Predictagram Abstract

Team Name: MALIGN
Logins: nahmed12, mgans, ihasan2, ljiang15

Goal

To musicians on Instagram, the number of likes on a post is a virtue signaling tool that indicates marketability and popularity. In the wake of Instagram's shift to eliminate the 'likes' count for posts, the need for a tool to accurately predict the number of likes on a post will arise amongst managers, record labels, and others who use this metric as a proxy to understand artists' engagement with their fan base. Therefore, we set out to predict the number of likes a post by a musician will accumulate given data about Instagram posts made by musicians in the same year.

Data

To obtain our data, we found a list of 500 music influencers on Instagram from https://hypeauditor.com/top-instagram-music/, and hand-selected the users that would be relevant to this project. That is, we chose users who were actively creating music and sharing content related to their work. We used an online Instagram scraping tool (found on apify.com) to scrape information on these users' posts over a one-year range. After cleaning for outliers (i.e. posts with an abnormally high or low number of likes, and users with abnormally high or low post counts), we were left with data pertaining to 42,165 Instagram posts from 323 unique users.

Model and Evaluation Setup

We seek to generalize our prediction across Instagram users in our dataset, so we train on 33,732 posts and test on 8,433 posts, maintaining the same distribution of users across train and test sets. Thus, we also ensure that there is no overlap between the training and testing sets. The model being trained is a ridge regression, and it is evaluated using the reported root mean-squared error and R-squared value. The features used to predict the number of likes include scraped features (such as number of followers, number of posts), extracted features (time of day, day of week, month of year extracted from the timestamp), and generated features (for example, "normalized likes" that controls for linear regression trends over the course of the year per user and features obtained from Natural Language Processing). The RMSE and R-squared of the ridge regression is compared to a baseline model.

Results and Analysis

Claim #1: The classifier trained using all features does not outperform the baseline model.
Support for Claim #1: The table below shows the accuracy of our full model compared to a baseline model that predicts the number of likes to be the average number of likes for posts by the same user. The RMSE of our model is higher and the R-squared value is lower. We cannot say whether this difference is significant.

Model	Test RMSE	R-squared
Baseline	291,943.2300	0.6034
Full Model	294,412.4910	0.5967

Claim #2: Most of the predictive power of the model comes from our "temporal likes" feature.
Support for Claim #2: We train several versions of our model, using different subsets of features. We see most of the predictive performance, as evaluated by the RMSE and R-squared metrics, is coming from the "temporal likes" feature (extracts the slope of single regressions comparing a user's number of likes to the timestamp of when they posted). Adding the rest of the features actually increases the RMSE and decrease R-squared.

Feature	Test RMSE	R-squared
Temporal Likes Only	283,954.6241	0.6248
Full Model	294,412.4910	0.5967

Claim #3: Including NLP features actually decreases the predictive power of the model.
Support for Claim #3: We train an additional version of our model that doesn't include the NLP features (specifically a bag of words vectorization of the top 2,000 words that showed up in post captions), and notice that while the other features slightly decrease RMSE and increase R-squared, appending the NLP features results in a worse fit. We cannot determine whether these differences are significant.

Feature	Test RMSE	R-squared
Temporal Likes Only	283,954.6241	0.6248
Full Model w/o NLP Features	283,491.8571	0.6260
Full Model	294,412.4910	0.5967