NewsWorthy

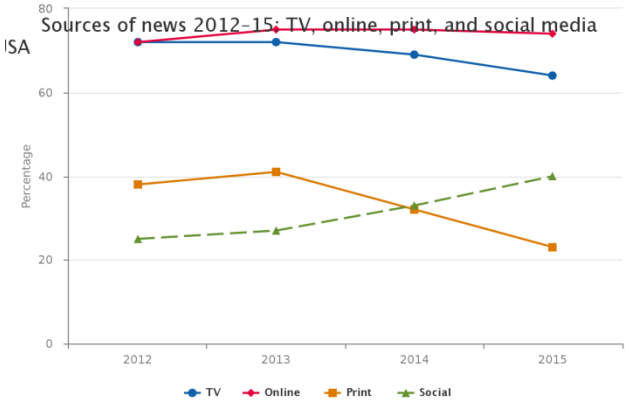
Pre-Publication Popularity Prediction

Qiuyi (Helen) Hong, Yanhong (Haley) Huang, Tom Meagher, and Tyler Reese

ABSTRACT− NewsWorthy is a new service targeted towards publishers, authors, bloggers, and all producers of online news content. NewsWorthy uses data science techniques to predict, pre-publication, whether an article is likely to be unpopular. This report serves as a summary of all facets of the NewsWorthy venture. The business problem addressed by NewsWorthy, and its significance in the market, is summarized. Next, NewsWorthy's mathematical solution to this problem is developed. Finally, a proof-of-concept is presented on a sample data set.

INTRODUCTION

News Media is a booming business market. In 2013, syndicated news outlets in the United States alone generated more than $63 Billion in revenue. Billions in additional revenue were generated by "for-profit digital-news outlets" such as Huffington Post and Buzzfeed (Holcomb & Mitchell, 2014). What is more, the platforms by which Americans are retrieving their news are changing rapidly. A recent study conducted by Reuters (Newman, 2015) indicates that in the last 4 years, the internet has begun to out-pace Television as a primary source for news consumption. Even more interesting, 2015 marked the first year in which Social media cites were a more popular source of news than newspapers: with 40% of people turning to social media for news, compared to only 23% consulting newspapers.



Source: Reuters, (Newman, 2015)

Amid this changing news landscape, the success or failure of each article published is becoming more important to authors and publishers. As a result, it is becoming more important to avoid releasing an article which is perceived as "unpopular." A method of projecting, before publication, when an article is likely to "flop" can be of certain value to digital media outlets. NewsWorthy is the first data-driven service on the market to do just that. In this report, we will detail the process and goals of NewsWorthy through three primary objectives.

Objective 1: The Business Part. Precisely describe the business problem NewsWorthy aims to solve, discuss why this problem is important, and explore how our data-science techniques can make a difference.

Objective 2: The Math Part. Formulate the business problem as a math problem, and develop a mathematical solution for implementation.

Objective 3: The Hacking Part. Develop a prototype of NewsWorthy by acquiring a sample data set and implementing the previously designed math solution.

OBJECTIVE 1: The Business Part

The business problem to solve.

NewsWorthy is an analytical tool targeted towards authors, publishers, bloggers, and all other generators of online news content. The goal is to understand when a news article, published online, is likely to be "popular" in comparison to other articles released in similar formats. More precisely, the goal is to predict−before publication−when an article is likely to be *unpopular*. This tool may be used by online contributors in selecting content for release or targeted revision. NewsWorthy is designed specifically for predicting when an article will be unpopular: the problem of projecting unpopularity is of basic importance to content creators. While creating a "viral" article is certainly desirable, avoiding articles which are irrelevant or unread is essential for success.

Why this problem is important to solve.

As news consumption continues to transition towards purely digital formats, this poses an additional challenge to content-creators. Before the rise of the internet, news was distributed through newspapers, television and radio. The number of such outlets available to an individual was limited, resulting in a source-based news consumption model: many people relied on a specific newspaper or television station. As a result, the success or failure of a specific story was immaterial. The rise of the internet and digital news allows individuals story-based news consumption: there are countless articles written about a given event, all of which can be centralized by a simple Google search. That is, people may now access news on a per-story basis, as opposed to relying on a specific source channel. As a result, the popularity or unpopularity of each article is now significant in the success or failure of news media outlets. A method of understanding an article's likely popularity, before publication, can be extremely useful in deciding which content to release. Perhaps more importantly, using such methods to help avoid releasing unpopular articles may be attractive to potential advertisers (more on this later).

The ideas behind NewsWorthy to solve this problem.

There are many metrics by which one may judge the "popularity" of an online article, such as page-visits, or external links. In recent years, however, social media platforms have become the primary outlet for many to share their opinions. What is more, studies indicate that an increasing number of Americans rely upon social media as a means of news consumption (Newman, 2015). In 2015 it was estimated that 40% of Americans turned to Social sources of news, while only 23% relied upon print sources. Even beyond those who retrieve their news via Social platforms, many more share external news articles via social media. In order to utilize, and perhaps pre-empt, these current trends, NewsWorthy uses the number of shares via social media as our metric of "popularity."

NewsWorthy uses another novel approach, by considering structure-based metrics of each article. The specific words in an article certainly influence the reader's enjoyment, but the structure of the article itself also plays a role in its popularity. For example, perhaps longer articles are not read when published on a Monday. Technology articles with very short titles may not generate interest. Rather than considering the individual tokens within a text, NewsWorthy utilizes structure-based analysis: considering metrics such as number of words, weekday of publication, length of title, polarity of title, genre, and many more.

Using these structural predictors, NewsWorthy trains a few standard machine learning models, and makes a final prediction via ensemble learning. This ensemble-learning is executed in a way which attempts to decrease the false-negatives. That is, NewsWorthy aims to successfully identify all *unpopular* articles. The costly real-world mistake in this problem is identifying an article as "popular" which turns

out to be *unpopular*. At the cost of some overall prediction accuracy, NewsWorthy is designed to avoid these (costly) false-negatives.

Differences that can be made via our data science approach.

Utilizing structure-based analytics offers a number of advantages as a data science approach. If the sample size of comparable articles is small, there may be little (or at least little relevant) overlap in the tokens contained within the text. Attributes based on the structure of an article may be computed for any text, and are comparable in a meaningful way regardless of the number of similar documents available. Moreover, structure-based predictors keeps the data under consideration relatively low-dimensional. Token-based predictors (such as n-grams) computed on a collection of documents grow in number very quickly. The result is very high-dimensional data: there are *many* more predictors than there are data instances. Such overly high-dimensional data is often difficult to reliably and accurately make predictions. NewsWorthy considers a fairly small number of structure-based predictors (approximately 50), generating relatively low-dimensional data from which it makes predictions.

Structure-based predictors also appeal to all contributors of online content. Large-scale publications may use our services with confidence that their pre-release content cannot be leaked. NewsWorthy does not require input of the text body: it takes as input the sequence of structural predictors. These may be computed by the publishers in-house, or via a simple software package they can purchase from NewsWorthy. On the other hand, large amounts of online content is generated by individuals. Professional bloggers, freelance writers, and "contributing authors," depend on article popularity and social media buzz to maintain their careers and attract potential advertisers. Most such individuals will lack the tools (and skills) to perform data analysis, and thus can benefit from our services.

This idea deserves investment from the Sharks.

NewsWorthy is designed to fill a need in the world of digital news. Studies indicate that more and more people rely on online outlets to receive news and information. The amount of content generated online continues to grow as well. The result is a story-based model of news consumption, putting an increased emphasis on the success of each individual article. Thus the ability to understand, before release, an article's likelihood of success will become increasingly beneficial to online contributors. Moreover, using a structure-based approach allows our services to appeal to all sources of online articles, ranging from individuals to large-scale publications.

In 2013, more than two-thirds of domestic news revenue came from advertising, totaling more than $40 Billion (Holcomb & Mitchell, 2014). More than $500 Million in additional revenue was generated by “for-profit digital news outlets” such as Huffington Post and BuzzFeed. These margins have only increased in the last 3 years. That is, large profits are available for online news outlets which can distinguish themselves to big-time advertisers. NewsWorthy presents an excellent method for online content creators to appeal to potential advertisers, by providing the ability to anticipate and avoid release of an unpopular article. The potential value-gain offered to publishers, in the form of advertising revenue, makes NewsWorthy a valuable tool.

Therefore, NewsWorthy fills a need in the world of digital news media. Moreover, the service it provides stands to increase advertising profits of publishers. Perhaps most important (for “shark” investors), there are currently no other companies in the market offering comparable services. As a result, we project that the NewsWorthy service will be adopted, and perhaps even relied upon, by online contributors ranging from large-scale publications, to individual bloggers and freelance authors. Such wide-ranging adoption in a variety of digital news media will lead to significant profits for the NewsWorthy venture. As a result, NewsWorthy presents itself as an excellent investment opportunity for the Sharks.

We are seeking a $120,000 investment from the Sharks, in return for a 7% equity stake in our company. We believe this valuation is reasonable given the need (and future need) which our product fulfills, the large market size, and the high potential for future growth. This valuation is in-line with the model of Y Combinator (https://blog.ycombinator.com/the-new-deal), who have helped bring to market successful startups such as Reddit, Dropbox, and Airbnb.

OBJECTIVE 2: The Math Part

Step 1: Formulate the business problem as a math problem.

As described previously, NewsWorthy aims to project when an online news article will be unpopular, based on structural features of the text. To begin, we must determine an appropriate metric of "popularity." Social media platforms are a primary outlet in which individuals express our opinions: we often "share" an online article that we believe others should read. Thus "shares" on social media can be used to estimate popularity. We will consider *two* separate measures of popularity based on "shares":

Popularity: The raw number of shares an article receives. This metric is only relevant given a base-line number of shares for other articles posted in similar formats.

Buzz-Factor: While the total number of shares an article receives can be used to estimate popularity, the speed at which those shares are generated is also of interest. Suppose articles A and B each receive 2,000 shares. If article A generated those shares over 6 months, while article B gained the same shares in only 2 days, article B can be judged as more "popular." The number of shares received per day, an estimate of the share-rate, will be referred to as buzz-factor.

The scope of NewsWorthy is focused even further. While it can be useful to estimate the extent of an articles popularity, of the utmost interest of content producers is being able to anticipate when an article is unpopular. That is, while it would be convenient to predict whether an article receives 1,500 instead of 1,200 shares, it is more important to authors and publishers to project when an article will receive a very low number of shares. In this case practically, it is worthwhile to consider changing the format of the article, or perhaps not releasing the content. Therefore, NewsWorthy is targeted at predicting when an article will be unpopular or generate no buzz. Once again we must define these terms mathematically.

Unpopular: NewsWorthy's metric of popularity is number of shares. We consider an article to be unpopular if, among other articles of similar formats, it generates a number of shares which is in the bottom 25%. That is, given a collection of articles, the unpopular articles are those in percentiles 0-25 in number of shares.

No-Buzz: Our metric "buzz-factor" is number of shares per day. We consider an article to have *no buzz* if, among other articles of similar formats, it generates a buzz which is in the bottom 25%. That is, given a collection of articles, the articles with no-buzz are those in percentiles 0-25 in number of shares per day.

Therefore, our business problem can be phrased mathematically as follows:

Given a collection of online news articles (of similar formats or origin) predict, which among those, will rank in the bottom 25% in numbers of shares or number of shares per day.

Step 2: Devise a Math Solution to the problem.

As described previously, NewsWorthy makes a deliberate choice of how to transform articles into data for analysis. Specifically, we choose to compute attributes based on the structure of the article, as opposed to the textual content. This offers a few advantages. Attributes based on the structure of an article may be computed for any text, and are comparable in a meaningful way regardless of the number of similar documents available. These predictors also generate data which is considerably low-dimensional compared to token-based predictors. Moreover, software can be sent to customers which inputs an article, and outputs a set of structure-based attributes which are then used for analysis. This enables us to perform data analysis without needing access to the original text, thereby protecting the unpublished content of our customers.

Attributes: The following attributes are computed for training (comparable) articles:

1. Number of "Shares" on social media **('shares')**
2. Number of days since publication **('timedelta')**

The following attributes are computed for both training and testing articles:

3. Number of words in the Title **('n_tokens_title')**
4. Number of words in the text body **('n_tokens_content',)**
5. Number of unique words in the text body **('n_unique_tokens')**
6. Number of unique non-stop words in the text body **('n_non_stop_unique_tokens')**
7. Number of videos included in the article **('num_videos')**
8. Number of images included in the article **('num_imgs')**
9. Average word length **('average_token_length')**
10. Number of keywords associated with the article **('num_keywords')**
11. Genre of the article (As 6 dummy variables: **'is_lifestyle', 'is_entertainment', 'is_business', 'is_social_media', 'is_tech', 'is_world'**,)
12. Day of the week of publication (As 8 dummy variables: **'is_monday', 'is_tuesday', 'is_wednesday', 'is_thursday', 'is_friday', 'is_saturday', 'is_sunday', 'is_weekend'**)
13. Text subjectivity **('global_subjectivity')**
14. Overall text polarity **('global_sentiment_polarity')**
15. Percent of positive words in the content **('global_rate_positive_words')**
16. Percent of negative words in the content **('global_rate_negative_words')**
17. Rate of positive words among non-neutral tokens **('rate_positive_words')**
18. Rate of negative words among non-neutral tokens **('rate_negative_words')**
19. Avg. polarity of positive words **('avg_positive_polarity')**
20. Min. polarity of positive words **('min_positive_polarity')**
21. Max. polarity of positive words **('max_positive_polarity')**
22. Avg. polarity of negative words **('avg_negative_polarity')**
23. Min. polarity of negative words **('min_negative_polarity ')**
24. Max. polarity of negative words **('max_negative_polarity')**
25. Title Subjectivity **('title_subjectivity',)**
26. Title Polarity **('title_sentiment_polarity',)**

For each training article, we split the articles into four "popularity" bins based on percentile of 'shares':

0-25%-- "Unpopular" ; 25%-50%-- "Mildly Popular"
50%-75%--"Popular" ; 75%-100%-- "Very Popular"

We store the popularity classification as an additional attribute (**'popularity'**). For each training article, we compute a 'buzz_factor' attribute,

$$\text{'buzz factor'} = \frac{\text{'shares'}}{\text{'timedelta'}}$$

For each training article we divide the articles into four "buzz" bins based on percentile of 'buzz_factor'

0-25%-- "No Buzz" ; 25%-50%-- "Some Buzz"
50%-75% -- "Buzz" ; 75%-100% "Lots of Buzz"

We store the "buzz" classification as an additional attribute, ('**buzz**').

Target Variables: Our goal is to predict whether or not an article will be "unpopular" or generate "no buzz." Therefore we generate two (separate) Boolean target variables,

'unpopular'—Takes value "True" if and only if **'popularity'** = "unpopular"
'no_buzz'—Takes value "True" if and only if **'buzz'** = "No Buzz"

Feature Selection: After all attribute constructions (including a number of dummy variables for classification), the data generated by a collection of documents has approximately 50 predictors. In order to aid in interpretability of results, and attempt to reduce over-fitting, we reduce the number of features considered to 10. We perform this process two separate times, using a feature-importance metric to determine those which are most relevant to each of the two target variables (**'unpopular'**, and **'no_buzz'**). This is done using the ExtraTreesClassifier on a set of training documents. The ExtraTreesClassifier is a "class [that] implements a meta estimator that fits a number of randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset and use averaging to improve the predictive accuracy and control over-fitting" (3.2.4.3.3. sklearn.ensemble.ExtraTreesClassifier). This class includes a feature_importance attribute, from which we extract the top 10 features.

We choose to select from the raw features−instead of considering principal components, for example−for interpretation purposes. By searching for trends between popularity and untransformed structural predictors, this may allow us to make suggestions of how to increase popularity: such as "choose a longer title" or "use shorter words." Moreover, reducing the number of such predictors considered can allow for more targeted suggestions. Although principal components could allow for more accurate prediction, these features are abstracted from the structure of the text. We are willing to sacrifice a small amount of accuracy to provide interpretability.

Machine Learning: We now train machine-learning algorithms on the set of structure-based data generated by training documents. *For each target variable*, these models only consider the top 10 features determined in the previous step. Our method considers two machine learning algorithms:

1. Random Forest Classifier: We train 100 classification trees, each of which is trained on a random boot-strapped sample of the data, and each "split" in the tree may choose from a random subset of predictors.
2. KNN : The value of neighbor-parameter K is chosen based on cross-validation on the set of training documents.

Therefore NewsWorthy trains four models: A random forest and KNN model to predict **'unpopular**', and an additional random forest and KNN model to predict **'no_buzz'**.

Prediction: NewsWorthy uses ensemble learning to make predictions for each target variable. Suppose the desired target variable is **'unpopular'.** Given a test article, the machine learning methods above are trained on a set of articles of a similar format or origin. In order to predict whether the test article will be

"unpopular", each of the algorithms makes a prediction. The final prediction is chosen as "unpopular" if *either* of the trained models predicts that it is unpopular. This method sacrifices some prediction accuracy, however, it acts to minimize false negatives produced by NewsWorthy. That is, by making a final prediction of "unpopular" if either of the individual models predicts "unpopular", it serves to *decrease* the number of articles which are unpopular that we (incorrectly) predict as "popular." Predicting an unpopular article as "popular" is the more costly error in real world applications, and thus NewsWorthy is designed to reduce these errors.

Step 3: Implement the Math Solution

The implementation of this solution has, in essence, been explained in the process of developing a math solution. We provide a summary of this functionality here. NewsWorthy operates within a Python interface. Given a collection of text documents, the set of structure-based statistics can be computed on a per-document basis using a combination of list comprehensions and Python's Natural Language Toolkit (NLTK) package. This data is then compiled into a Pandas data frame, indexed by the articles contained in the collection. The remaining data analysis conducted by NewsWorthy is built on top of Python's Scikit-Learn package. For each target variable (**'unpopular'** or **'no_buzz'**), feature importance ranks are determined using the ExtraTreeClassifier class. The top 10 ranked features (for each target variable) are isolated and passed on for further analysis. NewsWorthy uses two standard classification models: Random Forests and K-Nearest Neighbors, functionality for which is provided by Scikit-Learn. Given one of the target variables, say–**'unpopular'**–these models are trained on the set of testing data, restricted to the 10 selected features. To make predictions, both models are allowed to predict on a test article, and NewsWorthy classifies that document as "unpopular" if either of the models predicts "True" for the **'unpopular'** target variable.

OBJECTIVE 3: The Hacking Part

Though NewsWorthy has not undergone full-scale software and product development, we have executed a prototype of the NewsWorthy methods in order to validate the underlying ideas.

Data: For this proof-of concept, we use the "Online News Popularity Dataset" available through the UCI machine learning repository (http://archive.ics.uci.edu/ml/datasets/Online+News+Popularity). This pre-constructed data set consists of 39,797 instances, each representing an article published on Mashable.com. For each instance, 59 pre-computed structural statistics are included in the data set. From these 59 attributes, the complete set of predictors and target variables discussed in Objective 2 can be either derived or computed.

Model Assessment: We use a held-out validation set to understand the accuracy of our models. The data is split randomly into 60% training data and 40% testing data. The testing subset is reserved for a final validation at the end of analysis. The parameters of our models are tuned using cross-validation on exclusively training data.

Data Analysis: After downloading the data set from the source URL, the CSV file is read and stored into a Pandas data frame. Using this data frame, all desired statistics defined in Objective 2 are computed (Note: some additional predictors were included in the original data set, which are removed from this data frame). The data is then split randomly into 60% training data and 40% testing. The goal of NewsWorthy is two predict two different measures of article "failure": **'unpopular'** and **'no_buzz'**. As such we separate our data analysis accordingly into two subsections, one for each target variable.

*Predicting **'unpopular'***

Using the feature importance attribute of Python's ExtraTreesClassifier class, we determine the top 10 ranked features for use in predicting the **'unpopular**' variable. The top 10 features related to the **'unpopular'** target are:

Table 3.1: Top 10 Important Features, 'Unpopular'

1	'timedelta'	6	'global_subjectivity'
2	'num_keywords'	7	'global_rate_positive_words'
3	'n_tokens_title'	8	'avg_positive_polarity'
4	'average_token_length'	9	'n_unique_tokens'
5	'n_non_stop_unique_tokens'	10	'global_sentiment_polarity'

We now train two machine learning algorithms on the training subset restricted to these 10 predictors. First, we train a random forest classifier that uses 100 decision trees: each is given a bootstrapped sample of the data, and every "split" may only consider a random subset of the 10 predictors. Next we train a KNN model, where the neighbor parameter K is chosen by cross-validation on the training data. In particular, we split the training data into 70% sub-training data, and 30% sub-validations set. We train 10 KNN models on the sub-training data, with K values of 5, 10, 15, 20, 25, 30, 35, 40, 45, and 50. The validation scores on the sub-validation set are displayed in Figure 3.1. Based on these validation scores, we choose the KNN model with K = 30. Given these two models, we allow each to make predictions on the held-out validation set. For each test instance, we make a final prediction of **'unpopular'** = True if *either* of the individual models predicts **'unpopular'** = True. This prediction strategy has an accuracy of 72.9% on the validation set.

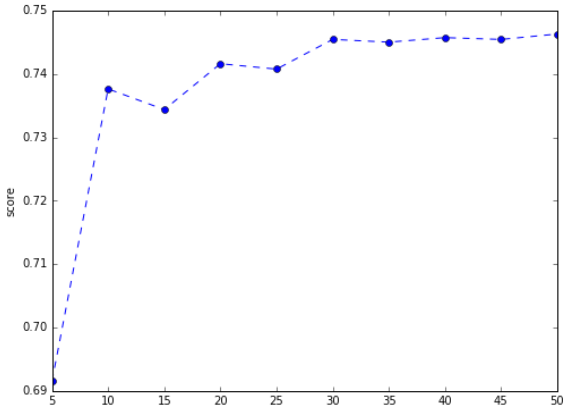


Figure 3.1: KNN validation

Although the accuracy is far from perfect, this result is actually quite significant. The "Online News Popularity Dataset" was originally assembled by Fernandes et. al. to develop an Intelligent Decision Support System for predicting popularity of online news, published in a paper tied to the data (Fernandes, Vinagre, & Cortez, September 2015). In that paper, the authors use 5 machine learning techniques, in combination with an optimization process, in order to make predictions. Their prediction target is similar to the **'unpopular'** target chosen by NewsWorthy. That is, those authors use the Number of Shares to classify articles as "popular" and "unpopular," and use a number of methods to address the associated classification problem. Like NewsWorthy, they also choose hold-out validation set to judge model performance. The significance of this paper to the News Worthy venture is that these original authors could generate a *highest* accuracy of only 67%. By implementing the essential math solution developed for NewsWorthy, our simple prototype was able to improve upon this accuracy by more than 5%! What is more, this proof-of-concept used rather simple methods of "unpopularity" cutoff and parameter choice. If higher-level statistical techniques were used to tune these parameters, this accuracy would likely only improve. The improved accuracy of News Worthy, using only a simple implementation, over that achieved by the original authors of the data supports the validity of the NewsWorthy ideas: using feature-importance to reduce the number of predictors, training flexible machine learning algorithms, and using ensemble learning to make final predictions.

*Predicting **'no_buzz'***

Using the feature importance attribute of Python's ExtraTreesClassifier class, we determine the top 10 ranked features for use in predicting the **'no_buzz**' variable. The top 10 features related to the **'no_buzz'** target are:

Table 3.2: Top 10 Important Features, 'no_buzz'

1	'n_tokens_title'	6	'num_keywords'
2	'num_self_hrefs'	7	'global_rate_positive_words'
3	'average_token_length'	8	'n_tokens_content'
4	'n_non_stop_unique_tokens'	9	'num_imgs'
5	'n_unique_tokens'	10	'avg_positive_polarity'

We now train two machine learning algorithms on the training subset restricted to these 10 predictors. First, we train a random forest classifier that uses 100 decision trees: each is given a bootstrapped sample of the data, and every "split" may only consider a random subset of the 10 predictors. Next we train a KNN model, where the neighbor parameter K is chosen by cross-validation on the training data. As before, we split the training data into 70% sub-training data, and 30% sub-validations set. We train 10 KNN models on the sub-training data, with K values of 5, 10, 15, 20, 25, 30, 35, 40, 45, and 50. The validation scores on the sub-validation set are displayed in Figure 3.2. Based on these validation scores, we choose the KNN model with K = 35. Given these two models, we allow each to make predictions on the held-out validation set. For each test instance, we make a final prediction of **'no_buzz'** = True if *either* of the individual models predicts **'no_buzz'** = True. This prediction strategy has an accuracy of 77.8% on the validation set.

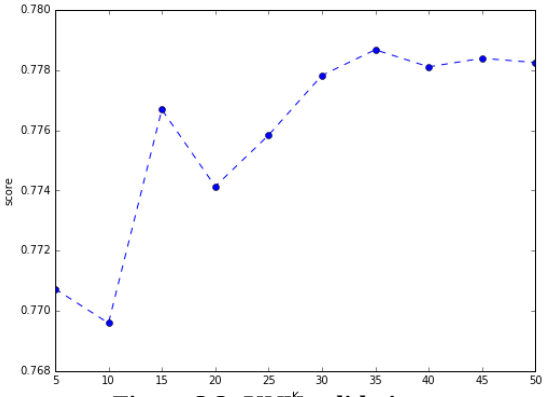


Figure 3.2: KNN validation

This prediction accuracy is relatively high, and indicates that the NewsWorthy methods are performing rather well when predicting when an article will have "no buzz." What is perhaps more important in this case is the *nature* of errors that are being made by our prediction strategy. The confusion matrix on the validation set is:

$$\begin{bmatrix} 9453 & 2343 \\ 1130 & 2726 \end{bmatrix}$$

A color plot of this confusion matrix is given in Figure 3.3. In this validation set there are 3,856 articles that generated "no buzz" (i.e. **'no_buzz'** = True). Among those, this ensemble learning prediction method correctly identifies 2,726: more than 70%. As mentioned previously, in the real-world setting of this problem the more costly error is false negatives: predicting that an *unpopular* article will be *popular*. In the confusion matrix this corresponds to "True Label" 1 and "Predicted Label" 0. NewsWorthy's ensemble prediction strategy performs relatively well overall−with a prediction accuracy of 78%−while also doing a good job of avoiding these more costly mistakes.

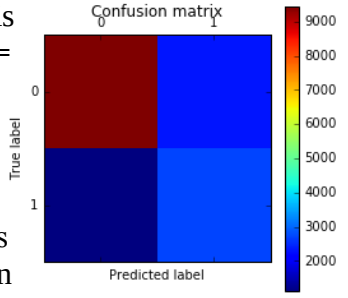


Figure 3.3: Confusion Matrix

BUSINESS GROWTH AND FURTHER APPLICATIONS

NewsWorthy has been developed specifically for implementation within the world of digital news media. However the ideas behind NewsWorthy−using structure-based attributes to predict when an article is likely to be "unpopular"−are in no way specialized to the news sector. Projecting the likelihood of failure of online content can prove extremely valuable to a wide variety of content generators, ranging from individuals such as bloggers, to large-scale advertisers. With some (rather small) modifications, NewsWorthy could be rolled-out as a whole suite of text-analysis software targeted at individual segments of the online publication market (which generated $39 Billion in revenue last year (Internet Publishing and Broadcasting in the US: Market Research Report)). Sister products such as TweetWorthy, BlogWorthy, or AdWorthy could be built on top of the NewsWorthy idea, expanding this method's (and our company's) applications. The potential for branch-out and rapid growth point to a bright future for the NewsWorthy venture. Thus while still on the ground-floor of our company's potential, NewsWorthy offers an investment opportunity with a high ceiling of future returns.

CONCLUSIONS

In this report, we described the market size and changing nature of the news media industry, and then introduced our new product, NewsWorthy. NewWorthy makes use of data science techniques to predict whether a news article is likely to be unpopular, *before publication*. Based on current trends, NewsWorthy defines "popularity" as the number of shares through social media. NewsWorthy's unique methods require structure-based metrics, including number of words, weekday of publication, length of title, polarity of title, and genre. As NewsWorthy doesn't require input of the text body, but rather this set of structural predictors, publishers can be confident that their pre-release content is not at risk of leak. NewsWorthy is the first and only pre-release popularity service in the market. Thus based on the growing value of digital media, we are confident our product is a superb investment opportunity for the Sharks.

In order to validate NewsWorthy's underlying ideas, we developed a proof-of-concept using a sample data set. In order to train machine-learning algorithms, we calculate 26 structure-based article attributes for a collection of training documents. Structure-based statistics are computable and comparable for any collection of articles, and also keep our data low-dimensional. NewsWorthy is designed to predict 2 target variables: **'unpopular'** and **'no_buzz'**. These correspond to the bottom 25% of articles' measure "popularity" and "buzz-factor." In order to aid in interpretability of results, and attempt to reduce over-fitting, we reduce the number of features considered to 10 (after all attribute constructions) using ExtraTreesClassifier. We perform feature selection on raw predictors, perhaps sacrificing some accuracy in favor of interpretability. For machine-learning, NewsWorthy uses Random Forest Classifier and KNN to predict target variables **'unpopular'** and **'no_buzz'**. Final predictions are made via ensemble learning, designed to reduce the number of false-negatives.

To test these methods on the sample data, we generated 10 important features for both **'unpopular'** and **'no_buzz'** and restricted our machine learning to these sets. A random forest model was developed using bagging and 100 decisions trees. Prior to KNN, we determined the optimal K value (for predicting each of **'unpopular'** and **'no_buzz'**) via cross-validation on the training set. We evaluated algorithm performance on a held-out test set, where we predict Target = True if random forest or KNN predicts Target = True. NewsWorthy's ensemble prediction strategy achieved a prediction accuracy of 73% when predicting **'unpopular'**, an improvement of 6% accuracy over the original authors of the data set. This validates NewsWorthy's underlying concept. In predicting **'no_buzz'**, NewsWorthy had an accuracy of 78% on the held-out test set, including a better than 70% true-positive rate.

APPENDIX 1: 90 SECOND "SHARK TANK" PITCH

Hello Sharks, we are News-Worthy. We are seeking $120,000 investment in return for 7% Equity in our company.

Consumption of News Media is rapidly changing. The internet recently replaced Television as the most popular source of News, and social media now out-paces newspapers. In this transition, the success or failure of each individual news article is becoming more important to publishers, especially in a competitive News media market that generated $63 Billion in US revenue last year.

News-Worthy is Software-as-a-Service targeted towards publishers and authors of online news. It is a pre-publication tool which predicts the likely popularity of a news article. This helps publishers avoid release of unpopular content, and distinguish themselves to potential advertisers.

News-Worthy's unique methods offer competitive advantages. We rely upon structure-based article statistics: such as sentence length and title polarity, as opposed to word-based analysis. This allows publishers to use our services with confidence that their pre-release content is not at risk.

News-Worthy has been developed specifically for online news. However, we are the first and only pre-release popularity service in the market. Online publishing generated $39 Billion in revenue last year. Branch-out software such as Tweet-Worthy or Blog-Worthy can easily be built upon News-Worthy's ideas, giving News-Worthy a high-ceiling for future growth.

Thank you for your time.

REFERENCES

3.2.4.3.3. sklearn.ensemble.ExtraTreesClassifier. (n.d.). Retrieved April 2016, from Scikit-Learn Documentation: http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.ExtraTreesClassifier.html

Fernandes, K., Vinagre, P., & Cortez, P. (September 2015). A Proactive Intelligent Decision Support System for Predicting the Popularity of Online News. *Proceedings of the 17th EPIA 2015-Porgugese Conference on Artificial Intelligence.* Coimbra, Portugal.

Holcomb, J., & Mitchell, A. (2014, March 26). *Revenue Sources: A Heavy Dependence on Advertising.* Retrieved April 2016, from Pew Research Center: Journalism and Media: http://www.journalism.org/2014/03/26/revenue-sources-a-heavy-dependence-on-advertising/

Internet Publishing and Broadcasting in the US: Market Research Report. (n.d.). Retrieved Aspril 2016, from IBISWorld.com: http://www.ibisworld.com/industry/default.aspx?indid=1974

Newman, N. (2015). *Executive Summary and Key Findings of the 2015 Report.* Retrieved April 2016, from Reuters Digital News Report 2015: http://www.digitalnewsreport.org/survey/2015/executive-summary-and-key-findings-2015/