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 **??** 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 Data

In total, two datasets were used to create features for testing the different classifiers: *movies_metadata.csv* and *credits.csv*. The dataset *movies_metadata.csv* contains 45,463 rows and 23 columns excluding the id-column. The dataset *credits.csv* contains 45,463 rows and 2 columns excluding the id-column.

3.1 Vertical scaling

Both dimensions for the dataset *movies_metadata.csv* were processed. As section 3.3 explains, the revenue and budget played a major role for the data mining task. Therefore all datasets without information on each column had to be dropped out, which was a major part of the dataset. Also duplicates were identified using the ID and discarded. After this, the dataset shrunk to roughly 4000 rows. To retain some numbers and still use datasets which had only missing out revenue or budget two approaches were implemented: A parser for the IMDB database [1] API and for the The Numbers[2] API was programmed and run. **Was war hier das Problem?**.

3.2 Horizontal scaling

Based on the assumptions that budget and revenue were crucial numbers, the release year has an impact on those numbers and the genre, the production country such as production company, the spoken languages, the runtime and the fact whether a movie belongs to a collection or not are further important information, all others columns were dropped out from the *movies_metadata.csv*. **Nicht gerade**

[1] The IMDB database: http://www.imdb.com/

[2] the numbers website: http://www.the-numbers.com/research-analysis

wisschenschaftlich hier Figure 3.1 gives an overview on which information was retained.

The reason why information, which could have been potentially interesting, had to be dropped, was mainly for time reasons. Preprocessing took about 70% of the timeperiod[3] of the whole project. Thus, the team was able to focus on preprocessing of mentioned columns. Still, chapter 5 provides a prospect, which steps were possible, if a larger timeframe was dedicated to this project.

After dropping out information, eleven columns remained. Combined with the two columns from the *credits.csv* dataset, thirteen columns were used as a basis to create features for finding the best performing classifiers.

In order to transform the data into a suitable representation for forecasting a movie's success, preprocessing was mandatory. For each column zero or more preprocessing steps from the following list were performed:

- Merging of columns
- Binning of features
- Extracting information out of columns
- One hot encoding
- Normalizing

The following sections explain the preprocessing in detail and provide Python code-snippets. Figure 3.2 shows precisely, which operations were executed on each column and how the data types changed.

3.3 Merging and creating columns

When a new movie is planned, the finances are one of the most important concerns. Where the budget can be circumscribed upfront, revenue is nearly impossible to guess. As a result, the prediction of a model should consider the revenue as a key factor for it's monetary success. To only predict the revenue (using multiple bins or binary binning, like e.g. "will the revenue of a new movie be higher or lower than $500,000?") would not have worked out due to multiple reasons: Both the revenue and budget of a movie in earlier years, like e.g. the 1950's, was considerably less than today, so total numbers are not comparable. Additionally, inflation plays a role in comparing financial numbers of elder movies to newer ones. Furthermore, the

[3] Mainly due to the fact that heaps of problems arose from the dataset, which can be read in chapter 2.

adult	False
belongs_to_collection	{'id': 10194, 'name': 'Toy Story Collection', ...
budget	30000000
genres	[{'id': 16, 'name': 'Animation'}, {'id': 35, '...
homepage	http://toystory.disney.com/toy-story
imdb_id	tt0114709
original_language	en
original_title	Toy Story
overview	Led by Woody, Andy's toys live happily in his ...
popularity	21.9469
poster_path	/rhIRbceoE9lR4veEXuwCC2wARtG.jpg
production_companies	[{'name': 'Pixar Animation Studios', 'id': 3}]
production_countries	[{'iso_3166_1': 'US', 'name': 'United States o...
release_date	1995-10-30
revenue	373554033
runtime	81
spoken_languages	[{'iso_639_1': 'en', 'name': 'English'}]
status	Released
tagline	NaN
title	Toy Story
video	False
vote_average	7.7
vote_count	5415

Figure 3.1: Dropped columns of *movies_metadata.csv*. All retained columns are marked in yellow

dataset contained different currencies like dollars, euros or indian rupees without indicating which currency was provided per dataset. This is why a new column was added, namely the productivity. It is a quotient, computed by dividing revenue through budget. If the productivity is higher than one, the movie derives profit, if the productivity is less than one, the movie derives a loss. That way, above mentioned issues can be avoided. The column revenue was dropped afterwards.

Considering the release date of a movie, the assumption was made that the demand for movies is higher in quarter four of the year (time of winter and christmas). This was confirmed by checking the numbers[4]. Hence, a new column of numeric type with quarter one to four was created. The previous column release date was converted to release year only as a numeric format.

3.4 Discretization

During this preprocessing step, the created column productivity was binned two ways: Once multi-binned into four different bins having bins between 0.0 to 0.99 (label "*unproductive*"), 1.0 to 1.99 (label "*smallProductivity*"), 2.0 to 4.99 (label "*goodProductivity*") and 5.0 to infinity (label "*highProductivity*"), and once binary-binned into two bins having 0.0 to 0.9 (label "*no*") and 1.0 (label "*yes*") to infinity. For each bin a new column was added, the former productivity column was dropped.

3.5 Extracting information

Regex + Bild von Zelle und code

3.6 One hot encoding

Not only the in the previous step extracted information on genre, production country, production company director and actor was one hot encoded using the pandas get_dummies()[5] function, but also the original language. After one-hot-encoding, the dataset consisted of 7540 columns[6].

[4]Check for details on revenues in video-selling: https://de.statista.com/statistik/daten/studie/182319/umfrage/umsatzentwicklung-im-video-kaufmarkt-quartalszahlen/

[5]The Documentation on pandas get_dummies() function can be found online

[6]The high number is due to the high number of different actors, directors, production companies and production countries. When creating a model, a threshold was used to exclude rare occurrences for each feature and **aggregate them in a bin "other"**

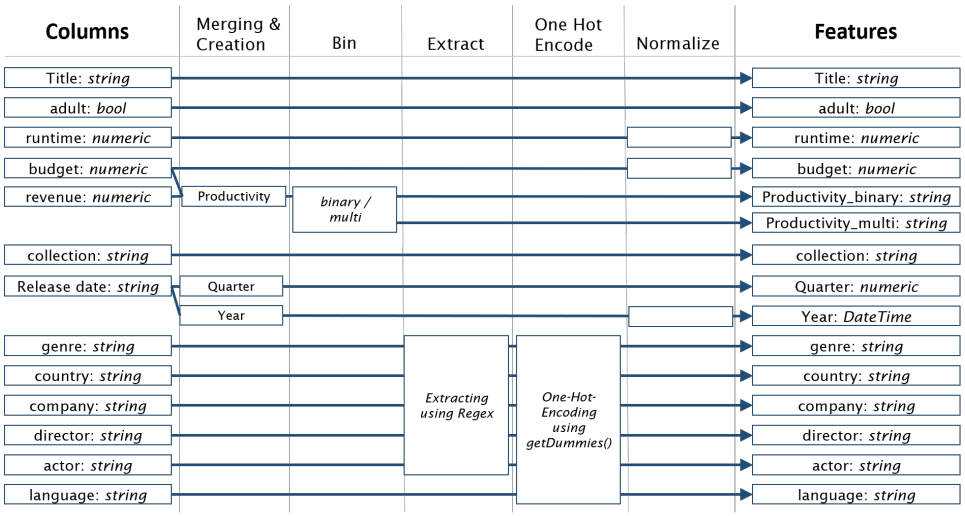

Figure 3.2: Features created during preprocessing

3.7 Normalizing

As a last step the remaining continuous numeric columns were normalized on a scale from zero to one: the runtime, the budget and the year.

3.8 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

- Input: Preprocessed Data
- Output: Model / Patterns

1. Apply data mining method
2. Evaluate resulting model / patterns (using P, R, F1, not accuracy)
3. Iterate:
    - Experiment with different parameter settings
    - Experiment with different alternative methods Improve preprocessing and feature generation Combine different methods

4.1 Algorithms

- **Random Forest**
- **Decision Tree**
- KNN
- Bayes
- NeuralNet
- svc

4.2 Three best performing algorithms

- Pick best three algos
- GridSearch
- Why does each classifier perform how it performs (unausgeglichene Klassen, ...)?

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