NOVA IMS Information Management School

MGI

Mestrado em Gestão de Informação
Master Program in Information Management

Movie's box office performance prediction

An approach based on movie's script, text mining and deep learning

Rafael Castilho Corrêa de Sá

Dissertation presented as a partial requirement to achievement of master's degree in information management

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

Movie's box office performance prediction

An approach based on movie's script, text mining and deep learning

Rafael Castilho Corrêa de Sá

Dissertation presented as the partial requirement for obtaining a Master's degree in Information Management, Specialization in Knowledge Management and Business Intelligence

Advisor / Co-supervisor: Roberto André Pereira Henriques

March 2020

RESUMO

A capacidade de prever a bilheteria de filmes tem sido atividade de grande interesse para investigadores. Entretanto, parcela significativa destes estudos concentra-se no uso de variáveis disponíveis apenas nos estágios de produção e pós-produção de filmes. O objetivo deste trabalho é desenvolver um modelo preditivo de bilheteria baseando-se apenas em informações dos roteiros dos filmes, por meio do uso de técnicas de processamento de linguagem natural (PLN), mineração de texto e de redes neuronais profundas. Essa abordagem visa otimizar a tomada de decisão de investidores em uma fase ainda inicial dos projetos, com foco específico na melhoria dos processos seletivos da Agência Nacional do Cinema do Brasil.

PALAVRAS-CHAVE

Mineração de Texto; Roteiros de filmes; Bilheteria; Redes neuronais; Modelagem Preditiva;

ABSTRACT

The ability to predict movies box-office has been a field of interest for many researchers. However, most of these studies are concentrated on variables that are available only in later stages as in production and pos-production phase of films. The objective of this work is to develop a predictive model to forecast movie box-office performance based only on information in the movie script, using natural language processing techniques, text mining and deep learning neural networks. This approach aims to optimize the investor's decision-making process at earlier steps of the project, with special focus on the selection process of the Brazilian Film Agency (ANCINE – Agência Nacional do cinema).

KEYWORDS

Predictive modelling; Word Embeddings; Deep Learning; Movie Script; Box-office

INDEX

1. Introduction 8
   1.1. Problem identification and contextualization 8
   1.2. Contributions 10
   1.3. Study's objectives 10
   1.4. Relevance and importance of the study 10
2. Literature review 12
3. Methodology 16
   3.1. Business Understanding 17
      3.1.1. Market Understanding 17
      3.1.2. Business Context 21
   3.2. Data Understanding 24
      3.2.1. Data Understanding Macroprocess 25
      3.2.2. Data Extraction Techniques and Tools 25
      3.2.3. Data Extraction "The Numbers" 27
      3.2.4. Movie's script Data Extraction 28
      3.2.5. Joining Movie Scripts dataset with Box Office Dataset 29
      3.2.6. Data exploration and quality assessment 30
   3.3. Data Preparation 31
      3.3.1. Splitting text into words (Tokenization) 31
      3.3.2. Expanding contracted forms of words 31
      3.3.3. Removing punctuation and Normalizing Case 32
      3.3.4. Removing stop words 32
      3.3.5. Removing General data quality problems 32
      3.3.6. Word Embeddings 32
      3.3.7. Target variable Discretization 38
   3.4. Modelling 40
      3.4.1. Support Vector Machines 40
      3.4.2. Artificial Neural Networks 42
      3.4.3. Convolutional Neural Networks 45
      3.4.4. Data Partition 46
      3.4.5. Model performance Evaluation 47
   3.5. Evaluation 50
4. RESULTS AND DISCUSSION 52

4.1. Gross Revenue (Binary format) 52
4.2. Gross Revenue (4 Classes Format) 55
4.3. Tickets Sold (Binary Format).....56
4.4. Tickets Sold (4 Classes format).....58
5. Conclusion 60
6. Limitations and recommendations for future work.....62
7. BIBLIOGRAPHY.....63

INDEX OF FIGURES

Figure 1 – CRISP-DM reference Model 16
Figure 2 – ANCINE's organogram 21
Figure 3 – Data Understanding Macroprocess 24
Figure 4 – Data Extraction Macroprocess 25
Figure 5 – HTML parse tree 26
Figure 6 – Movie's script dataset creation process 28
Figure 7 – Movie's script web scrapping process 28
Figure 8 – Word2Vec architectures - CBOW and skip-gram schemas 34
Figure 9 – Glove's embeddings examples 36
Figure 10 – Model architecture of FastText for a sentence with N n-gram features $W_1 \ldots W_N$. The features are embedded and averaged to form the hidden variable.37
Figure 11 – Linear SVM schema (Raschka, Julian, & Hearty, 2017) 41
Figure 12 – Schematic representation of a perceptron and a human neuron structure43
Figure 13 – 2-Hidden layers MLP example. Adapted from CS231n Convolutional neural networks for visual recognition. Retrieved October 21, 2009, from https://cs231n.github.io/convolutional-networks/ 44
Figure 14 – CNN Filter and Pooling Architecture for Natural Language Processing example. Taken from Convolutional Neural Networks for Sentence Classification 45
Figure 15 – 10-Fold Cross-validation schema 47
Figure 16 – ROC curve illustration 50
Figure 17 – Confusion Matrix for the SVM-300-FASTTEXT classifier 55
Figure 18 – Confusion Matrix for the Neural Network N-200(PRE)-GLOVE classifier 58

INDEX OF TABLES

Table 1 – Total quantity of Brazilian's movies tickets sold (2009-2016) 9
Table 2 – Data sources scrapped 24
Table 3 – Type of information and main format in each data source 25
Table 4 – Description of the dataset obtained from information on the website "The Numbers" 27
Table 5 – Intermediate dataset containing information of all movie's script 29
Table 6 – Final dataset before data preparation 29
Table 7 – Examples of problems found on the text of movie's script 30
Table 8 – Pre-trained GloVe word embeddings dataset 36
Table 9 – Pre-trained FastText word embeddings dataset 38
Table 10 – Gross Revenue discretization 39
Table 11 – Tickets Sold discretization 39
Table 12 – Evaluation metrics 47
Table 13 – Confusion Matrix 48
Table 14 – Project's Results x Current Process 52
Table 15 – Results for all models with GROSS REVENUE binary format target 54
Table 16 – Support vector machine configuration for the best model 54
Table 17 – Results for All Models with GROSS REVENUE (4-classes Format) TARGET ... 56
Table 18 – Results for All Models with GROSS REVENUE (4-classes Format) TARGET ... 56
Table 19 – Results for all models with TICKETS SOLD binary format target 57
Table 20 – Configuration of the Neural Network which achieved the best performance 57
Table 21 – Results for all models with TICKETS SOLD as a Multiclass Target 59
Table 22 – Configuration of the neural network that achieved the best performance. 59

INDEX OF GRAPHICS

Graphic 1 – Global Box Office Revenue 2005 to 2017 (in billion U.S. dollars) 17
Graphic 2 – Total number of movies released per year 18
Graphic 3 – Total Amount of tickets sold evolution in U.S. market 18
Graphic 4 – Market share distribution per studio in U.S. Market 19
Graphic 5 –Box office revenue evolution on Brazilian market from 2005 to 2017 19
Graphic 6 – Nº of movies released on Brazilian market (2005-2017) 20
Graphic 7 – Nº of tickets sold evolution in Brazilian Market 20
Graphic 8 – Brazilian movies released market share distribution 21
Graphic 9 – FSA annual Budget Evolution 22
Graphic 10 – Annual Nº of projects submitted for analysis of FSA evolution..... 22

1. INTRODUCTION

The global film market has reached a US$41,1 billion of revenue in 2018, representing a significant economic segment around the globe.[1]. Nevertheless, companies playing on this market suffer with the lack of capacity to predict which movies will have good box-office performance, and this remains a great challenge to this segment.

1.1. PROBLEM IDENTIFICATION AND CONTEXTUALIZATION

Movies, in general, are products that have a long development stage until they reach final consumers and normally at a high cost level. We can describe a movie development process, in a broader way, as being composed of 4 (four) stages: Pre-production, production, post-production and distribution.

Although decision-making about the larger portion of investment occurs still on the early stage of pre-production, on this phase decision makers only have access to a reduced set of information when compared to later stages such as production and post-production.

Furthermore, even nowadays, a great part of the investment decision in the production of movies is based essentially on the executive's previous experience, which usually resort to the work of some specialized professionals on movie script Reading and analysis, and to other criteria like the screenwriter's retrospect and production firm previous performance.

On this context, the percentage of movies that produces positive financial results is only 6%, and they account for around 80% of industry's profit in the last decade (Im & Nguyen, 2011).

In Brazil, this scenario is not different, data from Brazilian Film Agency (ANCINE), indicate that although the cinematographic policy may be considered successful on increasing the total number of national movie's tickets sold[2], as observed on table 1, only a few numbers of movies concentrate the great majority of tickets sold. Considering the period from 2009 to 2016, it was observed that the average of tickets sold increased, going from 191.374 in 2009 to 214.182 in 2016. Despite of that, if we look at the movies' individual performance, it is possible to verify that, on the same period, only 8.15% of the movies have accounted for 88.74% of the total quantity of tickets sold.

[1] Data from Statista's portal(Https://statista.com) on 19/08/2019

[2] Data from statistic directory 2016(ANCINE – Brazil. Available at https://oca.ancine.gov.br/cinema

Year	Tickets Sold (Total)
2009	16.075.409
2010	25.687.438
2011	17.687.772
2012	15.654.862
2013	27.789.804
2014	19.060.705
2015	22.500.245
2016	30.413.419

Table 1 – Total quantity of Brazilian's movies tickets sold (2009-2016)

From this perspective, several predictive models have been proposed to the complex task of predicting film's financial results, total number of tickets sold or even more abstract concepts such as success. However, most of these models have used variables that are available only after the decision-making time about investment, data regarding for example comments on social networks and other online platforms, cast members, production's sites.

Another factor of relevant importance in this project is the definition of success of a movie, some previous works have put its focus on box-office's gross revenue (Apala et al., 2013; Asur & Huberman,2010; Gopinath, Chintagunta & Venkataraman,2013; Mestyán, Yasseri, & Kertész,2013; Parimi & Caragea, 2013), while other on total number of tickets sold (Baimbridge, 1997; Meiseberg & Ehrmann, 2013). Although these two metrics ignore production cost, what certainly is a problem for direct investors that are expecting profit from their investment, in the context of this work this does not provide a relevant concern once the main objective here is to develop a classification model that is able to support on the decision-making process, helping on the promotion of the Brazilian culture and Brazilian's movies box-office performance improvement.

Specifically in what concerns to previous works that have approached the same task of box-office performance prediction, it is possible to enumerate a set of researches that have employed variables that are made available only closer to the movie's release (Eliashberg, Jonker, Sawhney e Berend, 2000) and after the official release, more specifically on the distribution stage (Boccardelli, Brunetta e Vicentini, 2008; Meiseberg & Ehrmann, 2013; Zhang & Skiena, 2009), having obtained more accurate results due to the larger volume of data available. However, this approach demonstrated to be very limited in practical terms, once these models are only able to predict a movie's box-office performance after the investment decision has already been done.

Therefore, based on previous works (Eliashberg, Hui, & Zhang, 2007; Lash, Fu, Wang & Zhao, 2015; Wang, 2017; Vecchio et.al, 2018), this project proposes to create a classification model that is able to predict the box-office performance of a movie that uses as predictors only features that are available before the investment decision, more specifically features extracted from the text of the movie's script, trying to improve the reliability of an investment decision. But the main difference proposed on this work is that, instead of engineering handy-crafted features

and then extract them from the body of a movie script, it will be tried to extract features using word embedding models from the text of the movie script.

1.2. CONTRIBUTIONS

The main contributions of this work are:

- Demonstrate how unstructured data sources, such as movie's script text, extracted from publicly available sources, after employment of adequate techniques of text mining, natural language processing and deep neural networks, can create good predictive models.
- Show how natural language processing techniques can reach state-of-art results in text classification field;
- Enable public managers to support their decision-making process with the aid of artificial intelligence algorithms.

1.3. STUDY'S OBJECTIVES

The main objectives of this work are:

- Develop a classification model that can predict a movie's box-office performance helping ANCINE on the decision-making process, regarding to the selection process of projects that apply for public funding;
- Promote the increase on Brazilian's films box-office performance;
- Reduce time spent on project's analysis, by automatizing a part of the process;
- Reduce costs with the hiring of specialized professionals that are needed to analyze movie's script;
- Improve ANCINE's selection process transparency by making available publicly and automatically the results of the evaluation of each movie script submitted to the classification of the model;
- Serve as a paradigm to other initiatives at ANCINE and other Brazilian public organizations, expanding machine learning's utilization to deliver more agile and efficient services to citizens.

1.4. RELEVANCE AND IMPORTANCE OF THE STUDY

Currently, one of the strategic objectives of Brazilian Film Agency (ANCINE), specifically in what concerns to the society component, is: "Being a knowledge center and main inductor of balanced development in the audiovisual sector, expanding the access, diversity and valuing

Brazilian content", accordingly to the strategic plan for the period of 2016-2020[3]. Despite of that, by observing information available on the "Cinema and Audiovisual Observatory (OCA)" it is possible to verify that, besides having reached significant increase on the volume of public resources invested in the Brazilian cinema market, the results are still highly dependent on a few very successful movies.

Thus, this project aims to create a facilitator mechanism for the process of investment decisions in the scope of the agency, with the objective of improving the performance of national films, helping ANCINE to accomplish its strategic goals.

It is important to highlight that ANCINE has faced a significant expansion of its competencies roles, incorporating assignments from other audiovisual segments such as conditional access services (SEAC), video on demand regulation (VOD) and electronic games, hence results from this project can be extended to a broader range of processes, with great potential to increase operational capacity of the agency.

[3]ANCINE's Strategic Map, available at https://www.ancine.gov.br/sites/default/files/MapaEstrategicoPortal.pdf

2. LITERATURE REVIEW

Businesses are being disrupted as never seen before, with the advent of the so called 4th revolution, that is driven mostly by exponential technologies, like machine learning (ML), Blockchain, Big data, Cloud, In-memory computing and Internet of Things (IoT), new business models rise every day and organizations struggle to keep their pace while incorporating these technologies into their business processes. The exponential increase in computer power and decreasing cost in storage, which obeys a geometric relation commonly known as Moore's Law (Penprase, 2018), is one of the main enablers of this digital transformation.

Data has become a central asset to most of the organizations enabled by the continuously increasing capacity of processing, network bandwidth and storage. Businesses processes are being reengineered to better use this huge amount of data to generate value in completely new ways.

In the audiovisual industry is no different, motion picture studios have highly incorporated new technologies to their businesses processes and are fostering innovation to keep alive and being able to compete against new business models that are moving forward over their market, like video on demand companies, mostly widely represented by the giant Netflix.

More precisely, as a high-risk investment industry, their success is directly connected to higher tickets volume being sold, with direct impact on distributor's profit margins as a key performance indicator (KPI) of movie's commercial success (Wallace, Seigerman, & Holbrook, 1993). Additionally, it has been verified that 40% of total box office profit comes from the premiere week (Einav, 2007).

Another important challenge to this industry is the variation on movie's popularity, a market on which a single movie can mean the difference between millions of dollars profit or a huge loss to the production studio (Simonoff & Sparrow, 2000). To overcome these challenges, many studies have been developed aiming to forecast movies success. More specifically, we can bring the diversity of works on the field of machine learning.

Machine Learning (ML) is a field of artificial intelligence concerned with the development of algorithms that learn directly from data, mostly in the sense of recognizing patterns and extracting information, but not limited to, being able to learn how to autonomously execute complex tasks too.

ML is usually divided into supervised learning, unsupervised learning and reinforcement learning. Supervised learning is the process to learn the implicit relations between independent variables and a dependent variable, the one we want to predict the value. The learning process is implemented using previous labeled examples, through which the machine can learn what is required in the context of a problem directly from the data.

As a field of research that has attracted much attention, movie's success forecast studies have been tried with different strategies, with a very diversified range of variables and different classification models. One of the pioneers on this research field was (Litman, 1983), he developed a multiple regression model that attempted to predict the financial success of films. The variables used in his work were movie genre (science fiction, drama, action, adventure,

comedy, and musical), Motion Picture Association of America rating (G, PG, R and X), superstar in the cast, production costs, release company (major or independent), Academy Awards (nominations and winning in a major category), and release date (Christmas, Memorial Day, summer). On this initial work Litman achieved a poor result with a R2 of 48,5%.

Lately (Sharda & Delen, 2006) proposed a new approach, they converted the forecasting problem into a classification problem, hence rather than forecasting the point estimate of box-office receipts, they classified a movie based on its box-office receipts in one of nine categories, ranging from 'flop' to 'blockbuster'. This study bring yet another important contribution that is, the segmentation of the previous studies into 2 (two) different major groups of approach : "(i) econometric/quantitative models-those that explore factors that influence the box-office receipts of newly released movies (Litman, 1983; Litman & Kohl, 1989; Litman & Ahn, 1998; Neelamegham & Chintagunta, 1999; Ravid, 1999; Elberse & Eliashberg, 2002; Sochay, 1994) and (ii) behavioral models - those that primarily focus on the individual's decision-making process with respect to selecting a specific movie from a vast array of entertainment alternatives (De Silva, 1998; Eliashberg & Sawhney, 1994; Eliashberg et al., 2000; Sawhney & Eliashberg, 1996; Zufryden, 1996; Sharda, 2006).

Besides these more traditional approaches, other studies have been developed, but most of them continued to focus on variables available only on later stages of the movie development process, more specifically after the film was already produced or immediately after its release. Although these models can be beneficial to distributors, delivering value and reducing their risks, they are not as helpful to filmmakers, producers or actors that want to reduce their risk of investment.

As an alternative to these approaches, more recently, many studies have been trying to use movie scripts as a feature source. The text of a movie script is a rich source of information that is intrinsically connected to the movies, representing the structure of the narrative of a film and the way the storytelling is developed. Therefore, this type of information may provide a more realistic approach to the task of movie's success prediction, enabling a better-informed decision-making process which reduces the risk of investment.

It's important to notice that this strategy converts the problem into a text classification issue, where it's possible to build handy-crafted features from the movie script's text and use them as input variables to different classification models or even the use of more modern techniques that automatically learn features directly from data. Text classification is a classic field from natural language processing, in which one needs to assign predefined categories to free-text documents.

As stated before, this strategy seems to be more aligned with the need of early prediction of movie performance, since when deciding which scripts to turn into movies, a process known as "green-lighting", movie studios and film makers need to assess the box performance of a movie based only on its script and allocated production budget, once most post-production drivers of box office performance e.g., actor, actress, director, MPAA rating are unknown at the point of green-lighting when financial commitments have to be made (Dabhade, 2015).

The importance of information encoded in movie script text find evidence on screenwriting theory as it is possible to identify interesting patterns and structures that acts as constructs to movie script and how they can be linked to movie's success. (Nelmes, 2007) states that movie scripts have a highly defined and tightly controlled nature, with generally three acts and around 100 to 120 pages, and the key events happening at the movie. He goes further explaining that these elements can be learnt as part of the craft.

In (James, 2016) a comprehensive literature review is made about the narrative structure theories and how popular movies has used this along the last decades. In his study he considered factors like shot durations, shot transitions, narrative shifts from one scene to the next, among others. In its conclusion he establishes that these theories have much similarities and, accordingly, that a large-scale formula has been used along the last 70 years on popular movie storytelling.

Pioneering the use of movie's script as source of information, (Eliashberg et al, 2006) extracted textual information from movie's script using domain knowledge from screenwriting and the bag-of-word model developed in Natural Language Processing. After that, they calibrated these types of textual information and then used it to predict the return-on-investment of a movie using Bag-CART (Bootstrap Aggregated Classification and Regression Tree) methodology developed in statistics (Breiman, 1996; Breiman et al., 1984). But this study faced a huge limitation, once not being apple to apply the text information extraction techniques to movie script directly, then they used spoiler's text as source, what may have impacted their results. Another important limitation was the need for human experts to extract a group of features to the model.

Eliashberg, Hui, & Zhang, 2010, evolved their previous work and developed a methodology to extract three textual features from scripts (genre/content, bag-of-words, and semantics) to predict revenue of a movie at the point of green-lighting, when only its script and estimated production budget were available, but now based on real movie scripts and not spoiler. Another improvement was the use of BART-QL model in place of BAG-CART that was used on their previous work which outperformed all benchmark comparisons with other methods.

Using only text data too, (Hunter, Smith, & Singh,2016) developed a research considering that the most important pre-production factors are the ones derived from script's text, and more specifically on their work those determined through the application of network text analysis—a method for rendering a text as a map or network of interconnected concepts. As expected, they found that the size of the main component of a screenplay's text network strongly predicts the completed film's opening weekend box office.

Latent semantic Analysis(LSA) a method for extracting and representing the contextual-usage meaning of words by statistical computations applied to a large corpus of text (Landauer, & Dumais, 1997) and term frequency-inverse document frequency, (TF-IDF), was used over a movie script dataset to develop a predictive model of movie success converting ROI into a binary dependent variable called "Success"(O'Driscoll, 2016).

Vecchio et.al, 2018 developed a model, strongly based on movie storytelling theories, that extracted emotional arcs from the text of movie scripts, and then used this information to predict overall success parameters of the movies including box office revenues, viewer satisfaction levels (captured by IMDb ratings), awards, as well as the number of users reviews and critic's reviews. Their research found that movies are dominated by basically 6 (six) arcs and that "man in a hole" shape are associated with the highest box office.

All these studies using the movie script text as input have demonstrated the high potential and the amount of information that is encoded in this source of data. But they approached the problem using mostly handly-crafted features or frequency-based metrics to extract information from text.

These techniques have clear drawbacks, they struggle or even ignore the semantics encoded in text, lose the order of the words and, in the case of handy crafted features, require an expert knowledge about the business domain. Another important issue related to frequency-based techniques like BAG-OF-WORDS is the high dimensional and sparse representations that derives from its implementation, which implies a hard work on dimensionality reduction to prevent model overfitting and the "curse of dimensionality" (data sparsity increases exponentially with the growth of dimensions).

Recent NLP researches have introduced a different set of techniques to text representation, mostly based on vectoral space representation method and due to their ability to recover semantic and syntactic information about natural language has become very popular (Major, Surkis, & Aphinyanaphongs 2017). Word embeddings models has reached state-of-art results as input in document classification tasks, sentiment analysis, sentence classification, text summarization, named entity recognition, among others classical NLP tasks.

Word embeddings share the basic idea that a word is well described by its context, they enable a more expressive and efficient representation by maintaining the word's context similarity and through the construction of low dimensional vectors (Naili, Chaibi, & Ghezala, 2017). Hence, words that have a similar meaning will have a similar representation.

Word2Vec (Mikolov et al. 2013), Global Vectors representation (Pennington, Socher, & Manning, 2014) and FastText (Joulin et al., 2017)) are the most popular methods for word representation and text classification task in recent researches. While Word2Vec and Glove works on the context of word level, FastText works on the character level enabling a more flexible approach on the representation of texts and helping to overcome challenges like representation of rare words once it is likely that a n-gram that composes a rare word has its own representation and then this word's representation can be built upon thin n-grams vector representations.

(Wang, 2017) on its master´s project proposed an approach that used word2vec models with both pre-trained dataset and domain-specific dataset to represent the text of approximately 3000 tv series and then used it combined with 3(three) other groups of input variables: Statistic and Network features, distributed representation variables and NLP based features. Although having achieved very good performance on Genre classification based only on the word representation models (F-score=87,21%), the work faces surprisingly decreasing performance after adding the others set of features to train new models, getting very poor results (F-score=48,50%). Despite of that, this work gives some insights on the usefulness of word embeddings as input to text classification models with movie script data and suggests some improvements that can be made on an effort to improve the results found.

Therefore, based partially on the approach used by (Wang, 2017)**,** on the present work, it is proposed then to develop and test a set of classification models, namely neural networks and support vector machines to predict movie box office performance based on two different metrics: Movie Box Office Gross Revenue and total quantity of tickets sold.

To the best of the knowledge of this author this work is the first to use word embeddings and deep learning networks to automatic extract features from movie script´s text as input to a predictive model of movie box office performance, specifically considering total quantity of tickets sold as the metric to assess the success of a movie.

3. METHODOLOGY

This work will follow the methodology CRISP-DM (Cross-Industry Standard Process for Data Mining) that has been established as an industry standard concerning the data mining segment. This methodology is comprised of 6 phases which represents the lifecycle of a data mining project. The reference-model for the CRISP-DM is illustrated on Figure 1.

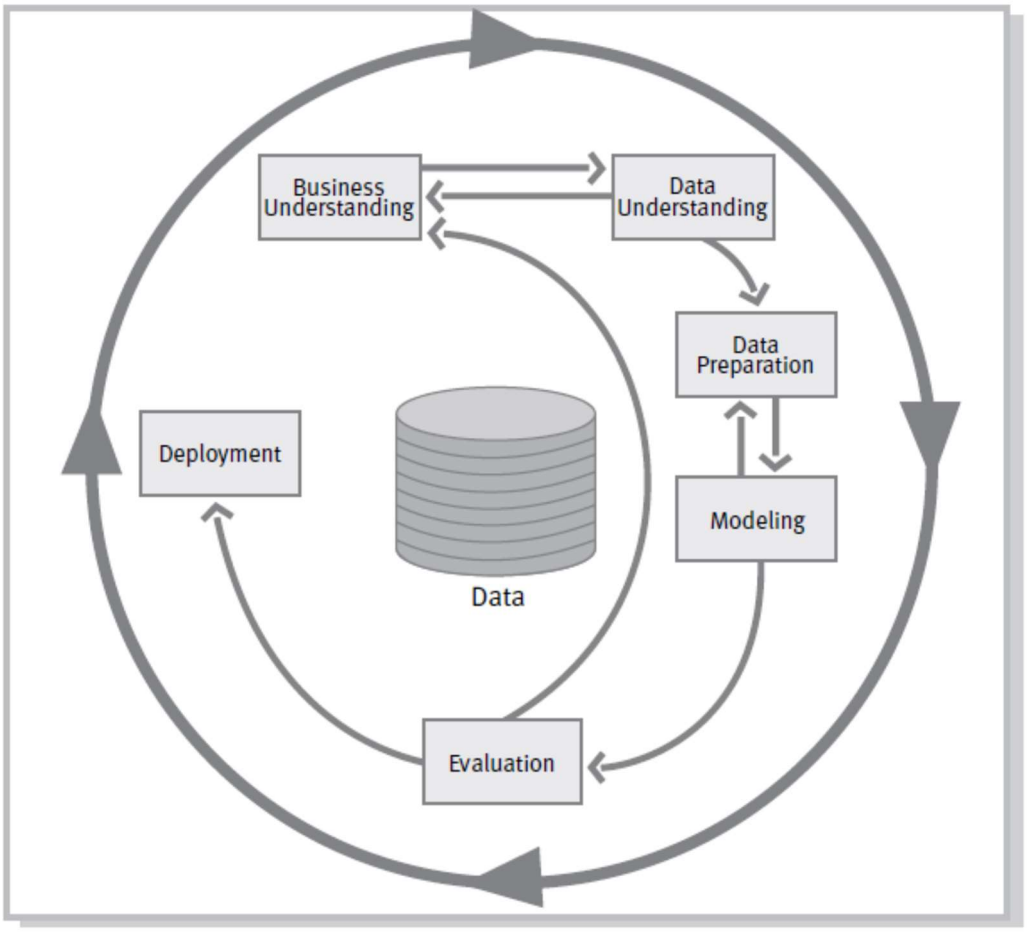


Figure 1 – CRISP-DM reference Model

From figure 1 we can identify the very dynamic nature of the data mining process, with interactions between the distinct phases and the outer circle that represents the cyclic nature of the process. It is important to notice that after a model is deployed the process isn't finished and requires periodic and criterions revisions to guarantee that conditions that originated the initial model are still in place and that the model continues to perform with the designed performance that was required by business.

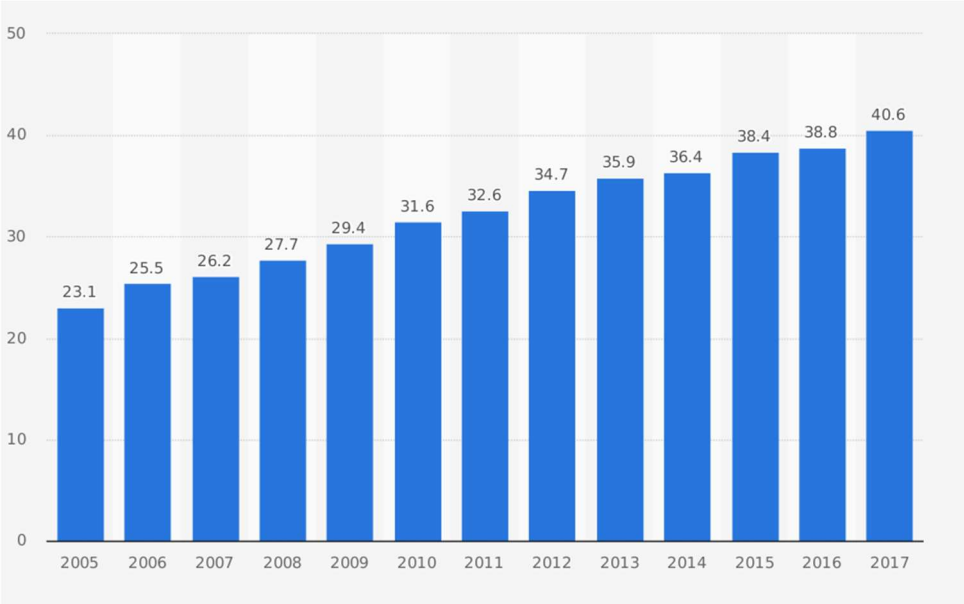
A very important characteristic of this methodology is the prescription of an initial phase that is concerned about understanding the business context, on this stage we have to identify project objectives and requirements from the business perspective and then transfer this knowledge into a data mining problem definition and develop an initial plan to achieve these goals.

Hence, on next section a market analysis of both U.S market and Brazilian Market will be done. Also, as the project will be held on the context of the Brazilian regulatory agency of audiovisual (ANCINE), a brief presentation of entity's structure will be held, this analysis aims to give perspective about the market and the business where the problem under study is inserted.

3.1. BUSINESS UNDERSTANDING

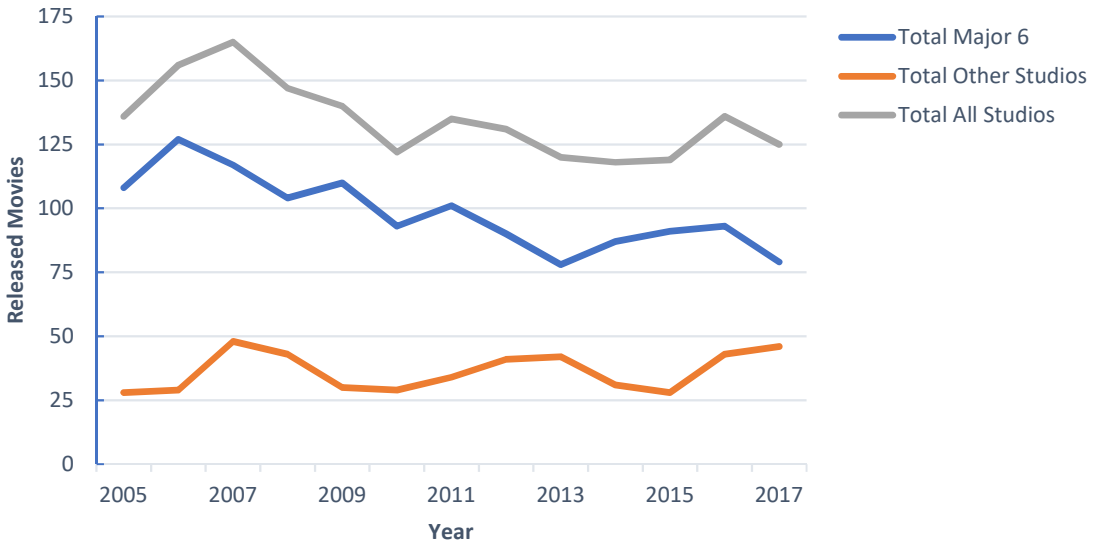
3.1.1. Market Understanding

The motion picture industry has faced a consistent growth along the last years, reaching a record revenue of US$40.6 billions of dollars on 2017 in the Global market. This behavior, opposing to a variety of other economic segments, has kept it pace even during the global economic recession of 2008 leading to a revenue growth of 75.75% since 2005.



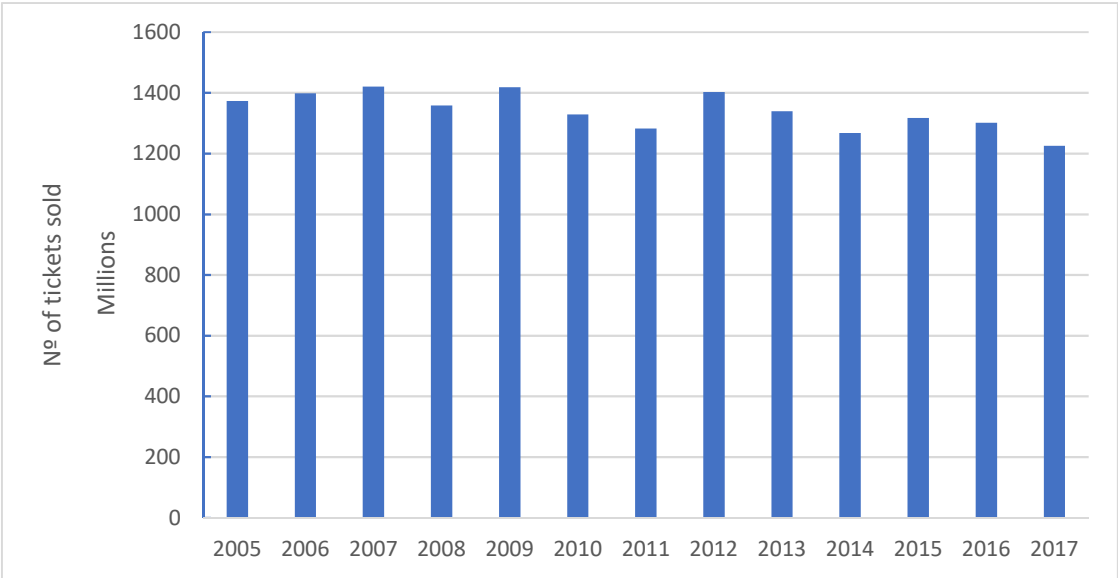
Graphic 1 – Global Box Office Revenue 2005 to 2017 (in billion U.S. dollars)

Although revenues are growing, the number of movies released per year has stayed very similar along the same period, decreasing around 8% when considering all the studios in the U.S market. Analyzing the 6 major studios line trend it is possible to verify an even bigger decrease reaching 26.9% as can be seen on Graphic 2.



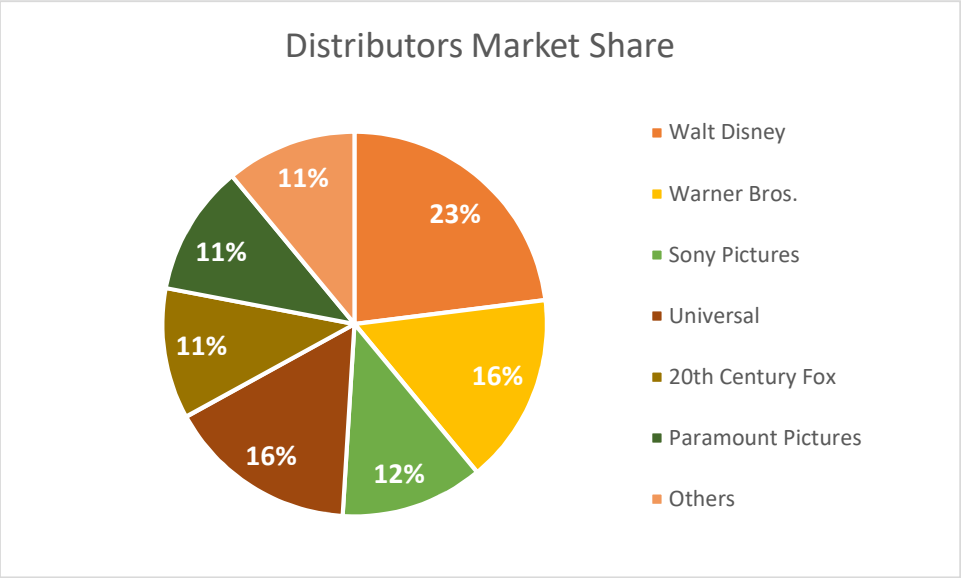
Graphic 2 – Total number of movies released per year

Also considering the U.S. Market, graphic 3 concerning to the number of spectators, it´s observed a decline of 10.8% on the period, in contrast with revenue growth too. The increase on the average ticket price on the period is the main contributor to revenue's growth and may indicate the reason for the decrease on the number of spectators on the period since the cost of this activity has raised by 40%.



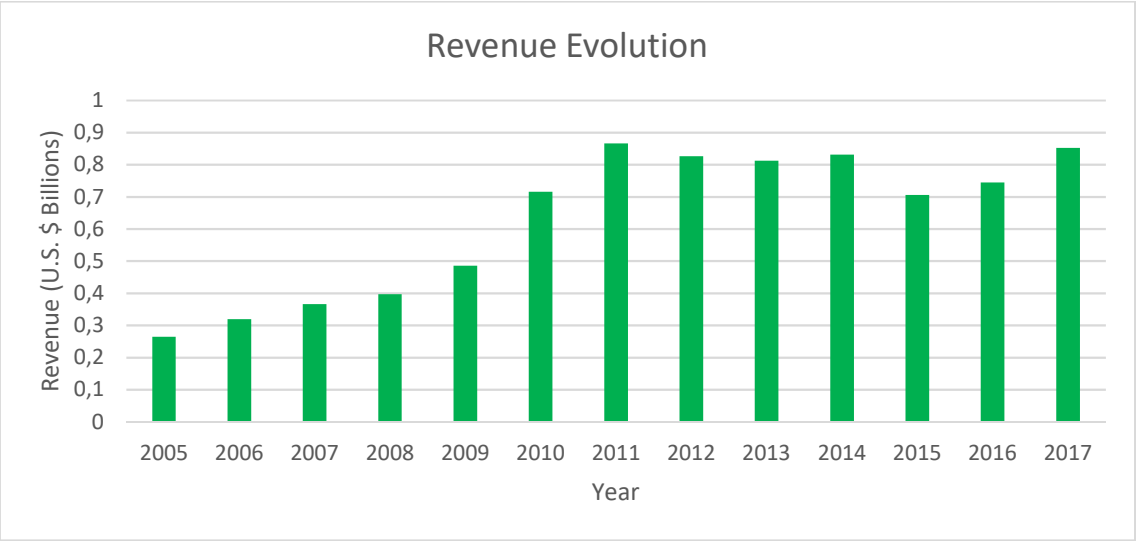
Graphic 3 – Total Amount of tickets sold evolution in U.S. market

In its turns, graphic 4 exhibits the market share distribution, where 6 majors distributors accounts for 78% of the market on means of Gross Revenue and all other studios together represents only 22%, numbers that demonstrate the high concentration of the market on the leaders.



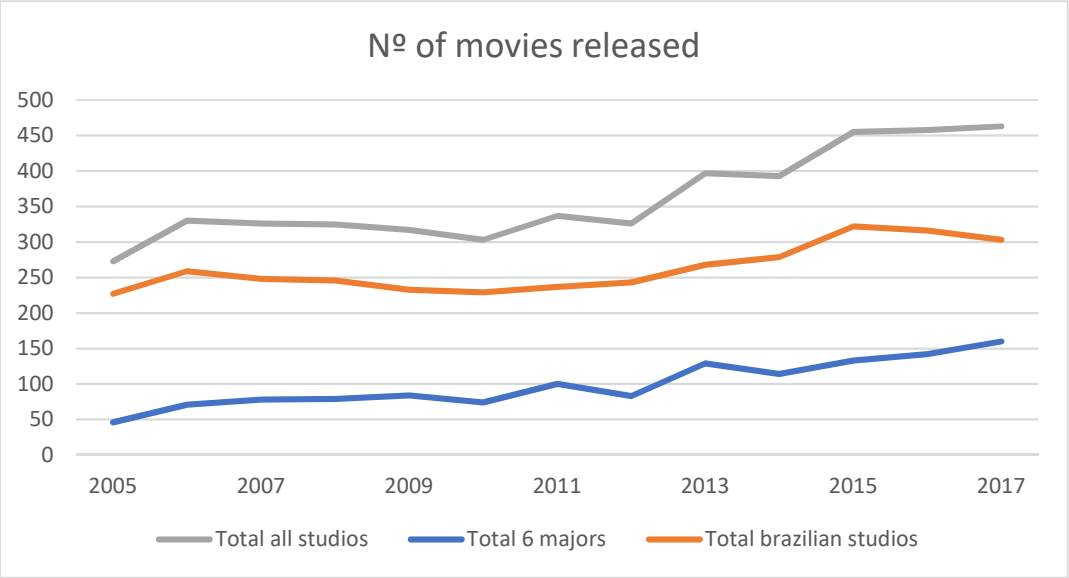
Graphic 4 – Market share distribution per studio in U.S. Market

Brazilian market, although much smaller than the U.S. market, has great influence of the later and presents some shared characteristics, as the growth on revenue and the 6 major companies in terms of market share. Graphic 5 represents the evolution of box office revenue on Brazilian market from 2005 to 2017 which has reached a 222% growth.



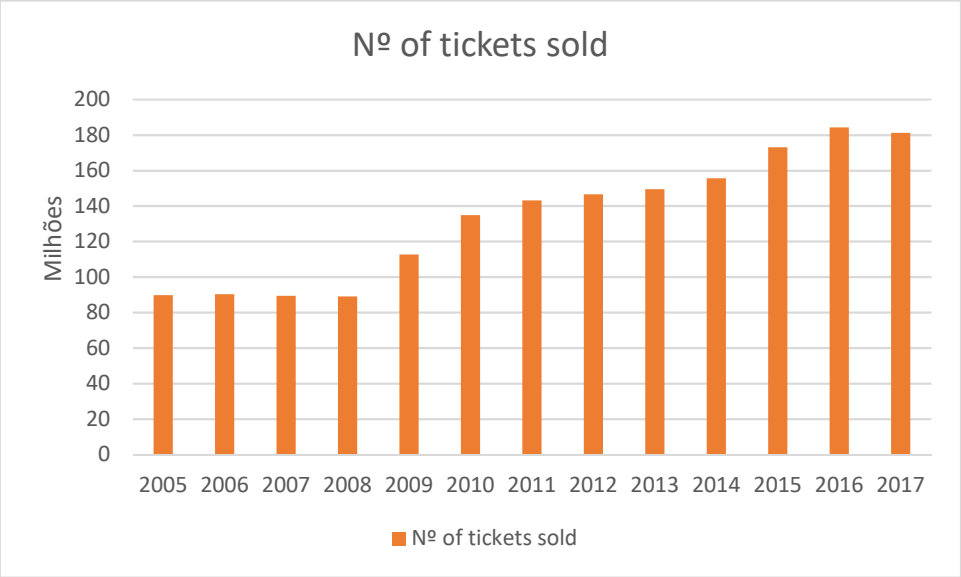
Graphic 5 –Box office revenue evolution on Brazilian market from 2005 to 2017

Opposing the behavior perceived by the U.S. market, on graphic 6 we can verify that the number of movies released on Brazilian market by the 6 major studios has increased by 247% while the movies launched by Brazilian companies has reached only 33,48% and the total number of releases has presented a 69% growth.



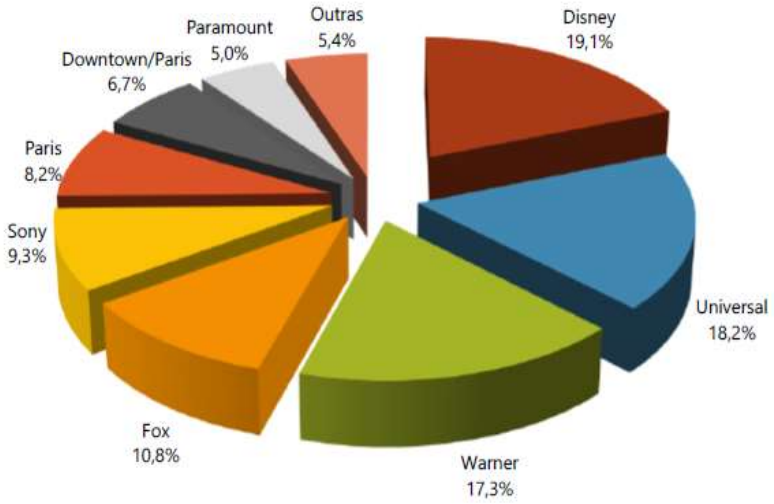
Graphic 6 – Nº of movies released on Brazilian market (2005-2017)

Additionally, on graphic 7 it is possible to identify that differently from the U.S. market the number of tickets sold in Brazil has perceived around 102% of increase.



Graphic 7 – Nº of tickets sold evolution in Brazilian Market

Despite of these differences on the evolution of tickets sales and movies released evolution, concerning to market share the composition is almost the same, graphic 8 presents a high concentration on the 6 major studios with 75.4% of market share and the main distributors in U.S are almost the same in Brazil.



Graphic 8 – Brazilian movies released market share distribution

3.1.2. Business Context

Once a better perspective of the market is available, it is possible to establish a brief presentation of the structure and the business context of ANCINE.

Brazilian film Agency (ANCINE) is a public regulatory agency which has as main attributions: public funding, regulation, monitoring and control of Brazilian cinema and audiovisual market. It is managed by a collegiate directorate that is composed of 1 president-director and 3 directors, all with fix mandate. Under this board are 5 oversights (SRE, SFI, SAM, SFO, SDE) and 3 secretaries (SGI, SEC, SEF) as shown on the organogram of figure 2.

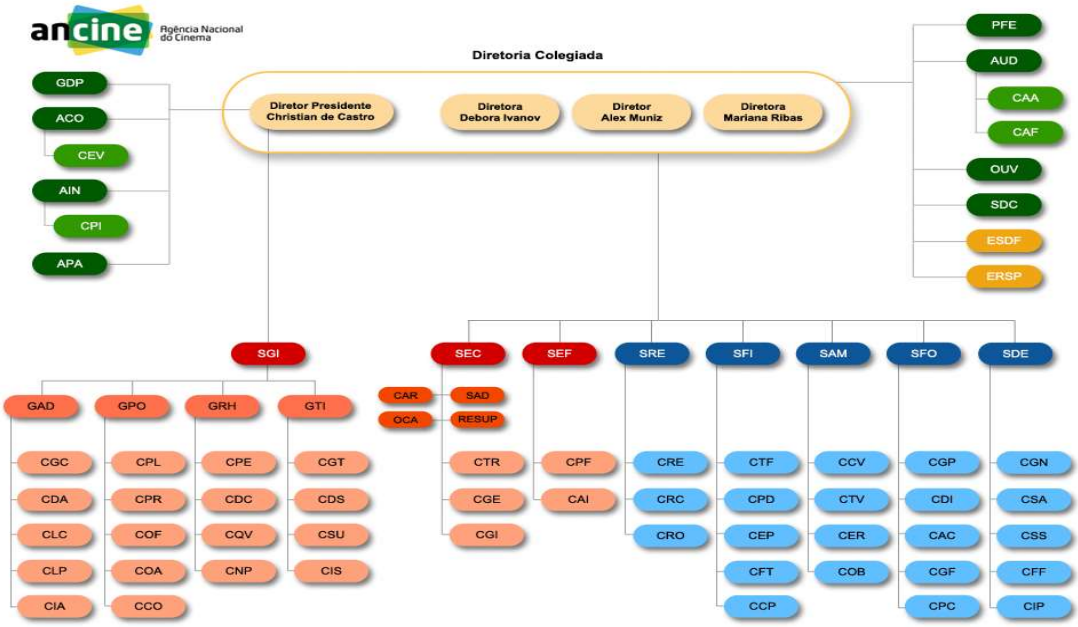
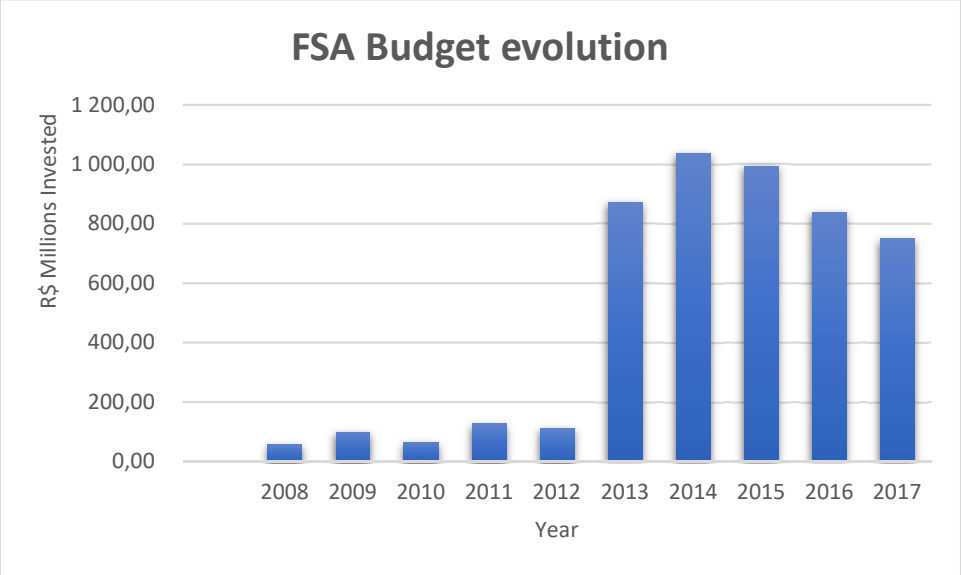


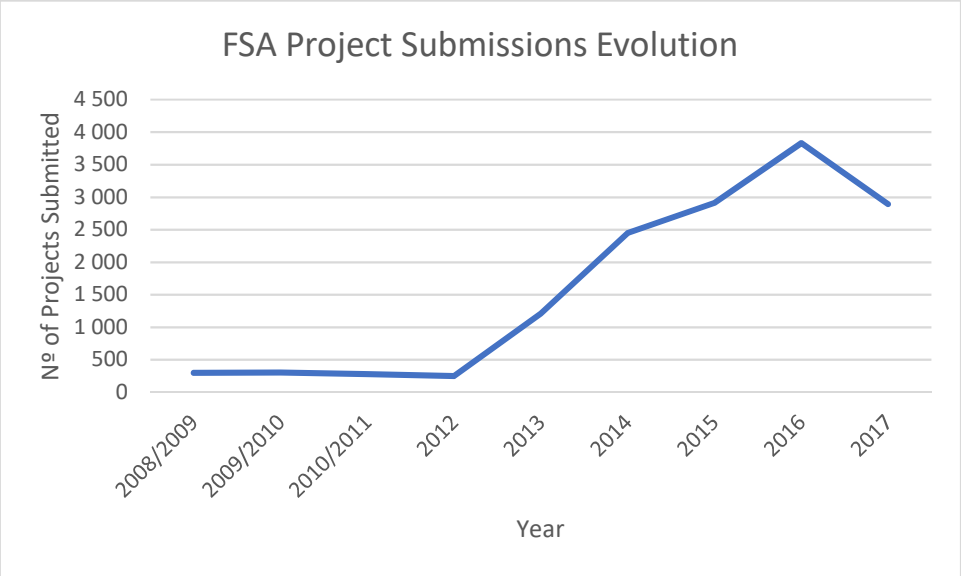
Figure 2 – ANCINE's organogram

More specifically, as part of its functions of public funding audiovisual market, ANCINE is responsible for managing **FSA** (Audiovisual Sectoral Fund). FSA is one of the main tools that Brazilian State makes use to help on the development of the Brazilian audiovisual market, having faced a significant growth of 1233% on its budget since 2008 as presented of **graphic 9.**



Graphic 9 – FSA annual Budget Evolution

Such increase has imposed a proportional growth on the demand, raising the number of projects submitted for analysis, from 299 in 2008 to 2892 projects in 2017, after a peak of 3833 projects in 2016 which represented an increment of 1182% (graphic 10).



Graphic 10 – Annual Nº of projects submitted for analysis of FSA evolution

Therefore, this scenario has led to huge challenges to the public policies designed, requiring a comprehensive restructuration on its workforce, work routines, system development and a diverse set of other aspects to enable an adequate response.

In this context, the selection process instituted at ANCINE has been under reformulation along this period. This transformation has served as inspiration to this master project given the very opportune business situation for process improvement.

Nowadays, one of the aspects that is analyzed on some of the projects that are submitted to ANCINE is the movie script of the project that aims to apply to public funding. For analyzing this documents that are fundamental to the production of a movie, a set of variables are considered, as established in the "External Analysts Manual"[4]. Basically, 2 (two) external analysts considering the aspects in the manual, make an evaluation of the project and gives a grade to it, then these grades are summed and results on the project's final grade. Based on this grade ANCINE selects the projects that are enabled to receive investment from FSA.

Beyond being a time consuming and human intensive working process, this analysis has represented a significant expense, once each project analyzed costs a base value of R$ 593,04 [5].

Although movie script's analysis is not the only aspect considered on the evaluation of the projects nor it's the more important, this part of the process represents a great opportunity for improvement with the help of machine learning techniques, not only reducing analysis's cost but mainly reducing the time required for evaluation of the proposals.

Hence, at this point it is possible to link more clearly the objectives of this project with the problem on the business context, which is: Improve ANCINE project's selection process with the introduction of automated movie script classification.

Another important aspect to be considered is how movie scripts will be classified and the relation of the target variable of the model with one of ANCINE's strategic business goal: market development. Thus, on this project, it is considered as indicator for market development, the number of spectators a movie achieves, and the global amount of gross revenue obtained by funded movies, so the greater the number of spectators of a movie more it contributes to market development and larger gross revenues represent improve in performance too.

Afterwards, several websites were identified to be used as data sources in the development of this project, website's names and types of information available at each of them are described on table 2. Detailed description of the data acquisition process and data exploration will be held on next section.

[4] Available at http://www.brde.com.br/wp-content/uploads/2018/04/Manual-de-Pareceristas-Externos.pdf

[5] Available at https://www.ancine.gov.br/sites/default/files/edital_credenciamento.pdf

Website name	Type of Information
The Numbers	U.S. Box Office data
IMSDb	Movie Script files
The Script Savant	Movie Script files
The Daily Script	Movie Script files
Simply Scripts	Movie Script files
Weekly Script	Movie Script files
Selling Your Screenplay.com	Movie Script files

Table 2 – Data sources scrapped

3.2. DATA UNDERSTANDING

Data understanding comprises data acquisition and data quality assessment through the implementation of some data extraction and integration techniques. Additionally, initial data exploration was executed to obtain insights by visualizing basic distribution statistics, variables types, volume of data acquired and description of variables meaning.

Before explaining how data was acquired, it is important to elucidate that the model of this project was designed to use text data from movie's script and U.S. box office data about the movies. Hence, for the movie's script, data was extracted from 6 different websites and for the Box office data was extracted from the site "The Numbers", as described previously on table 3.

A macroprocess that describes data understanding stage is shown on figure 3. Once the processes to extract data from these websites differed from each other, each of them will be described apart.

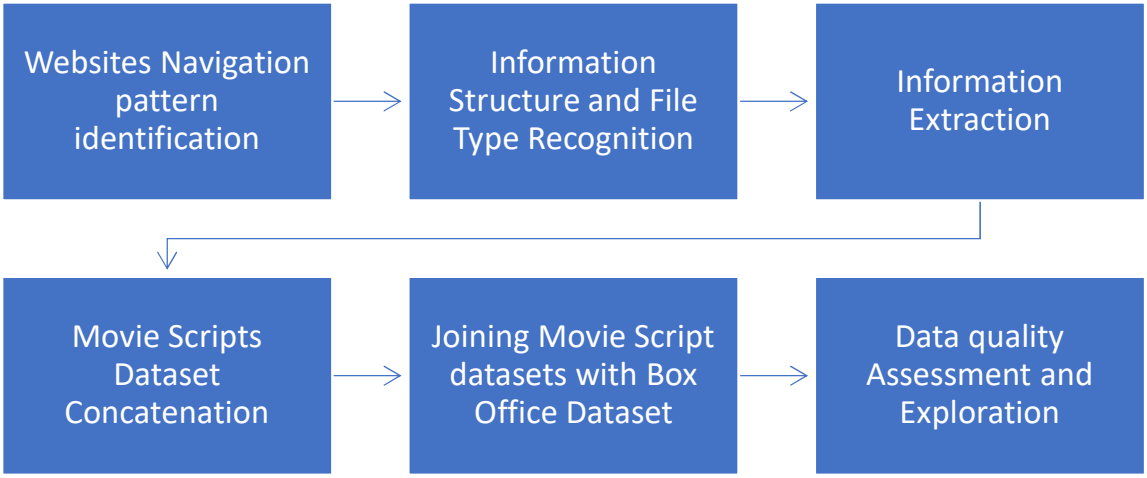


Figure 3 – Data Understanding Macroprocess

3.2.1. Data Understanding Macroprocess

This process started with the analysis of the structure of each website aiming to identify how the target content was distributed through the webpages, with this information was possible to recognize the URLS that were necessary to be accessed in order to retrieve the data for the project.

Afterwards, how information was presented in each website was verified and documented. On this step was identified that the movie´s scripts were in plain HTML, TXT and PDF format, in its turn box office data was organized in HTML tables divided per year.

Type of Information	Format
U.S. Box Office data	HTML tables separated per year
Movie Script data	Plain HTML, TXT and PDF

Table 3 – Type of information and main format in each data source

Hence, to extract data considering these distinct realities, web scrapping was employed to extract data from plain HTML and HTML tables, on the other hand TXT files were downloaded directly into folders with the support of python routines and PDF files were also downloaded into folders, but required further processing with the python library PyPDF2. Figure 4 illustrates data extraction macroprocess.

Figure 4 – Data Extraction Macroprocess

3.2.2. Data Extraction Techniques and Tools

Internet has been a great source of data about the movie industry, a lot of sites with data regarding consumers ratings for movies, movie's reviews, box office performance, casting and so on. Nevertheless, very often all this information is presented in unstructured format across multiples websites with no standard format.

Especially on the case of this project, no public database was found containing data about movie scripts and box office performance, considering the number of tickets sold as a metric. Therefore, it was necessary to construct this dataset using web scrapping from multiples websites.

3.2.2.1. Web scrapping

Web scrapping is basically the process of extracting data from a website. Generally, it is implemented using programs called “scrappers” that are developed with this specific purpose. Hence, to extract data, firstly is necessary to identify which HTML tag contains the target data, with this information we can develop a program that do a GET request to receive the HTML code of the webpage and after that parse the HTML structure of it to retrieve the designated data that was previously identified as needed. The parsing process is executed by decomposing the website content through the parse tree of figure 5.

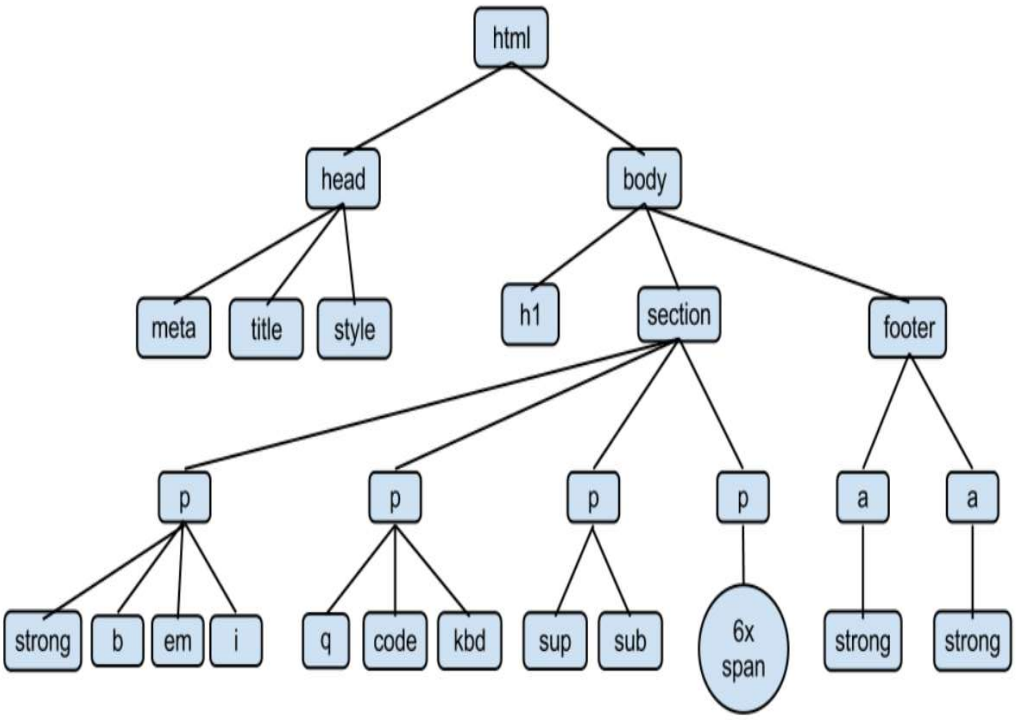


Figure 5 – HTML parse tree

Finally, the scrapper code needs to save retrieved data on a standardized output that can be a JSON, TXT, CSV, pandas data frame or any other format designated.

3.2.2.2. Beautiful Soup

Beautiful Soup is a python library that can be used to pull data from HTML and XML files, to extract data from HTML it creates a parse tree for parsed pages. Additionally, it has different options of parsers, what facilitates the process of scrapping different websites, reducing the development time. On this project beautiful Soup was used to develop the scrappers that retrieved data from the multiples target websites.

3.2.2.3. PyPDF2

PyPDF2 is a pure-Python PDF library that can be used for cropping, merging, splitting and transforming pages in PDF files. PyPDF2 website claims that it may also be used to add data, passwords and viewing options to the PDFs. Furthermore, it can retrieve text and metadata from PDF files. On this project, this tool was used to extract movie’s script text from pdf files.

3.2.3. Data Extraction "The Numbers"

"The Numbers" is a website dedicated to publishing movie's box office data on the US market. It has data since 1995 covering highly diverse aspects of movies theatrical market and home video statistics.

Although a very rich data source, the website publishes information about box office performance of the movies on HTML tables that are split by year in multiple pages. Hence, considering the requirement to get ticket sales and gross revenue data concerning to the movies launched on the U.S market since 1995, it was necessary to apply the following steps to retrieve data from this website:

1. Search and identify what pages host the target data
2. Locate on these pages tags containing required information
3. Retrieved target webpages
4. Parsed it with beautiful soup
5. Saved Each yearly table from HTML into a panda data frame
6. Concatenated all data frames from 1995 to 2017 into a CSV file

Steps 3 to 6 were automated using a web scrapper developed in python. Additionally, a verification of the resulting csv file with data from 1995 to 2017 was performed and one more step was required. Some movies were included in more than one year, in most of the cases the reason for that was that movies were launched very close to the end of the year and so they only leave theatrical market on the next year. Thus, to solve this issue, data regarding to movies on this situation was aggregated by year.

The final dataset concerning data extracted from the website "The Numbers" have 10.742 records and its characteristics are described on table 4.

Variables	Datatype	Description
Movie	String	Contain the movie's name
Release Date	Date	Date that the movie was released
Distributor	String	Company that distribute the movie
Genre	String	Movie's genre
MPAA	String	MPAA rating of the movie
Gross	Float	Total gross of the movie
Tickets Sold	Float	Total quantity of tickets sold for the movie

Table 4 – Description of the dataset obtained from information on the website "The Numbers"

3.2.4. Movie's script Data Extraction

Multiple websites were employed to obtain movie's script, the complete list of sites is on table 2. The process implemented to construct the dataset containing all information regarding to the scripts of movies consisted of 3 major steps:

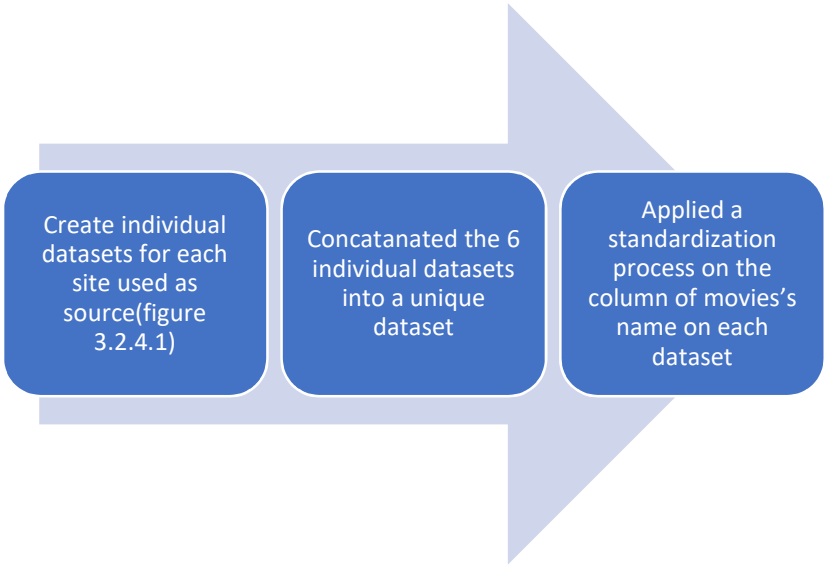


Figure 6 – Movie's script dataset creation process

Concerning each of these websites, movie's scripts were found in plain HTML, TXT and PDF format what required different approaches accordingly, thus it was created a separated dataset for each website following the steps described on figure 7. The main objective of using text from scripts of movies is to automatic extract features using word embedding techniques. These features will be the independent variables of the classification models of this project.

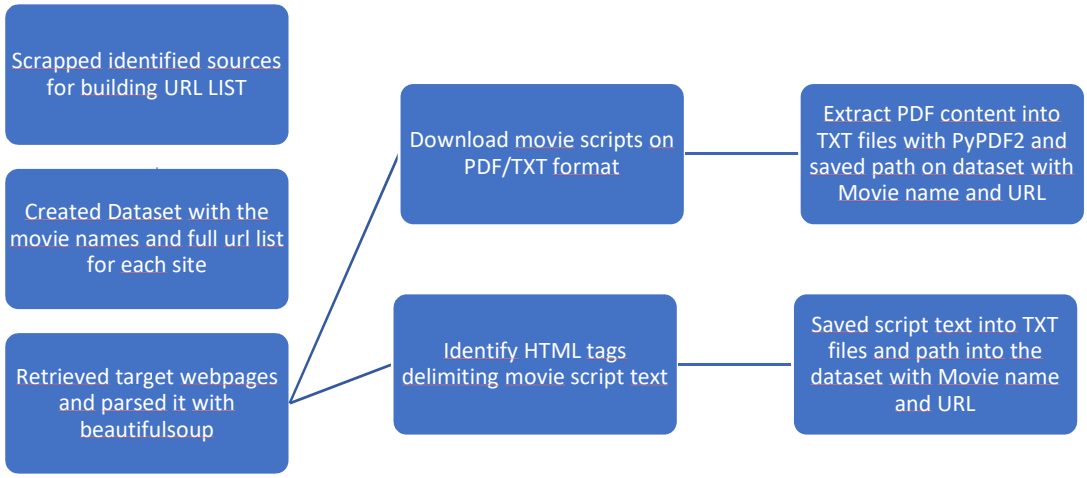


Figure 7 – Movie's script web scrapping process

After the creation of the individual datasets, considering that all of them were constructed sharing a common structure, the datasets were concatenated into a unique dataset and after that, the column representing the movie's name was standardized by applying some characters replacement and removal. Table 5 describes the properties of this unique dataset regarding information about movie's script.

Variables	Datatype	Description
Movie	String	Contain the movie's name
URL	String	Contain the hyperlink to the original file downloaded
Movie_path	String	Path to the TXT file containing the text of the movie script

Table 5 – Intermediate dataset containing information of all movie's script

3.2.5. Joining Movie Scripts dataset with Box Office Dataset

Once the 2 previous datasets were created, one containing information related to U.S. Box Office data and the other containing information regarding to Movie script, datasets were joined based on the movie's name column.

The resulting dataset had a lot of duplicated records, this is a clear result of the multiple source used for data acquisition, this issue was solved by dropping all duplicate entries based on movie's name and date of release, so every repeated record having the same name and same date of release were dropped, leaving on the dataset only the first occurrence. This joined dataset contains 1200 records and its structure is described on table 6.

Variables	Datatype	Description
Movie	String	Contains the movie's name
Release Date	Date	Date that the movie was released
Distributor	String	Company that distribute the movie
Genre	String	Movie's genre
MPAA	String	MPAA rating of the movie
Gross	Float	Total gross of the movie
Tickets Sold	Float	Total quantity of tickets sold for the movie
URL	String	Contains the hyperlink to the original file downloaded
Movie_path	String	Path to the TXT file containing the text of the movie script

Table 6 – Final dataset before data preparation

At this point of the project, data acquisition is considered concluded and it is possible to go forward to data exploration and quality assessment.

3.2.6. Data exploration and quality assessment

On this project, it was selected the strategy of using as input automatic extracted features from distributed representation of the movies script's text. Therefore, instead of more traditional data exploration techniques that focus on identifying data distribution, missing values and outlier's detection and treat them adequately on data preparation, it is applied only assessment of the quality of the content of movie scripts on an effort to find problems, such as wrong characters encoding, that may affect distributed representation techniques.

3.2.6.1. Movie scripts quality assessment

So that I could assess the movie's script, approximately 3% of the files of each source were read, which represented around 40 movie scripts. On this analysis were found some problems related to the movie's script extracted data and some unwanted texts like advertises were present is some of the movie's script.

Another problem found was that some movie scripts were too short, considering the premise that a page represents 1 minute of movie, files with less than 40 pages which violates the most basic definition of a feature film of the Academy of Motion Picture Arts and Sciences, which states that a feature movie must have at least 40 minutes, were removed from the dataset. Some examples of these unwanted text are shown on table 7.

No encoding problems were found. Missing value also was not a problem on the context of this project once the datasets were built and integrated specifically to the purposes of this study. Therefore, as stated before the problems that needs to be addressed on data preparation were those more related with the semantics of the words: synonyms, non-value adding terms (stop words), slangs and contracted forms of words.

Problem Found	Example
Advertise from html extracted movies	<!-- if (window!= top) top.location.href=location.href // -->
Websites markups	Script provided for educational purposes. More scripts can be found here: http://www.sellingyourscreenplay.com/library
Movie script too short	Some files containing the text of movie script were clearly incomplete, like
Contracted forms	Don't, IT'S

Table 7 – Examples of problems found on the text of movie's script

3.3. DATA PREPARATION

Data preparation is the most time-consuming stage in any machine learning project, accounting for 60% to 80% of the time spent. On this part of the project, a set of transformations, formatting and cleansing were performed aiming to take the data into the adequate format to serve as input to the models that would be developed.

The first part of data preparation consisted on cleaning and formatting the text from movie's script, a set of operations was performed with the support of Regular expressions and python's libraries, like NTLK, Inflect and Contractions.

Natural Language toolkit (NTLK) is a leading platform for building Python programs to work with human language data. It comes along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries.

Inflect library is used basically to convert numbers into text. Contractions library was used to expand the contracted form of words, for example the term "don't" is expanded to "do" and "not".

It is important to notice that text cleaning is a problem-specific task and in general it is not feasible to obtain a text that can be considered totally cleaned, once some tradeoffs between different approaches have to be considered in an effort to obtain the best possible results to the objective of the project.

Considering this scenario, the following tasks were performed for cleaning and formatting the text:

- removing punctuation
- removing stop words from the text
- splitting the text into tokens
- expanding contracted forms of words
- normalizing case

3.3.1. Splitting text into words (Tokenization)

Before cleaning the text, it is necessary to split the text into words, a process called tokenization. These words after proper cleaning will compose the vocabulary that is going to be used to represent the documents after the word embeddings representation is applied.

By performing this task, each document becomes a large list with all the words present on it. The content of each document was split by whitespaces.

3.3.2. Expanding contracted forms of words

Contracted forms of words are challenging on NLP tasks and can reduce the performance of the models once it is hard to extract semantic relations with this type of expression. Looking forward enhancing the text representation, the python's library "contractions" was employed to expand terms like don't, can´t and I´ve, for example.

3.3.3. Removing punctuation and Normalizing Case

When converting the text to vectoral representations like the ones provided by word embedding it is necessary to avoid that we have multiple representations for the same on the same context, hence the words “Hero”, “HERO” and “hero” that in fact have the same meaning could be represented differently if we don’t apply case normalization. The same problem applies to punctuation in the middle of words, something that can alter the word representation.

Therefore, to get better word representations, and consequently better document representations, all words were converted to lowercase, and punctuation was removed.

3.3.4. Removing stop words

Stop words are those words considered not to contribute to the meaning of the phrase. They are very common words such as: “the”, “an”, and “is”. On tasks like document classification, an important step is to remove stop words. Python’s library, NLTK, provides a list stop words for a variety of languages, such as English.

After text is cleansed and only the terms that are considered relevant to the analysis are available, the final corpus of the project is ready to a last and more complex transformation, which is embedding the words and, on the sequence, the overall documents.

3.3.5. Removing General data quality problems

Regarding problems like the ones described on table 7, all of them were individually treated with the employment of python routines, and hence texts that were not related with the movie script but were present due to the extraction process were removed from the original downloaded data.

3.3.6. Word Embeddings

“You shall know a word by the company it keeps!”, this sentence from John Firth in the book Studies in Linguistic Analysis, represents the central idea behind word embedding techniques.

Word embedding is a distributed representation of text in which words that have the same meaning have similar representations. They are a set of techniques where individual words are represented as real-valued vectors in a predefined vector space. Many authors cite the employment of these techniques as a breakthrough moment to the NLP field, given the great improvement on the performance of NLP tasks enabled by them.

Once words and documents are represented using dense and low-dimensional vectors, word embeddings offer a great computational advantage, given that most of neural network toolkits do not play well with very high-dimensional, sparse vectors (Goldberg, 2017). Another benefit of dense representations is generalization power, hence established the hypothesis that some features may provide similar clues, it is advantageous to provide a representation that is able to capture these similarities (Goldberg, 2017).

The distributed representation is learned from the word’s utilization, and because of that words that are used in similar forms have similar representations. In contrast with approaches like one-

hot-encoding where a term is represented by a high-dimensional vector, often with thousands or millions of dimensions, word embeddings provides a much more dense and low dimensional representation, usually with some hundreds of dimensions or even less.

Embeddings were trained on the Corpus of the project and pre-trained embeddings were applied. These pre-trained embeddings are made available by others professional and researchers that have trained their models on huge datasets with billions of tokens like the Wikipedia dataset and have shown to perform as well on embedding text from different domains that don´t have high level of language specialization.

Therefore, to train the models the text of the movie's scripts was embedded by the application of 3 different techniques: Word2Vec, Glove and FastText embeddings.

3.3.6.1. Word2Vec

Word2Vec is a tool that implements two novel model architectures for computing continuous vector representations of words from very large data sets. These architectures were proposed as an alternative to previous NLP systems and techniques that treat words as atomic units, in which it is not considered the similarity between words, once they are represented as indices in a vocabulary (Miklov, Chen, & Corrado, 2013).

Essentially, it uses a training text data to constructs a vocabulary and then learns vector representation of words. In order to assess the quality of the representations learned, the architectures ground on the idea that similarity of word representations goes beyond simple syntactic regularities, and then they used a word offset technique where simple algebraic operations were applied to the word vectors, and demonstrated for example that vector("King") - vector("Man") + vector("Woman") results in a vector that is closest to the vector representation of the word Queen (Mikolov, Yih, & Zweig, 2013).

Aiming to maximize accuracy of these vector operations 2 new model architectures were proposed for learning the word embeddings, these architectures preserve the linear regularities among words, the proposed models were:

- Continuous Bag-of-Words, or CBOW model.
- Continuous Skip-Gram Model.

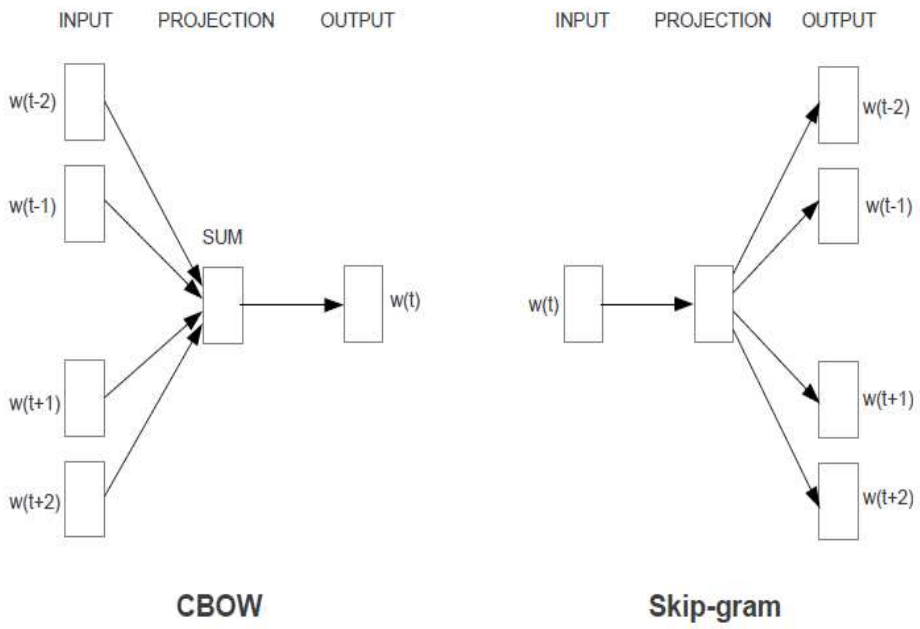


Figure 8 – Word2Vec architectures - CBOW and skip-gram schemas

Both models use a sliding window to limit the context from which words representation is learned, the neighborhood of a given word is then defined by the size of the window, which is a configurable parameter of the model. This abstraction enables the models to learn considering the local context where the word occurs.

Additionally, the size of the sliding window has a strong effect on vector similarities calculated, where large windows usually produces more topical similarities, and on the other hand smaller windows normally produces more syntactic and functional similarities (Goldberg, 2017).

Continuous Bag-of-Words Model (CBOW) tries to predict the current target word (the center word) based on the source context words (surrounding/neighbor words). Taking as an example the sentence, "*The queen of England is very powerful*", CBOW model consider all pairs of the type (context window, target word), where we have:

- *Size of context window = 2,*
- *Sample pairs = {([The, of], queen), ([is, powerful], very), ([of, is], England)}*

Furthermore, the weight matrix between the input and the projection layer is shared for all word as shown on figure 8**.**

In its turn, Continuous Skip-gram Model use a reversal strategy when compared to CBOW. Thus, it tries to predict the source context words (neighboring/surrounding words) given a target word (the center word). Again, for illustrating the logic behind the model, it will be used the same example sentence, and hence the given scenario is:

- *Size of context window* = 2,
- *Sample pairs = {([The, of], queen), ([is, powerful], very), ([of, is], England)}*

But here, the skip-gram model inverts the roles of the items in the pairs, hence it tries to predict each context word from its target word. Simplifying, given the target word “queen” it tries to predict its context window [the, of].

On this project, the Keras implementation of Word2Vec with the CBOW model was used. The embeddings for the words on the text of movie’s script were obtained with the support of the pre-trained dataset “GoogleNews-vectors-negative300” 6.
Additionally, 100/200/300-dimensional Embedding were obtained by training Word2Vec on the Corpus of this project.

3.3.6.2. Global Vector for Word Representation (GloVe)

GloVe is a new is a new global log-bilinear regression model that combines the advantages of the global matrix factorization and local context window methods (Pennington, Socher, & Manning, 2014). Their authors claim that by training only on the elements that are different from zero in a word-word co-occurrence matrix, rather than on the entire sparse matrix or on individual context windows in a large corpus, the model can improve statistical information obtained.

This model is an unsupervised learning approach, and uses weighted least squares that, by training on global word-word co-occurrence counts, makes efficient use of statistics. Glove produces a word vector space with meaningful substructure and has reached state-of-the-art performance on the word analogy dataset (Pennington, Socher, & Manning, 2014).

Essentially, the GloVe model creates firstly a huge word-context co-occurrence matrix consisting of (word, context) pairs, in a way that each element in this matrix represents the frequency of occurrence of a word within the context, notice that the context can be a sequence of words).

This word-context matrix WC is very similar to the term-document matrix popularly used in text analysis for various tasks. Matrix WC is factorized then and its representation is a product of two matrices, the **Feature-Context (FC)** and the **Word-Feature (WF)** matrix, as the formula that follows:

WC = WF x FC

Additionally, the following steps are applied:

1. WF and FC are initialized with some random weights
2. WF and FC are multiplied and a WC’ is obtained, where WC’ is approximation of WC
3. Distance between WC and WC’ is measured
4. Iterate through steps 1 to 3 trying to minimize the error using Stochastic Gradient Descent (SGD)

After some stop criteria is reached, a WF matrix with the word embeddings for each word is obtained, where the number of dimensions of F is configurable. Figure 9 illustrates some

[6] Pre-trained dataset available at https://drive.google.com/file/d/0B7XkCwpI5KDYNlNUTTlSS21pQmM/edit

examples of embeddings produced with the GloVe Model and how it is efficient on capturing semantic similarity.

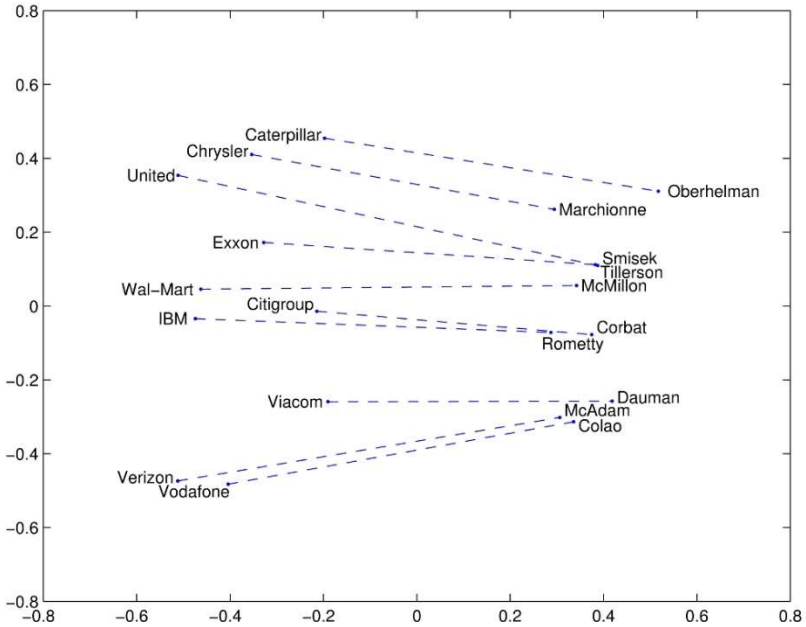


Figure 9 – Glove's embeddings examples

Considering that GloVe is one of the most used feature-extraction techniques used in the NLP field, as on the case of Word2Vec a pre-trained dataset was used on this project, and its characteristics are:

	GloVe6b
Origin of Tokens	WikiWikipedia 2014 + Gigaword 5
Number of tokens	6 billion
Size of vocabulary	400.000 tokens
Casetype	Uncased
Embedding vectors Sizes	50,100,200 and 300 dimensions

Table 8 – Pre-trained GloVe word embeddings dataset

The embeddings for the words on the text of the movie's script of this work will be obtained using the pre-trained dataset and the process to develop the embeddings for the document level (for an entire movie script) is described further on this chapter.

3.3.6.3. FastTExt

Word2Vec and GloVe models represent each word of the vocabulary by a distinct vector, without parameter sharing. Mostly, these approaches ignore the internal structure of words (morphological aspects), which is a significant limitation (Bojanowski et al., 2017). Thus, to overcome this limitation was proposed a method (Bojanowski et al., 2017) that consider sub

word information to learn representations for character n-grams, and to represent words as the sum of the n-gram vectors as an extension of the continuous skip-gram model (Mikolov et al., 2013b).

Considering this strategy, FastText (Joulin et al., 2017) was developed. It is a library for efficient learning of word representations and sentence classification that is written in C++ and supports multiprocessing during training, it also supports training continuous bag of words (CBOW) or Skip-gram models using negative sampling, softmax or hierarchical softmax loss functions.

Each word is represented as a bag of character n-grams in addition to the word itself, hence taking as an example the word **"fatter"** and a window of size n = 3, then FastText represents the character n-grams on the following format:

- <fa, fat, att, tte, ter, er>, after that
- < and > are added as boundary symbols

This is made so that it is possible to distinguish the n-gram of a word from a word itself, thus, if the word fat is part of the vocabulary, it is represented as <fat>. This helps preventing that shorter words that may appear as n-grams of longer words have their meaning lost. Intrinsically, this strategy also allows to capture meaning for suffixes/prefixes.

Figure 10 shows the proposed FastText architecture.

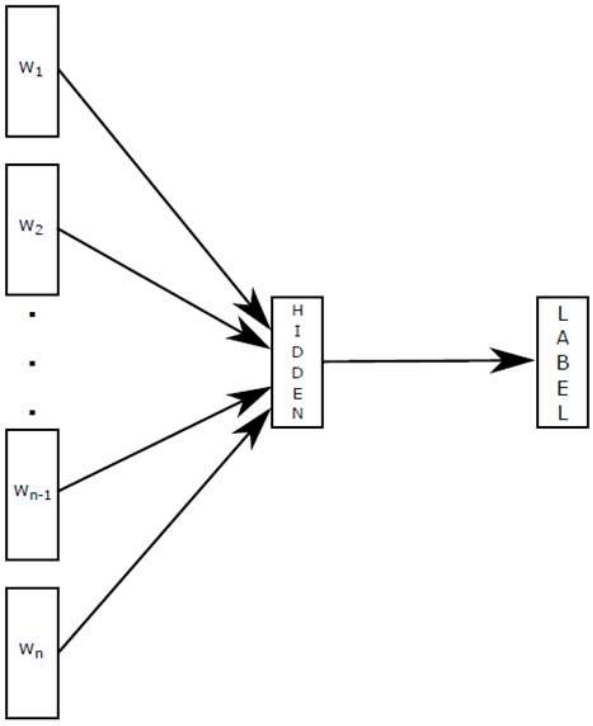


Figure 10 – Model architecture of FastText for a sentence with N n-gram features $W_1 \ldots W_N$. The features are embedded and averaged to form the hidden variable.

It worth to mention that, through the use of character level information, FastText is able to achieve good performance for word representations, especially in the case of rare words and still can overcome problems like out of the vocabulary words (Bojanowski et al., 2017).

Another important premise of FastText is that very frequent words provides less information than words that are rare, and that these word's representation does not differ a lot after many instances of the same word are seen.

Once again pre-trained word embedding obtained with FastText were used to get the embeddings for the words on the dataset of this project and the document embedding is described on next section. Additionally, 100/200/300-dimensional Embedding were obtained by training FastText on the Corpus of this project.

The pre-trained embeddings have the following characteristics:

wiki-news-300d-1M-subword.vec	
Origin of Tokens	Wikipedia 2017, UMBC webbase corpus and statmt.org news
Number of tokens	16 billion
Size of vocabulary	1.000.000 tokens
Casetype	Uncased
Embedding vectors Sizes	300 dimensions

Table 9 – Pre-trained FastText word embeddings dataset

3.3.6.4. Document Embedding

Besides word embeddings obtained, it is necessary to get the embeddings to represent the entire movie script, hence the document level embedding. Thus, the strategy employed was the one proposed by (Corrêa Jr., Marinho, & dos Santos, 2017). This approach aims to represent each document as a weighted average of the word embeddings of each word found on a movie script that integrates this project's dataset. The weight for each word is retrieved from the Term Frequency-Inverse Document Frequency (TF-IDF) matrix.

Therefore, after this stage, 11 datasets containing the embedding for each of the movie scripts and the two target variables that will be assessed on the classification task were made available.

3.3.7. Target variable Discretization

Given that the problem is modelled as a classification task, it becomes necessary to transform the target variables that are continuous into categorical variables, a process called discretization. Hence, for the dependent variable "Gross Revenue" the bins that will represent the classes that are going to be predict are derived from Delen & Sharda 2006.

Despite of having been considered nine classes on the original research, in the context of this project, "Gross Revenue" will be transformed into a binary categorical variable and into a 4-class variable, the classes and their correspondent intervals are described on **table 10**:

Original Variable	Transformed Variable	Interval (US$ million)	Class	Type of Variable
Gross Revenue	Performance Level	[0,65]	1(BAD)	Binary
Gross Revenue	Performance Level	> 65	2(GOOD)	Binary
Gross Revenue	Performance Level	[0,20]	1(VERY BAD)	Multi-class
Gross Revenue	Performance Level	[20,65]	2(BAD)	Multi-class
Gross Revenue	Performance Level	[65,150]	3(GOOD)	Multi-class
Gross Revenue	Performance Level	> 150	4(VERY-GOOD)	Multi-class

Table 10 – Gross Revenue discretization

The discretization of the variable Gross Revenue to the binary format was made considering the class1(BAD) as the concatenation of classes 1 to 5 from the original paper and class 2(GOOD) as the grouping of classes 6 to 9 from Delen & Sharda 2006. On the other hand, the discretization to the multiclass format was made by aggregating classes 1,2 and 3 from the original paper into Class 1 (very BAD), classes 4 and 5 into Class 2(BAD), classes 6 and 7 to Class 3(GOOD) and finally Class 4(Very GOOD) was derived from classes 8 and 9 from Delen & Sharda 2006.

In its turn, the variable "Tickets sold" was transformed accordingly with table 11:

Original Variable	Transformed Variable	Interval (million)	Type of Variable	Class
Tickets sold	Performance Level	[0,10]	Binary	1(BAD)
Tickets sold	Performance Level	> 10	Binary	2(GOOD)
Tickets sold	Performance Level	<1	Multi-class	1(VERY BAD)
Tickets sold	Performance Level	[1,5]	Multi-class	2(BAD)
Tickets sold	Performance Level	[5,20]	Multi-class	3(GOOD)
Tickets sold	Performance Level	> 20	Multi-class	4(VERY-GOOD)

Table 11 – Tickets Sold discretization

On the case of Target variable Tickets sold, it was not possible to identify previous works using this metric to measure box office performance, hence it was necessary to adopt an alternative strategy for the discretization of this variable. The discretization was made using the size of the bins obtained on the discretization of the variable Gross revenue, by following this approach the dataset was ordered by Tickets Sold (Ascending) and classes were defined in order to have the same quantity of individuals that was obtained by the discretization of the Gross Revenue variable. Hence the boundaries for the classes was obtained by rounding the value verified on the limits of the classes.

3.4. MODELLING

On this project, the problem was designed as a supervised learning, more specifically a classification task, where the dependent variable is either a binary variable Performance Level (BAD/GOOD) or a four-class variable (VERY BAD, BAD, GOOD, VERY GOOD). The transformation of the continuous variables T*ickets sold* and *Gross Revenue* into categorical variable Performance level is described on the data preparation section.

Supervised learning is characterized by the existence of a dependent variable, the one we want to predict the value and a set of independent variables that are used to predict the dependent variable. Its goal is to learn a function that maps the input to the output, and it is accomplished by learning directly through data, so after being trained on a sample data(training) with labelled output, the model is able to learn a function that maps the input to the output.

We can still divide supervised learning into Classification and regression, where the first try to predict the category of the output and the last tries to predict a real valued, continuous output.

Before selecting the models that were applied, it was performed a review of the input data format and the goals that needed to be achieved, in order to verify which models were suitable for these requirements. It is important to notice though, that in modelling there is no one-fits-all solution, therefore this stage is characterized not for the selection of an individual model to be trained and tested, but by the selection of multiples algorithms that, after being trained and tested, will have its performance evaluated and compared.

Thus, based on the literature review it was possible to realize that Support vector machines (SVM) has been employed extensively as a text classifier, with great results on the past on tasks such as on text categorization (Pilaszy, 2005; Joachims, 1999; Nie et al., 2014), and more recently Convolutional neural networks (CNN) has gain a lot of traction and reached state of the art results on a variety of text classification problems, such as product's reviews classification and movie´s review classification.

Hence, SVM, Neural networks and CNNs were selected as the models that will be trained and tested at this project. Next section describes the theoretical background and operation of each of these models.

Besides selecting models, training strategy and model evaluation metrics are required at this stage to enable proper evaluation of the performance of the models on the designated task, and so further selection of a set of models that will be assessed on aspects relating to business goals.

3.4.1. Support Vector Machines

Introduced by Boser, Guyon and Vapnik in their seminal paper (Boser, Guyon, & Vapinik, 1992), Support Vector Machines (SVM) have been used with success on text classification tasks and many authors have claimed its relevance for this field (Dumais et al., 1998; Taira, & Masahiko, 1998). SVM most prominent characteristics are:

- Great ability to generalize
- Adequacy to cope with high-dimensional data (Siolas, & d'Alché-Buc, 2000)

- Solid grounding in statistical learning theory (Bloehdorn et al., 2006).
- Capacity to incorporate prior knowledge about the target domain (Bloehdorn et al., 2006).
- Perform well when there is reduced amount of data available

On this context, Bag-of-words feature representation derived from Information Retrieval have been largely employed for text classification with SVMs and more recently using more advanced feature representations as word embeddings as the input to SVM classifiers (Lilleberg, Zhu, & Zhang, 2015).

An SVM is a type of large-margin classifier, it is a vector-space based machine learning method that given two classes, tries to find a decision boundary between them that is maximally far from any point in the training data, they use mathematical functions so-called kernel, that take data as input and transform it into the required form. Kernel functions can be from different types: linear, nonlinear, polynomial, radial basis function (RBF), and sigmoid. SVMs tries to maximize the margin, once in general the larger the margin the lower the generalization error of the classifier. Figure 11 illustrates a linear support vector machine.

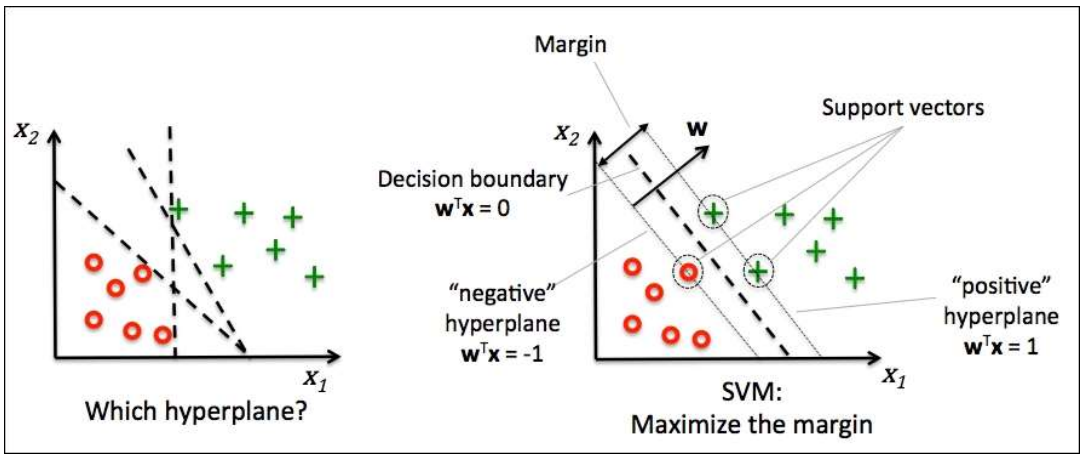


Figure 11 – Linear SVM schema (Raschka, Julian, & Hearty, 2017)

The margin of the classifier is determined by the distance from the decision surface to the closest data point and support vectors are the data points that lie closest to the decision surface (or hyperplane), they are the data points most difficult to classify and they impact directly on the optimum location of the decision surface.

The decision surface (hyperplane) separating the classes has the following form:

$W^Tx + b = 0$, where:

- w is a weight vector
- x is input vector
- b is bias

• And, the parallel hyperplanes are described by:

$W_0 + W^TX_{pos} = 1$

$W_0 + W^T X_{neg} = -1$

Resulting from the equations above we have that the margin to be maximized can be described as:

$\frac{2}{\|w\|}$ and it is converted into a quadratic optimization problem where the goal is to minimize:

$$\frac{1}{2} * \|w^2\|$$

The conditions described previously, holds only when dealing with linear separable case. In the case of very high dimensional problems, which are common in text classification, in general, data is nonlinearly separable, and so we need a solution that can ignore a few weird noise documents. Therefore, Vladimir Vapnik introduced in 1995, the concept of slack variable which led to the soft-margin classification. Slack variable was necessary to allow convergence of the optimization when misclassifications occur, under the appropriate cost penalization.

$$\frac{1}{2}\|w^2\| + C\left(\sum_i \xi^i\right)$$

A different approach is necessary for the case of multi-class classification, where the most common technique applied has been to build one-versus-rest classifiers (commonly referred to as "one-versus-all" or OVA classification), and to choose the class which classifies the test datum with greatest margin. As this project is implemented with python Scikit-Learn library (Scikit documentation reference), "one-against-one" approach (Knerr, Personnaz, & Dreyfus, 1990) for multi- class classification is applied.

Once this project works with two different target variables (binary and 4-class), each output variable was transformed accordingly to fit the requirements of support vector machine models, as described on data preparation section. It also worth to mention that, the inputs to the SVM classifier, will be the features extracted from movie script's n-dimensional vectoral representation derived from the word embeddings of the text of each movie script.

3.4.2. Artificial Neural Networks

Artificial Neural networks (ANN) are widely employed on text classification tasks (Manevitz, & Yousef, 2007; Ramasundaram, & Victor, 2010; Jo,2010; Kothari, Naik, & Rana, 2015; Ghiassi, Lio, & Moon, 2014). Advances in neural networks architectures within deep learning field has enabled state of the art achievements on several NLP tasks.

Ghiassi, Lio, & Moon, 2014, has achieved a 32.8% performance improvement over previous benchmarks on box office revenue prediction with an ANN model called Dynamic Artificial Neural Network (DAN2).

Artificial Neural networks can be basically described as mathematical models that tries to mimetize the human's brain learning process. As in our brains, neural networks have computational units called neurons, and connections between neurons that emulate human's brains synapses. A biological neuron works by receiving inputs signals from dendrites, use its cell body to process them and sign out through the axon. Artificial neurons do basically the same,

they have input channels, a processing unit, and an output that can fan out multiple other artificial neurons.

ANN is characterized by their large number of highly interconnected elements called nodes or neurons. ANNs collective behavior is recognized by their ability to learn, recall and generalize the training patterns or data like that of a human brain (Kothari, Naik, & Rana, 2015).

ANN's birth reminds to 1959, when Frank Rosenblatt proposed the idea of a Perceptron, calling it Mark I Perceptron, it was the first neural network model.

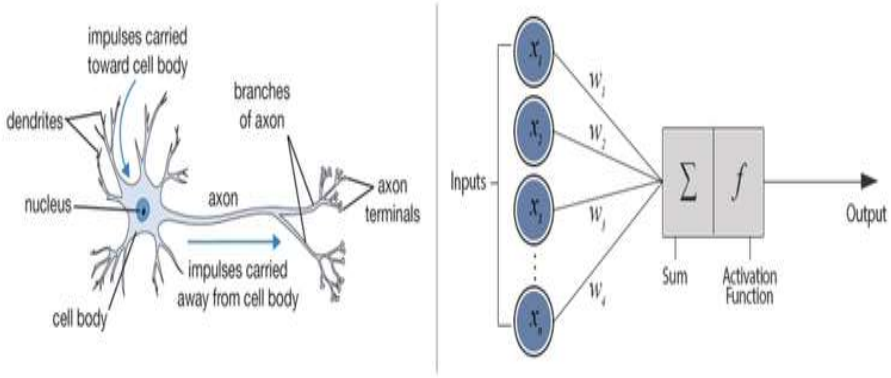


Figure 12 – Schematic representation of a perceptron and a human neuron structure

The perceptron is a simple network that can work with a unique neuron or several neurons grouped on a single layer. Since the flow of information is unidirectional, they are called also Feed Forward Neural Networks.

The learning process of a perceptron is simple:

1. Randomly initialize weights
2. For the inputs of an example in the training set, compute the Perceptron's output
3. If the output of the Perceptron(Y) is different from the correct known output(T) for the example apply:

   if $Y > T$, $W_i = W_i - \alpha X_i$

   if $Y < T$, $W_i = W_i + \alpha X_i$, where α is called learning rate

4. Take the next example in the training set, repeat steps 2-4 until the Perceptron makes no more mistakes or a fixed number of iterations

Once the perceptron learning phase is concluded it can be presented to new data on what is called the generalization phase.

Single neuron perceptron works as a binary classifier, to have more classes we need to combine more than 2(two) neurons in the network. Although it represented a huge advance, perceptron has one limitation, it can only solve linearly separable problems, thus to overcome this constraint multilayer perceptron (MLP) (Minsky & Papert, 1969) was proposed and later on

1980's backpropagation (Parker, 1985; Le Cun, 1985) was presented as a new learning algorithm that enabled the training process of MLP networks.

MLP networks are feed forward neural networks composed by 1 input layer, 1 or more hidden layers and 1 output layer, they use the backpropagation algorithm to modify the weights of the neurons that belong to the hidden layers using gradient descent to reduce the error. Figure 13 shows a 2-hidden layers MLP.

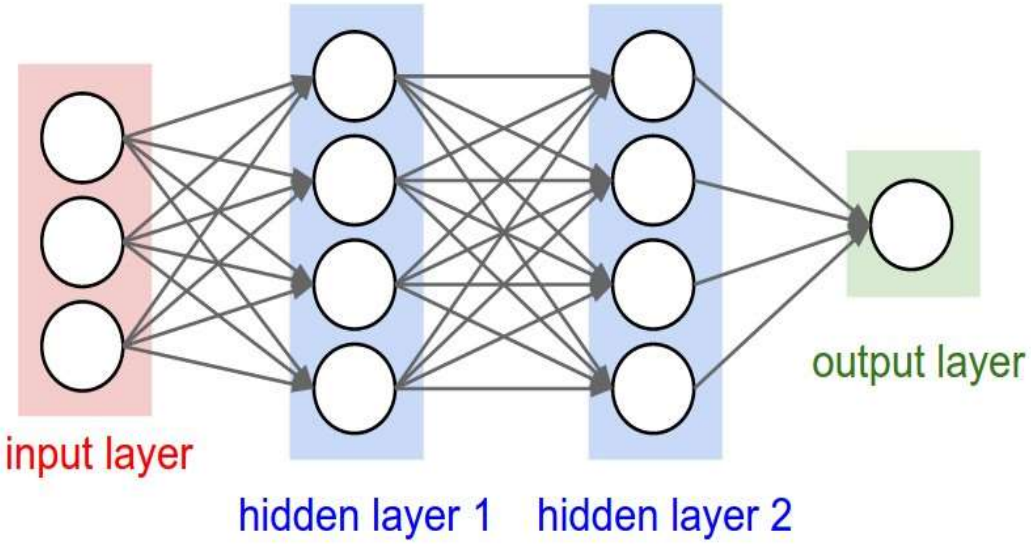


Figure 13 – 2-Hidden layers MLP example. Adapted from CS231n Convolutional neural networks for visual recognition. Retrieved October 21, 2009, from https://cs231n.github.io/convolutional-networks/

Backpropagation consists on propagating backwards to the hidden neurons, the error of the output neurons. Hence, the error of the hidden neurons is calculated as the sum of all the errors of all the output neurons to which it is directly connected.

MLP networks differentiates from single layer perceptron by the number of layers and for the way they adjust the weights, but in general terms the learning process is basically the same implemented for any single perceptron. It is important to notice that backpropagation can be implemented using any error measure, although quadratic mean square error is very commonly used.

Additionally, MLP have good generalization ability (Baum, Haussler, 1994) and have been demonstrated to be excellent classifiers (Kak, 2002), being very popular applications on a myriad of tasks like, text categorization, speech recognition and image recognition.

On this project MLP will be applied both as a binary classifier and a 4-class classifier, where the inputs to the network will be the features extracted from vectoral representation of the movie script, as described on data preparation section, and the outputs are categorical variables obtained through the discretization of the variable Tickets Sold and the variable Gross Revenue. Once selecting an MLP architecture is a very challenge task that requires a lot of experimentation to find the adequate performance, the final architecture is still not known at this point.

3.4.3. Convolutional Neural Networks

Convolutional neural networks are multilayer neural networks known to be effective at several feature's extraction tasks, mostly because their ability to identify salient features (e.g. tokens or sequences of tokens) independently to their position within the input sequences (Goldberg, 2016).

This type of networks has gained a lot of attention mostly after a computer vision challenge in 2012, with the work "ImageNet Classification with deep convolutional neural networks" (Krizhevsky, Sutskever, & Hinton, 2012) having far outperformed its challengers.

On the NLP field, more specifically on Text classification and document categorization, different works have been proposed (Zhang, Zhao, & Lecun, 2015; Johnson & Zhang, 2014; Santos & Gatti, 2014; Kim, 2014) and demonstrated CNN's ability to understand texts.

Another great advantage of CNNs is that they can be directly applied to distributed (Santos & Gatti, 2014) or discrete (Kim, 2014) embedding of words, without any knowledge on the syntactic or semantic structures of a language.

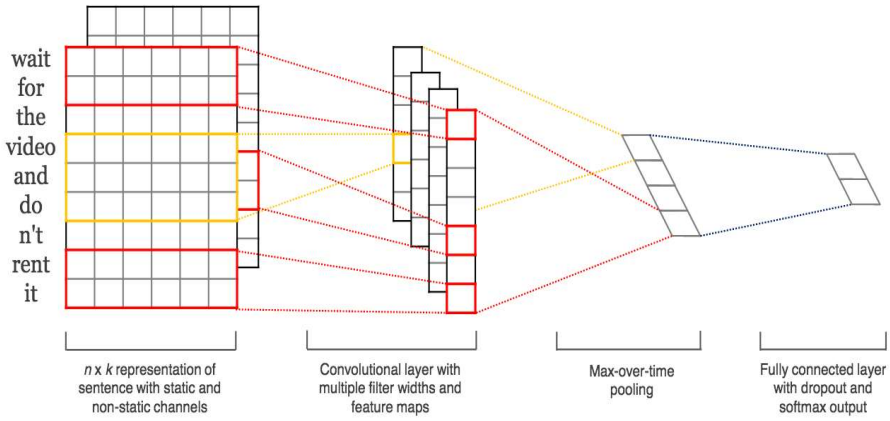


Figure 14 – CNN Filter and Pooling Architecture for Natural Language Processing example. Reprinted from Convolutional Neural Networks for Sentence Classification

CNN's architecture is characterized by a convolution layer, a pooling/subsampling layer and a fully connected layer as shown on figure 14. Convolution operations are linear transformations with an equivalent transformation matrix. It's important to notice that this operation reduces the size of the input and if we need to keep the size, we should artificially augment the input to the output have equal size to original input.

The convolution layer is responsible for learning local features in the data, on the sequence the pooling layer makes the CNN invariant to small changes of these features (if the feature suffers small changes, the network still is able to detect it), and finally we use fully connected layers, which aggregates all local features into a global representation, and makes the correct classification.

Although CNNs has demonstrated great capacity for the task on hand on this project, they have a clear drawback of being difficult to fine-tune, because they have a lot of hyperparameters to

be configured. Hence, to help on this configuration, the work of Ye Zhang and Byron Wallace “A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification”, will be used as a guide.

As described for the other models that will be trained on this project, several representations of the text of the movie scripts were generated and were used as input to the different classification models.

3.4.4. Data Partition

Before building the models that were selected to perform the text classification under study, it is necessary to design an evaluation approach which determines the learning schema that is going to be used.

This task is an essential part of machine learning projects, once error on the training data is not a good indicator of performance on future data, we must implement a data partition strategy to enable proper evaluation of the classifiers performance when dealing with new data, how well the model is learning and prevent problems like overfitting, which is when the model fits so well the training data that it is not able to generalize the knowledge into new instances.

Cross-validation is a resampling technique applied to evaluate machine learning models when there is limited amount of data. This method has become very popular because its easy understandability and because it generally provides a less biased or an estimate of the model ability to generalize that is less optimistic than other methods, such as a simple train/test split (Kohavi, 1995).

For this project, it was chosen a 10-fold-cross-validation strategy. K-fold Cross-validation (Stone, 1974) provides a simple and effective method for both model selection and performance evaluation. It splits data into k equals fold that are separated into training data and testing data.

K-fold Cross-validation procedure consists on the following steps:

1. Split data into k > 0 disjoint subsets of equal size

2. Use each subset in turn for testing, the remainder for training. The subsets can be stratified before the cross validation is performed. The error estimates are averaged to yield an overall error estimate.

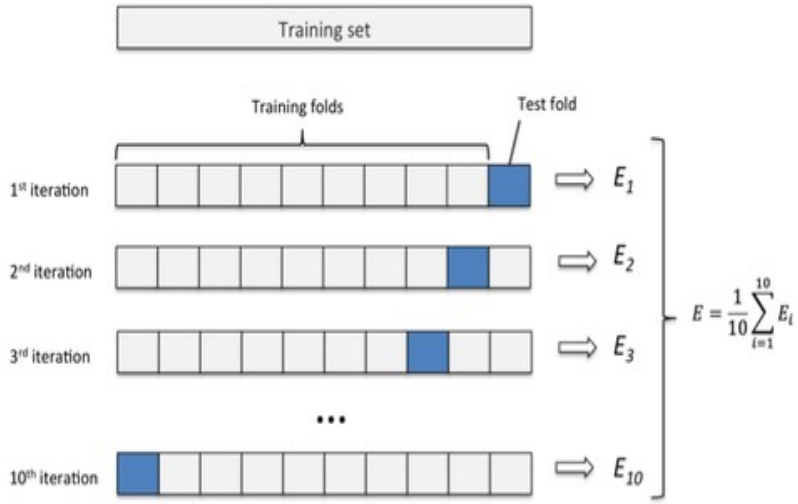


Figure 15 – 10-Fold Cross-validation schema

Once cross-validation can suffer from variance in estimates, stratification was applied to guarantee each fold was equally representative of the classes.

3.4.5. Model performance Evaluation

After models are trained, it is essential to evaluate their performance and compare each other. Thus, it is necessary to select metrics that enables adequate evaluation in accordance to the type of the problem and the models that were used. Although the problem is a classification task, it is important to consider that two different types of target variable were used, hence the metric derived from the confusion matrix have to be adapted to the multi-class model's evaluation.

Therefore, table 12 describes which metrics were used to evaluate the models in agreement with the type of the classification:

Model	Classification Problem	Metrics
SVM	Binary	Precision, Recall, F1-Score, ROC-AUC
Neural Network	Binary	
CNN	Binary	
SVM	Multi-class	
Neural Network	Multi-class	
CNN	Multi-class	

Table 12 – Evaluation metrics

3.4.5.1. Confusion Matrix

Confusion Matrix is a tabular representation that is composed by information about actual class and predicted class made by a classification model. Data in a confusion matrix can be used to

evaluate binary classifiers and multi-class classifiers, although it requires proper extension on the last case.

A great advantage of using a confusion matrix when compared to classification accuracy alone is that we can visualize and gain insights about the type of errors that the model is incurring and not only measuring the overall error. Table 13 presents a confusion matrix structure for a binary classification.

CONFUSION MATRIX		**Predicted by Classifier**		
		Positive	**Negative**	**TOTAL**
Actual Class	**Positive**	**A**	**B**	**G** = A+B
	Negative	**C**	**D**	**H** = C+ D
	TOTAL	**E** = A + C	**F** = B + D	**I** = A+ B + C + D

Table 13 – Confusion Matrix

The meaning of the fields in confusion matrix are:

A = Number of instances that were correctly classified as positive or ***True positives.***

B = Number of instances that wrongly classified as negative or ***False Negatives***.

C = Number of instances that were wrongly classified as positive or ***False Positives***.

D = Number of instances that were correctly classified as negative or ***True Negatives.***

E = Total of instances classified as positive by the model.

F = Total of instances classified as negative by the model

G = Total of instances actually positive.

H = Total of instances actually negative.

I = Total of instances

Several ratios metrics are derived from this table as:

Accuracy (AC) is the proportion of correctly classified instances.

$$Accuracy = \frac{a\ +\ d}{a\ +\ b + c\ + d}$$

The proportion of positive instances that were correctly identified is called recall, sensitivity or true positive rate (TP) and its formula is like below:

$$Recall(TP) = \frac{\mathrm{d}}{\mathrm{c}\ +\ \mathrm{d}}$$

The false positive rate (FP) is the proportion of negatives cases that were incorrectly classified as positive, as calculated using the equation:

$$False\ Positive\ Rate = \frac{\mathrm{b}}{\mathrm{a}\ +\ \mathrm{b}}$$

The true negative rate (TN) or specificity is defined as the proportion of negatives cases that were classified correctly, as calculated by the equation below:

$$True\ Negative\ Rate = \frac{\mathrm{a}}{\mathrm{a}\ +\mathrm{b}}$$

The proportion of positives cases that were incorrectly classified as negative is called false negative rate (FN) and is calculated using the equation:

$$False\ Negative\ Rate = \frac{\mathrm{c}}{\mathrm{c}\ +\mathrm{d}}$$

Precision (P) is the proportion of the predicted positive cases that were correct, as the formula below:

$$Precision\ = \frac{\mathrm{d}}{\mathrm{b}+\mathrm{d}}$$

3.4.5.2. F1-Score

This metric has been vastly applied no natural language processing works. The F1 score can be interpreted as a harmonic weighted average of the precision and recall, its value ranges from 0 to 1 where 0 is the worst result and 1 is the best performance. F1-score gives equal importance to both precision and Recall.

If it is necessary to seek a balance between Recall and Precision and an uneven class distribution (greater instances of actual negatives) is present, F1-score is usually a better measure to be employed.

The equation to calculate F1-Score is the one below:

$$F1 = 2 * \frac{precision * recall}{precision + recall}$$

It worth to mention that in the multi-class classification, it is calculated by the average of the F1 score of each class with weighting.

3.4.5.3. ROC-AUC

The ROC-AUC curve is a performance measurement for classification problems considering different thresholds settings. It enables us to identify how much a model can distinguish between classes. ROC curve is a probability curve and its axis are formed by true rate(sensitivity) and false positive rate (specificity).

To better interpret ROC-AUC we must consider that Sensitivity and Specificity are inversely proportional to each other. Hence when Sensitivity increases, Specificity decreases, and the inverse is true.

An ideal model would have a true positive rate of 1 and a false positive rate of 0. On this situation, the area under the roc curve (AUC) will be the maximum possible.

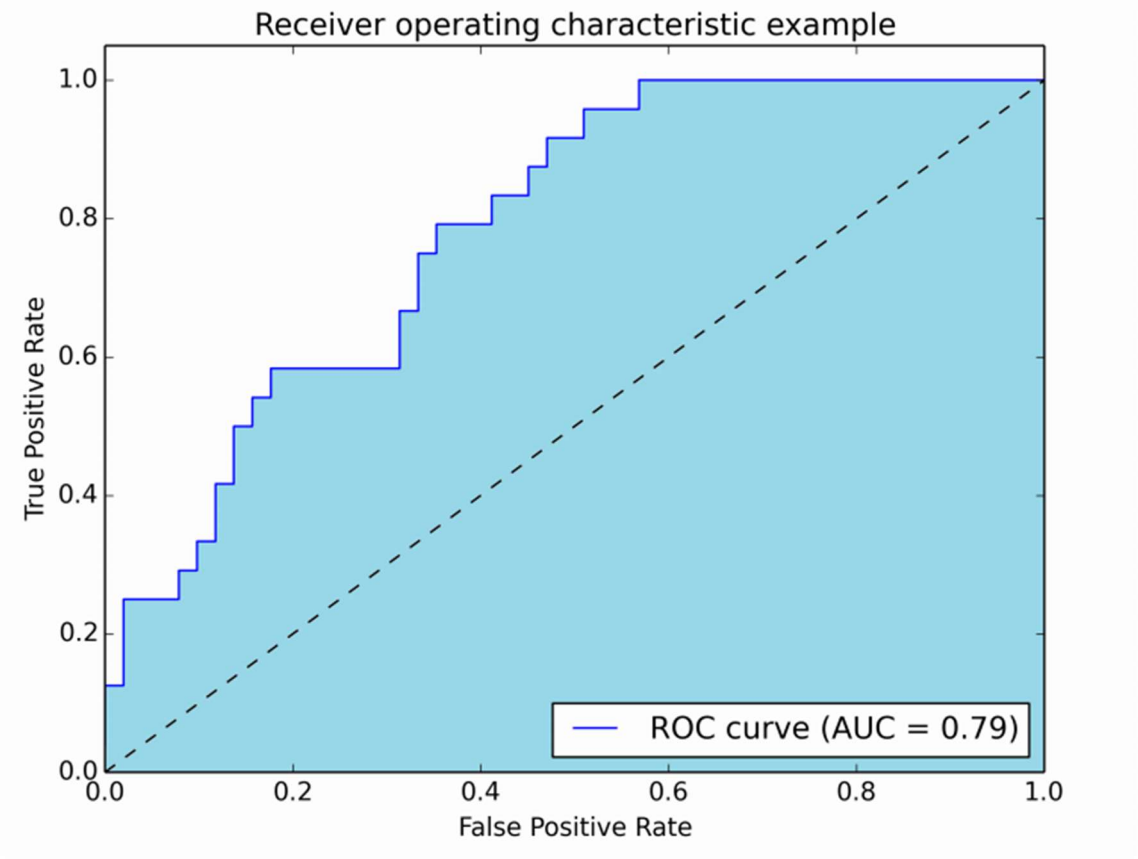


Figure 16 – ROC curve illustration

In its turn, for multi-classification if we have N classes, we can plot N number of AUC ROC Curves using One versus ALL methodology. Thus, for example, considering a 3-classes problem with classes A, B, and C, we employ one ROC for A classified against B and C, another ROC for B classified against A and C, and a third one of C classified against A and B. Thus, for both binary and multiclass classification models, the objective is maximize the area under the ROC curve.

3.5. EVALUATION

The main objective of the evaluation stage is to assess from a business perspective if the project achieved the outcomes that were expected. Thus, from section 1.2, it is possible to describe the following business objectives that were established and can be directly measured:

- Develop a classification model that can predict a movie's box-office performance helping ANCINE on the decision-making process, regarding to the selection process of projects that apply for public funding;
- Reduce time spent on project's analysis, by automatizing a part of the process;
- Reduce costs with the hiring of specialized professionals that are needed to analyze movie's script;
- Improve ANCINE's selection process transparency by making available publicly and automatically the results of the evaluation of each movie script submitted to the classification of the model;
- Serve as a paradigm to other initiatives at ANCINE and other Brazilian public organizations, expanding machine learning's utilization to deliver more agile and efficient services to citizens.
- The evaluation of these objectives is demonstrated at the results section of this work.

4. RESULTS AND DISCUSSION

This project aims to select a machine-learning model that is able to classify movie's box office performance based only on features automatically extracted from the movie's script. Hence, a diverse set of classifiers were trained and tested, using as target variable two distinct formats, namely binary and multiclass variables and different embedding sizes and models, for the representation of the movies' scripts.

On what concerns to the evaluation of the business objectives established on section 1.2 and expressed on section 3.5, the following results can be described:

The development of a classification model can be immediately assessed by the delivery of this master´s project. The transparency improvement and acting as a paradigm objective are indirect measurements and can only be evaluated if the project is approved for implementation on production.

Table 14 shows the results obtained on this project in what concerns to time and cost reduction when compared with the current analysis process implemented. This project can deliver assessment for a project, considering expected box office performance, 5760 times faster than the current process at a 0,1% of the current cost.

Process	Analysis Time	Analysis Cost(R$)
Current	1 movie script/48 hours*	593,04[7].
Proposed	1 movie script/30 seconds	0,62[8].

Table 14 – Project's Results x Current Process

Next section presents the result for each model and indicates criterion adopted to select the best model for each variable considered as movie's performance measure. Additionally, the results were compared with other works on this research field, but comparison is limited because in general most of the works considered a broader set of input variables that are available on production or post-production stage, while this project was limited to automatically extracted features from movie's script, which is available on pre-production phase.

4.1. GROSS REVENUE (BINARY FORMAT)

The models for these classifiers were trained and tested aiming to maximize the F1-score metric and Precision, AUC was used only as secondary evaluator when needed. The priority

[7] Available at https://www.ancine.gov.br/sites/default/files/edital_credenciamento.pdf

[8] Considering the cost of a windows on p2.16xlarge instance on AWS cloud. The calculation site is available at https://calculator.s3.amazonaws.com/index.html

to the precision metric is related with the assumption that is more important to assure that majority of the films predicted as having good box office performance are really successful.

All the results are on table 15. Three different algorithms were applied on this project: Support Vector machine (SVM), Neural Network (NN) and Convolutional Neural Network (CNN).

It's important to notice that all models utilized a 10-fold stratified cross-validation strategy. This strategy was defined to overcome the challenge posed by the imbalanced nature of the dataset that presented a proportion of 70% of the movies in class 1(BAD) and 30% on class2 (Good), and the size of the dataset that, after all data preparation tasks, remained with only 1141 movies' scripts.

For transforming the movie's scripts were used three Word embedding models, namely Word2Vec, Glove e FastText. Following, final document embedding was obtained by a weighted averaging based on TF-IDF. Hence, all the movies' scripts assumed a n-dimensional format (n ranging from 100 to 300), which was employed as input to the classification models. It is important to notice that for Word2Vec and FastText three embeddings (100, 200, 300) were trained on the corpus of this project.

EMBEDDING		METRIC	SVM	MLP	CNN
MODEL	SIZE				
WORD2VEC	100	PRECISION	0,73	0,70	0,69
		RECALL	0,75	0,73	0,75
		F1 SCORE	0,73	0,71	0,70
		AUC	0,60	0,58	0,53
	200	PRECISION	0,75	0,74	0,70
		RECALL	0,78	0,75	0,74
		F1 SCORE	0,75	0,74	0,71
		AUC	0,60	0,63	0,57
	300	PRECISION	0,77	0,71	0,68
		RECALL	0,79	0,72	0,75
		F1 SCORE	0,76	0,71	0,68
		AUC	0,62	0,59	0,51
	300(PRETRAINED)	PRECISION	0,78	0,70	0,58
		RECALL	0,80	0,73	0,76
		F1 SCORE	0,77	0,71	0,66
		AUC	0,64	0,56	0,50
GLOVE					
	100(PRETRAINED)	PRECISION	0,75	0,72	0,77
		RECALL	0,78	0,76	0,78
		F1 SCORE	0,75	0,72	0,71
		AUC	0,60	0,56	0,54
	200(PRETRAINED)	PRECISION	0,73	0,73	0,70
		RECALL	0,76	0,75	0,67
		F1 SCORE	0,74	0,74	0,69
		AUC	0,60	0,61	0,60
	300(PRETRAINED)	PRECISION	0,73	0,73	0,58
		RECALL	0,75	0,74	0,76
		F1 SCORE	0,73	0,73	0,67
		AUC	0,60	0,62	0,51

FASTTEXT	100	PRECISION	0,78	0,74	0,70
		RECALL	0,79	0,77	0,76
		F1 SCORE	0,74	0,75	0,67
		AUC	0,58	0,60	0,51
	200	PRECISION	0,74	0,66	0,64
		RECALL	0,77	0,68	0,74
		F1 SCORE	0,75	0,67	0,66
		AUC	0,61	0,53	0,50
	300	PRECISION	**0,80**	0,67	0,68
		RECALL	**0,81**	0,68	0,75
		F1 SCORE	**0,77**	0,68	0,68
		AUC	**0,62**	0,54	0,51
	300(PRETRAINED)	PRECISION	0,68	0,71	0,58
		RECALL	0,72	0,73	0,76
		F1 SCORE	0,70	0,72	0,66
		AUC	0,55	0,59	0,50

Table 15 – Results for all models with GROSS REVENUE binary format target

It is possible to observe from table 15 that the best F1-score was achieved by the SVM classifier. The first model used as input variables, features extracted from movie script with a WORD2VEC model and an embedding size of 300, which was obtained via pretrained dataset of Google-News. The second model had its features extracted with FASTEXT model with an embedding size of 300, which was obtained through training the model on the Corpus of this project.

Further, comparing these 2 classifiers by their PRECISION score it was possible to verify a slightly better performance of the classifier that was trained on the Corpus (domain-specific) of the project with FastText, which has reached a Precision of 0,80 against a Recall Score of 0,78 from the classifier that utilized a pretrained embedding of WORD2VEC model.

Hence, the best binary classifier (GROSS REVENUE) was a Support Vector Machine with an input of size 300(FASTTEXT). The configuration of the hyperparameters of the model was:

Parameter's Name	Parameter's Value
GAMMA	0.0001
C	100.
DECISION_FUNCTION_SHAPE	OVR
KERNEL	RBF

Table 16 – Support vector machine configuration for the best model

Figure 17 shows the confusion matrix for the best binary model measuring performance as Gross Revenue.

CONFUSION MATRIX		Predicted by Classifier	
		1(BAD)	2(GOOD)
Actual Class	1(BAD)	**86**	**2**
	2(GOOD)	**20**	**7**

Figure 17 – Confusion Matrix for the SVM-300-FASTTEXT classifier

4.2. GROSS REVENUE (4 CLASSES FORMAT)

Regarding to the classifiers that had target variable on multiclass format (4 classes), the performance was much lower. The algorithms and the input for the classifiers were the same for the binary classifiers, hence the only difference is on the format of the target variable. Table 17 shows the results.

EMBEDDING		METRIC	SVM	MLP	CNN
MODEL	SIZE				
WORD2VEC	100	PRECISION	0,35	0,41	0,36
		RECALL	0,42	0,45	0,48
		F1 SCORE	0,37	0,39	0,39
		AUC	0,53	0,55	0,55
	200	PRECISION	0,40	0,42	0,37
		RECALL	0,39	0,44	0,39
		F1 SCORE	0,38	0,42	0,35
		AUC	0,52	0,57	0,54
	300	PRECISION	0,37	0,42	0,19
		RECALL	0,36	0,40	0,44
		F1 SCORE	0,35	0,39	0,27
		AUC	0,50	0,54	0,50
	300(PRETRAINED)	PRECISION	0,35	0,36	0,29
		RECALL	0,35	0,35	0,34
		F1 SCORE	0,34	0,35	0,37
		AUC	0,49	0,50	0,49
GLOVE					
	100(PRETRAINED)	PRECISION	0,29	0,34	0,19
		RECALL	0,39	0,44	0,44
		F1 SCORE	0,32	0,34	0,27
		AUC	0,49	0,52	0,50
	200(PRETRAINED)	PRECISION	0,34	0,36	0,35
		RECALL	0,38	0,45	0,42
		F1 SCORE	0,35	0,37	0,36
		AUC	0,50	0,53	0,55
	300(PRETRAINED)	PRECISION	0,31	**0,48**	0,19
		RECALL	0,34	**0,49**	0,44
		F1 SCORE	0,32	**0,43**	0,27
		AUC	0,47	**0,51**	0,50
FASTTEXT	100	PRECISION	0,33	0,32	0,19
		RECALL	0,42	0,43	0,44
		F1 SCORE	0,35	0,33	0,27
		AUC	0,51	0,51	0,50

	200	PRECISION	0,32	0,33	0,20
		RECALL	0,34	0,42	0,44
		F1 SCORE	0,32	0,34	0,27
		AUC	0,47	0,51	0,50
	300	PRECISION	0,36	0,29	0,30
		RECALL	0,36	0,41	0,44
		F1 SCORE	0,34	0,33	0,29
		AUC	0,49	0,50	0,50
	300(PRETRAINED)	PRECISION	0,31	0,31	0,19
		RECALL	0,35	0,39	0,44
		F1 SCORE	0,32	0,32	0,27
		AUC	0,49	0,48	0,50

Table 17 – Results for All Models with GROSS REVENUE (4-classes Format) TARGET

The best performance on this case was achieved by a Neural Network, reaching a poor F1-score of 0,43 and a Precision of only 0,48. This model utilized as input features extracted with pretrained embeddings of GLOVE model and embedding size of 300. The configuration of this Neural Network is as follow:

Layers	Nº of Neurons	Activation Function
INPUT	300	TANH
HIDDEN	450	RELU
OUTPUT	4	SIGMOID

Table 18 – Results for All Models with GROSS REVENUE (4-classes Format) TARGET

Observing the results of table 17 it's clear that was not possible to obtain a model with adequate performance for being implemented as a decision support tool.

4.3. TICKETS SOLD (BINARY FORMAT)

When the movies' box office performance was evaluated by the Tickets sold perspective, on a binary format, the best results were obtained by a Neural Network with Glove pretrained embeddings of size 200, as shown on table 19.

EMBEDDING		METRIC	SVM	MLP	CNN
MODEL	SIZE				
WORD2VEC	100	PRECISION	0,71	0,74	0,61
		RECALL	0,74	0,75	0,60
		F1 SCORE	0,68	0,71	0,61
		AUC	0,56	0,59	0,50
	200	PRECISION	0,68	0,73	0,61
		RECALL	0,72	0,75	0,60
		F1 SCORE	0,67	0,72	0,60
		AUC	0,56	0,61	0,51
	300	PRECISION	0,71	0,69	0,63
		RECALL	0,74	0,73	0,68
		F1 SCORE	0,70	0,70	0,65

		AUC	0,59	0,59	0,53
	300(PRETRAINED)	PRECISION	0,68	0,76	0,52
		RECALL	0,72	0,77	0,72
		F1 SCORE	0,69	0,74	0,60
		AUC	0,57	0,63	0,50
GLOVE					
	100(PRETRAINED)	PRECISION	0,73	0,75	0,77
		RECALL	0,75	0,76	0,75
		F1 SCORE	0,69	0,72	0,69
		AUC	0,57	0,61	0,57
	200(PRETRAINED)	PRECISION	0,72	**0,77**	0,66
		RECALL	0,75	**0,78**	0,72
		F1 SCORE	0,72	**0,75**	0,63
		AUC	0,61	**0,64**	0,52
	300(PRETRAINED)	PRECISION	0,66	0,64	0,52
		RECALL	0,69	0,69	0,72
		F1 SCORE	0,67	0,65	0,61
		AUC	0,56	0,54	0,51
FASTTEXT	100	PRECISION	0,69	0,72	0,63
		RECALL	0,73	0,75	0,71
		F1 SCORE	0,66	0,70	0,63
		AUC	0,59	0,54	0,51
	200	PRECISION	0,61	0,69	0,60
		RECALL	0,70	0,73	0,69
		F1 SCORE	0,62	0,69	0,62
		AUC	0,50	0,57	0,50
	300	PRECISION	0,74	0,69	0,63
		RECALL	0,75	0,73	0,66
		F1 SCORE	0,71	0,69	0,64
		AUC	0,60	0,58	0,53
	300(PRETRAINED)	PRECISION	0,68	0,65	0,52
		RECALL	0,72	0,69	0,72
		F1 SCORE	0,69	0,66	0,60
		AUC	0,57	0,54	0,50

Table 19 – Results for all models with TICKETS SOLD binary format target

The Neural Network with inputs obtained through pre-trained GLOVE embeddings of size 200 has achieved a F1 score of 0,75 and a Precision of 0,77, which is a little bit lower than the performance obtained by the SVM model using GROSS Revenue on binary format as target variable.

Configuration of the network is as follow:

Layers	Nº of Neurons	Activation Function
INPUT	200	TANH
HIDDEN	400	RELU
OUTPUT	4	SIGMOID

Table 20 – Configuration of the Neural Network which achieved the best performance

The confusion matrix for this model is illustrated on figure 18.

CONFUSION MATRIX		Predicted by Classifier	
		1(BAD)	2(GOOD)
Actual Class	1(BAD)	79	4
	2(GOOD)	21	11

Figure 18 – Confusion Matrix for the Neural Network N-200(PRE)-GLOVE classifier

4.4. TICKETS SOLD (4 CLASSES FORMAT)

Finally, for the classifiers using as a target the total quantity of Tickets Sold on a 4 classes format, again it was not possible to achieve an acceptable level of performance for any of the configurations, as shown on Table 21.

EMBEDDING MODEL	SIZE	METRIC	SVM	MLP	CNN
WORD2VEC	100	PRECISION	0,38	0,38	0,14
		RECALL	0,35	0,35	0,37
		F1 SCORE	0,35	0,34	0,20
		AUC	0,53	0,53	0,50
	200	PRECISION	0,37	**0,41**	0,14
		RECALL	0,36	**0,40**	0,37
		F1 SCORE	0,36	**0,39**	0,20
		AUC	0,54	**0,56**	0,50
	300	PRECISION	0,40	0,37	0,14
		RECALL	0,39	0,37	0,37
		F1 SCORE	0,39	0,37	0,20
		AUC	0,55	0,54	0,50
	300(PRETRAINED)	PRECISION	0,30	0,34	0,14
		RECALL	0,31	0,35	0,37
		F1 SCORE	0,30	0,34	0,20
		AUC	0,51	0,53	0,50
GLOVE					
	100(PRETRAINED)	PRECISION	0,33	0,34	0,14
		RECALL	0,33	0,35	0,37
		F1 SCORE	0,32	0,32	0,20
		AUC	0,51	0,52	0,50
	200(PRETRAINED)	PRECISION	0,34	0,41	0,14
		RECALL	0,34	0,40	0,37
		F1 SCORE	0,34	0,39	0,20
		AUC	0,52	0,55	0,50
	300(PRETRAINED)	PRECISION	0,25	0,39	0,14
		RECALL	0,25	0,38	0,37
		F1 SCORE	0,25	0,38	0,20
		AUC	0,46	0,55	0,50
FASTTEXT	100	PRECISION	0,38	0,30	0,14
		RECALL	0,35	0,33	0,37
		F1 SCORE	0,34	0,30	0,20
		AUC	0,52	0,50	0,50
	200	PRECISION	0,36	0,28	0,14

		RECALL	0,35	0,31	0,37
		F1 SCORE	0,35	0,29	0,20
		AUC	0,52	0,49	0,50
	300	PRECISION	0,32	0,39	0,14
		RECALL	0,31	0,38	0,37
		F1 SCORE	0,31	0,38	0,20
		AUC	0,50	0,55	0,50
	300(PRETRAINED)	PRECISION	0,38	0,29	0,14
		RECALL	0,37	0,28	0,37
		F1 SCORE	0,37	0,26	0,20
		AUC	0,55	0,46	0,50

Table 21 – Results for all models with TICKETS SOLD as a Multiclass Target

The best performance on this case was achieved by a Neural Network, reaching a poor F1-score of 0,39 and a Precision of only 0,41. This model used as input features extracted with 200 dimensional WORD2VEC embeddings trained on the Corpus of this project. The configuration of this Neural Network is as follow:

Layers	Nº of Neurons	Activation Function
INPUT	200	TANH
HIDDEN	450	RELU
OUTPUT	4	SIGMOID

Table 22 – Configuration of the neural network that achieved the best performance

5. CONCLUSION

This work has its main objective implementing a machine learning model that could predict movies' box office performance and as complementary goals enable a decision support tool do reduce cost and time of analysis spent by the Brazilian Film Agency (ANCINE).

Best performance was achieved by the SVM model with FastText embeddings trained on the domain-specific task, this model outperformed all others on this study reaching a F1-score of 0,77.

Although the champion model has been obtained with a domain-specific embedding, in general, pre-trained embeddings outperformed embeddings trained on the corpus of the project. This result is in accordance with literature about word embeddings techniques that demonstrate that for small datasets, which is clearly the case of this project, pre-trained word embeddings deliver better results than domain-specific ones.

Another important finding is that the performance of classifiers having a binary target was far better than the performance of the models that have a dependent variable on multi-class format, what is expected too once multiclass classification problems are more difficult to address.

It is important to notice that none of the multi-classes models has achieved an acceptable level of performance, with the best of them reaching a F1-score of only 0,43 and a Precision of at maximum 0,48.

When comparing the results of this project with similar works it is possible to verify that it has reached very similar performance to the work developed by O'Driscoll, S. (2016) which has reached at most a F1-score of 0,79 and 0,65 of precision.

Another similar work, which was not applied to movie scripts but to TV series scripts, had also similar performance to this work when compared the multiclass models, the performance achieved by the present work was a F1-Score of 0,43 on a 4 classes classification task while the similar study presented a performance of 0,48 (F1-score) on a 3 classes problem.

Overall, the neural network model achieved better performance than the SVM models and CNN models, considering all combinations of word embedding size and model, although a SVM model has achieved the best classification performance.

CNN models have achieved the poorest performance among the models tested on this work. These results may be explained by the size of the dataset, which was composed by only 1141 samples. CNN models are type of models that require many data to achieve good level of performance, hence the results are compatible with the literature on this field.

Despite of having achieved relevant performance level, for an enterprise level decision support tool, it is not possible to immediately conclude that the champion model can be directly applied to real business cases.

Considering that the performance for both target variables are similar, it is possible to adapt the use of each model to the business objective that is trying to be achieved which deliver great flexibility to decision-makers.

When evaluating process performance optimization from both a financial and time perspective the improvements that can be achieved by the incorporation of machine learning based decision support tools for the case under study is very clear, reducing the cost on scripts analysis from R$593,04 (five hundred ninety three and four cents) to R$0,62 (sixty two cents) and the time spent on movies' scripts analysis going down from 48 hours to 30 seconds.

6. LIMITATIONS AND RECOMMENDATIONS FOR FUTURE WORK

The main limitation of this work was the size of the dataset, with only 1141 samples it was very difficult to reach greater levels of performance, mainly for the CNN models and when the target variable assumed a 4-class format.

The strategy of working with word embeddings as features for the classification task came with the lack of interpretability it poses on the model. This can clearly represent a limitation for decision-makers that want to explain objectively their decision.

Aggregating more variables on a future work, as genre and others not only based on the movie script, can improve the performance of the model and enhance interpretability, fostering the adoption of this kind of model.

Another important recommendation is trying to explore different strategies to deal with out of vocabulary words, this can help reduce the impact of this type of problem on the construction of the representation of the movie script.

For future work, more recent embeddings techniques as (Bidirectional Encoder Representations from Transformers (BERT) and ELMO that can treat a word differently considering the context it appears in a text can help improve the performance of this type of document classification model too.

Considering that ANCINE is a Brazilian institution that deals with documents on Portuguese language, it is important on future work to develop a movie script database with scripts in Portuguese and that is concerned with layout standardization, which can significantly facilitate processing movies' scripts.

Another important improvement for the context of ANCINE is to adapt the performance metrics to consider the institutional objectives of this public organization.

7. BIBLIOGRAPHY

Apala, R. K., Merin, J., Motnam, S., Chan, C.-C., Liszka, Kathy, J., & Gregorio, F. (2013). Prediction of movies box office performance using social media. Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, 1209-1214. https://doi.org/10.1145/2492517.2500232

Asur, S., & Huberman, B. A. (2010). Predicting the Future with Social Media. Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 01, 492-499. https://doi.org/10.1109/WI-IAT.2010.63

Baimbrige, M. (1997). Movie admissions and rental income: The case of James Bond. Applied Economics Letters, 4(1), 57-61. https://doi.org//10.1080/758521834

Bloehdorn, S., Basili, R., Cammisa, M., & Moschitti, A., (2006). Semantic Kernels for Text Classification Based on Topological Measures of Feature Similarity. Proceedings of the 6th IEEE International Conference on Data Mining (ICDM 2006) 808-812. DOI:10.1109/ICDM.2006.141

Boccardelli, P., Brunetta, F., & Vicentini, F. (2008).What is critical to success in the movie industry? A study on key success factors in the Italian motion picture industry. Creative Industries and Intellectual Property conferenc, 22-23.

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching Word Vectors with Subword Information. TACL, 5, 135-146.

Breiman, L. (1996). Bagging Predictors.Machine Learning, 24(2), 123–140. https://doi.org/10.1023/A:1018054314350

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J., (1984). Classification and Regression Trees. Boca Raton, FL: CRC press. doi:10.1002/cyto.990080516

Cutting, J. E. (2016). Narrative theory and the dynamics of popular movies. Psychonomic Bulletin & Review, 23(6), 1713–1743. https://doi.org/10.3758/s13423-016-1051-4

Dabhade, K. R. (2015). A kernel-based approach: using movie script for assessing box office performance. Global Journal of Engineering Science and Research Management, 2(9), 175-179.

De Silva, I. (1998). Consumer selection of motion pictures, appeared in The Motion Picture Mega-Industry by B. Litman. Allyn & Bacon Publishing, Inc.: Boston, MA.

Del Vecchio, M., Kharlamov, A., Parry, G., & Pogrebna, G. (2018). The Data Science of Hollywood: Using Emotional Arcs of Movies to Drive Business Model Innovation in Entertainment Industries. arXiv:1807.02221 [cs.CL]

Dumais, S., Platt, J., Heckerman, J., & Sahami, M., (1998). Inductive learning algorithms and representations for text categorization. In Proceedings of ACM-CIKM98.

Einav, L. (2007). Seasonality in the U. S. motion picture industry. The Rand Journal of Economics, 38, 127–145.

Elberse, A. & J. Eliashberg (2002). The Drivers of Motion Picture Performance: The Need to Consider Dynamics, Endogeneity and Simultaneity, to appear in the Proceedings of the Business and Economic Scholars Workshop in Motion Picture Industry Studies, Florida Atlantic University, 1–15

Eliashberg, J., & Sawhney, M. S. (1994). Modeling goes to hollywood: Predicting individual differences in movie enjoyment. Management Science, 40(9), 1151–1173

Eliashberg, J., A. Elberse, & Mark A.A.M. Leenders (2006). The Motion Picture Industry: Critical Issues in Practice, Current Research, and New Research Directions. MarketingScience 25(6), 638–661.

Eliashberg, J., Hui, S.K., & Zhang, Z.J. (2010). Green-lighting Movie Scripts: Revenue Forecasting and Risk Management. Ph.D. thesis, University of Pennsylvania

Eliashberg, J., Junker, J. J., Sawhney, M. S., & Wierenga, B. (2000). MOVIEMOD: An implementable decision support system for prerelease market evaluation of motion pictures. Marketing Science, 19(3), 226–243.https://doi.org/10.1287/mksc.19.3.226.11796

Eliashberg, J., Sam, K., Hui, Z., & Zhang, J. (2007). From Story Line to Box Office: A New Approach for Green-Lighting Movie Scripts. Management Science, 53(6), iv-1031.https://doi.org/10.1287/mnsc.1060.0668

Firth, J.R. (1962). A synopsis of linguistic theory 1930-1955, in Studies in Linguistic Analysis 1930-1955, page 11

Ghiassi, M., Lio, D., & Moon, B., (2014). Pre-production forecasting of movie revenues with a dynamic artificial neural network. Expert Systems with Applications (2014). doi: http://dx.doi.org/10.1016/j.eswa. 2014.11.022

Goldberg, Y. (2016). A Primer on Neural Network Models for Natural Language Processing. J. Artif. Intell. Res., 57, 345-420.

Goldberg, Y. (2017). Neural Network Methods in Natural Language Processing. Synthesis Lectures on Human Language Technologies, 10(1), 1-309. https://doi.org/10.2200/S00762ED1V01Y201703HLT037

Gopinath, S., Pradeep, K. C., & Sriram, V. (2013). Blogs, Advertising, and Local-Market Movie Box Office Performance. Management Science, 59(12), 2635-2853. https://doi.org/10.1287/mnsc.2013.1732

Hunter, S., Smith, S., & Singh, S. (2016). Predicting box office from the screenplay: A text analytical approach. Journal of Screenwriting. 7. 135-154. https://doi.org/10.1386/josc.7.2.135_1.

Im, D., & Nguyen, M. T. (2011). Predicting Box-Office Success of Movies in the U. S. Market, 1–5.

Jo, T., (2010). NTC (Neural Text Categorizer): Neural Network for text categorization. International Journal of Information Studies, 2(2).

Joachims, T., (1998). Text categorization with Support Vector Machines: Learning with many relevant features. Machine Learning: ECML-98, 1398, 137-142.

Johnson, R., & Zhang, T., (2014). Effective use of word order for text categorization with convolutional neural networks. Doi: 10.3115/v1/N15-1011

Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2017). Bag of Tricks for Efficient Text Classification. EACL.

Júnior, E.A., Marinho, V.Q., & dos Santos, L.B. (2017). NILC-USP at SemEval-2017 Task 4: A Multi-view Ensemble for Twitter Sentiment Analysis. SemEval@ACL.

Kim, Y., (2014). Convolutional neural networks for sentence classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 1746–1751.

Knerr, S., Personnaz, L., & Dreyfus, G., (1990). Single-layer learning revisited: A stepwise procedure for building and training neural network. Neurocomputing: Algorithms, Architectures and Applications, NATO ASI, Berlin: Springer-Verlag

Kohavi, R., (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In Proceedings of International Joint Conference on AI, 1137–1145

Kothari, V., Naik, C., & Rana, Z., (2015). Document Classification using Neural Networks Based on words. International Journal of Advanced Research in Computer Science, 6 (2), 183-188.

Krizhevsky, A., Sutskever, I., & Hinton, G., (2012). ImageNet classification with deep convolutional neural networks. Proceedings of the 25th International Conference on Neural Information Processing Systems, 1, 1097-1105.

Landauer, T. K., & Dumais, S. T. (1997). A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. Psychological Review, 104(2), 211-240. http://dx.doi.org/10.1037/0033-295X.104.2.211

Lash, M., Fu, S., Wang, S., & Zhao, K. (2015). Early Prediction of Movie Success — What, Who, and When.International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction SBP (2015), 345-349.https://doi.org//10.1007/978-3-319-16268-3_41

Le Cun, Y. (1985). A Learning Process in an Asymmetric Threshold Network. Proceedings of Cognitiva 85, 599-604

Lilleberg, J., Zhu, Y., & Zhang, Y., (2015). Support vector machines and Word2vec for text classification with semantic features. IEEE 14th International Conference on Cognitive Informatics & Cognitive Computing (ICCI*CC), 136-140. Doi:10.1109/ICCI-CC.2015.72593772015

Litman, B. R. (1983). Predicting Succes of Theatrical Movies: An Empirical Study. Journal of Popular Culture, 159–175.

Litman, B. R., & Ahn, H. (1998). Predicting financial success of motion pictures. In B. R. Litman (Ed.), The motion picture mega-industry. Boston, MA: Allyn & Bacon Publishing, Inc.

Litman, B. R., & Kohl, L. S. (1989). Predicting Financial Success of Motion Pictures: The ’80s Experience. Journal of Media Economics, 2(2), 35–50. https://doi.org/10.1080/08997768909358184

Major, V., Surkis, A., & Aphinyanaphongs, Y., 2017. Utility of General and Specific Word Embeddings for Classifying Translational Stages of Research. AMIA 2018 Annual Symposium.

Manevitz, L., & Yousef, M., (2007). One-class document classification via Neural Networks. Neurocomputing, 70, (7–9), 1466-1481.

Meiseberg, B., & Ehrmann, T. (2013). Diversity in teams and the success of cultural products. Journal of Cultural Economics, 37(1), 61-86. https://doi.org/10.1007/s10824-012-9173-7

Mestyán, M., Yasseri, T., & Kertész, J. (2013). Early Prediction of Movie Box Office Success Based on Wikipedia Activity Big Data. PloS one, 8(8), 1-8. [e71226]. https://doi.org/10.1371/journal.pone.0071226

Mikolov, T., Chen, K., Corrado, G., & Dean, J., (2013). Efficient Estimation of Word Representations in Vector Space. In Proceedings of Workshop at ICLR, arXiv; 2013. p. 1301-3781.

Mikolov, T., Chen, K., Corrado, G., & Dean, J., (2013b). Distributed Representations of Words and Phrases and their Compositionality. NIPS

Mikolov, T., Yih, W.T., & Zweig, G., (2013). Linguistic Regularities in Continuous Space Word Representations. Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL|HLT), 746-751.

Minsky, M., & Papert, S. (1969). Perceptrons. Oxford, England: M.I.T. Press

Naili, M., Chaibi, A. H., & Ghezala, H.H.B., (2017). Comparative study of word embedding methods in topic segmentation. Procedia Computer Science, 112, 340-349.

Neelamegham, R., & Chintagunta, P. (1999). A Bayesian Model to forecast new product performance in domestic and international markets. Marketing Science, 18(2), 115–136

Nelmes, J. (2007). Introduction to film studies, 4th edn, Routledge, London.

Nie, F., Huang, Y., Wang, X., & Huang, H., (2014). New primal SVM solver with linear computational cost for big data classifications. 31st International Conference on Machine Learning (ICML 2014), 3, 1883-1891.

O'Driscoll, S. (2016). Early prediction of a film's box office success using natural language processing techniques and machine learning. Masters thesis, Dublin, National College of Ireland.

Parimi, R., & Caragea, D. (2013). Pre-release Box-Office Success Prediction for Motion Pictures. Machine learning and data mining in pattern recognition. 9th international conference, 571-585. https://doi.org/ 10.1007/978-3-642-39712-7_44

Parker, D.B., (1985). Learning-Logic: Casting the Cortex of the Human Brain in Silicon. Technical Report Tr-47, Center for Computational Research in Economics and Management Science. MIT Cambridge, MA

Pennington, J., Socher, R., & Manning, C.D. (2014). Glove: Global Vectors for Word Representation. In EMNLP, 14, 1532-1543. doi=10.1.1.671.1743

Penprase, B.E. (2018) The Fourth Industrial Revolution and Higher Education. In: Gleason N. (eds) Higher Education in the Era of the Fourth Industrial Revolution. Palgrave Macmillan, Singapore. https://doi.org/10.1007/978-981-13-0194-0_9

Pilászy, I., (2005). Text Categorization and Support Vector Machines. Proceedings of the 6th International Symposium of Hungarian Researchers on Computational Intelligence.

Ramasundaram, S., & Victor, S., (2010). Text Categorization by Backpropagation Network. International Journal of Computer Applications, 8(6). Doi: 10.5120/1217-1754

Raschka, S., Julian, D., & Hearty, J., (2017) Deeper Insights into Machine Learning.

Ravid, S. A. (1999). Information, blockbusters, and stars: A study of the film industry. Journal of Business, 72(4), 463–492

Rosenblatt, F., (1958). The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. Cornell Aeronautical Laboratory, Psychological Review, 65 (6), 386–408. doi:10.1037/h0042519

Santos, C., & Gatti, M. (2014). Deep convolutional neural networks for sentiment analysis of short texts. In Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, 69–78.

Sawhney, M. S., & Eliashberg, J. (1996). A parsimonious model for forecasting gross box-office revenues of motion pictures. Marketing Science, 15(2), 113–131

Sharda, R., Delen, D. (2006). Predicting box-office success of motion pictures with neural networks. Expert Systems with Applications 30(2), 243–254. https://doi.org/10.1016/j.eswa.2005.07.018

Simard, P., LeCun, Y., & Denker, J., (1993). Efficient pattern recognition using a new tranformation distance. Neural Information Processing Systems, 5, 50-58.

Simonoff, J., & Sparrow, I. (2000). Predicting movie grosses: Winners and losers, blockbusters and sleepers. Chance-Berlin Then New, 13(3), 40. https://doi.org/10.1080/09332480.2000.10542216

Siolas, G., d'Alché-Buc, F., (2000). Support Vector Machines based on a Semantic Kernel for Text Categorization. IEEE-INNS-ENNS International Joint Conference on Neural Networks, 5. DOI 10.1109/IJCNN.2000.861458

Sivanandam, S. N., & Deepa, S.N., (2011). Principles of Soft Computing, second edition. Wiley India Pvt. Ltd

Sochay, S. (1994). Predicting the performance of motion pictures. The Journal of Media Economics, 7(4), 1–20

Stone, M., (1974). Cross-validatory choice and assessment of statistical predictions. Journal Royal Statistic Society, 36(2), 111–147

Taira, H., & Masahiko, H., (1998). Text categorization using support vector machines. In IPSJ SIG-NL, 98(128-24),173-180. American association for artificial intelligence.

Vapnik, V., Boser, B., & Guyon, I., (1992). A training algorithm for optimal margin classifier. In Fifth Annual Workshop on Computational Learning Theory, 144-152.

Wallace, W., Seigerman, A., & Holbrook, M. (1993). The role of actors and actresses in the success of films: How much is a movie star worth? Journal of Cultural Economics, 17(1), 1–27.

Wang, J. (2017). Information Extraction from TV Series Scripts for Uptake Prediction. Retrieved from http://aut.researchgateway.ac.nz/handle/10292/10968

Zhang, W., & Skiena, S. (2009). Improving Movie Gross Prediction through News Analysis.Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, 01, 301-304. https://doi.org/ 10.1109/WI-IAT.2009.53

Zhang, X., Zhao, J., & Lecun, Y., (2015). Character-level Convolutional Networks for Text Classification. Neural Information Processing Systems, 28

Zufryden, F. S. (1996). Linking advertising to box office performance of new film releases: A marketing planning model. Journal of Advertising Research, 29–41