets, for example. On the other hand, the constant posting of that fad may wear out the novelty and edge of the trend and flush it out of the network faster.

The rapidly changing structure of the industry leaves newcomers to fashion at a disadvantage. No longer is there an obvious source for fashion inspiration and guidance. Runways are fantastic sources for the professional fashionista and fashion show goer, but runway fashion is often bizarre and outlandish. Magazines cannot keep up with the speed of online trends. Even social media itself is lacking. Sites like Pinterest and Instagram make following individuals' styles simple, but don't provide a way to see a bigger picture of the fashion world. Therefore, we wanted to provide a source of fashion inspiration and trend-tracking that directly interacts with social media without requiring background knowledge of the users.

2. SOCIAL MEDIA SOURCE

The first step in implementing this project was choosing a social media site from which to collect fashion data. After looking at many other options such as Pinterest, Facebook, and Twitter, we found Instagram to be the ideal option.

2.1 Instagram Basics

Instagram is a photo-based sharing site where users can upload photos, follow other users' accounts, and like and comment on other photos. Photos are also tagged with relevant topics such as "#fashion" or "#katespadepurse." Users can also explore photos based on these tags. Displayed photos are ordered by the time they were posted.

2.2 Choosing Instagram

There were three main reasons we picked Instagram over the alternatives. First, the photo-heavy nature of the site lends itself well to fashion related posts. Text-only posts that might appear on sites like Facebook, Twitter, or Tumblr are

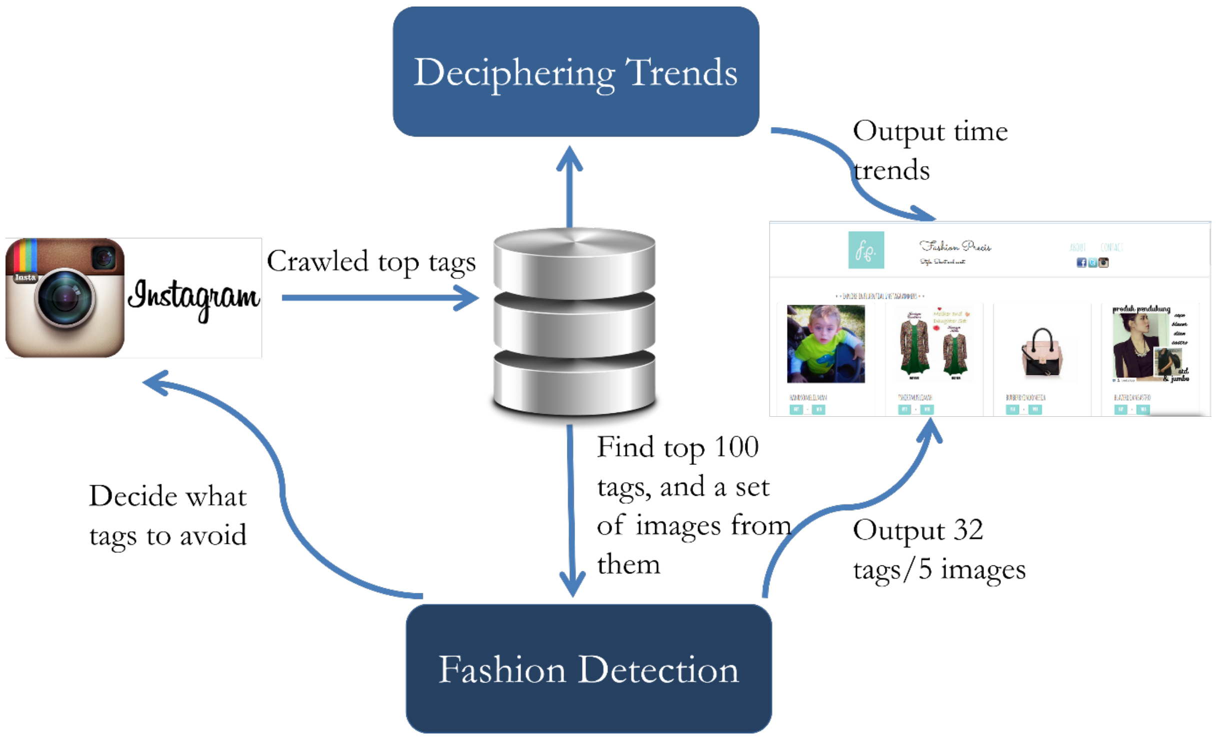

Figure 1: Overall infrastructure layout. Posts are crawled throughout Instagram, which we then store in Dropbox. This data is used to generate time trend data as well as filtered out using fashion detection. These results are combined to output to the website.

not as useful for users trying to get a sense of hot trends, given that fashion is an aesthetic subject.

Second, we wanted to take advantage of the tagging style of Instagram. Each post can have many tags, and they are defined by the poster. Tags provide us with a baseline filter for fashion-related posts.

Third, we found that the API Instagram provides best suited our project's needs. It allows our crawler to access posts at a rate that makes updating trends often more feasible than the APIs provided by other sites that were otherwise suitable, such as Pinterest.

3. DATA COLLECTION

We detect trending tags on Instagram and then find popular photos for each trending tag. We also find influential users relation to fashion. This process involves crawling tag count data, detecting trending tags, crawling for posts with high numbers of likes, and detecting users with both high number of fashion related posts and likes.

3.1 Tag count crawler

The Instagram API gives access to tag data like counts and recent posts, media data including comments and likes, and user data which includes number of posts and likes. Because the API only gives the counts of a given tag at that particular moment, our system must crawl the counts of any tags related to fashion periodically. We start with a list of approximately 1000 seed tags related to fashion and clothing compiled from an initial crawl of Instagram and from a list of fashion brands. Every hour, the crawler requests the tag count (number of posts with that tag) through the Instagram API and saves the data. Each request for the counts of a single tag also returns the tag counts of 19 other similar tags so we save this additional data. The crawler records the counts of approximately 20,000 different tags along with the time stamp. The API does not give geo-specific count data and so the counts take into account posts globally.

3.2 Trend Detection

We first filter out tags based on a shadow tags list. This list has been generated from the image filter component (described below) and from manual inspection of the tags. We then use the Holt-Winter's method to forecast popular tags. The Holt-Winter's method performs triple exponential smoothing to take into account small fluctuations and any seasonal trends in the data. In our case, season could cyclical patterns over the day like posting being more popular at night or weekly patterns like more posts occurring on weekends. Suppose we have a sequence of counts $\{x_t\}$ beginning at time $t = 0$ with a cycle of seasonal change of length L. We let $\{s_t\}$ represent the smoothed value for time t and $\{b_t\}$ be the sequence of best estimates of the linear trend superimposed on the seasonal changes. $\{c_t\}$ is the sequence of season correctional factors and c_t is the expected proportion of the predicted trend at any time t mod L. F_{t+m} is the final output and the estimate of the value of x at time $t+m$ based on the raw data counts up to time t. Then triple exponential smoothing is given by the following formulas [1].

$$s_0 = x_0 \tag{1}$$
$$s_t = \alpha \frac{x_t}{c_{t-L}} + (1-\alpha)(s_{t-1} + b_{t-1}) \tag{2}$$
$$b_t = \beta(s_t - s_{t-1}) + (1-\beta)b_{t-1} \tag{3}$$
$$c_t = \gamma \frac{x_t}{s_t} + (1-\gamma)c_{t-L} \tag{4}$$
$$F_{t+m} = (s_t + mb_t)c_{t-L+1+(m-1) \mod L}, \tag{5}$$

where $0 < \alpha < 1$, $0 < \beta < 1$, and $0 < \gamma < 1$ are the data, trend, and seasonal change smoothing factors respectively. Usually the minimum amount of time series data required is atleast 2*L since to estimate the initial trend estimate b_0 we use the formula.

$$b_0 = \frac{1}{L}\left(\frac{x_{L+1} - x_1}{L} + \frac{x_{L+2} - x_2}{L} + \ldots + \frac{x_{L+L} - x_L}{L}\right) \tag{6}$$

To get the initial seasonal factors c_i for i from 1 to L we use

$$c_i = \frac{1}{N}\sum_{j=1}^{N} \frac{x_{L(j-1)+i}}{A_j} \tag{7}$$

where A_j is the average value of x in the jth cycle of the time series data.

$$A_j = \frac{\sum_{i=1}^{L} x_{L(j-1)+i}}{L} \tag{8}$$

Using this method on the tag count time series data we can then predict what tags will have the greatest percent change in number of posts. We consider tags that are going to increase the most relative to their own count to be trending tags. From these trending tags we compile a list of top trending tags to crawl. We diversify this list by computing the Levenshtein distance between any two tags a and b given by the following formula.

$$lev(i,j) = \begin{cases} \max(i,j) \text{ if} \min(i,j) = 0, \\ \min \begin{cases} \mathrm{lev}_{a,b}(i-1,j) + 1 \\ \mathrm{lev}_{a,b}(i,j-1) + 1 \\ \mathrm{lev}_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} & \text{otherwise.} \end{cases}$$

Tags that are similar to a word already in the top tags list are not added. So for example, the more popular of 'chanelhandbags' and 'fenihandbags' would be added to the list. This avoids having multiple items of the same type and multile items of the same brand.

3.3 Media post crawler

The media post crawler finds the top posts for a given tag from the top trending tags list computed from the trend detection system. While the tag count crawler runs roughly every hour, the media post crawler runs twice a day. We did not observe any rapid changes in tag counts to warrant crawling more frequently.

Given the crawl period of roughly half a day, the crawler should find all popular posts for a given tag posted in the last twelve hours. The number of posts for a given tag per day can range anywhere to 500 to 25,000. The Instagram API orders media for a tag based on recency which is when a post was tagged with that tag. For a given tag, the crawler finds the 5000 most recent posts and then saves the top 100 posts with the most likes.

3.4 Influential User Detection

Based on the posts the media post crawler finds, we try to detect which users are the most influential. We first filter out users with low numbers of likes or shares. Then we find users that both have posts with trending tags and posts with high numbers of likes. These top users give a sense of who to follow on Instagram for emerging fashion trends.

4. IMAGE FILTERING

Even after looking through fashion related tags, a lot of images gathered were still not relevant for fashion. Images of scenery, cars, etc. often came up. In order to improve the image quality of the website, we implemented a classification scheme to distinguish between fashion and non-fashion images

To implement this, we exploited the Scale-invariant feature transformation (SIFT). To compute the SIFT features, there are four major stages: extrema detection, keypoint localization, orientation computing, and creating the keypoint descriptors.

For efficient detection of keypoints, we need to compute both over space and different scales: at small scales we could see small differences, but over larger scales, we might see a bigger change, such as a foreground, background change. We use a Gaussian as the scale-space kernel

To do this, we compute an approximation to the Laplacian of the Gaussian over different scales. The Lapalacian of the Gaussian, in this case, is

$$\frac{\partial G}{\partial \sigma} = \sigma \nabla^2 G \tag{9}$$

where $G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$ From this, we can do a linear approximation, which gives us

$$\frac{G(x,y,k,\sigma) - G(x,y,\sigma)}{k\sigma - \sigma} \tag{10}$$

Therefore, as approximation, we just compute the difference in Gaussian over the different scales. which is a slightly more efficient approximation. This gives us.

$$\begin{aligned} D(x,y,\sigma) &= (G(x,y,k\sigma) - G(x,y,\sigma))I(x,y) \\ &= L(x,y,k\sigma) - L(x,y,\sigma) \end{aligned} \tag{11}$$

Where $I(x,y)$ is the Image representation in 2D space, and $*$ is the convolution operator, and

Figure 2: Visualization of SIFT keypoints without orientation. Here the circles represent where the SIFT algorithm has determined are the important points. The relative location and later orientation is then used to help cluster and identify different types of images

Once a set of possible keypoints are identified, they are further localized, as to eliminate regions that have low contrast or keypoints that are representing edges instead of useful features. To do this, we take the Taylor expansion of $D(\mathbf{x})$

$$D(\mathbf{x}) = \mathbf{D} + \frac{\partial \mathbf{D}}{\partial \mathbf{x}}\mathbf{x} + \frac{\mathbf{1}}{\mathbf{2}}\mathbf{x}^{\mathbf{T}}\frac{\partial^{\mathbf{2}} \mathbf{D}}{\partial \mathbf{x}^{\mathbf{2}}}\mathbf{x} \tag{12}$$

We then take the derivative of this and find the local extrema $\bar{\mathbf{x}}$ The derivative is approximated by using the differences of neighboring points. For local extrema, with a norm $|D(\bar{\mathbf{x}})| < \delta$, we discard these candidates as areas with too small contrast. In most applications $\delta = 0.03$ serves well.

Once the regions of these keypoints are determined, an orientation to each keypoint is assigned to get invariance to image rotation. Gradient orientations are computed from a region around the keypoint, and binned into 36 bins covering 360 degrees. Each sample is weighted by the magnitude. An orientation histogram is created, and one takes the highest peak in the histogram as well as other peaks that are within 80% of the histogram. While this creates a separate keypoint for the point, the different orientations help distinguish important features within the image

From these keypoints, we create a descriptor for each one, in which we take a 16×16 neighborhood, divided into 16 subblocks, in which we create a 8 b in orientation histogram, which totals to 128 bins, which we interpret at the 128 dimension vector.

From this process, keypoints are identified and localized. Figure 2 has an example of visualized SIFT features. [3]

After generating these keypoints for each image, we implemented a nearest neighbors clustering scheme. From Kambhampati's research on trends on Instagram, the author notices about 10 different classes of images. [2]. We use these categories as the basis of our classification scheme. After fiddling around with the images, we decided to expand this classification to twelve schemes, which was meant to help differentiate between face shots (selfies), full body shots of individuals and group photos.

A representative set of images were used as a model for these categories. Afterwards, we collecting training data based on pulling the most recent post from Instagram periodically throughout the day. After manual classification of about 200 images, and out of sample testing of about 400 images, we came up with our classifier. Out of all of the possible classifications, we let images that were classified as "fashion items", "face", "single person", and "groups" be noted as fashion related items, and the remaining classes as not fashion related items.

We used this filtering mechanism for two purposes. One was to help determine fashion related tags. While crawling, list of images associated with specific tags would be stored. We would go through these large list of media associated with these tags, and classify each one of them as fashion or not

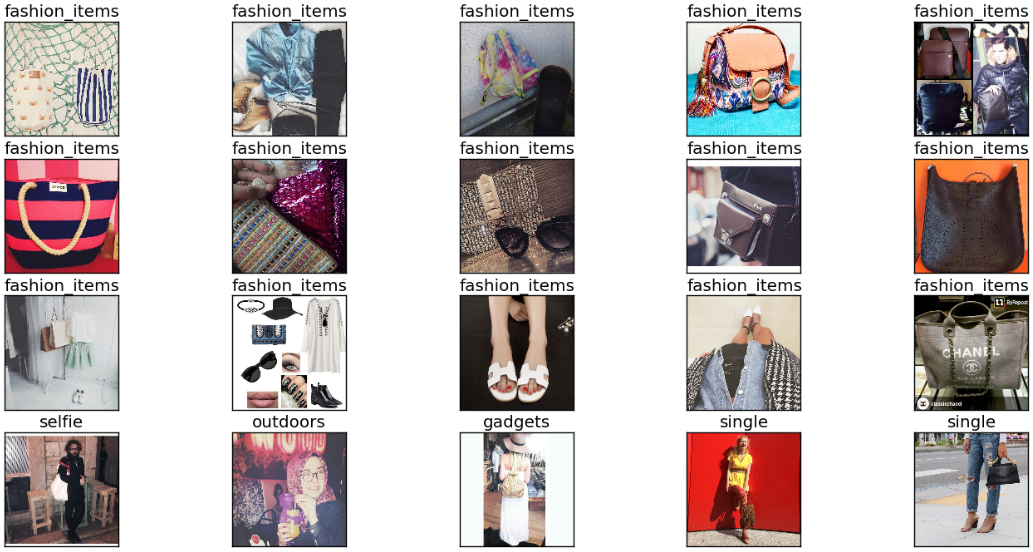

Figure 3: Sample predictions of the classification scheme.

fashion. If the percent of these images were below some threshold, we would consider this tag, not fashion, and exclude the tag from future crawling. In this manner, the image classification would feedback into the crawling, and overtime improve crawling results.

The second use for image based classification was to narrow down top images. The initial crawling would produce a set of 100 tags with 10 images each. The image filter could go through each of those tags, and determine if those images were also fashion related or not. This was used to help select which tags, and images to display onto the final website, tags that had a high amount of non-fashion related images would be removed, and even tags that tended to have mostly fashion related items, would have non-fashion related items removed as to ensure proper image quality.

5. TREND VISUALIZATIONS

We used d3.js to create visualizations of the trends we found with respect to time. The graphs compare total count of photos using a given trending tag over time. Note that this means the counts are cumulative over time.

5.1 Comparing Trends

As a measure of accuracy, we can compare our graphs to Google Trend results for the same topics. Google's trend reports display interest in a topic at any given time based on the number of times it is searched. The graphs are then normalized to a range of zero to one-hundred. This means there are no absolute numbers, but the popularity of different topics can be compared.

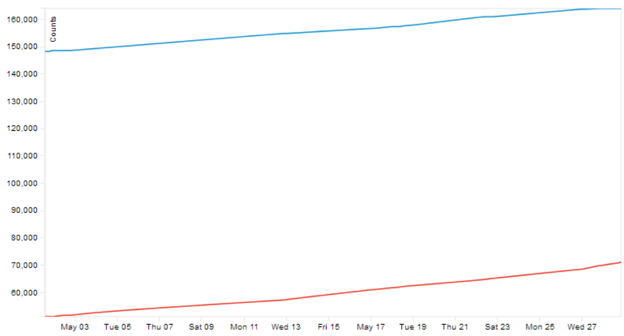

Figure 4: D3.js trend visualizations for #katespade-bag and #fendipeekaboo.

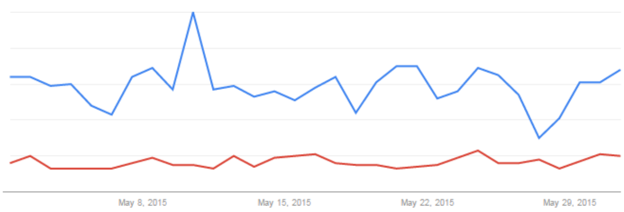

Figure 5: Google Trend results for "Kate Spade Bag" and "Fendi Peekaboo".

We can see that the two graphs show similar results. Both Kate Spade bags and Fendi Peekaboo bags received a reasonably consistent amount of attention for the given time frame, and both graphs show that Kate Spade was consistently more popular.

5.2 One-time Events

Given our method of trend detection, our site is particularly receptive to event-based trends. These are trends that appear very quickly based on some specific event, and disappear very soon after.

An example of this is the popularity of Floyd Mayweather's clothing brand on Instagram the weekend of the Mayweather-Pacquiao fight in early May. Both our data and Google Trends show the spike of attention the topic received just after the fight on May second.

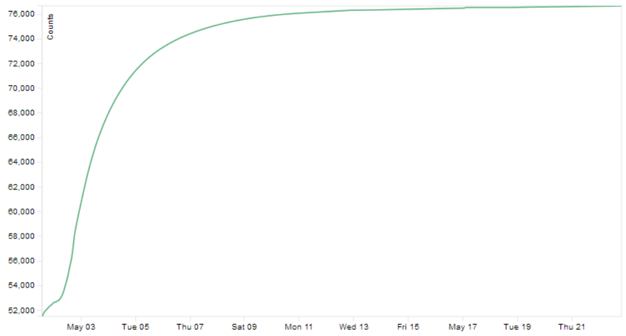

Figure 6: D3.js trend visualization for #moneymayweather. The topic saw a rapid rise in poularity, but, as indicated by the plateau, had a short lifespan as a popular trend.

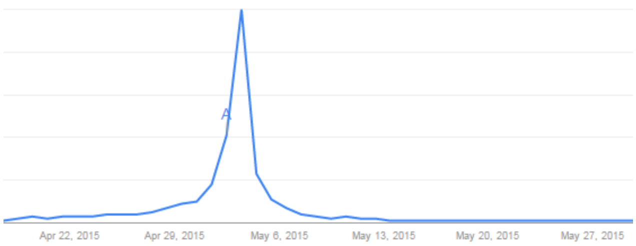

Figure 7: Google Trends for "Money Mayweather". Note that these results also include searches for "Money Mayweather" regarding the person, and not necessarily the the clothing brand. However, the same spike in attention seen in the fashion data graphs on our site is displayed here as well.

6. FRONT-END

The product has been deployed to the URL: www.fashionprecis.com with the name "Fashion Precis."

6.1 Server set up

The website is hosted by Heroku. The backend is started using Node.js express module, and the website itself uses jade, html, bootstrap for layout, and css for styling. Although a database is set up, the data transfer is so minimal that we store our information on Dropbox. The website is auto-deployed every two hours to deliver current trending posts.

6.2 Interface

The main page of the website displays thirty-two top trending Instagram tags with relevant photos, Google Trends graphs, and a link to buy an article of clothing related to the tag.

The website also has pages which display top ten fashion related Instagrammers for the user to directly follow as well as the trend visualizations discussed earlier.

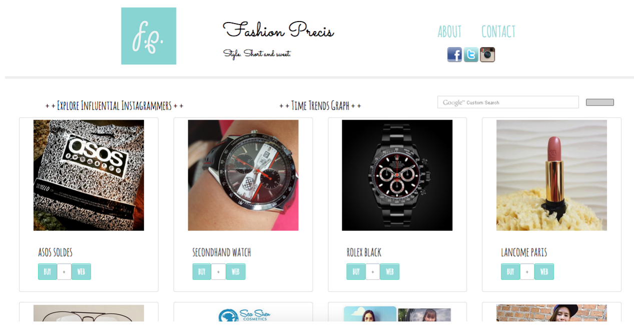

Figure 8: Top 32 Image page. Users can scroll through these images and, if any of the items appear interesting, the user can click to learn more.

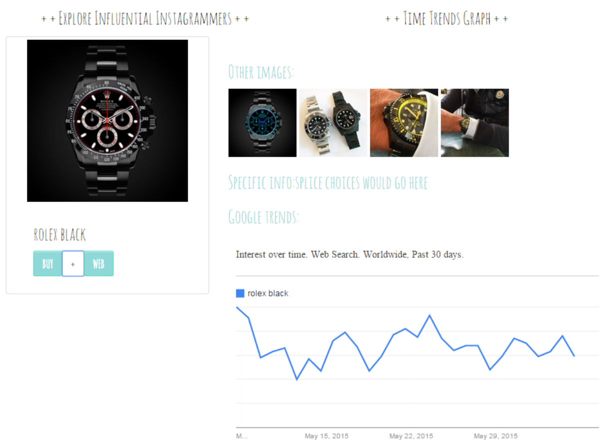

Figure 9: After clicking on a picture, the user gets other related pictures as well as Google Trend information

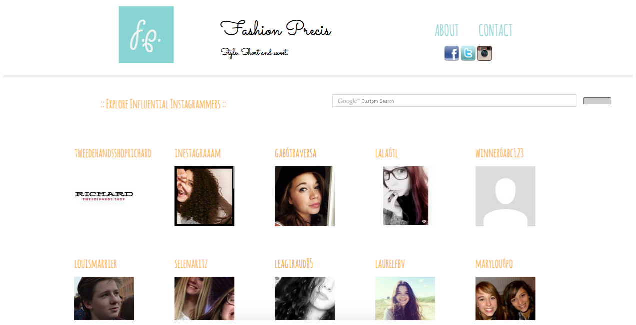

Figure 10: Top Instagrammers page on the website. These are determined by the users that have a high number of likes or shares and have fashion related tags

The design is minimalistic to match the current trends in UI design and to allow the user to focus on only the important contents.

7. FUTURE WORKS

Currently, this product utilizes data from Instagram, due to the ease of use of its API and relatability to fashion content. To correctly represent the trendiness of a style correctly, information must be gathered from a variety of sources as initially planned. Additional potential sources include other social media: Facebook, Twitter, and Pinterest; designers: Michael Kors, Kate Spade, and Marc Jacobs; and stores: forever21, Urban Outfitters, and Anthropology, to name a few. A weighting system would be implemented to give different sources an appropriate level of influence on determining what is actually trending.

7.1 Other capabilities

Fashion is an idea that cannot be generalized to fit everyone. The future product would include user-specific personalization.

The user can filter by the type of article of clothing that they wish to explore. For example, a high school senior looking for a prom dress would filter only by the keyword "dress."

The user can input their age, gender, and location for the product to automatically suggest clothing fitting for that group of people. Alternatively, the user can click a variety of keywords and/or images that best describe their style, and from those choices, the product would find fashion that the specific user would enjoy. The user can continue to give feedback by clicking an up or down button to allow the product to learn what is the user's style really.

7.2 Item Segmentation

In the literature, efforts have also been made to identify specific fashion related objects in photos. [4] In the future, it would, therefore be useful, to look through images on social media, and identify the type of fashion item a certain picture has (boots/scarves, etc.) and identify trends purely based on the type of items within the image.

7.3 Marketing

This product caters to the social media users and utilizes the significant demand in fashion information of recent times. By featuring products, styles, designers, and stores that sell or designed the particular article of clothing that is trending at the time, the sources of this fashion benefit from the advertisement. This way, the product brings in income from the fashion sources who would actively like more advertisement from this website or app.

Another option that the product provides is a method to purchase the item immediately. Retail stores will take advantage of this redirect and pay to be sponsored on the website or app as a seller for particular articles of clothing.

8. ACKNOWLEDGEMENTS

We would like to thank Adam Wierman for all of his feedback and guidence for this project.

9. REFERENCES

[1] C. Croarkin. Nist sematech e-handbook of statistical methods, 2013.

[2] Y. Hu, L. Manikonda, and S. Kambhampati. What we Instagram:A First Analysis of Instagram Photo Content and User Types. *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media*, pages 595–598, 2014.

[3] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[4] K. Yamaguchi, M. H. Kiapour, L. E. Ortiz, and T. L. Berg. Parsing clothing in fashion photographs. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3570–3577. IEEE, 2012.