# Predicting Ratings for New Movie Releases from Twitter Content

**Wernard Schmit**
Tilburg University
Postbus 90153
5000 LE Tilburg
w.h.w.schmit@uvt.nl

**Sander Wubben**
Tilburg University
Postbus 90153
5000 LE Tilburg
s.wubben@uvt.nl

## Abstract

With microblogging platforms such as Twitter generating huge amounts of textual data every day, the possibilities of knowledge discovery through Twitter data becomes increasingly relevant. Similar to the public voting mechanism on websites such as the Internet Movie Database (IMDb) that aggregates movies ratings, Twitter content contains reflections of public opinion about movies. This study aims to explore the use of Twitter content as textual data for predictive text mining. In this study, a corpus of tweets was compiled to predict the rating scores of newly released movies on IMDb. Predictions were done with several different machine learning algorithms, exploring both regression and classification methods. In addition, this study explores the use of several different kinds of textual features in the machine learning tasks. Results show that prediction performance based on textual features derived from our corpus of tweets improved on the baseline for both regression and classification tasks.

## 1 Introduction

Textual data from Twitter can be seen as an extensive source of information regarding an extremely broad variety of subjects. With millions of users actively expressing themselves online, a huge amount of data is generated every day. Since this data for a large part consists of human expressions, Twitter data can be seen as a valuable collection of human opinion or sentiment, which can be automatically extracted with relatively high accuracy (Pak & Paroubek, 2010).

Automatic sentiment analysis has been applied to many different fields, showing both scientific and commercial value. Sentiment analysis is a powerful way of discovering public attitude towards a variety of entities, including businesses and governments (Pang & Lee, 2008). Although brief of nature, tweets can serve as source of information regarding the overall appreciation of these entities. This has been demonstrated in a study that focused on brand management and the power of tweets as electronic word of mouth (Jansen, Zhang, Sobel, & Chowdury, 2009). Sentiment analysis is often treated as a classification task, by automatically predicting classes corresponding to sentiment values (Agarwal, Xie, Vovsha, Rambow, & Passonneau, 2011).

Besides extracting sentiment through classification, textual data has proven to be useful in machine learning tasks aimed at predicting numerical values. This type of predictive text mining has been applied in a useful way to economics, by making predictions of stock prices based on press releases (Mittermayer, 2004). Similarly, text mining has also been used to predict box office revenues of films, using a corpus of tweets (Asur & Huberman, 2010).

This study aims to continue the exploration of the predictive capabilities of Twitter data by using a corpus of tweets to predict rating scores of newly released movies on IMDb. The prediction of IMDb scores through data from social media has been explored before (Oghina, Breuss, Tsagkias, & De Rijke, 2012). However, this study differs from previous work by focusing solely on textual data from Twitter as opposed to non-textual data from other social media.



*Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA 2015)*, pages 122–126,
Lisboa, Portugal, 17 September, 2015. 

In order to explore the predictive capabilities of tweets, several machine learning experiments were conducted for this study. This includes regression experiments in order to predict the IMDB rating of the movie. Alternatively, this study also explores the prediction of classes corresponding to a range of numerical values: a classifcation approach. Both regression and classification methods have proven useful in the field of text mining, specifically concerning user sentiment (Pang & Lee, 2005).

## 2 Methodology

Several machine learning experiments were conducted for this study. These experiments required the collection and preprocessing of the Twitter corpus, which will be briefly discussed in the following sections, as well as the experimental setup.

### 2.1 Data collection and processing

Tweets were collected using Twitter's API. Between March 30th 2015 and April 28th 2015, Tweets were collected that mentioned one of 68 recently released movies. The IMDB scores of these movies ranged from 5.0 to 8.9 out of 10.

In order to eliminate uninformative tweets, all retweets and tweets containing hyperlinks were excluded from the dataset. Similarly, all Twitter usernames were removed from the tweets. All movie titles were replaced with the string: '<TITLE>' and the tweets were saved in tuples with their corresponding IMDb rating score. After preprocessing the data, the corpus consisted of 118,521 tweets usable for experimentation. This anonymized, preprocessed corpus has been made available online.[1] Examples of tuples with tweets and scores include: (*'just watched <TITLE> for the first time. absolutely fantastic film.'*, 8.5) and (*'<TITLE> would be a good movie if it didn't suck so much'*, 5.4).

The IMDb rating scores served as the target valuables in the regression experiments. For the classification experiments, classes were constructed as target valuables. The following classes corresponding to the IMDb scores were created for classification tasks:

- '*Very High'*: 8.0 and above (ca. 29K tweets)
- '*High'*: between 7.0 and 8.0 (ca. 42K tweets)
- '*Average*: between 6.0 and 7.0 (ca. 31K tweets)
- '*Low'*: between 5.0 and 6.0 (ca. 16k tweets)

We used a held out development set of 3400 tweets to optimize parameters for the machine learning experiments.

### 2.2 Experimental setup

The Python module Sci-Kit Learn was chosen as the tool for the machine learning experiments.[2] Sci-Kit Learn provides options for various machine learning algorithms usable for both regression and classification tasks. This module makes a convenient tool for our machine learning tasks.

For the machine learning experiments we used textual features from the tweets as input, and performance scores after 10-fold cross validation as output, similar to previous experiments in this field (Oghina, Breuss, Tsagkias, & De Rijke, 2012). For regression tasks, the mean-squared error (MSE) was used as the performance metric, as this metric takes the severity of the prediction errors into account. For this metric, lower scores mean better results (Witten, Frank, & Hall, 2011). Classification tasks used F1-scores to measure performance (Witten, Frank, & Hall, 2011).

Since our data is not evenly distributed among classes (popular movies generate more tweets), our experiments used baselines for comparison that take into account the distribution of the dataset. Regression performances were compared to a baseline performance of predictions based on the mean of the target valuables. Classification performances were compared to baseline performance of stratified predictions: a classifier that makes predictions based on the data distribution over classes.

### 2.3 Features

Features were constructed from the textual content of the tweets. *N*-grams in tweets were transformed into numeric TF-IDF vectors, similar to the predictive text mining experiment of Mittermayer (2004). TF-IDF vectors were incorporated in order to appropriately apply weights to the terms in our corpus.

Experiments were run with several ranges of *n*-grams as basis for the TF-IDF vectors. The use of unigrams, bigrams, trigrams and combinations of these *n*-grams was explored in experimentation on

[1] https://dl.dropboxusercontent.com/u/20984922/Dataset%20Tweets%20%2B%20IMDb%20Rating%20Scores.csv

[2] http://www.scikit-learn.org

a held out development set. Additionally, the use of *stemming* was explored by applying a Porter Stemmer from Python module NLTK.[3] This was done in order to reduce model complexity (Meyer, Hornik, & Feinerer, 2008). The constructed TF-IDF vectors for the (stemmed) *n*-grams were used as training input for the machine learning algorithms.

### 2.4 Machine learning algorithms

For both regression and classification tasks, several different algorithms were used for experimentation. For regression tasks, we used the Linear Regression (LR) and Support Vector Regression (SVR) implementations from Sci-Kit Learn. Both algorithms have been used successfully in previous experiments. LR was used in a previous experiment regarding the prediction of IMDb rating scores (Oghina, Breuss, Tsagkias, & De Rijke, 2012). SVR has been used similarly for predicting ordinal sentiment scores (Pang & Lee, 2005).

For classification tasks, Support Vector Classification (SVC) and Stochastic Gradient Descent Classification (SGD) were used. SGD is considered a useful algorithm for experiments with large amounts of training data (Bespalov, Bai, Qi, & Shokoufandeh, 2011). Similar to SVR, the use of support vector machines can lead to accurate decision boundaries for classification tasks (Gunn, 1998). The SGD implementation used a hinged loss function, similar to the loss function used in SVC.

For both SVR and SVC an automatic grid search was performed on the development set to determine the optimal parameters. This grid search showed that for both SVR and SVC a linear kernel and a C value of 1.0 led to the best performance results.

## 3 Results

While both regression and classification experiments used the same features, performances were different between regression and classification tasks. This section shows the results for the best performing configurations for both regression and classification tasks.

### 3.1 Regression results

| ***N*-grams** | **Stems** | **Algorithm** | **Baseline** | **MSE** |
|---|---|---|---|---|
| Unigrams, Bigrams, Trigrams | YES | SVR | .998 | **.529** |
| Unigrams, Bigrams | YES | SVR | .998 | .536 |
| Unigrams, Bigrams, Trigrams | YES | LR | .998 | .569 |

Table 1: Best regression results

The best performing regression configurations show a relatively large improvement on the baseline, as can be witnessed in Table 1. Results show that the best regression result is achieved by using the SVR algorithm on stemmed combinations of unigrams, bigrams and trigrams. For the three best configurations, combinations of *n*-grams yielded the best results, when combined with stemming.

Experimentation with different amounts of training data show that results improved with larger amounts of data. Figure 1 shows the learning curve for the best performing regression configuration, performing 10-fold cross validation for each experiment. This curve shows that it is likely that performance will improve with more data than was used in these experiments.

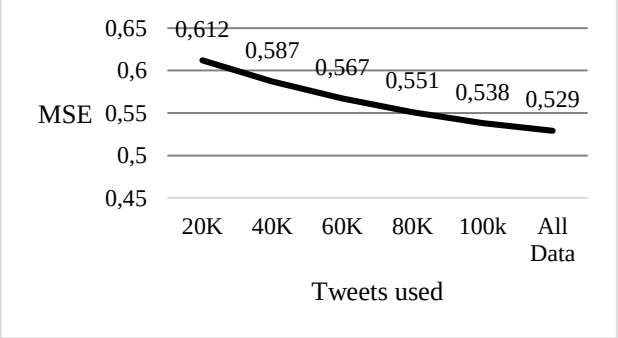


Figure 1: Learning curve regression

[3] http://www.nltk.org.

### 3.2 Classification results

| *N*-grams | Stems | Algorithm | Baseline | F1 |
|---|---|---|---|---|
| Unigrams, Bigrams, Trigrams | YES | SVC | .274 | **.534** |
| Unigrams, Bigrams, Trigrams | NO | SVC | .274 | .529 |
| Unigrams, Bigrams, Trigrams | YES | SGD | .274 | .529 |

Table 2: Best classification results

Table 2 shows that the best performing classification configurations also managed to improve over the baseline. The best performing configuration used stemmed combinations of unigrams, bigrams and trigrams and the SVC algorithm. The three best performing configurations all show that the combination of these *n*-grams leads to the best results. However, the use of stemming is not always required to achieve a relatively high performance, as is shown by the second best performing configuration.

Experiments with different amounts of training data for the best performing classification configuration again show that more data led to better results. These experiments again used 10-fold cross validation for each experiment. The learning curve for the best performing classification configuration shows that it is likely that the optimal amount of training data has not yet been reached.

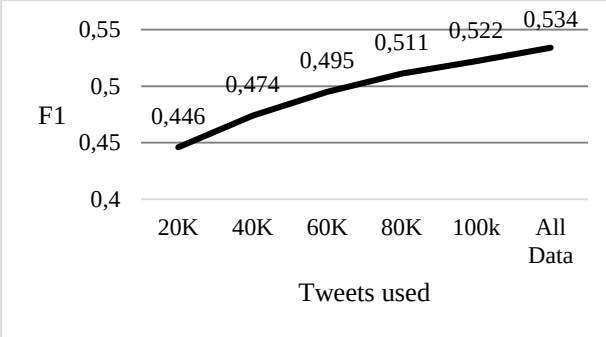


Figure 2: Learning curve for classification

## 4 Conclusion

As the results of the experiments show, IMDb rating scores can be predicted to a certain extent using a supervised machine learning approach. Both the prediction of exact numerical rating scores and the prediction of classes corresponding to a range of numerical scores, achieved a certain degree of success compared to their respective baselines.

The best performing regression configuration achieved an MSE of .529. This was achieved by using stemmed combinations of unigrams, bigrams and trigrams. While this configuration led to an improvement on the baseline of mean predictions, which achieved an MSE of .998, there is still room for improvement. The best performing configuration of Oghina, Breuss, Tsagkias, & De Rijke (2012) achieved a RMSE of .523 for the prediction of IMDb rating scores, which translates to an MSE of .273. This model clearly outperforms our best performing configuration. However, our experiments focus solely on textual features derived from Twitter, as opposed to also including numerical features from other social media. Furthermore, in their model, more than 1,6 milion tweets were used, whereas this study used a dataset consisting of roughly 118K tweets. It can be concluded that our best performing model is not the optimal prediction model for IMDb scores, but it does show that textual features can be useful for prediction of this kind.

Classification results also showed that predicting IMDb rating scores using tweets as training data can have a certain degree of success. The best performing configuration had an F1- score of .534, while the stratified baseline achieved an F1-score of .274, based on predictions according to the class distribution of the training data.

Our classification results can be compared to other studies that performed classification tasks. The study of Agarwal, Xie, Vovsha, Rambow, & Passonneau (2011) explored 3-way classification for sentiment analysis. Their best performing model attained an F1-score of .605. This is higher than our best performing score, but note that our experiments dealt with one more target variable. It should also be noted that this study deals with more general sentiment analysis, while our study is specifically aimed at predicting classes corresponding to IMDb scores. Our results show that a classification approach can be useful in predicting these classes.

## 5 Discussion

While this study has shown some interesting results regarding the predictive capabilities of tweets, there remains plenty of room for future research. There are more possibilities to explore regarding the dataset, the algorithms and the features. Learning curves show that it is likely that the optimal amount of data was not used in these experiments, which is something to be explored. Additionally, this study shows that the use of

stemming and combinations of *n*-grams should always be explored.

This study shows that using merely textual features is not the optimal method of predicting scores on IMDb, as the model of Oghina, Breuss, Tsagkias, & De Rijke (2012) clearly outperforms our configurations, which expanded on merely using textual features. For future research, if the goal is to optimize these predictions, it is clear that expanding on textual features is wise, for example by including metadata from the tweets. A well functioning system that uses data from social media could serve as a barometer that forecasts appreciation of newly released films. Such a system would also provide insight into the opinions of a different population of the internet rather than merely IMDb voters.

When focusing specifically on the predictive capabilites of textual data from Twitter, there are other options to consider for future research. Features used in our experiments can prove valuable, but different options should also be explored. For example, the use of character *n*-grams may prove useful. Similarly, the ratio of positive to negative tweets as a feature may lead to better predictions. This would require first performing sentiment classification on the tweets, before attempting to predict the IMDb scores.

Besides further possibilities regarding the size of the dataset and feature engineering, other machine learning algorithms can also be explored. Different algorithms are better suited for datasets of different sizes, it is worth researching which algorithms lead to the best performance for different sizes of training data. By continuing research in this field, predictive possibilities of tweets can be further explored, discovered and applied, not merely for IMDb scores, but for many different fields.

## References


Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. (2011). Sentiment analysis of Twitter data. *LSM '11 Proceedings of the Workshop on Languages in Social Media*, Pages 30-38.

Asur, S., & Huberman, B. A. (2010). *Predicting the Future With Social Media.* Palo Alto: HP Labs.

Bespalov, D., Bai, B., Qi, Y., & Shokoufandeh, A. (2011). *Sentiment Classification Based on Supervised Latent.* Princeton, NJ: NEC Labs America.

Gunn, S. (1998). *Support Vector Machines for Classification and Regression.* Southampton: University of Southampton.

Jansen, B., Zhang, M., Sobel, K., & Chowdury, A. (2009). Twitter Power: Tweets as Electroninc Word of Mouth. *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE AND TECHNOLOGY*, 2169-2188.

Meyer, D., Hornik, K., & Feinerer, I. (2008). Text mining infrastructure in R. *Journal of Statistical Software*, 1-54.

Mittermayer, M. (2004). Forecasting Intraday Stock Price Trends with Text Mining Techniques. *Proceedings 37th Annual Hawaii Int, Conference on System Sciences (HICSS)*, (p. 64). Big Island.

Oghina, A., Breuss, M., Tsagkias, M., & De Rijke, M. (2012). *Predicting IMDB Movie Ratings Using Social Media.* Amsterdam: ISLA.

Pak, A., & Paroubek, P. (2010). *Twitter as a Corpus for Sentiment Analysis and Opinion Mining.* Orsay Cedex, France: Laboratoire LIMSI-CNRS.

Pang, B., & Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *Proceedings of ACL 2005*.

Pang, B., & Lee, L. (2008). *Opinion Mining and Sentiment Analysis.* Sunnyvale, CA: Yahoo! Research.

Witten, I., Frank, E., & Hall, M. (2011). *Data Mining: Practical Machine Learning Tools and Techniques.* Amsterdam: Morgan Kaufman.