Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 198 (2022) 94–101

Procedia Computer Science

www.elsevier.com/locate/procedia

The 12th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2021)
November 1-4, 2021, Leuven, Belgium

# Approximating Viewership of Streaming T.V Programs Using Social Media Sentiment Analysis

Haroon Malik[a]*, Elhadi M. Shakshuki[b], Ansar-Ul-Haque Yasar[c]

[a]*college of Engineering and Computer Sciences, Marshall University, WV, USA*
[b]*Jodrey School of Computer Science, Acadia University, Wolfville, Canada*
[c]*Transport Research Institute (IMOB), Hasselt University, Belgium*

**Abstract**

Approximating viewership of a program is essential for broadcast networks and cable (TNT, AMC, and HBO) as they profit by selling advertising space during programming (airing of the show). With the advent and increased popularity of streaming services such as Netflix and Amazon Prime Video, the model for providing television content to viewers has fundamentally changed. However, this change in the delivery model for television programs has also made it more challenging to determine the actual viewership of television programs on streaming services. Because television programs are no longer aired in pre-determined timeslots, viewership shares cannot be determined. To make matters even more complicated, streaming services like Netflix and Amazon Prime Video are proprietary platforms. They have traditionally kept viewership numbers closely guarded and have not released them to the public. This paper proposes a methodology to approximate the viewership of streaming television shows. This is achieved by exploiting data from social networks, i.e., publicly available information and sentiment score derived from sentiment analysis on tweets published about television programs, using Random Forest classifier. The proposed methodology achieved 85% accuracy in predicting the viewership of the streaming shows.



*Keywords:* Sentiment Analysis, Social Media, Random Forest

* Corresponding author. Tel.: +-304-696-5655.
*E-mail address:* malikh@marshall.edu


10.1016/j.procs.2021.12.215

## 1. Introduction

In recent years, there appears to be a fundamental shift in the way television programs and shows are delivered to viewers. For decades, television shows were delivered to viewers by networks (e.g., ABC, NBC, FOX, etc.) and cable (USA, ESPN, etc.) channels through a cable provider. Shows would be broadcasted in pre-determined timeslots. More importantly, viewership was measured through sample audiences and ratings for each television program would be published and made publicly available through services like Nielsen. *Viewership* is an audience measure, i.e., how many people are in an audience of viewers, especially of television, either generally or of a particular kind of program and is a composite of two measures: Rating and Share. *Ratings* are essentially percentages, measure the portion of a given group — be it households, adults 8-49 or women 5-54 — watching a given show. Adults 18-49 is the primary demographic by which ad rates are set for entertainment programming, so it is the most commonly reported (one point in that demo equals 1.28 million people). Share is the percentage of a given group who are *watching TV at that time* and are tuned into a given program. Whereas total viewers is the average number of people watching a program in any given minute while it airs. TV ratings provide valuable insights into how many people are consuming content, as well as how and when they do so.

Approximating viewership of a program is important for broadcast networks and cable (TNT, AMC, USA, etc.), as they profit by selling advertising space during programming (airing of the show) while premium cable and streamers reply on subscription. Thus, all channels use viewer analytics to decide if a TV show is profitable and worth keeping on their schedules. Also, the TV ratings help calculate how much a network can charge advertisers for airtime during specific programs. One of the most important pieces of data used to determine ad pricing is the adult's 18-49 rating share.

### *1.1. Problem Statement*

With the advent and increased popularity of streaming services such as Netflix and Amazon Prime Video, the model for providing television content to viewers has fundamentally changed. Now, viewers pay for subscriptions to streaming services to stream television programs from large libraries. In other words, television programs are no longer aired in pre-determined timeslots but are viewed by the consumer "on-demand."

However, this change in the delivery model for television programs has also made it more difficult to determine the actual viewership of television programs on streaming services. Because television programs are no longer aired in pre-determined timeslots, viewership shares cannot be determined (which has traditionally been the method of Neilson; the de-facto national measurement service for the television industry in USA). Moreover, simple surveys about streaming shows may not be accurate. To make matters even more complicated, streaming services like Netflix and Amazon Prime Video are proprietary platforms. They have traditionally kept viewership numbers closely guarded and have not released them to the public.

### *1.2. Contribution of the Paper*

The paper aims to approximate the viewership of streaming television shows by exploiting data from social networks, i.e., publicly available information. This is important for many reasons:

- Streaming networks are dumping large piles of money (sometimes upwards of $10 per episode) into television programs for their streaming networks in hopes of attracting subscribers. Finding a reliable way to approximate the viewership of these streaming programs will give investors a glimpse as to the actual viewership of these programs to determine if the large investment in the production of the programs is worth it.
- The approximated viewership of streaming television shows would also be significant to industry personnel, such as actors, scriptwriters, producers, and directors, who depend on viewership financially. Indeed, such industry personnel may have a direct financial tie to viewership or may need significant viewership in their current programs to be perceived as successful in obtaining future work.

## 2. Proposed Methodology

The proposed methodology is data-driven. The section details the steps involved in mining the data from social networks and building a prediction model to approximate the viewership of the TV programs.

### *2.1. Collecting TV Shows Data*

Table 1 lists the fifteen most popular shows (over 5,400 hours) of 2020. The shows were mined from streaming services for only one season. Each television show generally ranges between eight and 12 episodes per season.

Table 1 . Derived features of the shows from social media

| No | Shows | Provider | Neilson Rating | Cite Score | Audience Score (%) | Sentiment Score | No. of Tweets |
|---|---|---|---|---|---|---|---|
| 1. | Lovecraft County | HBO | 0.2 | 5 | 45 | -30 | 536 |
| 2. | The Undoing | HBO | 0.8 | 5 | 50 | 30 | 192 |
| 3. | I May Destroy You | HBO | 0.6 | 6 | 65 | 35 | 500 |
| 4. | Better Call Saul | AMC | 0.5 | 3 | 25 | -10 | 3,000 |
| 5. | Brockmire | IFC | 0.3 | 5 | 62 | 44 | 2,690 |
| 6. | Better Things | FX | 0.8 | 9 | 88 | 89 | 1,699 |
| 7. | Big Sky | ABC | 0.5 | 6 | 65 | 30 | 3,690 |
| 8. | Perry Mason | HBO | 0.4 | 4 | 72 | -5 | 352 |
| 9. | Fargo | IFC | 0.9 | 10 | 89 | 78 | 257 |
| 10. | Curb Your Enthusiasm | HBO | 0.7 | 6 | 75 | 80 | 950 |
| 11. | P-Valley | Starz | 0.8 | 6 | 82 | 45 | 1,001 |
| 12. | The Good Lord Bird | Showtime | 0.6 | 5 | 75 | -29 | 8,200 |
| 13. | Star Trek Discovery | CBC All Access | 1.0 | 10 | 97 | 93 | 30,000 |
| 14. | Star Trek Picard | CBC All Access | 0.8 | 5 | 54 | 0 | 7,000 |
| 15. | The Mandalorian | Disney Plus | 0.6 | 5 | 60 | 10 | 690 |

The collected data for the shows (interchangeably called T.V programs and streaming services in the paper) falls into two categories:

a. A set where the Neilson ratings for viewership were known against which the prediction of the methodology can be validated. Section 2.2 details the Nielsen ratings.

b. A set of programs for which the Neilson ratings and actual viewership were not known.

Notably, the program listed in Table 1 is chosen to evaluate our proposed methodology for two reasons:

1. First, all the programs fall in the "prestige television" category, i.e., television shows that air on premium or upscale cable networks. Thus, all the shows are similar in ratings for the comparison's sake.
2. Second, the shows listed in Table 1 were the only shows/dramas aired on the premium network in 2020 for which the corresponding ratings were available. Due to the COVID-9 pandemic, television production in the year 2020 was scaled back. Moreover, more and more dramatic content is being moved to streaming platforms such as Netflix, Amazon Prime, Disney Plus, Peacock and HBO Max from network television. Indeed, most cable networks focus on reality TV shows, sports, or re-runs instead of releasing new content.

To avoid skew in the data, no reality television shows, or sporting events were chosen.

### 2.2. Neilson Ratings and Classification

A custom-based script was written to crawl social networks, especially Twitter and to mine the Neilson Rating of all the episodes of each program listed in Table 1. Neilson TV ratings are the audience measurement system operated by Neilson Media Research that seeks to determine the audience size and composition of television programming in the United States [1]. It is the standard for measuring the viewership of networks (ABC, NBC, CBS, FOX) and cable TV (USA, TNT, ESPN, etc.) programs in the United States. Ratings are measured by "shares." More specifically, one Neilson rating point corresponds to 1% of all households in the United States or roughly 1.15 million homes tuning in.

The programs listed in Table 1, generally ranged from a 0.2 share and a 1.4 share Neilson rating per episode. All the programs were further categorized into two groups: (1) episodes with less than a 0.5 Neilson share rating, and (2) episodes with above a 0.5 Neilson share rating. The Neilson Program ratings are used as the Target variable/class variable in the prediction model used by the proposed methodology.

### 2.3. Tweets Collection

To gauge the sentiment of the viewers for the programs listed in Table 1, corresponding tweets for each program were collected from Twitter using a simple keyword search for a specified period of time, i.e., for two days after the program's episode has aired. The rationale for choosing the time window is based on the fact that viewers reaction fades away after two days of watching the episode. Also, most viewers watch a television show when it is released or within no more than a couple of days of its release. Twint, an advanced Twitter scraping tool written in Python, was used for collecting the corresponding tweets of the program. Twint was used instead of Twitters proprietary API because of the following: (a) It can bypass the rate limitation Twitter possess for the collection of tweets, (b) It operates by utilizing Twitters search on its page and facilitate scraping Twitter front end based on keyword search, much needed for us to collect the corresponding tweets of the shows, and (c) Twitter API, which provided harvesting of tweets going back seven days [4] makes pulling historical tweets about the shows for our study not possible. Additionally, Twitter limits the number of times a user can query its API. Our methodology requires thousands of API queries. Thus, the use of the Twitter API was again impractical. The Twit takes care of both the listed challenges.

#### 2.3.1. Preprocessing of the Tweets

Twint obtains a host of variables and metadata as a 'Jason' object. Therefore, preprocessing of the data obtained from Twitter is required prior to conducting sentiment analysis. The preprocessing included are as follows:

(a) Removal of special characters. With the exception of # and @) and hyperlinks in the body of the tweets.
(b) Removal of duplicate tweets.
(c) Removal of any tweets, containing foreign language, that the sentiment analyzer would be unable to process.
(d) Truncation of all text to lower case to ensure consistency among the tweets.
(e) Removal of any tweets that the official Twitter sent out accounts for the television shows or streaming networks in the sample to prevent skewing of the sentiment analysis.

#### 2.3.2. Sentiment Analysis

Sentiment analysis is performed on the pre-processed tweets using a Python-based text processing tool called TextBlob [2]. The tool provides several other robust features such as Natural Language Processing, Classification (Naïve Bayes and Decision Tree), Tokenization, Parsing and Word Inflection and Lemmatization, to name a few much needed for building a sentiment analyzer [9].

TextBlob provides a built-in model for conducting sentiment analysis. Nevertheless, we trained a model (in TextBlob) using Naïve Base on the five thousand tweets that were manually labeled by the two members of our research group. Disagreement on the sentiment of the tweet was resolved by a third member of the research group with strong expertise on the sentiment analysis topic. The trained model is used in the supplement to the sentiment analysis text model already existed in TextBlob.

The sentiment analyzer assigned each tweet a notation of "positive," "negative," or "neutral". It also assigned each tweet a sentiment score. Any sentiment score that was positive meant that the tweet was a favorable reaction to the corresponding T.V program — greater the number, the stronger the positive sentiment. Any sentiment score that was zero meant that the tweet was neutral. Any negative sentiment score indicated that the tweet had an adverse reaction. The more negative a number, the stronger the pessimistic and adverse sentiment.

An overall sentiment score for each episode of the program is created by aggregating the individual sentiment score of all the tweets associated with the program. Then, it is factored and one of the prediction variables in building the prediction model is discussed in section 3.

### *2.4. Model Generation*

There are several machine learning techniques exist in the literature, such as Neural Networks, Naïve Bayes, and Stochastic Gradient Descent suitable for solving the classification problem. In our proposed methodology we selected Random Forest (RF) – an advanced decision tree. We choose Radom Forest machine learning classifier, because the algorithm outperforms basic decision tree and other advanced machine learning algorithms in prediction accuracy. Moreover, the Random Forests is more resistant to noise in the data that is an important advantage. We expect that the data used in our work to have noise due to its massive size and the length, i.e., thousands of tweets and hundreds of videos' metadata. The Random Forests algorithm requires a limited number of configuration parameters and produces robust and stable models [5].

Finally, often the prediction accuracy of basic decision tree algorithms suffers when many of the attributes are correlated. Given the large number of attributes in our analysis, we need an algorithm that does not suffer from correlated attributes. Fortunately, the Random Forests algorithm deals well with correlated attributes while maintaining a high accuracy in its prediction [7][8]. In contrast to simple decision tree algorithms, the Random Forests algorithm builds a large number of basic decision trees (40 trees in our case). Each node in each tree is split using a random subset of all the attributes to ensure that all the trees have low correlation between them. We use the default random subset value which is the square root of all the studied attributes. The trees are built using 2/3 of the available data using sampling with replacement. The 1/3 of the remaining data is called the Out of Bag (OOB) data and is used to test the prediction accuracy of the created forest. The use of bootstrapping and random selection of attributes at each node greatly improves the accuracy of tree-based classifiers [6].

## 3. Performance Evaluation

In our proposed work, we used OOB data to measure the accuracy of the created forest. Each sample in the OOB is pushed down all the trees in the forest and the final class of the sample is decided by aggregating the votes (i.e., predicted class) of each tree. One major benefit of using this approach is that we will be able to adjust the votes based on the skewness in the data accordingly. Basic decision trees are known to perform badly with highly skewed data since the tree always changes to predict the dominant class. To overcome this problem in a Random Forest, we can assign weights to votes to offset the data skew. For example, in our analysis of approximating the viewership, the ratio of (YES), i.e., < 0.5 Neilson rating to < 0.5 Neilson rating (NO) is 1:16 . Therefore, we assign the weights 16:10 for the YES and NO classes.

To measure the accuracy of the prediction produced by the Random Forests algorithm, we calculate the overall YES and NO misclassification rates. We desire the lowest overall and per-class misclassification rates. The rates are defined using the confusion matrix, shown in Table 2.

Table 2 . Confusion matrix

| True Class | Classified As | |
|---|---|---|
| | YES | NO |
| YES | a | B |
| NO | c | D |

The YES and NO represents the two classes: (a) Neilson rating < 0.5 and Neilson rating >= 0.5. "True Class" column represents the actual number of ratings for the programs/shows. Whereas a, b, c & d under "Classified As" column represent arbitrary values of correctly or misclassified instances by predictor against true class. For example, if there are 100 instances of an attribute for which Neilson Rating was < 0.5 (True class: YES), the classifier may correctly predict 90 instances (a=90) and may predict 10 incorrectly classified instances (b=10) for that class. We further explain how we derived the misclassification rate with the help of Table 2.

- YES misclassification rate. This captures the performance of the forests for show with Neilson rating < 0.5. It is defined as: b/(a+b).
- NO misclassifications rate. This captures the performance for identifying programs with Neilson rating >= 0.5. It is defined as: c/(c+d).
- Overall misclassification rate. This captures the overall performance of the forests for both classes (YES and NO). It is defined as: (b+c)/(a+b+c+d).

### *3.1. Comparison With Other Classifiers*

The Random Forest achieved an accuracy of 85% to predict the viewership as per Neilson ratings shown in Fig. 1. Whereas, Support Vector Machine (SVM) achieved similar accuracy, i.e., 86% but with low harmonic means, i.e., F-measure shown in Fig. 3. The Linear Regression classifier achieved an accuracy of 78%, lower than both the SVM and RF which is shown in Fig. 2. Also, the LR took the most time, i.e., 87 seconds, to achieve a precision of 75%. Unexpectedly the basic implementation/algorithm of a decision tree, i.e., C 4.5 achieved both high accuracy and precision, but at the cost of additional time, i.e., 97 seconds, as shown in Fig. 4. Random Forest required little training to achieve the highest accuracy and it is maintained throughout the 10-k folds. However, in contrast to other classifiers, SVM requires extra training to reach its accuracy equilibrium, as shown in Fig. 5. When we consider precision, the LR converges to its equilibrium state in the least time, i.e., 102 seconds (both for the testing and training phases). Whereas, the basic tree algorithm, i.e., C 4.5 performed worst in approximating the viewership of the program taking 397 for constructing the training model. Fig. 6. shows the performance comparison of classifiers for testing and training phase

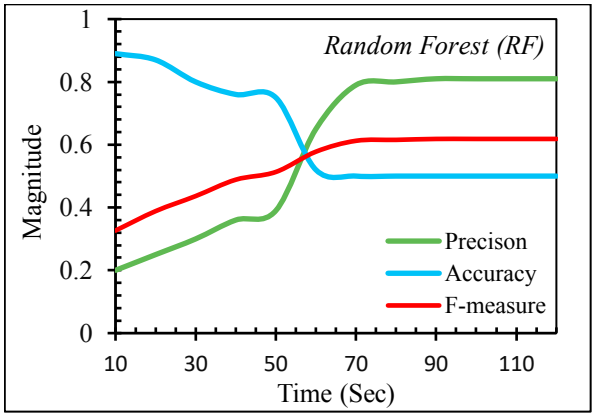


Fig 1 . Classification performance of RF.

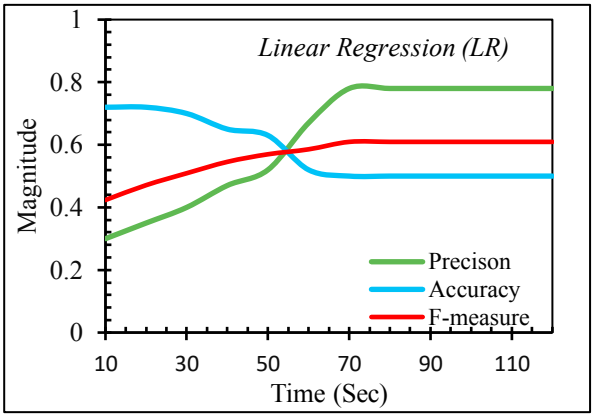


Fig 2. Classification performance of LR.

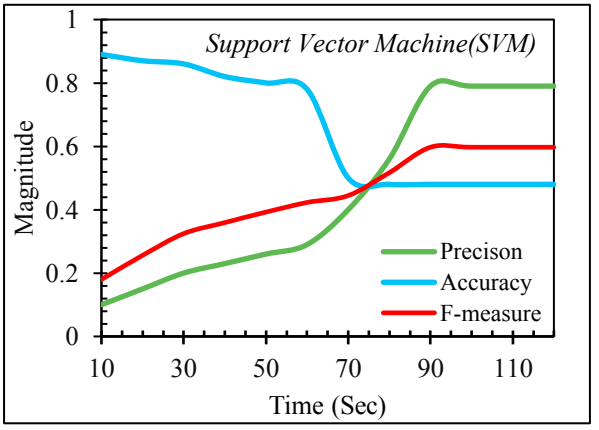


Fig 3. Classification performance of SVM.

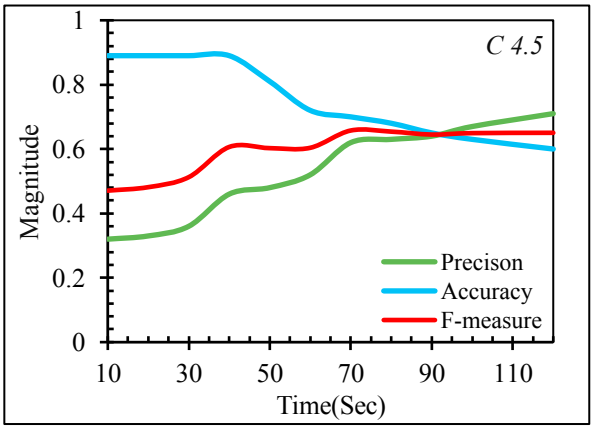


Fig 4. Classification performance of C 4.5.

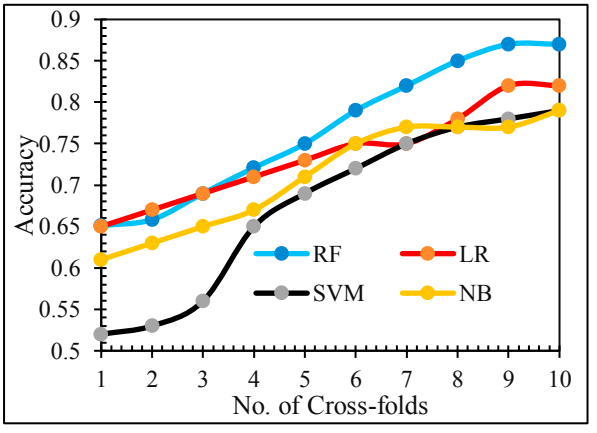


Fig 5. Accuracy of machine learner across folds.

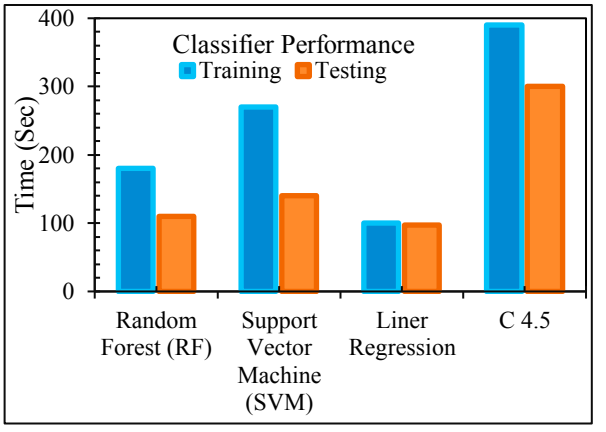


Fig 6. Performance comparison of classifiers for testing and training phase.

## 4. Conclusion and Future Work

Approximating viewership of a program is important for broadcast networks and cable (TNT, AMC and USA), as they profit by selling advertising space during programming (airing of the show) while premium cable and streamers reply on subscription. This paper proposed a methodology to approximate the viewership of streaming television shows by exploiting data from social networks, i.e., publicly available information as well as sentiment score derived from sentiment analysis on tweets published about television programs, using Random Forest classifier. The methodology achieved 85% accuracy in predicting the viewership of the streaming shows. Moreover, a comparison with other classifiers reveals that Random Forest produces an excellent balance of Precision and Accuracy while maintaining a high F-measure. However, linear regression requires little time to construct a model in comparison to RF. In the future, we plan to enhance the prediction accuracy of the proposed methodology by incorporating data from other media sites, particularly from YouTube.

## References

[1]. Neilson Ratings, [WB], https://en.wikipedia.org/wiki/Nielsen_ratings. Last accessed, March 2022.

[2]. Sentiment Analysis, “TextBlob”, https://textblob.readthedocs.io/en/dev/. Last accessed, March 2021.

[3]. Rotten Tomatoes, https://en.wikipedia.org/wiki/Rotten_Tomatoes. Last accessed, April 2021.

[4]. Developer Docs, https://developer.twitter.com/en/docs., Last accessed Feb 2021

[5]. R. Diaz-Uriarte, S. Alvarez de Andres. Variable selection from random forests: application to gene expression data.TR009, 2005.

[6]. L. Breiman, Bagging Predictors, Machine Learning, 26, pages. 123-140, 1996.

[7]. David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Mü ller. 2010. How to Explain Individual Classification Decisions. JMLR , Vol. 11, 2012.

[8]. Bouaziz A., Dartigues-Pallez C., da Costa Pereira C., Precioso F., Lloret P. Short Text Classification Using Semantic Random Forest. In: Bellatreche L., Mohania M.K. (eds) Data Warehousing and Knowledge Discovery. DaWaK 2014. Lecture Notes in Computer Science, vol 8646. Springer, 2014. Cham. https://doi.org/10.1007/978-3-319-10160-6_26

[9]. R. F. Alhujaili and W. M. S. Yafooz, "Sentiment Analysis for Youtube Videos with User Comments: Review," *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, pp. 814-820, 2021, doi: 10.1109/ICAIS50930.2021.9396049.