

WHAT'S HOT: LINEAR POPULARITY PREDICTION FROM TV AND SOCIAL USAGE DATA

Jan Neumann, Xiaodong Yu, and Mohamad Ali Torkamani
Comcast Labs

Abstract

Large numbers of TV channels are available to TV consumers these days. Such choice is both a blessing and a curse, because too many options can overwhelm the consumer and due to the limited screen real estate on devices, only a small number of programs can be presented at a given time. To address this issue, at Comcast Labs we work on algorithms that compute rankings of current and upcoming programs based on various relevance criteria.

In this paper we describe one of our algorithms, where we predict the future popularity of programs by combining information from historical Nielsen ratings, DVR scheduling activity, and social web activity (e.g. Facebook, Twitter).

INTRODUCTION

The question of "What's On TV?" is part of the daily ritual of watching TV. Usually we start by examining "the grid" and surf from channel to channel to find out what programs are playing on what channel. The order of the channels in the channel lineup rarely changes and though it is based on thematic groupings, it does not reflect that the themes and popularity of different channels and programs changes over the course of a day.

To address this issue, at Comcast Labs we developed an algorithm that predicts the popularity of programs that are currently on TV or will be playing in the next 24 to 72 hours. The output of this algorithm is then used to present schedule information to customers in order of (predicted) popularity of a given program and aims to give them an improved user experience (see Figures 1 and 5 for screen shots of Comcast Interactive Media's "What's On" iPhone app).



Figure 1 – Screenshot of "What's On" iPhone App

Currently, the most prominent metric to measure the popularity of TV programs and channels is provided by Nielsen Media. They publish the well-known suite of Nielsen TV ratings. One of the ratings, for example, is the percentage of TV consumers that are currently tuned to the program of interest.

During the last couple of years the consumption patterns of TV consumers are undergoing a rapid change where content is consumed on a range of devices such as cell phones, computers and tablets in addition to the TV. Also audiences nowadays tend to interact socially with TV programs via Twitter, Facebook and other social web sites and such activity can be utilized to further gauge the engagement of the audience with a program, as we will do in this paper.

On these sites viewers of a program indicate their level of like (or dislike) for it by publishing messages related to the program content or actors (e.g. Twitter), give explicit feedback via like/dislike buttons (e.g. Facebook), or even indicate that they are currently watching a TV program (e.g. Zeebox, GetGlue, IntoNow, Shazam, etc.).

Our approach uses machine learning to build a model that combines statistics about past Nielsen ratings, and scheduled DVR recordings, together with current social signal activity to accurately predict the popularity of one program relative to another.

Each of these sources of information captures a different notion of popularity. In the following sections we will describe the sources of information that we are using and the algorithm in more details

NIELSEN TV RATINGS

Nielsen ratings have been used by content providers for a long time to measure the audience participation of TV programs. The Nielsen shares, the percentage of viewers that are tuned to a given channel or program compared to all consumers that use their TV at the moment, are used to judge the success of a program and to set the rates for advertisers. Due to the nature of the data collection, Nielsen audience measurements are only available with some delay for most channels and programs since the viewing numbers also include the delayed consumption of programs on the DVR.

Nielsen national channel ratings are determined by monitoring the TV consumption behavior of a small sample of households and then extrapolating these sample statistics to the universe of all TV consumers in the US. We looked at the average number of TV viewers tuned to a given channel across the US for any 15 minute interval of a given day, which we will refer to as *Nielsen channel rating* from now on, as well as the average number of TV

viewers tuned to a given program, which we will denote as *Nielsen program rating*.

We need to do some preprocessing steps before Nielsen ratings can be used together with the other usage data. Since a Nielsen channel or viewing source corresponds to a number of physical related channels (e.g. all NBC broadcast channels are aggregated in a single “NBC Nielsen channel”, all HBO-East, HBO-West and SD//HD channels map to a single “Nielsen HBO channel”), we semi-automatically created a mapping between the physical stations to the Nielsen aggregate channels.

Also, since Nielsen does not use unique ids to identify programs, we need to match a given program in the schedule to the corresponding program in the Nielsen ratings report. This is implemented using a combination of editorially created regular expression matching together with natural language based distance metrics. After establishing correspondence we find the ratings for the program at the same time and weekday for a fixed number of preceding weeks.

The same process is repeated for the channel popularity. In our experiments we utilized six months of Nielsen national channel and program ratings.

DVR SCHEDULED RECORDINGS

Using Nielsen TV ratings to predict popularity of programs that were never or not recently aired on TV is challenging, because there are simply not enough samples available for an accurate prediction. Examples of such programs are yearly awards shows such as the Oscar’s, Emmy’s, Grammy’s, large sporting events like the Olympics, NFL or NBA playoffs, news breaks, and newly scheduled programs

To be able to deal with such programs, we utilized the DVR scheduling statistics to count how many customers have scheduled their DVR to record a given program.

We compute the DVR score by aggregating the number of scheduled recordings for specific episodes as well series across all users that are stored in Comcast’s online DVR scheduling service.

While doing so we make sure to account for differences in the number of customers in different markets, so that we arrive at a normalized DVR score that can be integrated with the remaining scores.

SOCIAL WEB ACTIVITY

There are two main types of information that can be gathered from social networking web services. One examines the connections between different participants in a social network (e.g. friends, followers), and the other looks at the activity between the participants in such networks. In the approach described in this paper, we only considered social activity based measurements since we are interested in aggregate popularity estimates, not personalized recommendations.

According to a recent Nielsen/SocialGuide study there is a strong correlation between the Twitter activity related to a program, as measured by tweets containing the hash tags

associated with it, and TV ratings [1]. The study found that for young adults (14-34 years old), a 8.5% increase in Twitter activity correlated with a 1% increase in TV ratings for premiere episodes, and a 4.2% increase in Twitter activity correlated with a 1% increase in ratings for mid-season episodes. For older TV consumers this effect was weaker, but still present (i.e. a 3.5% increase in Twitter activity correlated with a 1% increase in ratings).

In contrast to watching TV, information about participation in social web services is usually made available to third-parties via APIs. For example, when a someone tweets a message related to a TV program, Twitter makes this message instantly available on its message feed and a third party can easily analyze and filter the information and make aggregate information available in real-time.

We used an external company to provide us with the aggregate counts of Twitter and Facebook activity for the time period a program aired on TV +/- 3 hours for various markets. See Figure 2 for some example data.

As with the DVR score it is important that the social activity signal is normalized with respect to the number of participants in

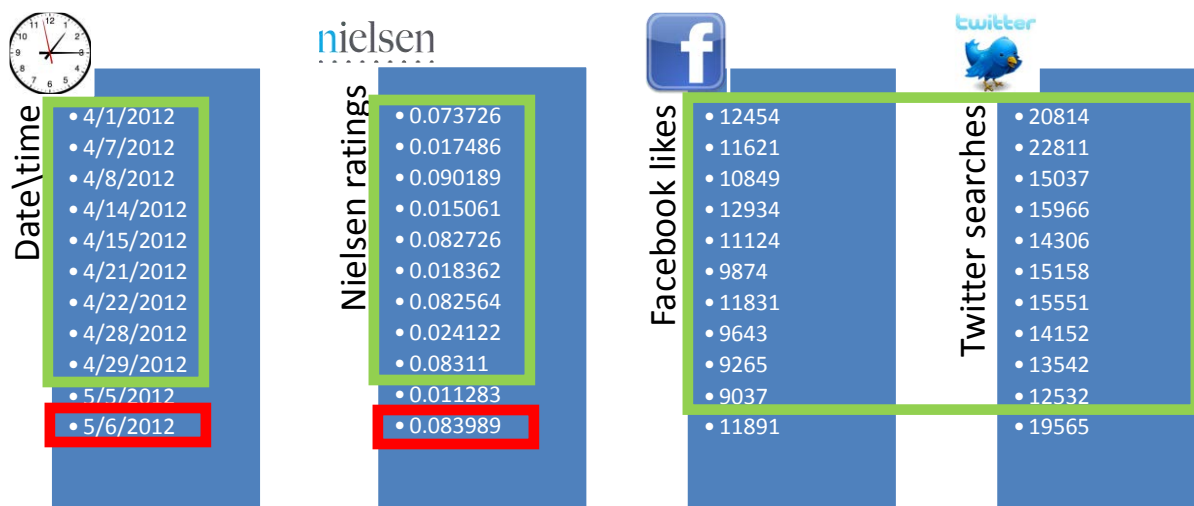


Figure 2 – TV rating prediction model: we use past Nielsen ratings, and current + past social activity signals associated with a program to predict its future rating (red is target value, green are input feature values for the regression model)

For each program and 30min time interval/date do the following:

1. **Extract sufficient statistics:** for each popularity score (Nielsen, DVR, Facebook likes, Twitter activity,...)
 - a. Find Nielsen program scores for 7 last airings of a program, if program scores for less than 7 prior programs can be found, use channel scores instead.
 - b. Find social signal and DVR scores for current program
 - c. Compute the following statistics: Max, Mean, Median, Last value, Mean of the last 3 values, median of the last 3 values
2. **Model Estimation (only during training phase):** Train a regression or classification function for past airings of this program for which we have data, use historic Nielsen program and channel scores as target variables.
3. **Prediction:** Based on the trained model predict the current program popularity.
4. **Ranking:** Based on the predicted scores, sort the programs.

Figure 3 – “What’s hot” prediction algorithm

different geographical regions, so that the scores we use can correctly be used to predict popularity for a target distribution whose statistics differ.

TECHNICAL APPROACH

To predict the future popularity we have to build a model of how the different sources of information about customer activities predict future popularity. We start with the schedule for the upcoming 72 hours, identify all the programs for each station that are playing during each 30 min interval and collect relevant historical information for the different sources, e.g. Nielsen channel and program ratings, number of scheduled recordings of a given program, and the associated social activity signal.

Combining these different scores into a consistent ranking function is not straight forward, since not every score is available for each program, and scores differ in how much they change over time or correspond to different embodiments of user behavior. For example, the coverage of program ratings by Nielsen is only about a third of the programs that are scheduled for a given 24 hour period, while Nielsen national channel ratings are

available for about 120 channels that cover 90% of the programs that are typically being watched. On the other hand, the distribution of DVR scheduled recordings is much more peaked, than the distribution of Nielsen ratings across programs. This is likely due to the fact that a customer only schedules a handful of programs for recording, while not being as selective while browsing the TV.

Using future Nielsen program and channel ratings as the target variable, we compute a range of statistics on each input, which is then used as a feature in a regression or classification framework to approximate the target variable as closely as possible.

This prediction component is then input into a temporal filtering framework to compute the final ranking function that is used to sort the programs. The full high-level algorithm that we implemented is described in Figure 3.

PREDICTION MODEL

We will start by defining the notion of a rank function. Our goal is to learn a function f , so that $f(x) > f(y)$ if program x is supposed to be ranked higher than program y . We explored a number of approaches to learn such a ranking function. The function f can be

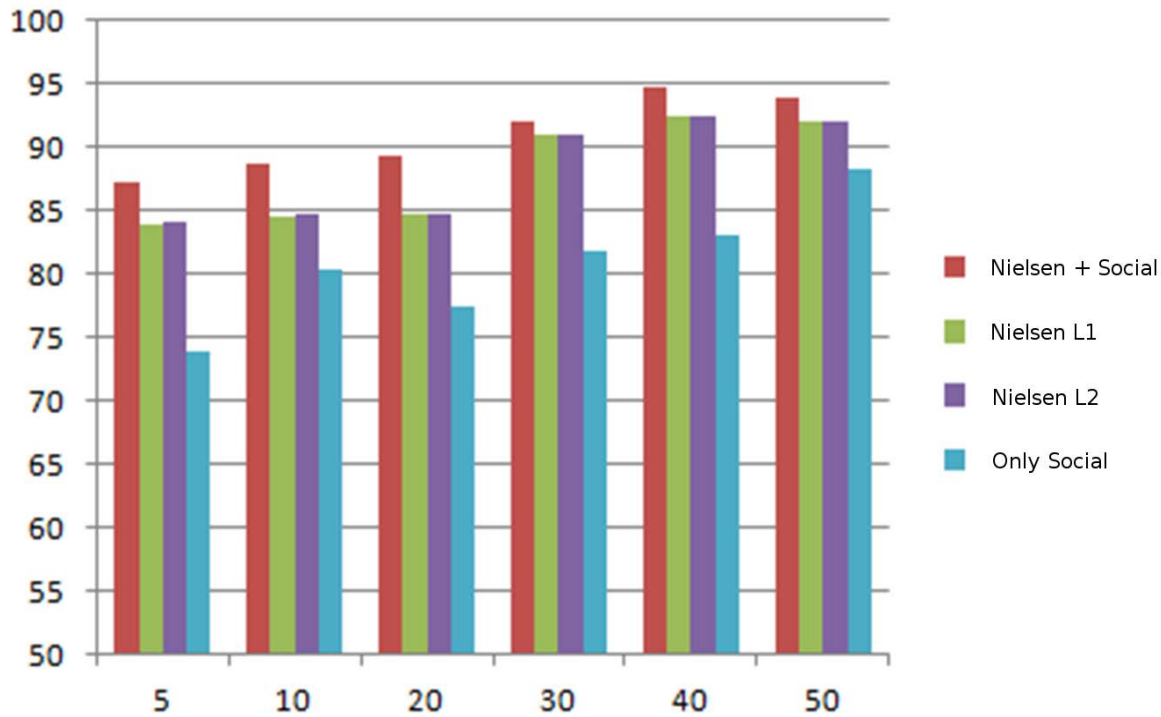


Figure 4 – Top k accuracy of “What’s hot” prediction for different input sources

optimized in many different ways. We studied modeling the ranking problem as a pairwise classification problem, i.e. find a classification function that returns a positive value if x should be ranked higher than y , and negative otherwise. We also looked at regression functions to model the ranking f directly. For the classification approach, we explored support vector machines, k-nearest neighbor approaches, as well as a random forest classifier [2]. For the regression models, we looked at linear models (both with L1 (absolute value) and L2 (least squares) regularization terms, k-nearest neighbor regression, support vector regression, decision trees, and random forest regression [2]. At the end we got the best results using shallow random forest regression trees with past Nielsen scores and current and past social signals as feature inputs as described in the (see Figure 2 for an illustration of the input features and target variables).

EVALUATION AND RESULTS

To evaluate our system, we take the true Nielsen scores as our gold standard and evaluate our predicted popularity ranking against it. We used viewing data from June 2012 to train our predictors and predicted the ranking of programs for every 30 minute interval for the first week of July 2012. The evaluation criteria we are using is the Top-k criterion, i.e. how many of the top k programs of the ground truth data can be found among the top-k programs of the predicted data set. In our experiments we varied k from 5 to 50 in increments of 5 (see Figure 4). As described before, we got the best results using random forest regression trees, but we varied the set of input features that we considered. The results are summarized in Fig 4. One can see that social network activity by itself does not perform very well compared to using a moving least squares (L2) or robust estimation (L1) using a window of Nielsen

ratings. If we combine both historical Nielsen ratings using our non-linear temporal filtering framework with random forest regression trees and all the social signals leading up to the show, then we can get a 4% increase in Top-10 accuracy over only using Nielsen ratings for prediction.

APPLICATIONS

The prediction of the most popular program for a customer has many applications. To name just one, some example screen shots of the “What’s On” app, developed by Comcast Interactive Media, can be seen in Figures 1 and 5. This app allows a customer to see what is currently or soon showing on TV, sorted by different criteria such as most popular, favorite channels, movies, etc. The output of the algorithm can also be used in any other set-top box and mobile application where we want to return a popularity-ranked list to the customer.

CONCLUSION AND FURTHER WORK

In this paper we presented an approach that combined Nielsen ratings, DVR schedule information, and social networking activity measurements in a temporal filtering framework to predict the popularity of future programs. The experimental results showed that combining TV ratings with measures of social network engagement leads to more accurate predictions for relative popularity rankings of TV programs than just using TV viewership numbers alone.

The framework we described in this paper can be extended in a number of ways. For example, one could design more complex models to predict a program’s popularity that incorporate both program related attributes and other non-TV measures of popularity. Examples of program attributes are indicators if the program is a new program or if the episode of interest is the season premiere, what genres a program is associated with, the actors in it, directors for movies, etc. Other

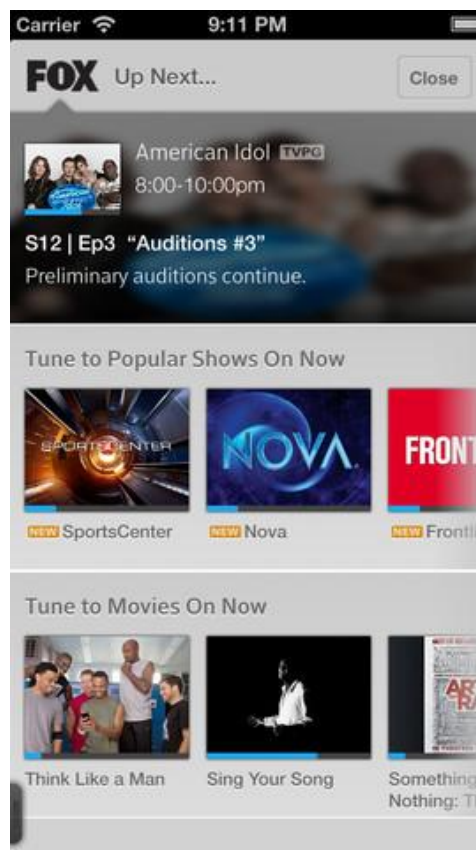


Figure 5 – Sample Client App Screen

measures of popularity we are looking at are box office numbers for movies, Rotten Tomatoes reviews [3] and even the presence or absence of editorial recommendations.

Finally, we are also looking at combining the aggregate popularity prediction described in this paper, with personalized recommendation algorithms that take a user’s TV consumption history into account to deliver truly personalized TV recommendations to customers.

REFERENCES

- [1] <http://www.nielsen.com/us/en/newswire/2013/new-study-confirms-correlation-between-twitter-and-tv-ratings.html>
- [2] T. Hastie, R. Tibishirani & J. Friedman (2001), The Elements of Statistical Learning, Springer, New York, NY
- [3] www.rottentomatoes.com

