Fraudulent Credit Card transaction detection using state-of-the-art Machine Learning and Deep Learning Techniques

Dissertation submitted in partial fulfillment

of the requirements for the degree of

Master of Technology

in

ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

by

EDAWANBIANG DHAR
AP21122040012

Under the Guidance of

Dr Rajiv Senapati

Department of Computer Science and Engineering

SRM University–AP
Neerukonda, Mangalagiri, Guntur
Andhra Pradesh – 522 240
May, 2023

Declaration

I affirm that the project titled " Fraudulent Credit Card transaction detection using state-of-the-art Machine Learning and Deep Learning Techniques " submitted for the Master of Technology degree is entirely my own work. This project has not been used to obtain any other degree, diploma, fellowship, or similar accolades.

Place:
Date:

Signature of the student

Certificate

Date:

I hereby certify that the research presented in this Project titled " Fraudulent Credit Card transaction detection using state-of-the-art Machine Learning and Deep Learning Techniques " has been conducted by Ms. Edawanbiang Dhar under my guidance. The work is authentic, original, and appropriate for submission to SRM University-AP for the purpose of obtaining a Master of Technology degree from the School of Engineering and Sciences.

Dr. Rajiv Senapati **Dr. Jatindra Kumar Dash**

Supervisor **Head of The Department**

Acknowledgements

I would like to express my deep gratitude to my parents, who have been the greatest sources of inspiration in my life. I am truly fortunate to have such loving and encouraging parents. Additionally, I extend my heartfelt thanks to the rest of my family and friends. Their presence and unwavering belief in my abilities have pushed me to go beyond my limits and give my best in everything I do. Lastly, I want to express my sincere appreciation to my advisor, Dr. Rajiv Senapati. His dedication and commitment to my development have been truly remarkable. I am grateful for the countless hours he has spent with me, brainstorming ideas and providing guidance. His mentorship has been invaluable in shaping my journey and helping me discover my true potential.

Student's Signature

Table of Contents

Abstract

Fraudulent activity detection in credit card is one of the important problem in online payment systems. Many credit cards offer various rewards programs, cashback offers, travel benefits, and discounts on purchases. Therefore the number of credit card users have increased day by day. From providing a convenient way to make purchases internationally to record keeping. Several benefits have been provided to the credit card users. However, it is very important to study about the safety, security and the cause of Credit card frauds. Credit card institutions should be able to detect the fraud in real time in order to avoid losses from the card owner and the bank. To detect credit card fraud, our study utilizes both machine learning algorithms and deep learning techniques. By leveraging these approaches, we aim to train models capable of identifying fraudulent transactions by analyzing patterns and anomalies within the data. Our experimentation involves extensive analysis using a dataset containing credit card transactions made by European cardholders in September 2013. The objective of our study is to compare the effectiveness of various machine learning algorithms such as logistic regression (LR), decision trees (DT), random forests (RF), and support vector machines (SVM), along with deep learning models like Feed Forward Neural Networks and CNN models, for credit card fraud detection. By using these techniques, we can improve the overall performance of credit card fraud detection models and reduce the financial losses caused by credit card fraud.

Key words: Machine Learning, Deep Learning, fraud detection, Algorithm.

List Of Table

List Of Figures

Chapter 1

Introduction

Fraudulent activity detection in credit cards is one of the important problem in online payment systems. With the increasing popularity of digital payments and online shopping, the risk of fraud has grown, and fraudsters have become more skilled and elusive. To combat this issue, financial institutions and merchants have implemented fraud detection systems that utilize rule-based methods and human review. However, these methods are time-consuming, expensive, error-prone, and struggle to keep up with evolving fraud tactics. Machine learning (ML) has shown promise in detecting credit card fraud by automatically learning patterns and anomalies from large datasets of transactional data. This thesis aims to contribute to the field of credit card fraud detection by investigating the performance and feasibility of various machine learning techniques. The study will focus on developing and evaluating machine learning-based systems that can accurately and efficiently detect credit card fraud in real-time, minimizing false positives. Different machine learning algorithms and techniques for feature selection, data preprocessing, and model optimization will be explored and compared in terms of accuracy, efficiency, and interpretability. Additionally, the study will examine the impact of various data sources, such as transactional, contextual, and behavioral data, on the proposed system's performance. The findings will provide insights into the factors that contribute to successful credit card fraud detection using machine learning, helping financial institutions and merchants enhance their fraud detection systems and improve security in electronic payments.

Motivation

Credit card fraud is a significant financial crime where unauthorized individuals exploit someone else's credit card information to carry out transactions without their consent. This problem poses a threat to both individuals and financial institutions, highlighting the need for effective detection and prevention methods. To address this issue, there is a growing demand for advanced and automated fraud detection techniques using machine learning. However, several challenges remain in the development of robust machine learning models, including data imbalance, feature selection, and model interpretability. Overcoming these challenges is crucial to enhance the fraud detection systems of financial institutions and merchants, leading to reduced losses and improved security and trust in electronic payments. This research centers on the examination of credit card fraud detection using machine learning techniques, investigating the obstacles that are involved in this domain, emphasizing the importance of feature engineering, and presenting various machine learning algorithms commonly employed in this domain. Furthermore, a case study is provided to demonstrate the effectiveness of machine learning algorithms in detecting credit card fraud.

Objective

The main objectives of this thesis are as follows:

- Design and assess a credit card fraud detection system based on machine learning, aiming to combat fraudulent activities associated with credit cards.

- Primarily the objective is to develop a system that utilizes machine learning model to effectively identify fraudulent transactions in real-time. The system should achieve high accuracy in detecting fraud while minimizing false positive rates.

- Explore and compare various machine learning algorithms and techniques: The thesis aims to study and analyze different machine learning algorithms and techniques, specifically focusing on feature selection, data preprocessing, and model optimization. By comparing their performance in terms of detection accuracy, efficiency, and interpretability, we can determine the most suitable approaches for credit card fraud detection.

- Investigate the impact of data sources on system performance: The thesis seeks to examine how different data sources, including transactional, contextual, and behavioral data, influence the performance of the proposed fraud detection system. By understanding the contribution of each data source, valuable insights can be gained to enhance the success of credit card fraud detection using machine learning.

- Contribute to the development of robust fraud detection systems: Ultimately, the findings and conclusions drawn from this thesis will contribute to the advancement of more robust and effective fraud detection systems. By implementing these systems, financial institutions and merchants can better protect their customers from fraud and mitigate financial losses.

- Overall, this thesis aims to tackle the difficulties encountered in the detection of credit card fraud, providing solutions and improvements in the field through the development and evaluation of a machine learning-based system. By exploring different algorithms, investigating data sources, and providing insights into system optimization, the goal is to enhance the

security and trust of electronic payments while reducing financial risks for individuals and organizations.

Thesis Organization

The rest of this thesis is organized this way as follows: Chapter 1 briefly discusses about the background of the Study and problems are identified; Chapter 2 covers the latest Relevant literature; Chapter 3 presents the proposed framework and research; Chapter 4 discusses the performance and evaluation of the algorithms; Chapter 5 concludes the thesis.

Chapter 2

STATE-OF-THE-ART TECHNIQUES AND LITERATURE

Machine learning is an area of research that focuses on utilizing computational algorithms to transform empirical data into practical models. Its primary goal is to enhance the performance of computer systems by analyzing data and developing algorithms and statistical models. By recognizing patterns and leveraging data inputs, machine learning algorithms can make predictions. These algorithms undergo training using labeled data, wherein they are presented with examples of expected inputs and outputs for the given task. Through this process, the machine learning algorithm learns to identify patterns and correlations between inputs and outputs, enabling it to make decisions on new unlabeled data. Supervised learning is a specific type of machine learning where algorithms learn from labeled data, encompassing pairs of input and output, to make predictions on unseen data. In supervised learning, the algorithm is trained on a dataset that contains pre-determined input and output information, referred to as features and targets respectively.

Credit card companies face the challenge of identifying transactions that are not made by the card owner in real-time, which can vary in nature, and the number of fraud cases is typically low compared to non-fraudulent transactions, resulting in class imbalance. However, machine learning models are well suited for handling data with different classes. Evaluations conducted in previous research [5-10] have demonstrated improved results. These studies have performed various experiments by utilizing deep learning techniques and implementing data balancing methods to minimize false negative

rates. Initially, a machine learning algorithm was applied to the dataset, resulting in enhanced detection accuracy. Subsequently, deep learning techniques, including a simple feed-forward neural network architecture, were employed to further improve the performance of fraud detection.

Literature

The literature review will cover the current state of credit card fraud detection techniques, with a focus on machine learning-based approaches.

In [1], machine learning and deep learning algorithms were applied and compared with traditional machine learning methods, achieving improved accuracy, precision, and AUC curves. The proposed approaches show promise for real-world credit card fraud detection.

In a recent study, a novel technique is presented [2], which demonstrates a highly effective strategy for fraud detection. The results of this method surpass those of alternative algorithms, exhibiting exceptional sensitivity of 99.6% and specificity of 99.8% respectively.

The study by Kalid et al. [3] focuses on anomaly detection in credit card data, specifically addressing default payments and fraud as the main anomalies. The researchers propose using a Multiple Classifiers System (MCS) to deal with the challenges posed by overlapping class samples and imbalanced class distribution. By employing this combination strategy, the MCS approach demonstrates accurate detection of credit card transaction anomalies.

In the realm of credit card fraud detection within the e-commerce domain, an insightful investigation was conducted in [4], specifically addressing the application of transfer learning techniques across different countries. The study compares and evaluates fifteen distinct approaches, shedding light on the significance of labeled samples within the target domain. In order

to tackle this challenge effectively, the paper introduces an ensemble solution that leverages self-supervised and semi-supervised domain adaptation classifiers. This ensemble solution not only demonstrates remarkable accuracy but also showcases robustness when confronted with varying numbers of labeled samples.

This study[5] utilizes credit card datasets from previous research works to design and implement the MCS approach.The specific characteristics of credit card data, along with the complexity of learning algorithms, have contributed to the challenge of achieving high anomaly detection rates. As a result, relying on single classifiers alone may not yield satisfactory classification results.

This study[6] proposes behavior- and segmentation-based features informed by financial expertise, providing cause-effect relationships and accurate predictions. Time-inhomogeneity influences data imbalance handling and requires careful resampling techniques such as simple oversampling and GANs for synthesizing fraudulent samples.

Cui, Jipeng, Chungang Yan, and Cheng Wang in [7] through their experiment concluded that traditional anomaly detection for online banking fraud faces limitations in terms of limited historical behavior data, heterogeneous attribute values, and highly skewed transaction data. To address these challenges, a ReMEMBeR model is proposed, approaching fraud detection by treating it as a problem akin to a pseudo-recommender system. It leverages collaborative filtering, embedding-based methods, and multi-contextual behavior profiling to achieve superior performance compared to benchmarks on all metrics.

This study[8] addresses the gap in understanding fraudulent card transactions by constructing the largest dataset with 4B non-fraud and 245K fraud transactions from 35 Turkish banks. Fraud detection models based on profiling, encompassing card type, transaction attributes, and amount-based approaches, are introduced and their performance is evaluated. Temporal and spatial analysis showcases the models' resilience against aging and zero-day attacks.

This article[9] focuses on enhancing the performance and stability of fraud detection models by obtaining deep feature representations using a novel loss function called full center loss (FCL). Through extensive experiments on large credit card transaction datasets, FCL demonstrates superior detection performance and model stability compared to other state-of-the-art loss functions

Dal Pozzolo et al. [10] conducted an experiment in credit card fraud detection and encountered various challenges due to fraudsters constantly changing their strategies over time.

This research[11] utilizes a real-world credit card dataset to detect fraud transactions. Three ML algorithms (RF, LR, and AdaBoost) are employed, with Random Forest achieving the highest accuracy and Mathews Correlation Coefficient (MCC) score. The ML web application is built using the Streamlit framework.

Carrasco and Sicilia-Urbán [12] investigated the benefits of deep learning algorithms for credit card fraud detection. They found that deep learning models excel in real-time fraud detection compared to traditional rule-based systems, which often require manual intervention and struggle to detect new types of fraud. The study evaluated multiple deep learning neural networks to measure their effectiveness in reducing false positives triggered by fraud detection systems. The optimal setting achieved a significant reduction in false positives, capturing 91.79% of fraudulent cases with 35.16% fewer alerts. The cost efficiencies and improved detection achieved through automation using deep learning algorithms were highlighted.

Ileberi, Emmanuel, Yanxia Sun, and Zenghui Wang developed a machine learning framework in [13] for fraud detection using real-world imbalanced datasets. We employed SMOTE for dataset resampling and evaluated various ML algorithms with AdaBoost. Results showed boosted models outperformed existing methods.

This paper[14] conducts a comprehensive experimental study on imbalance classification solutions and machine learning algorithms for fraud detection. Findings highlight the weaknesses of current approaches, including high false alarm rates and inaccurate detection, leading to increased fraud occurrences and financial costs.

The work reported in [15] introduces an intelligent approach, OLightGBM as a hyperparameter optimization algorithm. It achieves exceptional performance with an accuracy, AUC, precision and F1-Score of 98.40%, 92.88%, 97.34%, and 56.95% respectively.

This study[16] extensively examines techniques for detecting and preventing cybercrimes. It explores the types of cybercrimes, their threats to computer systems, strategies employed by cybercriminals, and existing detection/prevention methods. The study assesses the strengths, vulnerabilities, and offers recommendations for developing an effective cybercrime detection model. Cybercrimes encompass various illegal activities that target computer systems or communication tools, utilizing them as instruments or being closely associated with the prevalence of technology.

As estimated by Nilson in [17] in 2020, global fraud loss is predicted to surpass $35 billion by 2025. Thus, there is an urgent need for innovative methods to combat this issue. This research focused on analyzing additionally The Vesta Corporation has provided the IEEE-CIS Fraud Detection Dataset for the study. Instead of detecting fraudulent transactions, the study aimed to predict fraudulent credit cards by employing a user separation approach. The proposed model utilized CatBoost and Deep Neural Network techniques for old and new users, respectively. The paper also elaborated on techniques to enhance detection accuracy, including addressing imbalanced datasets, feature transformation, and engineering. Experimental results demonstrated the model's effectiveness, with AUC scores of CatBoost is 0.97 and Deep Neural Network is 0.84.

In the study conducted by Pradhan et al. [18], an extensive comparison of ML algorithms for credit card fraud detection was carried out. The evaluated algorithms included ANN, RF, SVM, DT, LR, Gaussian Naive Bayes, and K-NN. Notably, the results revealed that the Artificial Neural Network algorithm outperformed the others in accurately distinguishing between fraudulent and non-fraudulent transactions, as evidenced by metrics such as Recall, Accuracy, and Precision. This comprehensive evaluation and comparison of algorithm performance underscored the effectiveness of the ML approach in tackling credit card fraud detection.

Singh et al. [19] conducted a review study to identify commonly used techniques for credit card fraud detection, assessing their effectiveness and associated limitations or challenges. They evaluated different algorithms and techniques, including DT, LR, RF, SVM, NB, K-NN, and Gradient Boosting Classifier. The review also highlighted recent advancements in the field and their potential impact on credit card fraud detection.

In [20], the focus is on the utilization of neural learning techniques to tackle classification difficulties arising from unbalanced and noisy data. To overcome the identified challenges, the study proposes a methodology for generating additional training data examples around noise densities. The authors argue that by doing so, neural networks can achieve improved generalization capabilities and enhanced control over classification errors.

In [21] they propose weight-tuning as a pre-processing step and leverage CatBoost, XGBoost, and LightGBM algorithms to enhance the voting mechanism. Additionally, deep learning is applied to fine-tune hyperparameters, including the proposed weight-tuning method. Real-world data experiments are conducted, evaluating the performance using ROC-AUC and recall-precision metrics to account for unbalanced datasets. LightGBM and XGBoost demonstrate exceptional performance, delivering outstanding results with a ROC-AUC score of 0.95. Moreover, they achieve a commendable precision of 0.79, a recall of 0.80, an F1 score of 0.79, and an MCC (Matthews correlation coefficient) of 0.79. These findings highlight the impressive capabilities of both LightGBM

and XGBoost in accurately predicting the target variable. Further improvements are observed when using deep learning and Bayesian optimization, yielding ROC-AUC = 0.94 surpassing state-of-the-art methods.
The objective in [22] is to develop a method that generates test data and effectively detects fraudulent transactions using this algorithm. By employing genetic algorithms, the algorithmic approach implicitly generates results by iteratively evolving potential solutions. This research paper specifically investigates the detection of credit card fraud, placing emphasis on evaluating the algorithm's performance based on its principles and methodologies

In the research paper [23], a pioneering approach leveraging deep learning techniques is proposed to effectively identify fraudsters in credit card transactions. The study centers around the utilization of Kaggle's credit card dataset and introduces a specialized model specifically designed to accurately classify transactions as legitimate or fraudulent. This model, named OSCNN (Over Sampling with Convolution Neural Network), combines over-sampling preprocessing methods with a convolutional neural network (CNN). Additionally, to ensure a thorough analysis, the dataset is also evaluated using the MLP (Multi-layer perceptron) algorithm for comparative evaluation.

In the research paper [24], they present the results of our investigation into neural learning techniques aimed at tackling classification issues associated with imbalanced and noisy data.

In the study [25], they tackle the challenge of imbalanced data by implementing the synthetic minority oversampling technique (SMOTE) to augment the number of fraudulent instances in the dataset. To assess the efficacy of our model, we utilize precision, recall, accuracy, and F1 score as evaluation metrics. Notably, when considering a feature correlation threshold of 0.1, the SVM classifier exhibits the highest recall score, achieving an impressive 88.55%. Furthermore, when integrating SMOTE into the approach, the k-NN classifier demonstrates the highest F1 score and precision. By employing various classifiers and effectively

addressing the issue of imbalanced data through SMOTE, we achieve promising results in terms of recall, precision, and F1 score.

In [26], the study reveal that in cases where the data is significantly affected by these factors, re-sampling methods that specifically target these challenges, such as NCR and our proposed SPIDER2, exhibit superior performance compared to oversampling methods. Importantly, these findings are consistent when applied to real-life datasets as well, as visualizations using PCA (Principal Component Analysis) indicate the existence of noisy examples and substantial overlap between classes.

This research paper[27] aims to propose a novel method for detecting fraud in streaming transaction data. The main goal is to analyze the transaction history of customers and identify their behavioral patterns. To achieve this, the customers are initially grouped into clusters based on their transaction amounts. A sliding window strategy is then utilized to aggregate transactions from these different clusters, allowing for the extraction of behavioral patterns within each group. Finally, distinct classifiers are trained on these groups individually.

This paper [28] focuses on utilizing web frameworks to deploy ML and DL models as local web services. It covers three key areas, namely the implementation of web frameworks, hosting models as web services, and the deployment process, enabling the integration of machine learning models into web-based applications.

This paper[30] aims to provide a comprehensive review of the research development in learning from imbalanced data. Furthermore, the study emphasize the significant opportunities, challenges, and potential research avenues that can serve as inspiration for future investigations in this field.

In the research paper [31], we address the issue of class imbalance in real-world classification tasks and object detection. Our study introduces a deep neural network that incorporates cost sensitivity, to enable robust feature representations for majority and minority classes. Through extensive experimentation, we showcase the

superior performance of our approach compared to baseline algorithms. Moreover, we conduct evaluations comparing our method to commonly employed data sampling techniques and cost-sensitive classifiers, providing valuable insights into its effectiveness. This work contributes to the field by offering an effective solution for handling class imbalance and improving the overall performance of credit card fraud detection systems.

Chapter 3

METHODOLOGY

In this section, we provide details about the dataset considered for this research work followed by proposed framework, implementation and model training and testing. The data set we employed was publicly released by the Machine Learning Group where the study was done at Université Libre de Bruxelles. It has been extensively used in academic research and for the development of fraud detection algorithms, owing to its availability and relevance.

SL. NO	FEATURES	DESCRIPTION
1	TIME	Duration measured in seconds.
2	AMOUNT	This feature denotes the transaction amount.
3	V1, V2……. V28	These 28 columns correspond to the principal components derived by applying PCA.
4	CLASS	This binary class feature categorizes transactions into two distinct labels: 1 represents fraudulent transactions, while 0 corresponds to non-fraudulent transactions.

Table 1. Data representation.

PROPOSED FRAMEWORK

Researchers are actively exploring this particular field i.e., Credit Card Fraud Detection by investigating diverse algorithms and approaches to enhance the effectiveness of detection methods. Figure 1 illustrates a proposed framework highlighting different steps and methods under evaluation. The study incorporated several commonly utilized ML methods including LR, DT, RF, Gradient Boosting Classifier, NB Classifier, and K-NN Classifier are employed in these studies. These methods often utilize sample training data that has been collected and organized into databases by prominent research organizations (refer to Figure 1).

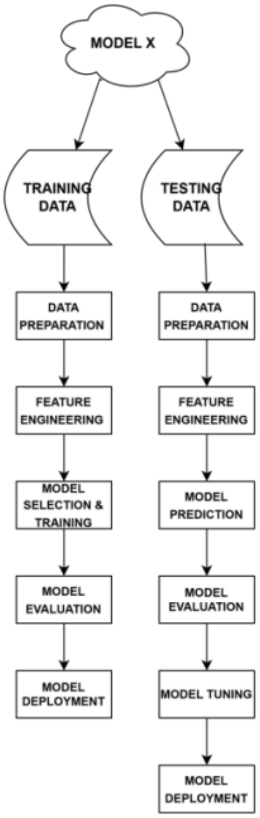


Figure 1. Proposed Model

Data Collection: The process involves gathering credit card transaction data, specifically focusing on fraudulent and non-

fraudulent transactions. The data is typically obtained from real-world that is publicly available datasets.

Out[7]:

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V2
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.110474	0.06692
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.101288	-0.33984
2	1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.909412	-0.68928
3	1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.190321	-1.17557
4	2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	-0.009431	0.798278	-0.137458	0.14126
284802	172786.0	-11.881118	10.071785	-9.834783	-2.066656	-5.364473	-2.606837	-4.918215	7.305334	1.914428	...	0.213454	0.111864	1.014480	-0.50934
284803	172787.0	-0.732789	-0.055080	2.035030	-0.738589	0.868229	1.058415	0.024330	0.294869	0.584800	...	0.214205	0.924384	0.012463	-1.01622
284804	172788.0	1.919565	-0.301254	-3.249640	-0.557828	2.630515	3.031260	-0.296827	0.708417	0.432454	...	0.232045	0.578229	-0.037501	0.64013
284805	172788.0	-0.240440	0.530483	0.702510	0.689799	-0.377961	0.623708	-0.686180	0.679145	0.392087	...	0.265245	0.800049	-0.163298	0.12320
284806	172792.0	-0.533413	-0.189733	0.703337	-0.506271	-0.012546	-0.649617	1.577006	-0.414650	0.486180	...	0.261057	0.643078	0.376777	0.00879

10 rows × 31 columns

Table 2. Features of the Dataset

Data Preprocessing: This step involves preparing the collected data for analysis. It includes tasks like data cleaning, handling missing values, dealing with outliers, and ensuring data consistency. Additionally, data normalization or scaling may be applied to ensure fair treatment of different features. Data preprocessing plays a crucial role in credit card fraud detection as it ensures the integrity, accuracy, and suitability of the collected data for machine learning algorithms. By employing data preprocessing techniques, the fraud detection system can be fine-tuned to effectively identify and prevent fraudulent activities, thereby minimizing financial losses. This process also helps in instilling trust and fostering customer loyalty by ensuring the system's reliability and accuracy in detecting fraudulent transactions.

1. **Data cleaning**: The process of identifying and removing incomplete, incorrect, or irrelevant data points from the dataset is known as data cleaning. This step is crucial as it ensures that the machine learning algorithms can operate on reliable and consistent data.

2. **Data normalization**: This process involves scaling the data to ensure that all features have a similar range of values.

from dominating the training process and to enhance the accuracy of the model's predictions.

3. **Data transformation**: Data transformation focuses on converting categorical data into numerical representations or suitable forms for applying ML algorithms.

4. **Feature selection**: It is the process of choosing the most relevant features that are highly informative for the machine learning algorithms. By selecting the most meaningful features, the model's accuracy is improved, and the dimensionality of the dataset can be reduced, resulting in better computational efficiency.

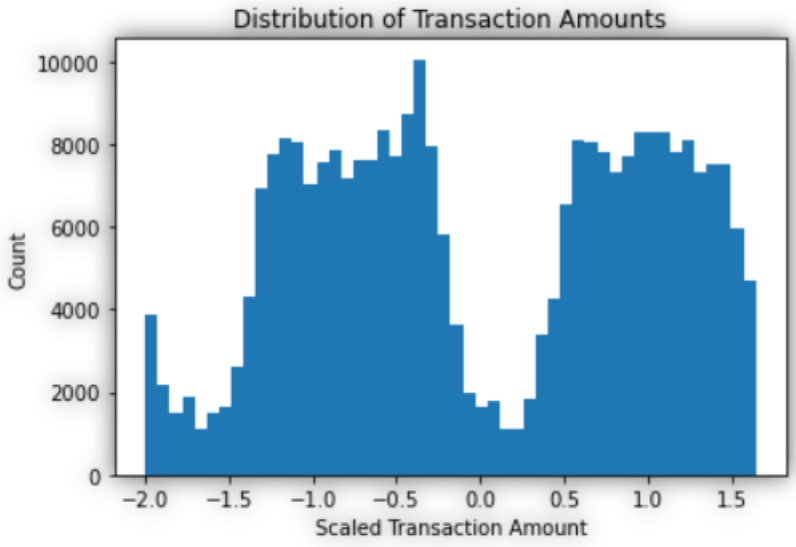


Figure 2. Graphical Distribution of Transaction Amounts

Feature Engineering: In this phase, relevant features are extracted or constructed from the available data. It involves techniques such as dimensionality reduction, transforming variables, creating new features based on domain knowledge, or incorporating external data sources.

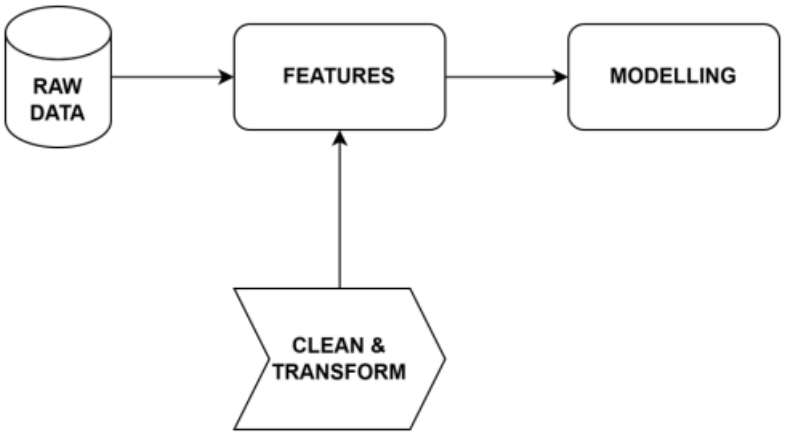


Figure 3. Applying Feature Engineering in the given data.

Model Selection: Various ML models are explored to identify the most suitable algorithm for fraudulent activity detection in credit card. The classification algorithms employed in this study include LR, DT, RF, Gradient Boosting Classifier, NB Classifier, and K-NN selection is based on factors like model performance, interpretability, computational efficiency, and specific requirements of the problem.

Evaluation: The selected model is evaluated using the formance metrics such as accuracy, precision, recall, F1-score, and AUC. Cross-validation or train-test splits are employed to assess the model's generalization ability and mitigate overfitting. The model's performance is compared with the earlier models available in the literature.

Implementation and Model Training

Logistic Regression

Logistic regression is a widely used statistical technique in credit card fraud detection. It is a supervised learning algorithm that is commonly employed to classify data into two classes: fraud and non-fraud. In contrast to linear regression, which predicts

continuous numerical values, logistic regression is utilized to estimate the probability of an event taking place. In credit card fraud detection, logistic regression identifies suspicious transactions and flag them as potential fraud. The performance is presented in the below table.

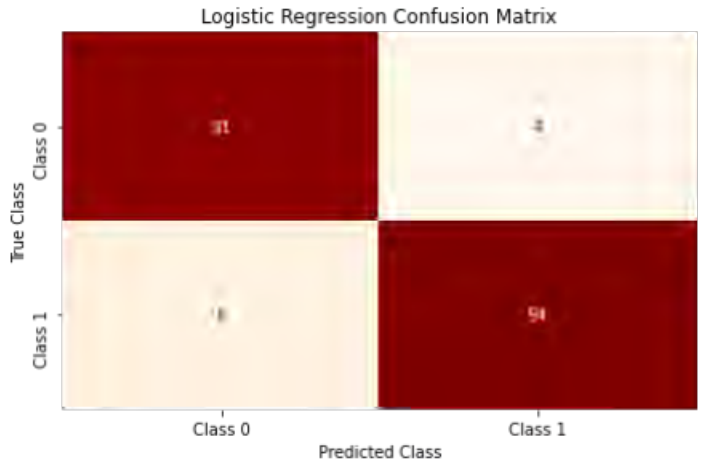


Figure 4. Logistic Regression Confusion matrix

Model	Precision score	Recall Score	Accuracy Score
Logistic Regression	0.936363	0.892157	0.934010

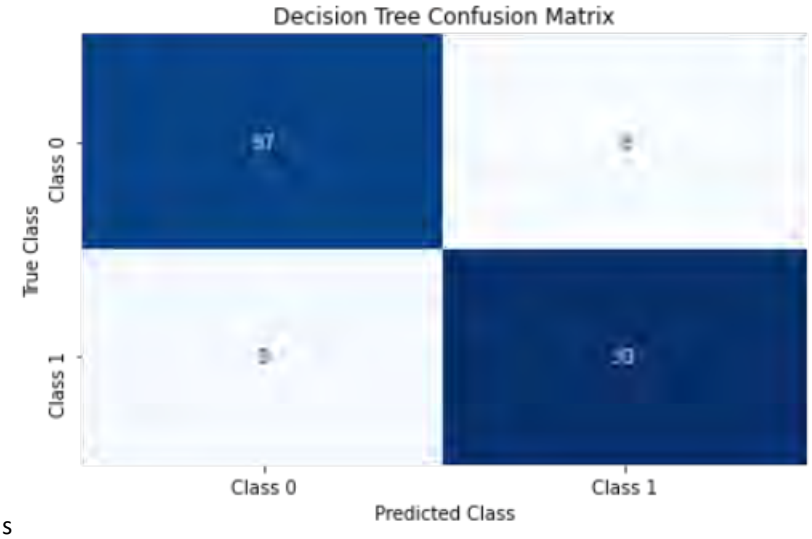
Table 3: LR Score

Decision Tree

Decision trees are widely employed in credit card fraud detection due to their capability to handle categorical and numerical data effectively. Additionally, decision trees offer interpretable outcomes, further contributing to their popularity in this domain. Decision trees create a hierarchical structure of decision rules using the features in the dataset, allowing for efficient fraud classification. The performance metrices is presented in the below table.

Model	Precision Score	Recall Score	Accuracy Score
Decision Tree	0.956731	0.911764	0.954315

Table 4: DT Score



s

Figure 5. Decision Tree Confusion Matrix

Random Forest

Random Forest is widely recognized as a favored machine learning algorithm in credit card fraud detection, primarily attributed to its proficiency in effectively handling high-dimensional data. Moreover, Random Forest excels in capturing intricate relationships within the data and delivering robust predictions, further enhancing its appeal in this domain. It combines multiple decision trees to improve the overall performance and generalization of the model. The performance is presented in the below table.

Model	Precision Score	Recall Score	Accuracy Score
Random Forest	0.936363	0.892157	0.934010

Table 5: RF score

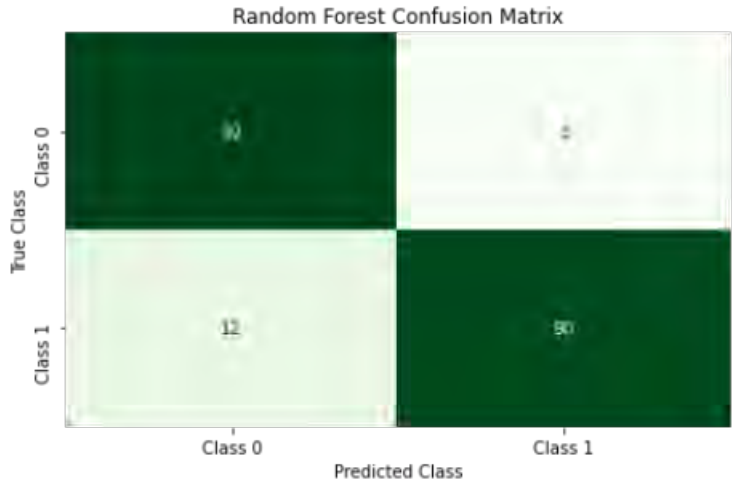


Figure 6. Random Forest Confusion Matrix

Support Vector Machines (SVM)

SVM is a highly potent machine learning algorithm commonly employed in credit card fraud detection. Its effectiveness stems from its capability to effectively handle high-dimensional data and handle class imbalance. SVMs are particularly effective when the data is not linearly separable and when the focus is on identifying the boundary between fraud and non-fraud cases. The model's performance is presented in Table 5.

Model	Precision Score	Recall Score	Accuracy Score
SVM	0.918608	0.911765	0.918782

Table 6: SVM score

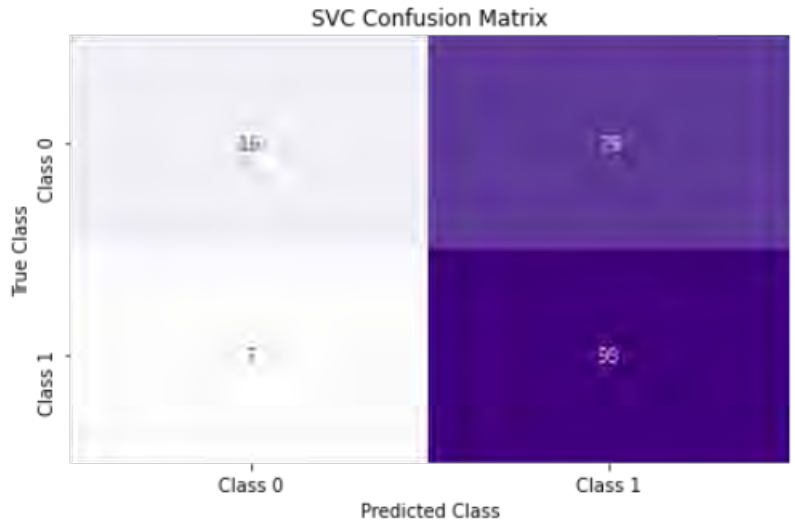


Figure 7. SVC Confusion Matrix

K-NEAREST NEIGHBORS CLASSIFIER

K-NN is a popular machine learning algorithm utilized in credit card fraud detection. It is favored for its simplicity and effectiveness in handling non-linear data. KNN classifies new instances based on the majority class observed among their neighboring data points.Here's an explanation of how KNN is used in credit card fraud detection:

Data Collection: Historical transaction data of a credit card is collected, which includes the features such as location, transaction amount merchant type, and cardholder information and time. The dataset is labeled as fraud or non-fraud based on known fraudulent activity.

Data Preprocessing: To prepare the collected data for constructing a KNN model, several preprocessing steps are undertaken. These steps involve addressing missing values, handling outliers, and normalizing numerical features. These actions are crucial to ensure that the data is appropriately formatted and ready for utilization in the KNN model.

Feature Selection/Extraction: Relevant features that are most informative for fraud detection are selected or extracted. To identify the most relevant features, various techniques can be applied, such as correlation analysis, feature importance analysis, or dimensionality reduction methods. These techniques help in determining which features have a strong correlation with the target variable, which features contribute significantly to the predictive power of the model, or which features can be reduced or combined to reduce dimensionality without losing important information. By employing these methods, the most relevant features can be identified and selected for further analysis or model development.

Model Training: The KNN algorithm does not require explicit training. Instead, it stores the entire training dataset as the model

itself. The algorithm simply memorizes the training instances and their corresponding class labels.

Distance Metric Selection: To assess the similarity between instances, it is necessary to choose an appropriate distance metric, such as Euclidean distance or Manhattan distance. These distance metrics serve as measures to quantify the dissimilarity or similarity between data points in the dataset. By selecting a suitable distance metric, we can effectively evaluate the proximity between instances in the context of the KNN model. This choice is crucial in accurately determining the nearest neighbors for a given instance during the prediction process.

Model Evaluation: Evaluate the KNN model using a validation set or through cross-validation. Calculate metrics such as accuracy, precision, recall, and F1-score to assess the model's performance in fraud detection.

Model Deployment: If the KNN model demonstrates satisfactory performance, it can be deployed in a real-time credit card fraud detection system. New incoming credit card transactions can be classified as fraud or non-fraud based on the majority class label of their K nearest neighbors.

Regular Model Updates: Credit card fraud patterns evolve over time, so it is crucial to regularly update and retrain the KNN model with new labeled data. This helps the model adapt to changing fraud patterns and maintain its effectiveness. Evaluate the KNN model using performance matrices, which is presented in the below table.

Model	Precision Score	Recall Score	Accuracy Score
KNN	0.942184	0.940815	0.939086

Table 7: KNN Score

Figure 8. K-Nearest Neighbors Confusion Matrix

GRADIET BOOSTING

Gradient Boosting is a highly effective machine learning technique utilized in the field of fraud detection . It is renowned for its remarkable capability to effectively handle intricate patterns, address the issue of class imbalance, and deliver exceptional predictive accuracy. It combines weak predictive models, typically decision trees, into an ensemble model that makes accurate predictions by iteratively correcting the mistakes of the previous models. The performance metrics is presented in the below table.

Model	Precision score	Recall Score	Accuracy Score
Gradient Boosting	0.949340	0.931373	0.949239

Table: 8 GB Score

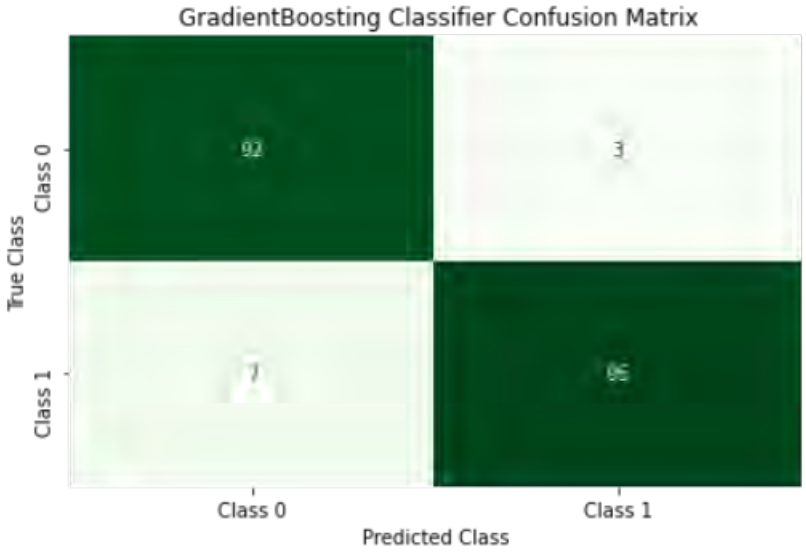


Figure 9. GradientBoosting Classifier Confusion Matrix

Naive Bayes

Due to its effectiveness in handling high-dimensional data, computational efficiency, and simplicity, Naive Bayes is a commonly employed algorithm in credit card fraud detection. It has gained widespread usage in this domain. It relies on Bayes' theorem to calculate the probability of an event happening, considering the probabilities of related events. Here's an explanation of how Naive Bayes is used in credit card fraud detection:

Data Collection: Data from past credit card transactions is gathered, comprising various features like merchant type, and cardholder information, transaction amount, location and time. The dataset is labeled as fraud or non-fraud based on known fraudulent activity.

Data Preprocessing: The collected data is further preprocessed by dealing with the missing values, the outliers and normalizing numerical features. This step ensures that the data is appropriately prepared and structured to create a Naive Bayes model.

Feature Selection/Extraction: Relevant features that are most informative for fraud detection are selected or extracted. Various

techniques can be utilized to determine the most significant features, including correlation analysis, feature importance analysis, or dimensionality reduction methods. Handling Class Imbalance: Credit card fraud cases are typically rare events, resulting in class imbalance in the data.

Model Training: The Naive Bayes algorithm learns the probabilities of different features given each class (fraud or non-fraud) based on the training data. The algorithm makes the assumption that all features are conditionally independent of each other when considering the class label. This simplifying assumption, known as the "naive" assumption, enables efficient computations.

Probability Estimation: For a new credit card transaction, Naive Bayes calculates the probability of it belonging to each class (fraud or non-fraud) based on the learned probabilities from the training data. By leveraging Bayes' theorem, the Naive Bayes algorithm computes probabilities by multiplying the prior probability of a class with the conditional probabilities of each feature given that class.

Classification: Naive Bayes assigns the new transaction to the class with the highest probability. If the probability of fraud is higher than a predefined threshold, the transaction is classified as fraud; otherwise, it is classified as non-fraud.

Model evaluation: To evaluate the Naive Bayes model in fraud detection, The model can be evaluated either by using a validation set or through cross-validation. Performance metrics, including accuracy, precision, recall, and F1-score, can be computed to gauge the efficacy of the model.

Model Deployment: If the Naive Bayes model demonstrates satisfactory performance, it can be deployed in a real-time fraud detection system. New incoming transactions can be classified as fraud or non-fraud based on the model's probability estimates.

Regular Model Updates: Credit card fraud patterns evolve over time, so it is crucial to regularly update and retrain the Naive

Bayes model with new labeled data. This helps the model adapt to changing fraud patterns and maintain its effectiveness. The performance metrics is presented in the below table.

Model	Precision score	Recall Score	Accuracy Score
Naive Bayes	0.933697	0.931011	0.949239

Table 9: NB Score

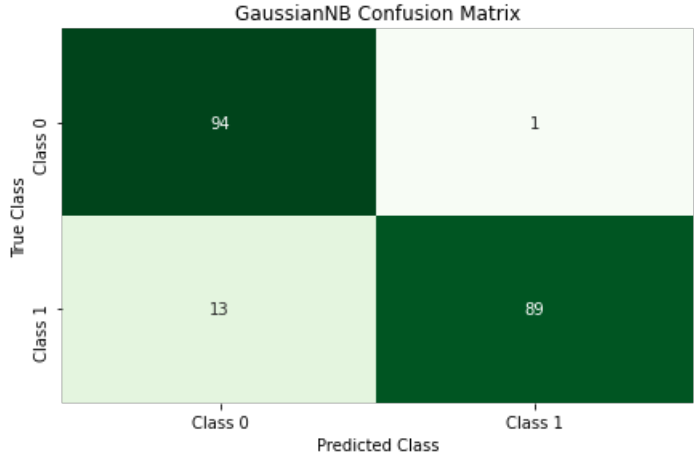


Figure 10. Gaussian NB Confusion Matrix

Deep Learning Model

Feed Forward Neural Network

The feedforward neural network model described here is commonly used for binary classification tasks. It consists of three dense layers: an input layer with 30 units. Consider a neural network architecture with 30 input features. The network consists of two hidden layers, the first one having 32 units and the second one having 16 units. The output layer contains a single unit and utilizes the sigmoid activation function.

It is crucial to mention that the input data should be preprocessed adequately before training the particular model, including steps

like scaling and handling missing values, before feeding it to the model.

Please note that the architecture assumes 30 input features, but you can modify it based on the specific requirements of your dataset. Additionally, you may need to adjust the hyperparameters for instance, one should consider factors like the hidden layers (the number of layers) and the number of units present in each layer, the activation functions, the optimizer, and the loss function based on your specific problem and data characteristics.

Model	Training Score	Test score	Precision Score	Recall Score	Accuracy Score
Feed Forward Neural Nework	0.99965	0.99942	0.83720	0.76470	0.99942

Table 10: NN Score

Chapter 4

RESULTS AND DISCUSSIONS

Our study involved evaluating various machine learning models using a dataset consisting of 284,807 credit card transactions. Within this dataset, there were 492 transactions labeled as fraudulent, making up 0.172% of the total and representing the positive class. The implementation of all experiments was carried out using the sci-kit-learn package. Our research primarily aimed to determine the optimal model for detecting credit card fraud. By conducting experiments, our objective was to showcase the efficacy of different machine learning techniques specifically applied to this field. Additionally, we examined the impact of data pre-processing techniques on the models' performance.

We found that the Decision Tree Model achieved the highest score across four criteria: Precision, Recall, and Accuracy Score. Generally, higher values for Precision, Recall, and Accuracy Score indicate better performance. In summary, our study compared the performance of different ML models to determine the most effective approach.

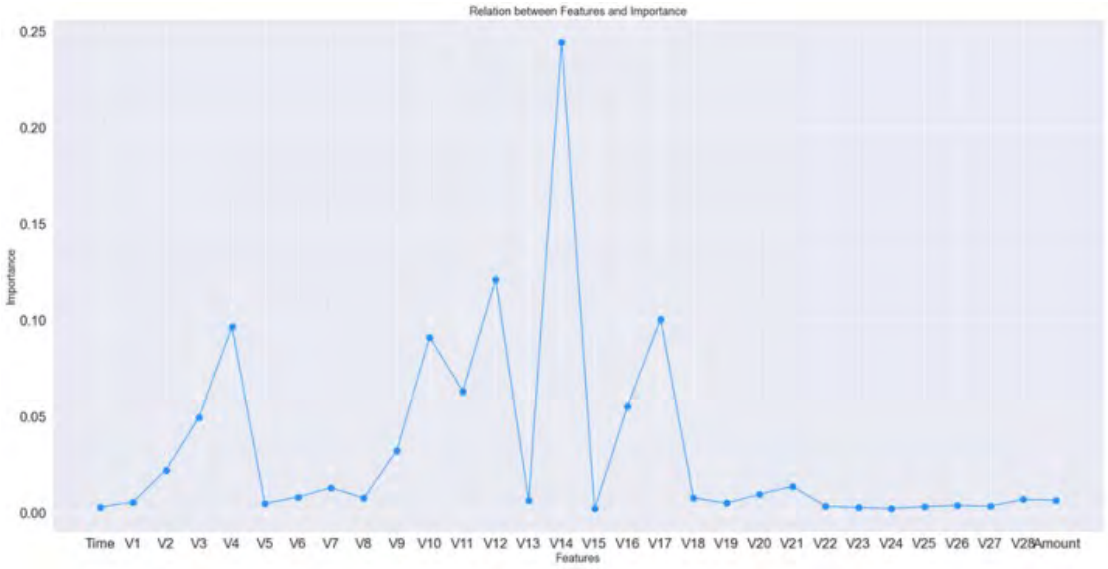


Figure 11. Relation between features

Experimental Results

The outcomes obtained through our experimental results by applying different ML techniques are presented in the below table based on the evaluation matrices.

SL NO	MODEL	TRAIN SCORE	TEST SCORE	PRECISION SCORE	RECALL SCORE	ACCURACY SCORE
1	GradientBoosting Classifier	100.00	94.92	0.949340	0.931373	0.949239
2	Decision Tree	98.09	91.37	0.956731	0.911765	0.954315
3	Random Forest	96.32	94.42	0.936363	0.892157	0.934010
4	KN-Neighbors Classifier	94.03	93.91	0.942184	0.940815	0.939086
5	Logistic Regression	93.65	93.40	0.936363	0.892157	0.934010
6	Naive Bayes classifier	91.61	92.89	0.933697	0.931011	0.949239
7	SVM	62.90	58.38	0.918608	0.911765	0.918782
8	Feed forward Neural network	99.6	99.9	0.837209	0.7941176	0.99942

Table 11: Comparisons of different algorithms

The experimental results in credit card fraud detection involve comparing different algorithms or variations of the same algorithm to determine their respective performance in terms of accuracy and other evaluation metrics. These results often include visualizations like ROC curves or precision-recall curves, which illustrate the trade-off between true positive rate and false positive rate, or precision and recall. These visualizations help in evaluating the models' overall performance across various thresholds.

The experimental results provide valuable insights into the performance, strengths, and limitations of the applied techniques in credit card fraud detection. They serve as a guide for researchers and practitioners in making informed decisions about the most effective approaches for detecting and mitigating credit card fraud. By analyzing the results, they can identify the algorithms or techniques that yield better accuracy and choose the ones that align with their specific requirements and constraints.

Pros and Cons of Machine Learning Algorithms

ALGORITHM	PROS	CONS
GRADIENT BOOSTING CLASSIFIER	● High accuracy ● It handles missing data and outliers. ● Can handle both numerical and categorical data. ● Provides feature importance scores.	● Slow training time and high computational requirements. ● Sensitive to overfitting. ● Difficult to interpret the model.
DECISION TREE	● Easy to understand and interpret. ● Fast training time and low computational requirements. ● It can work with numerical as well as categorical data. ● It can handle missing values.	● Prone to overfitting. ● Unstable in case of small variations. ● Can create biased trees if the data is imbalanced.
RANDOM FOREST	● High accuracy	● Slow training time and

	• Can handle missing data and outliers. • Reduces overfitting by combining multiple decision trees. • Provides feature importance scores.	high computational requirements. • Difficult to interpret the model.
KN-NEIGHBORS CLASSIFIER	• Simple and easy to implement. • Non-parametric • It can work with numerical as well as categorical data.	• Slow prediction time • Sensitive noisy data. • careful selection of k value is required.
LOGISTIC REGRESSION	• Simple • Provides probabilistic predictions and can handle imbalanced data. • Interpretable coefficients.	• It presents a linear relationship • Cannot handle nonlinear relationships or interactions between features. • Requires careful feature selection and data preprocessing.
NAIVE BAYES CLASSIFIER	• Simple • Fast training and prediction time. • Can handle a large number of features and high-dimensional data. • Robust to irrelevant features.	• Assumes features as if they are independent. • Can't model complex relationships.
SVM	• Effective in case of high-dimension feature space. • Capable in handling non-linear relationships.	• SVMs can be computationally expensive. • Difficult to interpret. • Sensitivity to parameter tuning.

	SVMs are less sensitive to outliers compared to some other algorithms.	• Lack of probability estimates.
FEED FORWARD NEURAL NETWORK	• Can capture complex non-linear relationships in the data. • It can work with numerical as well as categorical features. • Can identify complex patterns involving multiple features and interactions.	• computationally expensive, especially for large networks or datasets • prone to overfitting

Table 12. Pros and cons of the Algorithms

Chapter 5

Conclusion and Future Work

In summary, the detection of credit card fraud presents a substantial challenge, emphasizing the utmost importance of effectively identifying and preventing fraudulent transactions to minimize financial losses. The application of machine learning has gained prominence as an approach to credit card fraud detection due to its capability to analyze large volumes of data and uncover patterns indicative of fraudulent behavior. The primary focus of this study was to assess the efficacy of different machine learning techniques in detecting credit card fraud. Through rigorous evaluation, we have identified the most effective approach for detecting and combating fraudulent activities. Moreover, the study has shed light on the impact of deep learning techniques and assessed the performance of different models. Overall, the findings of this study underscore the potential of machine learning in credit card fraud detection and offer valuable insights into the most effective strategies for identifying fraudulent activity.

Furthermore, there is a need to explore advanced anomaly detection algorithms and techniques that can effectively identify previously unseen and evolving fraud patterns. Anomaly detection methods that possess the ability to adapt and learn from new data patterns without relying on manual rule-based updates would be particularly valuable. This would facilitate the detection of emerging fraud patterns and enable proactive measures to counteract fraud.

In addition, integrating multiple data sources and types, such as transaction data, user behaviour data, and contextual information, holds promise in improving the overall effectiveness of credit card fraud detection. Techniques like data fusion and feature engineering can be employed to combine and extract meaningful insights from diverse data sources, thereby enhancing fraud detection performance.

Lastly, developing methods that enable a better understanding and interpretation of the decision-making process employed by complex models, such as deep learning architectures, would foster trust and confidence in the generated predictions. This would allow human experts to validate the model's findings and potentially uncover new insights into fraudulent behaviour.

In conclusion, this work highlights the potential of ML in addressing fraudulent activities. By continually advancing and refining the application of machine learning techniques, it is possible to enrich fraud detection capabilities, minimize financial losses, and improve overall security in the credit card industry.

References

1. Alarfaj, Fawaz Khaled, et al. "Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms." IEEE Access 10 (2022): 39700-39715.

2. Esenogho, Ebenezer, et al. "A neural network ensemble with feature engineering for improved credit card fraud detection." IEEE Access 10 (2022): 16400-16407.

3. Kalid, Suraya Nurain, et al. "A multiple classifiers system for anomaly detection in credit card data with unbalanced and overlapped classes." IEEE Access 8 (2020): 28210-28221.

4. Lebichot, Bertrand, et al. "Transfer learning strategies for credit card fraud detection." IEEE access 9 (2021): 114754-114766.

5. Zheng, Lutao, et al. "Improved TrAdaBoost and its application to transaction fraud detection." IEEE Transactions on Computational Social Systems 7.5 (2020): 1304-1316.

6. Hsin, Yu-Yen, et al. "Feature engineering and resampling strategies for fund transfer fraud with limited transaction data and a time-inhomogeneous modi operandi." IEEE Access 10 (2022): 86101-86116.

7. Cui, Jipeng, Chungang Yan, and Cheng Wang. "ReMEMBeR: Ranking metric embedding-based multicontextual behavior profiling for online banking fraud detection." IEEE Transactions on Computational Social Systems 8.3 (2021): 643-654.

8. Can, Baris, et al. "A closer look into the characteristics of fraudulent card transactions." IEEE Access 8 (2020): 166095-166109.

9. Li, Zhenchuan, Guanjun Liu, and Changjun Jiang. "Deep representation learning with full center loss for credit card fraud detection." IEEE Transactions on Computational Social Systems 7.2 (2020): 569-579.

10. Dal Pozzolo, Andrea, et al. "Credit card fraud detection: a realistic modeling and a novel learning strategy." IEEE transactions on neural networks and learning systems 29.8 (2017): 3784-3797.

11. Jain, Vipul, H. Kavitha, and S. Mohana Kumar. "Credit Card Fraud Detection Web Application using Streamlit and Machine Learning." 2022 IEEE International Conference on Data Science and Information System (ICDSIS). IEEE, 2022.

12. Carrasco, Rafael San Miguel, and Miguel-Ángel Sicilia-Urbán. "Evaluation of deep neural networks for reduction of credit card fraud alerts." IEEE Access 8 (2020): 186421-186432.

13. Ileberi, Emmanuel, Yanxia Sun, and Zenghui Wang. "Performance evaluation of machine learning methods for credit card fraud detection using SMOTE and AdaBoost." IEEE Access 9 (2021): 165286-165294.

14. Makki, Sara, et al. "An experimental study with imbalanced classification approaches for credit card fraud detection." IEEE Access 7 (2019): 93010-93022.

15. Taha, Altyeb Altaher, and Sharaf Jameel Malebary. "An intelligent approach to credit card fraud detection using an optimized light gradient boosting machine." IEEE Access 8 (2020): 25579-25587.

16. Al-Khater, Wadha Abdullah, et al. "Comprehensive review of cybercrime detection techniques." IEEE Access 8 (2020): 137293-137311.

17. Nguyen, Nghia, et al. "A Proposed Model for Card Fraud Detection Based on CatBoost and Deep Neural Network." IEEE Access 10 (2022): 96852-96861.

18. Pradhan, Sasmita Kumari, et al. "Credit Card Fraud Detection Using Artificial Neural Networks and Random Forest Algorithms." 2021 5th International Conference on Electronics, Communication and Aerospace Technology (ICECA). IEEE, 2021.

19. Singh, Aditi, et al. "Design and Implementation of Different Machine Learning Algorithms for Credit Card Fraud Detection." 2022 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME). IEEE, 2022.

20. Aditi, Aditi, et al. "Credit Card Fraud Detection Using Advanced Machine Learning Techniques." 2022 Fifth International Conference on Computational Intelligence and Communication Technologies (CCICT). IEEE, 2022.

21. Sahin, Yusuf, and Ekrem Duman. "Detecting credit card fraud by ANN and logistic regression." 2011 international symposium on innovations in intelligent systems and applications. IEEE, 2011.

22. RamaKalyani, K., and D. UmaDevi. "Fraud detection of credit card payment system by genetic algorithm." International Journal of Scientific & Engineering Research 3.7 (2012): 1-6.

23. Abd El Naby, Aya, Ezz El-Din Hemdan, and Ayman El-Sayed. "Deep Learning Approach for Credit Card Fraud Detection." 2021 International Conference on Electronic Engineering (ICEEM). IEEE, 2021.

24. Murphey, Yi L., Hong Guo, and Lee A. Feldkamp. "Neural learning from unbalanced data." Applied Intelligence 21.2 (2004): 117-128.

25. K. Alsufyani, A. AlMuallim, M. AlShahrani, A. Alsufyani, O. Alhanaya and A. Zerguine, "Credit Card Fraud Detection via Machine Learning," 2022 19th International Multi-Conference on Systems, Signals & Devices (SSD), Sétif, Algeria, 2022, pp. 64-67, doi: 10.1109/SSD54932.2022.9955815.

26. K. Napierała, J. Stefanowski, and S. Wilk, ‘‘Learning from imbalanced data in presence of noisy and borderline examples,’’ in Rough Sets and Current Trends in Computing. Brookline, MA, USA: Microtome Publishing, 2010,pp. 158–167, doi: 10.1007/978-3-642-13529-3_18.

27. Credit Card Fraud Detection Accessed: Jan. 20, 2022 [Online].Available:https://www.kaggle.com/mlgulb/creditcardfrad.

28. Singh, P., 2021. Machine learning deployment as a web service. In Deploy Machine Learning Models to Production (pp. 67-90). Apress, Berkeley, CA.

29. Machine Learning Group. (2018). Credit Card Fraud Detection.[Online]. Available: https://www.kaggle.com/mlg-ulb/creditcardfraud.

30. H. He and E. A. Garcia, "Learning from imbalanced data," IEEE Trans.Knowl. Data Eng., no. 9, pp. 1263–1284, Jun. 2008.

31.S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel, and R. Togneri,"Cost-sensitive learning of deep feature representations from imbalanced data," IEEE Trans. Neural Netw. Learn. Syst., vol. 29, no. 8,pp. 3573–3587, Aug. 2018.

Multimodal Product Recommendation System using Deep Learning Approaches

A DISSERTATION

submitted in partial fulfillment of the requirements

for the award of the degree of

Master of Technology

in

ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

by

S.DHARANI SABARI

AP21122040005

Under the supervision of

Dr.ASHU ABDUL

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SRM UNIVERSITY–A.P

Mangalagiri–Mandal, Neerukonda

Andhra Pradesh–522240, India.

May, 2023

CERTIFICATE

I hereby certify that the work which is being presented in the M.Tech. The dissertation entitled **"Multimodal Product Recommendation System Using Deep Learning Approaches",** in partial fulfillment of the requirements for the award of the **Master of Technology in Artificial Intelligence & Machine Learning, Department of Computer Science & Engineering** is an authentic record of my own work carried out during a period from September 2021 to May 2023 under the supervision of **Dr.Ashu Abdul, Assistant Professor**, Computer Science & Engineering Department.

The matter presented in this thesis has not been submitted for the award of any other degree elsewhere.

Signature of Candidate

S.Dharani Sabari

AP21122040005

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

——————————

Signature of Supervisor

Dr.Ashu Abdul,

Asst.Prof, Dept.of CSE

ABSTRACT

Product recommendation in E-commerce platforms poses a challenge due to the similarity of displayed recommendations and user preferences. To address this, we propose a novel approach using transformers and a multimodal attention strategy to enhance the precision of product recommendations. In this study, we propose a method for a product recommendation that utilizes image and text vectors. These vectors, along with metadata and user information, are concatenated and stored in a database. Through a transformers encoder, we generate context vectors to capture the relationships between the visual and textual components. Subsequently, the initial data is retrieved from the database and processed by a Multilayer Perceptron. By applying the softmax function, we generate product recommendations based on user preferences and context vectors. Our approach aims to enhance the accuracy and relevance of product suggestions in a personalized manner.

Contents

List of Tables

List of Abbreviations

BERT Bidirectional Encoder Representations from Transformers

CNN Convolutional Neural Network

CF Collaborative Filtering

CB Content based filtering

CV Context Vector

GloVe Global Vectors

MLP Multi Layer Perceptron

RS Recommendation System

PROJECT PHASE 1

Chapter 1

INTRODUCTION

1.1 Motivation

The internet has brought forth numerous multimedia online services, including fashion recommendations, music recommendations, and more. This has led to the development and application of multimodal recommender systems (MRS) due to the progress in multimodal research.MRS is capable of effectively managing and utilizing diverse modalities of information that are intrinsic to multimedia services.

Additionally, MRS has the ability to leverage rich multimodal information of items to address the challenges of data sparsity and cold start commonly encountered in recommender systems. While traditional recommender systems primarily rely on collaborative or side information, which includes item identifiers and tabular features, MRS emphasizes the importance of multimodal features such as images, audio, and text. To simplify, we define MRS as a recommender system that utilizes multimodal features for item recommendations. There is a growing interest in MRS among researchers, highlighting the urgent need for a comprehensive review that surveys and categorizes these systems. While a previous review has made progress in organizing the research based on different modalities in real applications.

our survey takes a different approach by categorizing the techniques employed in MRS. Furthermore, we aim to include recent works to provide readers with insights into the latest advancements in this field. In the following section, we will present

the general procedures and our challenge to enhance the readability of the survey.

1.2 Explaining Recommender System

Intelligent agents [1], have been suggested as a solution for efficiently sifting through vast amounts of information and providing personalized recommendations to individual users. Recommender systems play a pivotal role in suggesting relevant items to users based on their unique profiles. By gathering user ratings and constructing user profiles, recommender systems enable a personalized recommendation experience. User ratings can be provided explicitly or inferred implicitly from the user's actions. In platforms like Movie Lens [2], users rate movies using a five-point scale. Alternatively, intelligent agents like Letizia [3], make use of heuristics to deduce a user's ratings for web pages based on their behavior. For instance, a brief visit to a page might indicate dislike, while saving a page suggests liking. Once an intelligent agent has gathered sufficient feedback from the user, it can generate personalized recommendations accordingly. Two fundamental approaches have arisen for the purpose of making recommendations: Content-based Recommender Systems and Collaborative Recommender Systems. In particular, Recommender systems have played a significant role in providing personalized recommendations by utilizing these two approaches.

In the recommendation process, there are three key components:

i. Background data refers to the information already available to the system prior to initiating the recommendation process.

ii. Input data, which is the information that the user needs to provide to the system for generating a recommendation.

iii. An algorithm that integrates the background and input data to generate relevant suggestions.

1.3 Literature Survey

By analyzing the input items of MRS, we have identified the unified procedures. These procedures encompass **Raw Feature Extracting**, **Feature Interaction**, and **Recommendation**. To provide a clear illustration, let's consider the example

of a movie recommendation.

Raw Feature Extracting involves processing the features of each movie, which can be categorized into tabular features and multimodal features. Tabular features are handled using embedding layers, similar to standard content-based recommendation systems [4]. On the other hand, multimodal features, such as the poster image and textual introduction, are fed into specific modality encoders for further processing. The modality encoders are versatile architectures employed in various domains, like ViT [5] for images and Bert for texts. These encoders extract representations, enabling us to obtain the respective representations v_f v_{image} and v_{text} and for each item, encompassing tabular, image, and text.

Feature Interaction plays a crucial role in our approach. While we obtain representations for different modalities v_f , v_{image}, and v_{text} for each item, these representations exist in separate semantic spaces. Additionally, users exhibit diverse preferences for modalities. To address this, our MRF (Multimodal Representation Fusion) technique aims to combine and interact with multimodal representations. This fusion process enables us to obtain item and user representations, which hold significant importance for recommendation models.

Recommendation Following the second step, we acquire the representations of the user and item, referred to as v_u and v_i. Conventional recommendation models, such as MF, utilize these representations to generate recommendation probabilities. Nevertheless, the issue of data sparsity often hampers the performance of recommendations. To address this challenge, numerous researchers have suggested augmenting the representations by integrating multimodal information.

1.4 Challenges:

Based on the aforementioned procedures, we have identified three key challenges

Challenge 1: Integrating modality features from different semantic spaces and determining preferences for each modality.

Challenge 2: Obtaining comprehensive representations for recommendation models when dealing with data sparsity.

Challenge 3: Optimizing lightweight recommendation models and parameterized

Table 1.1: Category for Multimodal Recommender Systems

Applications	Model	Feature Interaction	Feature Enhancement
General	6	Coarse-grained Attention	CL
	7	Coarse-grained Attention	None
	8,9	User-item Graph + Fine-gained Attention	None
	10	User-item Graph	CL
	11	Item-item Graph	CL
	12	Item-item Graph + Fine-gained Attention	None
	13,14	Knowledge Graph	CL
	15	Knowledge Graph + Fine-gained Attention	None
	16,17	MLP / Concat	DRL
	18	Knowledge Graph + Filtration (graph)	None
	19,20	Item-item Graph	None
Video	21,22	Fine-gained Attention	None
	23	Fine-gained Attention	None
	24	User-item Graph	CL
	25	Knowledge Graph + Fine-gained Attention	None
	26	User-item Graph	None
Fashion	27,28,29	Item-item Graph	CL
	30	Fine-gained Attention	None
	31	Fine-gained Attention	CL
News	32	Combined Attention	None
	33	Fine-gained Attention	None
Restaurant	34,35	MLP/Concat	None

modality encoders. Recent studies have made significant progress in addressing these three challenges and have proposed efficient techniques to tackle them..To organize these advancements, we can categorize them into three challenges:

Feature Interaction, **Feature Enhancement**, and **Model Optimization**.

The first challenge 1, Feature Interaction, primarily focuses on Challenge 1 by employing techniques such as Graph Neural Networks (GNN) and attention mecha-

nisms to fuse and capture different modality features. these works primarily target the second procedure within the MRS. In the domain of feature enhancement research, recent studies have proposed innovative approaches such as contrastive learning and disentangled learning. These techniques aim to enhance the representations of both items and users, particularly when limited data is available. This addresses Challenge 2, in the context of MRS. It is important to highlight that these works are closely intertwined with the recommendation procedure within MRS.To tackle Challenge 3, research in model optimization focuses on designing efficient methods for training lightweight recommendation models and parameterized modality encoders. These efforts span the entire process of MRS. Notably, our work is the first to systematically organize MRS research in this technical manner. We have summarized the existing works based on their respective categories, as presented in Table 1.

1.5 Brief Description of Work

Our objective is to create a group recommender system capable of suggesting movies to a group of users. Our proposed model leverages a rule learning algorithm, specifically based on the RIPPER [36] rule learner, to acquire rules from the users' viewing history. Additionally, we employ a method known as the Repeat Combined Rule Strategy, which draws inspiration from social choice theory strategies, to generate group ratings. The system follows the following sequential steps:

To gather feedback, we collect user input on their viewing history over a period of a few weeks. Utilizing this feedback, We employ the Decision List Rule Learner on the MovieLens dataset to generate a RuleBase specific to each user.

Once the RuleBase is acquired, we classify newly arrived movies accordingly.

Following the classification, we assign weights to each user within the group and apply the Recombined Combined Rule Strategy to generate recommended movies for the entire group of users.

Finally, the individual user or group of users is provided with personalized recommendations, along with an explanation of the underlying reasoning behind the recommendations.

1.6 Organization of Thesis

This thesis is structured as follows: Chapter 2 provides a brief overview of collaborative and content-based approaches. Chapter 3 discusses recent advancements in group recommender systems. Chapter 4 presents the proposed methods for group recommender systems. Chapter 5 presents the experimental results. Finally, the thesis concludes with a summary, bibliography, and screenshots showcasing the implemented program.

Chapter 2

Recommender System

Recommender systems play a vital role in assisting users with decision-making when navigating vast information spaces. These software applications recommend items to users based on their expressed preferences, whether explicitly or implicitly. As the volume and complexity of information on the web continue to grow, recommender systems have become indispensable tools for users engaging in various information-seeking or e-commerce activities. The recommender system serves as a solution to combat information overload by presenting users with the most captivating items while offering novelty, surprise, and relevance. There are two main types of recommender systems:
Collaborative Filtering Recommender Systems
Content-Based Recommender Systems.

2.1 Collaborative Filtering Recommender Systems

Collaborative Filtering Recommender Systems rely on a database that contains users and their respective ratings for a wide range of items they have encountered. The user seeking recommendations, referred to as the active user, can receive item suggestions by searching the database for other users who are similar to them. Recommendations are then made based on the preferences and choices of those similar

users. consider a scenario where there are six users, each with their own profiles containing items labeled as $A, B, C, \ldots Z,$ along with corresponding ratings. When it comes to the active user seeking recommendations, similar users are identified based on correlation matching. In the depicted figure, two users are found to be similar. Among these two users, one individual has given a rating of 8 for item C. As a result, the collaborative method recommends item C to the active user.

2.1.1 Methods Used in Collaborative Recommender Systems

Nearest neighborhood classification: This approach is straightforward and involves a user database that stores ratings given by users for various items. The active user, who seeks recommendations, is the focal point. To provide recommendations to the active user using this method, it identifies similar users through the nearest neighbor or k-similar users using the k-nearest neighborhood algorithm from the database. To determine similarity, the method employs the use of Euclidean distance. This distance metric helps assess the resemblance between users. Consequently, the items recommended by these similar users are then suggested to the active user. For instance, we can illustrate a database comprising three users: U1, U2, and U3, along with four movies: Dasara, Pushpa, Bahubali, and Krishna. The ratings for these movies can be represented as a vector with four entries.
$RatingVector =< RDasara, RPushpa, RBahubali, RKrishna >$. Here, RDasara represents the rating for the movie Dasara, RPushpa represents the rating for the movie Pushpa, RBahubali represents the rating for the movie Bahubali, and RKrishna represents the rating for the movie Krishna. $U1 =< 10, 3, 9, > U2 =<, 9, 6, 2 > U3 =< 1, 7, , 3 >$ Consider an active user whose rating vector is as $U4 =< 9, , , >$, When computing the Euclidean distance between the active user and the other three users, it is evident that user U1 exhibits similarity to the active user. Notably, user U1 has assigned a rating of 9 to the movie Bahubali. As a result, the movie Bahubali will be recommended to the active user. **Pearson correlation**: In statistical analysis, correlation serves as an indicator of the magnitude and direction of a linear association between two random variables. The widely adopted measure of correlation is the Pearson Product Moment Correlation, commonly known as Pearson's

correlation. The formula for calculating Pearson correlation is as follows: Pearson correlation between user a and user

$$u = CoV\,ariance(r_a, r_u)/\sigma r_a, \sigma r_u$$

Here σ is the standard deviation. Covariance and standard deviation are calculated as follows:

$$Covariance(r_a, r_u) = \frac{\sum_{i=1}^{m} (r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{m}$$

Here,

$$\bar{r}_a = \frac{\sum_{i=1}^{m} r_{a,i}}{m}$$

and standard deviation is calculated as

$$\sigma_{rx} = \frac{\sum_{i=1}^{m} (r_{x,i} - \bar{r}_x)^2}{m}$$

The Pearson correlation can be employed to identify similar users to the active user. By utilizing this correlation measure, we can determine the users who exhibit similar preferences and tastes. Consequently, we can recommend items that have been favored by similar users to the active user.

2.1.2 Problems with Collaborative Filtering:

Cold Start: Sufficient user data is necessary for the user database to find suitable matches. Without an adequate number of users, the recommender system may struggle to provide recommendations.

First Rater: The system cannot recommend an item that has not been rated before. This implies that if an item is new or has not received any previous ratings from other users, the recommender system will be unable to suggest that particular item. Popularity Bias: The recommender system may exhibit a bias toward recommending popular items, which can be problematic when catering to users with unique tastes. Since recommendations are often influenced by other user profiles, which tend to favor popular items, individuals with distinct preferences may not receive suitable recommendations.

2.2 Content Based Recommender Systems

In the content-based approach, recommendations are formulated by analyzing the characteristics and features of items, rather than relying on the opinions of other users. This approach employs machine learning algorithms to create a user profile by learning from training examples that provide a detailed description of item features. Using the learned user profile, items that align with the profile are recommended to the active user. Let's take the LIBRA system used by Amazon for book recommendations as an example. The system collects rated examples from Amazon pages, which serve as training data. Through a machine learning algorithm, the system learns the user profile based on these training examples. When it comes to recommending books to an active user, the system utilizes the learned user profile. The books that align with the user profile are then recommended to the active user.

2.2.1 Methods Used in Content-based Recommender Systems

Naive Bayes Classifier: The naive Bayes classifier is a practical Bayesian learning method that is widely used. It is applicable in learning tasks where each instance (denoted as x) is described by a combination of attribute values. The target function (denoted as f(x)) in these tasks can have any value from a finite set V.In the naive Bayes classifier, a set of training examples is given, where each example provides information about the target function. When a new instance is presented, described by a tuple of attribute values $< a_1, a_2, a_3, \ldots, a_n >$, the classifier's task is to predict this instance's target value or classification. The naive Bayes classifier makes the simplifying assumption that the attribute values are conditionally independent given the target value. In simpler terms, the naive Bayes classifier assumes that the probability of observing a particular combination of attribute values ($a1, a2, a3, \ldots, an$) given the target value is equal to the product of the probabilities of each individual attribute occurring.

$$P(a_1, a_2, a_3 \ldots, a_n | v_j) = \Pi_j P(a_i | v_j)$$

$$V_{N,B} = argmax_{vj \epsilon V} P(V_J) \Pi P(a_i | v_j)$$

For example, Let us consider three documents. $D1 = cat, dog, fly, cow \rightarrow yes$ $D2 = crow, strow, fly, zebra \rightarrow no$ $D3 = cat, dog, zoom, flex \rightarrow yes$ Consider an instance$D4 = cat, zoom, fly, dog$. To classify this instance using the naive Bayes classifier, we calculate the probabilities. There are a total of nine words in the instance. The number of unique words in the "yes" class is six, and the number of unique words in the "no" class is four. $P(cat|yes) = 2/6$

$P(cat|no) = \epsilon$

$P(zoom|yes) = 1/6$

$P(zoom|no) = \epsilon$

$P(fly|yes) = 1/6$

$P(fly|no) = 1/4$

$P(dog|yes) = 2/6$

$P(dog|no) = \epsilon$ And,

$P(yes|D4) = P(cat|yes) * P(zoom|yes) * P(fly|yes) * P(dog|yes)$

$= 2/6 * 1/6 * 2/6$

$= 0.003$

$P(no|D4) = P(cat|no) * P(zoom|no) * P(fly|no) * P(dog|no)$

$= \epsilon * \epsilon * 1/4 * \epsilon$

$= 1.6 * 10 - 4$

where $\epsilon = 0.1$

Here $P(yes|D4) > P(no|D4)$,, therefore, classification for the given example is yes.

Decision tree rule learner: The decision tree construction process in Quinlan's C4.5 rule learner is based on an information-theoretic approach using entropy. It follows a top-down, divide-and-conquer strategy. The algorithm starts by selecting an attribute and divides the training set into subsets based on the possible attribute values. This process is recursively applied to each subset until no subset contains objects from more than one class. This approach allows C4.5 to construct a decision tree that effectively classifies instances based on the given attributes.When the training data has a structured format, rule-based recommendation systems tend to perform better. In the case of a movie recommender system, where movie informa-

tion such as director, hero, etc., is related and structured, rule-based approaches can be effective. A commonly used rule learner for structured data is the decision tree rule learner. It learns a decision tree based on the provided training examples, which can then be used to generate a set of rules for classification purposes. By applying these rules, we can classify and recommend items. For instance, let's consider the structured data:

$$Director = Rajamouli \quad and \quad Hero = Chiranjeevi \rightarrow like$$

$$Title = policeacademy \rightarrow notlike$$

By utilizing the decision tree, we can classify movies and determine whether a specific movie is recommendable or not based on the classification.

Chapter 3

Working of Multimodal Recommendation System

3.1 Feature Interaction

The utilization of multimodal data, which encompasses different modalities of descriptive information, is crucial in the recommendation task due to their sparsity and existence in separate semantic spaces. Feature interaction plays a significant role in transforming the feature space nonlinearly into a shared space, thereby improving the performance and generalization of the recommendation model. To facilitate this, we classify feature interactions into three types: Bridge, Fusion, and Filtration. These techniques enable interactions from multiple perspectives and can be simultaneously applied to a single MRS model.

3.2 Bridge

In the context of multimodal recommendation systems (MRS), the concept of Bridge involves establishing a channel for transferring multimodal information. Its focus lies in capturing the interrelationships between users and items by considering the diverse modalities of information. Unlike traditional recommendation systems, where items typically lack multimedia information, multimedia recommendations leverage

the rich multimedia content associated with items. Early approaches often emphasized enhancing item representation with multimodal content, overlooking the interactions between users and items.

To address this limitation, the message-passing mechanism of graph neural networks has been employed to enhance user representations through information exchange between users and items. By leveraging this approach, user preferences for different modalities of information can be effectively captured. For instance, many studies aggregate the interactions between users and items for each modality to determine user preferences. Additionally, the representation of a movie's modality can be derived from the latent item-item graph.

This subsection will introduce various methods for building bridges within MRS, facilitating effective communication and integration of multimodal information.

3.2.1 User-item Graph:

In the context of capturing users' preferences for different modalities, the user-item graph plays a significant role. By facilitating information exchange between users and items, this graph enables the capture of user preferences. In order to leverage this capability, certain approaches utilize a user-item bipartite graph. For each modality, MMGCN establishes a separate graph. By considering the topology of adjacent nodes and the modality information associated with each item, the feature expression of each node can be updated accordingly. This allows for a more comprehensive representation of users' preferences across different modalities. Building upon the MMGCN framework, GRCN introduces an adaptive approach to enhance recommendation performance. It achieves this by dynamically modifying the structure of the graph during model training to eliminate incorrect interaction data, such as cases where users have clicked on uninteresting videos. While these methods have shown significant success in improving performance, they still have limitations in their treatment of user preferences across different modalities. Specifically, they utilize a unified approach to fuse user preferences, disregarding the varying degrees of user preference for different modalities.To address the issue of equal weight assignment to each modality, several methods have been proposed. DualGNN leverages the correlation between users and incorporates user co-occurrence graphs and bi-

partite graphs to learn user preferences. MMGCL introduces a multimodal graph contrastive learning approach, utilizing modal edge loss and modal masking to construct user-item graphs. It also employs a novel negative sampling technique to capture correlations between modalities. MGAT enhances the MMGCN model with an attention mechanism, enabling adaptive capturing of user preferences across different modalities. Furthermore, MGAT utilizes a gated attention mechanism to assess user preferences, enabling the detection of intricate interaction patterns embedded in user behavior. These methods aim to address the challenge of varying user preference levels for different modalities and improve the overall performance of multimodal recommender systems.

3.2.2 Item-item Graph:

The aforementioned studies primarily focus on leveraging multimodal features to capture user-item interactions, overlooking the latent semantic structures among items. Incorporating item-item structures can enhance item representation learning and improve model performance. For example, LATTICE constructs modality-specific item-item graphs based on the user-item bipartite graph, aggregating them to obtain latent item graphs. Similarly, MICRO also constructs item-item graphs for each modality but utilizes a novel comparison method to fuse features after performing graph convolution. However, these approaches fail to consider variations in preferences among specific user groups. In addition to the mentioned approaches, there are several other methods proposed to enhance multimodal recommendation systems. For instance, HCGCN introduces a clustering graph convolutional network that groups item-item and user-item graphs, enabling dynamic graph clustering for learning user preferences. PMGT leverages the pre-training strategy inspired by successful models like BERT, employing a pre-trained graph transformer to incorporate multimodal information and capture project relationships. BGCN, designed for bundle recommendation, integrates user-item interaction, user-bundle interaction, and bundle-item affiliation into a heterogeneous graph, utilizing graph convolution for extracting detailed features. Cross-CBR constructs the user-bundle graph, user-item diagram, and item-bundle graph, employing contrastive learning to align and integrate them from both bundle and item perspectives. These methods aim to improve

the performance and effectiveness of multimodal recommendation systems.

3.2.3 Knowledge Graph:

The integration of knowledge graphs (KG) and MRS has gained attention due to the additional information KGs can provide. Researchers have explored incorporating each modality of items into the KG as individual entities to facilitate this integration. One notable approach is MKGAT, which was the first model to introduce a knowledge graph into a multimodal recommendation. MKGAT introduces a multimodal graph attention technique that considers two key aspects: the aggregation of entity information and the reasoning of entity relationships within the multimodal knowledge graph. This technique enables more comprehensive modeling and reasoning capabilities in the context of KG-enhanced multimodal recommendation. Moreover, to address the challenge of data type diversity in multimodal recommender systems, researchers have proposed innovative approaches. SI-MKR introduces alternate training and leverages the knowledge graph representation based on MKR to enhance multimodal recommendation. This method incorporates user and item attribute information from the knowledge graph to account for data type diversity. Similarly, MMKGV utilizes a graph attention network to disseminate and aggregate information on a knowledge graph, effectively combining multimodal information and leveraging the triplet reasoning relationship of the knowledge graph. Another approach, CMCKG, considers descriptive attributes and structural connections as two modalities and learns node representation by maximizing consistency between these two views. These methods demonstrate the importance of integrating diverse data types and leveraging the knowledge graph in enhancing multimodal recommendation systems.

3.3 Fusion

In the context of multimodal recommendation, integrating diverse multimodal information is crucial, given the large quantity and types of data associated with users and items. To generate effective feature vectors for recommendation tasks, it is necessary to fuse the different multimodal information. Fusion methods fo-

cus on combining preferences from various modalities, with particular emphasis on the intra-relationships among multimodal data. Many multimodal recommender system models incorporate both fusion and bridge techniques, recognizing the significance of both inter- and intra-relationships in learning comprehensive representations. Among the various fusion approaches, the attention mechanism stands out as a widely adopted method. It allows for the flexible fusion of multimodal information with varying weights and attention focus. In this subsection, we classify attention mechanisms based on fusion granularity and also present other fusion approaches employed in multimodal recommender systems.

3.3.1 Coarse-grained Attention:

Certain models utilize attention mechanisms to fuse information from multiple modalities at a higher level of granularity. For instance, UVCAN divides multimodal information into user-side and item-side, including their respective ID information and side information. It leverages multimodal data on the user side to generate fusion weights for the item side through self-attention. Building upon UVCAN, MCPTR introduces parallel merging of the item and user information. In addition to user and item sides, some models merge information from different modal aspects. CMBF introduces the cross-attention mechanism to jointly learn semantic information from image and text modalities, followed by concatenation. Moreover, certain models assign varying proportions to different modalities. MML designs an attention layer based on ID information, complemented by visual and text information. In MCPTR, each modality holds an equal position, and the self-attention mechanism determines the fusion weight. Conversely, HCGCN focuses more on the visual and text information specific to the item itself.

3.3.2 Fine-gained Attention:

Fine-grained fusion is essential in multimodal scenarios where data contains both global and fine-grained features, such as audio tone or clothing patterns. Unlike coarse-grained fusion, which can be invasive and irreversible, fine-grained fusion selectively combines fine-grained feature information from different modalities, preserving the original modality's information and enhancing recommendation perfor-

mance. In the context of fashion recommendation, POG is a notable example of an online clothing recommender system based on a transformer architecture. POG's encoder leverages multi-layer attention to extract deep features related to fashion image collocation, facilitating continuous fine-grained integration. In contrast to POG, NOR utilizes an encoder-decoder transformer architecture with fine-grained self-attention structures to generate scheme descriptions based on collocation information. EFRM, on the other hand, prioritizes interpretability by employing a Semantic Extraction Network (SEN) to extract local features and subsequently fusing them with fine-grained attention preference. VECF adopts image segmentation to integrate image features from each patch with other modalities. Similarly, UVCAN performs image segmentation on video screenshots like VECF and employs the attention mechanism to fuse image patches with id information and text information separately. Lastly, MM-Rec applies the Mask-RCNN target detection algorithm to extract regions of interest from news images and then performs co-attention to fuse Points of Interest (POI) with news content. In order to achieve fine-grained fusion, certain models incorporate unique internal structures. For example, MKGformer facilitates fine-grained fusion through parameter sharing of QKV (Query-Key-Value) and a perceptual fusion module. MGAT employs a gated attention mechanism to prioritize the user's local preferences. MARIO takes into account the individual impact of each modality on each interaction and incorporates a modality-aware attention mechanism to identify the influence of different modalities on each interaction, performing point multiplication for the respective modalities.

3.3.3 Combined Attention:

In order to combine fine-grained fusion with the preservation of global information aggregation, several models have introduced combined fusion structures. NOVA addresses the issue of directly fusing different modal features using vanilla attention, which often leads to minimal impact or performance degradation. It proposes a non-invasive attention mechanism with two branches, separating the id embedding branch to preserve interactive information during the fusion process. NRPA introduces a personalized attention network that takes user preferences expressed in comments into account. It employs personalized word-level attention to select more

significant words in comments for each user/item and sequentially passes the comment information through fine-grained and coarse-grained fusion. VLSNR focuses on news recommendations and captures users' temporary and long-term interests through multi-head attention and GRU network, achieving both fine-grained and coarse-grained fusion. MARank designs a Multi-order Attention layer that combines Attention and ResNet in a unified structure to fuse information effectively.

3.3.4 Other Fusion Methods:

In addition to the attention-based fusion of multimodal information, some models incorporate simple methods such as average pooling, concat operations, and the Gating mechanism. However, these methods are often used in combination with graph and attention mechanisms to achieve better results, as mentioned earlier. It has been observed that these simple interactions when utilized appropriately, do not harm the recommendation effectiveness and can even reduce model complexity. Early models employed RNN and LSTM structures to capture user temporal preferences through multimodal information. However, with the advancements in deep learning techniques like attention mechanisms and CNN, their usage has decreased in recent years. Some models fuse multimodal features using linear and nonlinear layers. For example, Lv et al. employ a linear layer to fuse textual and visual features. In MMT-Net, three context invariants of restaurant data are identified and interaction is performed through a three-layer MLP network.

3.3.5 Filtration:

Multimodal data in recommendation tasks often contain noise that is unrelated to user preferences, requiring filtration to improve recommendation performance. Noise can exist in the interaction graph or multimodal features, and it can be addressed through filtration embedded in the bridge or fusion processes. Certain models utilize image processing techniques for denoising. For instance, VECF and UVCAN employ image segmentation to eliminate noise and capture the user's personalized interests more effectively. MM-Rec utilizes a target detection algorithm to identify significant regions in an image, filtering out irrelevant information. Furthermore, graph neural network structures are widely employed for denoising in multimodal recommenda-

tion tasks. Given the sparsity of user-item interactions and noise in item features, the learned representations of users and items through graph aggregation inherently contain noise. Several models address this issue.

FREEDOM introduces a degree-sensitive edge pruning method to denoise the user-item interaction graph. By considering the degree of nodes, it effectively filters out noisy interactions. GRCN goes beyond traditional graph convolutional network models by adaptively adjusting the interaction graph's structure during training. This allows it to identify and remove erroneous interaction information, preventing noise from affecting the model's performance.

PMGCRN takes into account user interactions with uninteresting items and employs an active attention mechanism to address mismatched interactions. Correcting users' wrong preferences mitigates the impact of noise in the recommendation process. MEGCF focuses on the mismatch problem between multimodal feature extraction and user interest modeling. It constructs a multimodal user-item graph and leverages sentiment information from comment data to finely aggregate neighbors' weights in the graph convolutional network module, effectively filtering information and reducing noise.

3.4 Multimodal Feature Enhancement:

Distinguishing the unique and common semantic information present in different modality representations of the same object can greatly enhance the performance and generalization of MRS. In recent developments, models have incorporated DRL and CL techniques to enhance features through interactions.

By leveraging DRL and CL, these models focus on learning disentangled representations that capture the unique characteristics of each modality while preserving the common semantic information. This approach enables the models to effectively enhance the features and improve the recommendation performance by emphasizing the distinctive aspects of the multimodal data. Through interaction-based methods, these models enhance the representations and enable a more comprehensive understanding of multimodal information, leading to more accurate and effective recommendations.

3.4.1 Disentangled Representation Learning:

In recommender systems, the importance of different modalities in influencing a user's preference for a specific aspect of the target item varies. However, the representations of these different aspects within each modality are often intertwined, making it challenging to extract precise factors that contribute to user preferences. To address this issue, researchers have introduced decomposition learning techniques to uncover intricate factors in user preferences. Prominent examples include DICER, MacridVAE, and CDR.

Furthermore, multimodal recommendation aims to uncover valuable insights hidden within the complex interplay of various factors present in multimodal data. These factors are intricately entangled with each other, posing a significant challenge for analysis and understanding. Researchers in this field strive to disentangle and identify the influential factors to gain a deeper understanding of user preferences and enhance the recommendation process.MDR introduces a novel approach to the multimodal recommendation by leveraging well-disentangled representations that capture both complementary and standard information from different modalities. DMRL takes into account the distinct contributions of various modality features for each disentanglement factor, allowing for more precise capture of user preferences. Additionally, PAMD incorporates a disentangled encoder that extracts modality-common features while preserving modality-specific features automatically.

In the pursuit of disentangled representations, contrastive learning is employed to ensure consistency and differentiation between separated modal representations. This helps maintain a clear distinction between the various factors present in multimodal data. Notably, SEM-MacridVAE extends the idea by incorporating item semantic information into the learning process of disentangled representations derived from user behaviors, thus enhancing the overall recommendation performance.

3.4.2 Contrastive Learning:

In contrast to DRL, contrastive learning approaches focus on improving representations through data augmentation, which is beneficial for addressing the sparsity issue. Several studies in MRS have incorporated contrastive learning loss functions, primarily for modality alignment and enhancing deep feature information be-

tween positive and negative samples. These methods aim to optimize the alignment and discriminative power of multimodal representations through contrastive learning techniques.MCPTR introduces a novel contrastive learning (CL) loss function that promotes semantic similarity between different modal representations of the same item. GHMFC constructs two contrastive learning modules that operate on entity embedding representations derived from a graph neural network. These CL loss functions operate in two directions, namely text-to-image, and image-to-text. Cross-CBR incorporates a contrastive learning loss to align the graph representation between the bundle view and item view. MICRO focuses on capturing shared and specific modal information. In CMCKG, entity embeddings are obtained from both descriptive attributes and structural link information in knowledge graphs, and contrastive loss is applied. HCGCN adopts contrastive learning inspired by CLIP to enforce the alignment of visual and textual item features in the same semantic space. Additionally, weights are assigned to different CL loss functions to control their contributions. Many models in recommendation scenarios employ data augmentation techniques to create positive samples for contrastive learning. MGMC introduces a graph enhancement method and incorporates meta-learning to enhance model generalization. MML, a sequential recommendation model, expands the training data by constructing subsets of users' historical purchase item sequences. LHBPMR selects items with similar preferences from graph convolutions to construct positive samples. MMGCL generates positive samples using modal edge loss and modal masking. Victor constructs samples based on Chinese semantics. Combo-Fashion, a bundle fashion recommendation model, constructs both negative and positive fashion matching schemes. While most models focus on removing irrelevant information from multimodal data, QRec takes an alternative approach by adding uniform noise to multimodal information as positive samples to improve model generalization. Additionally, UMPR, although lacking an explicit CL loss function, constructs a loss function that captures the difference between visual positive and negative samples.

3.5 Model Optimization:

Multimodal recommendation tasks introduce additional computational requirements due to the presence of multimodal information, resulting in increased training complexity when training multimodal encoders and recommendation models together. To address this, multimodal recommendation models can be categorized into two types of training approaches: End-to-end training and Two-step training. In End-to-end training, all layers of the model are updated simultaneously using back-propagation, allowing for the joint optimization of the multimodal encoders and recommendation model parameters. This approach considers the entire model as a single entity during training.

On the other hand, Two-step training involves a two-stage process. In the first stage, the multimodal encoders are trained separately, focusing on learning effective representations from the multimodal data. In the second stage, the pre-trained encoders are fine-tuned in a task-oriented optimization step, specifically tailored for the recommendation task. This two-step approach provides a more structured and controlled training process. Both training methods have their advantages and considerations in terms of computational efficiency and performance optimization for multimodal recommendation models. The choice of training approach depends on the specific requirements and constraints of the recommendation task at hand.

3.5.1 End-to-end Training:

Multimodal recommender systems leverage various types of multimedia information such as images, texts, and audio. To process this data, commonly used encoders from other domains are often employed, including ViT, ResNet, and BERT. These pre-trained models typically have a large number of parameters, such as the 86M parameters in the case of ViT-Base, posing a challenge in terms of computational resources.

To address this issue, most multimodal recommender systems adopt a strategy of using pre-trained encoders directly and focus on training the recommendation model in an end-to-end manner. In this approach, the parameters of the pre-trained encoders remain fixed, and only the recommendation model is trained. For instance,

in the case of NOVA and VLSNR, a pre-trained encoder is used to encode image and text features. The resulting multimodal feature vectors are then embedded into the model for recommendation to users. These models demonstrate that incorporating multimodal data without updating the encoder parameters can still enhance recommendation performance. In contrast, MCPTR takes a different approach by fine-tuning the parameters of the encoder. However, it achieves this with a relatively small number of training epochs (e.g., 100 epochs) using recommendation and contrastive loss functions.

By leveraging pre-trained encoders and adopting an end-to-end training approach, multimodal recommender systems strike a balance between computational efficiency and recommendation performance, allowing for the effective utilization of multimodal data. In addition to improving recommendation performance, certain end-to-end methods also strive to minimize computational requirements. These methods employ strategies to reduce the number of parameters that need to be updated during training, thereby optimizing efficiency. One approach is observed in MKG-former, which utilizes a multi-layer transformer structure. By sharing parameters across multiple attention layers, the computational burden is reduced, resulting in a more efficient training process. Another example is FREEDOM, which focuses on freezing specific parameters related to the graph structure. By doing so, memory costs are significantly decreased, while simultaneously achieving a denoising effect that enhances the overall recommendation performance. These end-to-end methods strike a balance between computational efficiency and recommendation quality, offering practical solutions for reducing computation while still achieving effective recommendations.

3.5.2 Two-step Training:

Two-step training schemes in multimodal recommender systems are less common due to their higher computational requirements compared to end-to-end approaches. However, they offer better targeting of downstream tasks. PMGT introduces a pre-trained graph transformer inspired by Bert's structure. It learns item representations through two objectives: graph structure reconstruction and masked node feature reconstruction. Similarly, in POG, a pre-trained transformer is trained to acquire

fashion-matching knowledge, which is then utilized in a cloth generation model for user recommendations. Two-step training is particularly prevalent in sequential recommendation tasks, where end-to-end training poses challenges. In the pretraining stage, MML adopts meta-learning to train the meta-learner and enhance model generalization. In the second stage, the item embedding generator is trained. TESM and Victor employ pre-training approaches as well, with a well-designed graph neural network and a video transformer, respectively. While two-step training demands more computing resources, it offers targeted training for specific tasks and is often employed in scenarios where end-to-end training is impractical or challenging.

Chapter 4

FUTURE WORK

In future research, we aim to employ transformers for product recommendation by processing multi-modal data, including text and images. The approach involves separate processing of each modality to create embedding's, followed by concatenation and input into the transformer encoder model. The self-attention mechanism is then employed to capture inter-dependencies between modalities, offering a powerful solution for handling complex and diverse data types, particularly beneficial for personalized product recommendations.

Bibliography

[1] Pattie Maes. 1994. Agents that reduce work and information overload. Communications of the ACM, 37(7): 30-40, July 1994.

[2]] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T Riedl. Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems(TOIS), 22(1):5–53,January 2004

[3] Lieberman H. Letizia: An agent that assists web browsing. In Proceedings of the 14th international joint conference on Artificial Intelligence (IJCAI'95), pages 924-929, San Francisco, CA, USA, 1995.

[4] Umair Javed, Kamran Shaukat, Ibrahim A Hameed, Farhat Iqbal, Talha Mahboob Alam, and Suhuai Luo. 2021. A review of content-based and context-based recommendation systems. International Journal of Emerging Technologies in Learning (iJET) (2021).

[5]] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020).

[6] Zhuang Liu, Yunpu Ma, Matthias Schubert, Yuanxin Ouyang, and Zhang Xiong. 2022. Multi-Modal Contrastive Pre-training for Recommendation. In Proceedings of the 2022 International Conference on Multimedia Retrieval.

[7] Xingyu Pan, Yushuo Chen, Changxin Tian, Zihan Lin, Jinpeng Wang, He Hu, and Wayne Xin Zhao. 2022. Multimodal Meta-Learning for Cold-Start Sequential Recommendation. In Proc of CIKM.

[8] Zhulin Tao, Yinwei Wei, Xiang Wang, Xiangnan He, Xianglin Huang, and Tat-Seng Chua. 2020. MGAT: multimodal graph attention network for a recommendation. Information Processing & Management (2020).

[9] Juan Ni, Zhenhua Huang, Yang Hu, and Chen Lin. 2022. A two-stage embedding model for recommendation with multimodal auxiliary information.Information Sciences (2022).

[10] Junliang Yu, Hongzhi Yin, Xin Xia, Tong Chen, Lizhen Cui, and Quoc Viet Hung Nguyen. 2022. Are graph augmentations necessary? simple graph contrastive learning for a recommendation. In Proc. of SIGIR

[11] Jinghao Zhang, Yanqiao Zhu, Qiang Liu, Mengqi Zhang, Shu Wu, and Liang Wang. 2022. Latent Structure Mining with Contrastive Modality Fusion for Multimedia Recommendations. IEEE Transactions on Knowledge and Data Engineering (2022).

[12] Yong Liu, Susen Yang, Chenyi Lei, Guoxin Wang, Haihong Tang, Juyong Zhang, Aixin Sun, and Chunyan Miao. 2021. Pre-training graph transformer with multimodal side information for a recommendation.In Proc. of ACM MM.

[13] Xianshuai Cao, Yuliang Shi, Jihu Wang, Han Yu, Xinjun Wang, and Zhongmin Yan. 2022. Cross-modal Knowledge Graph Contrastive Learning for Machine Learning Method Recommendation. In Proc. of ACMMM

[14] Peng Wang, Jiangheng Wu, and Xiaohang Chen. 2022. Multimodal Entity Linking with Gated Hierarchical Fusion and Contrastive Training. In Proc. of SIGIR

[15] Xiang Chen, Ningyu Zhang, Lei Li, Shumin Deng, Chuanqi Tan, Changliang Xu, Fei Huang, Luo Si, and Huajun Chen. 2022. Hybrid Transformer with Multi-level Fusion for Multimodal Knowledge Graph Completion. arXiv preprint arXiv:2205.02357 (2022)

[16] Xin Wang, Hong Chen, and Wenwu Zhu. 2021. Multimodal disentangled representation for a recommendation. In Proc. of ICME