Predicting the Popularity of News Articles

Yaser Keneshloo* Shuguang Wang† Eui-Hong (Sam) Han† Naren Ramakrishnan*

Abstract

Consuming news articles is an integral part of our daily lives and news agencies such as The Washington Post (WP) expend tremendous effort in providing high quality reading experiences for their readers. Journalists and editors are faced with the task of determining which articles will become popular so that they can efficiently allocate resources to support a better reading experience. The reasons behind the popularity of news articles are typically varied, and might involve contemporariness, writing quality, and other latent factors. In this paper, we cast the problem of popularity prediction problem as regression, engineer several classes of features (metadata, contextual or content-based, temporal, and social), and build models for forecasting popularity. The system presented here is deployed in a real setting at The Washington Post; we demonstrate that it is able to accurately predict article popularity with an $R^2 \approx 0.8$ using features harvested within 30 minutes of publication time.

1 Introduction

News is an integral part of our daily lives and agencies such as The Washington Post publish more than a thousand pieces of news content every day. However, not all of these articles become equally popular, and thus popularity prediction is an invaluable strategy for journalists and editors to prioritize which articles need to be refined.

Two issues are critical in the resolution of the populatity prediction problem. First, since people consume their news via a variety of channels nowadays, there are multiple measures of popularity, e.g., number of page views on the WP site, number of likes or shares on Facebook, or the number of searches in a search engine. Second, article popularity can be defined in a local or a global context. Local context measures are primarily meant for use within a single news agency whereas global context measures help ascertain the popularity of an article amongst articles from other new agencies as well. While our ultimate goal is to integrate a range of popularity measurements across different news channels in a global context, in this paper we primarily focus on predicting popularity in a local context and use the number of page views of an article as a surrogate for its popularity.

By its nature, the life span of a news article is very short (following its publication). Interviews with journalists and editors at The Washington Post suggsted that it is more interesting and valuable to predict the early popularity of an article rather than its long-term popularity. In this study, therefore, popularity of an article is defined as the number of page views within the first 24 hours following publication. We cast popularity prediction as a regression problem, extract features from a news article for upto 30 minutes following publication, and use these features to project the page views the article will receive within 24 hours.

Our main contributions are:

1. We evaluate an extensive set of metadata, contextual or content-based, temporal, and social features to predict the popularity of a news article. For instance, in addition to the click-stream and full content of the articles, we utilize features from Twitter users who shared the articles, features estimating freshness of an article, as well as sentiment features.

2. We evaluate multiple regression models for popularity prediction and deploy our best performing models in a real-time system at The Washigton Post.

2 Related Work

Popularity prediction for news articles is a relatiely novel problem and very few studies addressed this problem. However a growing number of studies have been carried out on predicting the popularity of other types of online content. Several studies have analyzed the rate at which tweets diffuse on Twitter or that at which videos are viewed on YouTube. The goals of these studies include predicting the exact number of retweets for a tweet [26], predicting the number of YouTube views for a video [22, 12], estimating the number of votes to a Digg post [17] or the number of page views for a news article [18], ranking news articles [24], forecasting the ranges of popularity for a tweet/news article [1], and

*Dept. of Computer Science, Virginia Tech, VA, USA, **(yaserkl,naren)@vt.edu**

†The Washington Post, **(shuguang.wang, sam.han)@washpost.com**

prediction of the exact number of comments for a news article [25, 24].

The prediction of online content popularity can be undertaken at two stages: before or after content publication. There are many studies that focus on using the early measurements after publication to predict future success [22, 18]. On the other hand, studies such as [1, 25] aim to predict the popularity before publication. The study in [1] used the number of times an article is posted on Twitter along with some contextual features to predict tweet counts. Mentions in tweets can be used as a surrogate for popularity to some extent but this is less accurate than page views on an article (as used here). In another study [27], the number of followers of the user who retweets a post is used to predict the total number of retweets for a tweet.

A Bayesian approach is proposed by [26] to predict the number of retweets of a tweet according to two features: number of followers of retweeters and the depth of the retweeter in the retweet tree. Using a graphical model, this approach trains different parameters related to these features. It uses the reaction time of a user, i.e. the time between when user sees a post and when the user retweets it, as the main predictor to predict the final retweet count.

Existing studies frame the popularity prediction problem as one of regression [16], classification [25, 14], or even clustering [12]. A variety of features are used in these studies to predict the popularity of tweets/news articles [23]. We can categorize these features into: content-based and temporal features.

Content-based features are usually extracted from the text of a tweet/news article. Features such as the sentiment of a text [5, 21], emotions within the text [3, 2], subjectivity of its language [1], named entities [1], and freshness of a content [6, 8] are all considered as highly correlated factors to virality of content. The work in [1] suggested the idea of using the category and the name of the website that publishes the article to predict virality. Fig 1 plots the distribution of categories for the viewed and tweeted articles in our WP dataset. As can be seen, categories and their corresponding distributions differ between the articles posted on Twitter and the ones viewed on the WP website. This shows that users do not necessarily share the content that they read with their friends. Instead, they select specific stories and share them among their network. This result follows the finding in [4], where they found that users only tend to share selected stories with their friends.

Temporal features are mainly extracted from the click time-series of an article. Among all the temporal features, the number of retweets/clicks in the first hour

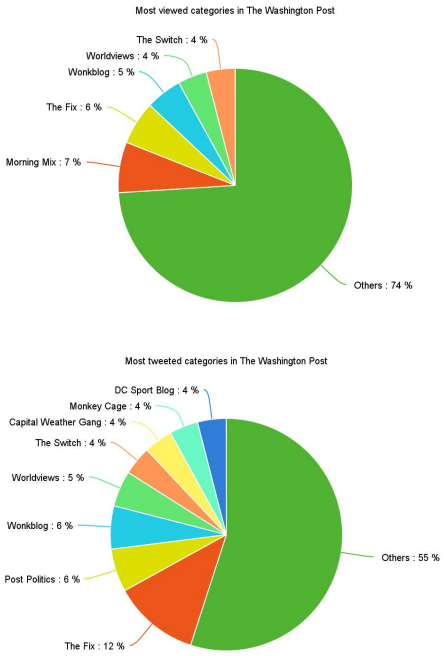

Figure 1: Distributions of categories for (top) most viewed and (bottom) most tweeted articles in The Washington Post website. Note that the distributions differ.

of publishing the tweet/article (n_0) is known to be highly correlated to the final counts of retweets/clicks at time t, i.e. n_t [22, 12]. According to [22], there is a high linear correlation between the log-transformed number of retweets/clicks in the first hour and its long-term popularity. This study proposes a simple constant multiplier α to estimate n_t according to n_0, i.e., $\log n_t = \alpha \cdot \log n_0$. (We use a similar approach as our baseline model.) An extension of this method is proposed by [20], where they replace n_0 with samples at regular intervals (15 minutes) up to the first hour. However, the relative importance of the clicks of an article, among all the other articles that are published at the same time, is ignored in this method. (We improve this feature by suggesting a normalized page view feature which is relative to all the articles that are published at the same time.) Along with these features, retweet acceleration and the retweet depth in the retweet tree have been used in [14] to predict the popularity of Twitter messages.

3 Popularity Prediction

In this section, we present our proposed method to predict the popularity of a news article. Our goal is to predict the number of page views that a news article

will receive within the first day since its publication. We track all articles for 30 minutes upon publications and extract a range of temporal, social, and contextual features for forecasting.

3.1 Metadata features. The WP metadata contains detailed information about each article such as title, full content, keywords, authors, type (blog or article), category, news section, and the publication date. From the publication date of an article we extract the hour of the day and day of week that the article is published. Along with these features, we use the author name, news type, category, and section as our metadata features.

3.2 Content-based features. Contextual features deal mostly with the title, keywords, and the content of the article. We separate these contextual features into two sets: the first group includes features extracted from the WP metadata dataset and the second group is extracted by querying our *pseudo* archive of the WP dataset (explained in detail later).

3.2.1 Sentiment. As another contextual feature, we extract the sentiment of a text segment as probabilities belonging to either positive, negative, neutral, or compound classes. We use the Vader sentiment analyzer [13] for this purpose. In addition to the sentiment of a text, the emotion of a text is also an important factor in influencing virality [3]. However, as also mentioned in [3], there is no linguistic tool that can capture emotions in a text, and we have to determine the emotions in the text using manpower. We create two different sentiment feature sets, one for the sentiment of the full content and the other for the titles of the news articles. We use these two different sets because most of the current sentiment analyzers face a problem when classifying long text.

3.2.2 Named Entity Extraction. Named entities are also an important factor in influencing the virality of news articles. News articles about a well-known local or a person can bring a lot of attention to itself. We use the Stanford NLP Library[1] for this purpose. Although capturing only the number of named entities may not capture the importance of each of these entities, it will provide an indicator of how thoroughly the article talks about a subject. In our experiments, we will show how these numbers help improve our predictions. (As described later, We also capture the importance of named entities by estimating counts from the *pseudo* dataset.)

[1] http://nlp.stanford.edu/software/

3.2.3 Readability. Berger and Milkman [3] claimed that longer articles tend to be shared far more often. On the contrary, we found that the correlation between the length of an article and the number of page views that it receives is small. The claim in [3] is based on the premise that journalists tend to write longer pieces when they are writing on hot topics. Along with the article length, the readability of the article is also a factor and there are several different metrics for this purpose (typically based on estimating the number of years of education required for a person to understand the text). Most of these methods use a combination of word and sentence length, number of complex words, and number of syllables within the text to estimate readability. For our purpose, we use the Flesch-Kincaid Readability Test, Gunning-Fog Score, Automated Readability Index, and the Coleman-Liau Index to determine readability (see [10] for a description of these measures). Along with these metrics, we also consider the number of sentences, number of complex words, percentage of the complex words to the total number of words, number of syllables, average number of syllables per word, average number of words per sentence, average sentence length, and length of the title and full content of an article.

3.2.4 Freshness. All afore-mentioned features are measured using standard methods and software libraries. In this section we propose a method that aims to determine the freshness of an article. As mentioned earlier, viral news articles are usually driven by fresh and surprising information. In order to capture the freshness of an article, we must model articles coming from other news agencies in the recent past. Access to such information is beyond the scope of this work; we create a *pseudo* dataset containing WP articles published in the past (in this paper, before Sep 2014) that received at least one page view (in the period of Sep 2014 to March 2015). The pseudo-archive is indexed using Lucene w.r.t. the title, full content, and keywords and we use Lucene's in-built scoring mechanism to identify the top-10 most similar articles. From the similarity results, we extract the following features:

- **Topic intersection.** The topic intersection between the queried article and the archived articles is defined as:

$$TI = \frac{|\text{keywords}_\text{q} \cap \text{keywords}_\text{a}|}{|\text{keywords}_\text{q} \cup \text{keywords}_\text{a}|} \tag{3.1}$$

  where keywords_q and keywords_a are the sets of keywords in the queried article and the top ten articles, respectively. This feature will be close to

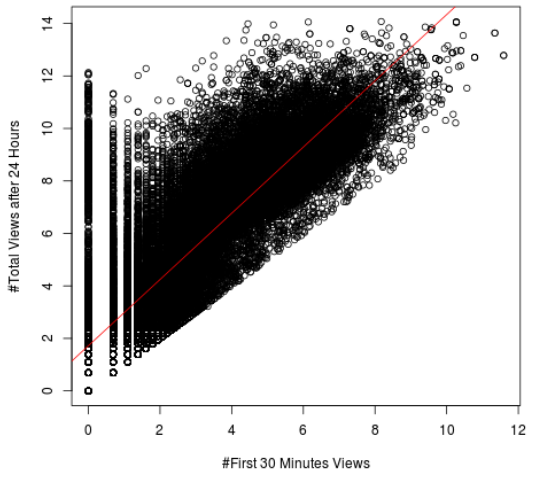

Figure 2: Correlation between the log-transformed number of first 30-minutes page views and the total number of page views after 24 hours for WP articles. The red line represents the simple linear regression estimation for the first 30 minutes page view count. The significant scatter around this estimate suggests that additional features are necessary for improving performance.

zero for fresh or new articles and close to one for old stories.

- **Top ten stories page view count.** For each of the top ten similar articles, we find the number of page views that the article received until the publishing date of the queried article. We consider these features as $TC_i, \text{for } i = 1, \cdots, 10$ sorted so that $TC_i < TC_{i+1}$. These features will provide us a rough estimate of the virality of the new article w.r.t. the performance of earlier similar articles.

- **Top ten stories content similarity.** For each of the top ten articles, we find the cosine similarity of its content against the content of the queried article. This feature also captures the freshness of content. We represent these features as $TS_i,\ for\ i = 1, \cdots, 10$ sorted so that $TS_i < TS_{i+1}$.

Along with the above features, we also capture the total number of articles that are similar to the queried article. We call this feature the *hits* number, which represents the number of hits for the queried article in our archived database. The lower this value, the more fresh the content of an article.

3.3 Temporal features. We extract temporal features from the click stream data of page views at 5 minutes intervals. In analyzing YouTube views, it has been found that there exists a high correlation between the log-transformed number of views in the first hour and the ultimate number of views an entry receives [22]. In this work, the number of page views in the first 30 minutes is considered as the primary feature for prediction. (Other works, such as [14], used additional features such as the retweet time series in the first 30 minutes and retweet acceleration.) In conducting a similar analysis for news articles, as shown in Fig 2, we find that while a similar relationship approximately holds, the observations are scattered more in space, suggesting that such an estimation can at best be used as a bound. Here, we propose additional features in conjunction to predict the number of page views:

- **Time difference between the publishing time and first page view.** This feature captures how fast people react to a news article. We also call this feature *page view reaction time*. We expect that for viral articles, this number will be small and for ordinary articles this be a higher number. In our WP dataset, the median time difference for all articles is 237 minutes. 85% of the top 1% viral articles have less than 200 minutes reaction time.

- **Number of page views after 30 minutes.** This number captures the total number of page views that an article receives within the first 30 minutes after its publication. On average, the top 1% articles received around 571 page views after 30 minutes and 300K after a day. It is worth mentioning that the average page views for the first 30 minutes of the rest of the articles is around 18.

- **Page view acceleration.** We use the approach from [14] to capture page view acceleration:

$$\text{Acceleration} = \frac{\sum_{t=2}^{N} n_t^x - n_{t-1}^x}{N} \tag{3.2}$$

where n_t^x is the number of page views of article x at time interval t and N is the total number of time intervals within the first 30 minutes. (Since we use a 5 minute time interval to build our time series, $N = 6$ for the first 30 minutes.)

- **Page view time series.** Similar to [14], we use the values in each time interval of page view time-series, i.e. n_t, as predictors.

- **Normalized page view time series.** We normalized the page view time series w.r.t. the time series of other articles published at the same time.

Therefore, given m articles that are published at the same time, the normalized time-series is as follows:

$$NC_t = \frac{n_t^x}{\sum_{i=1}^{m} n_t^i} \tag{3.3}$$

where n_t^x is the total number of page views of article x within the time interval t and n_t^i is the total number of page views of i^{th} article that is published at the same time as article x.
We could also use a time window to normalize these time series w.r.t. all articles published in the last few hours. The normalized count is a better measure than the count itself, since it finds the relative importance of an article among other articles that are published at the same time. Given m articles the one that gets more attention will have a higher normalized count. Therefore, a normalized count close to one means that the article is receiving more attention amongst all other articles published at the same time.

3.4 Social Media Features. Using the click stream dataset, we generate another dataset to capture social media (Twitter) activity related to each article. We use the Topsy API[2] to extract all tweets that share a WP URL. For each article we create a tweet and retweet time-series similar to the click stream dataset, i.e. each row of this dataset contains the timestamps of the tweet along with its TweetID. Using the TweetID of each tweet, we query Twitter using its API[3] to access the user profile and generate a user profile dataset. For each user, we extract the number of followers, number of listed counts, number of friends, and number of status message/updates. For each news article, from the extracted tweets related to each article, we build the retweet time series. We use the retweet time series of an article to generate another set of social media-driven temporal features:

- **Time difference between publishing time and first tweet.** We call this feature the *tweet reaction time* and like the *page view reaction time*, the tweet reaction time captures how fast people share the news on Twitter. Similar to the *page view reaction time*, we expect that viral articles are shared faster than other news articles. In our WP dataset, the mean and median of the tweet reaction time, for articles that have at least one tweet in Twitter, is 13 minutes. 83% of the top 1% viral articles have less than 200 minutes tweet reaction time.

[2]http://api.topsy.com/
[3]https://dev.twitter.com/

- **Number of retweets after 30 minutes.** This number finds the total number of retweets that an article receives in 30 minutes following publication. On average, the top 1% articles received around 106 retweets after 30 minutes. Surprisingly, the rest of articles receive around 194 retweets on average after 30 minutes. The reason for this behavior is that, as mentioned in [15], people use social media to share almost everything that happens around them. However, only a small portion of these news articles go viral. We will later see in Section 4 that unlike the page views in the first 30 minutes, the retweet counts do not significantly contribute to prediction performance.

- **Number of '30 minutes followers'.** We extract the number of followers of all users who shared the article and aggregate them to find this number. This feature captures the approximate number of people who were exposed to the article within the first 30 minutes. On average, the aggregate number of followers of users who shared the top 1% articles is around 380, while for the rest of the articles this number is 14. This shows that on average people who share viral articles tend to have more followers.

- **Retweet/followers acceleration**: Similar to page view acceleration, this feature is calculated using an equation akin to Equation 3.2.

- **Retweet/followers time-series**: We use the values in each time interval of retweet/follower time-series, i.e. n_t, as predictors.

- **Normalized retweet/follower time-series**: This feature is also calculated similar to how we calculated the normalized page view time-series.

Table 1 summarizes all the features used in our study.

4 Experiments

In overall, we extract 105 features for each article. These features are organized in sets and evaluated for their incremental improvement over the baseline method described earlier. All experiments are conducted using 10-fold cross validations and we use the average Adjusted R^2 ($AdjR^2$) value to report the performance. Besides overall performance, we are also interested in the performance of the models on viral articles. Therefore, in each fold of the run we first train the models on the training set and then test it on two test sets: one containing all test articles in that fold for the run, and the other containing only 1% of most popular articles in the first test set.

Table 1: Features evaluated in this work.

Metadata Features	
Article Type	Whether the article is a blog post or article.
Article Category	Different categories are viewed by different people; some categories do not not usually generate viral articles; others generate more popular news.
Article Section	Similar to the category of an article, the section of an article is also an important factor in influencing its popularity. Most published articles fall in the Sports, Politics, and Opinion sections.
Publication Date	We record this feature in terms of the time of day and day of week that the article is published.
Author Name	There are more than 12k authors in our dataset; authors such as Valerie Strauss and Dan Steinberg publish the most articles.
Contextual Features	
Sentiment	We extract sentiment scores for both the title and the main article. The sentiment is defined as probabilities in four categories: {negative, positive, compound, neutral}.
Named Entities	Number of persons, locations, and organizations in an article
Readability of Text	Five measures that captures different aspects of readability of a document
Freshness of Article	Organized into: (i) Topic Intersection; (ii) Click count of 10 most similar articles; (iii) Content similarity of 10 most similar articles; and (iv) Number of similar articles in the historical dataset
Temporal Features	
FirstViewTimeDiff	Time difference between publishing time and first page view.
First 30 Minute View	Number of page views after 30 minutes of publication.
Page View Acceleration	The rate at which an article is read within the first 30 minutes.
Page View Time Series Normalized Page View Time series	Time series of views in the first 30 minutes organized in 5-minute intervals
Social Features	
FirstTweetTimeDiff	Time difference between the publishing time and first tweet.
First 30 Minute Tweet Volume	Number of tweets after 30 minutes of publication.
First 30 Minute Followers' Number	Number of followers of users who post the news within 30 minutes.
Tweet/Follower Acceleration	The rate at which an article is being tweeted within the first 30 minutes.
Tweet/Follower Time Series Normalized Tweet/Follower Time Series	Time series of tweets in the first 30 minutes organized in 5-minute intervals.

Table 2: Comparison of different regression models on the complete (test) dataset and the top 1% (viral) dataset.

Model	Complete ($AdjR^2$)	Top 1% (R^2)
Multi Linear Regression	79.4	78.2
LASSO Regression	72	52.1
Ridge Regression	80.3	54.5
Tree Regression	82.9	42.5

4.1 Model Selection. Table 2 compares the performance of multiple regression models over our datasets. As shown here, tree-based regression performs the best on the complete dataset but performs poorly on the viral dataset. On the other hand, the multiple linear regression (MLR) model [11] has satisfactory performance over both test datasets, and we utilize this approach for the rest of our experiments.

The baseline method utilizes the strongest signal for predicting page views, i.e. the log-transformed number of page views in the first 30 minutes after the publication of article. Our dataset contains more than 41K articles that are published between September and October 2014, out of which metadata information is available for 37K articles. Therefore, for about 4K articles in our dataset, we have missing metadata features for whom we use zero as the default value. Additionally, for articles that have not been tweeted, we use custom default values to fill the missing features. For instance, for the time difference between the first tweet and publication date of an article, we use an extremely large value of this feature for articles that do not have this measurement. For other social features such as the number of retweets/followers in the 30 minutes and the retweets/followers time-series, we use zero. The following results show that our model is robust enough to deal with missing values.

Table 3 shows the result of our regression analysis on this dataset. The bold entries in this table show the feature set that provides the best boost w.r.t. the baseline model. Adding only metadata features provide the maximum boost among other set of features. If we use the full set of features, we improve the performance of the baseline method by 10%. In order to show that the small improvement achieved using temporal features is also a significant improvement, we use the paired t-test to examine the significance of the results, and find that the boost achieved using the temporal feature is extremely significant with a p-value < 0.0001. Additionally, to better understand the effect of freshness features in the performance of the model, we remove these features from the content features. Although

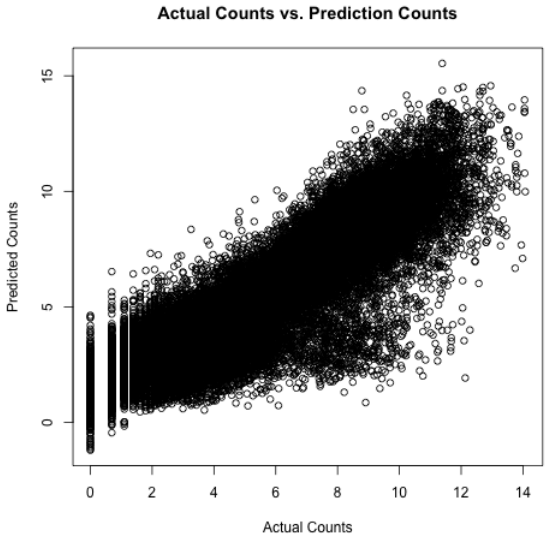

Figure 3: Plot of actual page views versus predicted values.

Table 3: The Adjusted R^2 and R^2 value for the regression model on the complete and top 1% datasets.

Model	Complete ($AdjR^2$)	Top 1% (R^2)
Baseline	69.4	74
Baseline + Temporal	70.4	72.1
Baseline + Social	72.5	77.3
Baseline + Content	71.1	**79.3**
Baseline + Content - Freshness	70.6	79
Baseline + Metadata	**77.2**	78.1
All Features	79.4	78.2

the absence of the freshness features causes only a slight deterioration in performance, this difference is extremely significant according to the t-test.

To see the performance of our model on the most viral articles, we use our trained model to predict the page views over this set. For this experiment, as the number of features exceeds the number of samples, we use the R^2 measure to report performance. Table 3 shows the performance of our model on this dataset. The baseline R^2 for this dataset is 69.4, and using the full feature set provides a 4% boost over the baseline. According to Table 3, although metadata information provides the most lift in the $AdjR^2$ score, the content-based features help the most in predicting the viral articles. Fig 3 shows the plot of actual click counts after 24 hours versus the predicted value of our whole model in the complete dataset (compare this against Fig. 2).

4.2 Important Features. In order to quantify the important features in our dataset, we explore each set of features separately to identify sets that provide the maximum boost in model performance. We aim to extract the best subset of features that provide the maximum $AdjR^2$ value. Tables 4 and 5 depict the result of this experiment on the complete dataset and the top 1% dataset, respectively. According to this experiment, among the temporal features, the page view time-series and its normalized time-series are the most important features in both datasets. Note that when we use only these two features to predict the page views in the top 1% dataset, we receive a higher R^2 value than when we use the full set of temporal features. This is due to possible multicollinearity that exists between some of the features in the temporal feature set.

Similarly, the important social features are identical between the complete dataset and top 1% dataset. From all the social features, the time difference between the first tweet and the publication date of the article, number of followers of users who shared the article after 30 minutes of the publication, and the retweet time-series and its normalized time-series are the most important features. As can be seen, the performance using these selected features is exactly the same as the results in Table 3.

Although, as shown in Table 4 and 5, the complete and top 1% dataset share similar social and temporal features, they have a different set of important contextual features. According to our experiments, in the complete dataset, a subset of top ten stories page views and content similarity was picked by the subset selection filter. Moreover, out of all the sentiment features, only the probability of content neutrality and compoundness is considered to be important in the prediction. Also, from the four different readability measures, the SMOGIndex was picked by the best subset selection method. Additionally, despite the known claim that article length is a good attribute to predict popularity [3], according to our analysis, title length is the feature that helps improve performance.

In the top 1% dataset, however, the topic intersection is one of the important features. Moreover, the compoundness of the content is not an important feature and none of the readability features are selected for this dataset. As can be seen in Table 5, among all the contextual features, all the proposed features related to the freshness of an article are selected by the best subset selection method. This magnifies the importance of article freshness in predicting the virality of an article.

4.3 Deployment at The Washington Post In this paper, we explored various features and trained a regression model for the popularity prediction task. In order to help journalists and editors at The Washington Post, we deployed this model and built a real time forecasting system for each article. Once a news article is published, the forecasting system begins to track it and extract features described in this paper. With the help of Splunk [4], we built a dashboard to order articles based on their forecasted popularity. As we make predictions for news articles, we also track the actual page views of articles for evaluation purposes. In addition to providing popularity prediction to editors and journalists, we are also able to ascertain how well our proposed regression model performs over the latest news articles. We use the same setup to evaluate the performance of the forecasting model. We collect all articles that have been published in August 2015 (after the forecasting system is deployed). Table 6 summarizes the performance of this model. Although the deployed model is not using all the proposed features, it has a comparable performance to the results shown in Table 3.

Table 4: The $AdjR^2$ value and list of the important features extracted from each feature set using the best subset selection method (all articles).

	$AdjR^2$	**Important Features**
Temporal	70.3	Page View Count and Normalized Page View Count time-series
Social	72.5	Tweet time difference, #Followers in the first 30 minutes, Retweet and Normalized retweet time-series
Content	70.9	Top ten stories page view, Top ten stories content similarity, Title length, Probability of content neutrality and compoundness, and SMOGIndex

Table 5: The R^2 value and list of the important features extracted from each feature set using the best subset selection method (viral articles).

	R^2	**Important Features**
Temporal	72.6	Page View Count and Normalized Page View Count time-series
Social	77.3	Tweet time difference, #Followers in the first 30 minutes, Retweet and Normalized retweet time-series
Content	79.4	Topic intersection, Top ten stories page view, Top ten stories content similarity, Title length, and Content neutrality

5 Conclusion

This paper is the first effort to predict page view counts of news articles. We explored different factors that play essential roles on the popularity of news articles, i.e., temporal, social, and contextual features. Our evaluation results show that among these three sets of features, the contextual features related to the freshness of an article are the most important factor in predicting the page views of viral articles,

[4] http://www.splunk.com/

Table 6: Evaluation of the deployed model for articles published in August 2015

Data	$AdjR^2$	Top 1% (R^2)
August 2015	0.829	0.620

while metadata features are the strongest signals for predicting the performance of news articles in general. From these three sets of features, we also identified the most effective features that significantly contribute to popularity prediction of news articles. Motivated by the excellent offline evaluation results, we deployed the model at The Washington Post. Future work is aimed at not just popularity prediction but also supporting other aspects of an article's creation, publication, and revision over its life cycle.

References

[1] Bandari, R.; Asur, S.; and Huberman, B. A. *The pulse of news in social media: Forecasting popularity. CoRR* abs/1202.0332, 2012.

[2] Berger, J., and Milkman, K. *Social transmission, emotion, and the virality of online content. Wharton Research Paper*, 2010.

[3] Berger, J., and Milkman, K. L. *What makes online content viral? Journal of marketing research*, 49(2):192–205, 2012.

[4] Berger, J., and Schwartz, E. M. *What drives immediate and ongoing word of mouth? Journal of Marketing Research*, 48(5):869–880, 2011.

[5] Berger, J. *Arousal increases social transmission of information. Journal of Psychological science*, 22(7):891–893, 2011.

[6] Borghol, Y.; Ardon, S.; Carlsson, N.; Eager, D.; and Mahanti, A. *The untold story of the clones: content-agnostic factors that impact youtube video popularity.* In *Proceedings of the SIGKDD'12*, 1186–1194, 2012.

[7] Castillo, C.; El-Haddad, M.; Pfeffer, J.; and Stempeck, M. *Characterizing the life cycle of online news stories using social media reactions.* In *Proceedings of the CSCW'14*, 211–223, 2014.

[8] Cha, M.; Kwak, H.; Rodriguez, P.; Ahn, Y.-Y.; and Moon, S. *Analyzing the video popularity characteristics of large-scale user generated content systems. IEEE/ACM Transactions on Networking (TON)* 17(5):1357–1370, 2009.

[9] Cherkasova, L., and Gupta, M. *Analysis of enterprise media server workloads: access patterns, locality, content evolution, and rates of change. IEEE/ACM Transactions on Networking* 12(5):781–794, 2004.

[10] DuBay, W. H. *The Principles of Readability.* Online Submission (2004).

[11] Freedman, D. *Statistical Models: Theory and Practice.* Cambridge University Press, 2005.

[12] Gürsun, G.; Crovella, M.; and Matta, I. *Describing and forecasting video access patterns.* In *Proceedings of IEEE INFOCOM'11*, 16–20, 2011.

[13] Hutto, C. J., and Gilbert, E. *VADER: A parsimonious rule-based model for sentiment analysis of social media text.* In *Proceedings of the ICWSM'14*, 2014.

[14] Kong, Shoubin, Y. F., and Feng, L. *Predicting future retweet counts in a microblog. Journal of Computational Information Systems* 10(4):1393–1404, 2014.

[15] Kwak, H.; Lee, C.; Park, H.; and Moon, S. *What is twitter, a social network or a news media?* In *Proceedings of the WWW'10*, 591–600, 2010.

[16] Lee, J. G.; Moon, S.; and Salamatian, K. *Modeling and predicting the popularity of online contents with cox proportional hazard regression model. Journal of Neurocomputing* 76(1):134–145, 2012.

[17] Lerman, K., and Hogg, T. *Using a model of social dynamics to predict popularity of news.* In *Proceedings of WWW'10*, 621–630, 2010.

[18] Marujo, L.; Bugalho, M.; Neto, J. P. d. S.; Gershman, A.; and Carbonell, J. *Hourly traffic prediction of news stories. arXiv preprint arXiv:1306.4608*, 2013.

[19] Mishne, G., and De Rijke, M. *A study of blog search.* In *Advances in information retrieval.* Springer. 289–301, 2006.

[20] Pinto, H.; Almeida, J. M.; and Gonçalves, M. A. *Using early view patterns to predict the popularity of youtube videos.* In *Proceedings of the WSDM'13*, 365–374, 2013.

[21] Reis, J., Benevenuto, F., Olmo, P., Prates, R., Kwak, H., and An, J. *Breaking the News: First Impressions Matter on Online News. arXiv preprint arXiv:1503.07921*, 2015.

[22] Szabo, G., and Huberman, B. A. *Predicting the popularity of online content. Communications of the ACM* 53(8):80–88, 2010.

[23] Tatar, A.; De Amorim, M. D.; Fdida, S.; and Antoniadis, P. *A survey on predicting the popularity of web content. Journal of Internet Services and Applications* 5(1):1–20, 2014.

[24] Tatar, A., Antoniadis, P., De Amorim, M. D., and Fdida, S. *Ranking news articles based on popularity prediction.* In *Proceedings of ASONAM'12*, 106–110, 2012

[25] Tsagkias, M.; Weerkamp, W.; and De Rijke, M. *Predicting the volume of comments on online news stories.* In *Proceedings of the CIKM'09*, 1765–1768, 2009.

[26] Zaman, T.; Fox, E. B.; Bradlow, E. T.; et al. *A bayesian approach for predicting the popularity of tweets. The Annals of Applied Statistics* 8(3):1583–1611, 2014.

[27] Zhao, Q.; Erdogdu, M. A.; He, H. Y.; Rajaraman, A.; and Leskovec, J. *SEISMIC: A self-exciting point process model for predicting tweet popularity. CoRR* abs/1506.02594, 2015.