ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ
ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Πειραματική Ανάλυση Συναισθήματος σε κριτικές βιντεοπαιχνιδιών.

Δημήτριος Σταθόπουλος

Επιβλέπουσα Καθηγήτρια:
Μαρία Χαλκίδη, Αναπληρώτρια Καθηγήτρια

ΠΕΙΡΑΙΑΣ

Ιούλιος 2023

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Experimental Sentiment Analysis on Video Game Reviews.

Δημήτριος Σταθόπουλος
Α.Μ.: E18151

ΠΕΡΙΛΗΨΗ

Η ανάλυση συναισθημάτων διαδραματίζει κεντρικό ρόλο στην κατανόηση των αντιλήψεων των πελατών στον δυναμικό κόσμο της βιομηχανίας των video games. Σε αυτήν τη μελέτη, ξεκινάμε μια εμπεριστατωμένη συγκριτική ανάλυση πέντε κορυφαίων τεχνικών επιβλεπόμενης μάθησης: Μηχανές Υποστήριξης (SVM), Λογιστική Παλινδρόμηση, Ακραία Ενίσχυση της Κλάσης (XGBoost), Τυχαίο Δάσος(Random Forests) και Δικτυακά Νευρωνικά Δίκτυα Συνέλιξης (CNNs), με σκοπό να διαπιστώσουμε την αποτελεσματικότητά τους στην κατηγοριοποίηση των συναισθημάτων σε αναθεωρήσεις video games. Ο σκοπός της έρευνάς μας είναι να παρέχει μια περιεκτική και επεξηγηματική εξέταση αυτών των αλγορίθμων, ρίχνοντας φως στα πλεονεκτήματά τους, τα μειονεκτήματά τους και τις πρακτικές επιπτώσεις τους. Διαβάζοντας προσεκτικά ένα ποικίλο σύνολο δεδομένων αναθεωρήσεων video games, αξιολογούμε την απόδοσή τους στον ακριβή καθορισμό των νοημάτων, εκτιμώντας τη δυνατότητά τους να επηρεάσουν αποφάσεις που βασίζονται σε δεδομένα. Μέσα από αυτήν την έρευνα, στοχεύουμε να παράσχουμε σημαντικές εισηγήσεις στους ενδιαφερόμενους φορείς στη βιομηχανία των video games, επιτρέποντάς τους να ενισχύσουν την ικανοποίηση των πελατών, να ενδυναμώσουν την πιστότητα στο brand και να προωθήσουν την διαρκή ανάπτυξη της επιχείρησής τους.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Ανάλυση Συναισθημάτων (Sentiment Analysis), Επιβλεπόμενη Μάθηση (Supervised Learning), Αλγόριθμοι Μηχανικής Μάθησης (Machine Learning Algorithms), Μηχανές Διανυσμάτων Υποστήριξης (SVM), Λογιστική Παλινδρόμηση, Ακραία Ενίσχυση της Κλάσης (XGBoost) Τυχαίους Δάσους (Random Forests) και τα Συνελικτικά Νευρωνικά Δίκτυα (CNNs), Αξιολόγηση Αλγορίθμων (Algorithm Evaluation), Βιομηχανία των Video Games, Ανάληψη Αποφάσεων (Data-Driven Decision-Making).

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Ανάλυση συναισθημάτων, Επιβλεπόμενη μάθηση, Αλγόριθμοι μηχανικής μάθησης, Αξιολόγηση αλγορίθμων, Βιομηχανία των video games, Ανάληψη αποφάσεων.

ABSTRACT

Sentiment analysis plays a pivotal role in understanding customer perceptions within the dynamic gaming industry. In this study, we embark on an in-depth comparative analysis of five prominent supervised machine learning techniques: Support Vector Machines (SVM), Logistic Regression, Extreme Gradient Boosting (XGBoost), Random Forests, and Convolutional Neural Networks (CNNs), to unravel their efficacy in sentiment classification within video game reviews. Our research aims to provide a comprehensive and explanatory examination of these algorithms, shedding light on their strengths, weaknesses, and practical implications. By scrutinizing a diverse dataset of video game reviews, we assess their performance in accurately capturing nuanced sentiments, gauging their potential to influence data-driven decision-making. Through this exploration, we aim to equip stakeholders in the gaming industry with valuable insights, empowering them to enhance customer satisfaction, foster brand loyalty, and drive sustained growth. Join us in this endeavor as we delve into the intricacies of sentiment analysis and illuminate the path to informed decision-making within the gaming realm.

SUBJECT AREA: Sentiment Analysis, Machine Learning Techniques, Video Game Reviews, Data-Driven Decision-Making, Gaming Industry.

KEYWORDS: Sentiment Classification, Machine Learning Algorithms, Video Game Reviews, Data Analysis, Gaming Industry, Support Vector Machines, Logistic Regression, Extreme Gradient Boosting, Random Forests, Convolutional Neural Networks (CNNs).

CONTENTS

LIST OF FIGURES

List of Tables

PROLOGUE

In today's digital era, where online interactions and user-generated content have become increasingly prevalent, customer reviews have emerged as a valuable source of information for businesses across diverse industries. Among these industries, the gaming sector stands out as a highly dynamic and competitive field where understanding customers' emotions, preferences, and feedback is crucial for success. In this landscape, businesses within the video game industry are recognizing the importance of data analysis to gain a competitive advantage and maximize their profits. By leveraging the power of sentiment analysis and data-driven insights, businesses can make informed decisions, improve their products, and enhance the overall gaming experience for their customers.

The primary objective of this project is to conduct a rigorous scientific analysis using real-world data, with a specific focus on accurate sentiment analysis within the gaming industry. Through a systematic exploration of various models designed to effectively capture and quantify sentiment related to individual or multiple products, this project aims to contribute to the advancement of sentiment analysis techniques tailored for the gaming sector.

The outcome of comparing different models will provide businesses with valuable insights into which models perform best in classifying sentiment within gaming reviews. Ultimately, this project aspires to empower businesses operating in the video game industry to make data-driven decisions that can enhance customer satisfaction, strengthen brand loyalty, and drive sustainable business growth.

1 INTRODUCTION

Sentiment analysis, also known as opinion mining, is a process that involves analyzing text to determine the sentiment or emotional tone conveyed by the author or speaker. However, this task is not without its challenges. One significant hurdle is the contextual understanding of language. Words and phrases can have different meanings based on the context in which they are used. For example, the word "sick" can refer to an illness or can be used to express admiration in colloquial language. Understanding the context is crucial for accurate sentiment analysis.

Another challenge is the presence of negation and sarcasm in text. Negations can invert the sentiment of a statement. For instance, the sentence "I don't dislike this product" may seem positive at first glance, but it conveys a negative sentiment. Sarcasm further complicates sentiment analysis, as sarcastic expressions often express sentiments opposite to their literal meaning. Interpreting such nuances accurately requires advanced models and techniques.

Additionally, sentiment analysis needs to account for domain-specific sentiment. Sentiments can vary across different domains, such as product reviews, movie critiques, or political speeches. A sentiment analysis model trained on movie reviews may not perform as effectively when applied to analyzing financial news. To achieve accurate results, models must be trained on data specific to the domain they will be applied to.

Language ambiguity is another challenge faced in sentiment analysis. Languages often contain ambiguous words, phrases, or idioms that can make it difficult to assign a precise sentiment label. These ambiguities can lead to misinterpretations and affect the overall accuracy of sentiment analysis.

Despite these challenges, sentiment analysis has found widespread use across various industries. For businesses, sentiment analysis provides valuable insights into customer opinions, allowing them to understand how their products, services, or brands are perceived. Social media monitoring helps companies assess customer satisfaction, manage their online reputation, and identify emerging trends. Customer feedback analysis enables businesses to quickly address customer concerns and improve their offerings.

Beyond business applications, sentiment analysis plays a role in political analysis by providing insights into public opinion on candidates, policies, and campaigns. It is also used in market research to analyze consumer sentiments, preferences, and purchase intentions. Furthermore, sentiment analysis is employed in customer service automation, where it helps chatbots and virtual assistants understand and respond to customer sentiment during interactions.

In the financial sector, sentiment analysis is utilized to analyze news articles, social media feeds, and other textual data to assess market sentiment. This information can be used to predict stock prices, make investment decisions, and manage financial portfolios.

In summary, sentiment analysis continues to evolve and find new applications across diverse fields. By analyzing and categorizing sentiments expressed through text, it provides valuable insights into human emotions and opinions, allowing businesses, organizations, and individuals to make informed decisions and take appropriate actions.

The field of Natural Language Processing (NLP) and Machine Learning has made significant progress in recent years, allowing researchers to extract valuable information from large volumes of text. One area where these techniques have proven particularly useful is sentiment analysis and topic extraction from customer reviews. In this project, we focus on the application of NLP algorithms and machine learning for the analysis and understanding of sentiment expressed in customer reviews for video games.

The primary goal of our project is to systematically assess and compare the performance of five distinct supervised machine learning techniques for sentiment analysis. We will employ these algorithms on a dataset comprising video game reviews. Through this comprehensive evaluation, our aim is to identify and select the most effective algorithm for accurately classifying sentiment in video game reviews. This endeavor will not only enhance our understanding of the strengths and weaknesses of various techniques but also enable us to provide valuable insights to businesses within the gaming industry, ultimately facilitating data-informed decisions to improve customer satisfaction and brand loyalty.

By utilizing advanced techniques such as Fast Text vectorization, Logistic egression, Support Vector Machines, Random Forest Classifier, Extreme Gradient Boosting Algorithm and our neural network CNN’s technique, our aim is to uncover underlying sentiments within the main topics discussed by customers in their reviews. Customer reviews are a rich source of information that can provide valuable insights to businesses and help them improve their products or services. However, manually analyzing and extracting information from many reviews can be a challenging and time-consuming process. For this reason, automated approaches that leverage NLP techniques and machine learning have gained popularity for sentiment analysis and topic extraction. By automating these processes, we can effectively categorize and understand the sentiments expressed by customers, as well as identify the main topics mentioned in the reviews. In this project, our main objectives are twofold.

Firstly, our primary objective is to obtain a dataset comprising customer reviews with meticulously cleaned text that is readily comprehensible by our algorithms. To achieve this, a comprehensive text cleaning process is exclusively applied to the reviews, and the intricacies of this process will be elaborated upon in subsequent sections of this article. Subsequently, the essential step of vectorization is executed, employing the FastText vectorizer. This algorithm possesses the capability to discern the intricate relationships between words and construct word embeddings based on a vocabulary established during the training phase.

The concluding segment of our project entails the development and training of five distinct supervised learning techniques or models. In the subsequent phase, we harness the vectorized sentences derived from the application of FastText and independently train each model. This training involves the assignment of labels based on star ratings provided by the same dataset, which have been transformed into a binary format, specifically within the range {0, 1}, representing negative and positive sentiment, respectively.

1.1 Project Framework Introduction

Here is a brief overview of our project framework.

To commence, the initial phase of our study involved the preprocessing of customer reviews, aiming to render them as clean and usable text. Subsequently, we employed the FastText vectorizer to transform these refined reviews into a format comprehensible by our ensemble of supervised learning models.

Following this preprocessing, we pursued the individual training of five diverse models, specifically, Logistic Regression (LR), Support Vector Classifier (SVC), Random Forest (RF), Extreme Gradient Boosting (XGB), and Convolutional Neural Networks (CNNs). These models were trained with the goal of predicting sentiment values, discretized into the binary range {0, 1} to denote negative and positive sentiments, respectively.

In addition to model training, we conducted a thorough evaluation of each model, meticulously assessing their respective strengths and limitations. Our objective was to discern the model that exhibited superior performance and results. The following sections provide insights into the selection of the optimal model, offering explanations for why one of the five models may outperform the others.

2 LITERATURE REVIEW

The following section describes the supervised machine learning algorithms used in this project.

2.1 Machine Learning Algorithms/ Supervised learning Techniques

LR is a powerful binary classification method widely used in ML. It is valued for its simplicity, interpretability, and strong empirical performance.
The LR model utilizes the logistic function, also known as the sigmoid function, to map the input variables to a probability between 0 and 1. By estimating the parameters of the linear equation through the maximization of the likelihood function, logistic regression finds the best-fitting line that separates the two classes.

The interpretability of LR sets it apart. The coefficients associated with each input variable reveal the direction and strength of their impact on the output, allowing for a deeper understanding of the relationship between predictors and the binary outcome.

Logistic regression has demonstrated success in various domains. Studies by Li et al. [1] and Bi et al. [2] showcased its effectiveness in predicting hospital mortality and assessing the risk of heart failure, respectively.
In the realm of (NLP), logistic regression has found utility in text classification and sentiment analysis tasks. Taric et al [3] employed LR for text classification and achieved competitive results.

Support Vector Machines (SVM) is a powerful supervised machine learning algorithm that has found widespread applications in various fields [4][5][6]. At its core, SVM aims to find an optimal hyperplane that maximizes the margin between different classes in a dataset. This hyperplane serves as a decision boundary, enabling SVM to excel in both classification and regression tasks.

SVM's versatility and robustness make it a valuable tool in numerous domains. In the field of computer vision, SVM has been instrumental in tasks such as image classification and object detection. In natural language processing (NLP), it has been used for text classification, sentiment analysis, and spam email detection. In the realm of bioinformatics, SVM aids in tasks like protein structure prediction [7]. Additionally, SVM has applications in finance for stock price prediction, in healthcare for disease diagnosis, and in remote sensing for land cover classification [8].

The success of SVM lies in its ability to handle high-dimensional data, nonlinear relationships, and class imbalances effectively. By mapping data points into a higher-dimensional space, SVM can transform complex problems into more manageable ones, making it a valuable asset in the toolkit of machine learning practitioners.

Random Forests [9] is a versatile ensemble learning algorithm that combines multiple decision trees. They are widely used in fields like ecology for species distribution modeling, finance for credit risk assessment, healthcare for disease prediction, and remote sensing for land cover classification. Random Forests excel in handling high-dimensional data and complex interactions between variables, making them valuable for data-driven decision-making.

Extreme Gradient Boosting (XGBoost) is a highly versatile and powerful gradient boosting algorithm that has achieved remarkable success across various machine learning applications. Recognized for its outstanding performance, XGBoost has become indispensable in the finance industry for tasks like stock price prediction [10], and in the healthcare sector for disease diagnosis [11] and medical image analysis [12]. Additionally, XGBoost has found extensive use in natural language processing (NLP) for text classification and sentiment analysis. What sets XGBoost apart is its ability to handle high-dimensional data, nonlinear relationships, and imbalanced datasets with remarkable efficiency. Its robustness, flexibility, and ease of use make it an invaluable tool for data-driven decision-making.

Convolutional Neural Networks (CNNs) have emerged as a groundbreaking deep learning architecture, particularly influential in computer vision and natural language processing. In the realm of computer vision, CNNs have significantly advanced object detection tasks [13] and image classification [14]. Furthermore, CNNs have paved the way for intricate medical image analysis, enabling precise disease identification and diagnostic assistance [15]. In natural language processing (NLP), CNNs have demonstrated prowess in text classification, sentiment analysis, and language understanding, showcasing their adaptability across domains. The strength of CNNs lies in their ability to automatically learn relevant features from data, making them adept at handling complex patterns and structures in various types of information. As a result, CNNs have become an indispensable tool for researchers and practitioners striving to extract meaningful insights from image and text data.

CountVectorizer is a widely used text preprocessing technique in natural language processing (NLP) and machine learning. It is employed to convert a collection of text documents into a matrix of token counts, representing the frequency of words or n-grams in the documents [16][17]. By tokenizing the text and counting the occurrences of each token, CV transforms the text data into a numerical representation suitable for machine learning algorithms.

CV has proven to be effective in various text analysis tasks. It has been utilized in sentiment analysis, where it extracts features from text data for sentiment classification [18]. In topic modeling, CV is used to identify main themes or topics in a collection of documents [19]. Additionally, in document classification, CV represents text documents and enables classification into predefined categories [20].

The simplicity and efficiency of CV make it a valuable tool in the NLP toolkit. It allows for the conversion of textual data into numerical representations suitable for machine learning algorithms.

FastText is a powerful word embedding and text classification technique. It represents words as bags of character n-grams, allowing it to capture morphological variations and handle out-of-vocabulary words effectively. FT has gained popularity in various NLP tasks, including text classification, sentiment analysis, and language identification. It has been widely used in academic and industrial settings, demonstrating its robustness and scalability. FT has shown promising results even with limited training data, making it suitable for scenarios with scarce labeled data [21].

Real-world applications of FT include e-commerce product categorization and recommendation systems, news topic modeling, and content recommendation [22]. FT's efficiency and accuracy in large-scale text classification tasks have been highlighted by Jin et al. [23].

FastText has emerged as a popular tool in NLP, offering efficient word embeddings and accurate text classification capabilities. Its ability to handle subword information and low-resource scenarios makes it suitable for various applications.

2.2 Related Work

Previous studies have been done on sentiment analysis by utilizing machine learning techniques. Britto and Pacifico [24] conducted a study on video game acceptance, using sentiment analysis on Brazilian Portuguese game reviews from the Steam platform. They employed the Bag-of-Words method for feature extraction and implemented algorithms such as Random Forest, SVM, and LR for sentiment classification. The study showcases the application of machine learning techniques in analyzing and understanding the sentiments expressed in video game reviews.

Bjorn Straat et al [25] conducted a study on user attitudes, using aspect-based sentiment analysis on Metacritic reviews regarding two core video game franchises, Mass Effect and Dragon Age. They calculated the frequency of each word using AntCont to select the core aspects for their analysis. In addition to that, the http://www.crowdflower.com was used to assign sentiment scores to the overall use of the aspect discussed in each review. The study showcases the importance of aspect-based approaches and that a given aspect can reflect the overall sentiment of a review. In addition to that, it is shown that reviewers tend to write their reviews towards different aspects of any given game.

Tan et al [26] conducted a study on a comparative analysis on machine learning approaches regarding sentiment analysis on customer review data provided from Metacritic and Steam. They used TF-IDF vectorizer for feature extraction and a wide range of machine learning approaches such as LR, MNB, SVC, XGB, MLP and compared the different models. In this study it is shown that classification using hyperplanes instead of probabilistic approaches might be a better way to perform text classification and therefore contribute efficiently to sentiment analysis tasks.

In the realm of sentiment analysis within the video gaming industry, there exists a notable gap in comprehensive experimentation and optimization efforts. This deficiency pertains to the systematic evaluation and refinement of various

machine learning models tailored for the classification of sentiment in gaming-related customer reviews. The gaming industry, marked by its dynamic and evolving landscape, demands specialized approaches for sentiment analysis. However, the existing research landscape falls short in its exploration of the most fitting models and methodologies for effectively categorizing sentiment within gaming reviews. Bridging this gap is imperative to enhance our understanding of sentiment dynamics in the gaming domain, enabling the identification of models that can accurately capture and classify sentiments, ultimately contributing to improved decision-making and customer satisfaction within the gaming industry.

3 METHODOLOGIES

This section presents the project frameworks, dataset, text preprocessing, data labeling, feature extraction, handling of imbalanced classes, model applications and hyperparameter tuning.

3.1 Project Framework

The project framework is displayed in figure 1, 2,3 and 4.

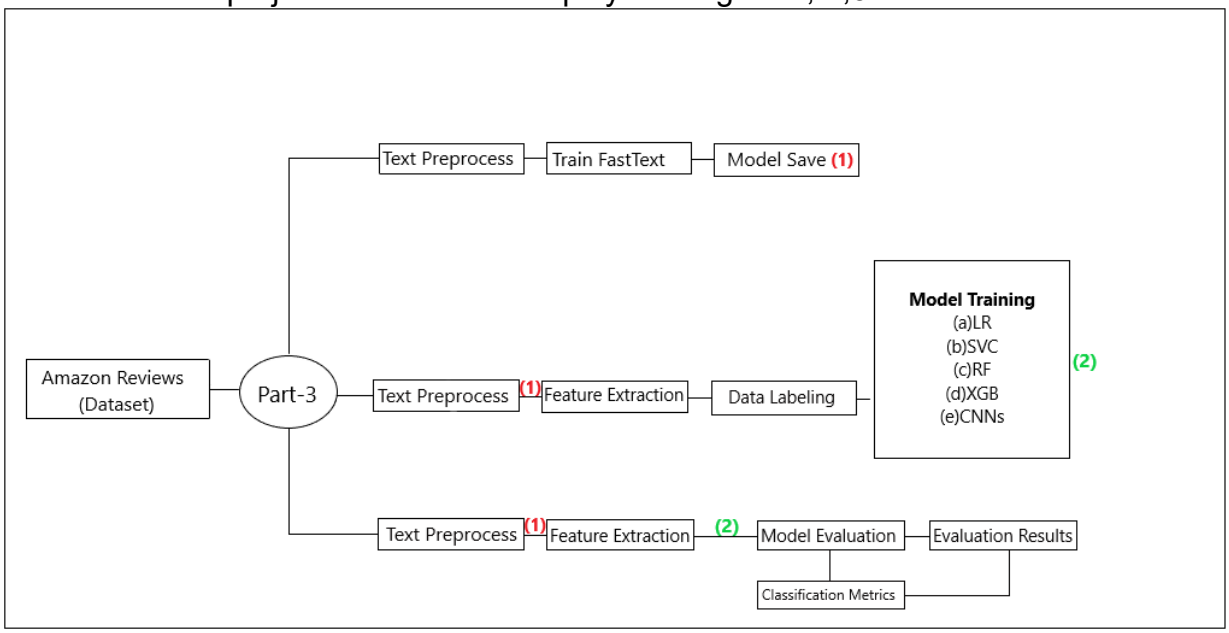

Figure 1: Project Framework.

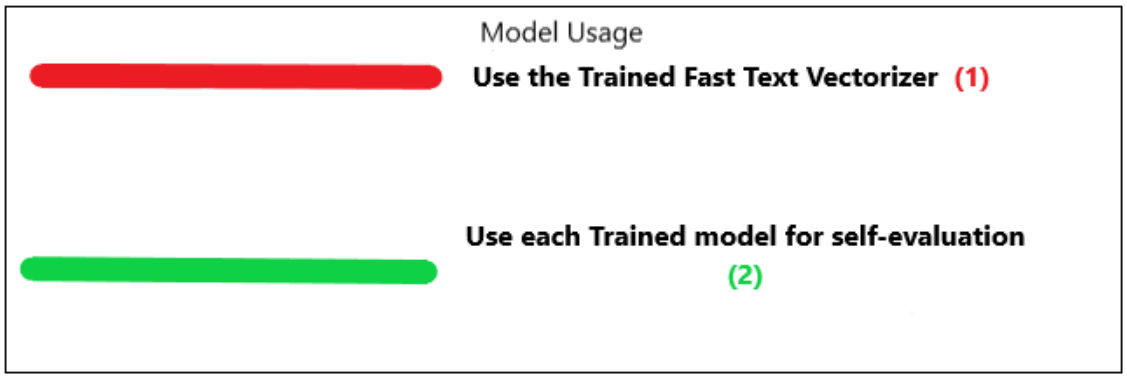

Figure 2: Model Usage.

3.2 Model Application Overview

The machine learning algorithms used in this project are as follows:

a) Logistic Regression

LR is by default used for binary classification, but its functionality is extended by the Scikit-Learn library to also perform multi-class classification.

b) Support Vector Machines

Support Vector Machines (SVMs) are powerful machine learning algorithms for classification and regression. They find the best way to separate data into different groups.\

c) Random Forests

Random Forests are ensemble machine learning models that excel in various tasks. They consist of an ensemble of decision trees, where each tree is trained on a different subset of the data and a random subset of features. This ensemble approach enhances prediction accuracy and reduces overfitting. Random Forests are particularly suitable for both classification and regression tasks and are known for their robustness and ability to handle high-dimensional data effectively.

d) Extreme Gradient Boosting

Extreme Gradient Boosting, or XGBoost, is a versatile machine learning algorithm renowned for its high performance in classification and regression tasks. XGBoost is an ensemble method that combines multiple decision trees to make accurate predictions. It's widely used for its robustness, efficiency, and capacity to handle complex data, making it a top choice in various domains.

e) Convolutional Neural Networks

Convolutional Neural Networks, or CNNs, are a type of deep learning model mainly used for tasks involving images and sequences. They are designed to automatically learn and identify hierarchical patterns and features in data. CNNs are especially effective in image-related tasks such as image classification, object detection, and image generation, as well as in natural language processing for tasks like text classification and sentiment analysis.

f) Fast Text

FT is a machine learning algorithm for learning word representations, incorporating sub-word information to handle out-of-vocabulary words and capture morphological details. It uses techniques like stochastic gradient descent to train word vectors based on context, enabling various NLP tasks such as sentiment analysis and text classification.

g) CountVectorizer

CountVectorizer is a text preprocessing technique used in natural language processing. It converts text into numerical features for machine learning. It tokenizes the text, counts the occurrence of each term, and creates a count matrix. This matrix represents the frequency of each term in the text. CountVectorizer is commonly used for tasks like text classification and sentiment analysis. It provides a simple and efficient way to transform text into a format that machine learning algorithms can understand.

3.3 Text Preprocessing

In this section the text preprocessing step for each part of the data set is discussed.

3.3.1Text Preprocessing on Part3

The data were preprocessed before being used to train the models as the element “Text preprocess” for each operation suggests in figure 1. The text preprocessing procedure is the same in each depiction in figure 1.

Firstly, we added the ‘summary’ column to the “reviewText” then the Nan values were replaced with an empty string (““), extra spaces were narrowed down to a maximum of one space and capital letters were transformed to lowercase. Also, the data meant to be used as labels for every model training operation were scaled to much the range of values: {0, 1} to convey the binary core values for negative and positive sentiment respectively.

Next, we trained the CountVectorizer with the reviews in hand and checked the cosine similarity between each review, and duplicates were removed. This is followed by the removal of punctuations, URLs, HTML tags and hyperlinks. In addition, we introduced contradiction and abbreviations expanding by using two custom dictionaries respectively. Lastly, we expanded the concatenated words.

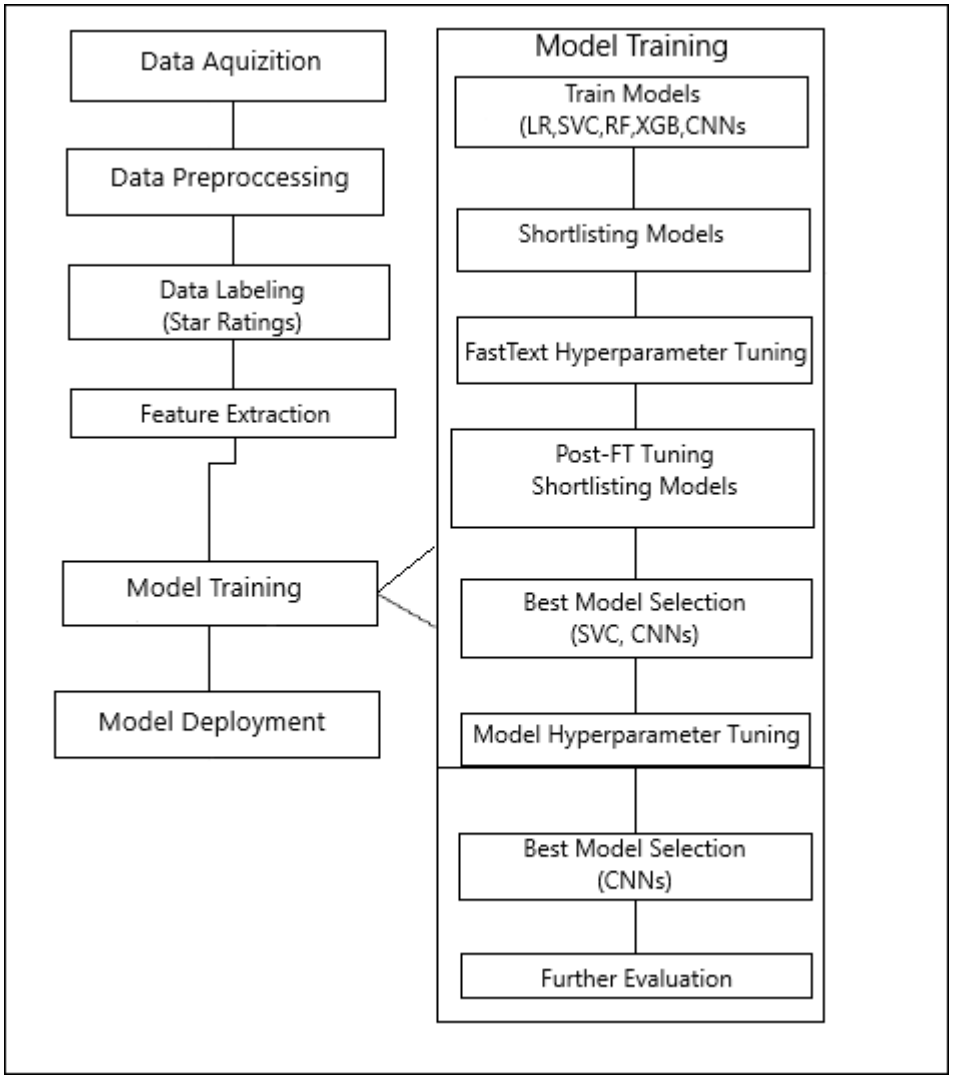

Figure 3: Explanatory Project Framework.

3.4 Data Labeling

As for data labeling operation the sentiments of the reviews (Part-3) were labeled as positive or negative by using the stars rating which was already contained inside the original data set. These ratings were scaled originally from one to five, but we scaled down the values to much the range {0, 1} because our goal is to mark label as negative and positive sentiment respectively.

3.5.1 Feature Extraction (FastText)

The word embedding approach has been applied by using Gensim's FastText to perform feature extraction. Following that we introduced a basic form of a FastText vectorizer and then we built the vocabulary of the vectorizer based on the tokenized text we obtained from our preprocessing step. Next, we trained the vectorizer by using again the tokenized text along with the length of it, to the respective parameters "corpus_iterable" and "total_examples". Lastly, by using Pickle's library function dump () we saved the model for every vectorization needed in our project as shown (in red) in figures 1 and 2.

3.5.2 Feature Extraction (CountVectorizer)

This feature extraction technique has been applied by using the Sci-kit Learn's library CountVectorizer and it was trained by using raw text data of our dataset and transforming the text data to respective vectors. The purpose of it was to calculate the cosine similarity between each vector using the cosine similarity function from Sci-Kit learn's library to find duplicate review (data-frame rows) as suggested in our preprocessing step.

3.6 Handling imbalanced classes

Since our data contains a significantly greater number of positive reviews than negative or neutral (neutral is not considered here) which may affect the performance of the models, we introduce a custom way of selecting randomly equal number of reviews per star review rating. This is done initially before we scale down the star rating to {0, 1} as previously discussed, so we initially choose an equal number of reviews for each star rating from one to five ex. 1000 reviews for every star rating from [1, 5] resulting in a total of 5000 equal distributed reviews (in hundreds).

.

3.7 Hyperparameter Tuning

To improve the performance of the models, a Randomized Search Cross Validation with 5 k-folds was carried out to find the best combination of hyperparameters for the SVC, CNNs and FastText models respectively.

4 EVALUATION RESULTS AND DISCUSSIONS

4.1 Dataset

The dataset contains 2,565,349 customer reviews from the Amazon company in the field of video games. The dataset was obtained by https://nijianmo.github.io/amazon/index.html and specifically from the section “Files” and sub-section “Complete review data” selecting the Video Games file for download. Furthermore, the dataset was precisely divided into three distinct parts, each consisting of approximately 855,116 reviews, to serve our project, we selected the 3rd part as it provided quality text in contrast to the first 2 parts.

Additionally, we kept only the four most essential columns of the dataset, which are: {asin, reviewText, summary, overall}. As for the contents of the aforementioned columns, asin is the product’s ID, the reviewText is the written review, overall is the rating from one to five and summary is the summary of the written text all provided by each customer respectively. Lastly, the column "overall" was renamed to "stars” for ease of reference reasons.

4.2 Evaluation Metrics

The result of the evaluation performed on the ensemble of all our models is presented to figures 4 through 10 and tables 1 through 4. The evaluation was performed using part3 of our dataset with a value count of 4000 reviews. The evaluation methods that we used are classification_report, confusion_matrix, roc_auc_score and roc_curve with each one of them contained in the SCI-KIT Learn’s library.

Accuracy, Precision, recall, and F1-score are metrics commonly used to evaluate the performance of classification models. They are derived from the concepts of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

Accuracy is the measure of correctly classified instances out of the total instances:

$$Accuracy = \frac{TP + TN}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}$$

Precision measures the proportion of correctly predicted positive instances (true positives) out of the total instances predicted as positive:

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

Recall, also known as sensitivity or true positive rate, measures the proportion of correctly predicted positive instances (true positives) out of the actual positive instances:

$$Recall = \frac{TP}{\mathrm{TP} + \mathrm{FN}}$$

F1-score is the harmonic mean of precision and recall. It provides a balanced measure that considers both precision and recall:

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}$$

These metrics are commonly used in binary classification tasks but can also be extended to multi-class classification by computing them for each class separately and then averaging or taking the weighted average based on class frequencies.

4.2.1 Evaluation Results

Logistic Regression

	precision	recall	f1-score	support
0	0.79	0.89	0.84	2447
1	0.79	0.63	0.70	1553
accuracy			0.79	4000
macro avg	0.79	0.76	0.77	4000
weighted avg	0.79	0.79	0.78	4000

Random Forests

	precision	recall	f1-score	support
0	0.77	0.89	0.83	2447
1	0.77	0.59	0.66	1553
accuracy			0.77	4000
macro avg	0.77	0.74	0.75	4000
weighted avg	0.77	0.77	0.76	4000

Support Vector Classifier

	precision	recall	f1-score	support
0	0.79	0.90	0.84	2447
1	0.80	0.62	0.70	1553
accuracy			0.79	4000
macro avg	0.79	0.76	0.77	4000
weighted avg	0.79	0.79	0.79	4000

Extreme Gradient Boosting

	precision	recall	f1-score	support
0	0.80	0.86	0.83	2447
1	0.75	0.67	0.70	1553
accuracy			0.78	4000
macro avg	0.78	0.76	0.77	4000
weighted avg	0.78	0.78	0.78	4000

Convolutional Neural Networks

	precision	recall	f1-score	support
0	0.81	0.89	0.85	2447
1	0.80	0.67	0.73	1553
accuracy			0.81	4000
macro avg	0.80	0.78	0.79	4000
weighted avg	0.80	0.81	0.80	4000

Figure 4: Classification Reports (All models).

As depicted in Figure 4, Convolutional Neural Networks Classifier stands out with the highest level of accuracy among all the models under consideration. This achievement reflects the model's proficiency in making correct predictions across our dataset. Furthermore, a closer examination of the remaining performance metrics reveals that they consistently exhibit either the highest scores or closely approximate the highest scores, underscoring the model's excellence.

In particular, the macro-averaged precision, recall, and F1-score metrics are of paramount importance. High macro-averaged precision implies that the model excels at minimizing false positive predictions for all classes, ensuring a low rate of erroneous positive classifications. High macro-averaged recall signifies that the model is highly effective at capturing positive instances across

all classes, leaving few positive cases unidentified. Lastly, a high macro-averaged F1-score suggests a harmonious balance between precision and recall across all classes, reflecting the model's overall proficiency in classification tasks.

Following this, we delve into the refinement process for our FastText vectorizer. This involves the utilization of a Grid Search, a systematic approach that explores various combinations of hyperparameters included in our predefined parameter grid, as outlined in Table 1. The primary aim of this procedure is to identify the optimal set of hyperparameters that produce the most favorable outcomes. These carefully selected hyperparameters are detailed in Table 1 for your reference.

Table 1:Tested hyperparameters and their respective values (SVC).

Hyper-Parameters	List of Values	Best Value
vector_size	100, 200, 300	200
window	3, 5, 7	3
min_count	1, 5, 10	1
sg	0, 1	0

Logistic Regression

	precision	recall	f1-score	support
0	0.79	0.89	0.84	2403
1	0.80	0.65	0.72	1597
accuracy			0.79	4000
macro avg	0.80	0.77	0.78	4000
weighted avg	0.80	0.79	0.79	4000

Random Forests

	precision	recall	f1-score	support
0	0.78	0.90	0.84	2403
1	0.81	0.61	0.70	1597
accuracy			0.79	4000
macro avg	0.79	0.76	0.77	4000
weighted avg	0.79	0.79	0.78	4000

Support Vector Classifier

	precision	recall	f1-score	support
0	0.79	0.91	0.84	2403
1	0.82	0.63	0.71	1597
accuracy			0.80	4000
macro avg	0.80	0.77	0.78	4000
weighted avg	0.80	0.80	0.79	4000

Extreme Gradient Boosting

	precision	recall	f1-score	support
0	0.80	0.85	0.83	2403
1	0.75	0.68	0.71	1597
accuracy			0.78	4000
macro avg	0.78	0.77	0.77	4000
weighted avg	0.78	0.78	0.78	4000

Convolutional Neural Networks

	precision	recall	f1-score	support
0	0.80	0.92	0.86	2403
1	0.85	0.66	0.74	1597
accuracy			0.81	4000
macro avg	0.82	0.79	0.80	4000
weighted avg	0.82	0.81	0.81	4000

Figure 5: Classification Reports (FT-refined).

In the subsequent phase, we conducted a retraining of our models, this time utilizing the enhanced FastText version, with the intention of achieving improved results. As illustrated in Figure 5, both the Support Vector Classifier (SVC) and Convolutional Neural Network (CNN) models exhibited the most promising outcomes. Notably, the CNN model achieved the highest accuracy, while the SVC model secured the second-highest accuracy among the entire model set. It's worth mentioning that all models demonstrated slight enhancements following the implementation of the refined FastText vectorizer. The most significant improvement was observed in the case of the CNN model, which not only witnessed an uptick in accuracy but also in the macro-averages, indicating an overall refinement of the CNN model.

So, SVC and CNN models are selected now to be evaluated in contrast to each other to determine the single best performing model of our experimental analysis. As the classification reports suggests in figures 4 and 5 the CNN model tends to perform better, but the models have a basic implementation and are not fine tuned for our specific purpose regarding text classification and/or for the dataset. So, our next step consists of a fine-tuning process for both SVM and CNN. The following tables contain the grids of hyperparameters that will be investigated along with the best value selected.

Table 2:Tested hyperparameters and their respective values (SVC).

Hyper-Parameters	List of Values	Best Value
C	0.1,1, 10	10
kernel	linear, poly, rbf, sigmoid	rbf
gamma	scale, auto, 0.1, 1, 10	scale

Table 3: Tested hyperparameters and their respective values (CNNs).

Hyper-Parameters	List of Values	Best Value
batch_size	32, 64, 128	32
epochs	10, 20 ,30	10
learning_rate	0.001, 0.01, 0.1	0.01
dropout_rate	0.2, 0.3, 0.4	0.4

After rigorous testing we implemented and trained both models to evaluate the performance of the new refined models. As shown in figure 6 some metrics of both models benefited from the previous classification report (figure 5). Regarding our CNN model, accuracy is a clear indication that the tuning process was successful with an increase of 1 (81% to 82%) percent which is an important and difficult to achieve increment especially in high levels of accuracy like the 80 percentiles here.

Convolutional Neural Networks

	precision	recall	f1-score	support
0	0.81	0.92	0.86	2443
1	0.84	0.67	0.74	1557
accuracy			0.82	4000
macro avg	0.82	0.79	0.80	4000
weighted avg	0.82	0.82	0.81	4000

Support Vector Classifier

	precision	recall	f1-score	support
0	0.80	0.90	0.85	2438
1	0.81	0.66	0.72	1562
accuracy			0.80	4000
macro avg	0.80	0.78	0.79	4000
weighted avg	0.80	0.80	0.80	4000

Figure 6:Post fine-tuning classification reports (CNNs, SVC)

As for the SVC the accuracy metric remained static at 0.80 (80%) and we can conclude that the fine-tuning process did not benefit this model as the CNN one. In addition to that it seems that some metrics have decreased values which indicates that some aspects of our model hurt because of the hyperparameters we introduced with an example being the decrease in precision regarding the positive class (depicted as 1 in figure 5 and 6) and aspects like macro average recall and f1-score that did in fact increase form 0.77 and 0.78 to 0.78 and 0.79 respectively. Furthermore, it should be noted that there was a significant increase in recall of an 0.04 increment. All in all, we can comfortably say that both models benefited from the fine-tuning process, which was a desirable outcome, but the Convolutional Neural Networks seem to have the advantage regarding all the evaluations. So, the CNNs is selected as the single best model out of experimental analysis of 5 different models.

In the upcoming sections, we will delve deeper into our CNNs model. We'll do this by examining two key metrics: the confusion matrix and the Receiver Operating Characteristics (ROC). The goal here is to gain a better understanding of our chosen model and potentially uncover the reasons that set it apart as the best model among the alternatives.

4.2.3 Receiver Operating Characteristic (ROC) Curve

The Receiver Operating Characteristic (ROC) curve is a graphical representation of the performance of a binary classification model. It is plotted by varying the threshold for classification and calculating the True Positive Rate (TPR) and False Positive Rate (FPR) at each threshold.

The equation for the True Positive Rate (TPR), also known as sensitivity or recall, is:

$$TPR = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

The equation for the False Positive Rate (FPR) is:

$$FPR = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TP}}$$

FPR = False Positives / (False Positives + True Negatives)

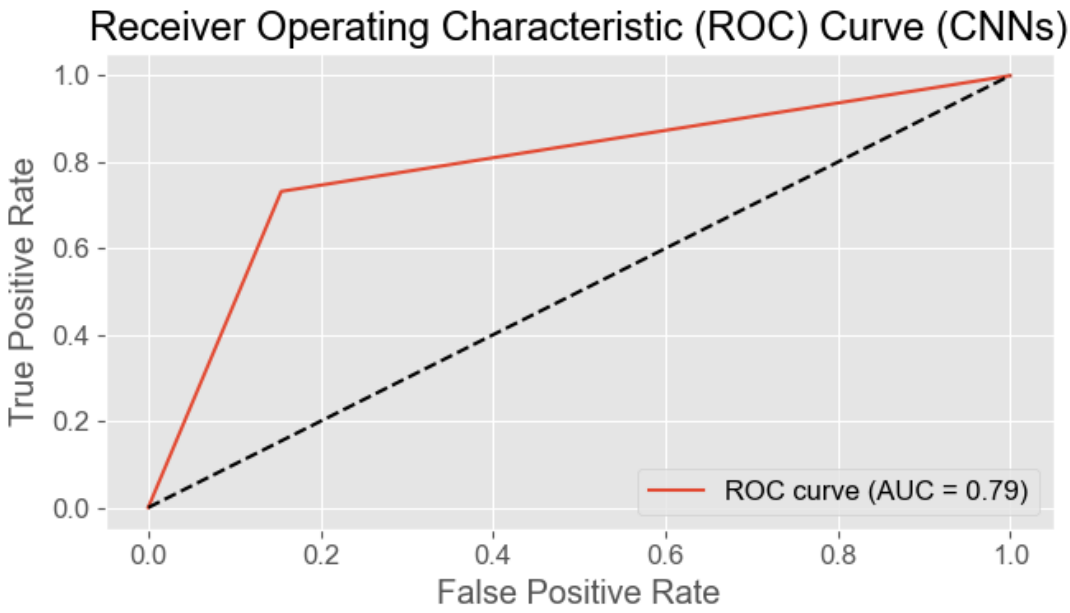

Figure 7: Receiver Operating Characteristic Curve for CNNs

As illustrated in Figure 7, the ROC curve reveals an AUC value of 0.79, surpassing the threshold of 0.5, thus signifying a favorable metric outcome. In a broader context, a heightened Area Under the Curve (AUC) indeed signifies an elevated True Positive Rate (TPR) or heightened sensitivity. To elaborate, when the AUC of the Receiver Operating Characteristic (ROC) curve approaches the value of 1, it reflects an improved capacity of our model to accurately discern positive instances (true positives) across a spectrum of classification thresholds.

In contrast, one of our less proficient models, specifically XGBoost or Random Forests, as evident in Figures 8 and 9, manifests a diminished AUC value. This reduced AUC value underscores a diminished proficiency in identifying true positive instances when compared to our CNNs model.

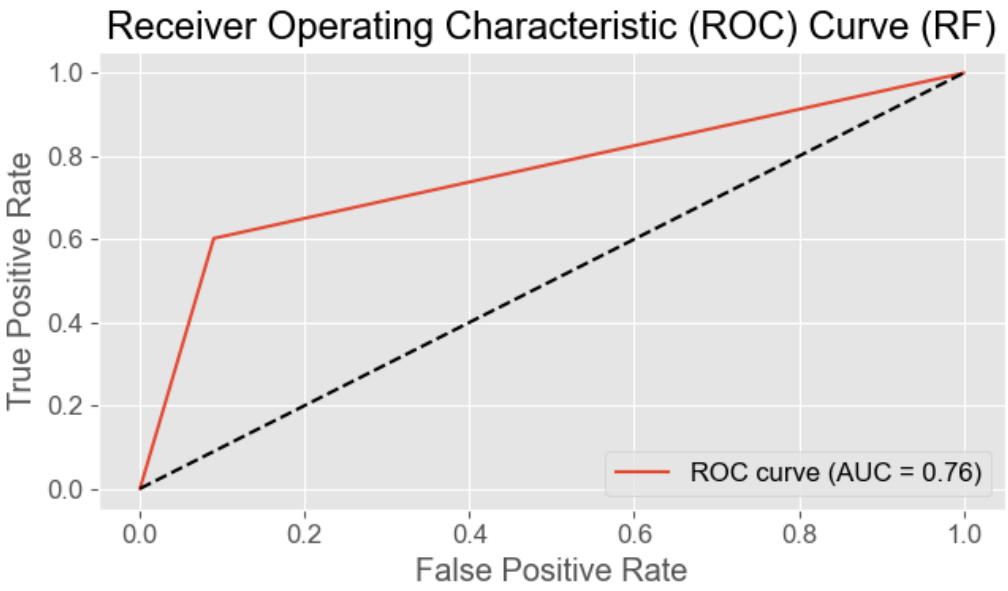

Figure 8:Receiver Operating Characteristic Curve for RF

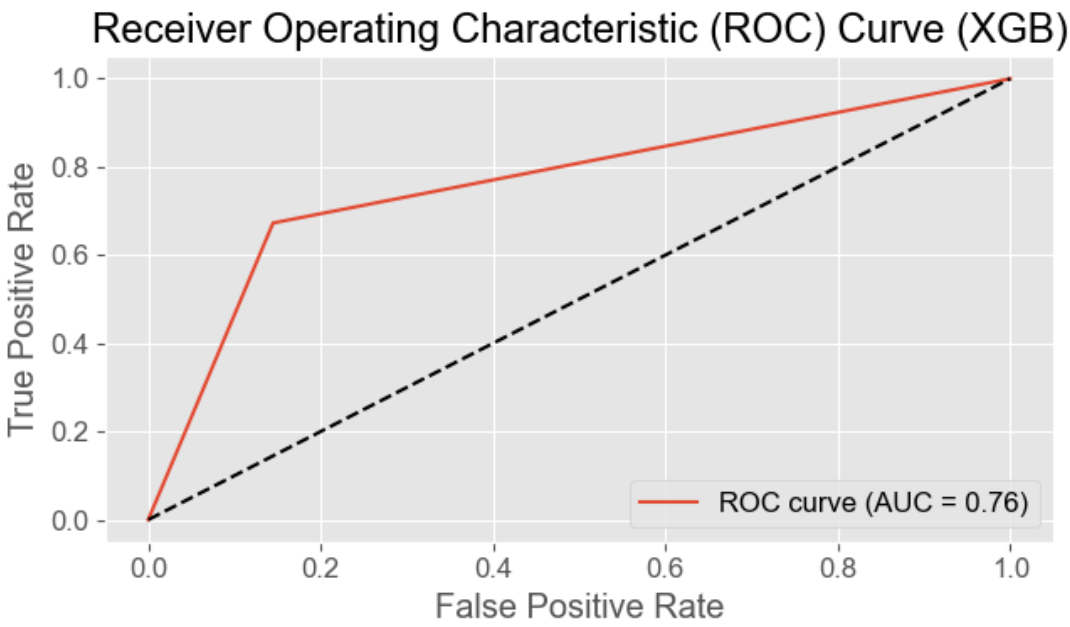

Figure 9: Receiver Operating Characteristic Curve for XGBoost

It is evident that Convolutional Neural Networks (CNNs) exhibit superior performance in the specific ROC-AUC metric, as indicated by their higher AUC value. This underscores yet another rationale for our decision to select CNNs.

4.2.3 Confusion Matrix

The confusion matrix is a table used to evaluate the performance of a classification model. It summarizes the predictions made by the model and compares them to the actual values. The confusion matrix consists of four metrics: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). The equations for these metrics are as follows:

True Positive (TP): The number of positive instances that were correctly predicted as positive.

False Positive (FP): The number of negative instances that were incorrectly predicted as positive.

True Negative (TN): The number of negative instances that were correctly predicted as negative.

False Negative (FN): The number of positive instances that were incorrectly predicted as negative.

A confusion matrix is typically represented as follows:

	Predicted Negative	Predicted Positive
Actual Negative	TN	FP
Actual Positive	FN	TP

Table 4: Confusion Matrix Typical Representation

The confusion matrix generated for CNNs is depicted in figure 9 bellow.

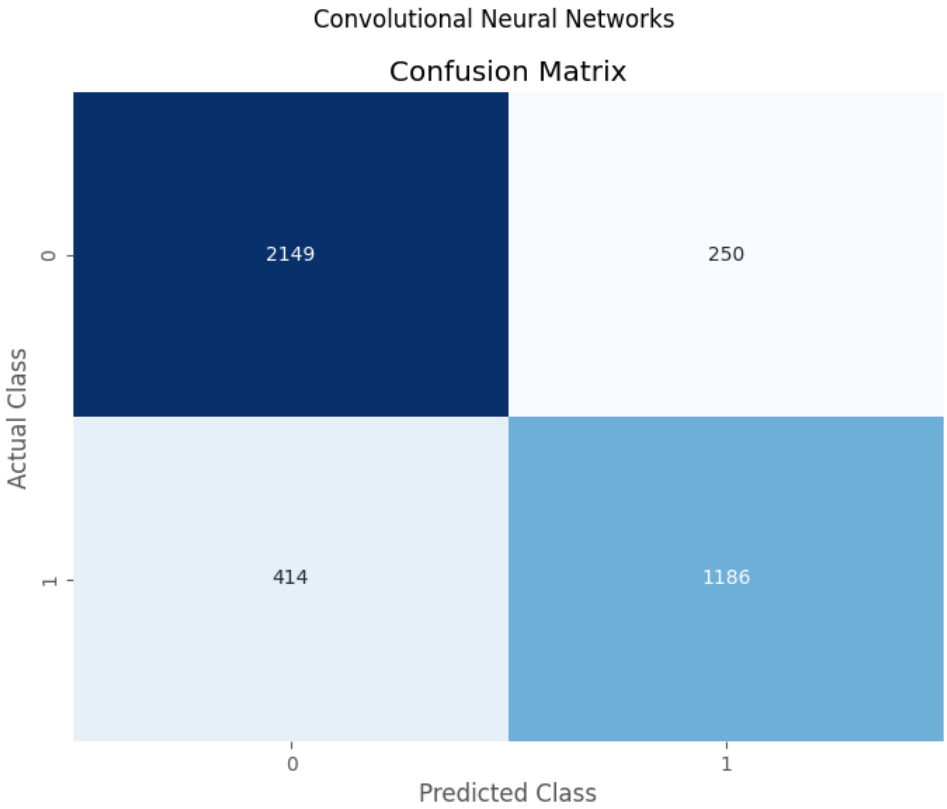

Figure 10: Confusion Matrix for CNNs

Upon careful examination of Figure 9, the confusion matrix reveals noteworthy observations regarding our CNNs model. It showcases elevated counts in true positives and true negatives, accompanied by relatively diminished figures in the categories of false negatives and false positives. This observation aligns harmoniously with our earlier assessment using the ROC-AUC curve metric, affirming the model's propensity for high true positive rates in its predictive capacity.

5 CONCLUSIONS

All in all, five machine learning algorithms were trained with game reviews obtained from Amazon. It was shown that CNNs model understands accurately the sentiment contained in a sentence regarding a portion of the original data frame as the “section evaluation results and discussion suggests”. Considering the results depicted in the aforementioned section CNNs model is regarded to be the best performing model as our experimental analysis suggests.

Through this project, game developers and studios will be able to have a tool to automate the task of sentiment analysis on user's opinions to make better decisions in game development and production. So, CNNs serves our purpose, and we are obliged to select it as our final model selected for this project.

Future work should focus on analyzing exclamation marks, capital letters and emojis or emoticons which are widely used by users to express their feelings. In addition, expanding the model's prediction scale and implementing neutral sentiment as one of the predicted classes or implement sentiment intensity expressed in combination with capital letter, exclamation marks etc.to acquire more accurate and domain specific knowledge regarding sentiment expressed in gaming reviews and therefore improve the gaming industry with customer related/based decisions.

ABBREVIATIONS – ARCTIC-TERMS – ACRONYMS

LR	*Logistic Regression*
CV	*Count Vectorizer*
FT	*Fast Text*
NLP	*Natural Language Processing*
ML	*Machine Learning*
SVC	Support Vector Classifier
XGB	Extreme Gradient Boosting Classifier
MLP	Multilayer Perceptron
ROC	*Receiver Operating Characteristic*
AUC	*Area Under the Curve*
TF-IDF	*Term Frequency-Inverse Document Frequency*
TP	*True Positive*
TN	*True Negative*
FP	*False Positive*
FN	*False Negative*
TPR	*True Positive Rate*
FPR	*False Positive Rate*

WORK ATTACHMENTS

DATASET (Sample of 10k Reviews)

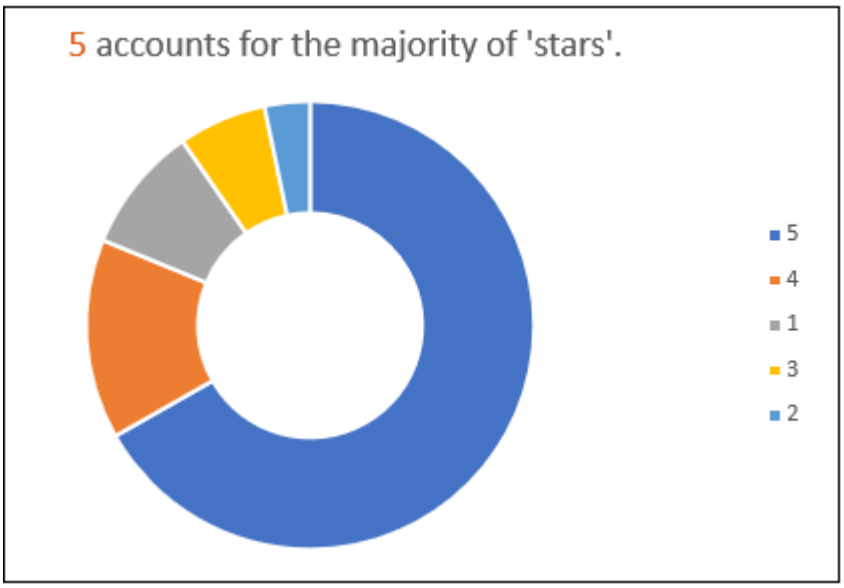

Figure 10: Chart showing Star ratings value counts (10.000 Reviews).

Table 5: Star ratings Value Counts (10.000 Reviews).

Row Labels	*Count of Stars*
5	*6677*
4	*1443*
1	*915*
3	*636*
2	*328*
Grand Total	*9999*

BIBLIOGRAPHY

1. Li J, Liu S, Hu Y, Zhu L, Mao Y, Liu J .Predicting Mortality in Intensive Care Unit Patients With Heart Failure Using an Interpretable Machine Learning Model: Retrospective Cohort Study. J Med Internet Res 2022;24(8):e38082 . URL: https://www.jmir.org/2022/8/e38082, DOI: 10.2196/38082 .
2. J. Bi, L. Song, L. Wang, B. Su, M. Wu, D. Li, S. Chen, Y. Liu, Y. Yang, Z. Zhou, Y. Hu, Y. Wang, S. Wu, Y. Tian. Transitions in metabolic health status over time and risk of heart failure: A prospective study, Diabetes & Metabolism. (2022). https://doi.org/10.1016/j.diabet.2021.101266 (https://www.sciencedirect.com/science/article/pii/S1262363621000495) .
3. T. Sabri, O. El Beggar, M. Kissi, Comparative study of Arabic text classification using feature vectorization methods, Procedia Computer Science, Volume 198, 2022, Pages 269-275, SSN 1877-0509, https://doi.org/10.1016/j.procs.2021.12.239, (https://www.sciencedirect.com/science/article/pii/S1877050921024789) .
4. A. Roy, S. Chakraborty, Support vector machine in structural reliability analysis: A review, Reliability Engineering & System Safety, Volume 233, 2023, 109126, ISSN 0951-8320, https://doi.org/10.1016/j.ress.2023.109126, https://www.sciencedirect.com/science/article/pii/S0951832023000418
5. Naman S. Bajaj, Abhishek D. Patange, R. Jegadeeshwaran, Sujit S. Pardeshi, Kaushal A. Kulkarni, Rohan S. Ghatpande, Application of metaheuristic optimization based support vector machine for milling cutter health monitoring, Intelligent Systems with Applications, Volume 18,2023, 200196, ISSN 2667-3053, https://doi.org/10.1016/j.iswa.2023.200196, https://www.sciencedirect.com/science/article/pii/S2667305323000212
6. Jianxin Tu, Lingzhen Hu, Khidhair Jasim Mohammed, Binh Nguyen Le, Peirong Chen, Elimam Ali, H. Elhosiny Ali, Li Sun, Application of logistic regression, support vector machine and random forest on the effects of titanium dioxide nanoparticles using macroalgae in treatment of certain risk factors associated with kidney injuries, Environmental Research, Volume 220, 2023, 115167, ISSN 0013-9351, https://doi.org/10.1016/j.envres.2022.115167, https://www.sciencedirect.com/science/article/pii/S001393512202494X
7. S. Saha and P. C. Shill, "Protein Structure Prediction in Structural Genomics without Alignment Using Support Vector Machine with Fuzzy Logic," 2023 International Conference on Electrical, Computer and Communication Engineering (ECCE), Chittagong, Bangladesh, 2023, pp. 1-6, doi: 10.1109/ECCE57851.2023.10100743.
8. Dash P, Sanders SL, Parajuli P, Ouyang Y. Improving the Accuracy of Land Use and Land Cover Classification of Landsat Data in an Agricultural Watershed. Remote Sensing. 2023; 15(16):4020. https://doi.org/10.3390/rs15164020

9. Abdulkareem, N. M., & Abdulazeez, A. M. (2021). Machine learning classification based on Radom Forest Algorithm: A review. International journal of science and business, 5(2), 128-142.

10. Yang, Y., Wu, Y., Wang, P., & Jiali, X. (2021). Stock price prediction based on xgboost and lightgbm. In E3s web of conferences (Vol. 275, p. 01040). EDP Sciences.

11. Ogunleye, A., & Wang, Q. G. (2019). XGBoost model for chronic kidney disease diagnosis. IEEE/ACM transactions on computational biology and bioinformatics, 17(6), 2131-2140.

12. Raja, R., Kumar, S., Rani, S., & Laxmi, K. R. (Eds.). (2020). Artificial intelligence and machine learning in 2D/3D medical image processing. CRC Press.

13. Hashi, A. O., Abdirahman, A. A., Elmi, M. A., & Rodriguez, O. E. R. (2023). Deep Learning Models for Crime Intention Detection Using Object Detection. International Journal of Advanced Computer Science and Applications, 14(4).

14. Zeng, W., Li, W., Zhang, M., Wang, H., Lv, M., Yang, Y., & Tao, R. (2023). Microscopic Hyperspectral Image Classification Based on Fusion Transformer with Parallel CNN. IEEE Journal of Biomedical and Health Informatics.

15. Oguz, C., Aydin, T., & Yaganoglu, M. (2023). A CNN-based hybrid model to detect glaucoma disease. *Multimedia Tools and Applications*, 1-19.

16. Liu, S. (2020). Sentiment Analysis of Yelp Reviews: A Comparison of Techniques and Models. available at: https://doi.org/10.48550/arXiv.2004.13851 .

17. M. Qorib, T. Oladunni, M. Denis, E. Ososanya, P. Cotae, Covid-19 vaccine hesitancy: Text mining, sentiment analysis and machine learning on COVID-19 vaccination Twitter dataset, Expert Systems with Applications, Volume 212, 2023, 118715, ISSN 0957-4174,.https://doi.org/10.1016/j.eswa.2022.118715, (https://www.sciencedirect.com/science/article/pii/S0957417422017407).

18. Shah, P., Swaminarayan, P., Patel, M. Sentiment analysis on film review in Gujarati language using machine learning. International Journal of Electrical and Computer Engineering (IJECE). Vol. 12, No. 1, February 2022, pp. 1030~1039. ISSN: 2088-8708, DOI: 10.11591/ijece. v12i1.pp1030-1039.

19. Älgå A., Eriksson O., Nordberg M. Analysis of Scientific Publications During the Early Phase of the COVID-19 Pandemic: Topic Modeling Study, J Med Internet Res 2020;22(11):e21559, URL: https://www.jmir.org/2020/11/e21559, DOI: 10.2196/21559.

20. B., Ankit, "DOCUMENT CLASSIFICATION USING MACHINE LEARNING" (2017). Master's Projects. 531. DOI: https://doi.org/10.31979/etd.6jmu-9xdt, https://scholarworks.sjsu.edu/etd_projects/531

21. Jin, Y., Yang, Y. ProtPlat: an efficient pre-training platform for protein classification based on FastText. BMC Bioinformatics 23, 66 (2022). https://doi.org/10.1186/s12859-022-04604-2

22. Kowsari K, Jafari Meimandi K, Heidarysafa M, Mendu S, Barnes L, Brown D. Text Classification Algorithms: A Survey. Information. 2019; 10(4):150. https://doi.org/10.3390/info10040150

23. Jin, Y., Yang, Y. ProtPlat: an efficient pre-training platform for protein classification based on FastText. BMC Bioinformatics 23, 66 (2022).https://doi.org/10.1186/s12859-022-04604-2

24. Britto, L. F., & Pacıfico, L. D. (2020) Evaluating Video Game Acceptance in Game Reviews using Sentiment Analysis Techniques. In Proceedings of SBGames 2020 (pp. 399-402).

25. Strååt, B., & Verhagen, H. (2017, April). Using User Created Game Reviews for Sentiment Analysis: A Method for Researching User Attitudes. In GHITALY@ CHItaly.

26. Tan, J. Y., Chow, A. S. K., & Tan, C. W. (2021, October). Sentiment Analysis on Game Reviews: A Comparative Study of Machine Learning Approaches. In International Conference on Digital Transformation and Applications (ICDXA) (Vol. 25, p. 26).