Mining the Success for Movies

Student Project Data Mining HWS17
Team 6

Presented by

Steffen Jung
Adrian Kochsiek
Martin Koller
Marvin Messenzehl
Daniel Szymkowiak

Submitted to the
Data and Web Science Group
Prof. Dr. Heiko Paulheim
University of Mannheim

October - December 2017

Contents

Chapter 1

Application Area and Goals

This paper represents a documentation for the data mining project *"Mining the Success for Movies"*[1]. The structure of this paper follows the classical data mining process. Chapter 1 provides an overview of the problem the project is based on and is complemented by the goals and objectives of this project. Afterwards, chapter 2 deals with the structure and size of the data. Here, a closer look will be taken at the dataset at hand. Questions that had to be answered were for example which information were provided in the original dataset or which problems were identified concerning for instance outliers or missing values. Upon that, chapter 3 explains which preprocessing steps had to be taken in order to cleanse the dataset to prepare it for the data mining step and model learning. Chapter 4 describes which data mining techniques regarding algorithms and parameters were used to learn an expedient model in respect of the goals set in Chapter 1. Finally, chapter 5 closes this paper by describing which insights could be won for the problem at hand. Here, a critical reflection is delineated how the model could be improved further in order to provide even more precise results.

1.1 Problem Statement

Already well before new movies are being produced, every stakeholder certainly is interested in the monetary success of the given movie. In order to predict the success costly methods are being applied, such as market investigations or deep analysis and .

The benefit which Data Mining brings to the analysis of large datasets, can also be transferred onto the stated problem of predicting a movie's success. Based on

[1] Information in this paper refers to the (sample) dataset and python scripts handed in for a classification problem

given data of already released movies and successful movies in the past, a model is being learned which shall be applied on upcoming or planned movies. In order to learn and apply the model various pieces of information are taken into account. Just a few are budget, revenues, runtime, genre and information on the release. Information on the dataset and on all preprocessing methods which were applied will be provided in chapter 2.

1.2 Goals

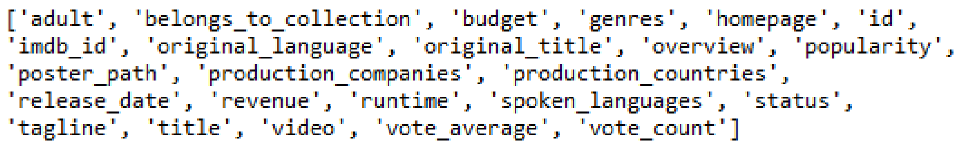
The goal of this project is to learn a model which will predict how successful a not yet released movie will be. This is done by using common data mining techniques in the Python programming language. As a main objective the question *"Based on revenue, will the movie be popular or will it be a flop?"* shall be answered for all possible combinations of information on a new movie as precisely as possible. In order to be as precisely as possible, not only different algorithms are being tested, but also parameter tuning is being applied with different performance measures [2].

[2]Further information on applied techniques and evaluation methods is provided in chapter 4

Chapter 2

Data Selection

The selected dataset onto which a classification model shall be learned is provided by Kaggle [1]. It is named *The movies Dataset*[2] and contains metadata on approximately 45,000 movies in its raw format. It is provided and updated by Rounik Banik. The complete dataset consits out of several files in the *.csv* format containing specific info about movie casts, and external score. For the outcome of this project the central file, used for further preprocessing is named *movies-metadata.csv*. This csv-file holds 24 columns in total, which can be expected in the graphic below.



```
['adult', 'belongs_to_collection', 'budget', 'genres', 'homepage', 'id',
'imdb_id', 'original_language', 'original_title', 'overview', 'popularity',
'poster_path', 'production_companies', 'production_countries',
'release_date', 'revenue', 'runtime', 'spoken_languages', 'status',
'tagline', 'title', 'video', 'vote_average', 'vote_count']
```

Untertitel bla bla

2.1 Structure and size of data

2.2 Basic data exploration

- In slides named: ”structure and size of data”
- min. 1 Page
- Selection:

[1]2017 Kaggle Inc
[2]Link to the dataset: https://www.kaggle.com/rounakbanik/the-movies-dataset

  - What data is available?
  - What do I know about the provenance of the data?
  - What do I know about the quality of the data?
  - Wrong values, lot of null values

- Exploration
  - Get an initial understanding of the data
  - Calculate basic summarization statistics
  - Visualize the data
  - Identify data problems such as outliers, missing values, duplicate records
  - problem with number scales, data formats, truth of contents

Main issue quality of data. Further explained in preprocessing...

Chapter 3

Preprocessing and Transformation

In order to pick out columns that have a significant impact on forecasting a movie's success, the following assumptions concerning the importance of information in the metadata were made:

- The budget and revenue are crucial numbers.
- The release year has an impact on numbers such as budget and revenue.
- The genre, the production company and the runtime are important information.
- Transform data into a representation that is suitable for the chosen data mining methods
  - number of dimensions
  - scales of attributes (nominal, ordinal, numeric)
  - amount of data (determines hardware requirements)
- Methods
  - Aggregation, sampling
  - Dimensionality reduction / feature subset selection
  - Attribute transformation / text to term vector
  - Discretization and binarization
- Good data preparation is key to producing valid and reliable models

- Data preparation estimated to take 70-80% of the time and effort of a data mining project!

3.1 Preprocessing steps according to Python script

3.2 A list of problems we encountered

1. **list further problems we had and solved!**
2. Prod. Comp.: Same prod. company named differently -¿ using Regex to solve (Steffen)
3. dataset: 5 datasets have duplicates

Chapter 4

Data Mining

After preprocessing the data, the next step is to build a classifier to predict the success of a movie. To find the best classifier a set of different algorithms is evaluated:

- K-Nearest Neighbor
- Naive Bayes
- Support Vector Classifier
- Neural Net
- **Decision Tree**
- **Random Forest**

The first goal of the analysis was to predict the success in five different classes. Since an analysis in that detail with the given data set is very unprecise as seen in table 4.1, a binary classifier was created. For all of the named algorithms an individual feature selection is applied to get the best results. The important features are selected by a greedy feature wrapper. This feature wrapper starts with the full set of features and drops features as long as the result improves. The results are checked with a ten fold cross-validation. Additionally to the selection of features with the feature wrapper, filters for features like the number of actors and number of production companies can be applied, to only use x% most occurring of them.
In the next step some hyperparameter tuning is needed to improve the classifiers. To find the best parameter setting, a gridsearch in combination with a ten-fold cross-validation, scoring the highest F1-score (macro), is applied to the classifiers. The achieved F1-scores of the three best binary classifier are listed in table 4.2. At first the classifiers are scoring the micro F1-score. Even though the result looks

Algorithm	F1 Macro
Decision Tree	36.2%
Random Forest	40.4%
Support Vector Classifier	37.5%

Table 4.1: Multi Label Classifier Results

Algorithm	F1 Macro	F1 Micro	Downsampled Macro	Downsampled Micro
Decision Tree	56.5%	75.5%	61.9%	63.0%
Random Forest	58.7%	76.0%	80.6%	81.5%
Support Vector Classifier	56.5%	75.0%	60.7%	60.8%

Table 4.2: Binary Classifier Results

promising at the beginning, the classifier are mostly guessing the majority class in this setting. For that reason every classifier is scoring on the macro F1-score. Since the data set for the binary classifier is unbalanced with a ratio of 76% ”successful” and 24% ”unsuccessful”, the next step to improve the classifiers is to balance the train set. That for the stratified cross validation is used to balance the train set. For further improvement the data set can be downbalanced by shrinking the majority class of the train set to the size of the minority class. But since the data set is quite small already after the preprocessing, the downsampling worsens the classifiers. To not shrink the dataset any further upsampling can be used to increase the minority class of the train set to the size of the majority class. In the following section the decision tree, random forest and the support vector classifier are examined in more detail.

4.1 Decision Tree

One way to deal with problem of an unbalanced data set is to use a tree-structured classifier, since the hierarchical structure allows them to learn signals from both classes. To find the best parameter setting for the decision tree the gridsearch tests the parameters splitcriterion, max depth, minimum samples to split and the class weight. After dropping the ”quarter”, ”runtime” and ”adult” columns, the classifier scores without any sampling, no class weight, an entropy split criterion and a max depth of 100 an F1 score of 58%. Both the down- and the upsampling worsens the classifier.

4.2 Random Forest

Like the decision tree, random forest builds a hierarchical structure, which can handle unbalanced data sets better than other algorithms. In addition it corrects the overfitting habit of a decision tree. In the gridsearch hyperparameters like the split-criterion, number of features, the minimum sample to split and whether bootstrap samples are used or not are evaluated. After dropping the actors and the release quarter, the classifiers scores a F1 score of 59%. As with the decision tree neither down- nor upsampling the train set improves the result.

4.3 Support Vector Classifier

Chapter 5

Interpretation / Evaluation

Maybe also prospect?

- Output of Data Mining
  - Patterns
  - Models
- In the end, we want to derive value from that, e.g.,
  - gain knowledge
  - make better decisions
  - increase revenue