# Box Office Success Prediction based on Movie Attributes

Jonathan Sumrall and Natalia Kuznetsova
Technical University of Eindhoven

## Goal

*Using only the attributes of a movie, predict whether or not it will be a success in the box office.*

## Data

We collected data from IMDB.com, who provide their database in a human-readable list (LST) format. IMDBPy builds a SQL database from the data, and provides methods for querying the data. We make a new collection of only the movies which contain the attributes we are interested in.

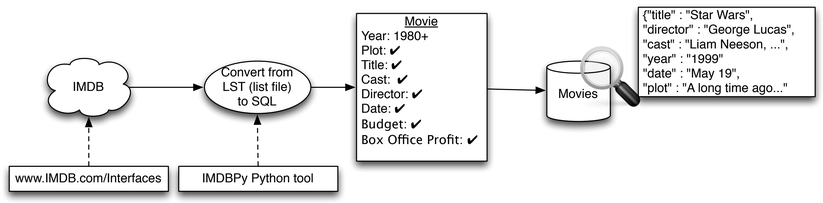



Figure: Data collection process

We give a score to each movie in our collection based on how many famous actors were in the movie, how experienced the director is, and whether or not the movie premiers near a holiday.

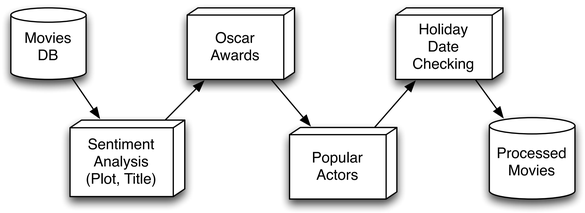



Figure: Scoring movies based on actors, director experience, and premier date.

## Sentiment analysis

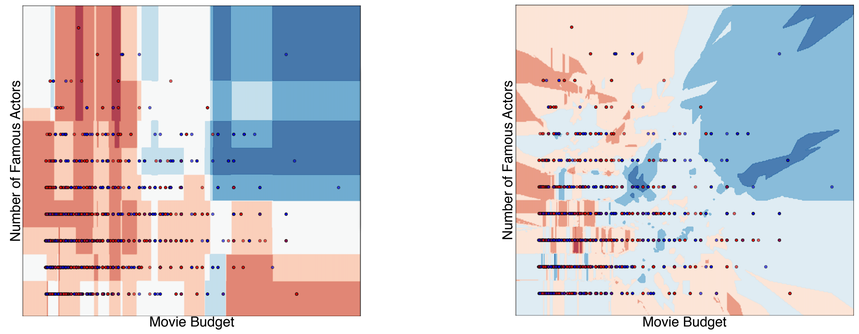
The plots on the IMDB web site are written by users, therefore we hypothesize that they contain emotional sentiment. We apply Sentiment Analysis on the plots to convert them to a numerical representation. Two approaches were tried and compared.

| **NLTK Approach** | **Web search approach** |
|---|---|
| Use Natural Language Toolkit's Naive Bayes Classifier | Compute average semantic orientation with the use of web search engine queries |
| 2000 training set from NLTK corpus | the Internet as a training set |
| Query words after tokenization | Query phrases |
| Take the proportion of positive and negative frequencies | Take the distance from positive and negative evaluation |
| The size of training set | The number of queries is restricted |

Due to the restriction the NLTK approach was chosen. Although the movie plot descriptions are written by users, the sentiment analysis does not make sense since the estimates are around 0 and hence there is no correlation between the plot and movie success.

## Classification

From our collection of movies, we try to classify them as either a Success or a Failure. An accurate definition of a movie's box office success is difficult to make, as a successful movie could have a long and profitable run in theaters after the first weekend, it could have excellent DVD sales, or it could be syndicated on television for years later. It could have all of these but not have a very profitable opening weekend. We label success as: $\textit{Opening Weekend Gross} > \textit{Budget} * 0.3$ The best classification algorithms for this problem are Gradient Boosting and K Nearest Neighbors. The Python library Scikit-Learn was used for the implementation of the machine learning algorithms. The probability of correctly classifying a movie is 64% with Gradient Boosting and 60% with K Nearest Neighbors.

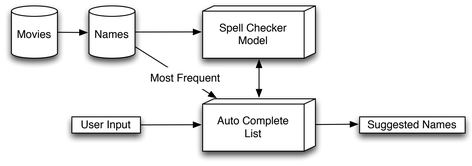



(a) Gradient Boosting (b) KNeighbors, K=15

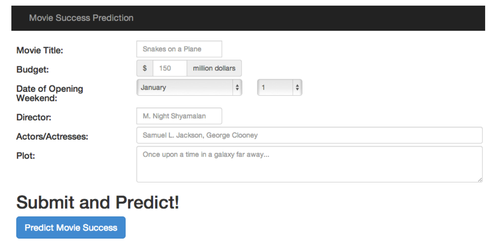
Figure: Classification of movies, Blue is success

## Tool and Features

The tool is a web site where users can enter information about a movie including a title, budget, premier date, director, cast, and a plot description. It is necessary to enter the names correctly, thus the functions of autocorrection and autocompletion is essential.




(a) Autocomplete and autocorrect procedure




## Conclusions

*The tool predicts the success of the described movie with an accuracy greater than random or naïve guessing. But overall the prediction is not as accurate as we thought it would be. There are other aspects of the data which could analyzed to yield better results, and a movie's success depends on other factors not included in the movie's attributes.*