JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

# Prediction of Star Rating based on DeepLearning

**Anagha A Hebbar**
Dept of Computer Science
K S Institute of Technology
Bengaluru, Karnataka

**Lisha C**
Dept of Computer Science
K S Institute of Technology
Bengaluru, Karnataka

**Deeksha Naidu**
Dept of Computer Science
K S Institute of Technology
Bengaluru, Karnataka

**Sindhu S**
Dept of Computer Science
K S Institute of Technology
Bengaluru, Karnataka

**Mrs. Kavya**
Assistant Professor
Dept of Computer Science
K S Institute of Technology

**Abstract** :The rating of an online product is a crucial indicator of how users feel about it. The rating is used by consumers to assess the superiority and calibre of an online purchase. It helps a customer to make an online purchase decision. Additionally, it enhances the producer's ability to modify the product in the future as it is being reproduced and updated. There are instances when someone would purchase a product online and then further write a text review of it without giving it a number rating, most often a star rating. To analyse their business, however, producers need to know how well the products are rated. This rating can help producers analyse their businesses and increase their revenue.

In order to predict ratings based on customer text reviews, we applied some supervised machine learning techniques. We then compared the performance of the Random Forest Classifier, XGBoost, and Logistic Regression algorithms with TF-IDF Vectorizer. On the dataset titled "Grammar and Product Reviews" provided by Datafiniti, we used the aforementioned methods. We evaluated each algorithm's performance using its accuracy, precision, recall, and f1-score. The study found that the Random Forest algorithm, as well as precision, recall, and f1-scores, respectively, performed better when compared to other methods.

## INTRODUCTION

An examination of text reviews submitted by online shoppers is used to forecast star or numerical ratings. Online shopping is a growing aspect of our lives on a daily basis. People are growing accustomed to online purchasing as the days pass. Product ratings and reviews are the most crucial factors that consumers consider when making an online purchase. Business reviews are crucial since they aid organisations in determining excellence and quality in a number of areas that influence how well a company will do in the

marketplace. Through these testimonials and rankings, a manufacturer can also learn whether his product is acceptable to the consumer. As a result, both consumers and producers value the rating and review. However, some internet buyers only leave text reviews after their purchases and forget or are uninterested in leaving number ratings. Here, we applied three distinct machine learning algorithms to forecast ratings based on review wording.

For the classification of ratings, we suggested combining the TF-IDF Vectorizer technique with the Random Forest Classifier, XGBoost, and Logistic Regression algorithm. The preprocessing of the dataset's data is the first stage in our methodology. We used the TFIDF feature extraction technique in the second stage. For predicting star rating, we will use three supervised machine learning algorithms in the third stage. We then compared the outcomes of the machine learning approach to determine which one performs better on the applicable dataset. The ratings that the users give in the dataset most probably change from user to user. The user may assign a separate numbered rating corresponding to the written review they would have given.

As far as we are aware, no research has been done on the dataset referred to as "Grammar and Product Reviews". Therefore, the selection of the dataset for our study was made in order to work with a novel dataset and assess how well it works along with various our methods.

## RELATED WORKS

In Paper(1) Public opinions play a significant role in a company's capacity to assess its quality and excellence in a number of ways, which helps it to project its market worth. In other words, customer input has a significant impact on a company's bottom line. In order to categorize customer feedback, we apply sentiment analysis and opinion mining in this study, which focuses on reviews from customers of all types of restaurants. A huge and comprehensive text reviews dataset from the Yelp Dataset Challenge round 13 is utilized to get successful results for binary and multi label classification. Word2vec and Global Vector have been thoroughly compared with the outcomes of a deep learning algorithm called convolution Long short term memory (CLSTM) and a machine learning technique called Multinomial Naive Bayes (Glove). It has been established that CLSTM is the best model for categorizing review ratings after assessing the efficacy of each model using a variety of metrics. Additionally, we now understand the significance of bias to the system and how it explains performance variances observed on particular problems.

In paper(2), Nowadays, as a result of the advancement of digital media and the rise in Internet usage, a massive amount of data is being produced. Every day, enormous amounts of brand-new data are uploaded to the Web for various purposes. To fully realise this potential, methods like text mining and analysis are needed. By

examining user reviews, we can clearly understand what consumers are saying about the product. In this article, we present a novel method for abstract-level emotive analysis by performing POS tagging and n-gram classification on user evaluations. The entropy from this classification is then used by the machine learning method. After considering several algorithms, including SVM and random forest, this study applies the offered technique with promising results and increased accuracy by evaluating the data with the use of two algorithms, MaxEnt model and Naive Bayes classifier.

In Paper(3) Look up the restaurant's ratings and reviews before going. This study forecasts ratings based on user comments. a methodical approach to carrying out user-based sentiment analysis on reviews. The first machine learning solution treats the problem as a text classification challenge, using supervised classifiers like the Naive Bayes algorithm, which is perfectly suited for text classification. The second method, lexicon-based, uses a vocabulary of words with ratings associated with the text-based review to evaluate if a review is favorable or unfavorable. In this study, we show that, depending on the size of the dataset and the length of the review, lexical and machine learning approaches can be used to improve Naive Bayes classification accuracy by 5% to 10%.

In paper(4)We present a novel aspect-based rating prediction model (AspeRa) that computes user ratings for objects based on review texts and also identifies cogent review parts that can be utilized to support predictions or assist users. Using the greatest margin losses for combined item and user embedding learning and a dual-headed architecture, the AspeRa model significantly outperforms previously recommended state-of-the-art models like DeepCoNN, HFT, NARRE, and Transfer on two real world user review data sets. By qualitatively analyzing the aspects and statistically assessing rating prediction models based on these aspects, we show the potential of aspect embeddings in recommender systems.

In Paper(5) Graph-based collaborative filtering recommender systems are relatively new, however one of their biggest problems is the temporal information. The goal of the new link stream paradigm is to accurately depict the dynamism of the network by extending graphs while retaining crucial information. We investigate the impact of these connection stream characteristics on recommender systems. To capture the essential dynamics and structure of the data, we establish connection stream characteristics. We show how these characteristics make our depiction of the underlying system delicately nuanced and susceptible. Using the MovieLens20M movie dataset and the Goodreads book dataset, we focused on the rating prediction context of a traditional recommender system. We show that a gradient boosting machine (XGBoost) outperforms a system that only uses content by fusing some link stream features with some

content-based ones. The unique approach has to be further studied and included into cutting-edge recommender system algorithms in order to achieve these promising results. Link streams and graphs may offer better interpretability as natural representations of recommender systems at a time when algorithm openness is a contentious issue. Furthermore, we anticipate that these findings will lead to active discussions on the similarities and differences between link streams and conventional procedures in the field (matrix factorization, deep learning).

## OBJECTIVES

- It allows businesses to make highly accurate guesses as to the likely outcomes of a question based on historical data.
- It helps customers to make judgments about the quality of a product and its usefulness.
- To give accurate analysis to business owners which helps them to improve their products or services.
- To make the entire experience easier for the consumers to review and rate.
- To increase the product quality and business along with customer satisfaction.

## METHODOLOGY

### Project Development Methodologies

Software that is of higher quality and performs better can be produced by concentrating on the SDLC (software development life cycle). An application's planning, development, testing, and maintenance are all facilitated by the SDLC. Nevertheless, without a thorough understanding of its stages, it might be challenging to implement the software development life cycle intelligently and effectively. You may learn more about the specific SDLC phases in this article and how to apply them to provide high-quality goods to clients.

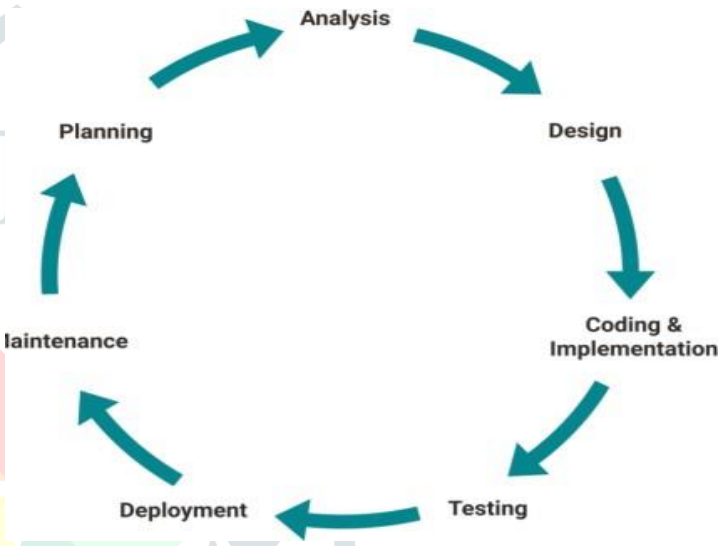


**Fig 4.1 : Work Procedure**

## WORKING PROCEDURE

### Analysis and defining requirements

The requirements the application must currently meet should be carefully considered. At this stage, the developers usually write a document called a software requirements specification. The SRS document describes the intended use and expected capabilities of the software. The functions the programme should offer are also included. The application will have a more user-friendly design with their needs clearly stated

**Design**

The code phase comes after the design phase. The application's specifications have now been determined and specified. The team will concentrate on the application's architecture and programming during this phase, including creating the programming language, industry standards, and procedures for handling particular issues and undertaking particular activities. The next step for the team is to create the software's user interface and choose the platforms it will run on. Security comes last but not least. What safety precautions will the program include? How will user data and passwords be protected? Now is the right time to deal with these problems. What other impacts does the SDLC process see as a result of design thinking? Check out our earlier article to see how the software product development process appears from a UX perspective. Many companies decide to develop a prototype at this stage of the SDLC process. An early-build prototype that is tested by intended users or the customer lowers project risk. This approach is the most effective for figuring out whether things are operating as intended and whether there is still room for improvement. But if you want a UX prototype to genuinely benefit your business, you need to be aware of the process. Following the conclusion of the design and prototype, coding and implementation take place.

**Implementation and coding**

The programmers are now working faster. Both the assembly of all parts and the coding of all previously designed capabilities are required. If there are numerous developers involved in the project, collaboration is even more important (which is the most typical situation). An important objective is to release high-quality code as soon as it is feasible to do so. It is advantageous to provide extensive documentation in the form of a handbook to facilitate user understanding of the program's purpose and operation and to facilitate the developers' task. There can be no high quality code without testing. Testing aids in identifying and fixing any specific flaws that could otherwise go unnoticed.

**Testing**

This step is finished before the product is made accessible to users. Even if not all of your tests are automated, you still need to set up a CI/CD pipeline. The testing phase's objective is to verify that each feature operates as planned. How can you create the testing process and pick the appropriate test types for your project? Our article on software testing and its significance for the SDLC process has the answers to these and other concerns.

**Deployment**

It's important to remember that the initial deployment is never easy. After testing is complete, it's time to make the application public and accessible to users or customers. The time has come to enhance hypothetical situations based on

actual events. Despite the fact that deployment is frequently automated, you and your staff should use caution because it is a complex process. This stage must frequently be completed while numerous systems and devices are connected. In other cases, more time and effort may be required. The deployment phase could at first appear to be the last action. Nothing could be more untrue; this is simply the beginning.

**Maintenance**

The maintenance stage of the SDLC process is probably one of the most important. Now that consumers have actually used the product, you can use the input that they give to create new features, fix any issues that are left out, and close any potential security loop holes. The development team's current job is to sustain the current product by keeping it current with both user preferences and technological demands that the users require.

## ACKNOWLEDGEMENT

We would like to express our deep gratitude to **Mrs. KAVYA M S** for her valuable and constructive suggestions during the planning and development of this project. Her willingness to give her time so generously has been very much appreciated. We would also like to thank all the professors of KSIT for their continuous support and encouragement.

## REFERENCES

[1] A. Rafay, M. Suleman, and A. Alim, “Robust review rating prediction algorithm based on machine and deep learning: Yelp dataset,” in Intern. Conf. on Emerging Trends in Smart Technologies, March 26-27, 2020, pp. 8138-8143.

[2] A. Shah, K. Kothari, U. Thakkar, and S. Khara, “User review classification and star rating prediction by sentimental analysis and machine learning classifiers,” in book: Information and Communication Technology for Sustainable Development, Proceedings of ICT4SD, June 26, 2019, pp. 279-288.

[3] R. Haji, K. Daanyaal, G. Deval, and G. Rushikesh, “Rating prediction based on textual review: machine learning approach, lexicon approach and the combined approach,” in International Research Journal of Engineering and Technology (IRJET), vol. 6 (3), March 2019, pp.5437-5443.

[4]E. Tutubalina, V. Malykh, S. Nikolenko, A. Alekseev, and I. Shenbin, “AspeRa: aspect-based rating prediction algorithm,” in Proceedings of the Workshop on Eidening NLP, 2019, pp. 11-13.

[5] T. Viard, R. Fournier-S’niehotta, “Augmenting content-based rating prediction with link stream features,” Computer Networks, vol. 150, February 26, 2019, pp. 127-133.