Movie Review Sentiment Analysis

Mykell Spencer
Computer Science Graduate
Florida Polytechnic University
Lakeland, Florida, United States
mspencer8054@floridapoly.edu

Andrey Cuevas-Andreev
Computer Science Graduate
Florida Polytechnic University
Lakeland, Florida, United States
acuevasandreev8490@floridapoly.edu

Eric Weakley
Computer Science Graduate
Florida Polytechnic University
Lakeland, Florida, United States
eweakley1379@floridapoly.edu

I. Introduction

For over a century, movies have dominated the entertainment industry because of the curiosity, imagination, and discourse they can spark among individuals. Throughout the Internet age, researchers have had access to enormous amounts of data regarding movies through social media, critical reviews, and websites such as IMDB. These platforms enable people to share their opinions and reviews with others, creating a snowball effect on accessible data. Discussions offer valuable insight into how audiences perceive films over time. The potential for mixed reviews makes the idea for sentiment analysis, the computational classification of expressed opinions, a favorable process for analyzing these perceptions.

From the wealth of data across the internet, this proposal aims to investigate how sentiment analysis of movie reviews from various sources can be utilized to predict box office performance and assess how sentiments evolve. Datasets from IMDB, Rotten Tomatoes, and X, formerly known as Twitter, will be used to apply a broader range of data to minimize bias towards one form of review. Correlations between a film's commercial success and overall sentiment classification will be analyzed. We will also examine the role of influential reviewers in disproportionately or otherwise shaping public opinion. Through this proposal, we will gain a deeper understanding of how the sentiments of individuals and their posts reflect and predict box office outcomes.

II. Motivation

This research is driven by the need to understand how online reviews by critics and social media influencers affect movie success. Platforms like IMDB, Rotten Tomatoes, and X offer vast amounts of user-generated data that, if analyzed effectively, could help predict box office outcomes before a release. However, the evolving nature of sentiments over time, especially before and after a film's release—remains under-explored. We aim to provide valuable insights into the industry's ebb and flow by studying these temporal sentiment shifts and their correlations with commercial performance.

Furthermore, this research seeks to address how the degradation of intellectual property (IP) value can occur when poorly received films are released. Negative audience reception can severely impact a franchise's financial future. Understanding the influence of top critics and how they shape public opinion is critical to predicting a film's success alongside the vocal general populace. This project bridges the gap between audience sentiment and box office performance, offering studios a more data-driven approach to evaluating film outcomes and steps to take for a prosperous future of IPs.

Finally, user reviews can give us a unique insight into the current demographic of people who engage with movies. By understanding their desires and how those desires affect the profitability of movies, it is possible to adjust the focus of future works in the medium to improve both customer experience and corporate profits.

III. Literature Review

Over the last two decades, studies of the relationship between reviews and box office performance have acquired considerable attention. Yang, Xu, and Tu [1] explore aspect-level sentiment analysis to predict the performance of the box office, demonstrating the value of specific features of their reviews, such as acting or the plot of the movie. This has value but does not assist in assessing overall sentiment from users. Their research highlights how certain movie features can be the main driving force behind its commercial success. Still, they do not fully expand upon the role of influential critics and reviewers in directing user sentiment. This presents a research gap in understanding how prominent individuals impact the perception of movies, a key component our project seeks to tackle. Similarly, Ya-Han et al. [2] examined customer reviews over five years from 2009 to 2014. They found a strong correlation between online sentiment and outcomes at the box office. However, the study does not delve into the evolution of said sentiment. This gap has been explored by Hao et al. [3], who focused on dynamic reviews but again overlooked the roles of influencers.

Temporal, or time-based, sentiment has been further researched by investigating the role of temporal dynamics in product reviews [4], though the study focused more on products than films. The concept of a time series proves very useful to the goals of this project. Further steps were taken with the development of a spatial-temporal model [5] for predicting the success of movies. Utilizing posts, replies, and retweet data from Twitter, the model displays changing sentiments over time and geographic regions. Deep learning sentiment analysis has seen significant advancements. Comprehensive Attention Recurrent neural network models combine attention mechanisms and recurrent structures to enhance sentiment classi-

fication [6]. A comparative study on deep learning sentiment analysis models such as Bidirectional Encoder Representations from Transformers (BERT) to better understand the context words are written in [7]. More neural networks, such as Long Short-Term Memory (LSTM) [8], polarized classifications for reviews. Although these papers emphasize improvements in classification, they do not link the sentiment to the success or failure of movies. With the application of the deep learning models and temporal sentiment, our analysis will directly connect to box office performance.

Analysis of time-based data regarding the evolution of sentiment over time remains relatively uncharted. Krauss et al. [9] highlight sentiment's critical role in movie success. Pappala [10] developed a recommendation system that was sentiment-driven. These works present a gap in time series analysis. Krauss' study lacks focus on sentiment change over time, an essential component in audience reception, while Pappala did not consider fluctuations in sentiment throughout the lifecycle of a film. Other studies [11, 12] recognize the variance of sentiment classification but lack a comprehensive time-based analysis. Cheng and Yang [13] also integrate sentiment analysis with economic modeling, but it has the same faults of analyzing static data. Collectively, these works tend to focus on that static data, missing opportunities to examine how reviews and opinions alter before and after the release of a film. Our project will aim to apply a time-series analysis to our datasets.

IV. Proposed Approaches

Our project aims to create a clearly defined framework for reading and predicting how different sentiments, from user-generated reviews to professional reviewers, affect the box office performance of released movies. The increasing availability of datasets through these movie-review websites, such as the ones stated in the Introduction and consumer sentiment, could show us how to predict success and failure before and after the release of specified movies. By attempting to apply Natural Language Processing (NLP) techniques such as VADER or TextBlob, the research data should be able to use mixed reviews of movies to forecast success or failure in films. The goal of bridging the gap between audience opinion, professional reviewers, and measurable commercial outcomes can revolutionize the sentiment of films over time. Data from various publicly recognized movie-reviewing websites were compared to public perception. This comprehensive analysis should give readers a deeper insight into how public and commercial opinions project a film's life cycle.

Furthermore, we can detect trends in consumer preferences and source relevance by examining shifts in these reviews over time. Changes in their quantity can reflect a changing engagement with the industry, changes in their aggregate sentiment can reflect shifting perceptions about the industry, and changes in their relevance for overall profitability can reflect the market's reliance on these tools for determining which movies to watch. Using a series of graphs to present this data in a human-readable form, trends like these should present themselves and initiate discussion about their potential causes and effects.

V. Planned Experiments

A. Data Collection

We will source the metadata of movies from Box Office Mojo, a trusted site that has been utilized for similar research in the past.[14,15] We will use it to acquire features like release date, the number of theaters the movie was released to, and its profitability over time. We will then combine that data set with a sentiment analysis of movie reviews taken from several review sites, like Rotten Tomatoes, IMDb, Yahoo, and Twitter. This data will be standardized for the number of theaters where each movie is released to minimize bias based on audience accessibility. This dataset can be further improved by refining our methodology for sentiment analysis. In addition to a generic sentiment for each review, we can apply aspect-level analysis to divide the sentiment between various aspects of the movie, like plot and cast. This will give a more complete picture of keywords an audience member may look for, rather than assuming they weigh all reviews solely on their overall sentiment.[15]

B. Data Analysis

Data will be divided into groups based on a movie's year of release. Then, we will analyze the correlation between the quantity and aggregate sentiment of reviews and the movie's profitability. Each review site will be analyzed separately, and the accuracy with which they can predict movie success will be noted. That data will then be used to answer several questions.

1) What is currently the most useful review source for overall product success?
2) How has the relative importance of various review sources changed over time?
3) How do expectations for reviews vary between sources?
4) How have audience expectations for reviews varied over time?

To answer question one, we will first take the R-squared value of the relation between overall movie sentiment and the movie's profitability. We will take the same value from the relation between the number of comments and the movie's profit. Strong positive values concerning sentiment imply that the opinions of a particular source matter greatly to the average moviegoer, and strong negative values imply a source has lost the trust of its consumer base and should be avoided. Concerning the number of comments, a strong positive R-squared value would imply that high levels of engagement on a source drove a movie's success. Alternatively, a strongly negative one would likely result from poor-quality movies receiving disproportionate negative reviews. This hypothesis can be confirmed by comparing any strong negative correlations to the overall sentiment value, and any trends that do not follow this pattern will be noted in our conclusion. In the case of either sentiment or quantity, weak values imply that a source is unimportant in driving sales for movies.

To answer question two, the experiment to answer question one will be repeated over at least a dozen years. The absolute value of the R-squared values of each site's sentiment accuracy over time will then be illustrated together with a line graph, and the same will be done for the absolute value of their quantity accuracy. This data will help visualize a source's relative relevance, and any significant jumps or drops in these values can invite future exploration into external factors that could explain the phenomenon. If there are sufficient samples of sources with both positive and negative correlations, we can then group those sources and do a feature analysis to determine the factors that could cause that differential.

To answer question three, we will use aspect-level analysis to break down each comment in a given time frame into its parts. For example, since a person's complaints about a movie's plot and complaints about a movie's acting are different, this method would split an overall review into those component categories. We then evaluate the R-squared value of each component compared to the movie's profit and tabulate those for every source. We will also tabulate the ratio between the individual components' R-squared value and the overall sentiment's R-squared value. If these ratios remain constant between all sources, it would imply that every movie review source could expect a common distribution of their users' desires when observing reviews. However, if there are outliers or the ratios have a random distribution, that would imply segmentation in the demographics of people who use different movie review sources and could be used to contextualize previous research with the studied sources.

To answer question four, we would take the data from question 3 and generate it for every time frame we have data for. We would then make a line graph for each review aspect and map every site's reviews to a line on that graph. If multiple sources show similar changes in the predictive value of a component, it would imply that a general movie-going audience's tastes in that component are shifting. The same goes for multiple sources that show stagnation in a component. However, suppose different sources have different changes in predictive validity. In that case, we will map how closely a particular component's accuracy change could be mapped to the change in accuracy for a site overall. If this is performed and any strong correlations reveal themselves in the data, we could find generic flags to look for when reviewing the usefulness of any movie review source.

VI. Progressed Methodology

A. Data Pre-processing

1) Importing Libraries and Data Collection: Various Python libraries are imported to perform data, sentiment, and statistical analysis. Some libraries will be utilized to handle datasets and visualization. The *Numpy* library is imported to perform data manipulation on our datasets along with mathematical functions operating on arrays and matrices. The Pandas library offers data manipulation for time-series, a useful package for sentiment analysis on movies over extended periods of time. The *seaborn* and *matplotlib* libraries are utilized for visual-based analysis of further regression, time-series, and sentiment analysis. The inclusion of the RE library is imported for regular expressions and prepares text data for analysis later down the processing pipeline. This package will help structure the unstructured text data, making text-based features easier to handle and analyze.

The *vaderSentiment* library's *SentimentIntensityAnalyzer* is employed to assess the sentiment of the textual data we receive in our datasets. The analyzer is mainly used for social-media text mining. It will be able to detect nuances in language, slang, emoticons, and acronyms; commonly used in user-generated text. It outputs a compound score that converts the text into a compound sentiment score on a negative-to-positive scale. Turning qualitative into quantitative data will invoke an easier understanding of how the text can be viewed as it can be visualized using plot-based libraries.

For regression modeling and classification, *traintestsplit* from the *sklearn.model selection* library is chosen to partition the data into training and testing sets. The sets are proportioned seventy-five percent to the training set and twenty-five percent to the testing set. The *LinearRegression* model from the *sklearn.linear model* will explore the relationships between engineered features as well as target features. Model evaluation will be performed by the *sklearn.metrics* package to display statistical metrics such as mean-squared error or the *r2score* to further explain what the data is illustrating.

For dataset collection, we have chosen four datasets so far: a box office [16], IMDB movies [17], and two rotten tomatoes datasets [18]; one is critic reviews, and the other is various movie reviews. The box office dataset provides approximately a decade of financial performance of movies, providing box office earnings, necessary for measuring commercial success. The IMDB movies dataset contains data from the IMDB platform, including ratings, directorial credits, and viewer sentiment. This will offer a more subjective measure of a movie's reception. The rotten tomatoes critic dataset provides information from the critical perspective on movies. This pairs nicely with the other dataset that discusses movie reviews because we can see how subjective opinions differ and change between two different groups of responses.

2) Data Cleaning and Handling: The first major step of data pre-processing is to inspect the datasets to provide information on their features and observations. Missing values within data can cause errors or interfere with modeling to skew results. It is important to clean this data and there are several cleaning methods: interpolation, before-and-after mean, or simply removing the data entirely. The latter option is not helpful as it would require the possible removal of other important feature data. Features that consist of text reviews will not be altered if there is no information as we can count that as a no-response review, should they exist. For numerical values, data will be interpolated with respect to values within the feature so as not to act as an outlier for our data. The *FindingNulls* function will assist in dynamically finding null values across the used datasets and print the sum of all of these null entries, respectively. It will also display the rows

that consist of missing data that can be converted to another CSV file for later use. The *HandlingNulls* function preserves null values located within text-based features, but numerical null entries are interpolated followed by forward and backward filling. Data continuity will be maintained without introducing new biases.

When handling data that is based on the reviews of individuals, ensure that multiple counts of review from the same person around the same time frame are nullified. Multiple reviews from a single entity can skew our data when it comes to sentiment analysis and classification, heavily inflating one side from multiple reviews. A one-to-many relationship will be removed from the dataset. The *FindDuplicates* function will search the datasets for duplicate entries and transmit them to a CSV file to confirm duplicates. The function, if duplicates are found, will choose to keep the first review and remove the rest. Unique data points will give us better trends when regressing and classifying the data.

Unstructured textual data poses a challenge when it comes to consistency. It can take various formats, like emails and social media posts, differing in structure and complexity. To maintain consistency and fairness to all datasets, textual data is converted to lowercase, removes non-alphanumeric characters, and corrects spacing. This is handled via the *CleanTextual Data* function. Normalizing the text is also necessary for natural language processing tasks that are implemented in sentiment analysis as it impacts performance and errors created by minute character formatting.

3) Feature Engineering: Feature engineering a dataset requires the use of correlation matrices, graphs that show how correlated two or more features are to one another. At higher dimensions, this can be difficult to visualize efficiently. One major issue arises when trying to correlate numeric and text data, it is impossible to do so unless we were to readily quantify the text data to a scale. Doing that would require sentiment analysis which would not be appropriate as our feature relationships have yet to be studied. Given the conflict, two separate matrices will be formed to test text and numerical data separately to a target feature. Here we can determine strong relationships between our target and features of both labels.

For the numeric data, a simple correlation matrix function will provide correlation graphs between all numeric features. This illustration will visibly show how strong or weak a certain feature is to others. For textual data, heatmaps from the *seaborn* package will display warm and cool colors to indicate the strength of the correlation. After choosing which features to proceed with, we will then divide each by the target feature and concatenate them into their respective datasets. Checks will be made to ensure that the division of these features did not provide any infinitesimal values within our datasets. Further data interpolation will remove any infinitesimal data without compromising the rest of the set.

With the engineered features concatenated, the datasets can now form a correlation matrix using the *data.corr()* function. The values will then be sorted in ascending order to find which features still consist of the highest correlations. The results that show the most interest with be utilized for further analysis using regression and classification.

B. Custom Data Set Creation

When comparing metrics across different websites, the best way to eliminate bias is to create custom sets based on the same criteria rather than obtaining premade data sets. Thus, a lot of time was spent attempting to complete this objective. While there were some successes, this process was mainly useful in revealing many difficulties that kept previous research from making a comprehensive review of movies from different sources.

For creating web scraping applications in this domain, the Scrapy Python library was cited in multiple widely-used applications [20], [21], [22]. It works by creating a spider to automatically move through a website, saving data from its search to a custom Python object, and then using a pipeline to convert the data in the object to a desired form. It allows for extensive customization of every step in this process, from a dozen integrated functions to assist in navigating the web page to compatibility with Python's CSV and JSON libraries to simplify data storage.

The most successful implementation of Scrapy was with BoxOfficeMojo analytics. Since the search function of this website can be limited by year and ordered by revenue, it is relatively simple to get a list of movie links to review. For each link, details about its associated movie are stored in a single table with relatively little superfluous formatting, which again makes it easy to programmatically obtain data. However, even though getting basic metrics was successful, BoxOfficeMojo does not contain movie reviews, one of the primary metrics under review, which made using it in this paper difficult.

The IMDb database was far less successful. Like BoxOfficeMojo, it had a search feature that could be limited by year and sorted by popularity. It even surpassed BoxOfficeMojo because it had access to many user reviews and budget estimates for movies whose budgets were unconfirmed, making it an invaluable source for our research. However, the data we wished to retrieve was contained in a table with a large quantity of styling and many inconsistent fields. This meant a high degree of difficulty in creating an algorithm capable of obtaining the relevant data. However, once the algorithm was created, a second problem was observed: IMDb has several restrictions to prevent unlicensed web scrapers from accessing their advanced search web page. This made previous efforts to parse data from individual web pages irrelevant and forced us to use a pre existing data set, as manually reviewing 100 websites per year of study was infeasible.

Rotten Tomatoes posed a problem for a different reason. It has sparse information about its movies, and its search functionality does not have a feature to limit results by year. There is a separate web page that lets the user sort movies by popularity, but that page only displays movies made within the last year. These challenges would make designing a web scraper in the same method as the other two functionally

impossible, and combined with the previous failure on the IMDb database, we decided to use a pre existing dataset for Rotten Tomatoes as well.

Despite the many failures faced here, several important details were revealed from them. First, the widespread use of BoxOfficeMojo in movie research is likely more due to its compatibility with web scraping applications than its accuracy or comprehensiveness. Second, the reason so many researchers observed a gap in comparing data across different sources is that every source contains a unique challenge for data retrieval. This makes the process of creating data sets that are balanced across multiple sources infeasible for most studies, and that would be necessary to guarantee accuracy.

C. Sentiment Analysis and Sourcing

Regarding data processing, the most complicated step is sentiment analysis. There are several libraries capable of doing this, from the "Twitter-roberta-base-sentiment" model that has been trained on millions of tweets to the "Distilbert-base-uncased-emotion" model, which claims to be able to detect nuanced emotions. Still, the most effective method would be to train a model of our own on movie-related texts. There is a simple method of doing this outlined by a Hugging Face blog [19], but due to the unreliability of the source, it will need to be tested before implementation. Testing is possible by running a random selection of positive and negative reviews through a model designed to its specifications and then manually reviewing the output compared to outputs of other, pre-built models.

In the best-case scenario, a custom model will be able to put a numerical value to both overall sentiment and categorical sentiment. 'Overall' refers to the entire comment, and 'categorical' refers to the broad noun a particular sentiment is associated with. The numerical value would be the model's confidence that a statement is either positive or negative. However, as this is a complex process with no publicly available library to support every piece of functionality, complexity constraints may force a compromise on categorizing sentiment based on specific objects.

Even in the worst case, though, this model can get a distribution of audience perceptions about various movies over time, along with a numerical representation of that perception from sources that include a numeric rating. These aggregate perceptions of movies can be compared over time, across different review sites, and between different genres. By doing this, data will be generated about gaps commonly found in research in this field. Here are some important questions this data has the potential to answer:

- What are the similarities and differences between communities on different review sites?
- How has public perception of various genres changed over time?
- How has the public perception of the movie industry as a whole changed over time?

The NLP with the best documentation on how to create a custom sentiment analysis model for movie reviews is DistilBERT, a derivative of the Bidirectional Encoder Representations from Transformers (BERT) model that prioritizes efficiency. However, other tools, like NTLK, TensorFlow Text, and CoreNLP show promise in completing similar sentiment analysis tasks and could theoretically be applied to movie reviews. If time and complexity allow, we will compare the accuracy of these models in the domain of review classification to generate more complete and accurate classifications, though that is a tertiary objective and not the primary goal of this project.

1) Feature Engineering Sentiments: With the implementation of sentiment scores to the datasets, further analysis needs to be completed to understand which features can explain the data's behaviors over time. Using a correlation plot, five feature relationships were chosen as the top five were filtered: Tomato Meter Count Per Critic Count, Meter Count Per-Fresh Count, Audience Count Per-Status Count, Tomato Meter Rating Per-Audience, and Tomato Meter Count Per-Audience Rating. These five features will be plotted using the sentiment scores as the target dependent variable over the independent x-axis variable of time. The intervals between measures will be annual. The change in sentiment over time will be studied to understand which features explain the changing behavior of movie sentiment the most.

D. Convolution Matrix Implementation

Whereas we would have been able to properly analyze the connections in our sample, we found—in testing along the way—that using the larger dataset as one matrix would lead to catastrophic memory and processing issues. Memory crashes happen when the system during processing realizes it's unable to allocate sufficient memory. Since we need both our textual and numerical features mixed in and have constrained processing power/hardware available, the only option is to split the dataset for processing. In order to determine if small groups can generate a correlation matrix, we intend to take an iterative sample of 1/10000 of the data. Therefore, using 'pandas.sample()' in Python to subset our presumed larger database, we will down sample to 0.01% of the data while still maintaining statistical significance of the sample. We will assess computation efficiency based upon this sample and go up and down from there. Exploratory data analysis was performed via Python libraries with pandas, matplotlib, and seaborn. We analyzed the correlation matrix of the subset via a function created that essentially timestamped the correlation to see if it was scalable and also used 'pandas.get_dummies()' and '.corr()' to convert categorical to numerical data and find correlation between features on a pairwise level. We timestamped our findings to see if it was scalable as well. For the 1/10000 sample, the dummies and correlation took X seconds rendering not only computations scalable but also convenient and quick from this smaller sample for preliminary exploratory investigation. This will be useful for the larger sampled datasets in future runs when it is fractionalized in larger amounts 1/500, 1/250, etc., until the perfect sample fraction is found that reduces time and memory. Then from the

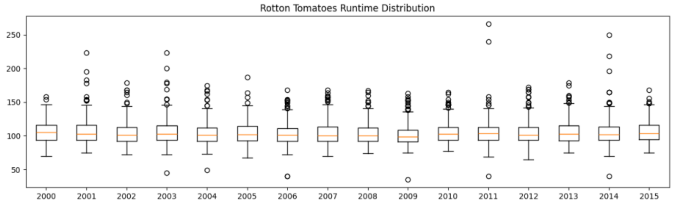

Fig. 1. Annual distribution of movie run times on Rotten Tomatoes.

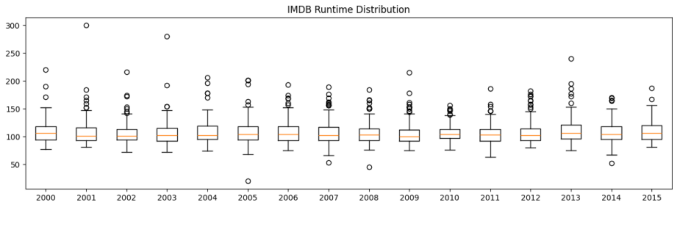

Fig. 2. Annual distribution of movie run times on IMDb.

memory need of the 1/1000 sample, since we have a constant, we can multiply to see what would be needed for the larger samples. This helps with the logistics in future runs when fractionalizing to allow future runs to process the datasets that have already been fractionalized in parallel. Fractal means that each fractionalized dataset can run independently and simultaneously generate their own correlation matrices, which is beneficial for distributed systems/cloud. This feedback loop solves the issue of having to experiment on the full dataset as it does so with what little resources are available. This maintains memory restrictions at a manageable level and provides a chronological tutorial of re-experimenting to assess the whole dataset.

VII. Results

Both the Rotten Tomatoes dataset and the IMDB dataset had common elements that could be compared. As a control, movie run times are compared to each other. Between 2000 and 2015,

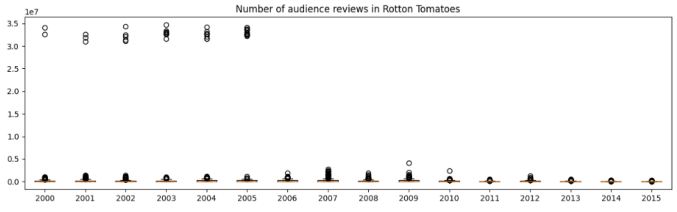

Fig. 3. Annual distribution of audience review quantity on Rotten Tomatoes.

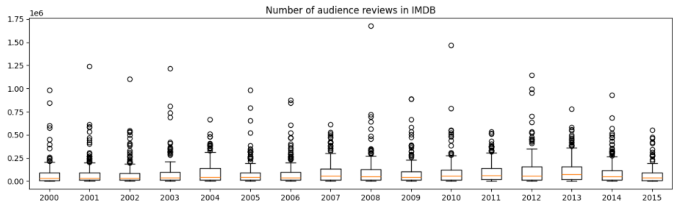

Fig. 4. Annual distribution of audience review quantity on IMDb.

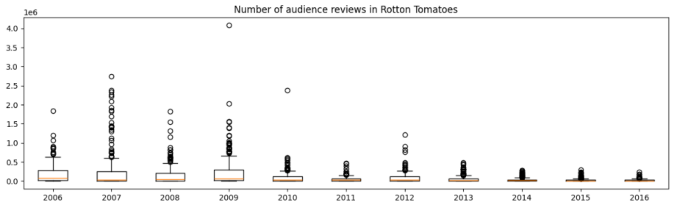

Fig. 5. Annual distribution of audience review quantity on Rotten Tomatoes, starting at 2006.

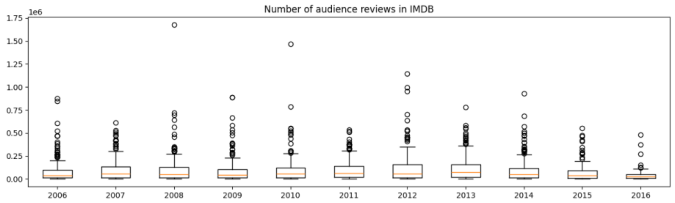

Fig. 6. Annual distribution of audience review quantity on IMDB, starting at 2006.

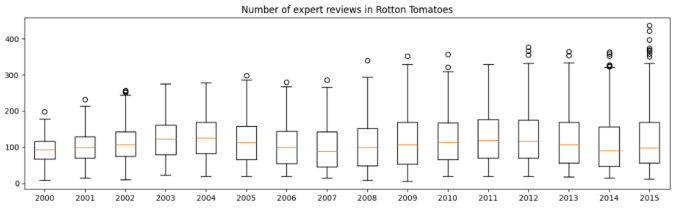

Fig. 7. Annual distribution of expert review quantity on Rotten Tomatoes.

Fig. 8. Annual distribution of expert review quantity on IMDb.

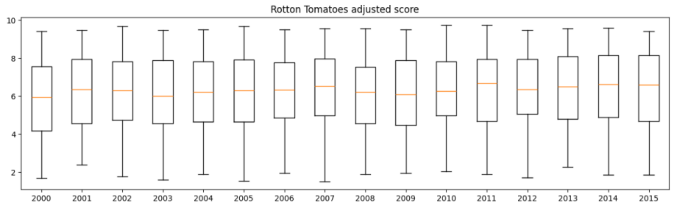

Fig. 9. Annual distribution of overall scores on Rotten Tomatoes.

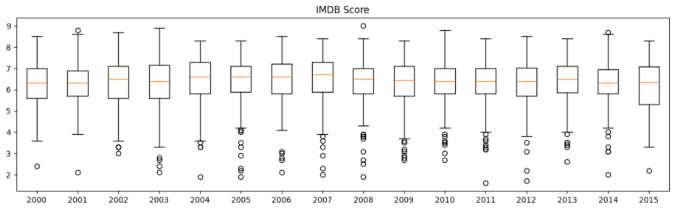

Fig. 10. Annual distribution of overall scores on IMDb.

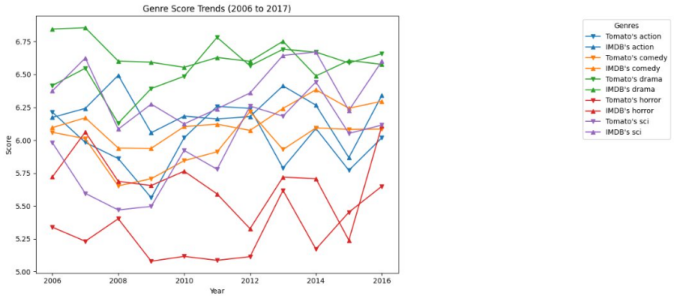

Fig. 11. Annual distribution of scores for each genre on each platform.

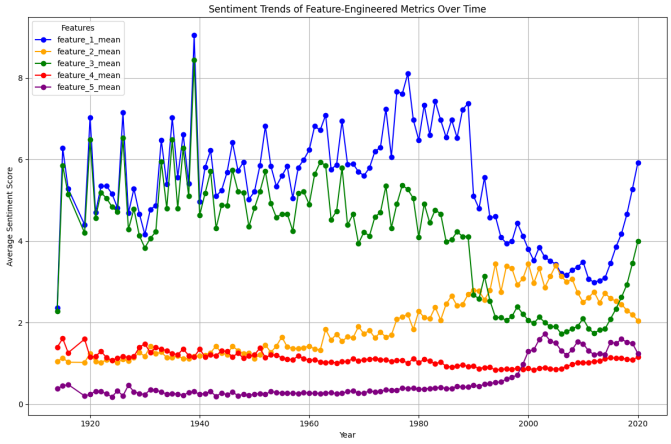

Fig. 12. Temporal trends of sentiment scores for feature-engineered metrics, derived from highly correlated feature pairs, illustrating variations in audience and critic reception over time.

the means and inter-quartile range of both data sets remained constant. This showed that both sites review similar types of movies, and the fundamentals of making movies changed relatively little over time. However, since the outliers of both datasets differ greatly, it can be assumed that the exact list of movies being reviewed have relevant differences.

Though relevant, they are not extreme enough to explain the differences found later in the analysis. For instance, when comparing the number of audience reviews on movies, Rotten Tomatoes had 5 years where a cluster of movies had 10 times more audience reviews than any of their peers. This trend is not seen in IMDb's statistics. The data gets more similar when these outlier years are removed, and the means of each interval of both data sets remain relatively close to zero, but the IQR and outliers of those sets are wildly different. They do both show a trend where the most famous movies get fewer comments over time, but IMDb looks more resilient to that trend than Rotten Tomatoes. While this is possibly because popular movies on their website have less reviews overall, it could als obe a result of IMDb retaining relevance for longer than Rotten Tomatoes has.

This distribution rule does not apply to expert analysis counts. Both have, on average, 100 experts analyzing each movie, and this has remained consistent from 2000 to 2015. That said, IMDb consistently gets more experts to review their top-performing movies than Rotten Tomatoes does, with it's peak being double that of Rotten Tomatoes'. Wether this trend is due to IMDb having a larger budget to insensitive professional reviewers to leave comments or if IMDb is just a larger cultural touchstone is hard to say.

The score distribution required more work to analyze, as Rotten Tomatoes and IMDb use different rating systems for their movies. To compensate for this, I averaged the user and critic scores on Rotten Tomatoes before min-max scaling it to have the same possible range of values as IMDb. After doing this, the means of both data remain steady between a 6 and 7, but IMDb has significantly more outliers and is far less likely to rate a movie extremely low or high compared to Rotten Tomatoes.

Dividing the scores by the genre of movie they were given to makes the differences even clearer. While dramas are generally rated highly and horror movies are generally rated poorly, no genre is consistent with either other genres or the same genre on another website. They have global minima and maxima at different years, and their slopes change seemingly at random.

The graph presented above illustrates the temporal trends of sentiment scores for five feature-engineered metrics, derived from the most highly correlated features identified within the Rotten Tomatoes Critic dataset. These engineered features, including ratios such as 'tomatometer-count' to 'tomatometer-top-critics-count' and similar such pairs, were selected to capture nuanced relationships between audience and critic dynamics. Each trend line represents the annual average sentiment score for one of these metrics, showcasing distinct variations over time. For example, 'feature-1-mean', which highlights the influence of top critics reviews, showing significant fluctuations across decades, reflecting shifts in critical consensus. The remaining features exhibit relatively consistent patterns, offering insights into the evolving reception of films by both critics and general audiences. This analysis highlights the potential of feature engineering in uncovering temporal trends and their implications in sentiment analysis.

VIII. Future Works

Given the problems outlined by the conclusions, future works should look at verifying results and finding solutions. Due to resource limitations, we were unable to construct custom data sets to check for differences between sources without confounding variables. There would be value in a study that confirmed our findings if the Rotten Tomatoes and IMDb datasets reviewed the same movies or had the same average reviews for each genre.

These data sets could also be solutions to the same problems they would be designed to study. There have been dozens of papers claiming hundreds of associations between audience perceptions and overall movie success. Creating a set of data that could test these conclusions across multiple sources, or that could contextualize them based on their source of origin, would be a valuable review tool. If we could create

a dataset that experimentally generates the same sentiments across multiple review sites, it could ensure future works do not fall into the pitfalls many past ones have.

For the annual distribution of scores, future efforts in applying this model to other datasets such as IMDB and BoxOffice would expand upon different features that cause sentiment scores change over time. Cross-references between datatset features can be ranked based on which features explain the most information about the behavior of the data.

IX. Conclusions

These results make it clear that movie data sets can heavily bias the results of a study that uses them. Since the means remained close for most comparable metrics, it may appear that they can be used interchangeably. However, since the outliers varied wildly, each dataset would generate quirks in their results that would be inconsistent with studies attempted with other sources. Even if the methods used are controlled for outliers, the relative density of different genres can cause the same inconsistencies. A data set with more dramas will be expected to skew positively and a data set with more horror movies will be expected to skew more negatively.

If applied to test mining, this can also lead to false associations within the data. If you build a model to gauge sentiment from movie reviews, that model is likely to shortcut phrases associated with different genres to different sentiments, even though the genre of the movie should not affect how positively critics view it. Many studies control for this by testing genres independently, but as shown in Figure 11, this method just ends up exacerbating differences between different data sets.

The temporal trends graph illustrates evolving sentiment trends from 1920-2020. Features 1 and 3 exhibit prominent inclines, showing increases of sentiment over time. In contrast, features 4 and 5 show a relatively horizontal graph throughout the century. This shows that these features may not have a significant contribution to the overall change in sentiment experienced by critics and audience members. Feature 2 display an interesting increase from 1980 on.

References

[1] G. Yang, Y. Xu, and L. Tu, "An intelligent box office predictor based on aspect-level sentiment analysis of movie review," Wireless Netw, vol. 29, no. 7, pp. 3039–3049, Oct. 2023, doi: 10.1007/s11276-023-03378-6.

[2] H. Ya-Han, W.-M. Shiau, S.-P. Shih, and C. Cho-Ju, "Considering online consumer reviews to predict movie box-office performance between the years 2009 and 2014 in the US," The Electronic Library, vol. 37, no. 6, pp. 1010–1026, 2018, doi: 10.1108/EL-02-2018-0040.

[3] Y. Hao, Y. Li, Q. Ye, and P. Zou, "Dynamic impacts of online reviews and other information sources on sales in panel data environment: Evidence from movie industry," in 2008 International Conference on Management Science and Engineering 15th Annual Conference Proceedings, Sep. 2008, pp. 493–500. doi: 10.1109/ICMSE.2008.4668961.

[4] P. Xia, W. Jiang, J. Wu, S. Xiao, and G. Wang, "Exploiting Temporal Dynamics in Product Reviews for Dynamic Sentiment Prediction at the Aspect Level," ACM Trans. Knowl. Discov. Data, vol. 15, no. 4, p. 68:1-68:29, Apr. 2021, doi: 10.1145/3441451.

[5] A. W. M. K. S. A. Wijekoon, T. C. Sandanayake, K. D. A. A. Jayawardena, A. L. Y. Buddhini, and U. K. D. G. S. Ariyawansha, "Spatio-Temporal Visualization Model for Movie Success Prediction Based on Tweets," in Proceedings of the 2017 International Conference on Information Technology, Singapore Singapore: ACM, Dec. 2017, pp. 227–231. doi: 10.1145/3176653.3176674.

[6] Y. Zhang, J. E. Meng, R. Venkatesan, N. Wang, and M. Pratama, "Sentiment classification using Comprehensive Attention Recurrent models," in 2016 International Joint Conference on Neural Networks (IJCNN), Jul. 2016, pp. 1562–1569. doi: 10.1109/IJCNN.2016.7727384.

[7] "Electronics — Free Full-Text — Sentiment Analysis Based on Deep Learning: A Comparative Study." Accessed: Sep. 08, 2024. [Online]. Available: https://www.mdpi.com/2079-9292/9/3/483

[8] "Sentiment Analysis from Movie Reviews Using LSTMs — IIETA." Accessed: Sep. 08, 2024. [Online]. Available: https://www.iieta.org/journals/isi/paper/10.18280/isi.240119

[9] J. Krauss, S. Nann, D. Simon, K. Fischbach, and P. Gloor, "Predicting Movie Success and Academy Awards through Sentiment and Social Network Analysis," presented at the 16th European Conference on Information Systems, ECIS 2008, Jun. 2008.

[10] S. Pappala, "Sentiment-Driven Movie Recommendation System: A Machine Learning Approach," vol. 6, p. 1, May 2024.

[11] A. Castillo, J. Benitez, J. Llorens, and X. (Robert) Luo, "Social media-driven customer engagement and movie performance: Theory and empirical evidence," Decision Support Systems, vol. 145, p. 113516, Jun. 2021, doi: 10.1016/j.dss.2021.113516.

[12] J. H. Lee, S. H. Jung, and J. Park, "The role of entropy of review text sentiments on online WOM and movie box office sales," Electronic Commerce Research and Applications, vol. 22, pp. 42–52, 2017, doi: 10.1016/j.elerap.2017.03.001.

[13] L.-C. Cheng and Y. Yang, "The Effect of Online Reviews on Movie Box Office Sales: An Integration of Aspect-Based Sentiment Analysis and Economic Modeling.," Journal of Global Information Management, vol. 30, no. 1, p. NA-NA, Jan. 2022, doi: 10.4018/JGIM.298652.

[14] T. Deng, "Investigating the effects of textual reviews from consumers and critics on movie sales," Online Information Review, vol. 44, no. 6, pp. 1245–1265, 2020, doi: 10.1108/OIR-10-2019-0323.

[15] C. Jiang, J. Wang, Q. TANG, and X. Lyu, "Investigating the Effects of Dimension-Specific Sentiments on Product Sales: The Perspective of Sentiment Preferences," Journal of the Association for Information Systems, vol. 22, no. 2, p. 4, Mar. 2021, doi: 10.17705/1jais.00668.

[16] skozilla, skozilla/BoxOfficeMojo. (Nov. 18, 2023). Python. Accessed: Oct. 2, 2024. [Online]. Available: https://github.com/skozilla/BoxOfficeMojo

[17] "IMDB Movie Analysis." Accessed: Oct. 3, 2024. [Online]. Available: https://www.kaggle.com/datasets/vikramchr/imdb-movie-analysis

[18] "Rotten Tomatoes - EDA." Accessed: Oct. 3, 2024. [Online]. Available: https://kaggle.com/code/stefanoleone992/rotten-tomatoes-eda

[19] "Getting Started with Sentiment Analysis using Python." Accessed: Oct. 23, 2024. [Online]. Available: https://huggingface.co/blog/sentiment-analysis-python

[20] V. Gupta, dojutsu-user/IMDB-Scraper. (Sep. 30, 2024). Python. Accessed: Nov. 26, 2024. [Online]. Available: https://github.com/dojutsu-user/IMDB-Scraper

[21] CodeMate TV, Scraping IMDB With Python 2024. without selenium!, (Jun. 03, 2024). Accessed: Nov. 26, 2024. [Online Video]. Available: https://www.youtube.com/watch?v=lXCNpC3ITLA

[22] Y. Jeong, "Scraping Box Office Info with Scrapy," Analytics Vidhya. Accessed: Nov. 26, 2024. [Online]. Available: https://medium.com/analytics-vidhya/scraping-box-office-info-with-scrapy-f23f1f2d684f