Market risk estimation using perspective graph representations based on graph retention convolutional neural networks

A thesis submitted in partial fulfillment of the requirements for the award of the degree of

B.Tech

in

Computer Science and Engineering

By

S Rahul Shanker (106120094)

Sarvesh Rajkumar (106120106)

Shyam Vaidyanathan (106120118)

COMPUTER SCIENCE AND ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY
TIRUCHIRAPPALLI – 620015

MAY 2024

BONAFIDE CERTIFICATE

This is to certify that the project titled **Market risk estimation using perspective graph representations based on graph retention convolutional neural networks** is a bonafide record of the work done by

S Rahul Shanker (106120094)

Sarvesh Rajkumar (106120106)

Shyam Vaidyanathan (106120118)

in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering** of the **NATIONAL INSTITUTE OF TECHNOLOGY, TIRUCHIRAPPALLI**, during the year 2023-24.

Dr. Rajeswari Sridhar
Guide

Dr. S. Mary Saira Bhanu
Head of the Department

Project Viva-voce held on ________________________

Internal Examiner

External Examiner

ABSTRACT

Financial risk assessment is critical for investment decisions and risk management. This paper introduces a novel approach to enhance risk classification of companies by integrating financial data and news articles. The dataset for this approach comprises annual, quarterly, and daily financial features of companies acquired through SimFin and two years of news articles of companies acquired through Eodhd API. Each company is depicted as a graph, with nodes representing financial attributes and textual representations derived from news articles, while edges signify connections between them. Utilizing graph neural networks (GNNs), hierarchical representations are acquired to classify risk levels. Our major contributions are the knowledge graph creation and representation of company information and the Graph Attention Network and CNN (GAT-CNN). The results highlight the efficacy of the approach, emphasizing the significance of integrating diverse data sources for comprehensive risk assessment. The advancements achieved in the approach demonstrate promise, with expectations for ongoing evolution and advancement in this research area.

Keywords : Risk classification, Knowledge Graph, Graph attention network, CNN

ACKNOWLEDGEMENTS

We would like to express our deepest gratitude to the following people for guiding us through this course and without whom this project and the results achieved from it would not have reached completion.

We extend our heartfelt gratitude to our guide **Dr. Rajeswari Sridhar**, Professor in the Department of Computer Science and Engineering, for her invaluable guidance and support throughout this project. Her expertise and mentorship have played a pivotal role in our journey, and we are truly grateful for her patience, encouragement, and warmth, which have served as constant sources of inspiration.

We would like to thank **Dr. S. Mary Saira Bhanu**, the Head of the Department, for generously providing us with access to the department's facilities.

We are also thankful to the faculty and staff members of the Department of Computer Science and Engineering, our individual parents and our friends for their constant support and help.

We also express our heartfelt gratitude to **Dr. G. Aghila**, Director, National Institute of Technology, Tiruchirappalli, for cooperating with the students .

S Rahul Shanker (106120094)
Sarvesh Rajkumar (106120106)
Shyam Vaidyanathan (106120118)

TABLE OF CONTENTS

Title **Page No.**

ABSTRACT i

ACKNOWLEDGEMENTS ii

TABLE OF CONTENTS iii

LIST OF FIGURES v

CHAPTER 1 INTRODUCTION 1

1.1 Risk Analysis 1

1.1.1 Current problems in risk analysis 1

1.1.2 Overview of Graph Representation in Risk Assesment 2

1.1.3 Introduction to Graph Neural Networks 2

1.2 Motivation 3

1.3 Problem Statement 3

1.4 Objectives 4

1.5 Organization of the thesis 4

CHAPTER 2 LITERATURE SURVEY 5

2.1 Existing Solutions 5

2.2 Gaps in the existing literature survey 6

CHAPTER 3 PROPOSED SYSTEM 7

3.1 Modules in the Project 7

3.2 Data Acquisition and Cleaning 9

3.3 LDA Analysis 10

3.4 Ground Truth Dataset 12

3.4.1 Data Extraction 13

3.4.2 Data Aggregation 13
3.4.3 Data Normalization 13
3.4.4 Data Merging 14
3.4.5 Feature Calculation 14
3.4.6 Constrained K-means clustering Architecture 14
3.5 Graph generation 15
3.6 HeteroGraph-CNN Architecture Implementation 19
CHAPTER 4 Performance Analysis 22
4.1 Articles Dataset Cleaning 22
4.2 Graph Creation 24
4.3 Hetero CNN and Temporal CNN Performance 25
4.4 Results 25
4.4.1 Model Evaluation 25
4.4.2 Confusion Matrix Analysis 25
4.4.3 Discussion 26
4.5 Results by Sector Group 28
4.5.1 Sector-Specific Performance Analysis 28
4.6 Conclusion 35
CHAPTER 5 CONCLUSION AND FUTURE WORK 37
5.1 Summary 37
5.2 Conclusion 38
5.2.1 Key Findings 38
5.2.2 Methodological Contributions 38
5.2.3 Practical Implications 38
5.2.4 Concluding Remarks 39
5.3 Constraints of the project 39
5.4 Future Scope 39
REFERENCES 41

LIST OF FIGURES

3.1 Overall System Architecture 8

3.2 Project workflow 8

3.3 Workflow of news articles acquisition 10

3.4 Workflow in creation of ground truth 13

3.5 Hierarchial Distribution of Data 17

3.6 Model Architecture 21

4.1 The distribution of data combed 23

4.2 Snippet of chosen keyphrases from LDA 23

4.3 News Articles dataset 23

4.4 Preliminary hierarchical representation of data 24

4.5 Knowledge graph of Company 'MSI' 25

4.6 Overall Confusion Matrix 26

4.7 ROC-AUC curve for each separate class 28

4.8 Health Care Confusion Matrix 29

4.9 Consumer Discretionary and Consumer Staples Confusion Matrix . . 30

4.10 IT and Communications Confusion Matrix 31

4.11 Utilities Confusion Matrix 32

4.12 Financials and Real Estate Confusion Matrix 33

4.13 Industrials and Materials 34

4.14 Energy Confusion Matrix 35

4.15 Train Loss vs Epochs 36

CHAPTER 1
INTRODUCTION

1.1 Risk Analysis

Accurate risk assessment is essential for making wise investment decisions and successful risk management plans in the fast-paced financial environment of today. It offers important insights into the possible risks and uncertainties the company may face to stakeholders including creditors investors and management. Stakeholders can make more informed decisions about their involvement with the company by identifying and evaluating a variety of risk factors such as financial operational and market risks. This include choosing investment strategies evaluating a borrowers creditworthiness and creating risk-reduction strategies to protect against future losses.

Companies can enhance their resilience and sustainability by employing effective risk analysis techniques to proactively manage and mitigate risks. By gaining a thorough understanding of potential threats and vulnerabilities companies can implement risk management strategies that are designed to lessen the negative consequences of unanticipated events. This could include tightening internal controls diversifying revenue streams and developing emergency plans. Ultimately thorough risk analyses help firms reduce risks to their operations and financial performance better manage the ever-changing business environment allocate resources efficiently and add value for stakeholders.

1.1.1 Current problems in risk analysis

In the form of records and reports financial documents provide more in-depth details about the financial status and prospects of a business. However traditional methodologies often overlook significant contextual information found in real-time news stories leading to insufficient risk assessments. These articles offer up-to-date information on market trends unexpected events and regulatory changes that may significantly affect a companys risk profile. If this crucial source of information is ignored chances to lower risk may be lost and risk assessments may turn out to be biased.

Our project fills this gap by providing a comprehensive approach to risk classification using advanced techniques such as graph neural networks to fuse financial data with textual insights. It does this by utilizing a variety of data sources to produce actionable insights. This integration which enables the assessment of both qualitative contextual data and quantitative financial metrics makes it feasible to better understand a firms risk exposure.

1.1.2 Overview of Graph Representation in Risk Assesment

Graph representations provide a strong foundation for simulating intricate dependencies and relationships found in news articles and financial data. Graphs are an ideal abstraction to depict entities including businesses financial characteristics and textual embeddings along with the relationships among them. The graphs edges depict the connections or interactions between the various entities that each node in the graph represents.

By combining disparate data sources into a single framework graph representations allow us to approach risk assessment from a comprehensive perspective. Innovative methods like graph neural networks (GNNs) can be used to learn hierarchical representations that capture the underlying patterns and dynamics of risk in the financial domain by encoding financial metrics textual sentiments and their relationships within a graph structure.

1.1.3 Introduction to Graph Neural Networks

Graph Neural Networks (GNNs) are a potent tool that can be used to analyze and process graph-structured data. They provide a flexible framework that can be used for a variety of tasks including graph classification link prediction and node classification. The graph structures that are frequently encountered in real-world scenarios are handled by GNNs differently to traditional neural networks which are built for grid-structured data such as images or sequences.

Message passing in which each node aggregates and passes data from its neighbors across several layers is the fundamental idea behind GNNs. Due to their ability to capture both local and global dependencies within the graph GNNs are able to learn rich representations that contain intricate patterns and relationships. Structured neural networks (GNNs) are able to capture hierarchical dependencies and structural information in a graph by iteratively updating node features based on neighboring nodes information.

1.2 Motivation

There are several reasons to build a risk classifying model for companies:

- **Incomplete Risk Assessment**: Traditional methods of risk assessment often rely solely on quantitative financial metrics, overlooking crucial contextual information embedded in real-time news articles. This incomplete evaluation may lead to inaccurate risk assessments and missed opportunities for risk mitigation.

- **Need for Timely Risk Identification**: In today's rapidly changing business environment, timely risk identification is essential for effective risk management. Real-time news articles provide valuable insights into market trends, regulatory changes, and geopolitical events that can impact a company's risk exposure. Integrating this timely information into risk assessment models enables stakeholders to identify emerging risks and opportunities more effectively.

- **Practical Implications for Stakeholders**: Effective risk assessment is vital for stakeholders, including investors, financial institutions, and risk managers, to make informed decisions and mitigate potential losses. By developing more accurate and comprehensive risk assessment models, we can empower stakeholders with actionable insights for navigating the complexities of the financial landscape and maximizing opportunities while minimizing risks.

- **Emergence of Alternative Data**: With the proliferation of digital media and the advent of big data analytics, there is a wealth of alternative data sources available for analysis. Integrating diverse data sources, such as financial data and news articles, offers the potential to enhance risk assessment models and provide more comprehensive insights into a company's risk profile.

1.3 Problem Statement

To develop a methodology for comprehensive risk assessment of companies by integrating financial data and news articles into a unified graph representation. The proposed methodology utilizes advanced machine learning techniques, including graph neural networks (GNNs), to classify risk categories based on the constructed graph representation. By leveraging heterogeneous data sources and modeling complex relationships between financial metrics and textual insights, the aim is to provide stakeholders with a holistic perspective on company risk profiles. The focus of this project is to enhance the accuracy and timeliness of risk assessment, thereby empowering stakeholders with actionable insights for informed decision-making in the dynamic financial landscape.

1.4 Objectives

- Performing data acquisition using web crawlers.
- Cleaning and appending relevant context-driven data to the finalized dataset.
- Obtain a temporal financial graph representation of a company.
- Utilize a GAN to input and process the represented graph to make relevant classification estimates.
- Utilize ML to estimate risks from the objective and subjective data acquired via SimFin and News articles respectively.

1.5 Organization of the thesis

Chapter 2 will cover the existing methods of risk analysis and other relevant methods which are related to our task. It will first go through working, advantages and disadvantages of the existing methods. Then, we identify gaps in the literature survey conducted and attempt to fix them in our model.

Chapter 3 discusses about the methodology used in our project in detail. It goes over the general architecture, algorithms, implementation and design details of various modules in the project.

Chapter 4 goes over the performance analysis of the project.

Chapter 5 discusses about the summary, conclusions and future works about the project.

CHAPTER 2
LITERATURE SURVEY

2.1 Existing Solutions

”Multi-granularity heterogeneous graph attention networks for extractive document summarization” by Yu et al. presents MHgatSum, integrating semantic nodes like keyphrases and topics with sentences. Employing a heterogeneous graph attention network (GAT), it discerns element significance for summarization, promising improved accuracy but necessitating further research for generalizability validation.

”Forecasting credit default risk with graph attention networks” by Zhou et al. introduces a novel method for predicting credit default risk utilizing graph attention networks (GATs). By leveraging network data about borrowers and their connections, GATs assess creditworthiness, potentially enhancing accuracy over traditional methods. However, careful consideration of potential biases in the network data is crucial to ensure fairness and reliability in the model’s predictions.

In ”Financial fraud detection using graph neural networks: A systematic review” by Luo et al., GNNs are explored for identifying fraudulent activity in financial data. While GNNs offer potential in capturing complex relationships for enhanced fraud detection accuracy, further research is warranted to address challenges like data availability, model interpretability, and potential biases.

Abraham et al. propose a hybrid system combining a neural network and a neuro-fuzzy system for stock forecasting. The system utilizes principal component analysis for data preprocessing but lacks evaluation of factors affecting market performance.

Tsang et al. also present a hybrid intelligent system for stock trading that incorporates case-based reasoning, rule-based reasoning, and a neural network. The paper does not assess the system’s profitability or address real-world market complexities.

The paper by Huang, Pasquier, and Quek introduces a hybrid genetic-neural architecture for stock market forecasting. It combines genetic algorithms with a feedforward neural network, partitioning input space using technical analysis indicators. While claiming superiority over buy and hold strategy, its high computational complexity limits its applicability to more amorphous data.

2.2 Gaps in the existing literature survey

In reviewing the existing literature, we've identified several gaps that our project aims to address. Firstly, while previous studies have focused on utilizing either financial data or textual information for specific tasks like fraud detection or credit risk assessment, there's a notable absence of approaches that integrate heterogeneous data sources comprehensively. Our project seeks to bridge this gap by developing a methodology that integrates both financial data and news articles into a unified graph representation, offering a more holistic approach to financial risk assessment.

Furthermore, while some studies have explored the use of graph neural networks (GNNs) for tasks like fraud detection and credit risk assessment, there remains a gap in the literature regarding their application to financial risk assessment in the context of heterogeneous data sources. As such, our project aims to contribute to filling this gap by leveraging GNNs to analyze graph representations constructed from financial data and news articles. This approach has the potential to capture complex relationships between different elements and improve the accuracy of risk assessment models.

Moreover, many of the reviewed studies acknowledge the need for further research to validate the generalizability of their proposed approaches. Our project seeks to address this gap by conducting experiments to evaluate the generalizability of the developed methodology across different datasets and financial contexts. By doing so, we aim to provide insights into the robustness and applicability of the proposed approach in various real-world scenarios.

Finally, some studies highlight the computational complexity associated with their proposed approaches, which could limit their practical applicability. In contrast, our project aims to develop methodologies that are computationally efficient while maintaining high accuracy in financial risk assessment tasks. By addressing these computational challenges, we aim to ensure that the developed approach is both practical and scalable for real-world applications. Overall, our project seeks to make significant contributions to the field of financial risk assessment by addressing these gaps and advancing the state-of-the-art in leveraging heterogeneous data sources and graph neural networks for improved risk evaluation.

CHAPTER 3
PROPOSED SYSTEM

3.1 Modules in the Project

The project consists of the following modules:

- Data Acquisition and cleaning
- Latent Dirichlect Allocation analysis on news articles
- Ground Truth Dataset creation
- Graph generation
- HeteroGraph-CNN Architecture implementation

The Data Acquisition and Cleaning module involves collecting financial data and news articles from various sources, followed by preprocessing to ensure data consistency and cleanliness, laying the foundation for subsequent analysis.

Latent Dirichlet Allocation analysis on news articles module applies topic modeling techniques to extract the topics from news articles, providing insights into the underlying themes and trends present in the textual data, thereby enriching the overall dataset for risk assessment purposes.

The Ground Truth Dataset creation module involves annotating the dataset with risk labels based on expert knowledge or historical data, serving as a reference for training and evaluating the risk classification model.

Graph generation module constructs a graph representation of the financial data and textual embeddings from news articles, capturing the relationships between different entities and features, facilitating the integration of heterogeneous data sources for risk assessment.

The HeteroGraph-CNN Architecture implementation module implements a novel architecture that combines graph attention networks (GATs) that can take HeteroGraphs with convolutional neural networks (CNNs) to analyze the graph representation and classify risk levels, leveraging both local and global information for accurate risk assessment.

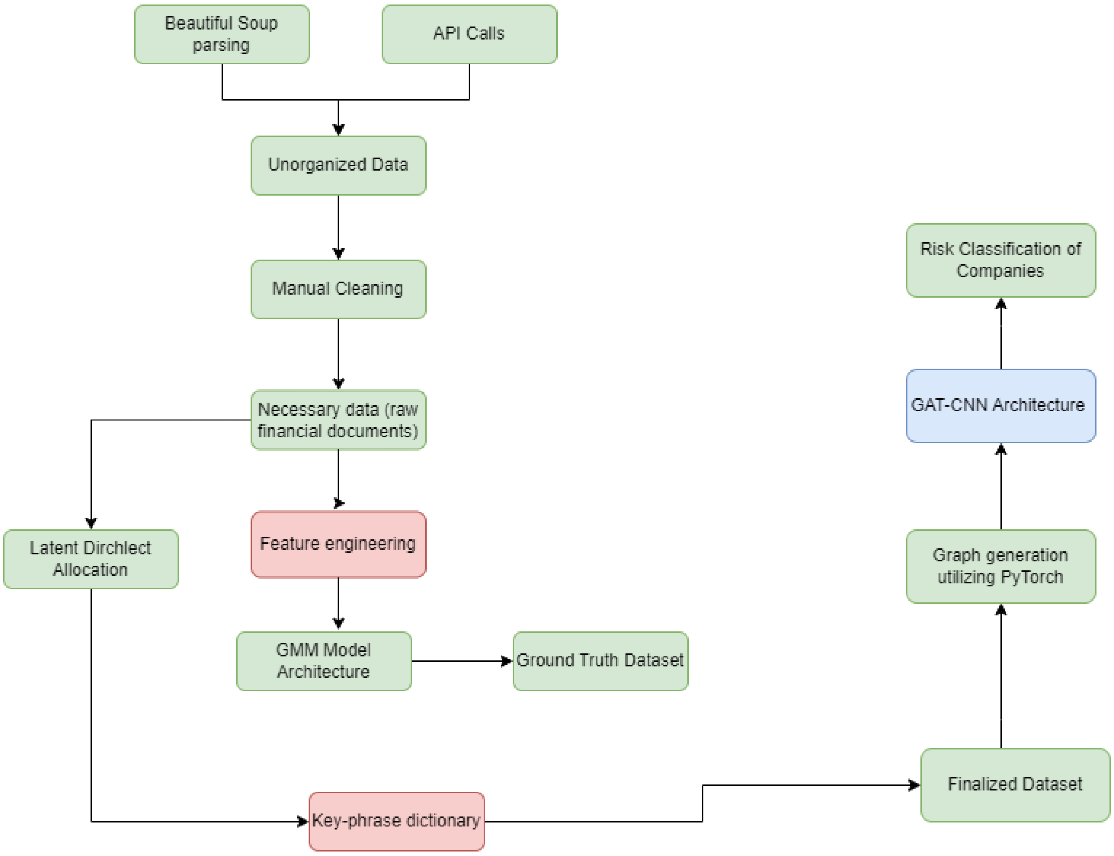


Figure 3.1: Overall System Architecture

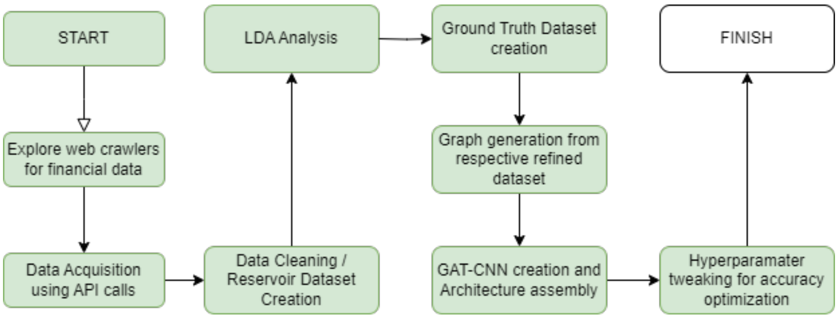

START
Explore web crawlers for financial data
Data Acquisition using API calls
Data Cleaning / Reservoir Dataset Creation
LDA Analysis
Ground Truth Dataset creation
Graph generation from respective refined dataset
GAT-CNN creation and Architecture assembly
Hyperparamater tweaking for accuracy optimization
FINISH

Figure 3.2: Project workflow

We will go through each module in detail in the upcoming sections. Fig 3.1 shows the overall architecture of our model. The overall work flow is shown in Fig 3.2.

3.2 Data Acquisition and Cleaning

Data Acquisition was an important step in the process. The emphasis was on creating a diverse dataset that could improve the system's robustness and performance. Financial data was acquired through SimFin (https://www.simfin.com/en/) while news articles was acquired through the Eodhd API (https://eodhd.com/). We could directly download financial data from the SimFin website. We had to come up with an algorithm to effectively utilize the Eodhd API. We focussed on the S&P 500 companies for this project. The algorithm for collecting news articles is given below:

Input: List of S&P 500 company tickers, List of date ranges for data retrieval

Output: Excel files containing news data for each company within specified date ranges

Procedure:

```
1. Define a function get_tickers() to extract S&P 500 company tickers from Wikipedia.
2. Retrieve tickers using the get_tickers() function.
3. Define a list of date ranges for data retrieval.
4. Iterate over each ticker:
    a. Iterate over each date range:
        i. Construct the URL for API request using ticker and date range.
        ii. Send GET request to the API URL.
        iii. Convert JSON response to pandas DataFrame.
        iv. Select relevant columnss.
        v. Save DataFrame to an Excel file including the ticker and date range.
5. End of iteration.
```

Fig 3.3 shows the workflow.

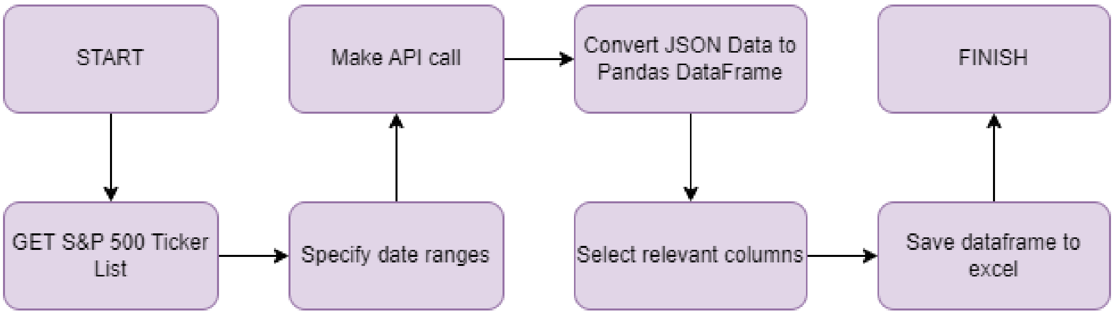


Figure 3.3: Workflow of news articles acquisition

Data cleaning was the next step in the project. We had to ensure the data was in a usable format for the modules ahead. This included cleaning both the financial ratios dataset and the news articles dataset.

For the Simfin Dataset, we performed the following steps for cleaning:

Input: Raw SimFin Financial Data of Companies

Output: Cleaned Financial Data of Companies

Procedure:

1. Retain only data of companies for which we have news articles.
2. Filter Data and store only those between the years 2022 and 2024.
3. Extrapolate missing values for companies.
4. Normalize and organize data accordingly for the time nodes.

3.3 LDA Analysis

Latent Dirichlet Allocation (LDA) is a generative probabilistic model used for topic modeling, a technique to identify topics present in a bunch of documents.The key idea behind LDA is that words in a document are generated from a mixture of topics, and the topic mixture proportions for each document are drawn from a

Dirichlet distribution. Similarly, the distribution of words within each topic is drawn from another Dirichlet distribution. By inferring these latent topic distributions from the observed data, LDA can uncover the underlying themes or topics present in the document collection. We ran an LDA analysis over the news articles we had collected to extract the keywords. Algorithm for LDA is given below:

Input: Corpus of text documents,Number of topics to extract (k)

Output: Set of k topics, each represented by a distribution over words, Topic proportions for each document in the corpus

Procedure: Initialization:

1. Initialize random topic assignments for each word in each document.
2. Initialize count matrices for topics-word assignments and document-topic assignments.
3. Iteration : For each document d in the corpus:
   (a) For each word w in document d:
   (b) Sample a new topic assignment for word w based on the current topic-word distribution and document-topic distribution.
   (c) Update the count matrices accordingly.
4. For each topic t: Update the topic-word distribution based on the count of words assigned to topic t.
5. For each document d: Update the document-topic distribution based on the count of words assigned to each topic in document d.
6. Repeat steps 3-5 until convergence or a predefined number of iterations.
7. Extract the top words for each topic based on their probabilities in the topic-word distribution.

After running LDA, we aggregated the most common words into categories prevelant in the industry. For example one category we found was: ”Fraud”: [”embezzlement”, ”insider”, ”insider trading”, ”Ponzi scheme”, ”counterfeit”, ”corruption”, ”money laundering”, ”laundering”, ”falsification”, ”misrepresentation”, ”whistleblower”,”SEC investigation”, ”audit failure”, ”financial irregularities”, ”kickbacks”, ”unethical practices”, ”unethical”, ”compliance breach”, ”breach”, ”scandal”, ”forgery”, ”deceptive”, ”swindle”, ”defraud”] Then for each of these categories we go through the news articles and add columns to indicate the presence of any of the keywords. The algorithm we used for that was:

Input: Excel file containing news articles with multiple sheets

Output: Processed Excel file with additional columns indicating the presence of keywords for each category

Procedure:

1. Define a dictionary keywords_dict containing keywords for various categories.
2. Create a pattern dictionary pattern_dict by compiling regular expression patterns for each category using the keywords.
3. Read the Excel file using pd.read_excel() method, which returns a dictionary of DataFrames, with each sheet represented as a DataFrame.
4. Initialize an ExcelWriter to write back results to the output Excel file using pd.ExcelWriter().
5. For each sheet:
   (a) Iterate over each category and apply the check_keywords() function to the 'content' column using .apply() method, which checks for the presence of keywords using regex pattern matching.
   (b) Create new columns in the DataFrame for each category, indicating whether the corresponding keywords are present.
   (c) Filter the DataFrame to keep only rows where at least one category matches by summing the values across columns and using boolean indexing.
   (d) Write the processed DataFrame to a new sheet in the output Excel file using .to_excel() method of the ExcelWriter
6. End of iteration.

3.4 Ground Truth Dataset

The creation of a ground truth dataset consisted of the following steps:

1. Data extraction
2. Data aggregation

3. Data normalization

4. Data merging

5. Feature calculation

6. Constrained K-means clustering architecture

The workflow is given in Fig 3.4.

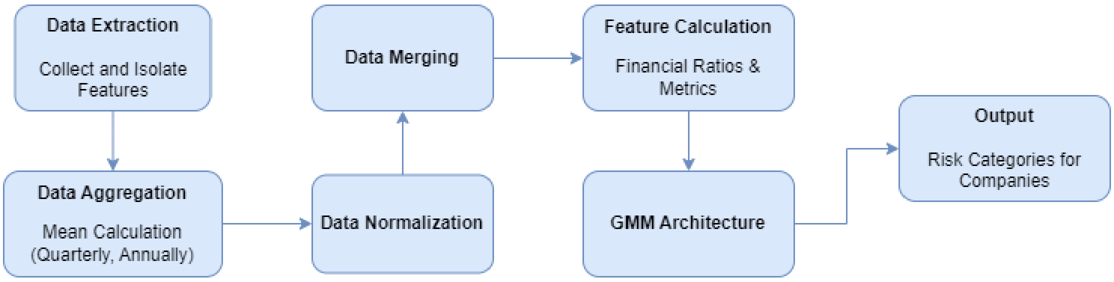

Data Extraction
Collect and Isolate Features
Data Aggregation
Mean Calculation (Quarterly, Annually)
Data Normalization
Data Merging
Feature Calculation
Financial Ratios & Metrics
GMM Architecture
Output
Risk Categories for Companies

Figure 3.4: Workflow in creation of ground truth

3.4.1 Data Extraction

Initially, features had to be isolated from a bigger dataset. This step involved retrieving raw data points such as financial ratios, stock prices, and economic indicators, ensuring a comprehensive representation of the company's financial health.

3.4.2 Data Aggregation

The data aggregation method employed mean calculation to aggregate daily and quarterly features into yearly summaries. By averaging values over these intervals, it provided a consolidated representation of the company's performance on an annual basis, accounting for fluctuations and helping in trend analysis for effective risk assessment.

3.4.3 Data Normalization

Normalization of the dataset was essential to standardize the scale and distribution of the data across different features. This step involved applying min-max normalization to rescale numerical values, ensuring consistency and comparability across diverse variables.

3.4.4 Data Merging

After normalization, the dataset underwent merging processes to integrate the isolated features in the previous steps. By merging relevant datasets, a single unified and informative dataset was created.

3.4.5 Feature Calculation

Feature calculation involved deriving meaningful metrics and indicators from the merged dataset to capture key aspects of the company's financial performance and risk exposure. This step included computing financial ratios, liquidity measures, profitability indicators, and other quantitative metrics essential for risk analysis and decision-making. The following features were calculated for the creation of ground truth:

- Current Ratio
- Debt-to-Equity Ratio
- Interest Coverage Ratio
- Net Profit Margin
- ROE
- Price to Earnings Ratio
- Price to Book Value

3.4.6 Constrained K-means clustering Architecture

Since we did not have a labeled dataset we had to use an unsupervised model. Constrained K-means clustering gave the most uniformly distributed dataset among other models that we tested. The algorithm used is described below:

Input: CSV file containing financial data with ratios, List of financial ratios for clustering

Output: Excel file with clustered data and risk labels assigned

Procedure:

1. Read the CSV file containing financial data into a pandas DataFrame.

2. Extract the specified financial ratios from the DataFrame.
3. Perform standard scaling on the extracted data to normalize it.
4. Define the number of clusters and constraints for cluster sizes.
5. Initialize the KMeansConstrained model with the specified parameters.
6. Fit the model to the scaled data and assign cluster labels to each data point.
7. Map cluster labels to risk levels based on predefined mappings.
8. Create one-hot encoded columns for the cluster labels.
9. Save the clustered data with risk labels to an Excel file.

At this stage the ground truth had categorized companies into 5 possible risk levels.

3.5 Graph generation

Using the finalized dataset we have created, which includes clean financial data as well as news articles enriched with LDA assisted incidents information, we construct a heirarchical graph for each company. The graph contains company ticker, year, quarter, month and day as nodes with each node having its own set of attributes. A detailed description of the graph is given below:

1. **Company Node**: Represents the company itself. It has a dummy feature indicating its presence.
2. **Year Node**: Represents each year within the specified date range. Contains features derived from annual financial data, such as total assets, total liabilities, total equity, and long-term debt.
3. **Quarter Node**: Represents each quarter within each year. Contains features extracted from quarterly financial data, including operating income, net income, total interest expense, net cash from operating activities, net cash from financing activities, change in working capital, EBITDA, net profit margin, return on equity, debt ratio, and net debt to EBITDA ratio.
4. **Month Node**: Represents each month within each quarter. Contains a single feature indicating the month value.

5. **Day Node**: Represents each day within each month. Contains features extracted from daily financial data and news articles. These features typically include metrics like closing price, trading volume, price-to-earnings ratio (P/E ratio), and enterprise value-to-EBITDA (EV/EBITDA) ratio, along with incident information from news articles.

The code to generate the graphs for different companies is described below:

Input: List of S&P 500 company tickers, Paths to financial datasets and news articles

Output: Saved graphs for each company within specified date ranges

Procedure:

1. Import necessary libraries including pandas, torch, and os.
2. Define data paths using a dictionary data_paths.
3. Define helper functions for data loading, feature extraction, and graph creation.
4. Iterate over each ticker symbol:
   (a) Load financial and news data using the load_data() function.
   (b) Create a heterogeneous graph representation for the ticker using create_heterogeneous_graph() function.
   (c) Save the generated graph as a PyTorch tensor.
5. End of iteration.

The create_heterogenous_graph() function is described below:

Algorithm - create_heterogeneous_graph() Function:

Input:

financial_data: Dictionary containing financial datasets for a specific ticker.

ticker: Ticker symbol for the company.

start_date: Start date of the date range for which the graph is to be created.

end_date: End date of the date range for which the graph is to be created.

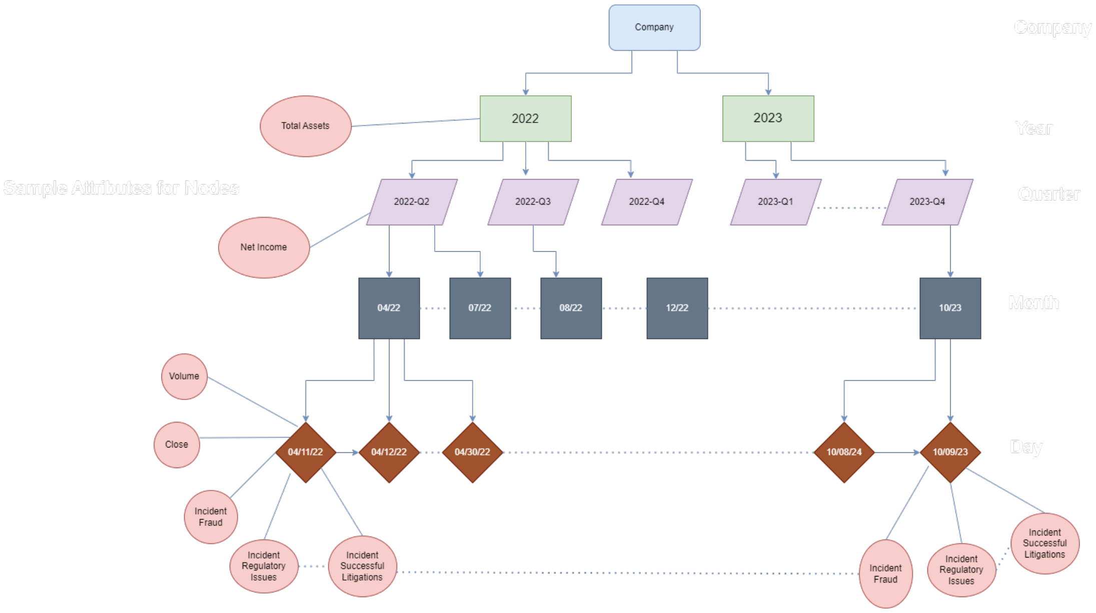


Figure 3.5: Hierarchial Distribution of Data

Output: Heterogeneous graph representation of financial data for the specified ticker symbol and date range.

Procedure:

1. Initialize a HeteroData object to store the heterogeneous graph.
2. Initialize edge_index tensors for all node types in the graph.
3. Define date range based on the provided start and end dates.
4. Initialize storage for node features for each node type in the graph.
5. Initialize indices dictionaries to track node indices for quarter, month, and year nodes.
6. Iterate over each date in the date range:
   (a) Extract daily data from the financial dataset for the current date.
   (b) Extract derived features and news features for the current date.
   (c) Create day node with daily features and connect it to the month node.
   (d) Update month node features with month label and connect it to the quarter node.
   (e) Update quarter node features with quarterly features and connect it to the year node.
   (f) Update year node features with yearly features.

7. End of iteration.
8. Connect company node to each year node in the graph.
9. Return the constructed heterogeneous graph.

A rough representation of the graph is shown in Fig. 3.5.

Initially setting edge weights to random numbers in order to work a skeleton of the graph dataframe, The transformations undergone in order to use edge weights that were not skewed, include aggregation and normalization of the included data. For daily nodes we chose to use the deltaClose and deltaAltman Z-Score between consecutive days. Similarly, consecutive months are linked with 6 attributes, and quarters with 2. This way inconsequential variances temporally can be removed, and volitaile shifts of the company are captured. The code to form edges is described below:

Input:

graph_directory: Path to the directory containing the input

graph files, csv_file: Path to the CSV file containing relevant data

output_directory: Path to save the modified graph files.

Output: Modified graph files saved in the specified output_directory.

Procedure:

1. Initialization: If output_directory does not exist, create it.
2. Iterate Over Graph Files: For each file in graph_directory:
   (a) Check if the file name ends with '_modified.pt'. If yes, skip to the next file.
   (b) Construct the full path of the current graph file.
   (c) Construct the output path for the modified graph file by replacing '_modified.pt' in the file name with '_modified.pt' and appending it to output_directory.
   (d) Call the modify_graph_attributes function with the current graph file, the CSV file, and the output path.

3. Modify Graph Attributes Function (modify_graph_attributes):

    (a) Input: graph_file: Path to the input graph file, csv_file: Path to the CSV file containing relevant data, output_path: Path to save the modified graph.

    (b) Load the graph from graph_file using torch.

    (c) Read the CSV data from csv_file using pandas.

    (d) Extract the ticker name from the filename of the graph file by splitting it and selecting the first part.

    (e) Filter the CSV data for the current ticker by selecting rows where the 'Ticker' column matches the extracted ticker name.

    (f) Sort the filtered data based on the 'prevQuarter' column.

    (g) Define the edge type to modify, e.g., ('quarter', 'to', 'nextquarter').

    (h) Check if the specified edge type exists in the graph. If not, print a warning message and exit.

    (i) If the edge type exists in the graph, check if there are enough rows in the CSV data to match the number of edges in the graph. If not, print a warning message.

    (j) Select the required columns from the relevant CSV data, such as 'deltaGross_Profit' and 'deltaNet_cash_from_operating_activities', and convert them to a tensor of type torch.float32.

    (k) Update the edge_attr tensor of the specified edge type in the graph with the new edge attribute data.

    (l) Save the modified graph to the specified output_path.

3.6 HeteroGraph-CNN Architecture Implementation

Discussion is held regarding the risk prediction model for classifying financial data. The model is meant to forecast the risk levels connected to financial entities. It is written in Python and is implemented with PyTorch and DGL. A TemporalEdgeConv layer and a RiskPredictionModel make up the model architecture. Within the heterogeneous graph the TemporalEdgeConv layer handles the temporal edges between nodes. In order to capture temporal dependencies in the data it combines temporal information from various time intervals such as day month quarter year.

Furthermore it uses an analysis of graphs using a heterogeneous graph attention network (GAT) to determine the importance of different elements for risk prediction. By projecting node features into a lower-dimensional space the RiskPredictionModel uses neural networks to further process node features. To capture intricate connections between nodes and temporal edges it makes use of multiple TemporalEdgeConv layers.

After employing HeteroGraphConv layers to aggregate node representations the model predicts risk levels by applying linear transformations and activation functions. Financial indicators that are displayed as heterogeneous graphs such as quarterly and annual financial indicators are loaded as part of the training process. The model is used to process these graphs and forecast the risk levels related to financial organizations.

With a learning rate of 0. 005,and an Adam optimizer is used to train the model. Furthermore a scheduler that utilizes exponential learning rate decay is utilized to modify the learning rate gradually. Throughout training the models performance is monitored and every epochs training loss is noted. Once the model has been trained it is saved along with its training metrics for later assessment and implementation. With possible uses in risk management and financial fraud detection the models performance is evaluated based on how well it can predict risk levels for financial entities.

Fig 3.6 shows the model architecture. In the model Architecture, the edge attribute values for edges between consecutive similar node types are passed through a TemporalEdgeConv that uses GATConv (Graph Attention Network Convolutional). The output from this model is appended to the features vectors of the node types that are first processed by a fully connected layer. The graph and modified feature vectors are then passed into a GAT CNN layer, where we input the feature dictionary dimensions for the node types. The output is passed into another projection layer and one more GAT CNN layer. The output from the second convolutional layer is aggregated with respect to the company node. The result is then categorized into 5 risk levels.

Final step was tofine-tune the parameters of the HeteroGraph-CNN architecture, optimizing model performance and ensuring robustness against overfitting or underfitting.

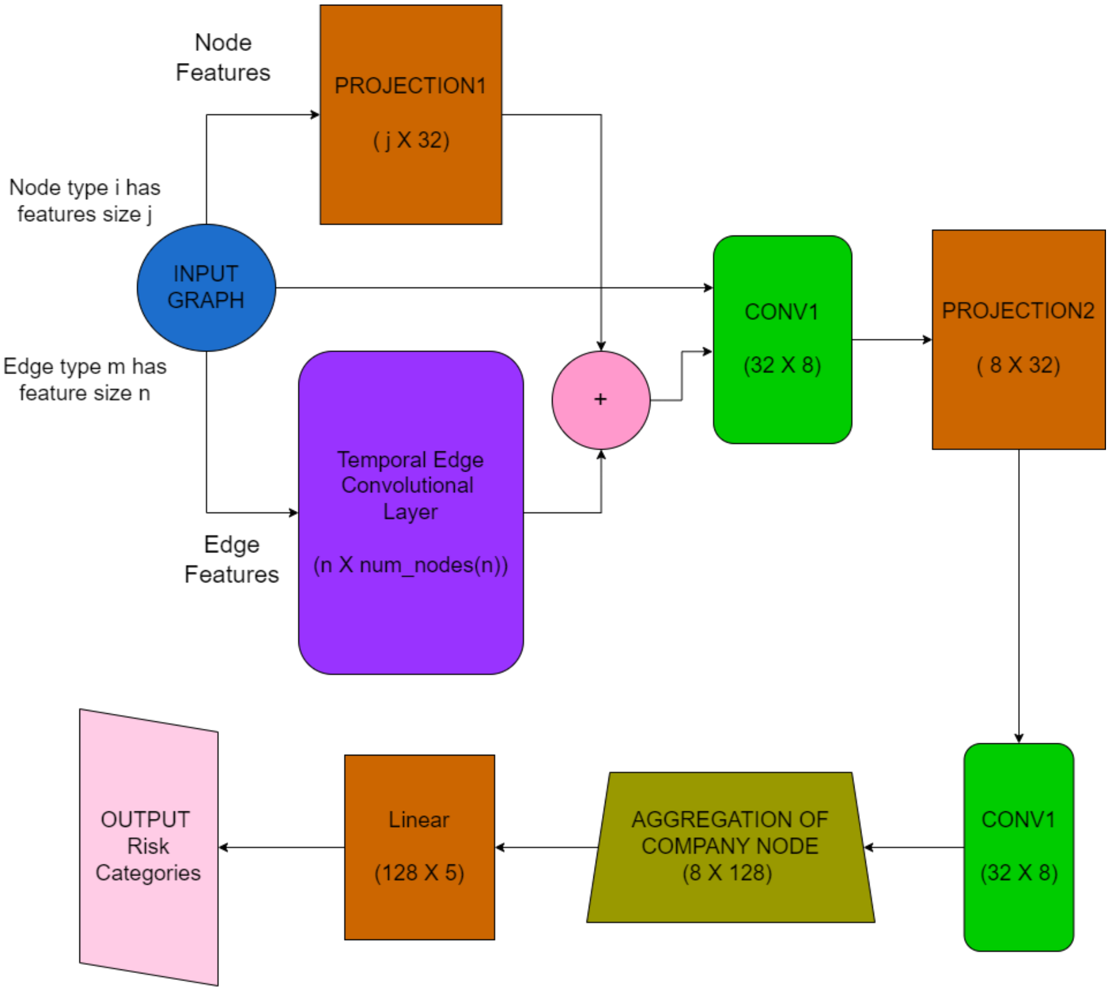


Figure 3.6: Model Architecture

CHAPTER 4

Performance Analysis

4.1 Articles Dataset Cleaning

The dataset cleaning process is essential to ensure the accuracy and effectiveness of our model. We conducted several steps to clean and preprocess our data:

1. **Removal of Outliers:** Data points that were significantly different from others were removed to prevent them from skewing the results. This was further emphasized with Polarity Analysis.

2. **Handling Missing Data:** Missing values were pruned, to prevent pollution of dataset.

3. **Normalization:** Data normalization involved scaling numerical inputs with zero mean and unit variance, facilitating faster and more stable training. This was done on a daily, monthly, and quarterly basis across companies.

4. **Dataset split:** After the acquisition of the Dataset using the cumulative articles produced by the FinHub API and the eodhd API to acquire a mass of 3.5 million articles. These articles were cleaned and filtered as represented in Figure 4.1

5. **LDA Analysis:** Utilising LDA as one of the components to determine the Overarching themes present, and splitting them broadly into 10 types: Regulatory issues, Product failure/Recalls, Data Breaches/Cyberattacks, Fraud, Management scandals, Mergers and Acquisitions, Product launches, New market expansions, Successful litigations, and Strategic partnerships. Given in Figure 4.2 the outputs of common-words LDA arrived at.

6. **Polarity Analysis:** After using FinBERT called in the eodhd function to store the pos, neg, polarity, and neu factors, indicating the positive, negative, extent of, and neutrality to determine the files to be chosen. Some values for said article can be seen in Figure 4.3. Use of FinBERT for accurate sentiment analysis and score 86

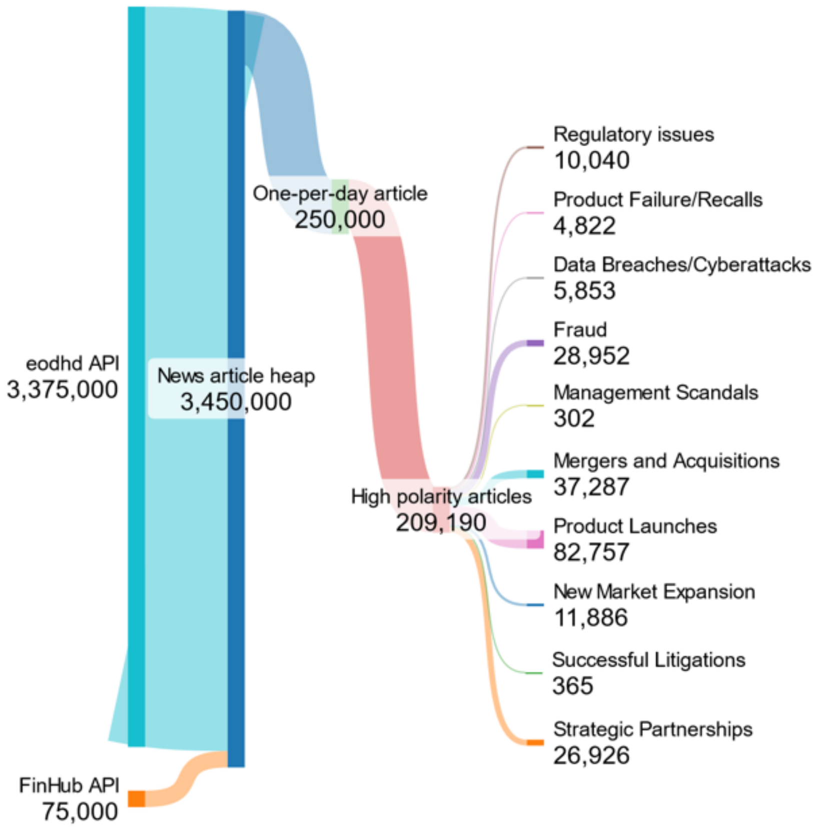
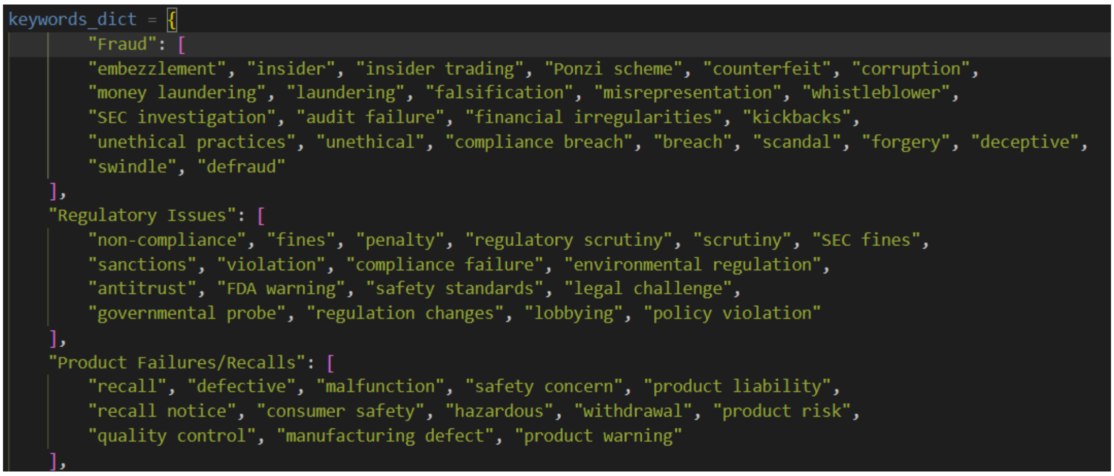
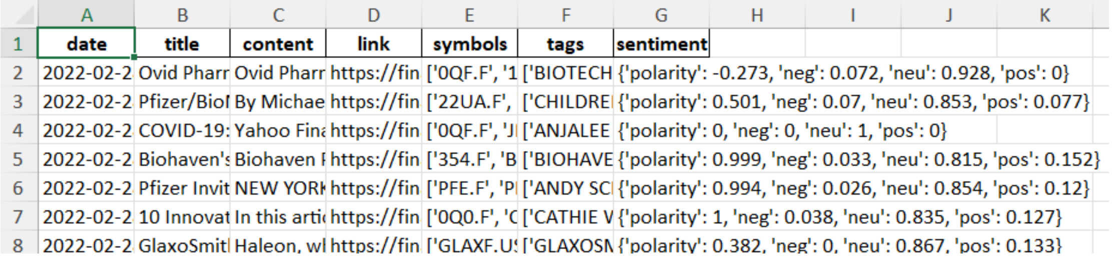


Figure 4.1: The distribution of data combed



```
keywords_dict = {
    "Fraud": [
        "embezzlement", "insider", "insider trading", "Ponzi scheme", "counterfeit", "corruption",
        "money laundering", "laundering", "falsification", "misrepresentation", "whistleblower",
        "SEC investigation", "audit failure", "financial irregularities", "kickbacks",
        "unethical practices", "unethical", "compliance breach", "breach", "scandal", "forgery", "deceptive",
        "swindle", "defraud"
    ],
    "Regulatory Issues": [
        "non-compliance", "fines", "penalty", "regulatory scrutiny", "scrutiny", "SEC fines",
        "sanctions", "violation", "compliance failure", "environmental regulation",
        "antitrust", "FDA warning", "safety standards", "legal challenge",
        "governmental probe", "regulation changes", "lobbying", "policy violation"
    ],
    "Product Failures/Recalls": [
        "recall", "defective", "malfunction", "safety concern", "product liability",
        "recall notice", "consumer safety", "hazardous", "withdrawal", "product risk",
        "quality control", "manufacturing defect", "product warning"
    ],
```

Figure 4.2: Snippet of chosen keyphrases from LDA



	A	B	C	D	E	F	G
1	date	title	content	link	symbols	tags	sentiment
2	2022-02-2	Ovid Pharr	Ovid Pharr	https://fin	['0QF.F', '1	['BIOTECH	{'polarity': -0.273, 'neg': 0.072, 'neu': 0.928, 'pos': 0}
3	2022-02-2	Pfizer/Bio	By Michae	https://fin	['22UA.F',	['CHILDRE	{'polarity': 0.501, 'neg': 0.07, 'neu': 0.853, 'pos': 0.077}
4	2022-02-2	COVID-19:	Yahoo Fina	https://fin	['0QF.F', 'J	['ANJALEE	{'polarity': 0, 'neg': 0, 'neu': 1, 'pos': 0}
5	2022-02-2	Biohaven's	Biohaven F	https://fin	['354.F', 'B	['BIOHAVE	{'polarity': 0.999, 'neg': 0.033, 'neu': 0.815, 'pos': 0.152}
6	2022-02-2	Pfizer Invit	NEW YORK	https://fin	['PFE.F', 'P	['ANDY SC	{'polarity': 0.994, 'neg': 0.026, 'neu': 0.854, 'pos': 0.12}
7	2022-02-2	10 Innovat	In this arti	https://fin	['0Q0.F', 'C	['CATHIE V	{'polarity': 1, 'neg': 0.038, 'neu': 0.835, 'pos': 0.127}
8	2022-02-2	GlaxoSmit	Haleon, wl	https://fin	['GLAXF.US	['GLAXOSM	{'polarity': 0.382, 'neg': 0, 'neu': 0.867, 'pos': 0.133}

Figure 4.3: News Articles dataset

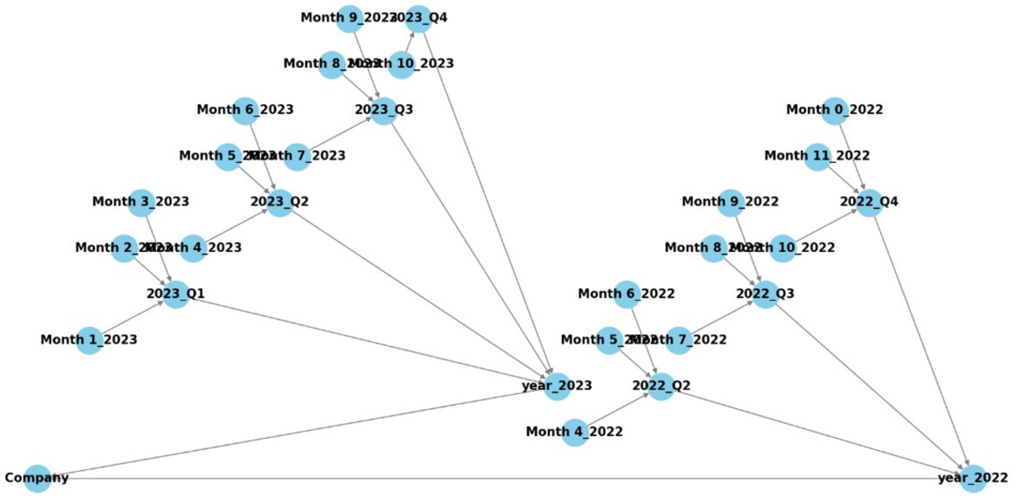


Figure 4.4: Preliminary hierarchical representation of data

4.2 Graph Creation

To effectively manage the diverse data from financial statements and news articles, we constructed a complex graph structure:

1. **Node Creation:** Nodes were created for different entities such as days, months, quarters, years, and companies, with each node containing relevant data attributes.

2. **Edge Formation:** Directed edges were established to show temporal and categorical relationships between nodes, supporting the propagation of information across the graph.

3. **Feature Engineering:** We engineered features on the nodes based on financial ratios, market trends, and sentiment analysis from news articles, enriching the node attributes for better prediction accuracy.

4. **Hierarchical representation:** Utilising all prior parameters, we developed a graph representation, describing the hierarchy of data being trained in the HeteroGraph CNN. Figure 4.4 represents the hierarchy

5. **Interconnected network rework** Attaching edge weights to consecutive nodes of the same label types, implementing on top of hierarchical data representation. Figure 4.5 shows the finalized Knowledge graph utilized to feed into the HeteroCNN model.

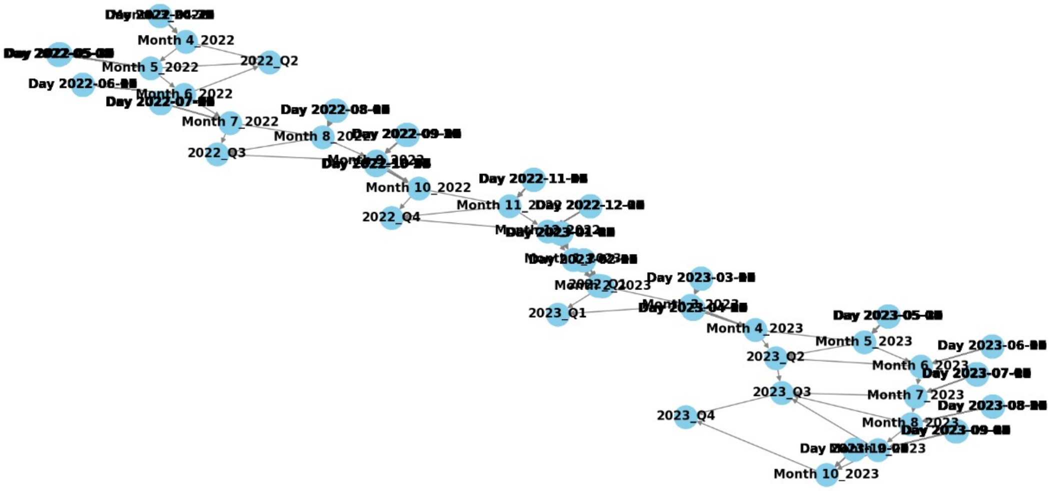


Figure 4.5: Knowledge graph of Company 'MSI'

4.3 Hetero CNN and Temporal CNN Performance

Our model utilizes Convolutional Neural Networks (CNNs) adapted for graph data, which perform feature extraction and risk classification:

1. **Training:** The model was trained using a batch-wise strategy, optimizing for a custom loss function - CrossEntropy loss for multi-label classification that accounts for the imbalanced nature of risk categories.

2. **Validation:** During validation, several metrics such as accuracy, precision, recall, and F1-score were monitored to gauge the performance and adjust hyperparameters accordingly.

3. **Testing:** The final testing involved comparing the CNN model's predictions with ground truth labels to evaluate its real-world applicability and reliability.

4.4 Results

4.4.1 Model Evaluation

The performance of the classification model was quantitatively assessed using several metrics to provide a comprehensive view of its effectiveness across multiple dimensions. The results are summarized in Table 4.8.

4.4.2 Confusion Matrix Analysis

The confusion matrix for the model is presented below:

Metric	Value
Accuracy	46.71%
Precision (Macro-Averaged)	47.07%
Recall (Macro-Averaged)	44.71%
F1 Score (Macro-Averaged)	41.13%
Balanced Accuracy	44.71%
Matthews Correlation Coefficient	0.34
Cohen's Kappa	0.32
Ratio Bin based correlation	71.93%
Normalized Square Error	81.60%

Table 4.1: Classification Performance Metrics

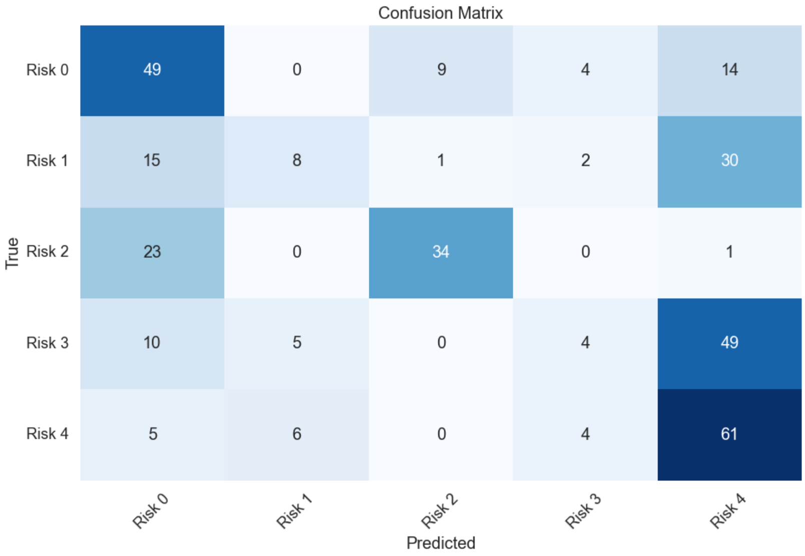


Figure 4.6: Overall Confusion Matrix

This matrix helps to visualize the accuracy of the classifier with respect to each class. The diagonal elements represent the number of points for which the predicted label is equal to the actual label, while the classifier mislabeled off-diagonal elements.

4.4.3 Discussion

Accuracy and Balanced Accuracy

The model's overall accuracy stands at approximately 46.71%, indicating that nearly half of the predictions made by the model are correct. The balanced accuracy, which considers the imbalance in class distribution, is similarly positioned at 44.71%. This suggests a moderate level of agreement between the predicted and actual class labels across all classes, which is particularly important given the varying sizes of each class in the dataset.

Precision and Recall

The model's macro-averaged precision and recall are approximately 47.07% and 44.71%, respectively. These metrics indicate the model's ability to identify relevant instances and its accuracy when labeling instances as positive across all classes. The relatively low values highlight potential areas for improvement, particularly in minimizing false positives and increasing the model's sensitivity.

F1 Score

The F1 score, a weighted average of precision and recall, stands at 41.13%. This score is helpful in cases where an equal balance between precision and recall is required. The lower F1 score compared to precision and recall suggests a notable disparity between these two metrics in certain classes.

Matthews Correlation Coefficient and Cohen's Kappa

The Matthews Correlation Coefficient (MCC) and Cohen's Kappa score are 0.344 and 0.320, respectively. These values indicate a moderate positive relationship between the predicted and actual classifications, factoring in the chance groupings that could occur in a random prediction scenario. Both metrics underscore that while the model does show predictive power beyond random chance, improvements are necessary to enhance reliability.

Ratio Bin based Correlation

This custom metric was introduced to account for how far the prediction is from the ground truth label. For example, a predicted value of 3 for ground truth 4 is better than that of 0 for the same ground truth. This calculates a deviation by multiplying the ratio of bin size by the difference and subtracting this from 1. If the model predicts shallow risk as the category for the extremely high-risk company, then the score is 0 as the difference in the label is 4.

$$Score = 1 - 0.25 * (abs(Predicted - GroundTruth))$$

Normalized Square Error

This custom metric considers the square of deviation from the target by introducing variance to the metric and normalizing the values. The score is then calculated by subtracting this normalized value from 1.

$$Score = 1 - ((Predicted - GroundTruth)^2)/16$$

ROC-AUC curve

This custom metric was introduced to account for how far the prediction is from the ground truth label. For example, a predicted value of 3 for ground truth 4 is better than that of 0 for the same ground truth. This calculates a deviation by multiplying the ratio of bin size by the difference and subtracting this from 1. If the model predicts extremely low risk as the category for the extremely high-risk company, then the score is 0 as the difference in the label is 4.

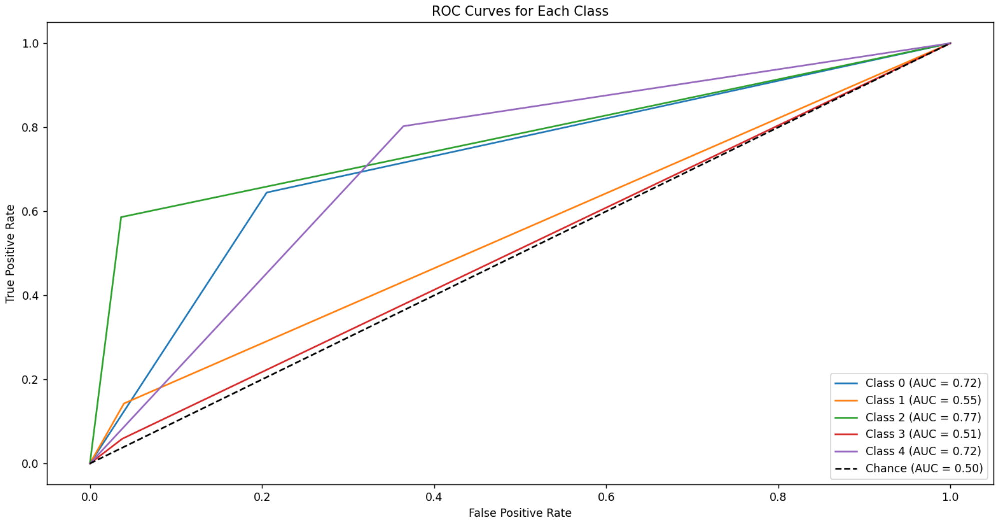


Figure 4.7: ROC-AUC curve for each separate class

4.5 Results by Sector Group

4.5.1 Sector-Specific Performance Analysis

We evaluate the classification model's performance across different sector groups. These groups represent unique categories within the business environment, each with distinct characteristics and risk profiles that may impact the prediction results.

Health Care

Confusion Matrix:

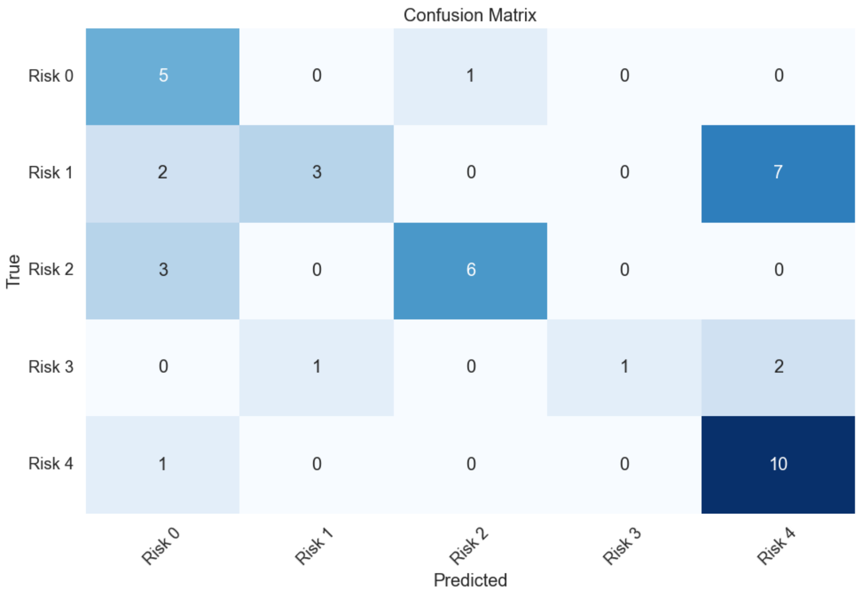


Figure 4.8: Health Care Confusion Matrix

Metrics:

Metric	Value
Accuracy	59.52%
Precision	71.76%
Recall	58.18%
F1 Score	55.60%
Balanced Accuracy	58.18%
Matthews Correlation Coefficient (MCC)	0.512
Cohen's Kappa	0.480
Ratio Bin based Correlation	76.78%
Normalized Square Error	84.67%

Table 4.2: Performance Metrics for the Health Care Sector

Discussion: The Health Care sector shows moderate accuracy. Higher precision effectively identifies true positives, which is particularly significant in predicting downside risk in a volatile health market. However, the variability in recall suggests challenges in consistently identifying all at-risk entities across different sub-sectors.

Consumer Discretionary and Consumer Staples

Confusion Matrix:

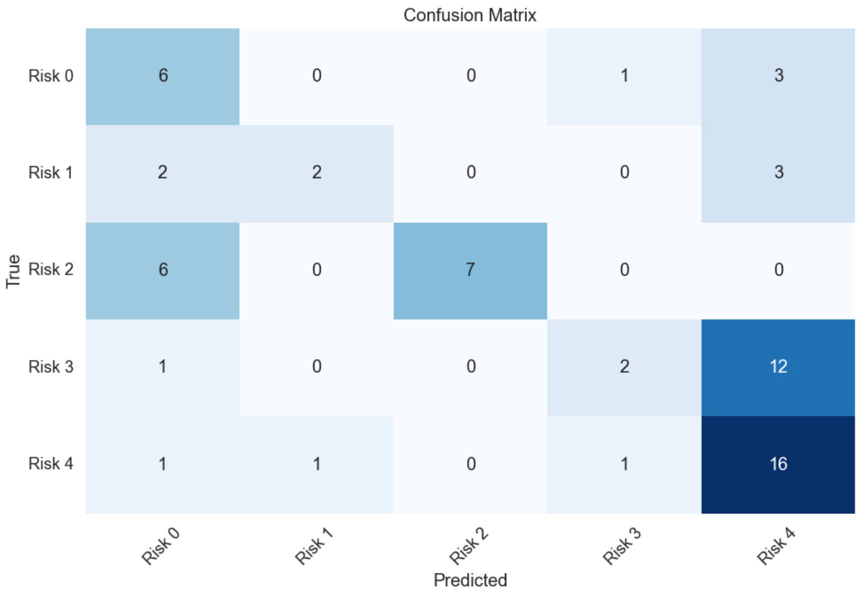


Figure 4.9: Consumer Discretionary and Consumer Staples Confusion Matrix

Metrics:

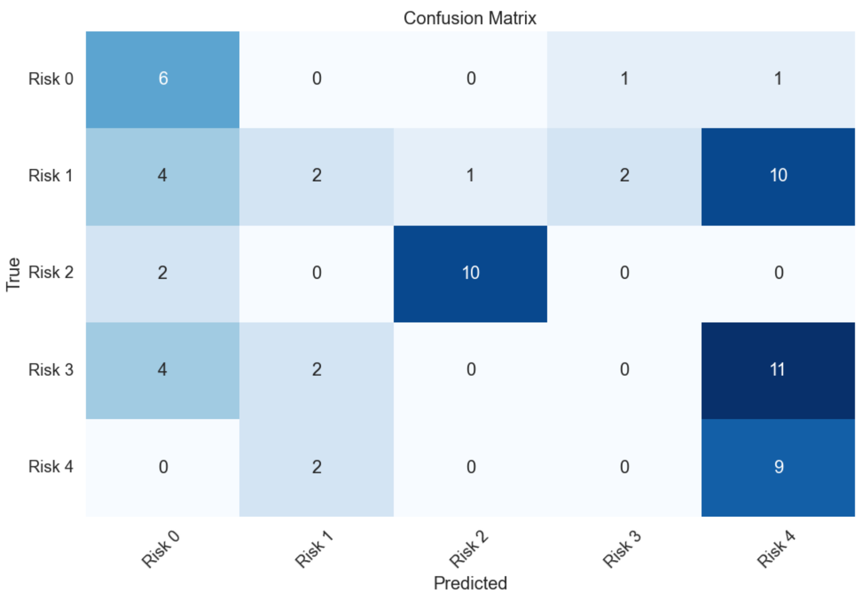
Metric	**Value**
Accuracy	51.56%
Precision	60.25%
Recall	47.99%
F1 Score	47.52%
Balanced Accuracy	47.99%
Matthews Correlation Coefficient (MCC)	0.393
Cohen's Kappa	0.364
Ratio Bin based Correlation	76.17%
Normalized Square Error	84.66%

Table 4.3: Performance Metrics for the Consumer Discretionary and Consumer Staples Sector

Discussion: Consumer sectors display lower accuracy and balanced accuracy, reflecting the diverse and dynamic nature of consumer behavior affecting risk assessment. Precision is higher than recall, indicating better performance in identifying true positives over false negatives, which is crucial for risk-averse strategies.

Information Technology and Communication Services

Confusion Matrix:



	Risk 0	Risk 1	Risk 2	Risk 3	Risk 4
Risk 0	6	0	0	1	1
Risk 1	4	2	1	2	10
Risk 2	2	0	10	0	0
Risk 3	4	2	0	0	11
Risk 4	0	2	0	0	9

Figure 4.10: IT and Communications Confusion Matrix

Metrics:

Metric	**Value**
Accuracy	40.30%
Precision	38.15%
Recall	50.14%
F1 Score	39.16%
Balanced Accuracy	50.14%
Matthews Correlation Coefficient (MCC)	0.316
Cohen’s Kappa	0.280
Ratio Bin based Correlation	69.02%
Normalized Square Error	80.50%

Table 4.4: Performance Metrics for the Information Technology and Communication Services Sector

Discussion: The high technological innovation rate and market volatility in IT and Communication Services may contribute to lower overall accuracy but higher recall. The sector’s complexity and rapid changes can hinder consistent classification performance.

Utilities

Confusion Matrix:

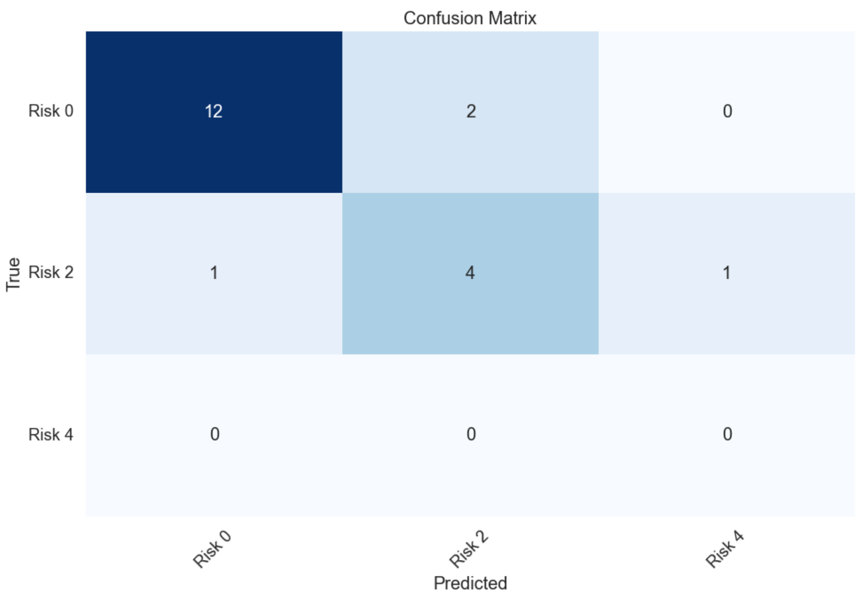


Figure 4.11: Utilities Confusion Matrix

Metrics:

Metric	**Value**
Accuracy	80.0%
Precision	52.99%
Recall	50.79%
F1 Score	51.85%
Balanced Accuracy	76.19%
Matthews Correlation Coefficient (MCC)	0.565
Cohen's Kappa	0.560
Ratio Bin based Correlation	90.00%
Normalized Square Error	95.00%

Table 4.5: Performance Metrics for the Utilities Sector

Discussion: The Utilities sector shows high and balanced accuracy, likely due to the regulated nature and less volatile environment, allowing more predictable risk assessment outcomes.

Financials and Real Estate

Confusion Matrix:

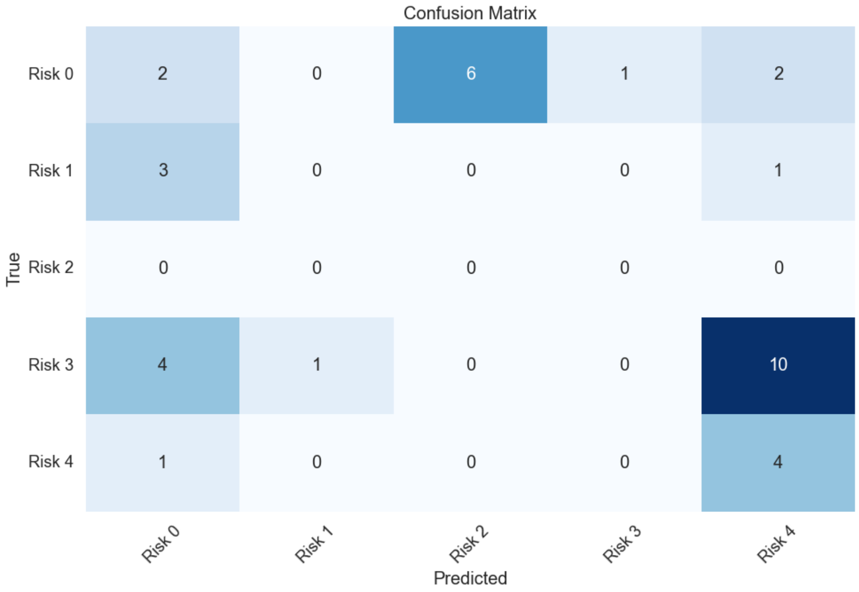


Figure 4.12: Financials and Real Estate Confusion Matrix

Metrics:

Metric	**Value**
Accuracy	17.14%
Precision	8.71%
Recall	19.64%
F1 Score	11.08%
Balanced Accuracy	24.55%
Matthews Correlation Coefficient (MCC)	-0.005
Cohen's Kappa	-0.004
Ratio Bin based Correlation	59.28%
Normalized Square Error	74.46%

Table 4.6: Performance Metrics for Financials and Real Estate Sector

Discussion: Financial and Real Estate sectors exhibit significantly lower accuracy, likely impacted by the complex risk factors and economic fluctuations that are challenging to predict with standard models.

Industrials and Materials

Confusion Matrix:

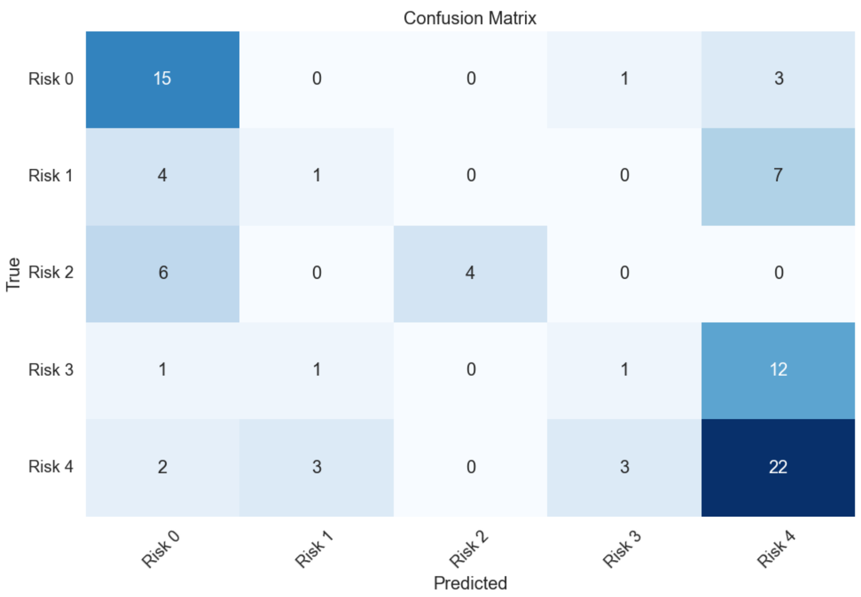


Figure 4.13: Industrials and Materials

Metrics:

Metric	**Value**
Accuracy	50.0%
Precision	48.71%
Recall	41.46%
F1 Score	40.44%
Balanced Accuracy	41.46%
Matthews Correlation Coefficient (MCC)	0.327
Cohen's Kappa	0.311
Ratio Bin based Deviation	74.12%
Normalized Square Error	82.92%

Table 4.7: Performance Metrics for Industrials and Materials Sector

Discussion: The mixed accuracy in these sectors can be attributed to the diversity within industrial applications and material production processes, affecting the uniformity and predictability of risk assessments.

Energy

Confusion Matrix:

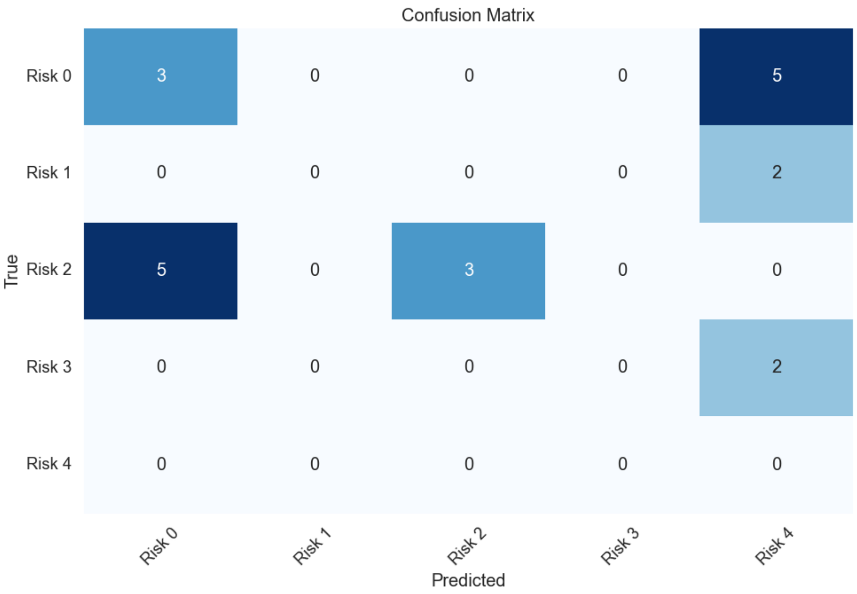


Figure 4.14: Energy Confusion Matrix

Metrics:

Metric	**Value**
Accuracy	30.0%
Precision	27.5%
Recall	15.0%
F1 Score	18.41%
Balanced Accuracy	18.75%
Matthews Correlation Coefficient (MCC)	0.126
Cohen's Kappa	0.103
Ratio Bin based Deviation	52.50%
Normalized Square Error	62.50%

Table 4.8: Performance Metrics for Energy Sector

Discussion: The Energy sector faces the lowest accuracy levels, possibly due to the high impact of external geopolitical and environmental factors, which are challenging to model accurately in risk assessments.

4.6 Conclusion

Chapter 4 presents a comprehensive analysis of the CNN model performance. The results highlight the efficacy of integrating graph-based representations and

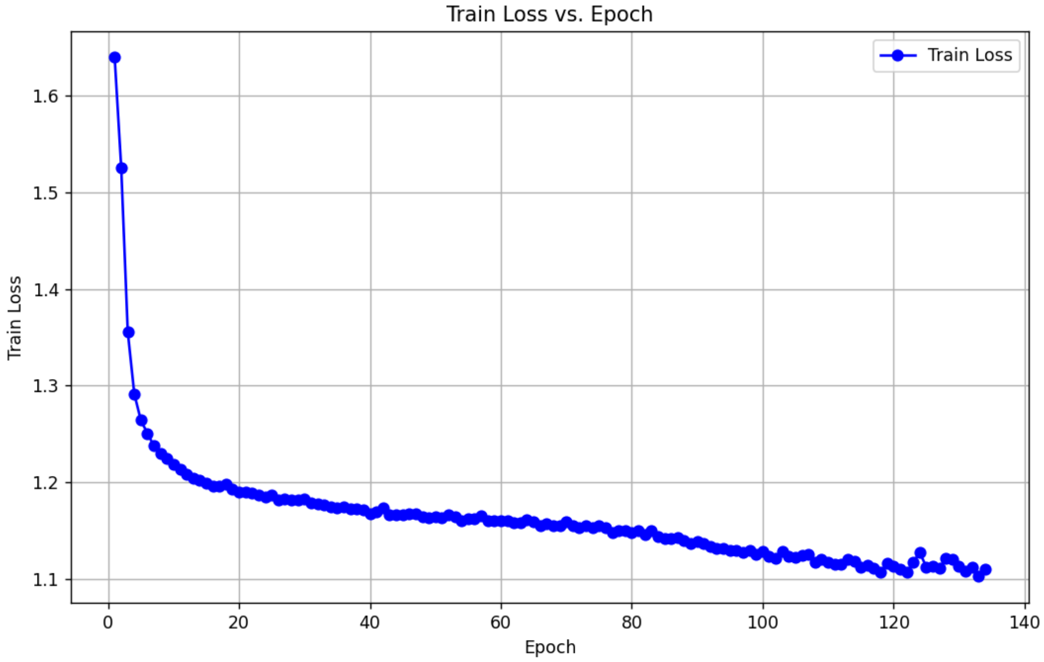


Figure 4.15: Train Loss vs Epochs

CNNs for financial risk assessment. Future work will focus on incorporating more dynamic data sources and improving the model's real-time prediction capabilities.

CHAPTER 5
CONCLUSION AND FUTURE WORK

5.1 Summary

By combining various data sources and utilizing cutting-edge machine learning techniques the project aims to improve how financial risk is classified for businesses. It starts off by emphasizing how important risk assessment is when making investment and risk management decisions. Conventional methods frequently depend only on financial data however this project presents a unique methodology that integrates textual data extracted from news articles obtained through the Eodhd API with financial data from sources such as SimFin.

The strategy seeks to offer a more thorough and nuanced understanding of a companys risk profile by integrating these various data streams. According to the approach every company is represented as a graph with nodes for different financial characteristics and textual representations taken from news stories. The dependencies and relationships between various data points are captured by the edges that connect nodes. Quantitative financial metrics and qualitative contextual information from news articles can both be included in this structured framework for analysis provided by this graph-based representation.

The project uses graph neural networks (GNNs) to derive meaningful insights from these intricate graphs. Specialized deep learning models called GNNs are made to work with graph-structured data. The technique extracts hierarchical representations that capture the underlying risk factors and patterns in the data by applying GNNs to the created graphs. This makes it possible to classify risks more precisely and nuancedly by accounting for both external factors reflected in news sentiment and financial performance metrics.

The projects outcomes show how well the suggested methodology improves risk assessment for businesses. The approach improves risk classification accuracy and comprehensiveness by utilizing cutting-edge machine learning techniques and the integration of multiple data sources.

The projects conclusion emphasizes the possibility of more developments and advances in this field of study with continuous improvements in machine learning algorithms graph-based modeling and data integration predicted to propel future advances in financial risk assessment and management.

5.2 Conclusion

Using financial data and news articles together, this thesis offered a thorough method for assessing financial risk that improves risk classifications precision and depth. Risk can now be identified and categorized more accurately than with conventional techniques thanks to the creative application of Graph Neural Networks (GNNs) to process these integrated data sources.

5.2.1 Key Findings

GNNs were used to explore intricate relationships in the data that were difficult to uncover with conventional flat-file database structures. Our model could capture the subtleties and dependencies that are essential for evaluating risk in real-time by depicting companies as graphs that combine financial attributes and textual data from news articles. The graphs hierarchical structure made sure that relationships at different levels—day month quarter and year—were taken into account offering a multifaceted perspective on risk that is representative of real market dynamics.

5.2.2 Methodological Contributions

Our approach moved beyond the conventional risk assessment models by:

- Integrating heterogeneous data sources, thus providing a richer dataset for analysis.
- Employing advanced machine learning techniques to interpret complex data structures.
- Utilizing temporal and structural data attributes to improve the prediction accuracy.

5.2.3 Practical Implications

The study's findings have practical implications for financial analysts, investors, regulators, and other players in the financial sector. Providing a more thorough and dynamic view of risk helps the stakeholders make more informed decisions that could lead to better financial outcomes and risk mitigation strategies.

5.2.4 Concluding Remarks

In summary, this thesis has successfully demonstrated the benefits and feasibility of applying graph-based representations and machine learning to improve financial risk assessment. In financial analysis, combining multiple data sources into a coherent graph-based model is a significant advancement. The focus of future work will be on refining these models for real-time analysis, increasing computational efficiency, and expanding the models' applicability to other financial risk domains outside market risk.

5.3 Constraints of the project

The project faced several major obstacles that affected its efficacy even though the combination of news articles and financial data represents a promising way to improve risk classification. The lack of diversity in the dataset was the most significant of these limitations. The projects use of the dataset might have been constrained in terms of the industries it covered or the kinds of financial metrics and news sources it contained. The absence of diversity may have limited the findings generalizability and introduced biases into the risk classification model.

The datasets size also turned out to be a significant limitation affecting the risk classification models robustness and accuracy. In order for machine learning models to effectively learn complex patterns especially deep learning algorithms such as graph neural networks they frequently need large volumes of data.

The collection of news articles for companies was a tedious task. The model may not have been able to fully capture the range of risk factors and patterns found in real-world financial data due to the relatively small size of the dataset used in the project. To improve the precision and resilience of the risk classification model overcoming these limitations would require access to larger and more varied datasets covering a wider range of businesses sectors and regions.

5.4 Future Scope

The project establishes the foundation for multiple future research and development directions. First broadening the dataset to encompass a wider variety of businesses sectors and geographical areas would improve the models applicability and efficiency in risk evaluation. Gaining access to a larger dataset may make it easier to find patterns and risk factors unique to a given industry leading to more accurate and sophisticated risk classification.

Combining data from sources other than news articles and financial metrics may offer a more complete picture of risk factors. For example adding macroeconomic data, legal documents ,sentiment analysis from social media or other data sources could enhance the risk assessment model by giving insightful information about general market trends and sentiment.

Within the existing dataset, particularly news articles, a weight indicating the severity or importance of the article can help give better results. Insights into the models efficacy usability and influence on decision-making processes could be gained by working with financial institutions or industry partners to put it into reality.

All things considered the projects future scope will include a multidisciplinary approach that will improve financial risk assessment and management practices by combining various data sources cutting-edge machine learning techniques and practical applications. It is possible that substantial progress in risk mitigation techniques and investment decision-making will come from ongoing research and innovation in this field.

REFERENCES

1. Y. Zhao et al., “Multi-granularity heterogeneous graph attention networks for extractive document summarization,” Neural Networks, vol. 155, Issue C, pp. 340–347, Nov. 2022
https://doi.org/10.1016/j.neunet.2022.08.021

2. B. Zhou et al., “Forecasting credit default risk with graph attention networks,” Electronic Commerce Research and Applications, vol. 62, p. 101332, Nov. 2023
https://doi.org/10.1016/j.elerap.2023.101332

3. X. Cheng et al., “Combating emerging financial risks in the big data era: A perspective review,” Fundamental Research, vol. 1, no. 5, pp. 595–606, Sep. 2021
https://doi.org/10.1016/j.fmre.2021.08.017

4. A. Gu and T. Dao, “Mamba: Linear-Time Sequence Modeling with Selective State Spaces.” ICLR 2024
https://doi.org/10.48550/arXiv.2312.00752

5. S. Motie and B. Raahemi, “Financial fraud detection using graph neural networks: A systematic review,” Expert Systems with Applications, vol. 240 p. 122156, Oct. 2023
https://doi.org/10.1016/j.eswa.2023.122156

6. A. Abraham, B. Nath, and P. K. Mahanti, “Hybrid Intelligent Systems for Stock Market Analysis,” Computational Science - ICCS 2001, pp. 337–345, 2001
https://doi.org/10.1007/3-540-45718-6_38.

7. M. T. Leung, A.-S. Chen, and R. Mancha, “Making trading decisions for financial-engineered derivatives: a novel ensemble of neural networks using information content,” Intelligent Systems in Accounting, Finance & Management, vol. 16, no. 4, pp. 257–277, Oct. 2009
https://doi.org/10.1002/isaf.308.

8. Huang, H., Pasquier, M., & Quek, C. (2009). Financial market trading system with a hierarchical coevolutionary fuzzy predictive model. IEEE Transactions on Evolutionary Computation, 13(1), 56-70.
https://ieeexplore.ieee.org/document/4769012

9. Priyanka, J. Woo (2019). “Financial fraud detection adopting distributed deep learning in big data” ,2019 International Conference on Computational Science and Computational Intelligence (CSCI), pp. 1467-1472. IEEE. https://www.researchgate.net/publication/349651354

10. Qin, Y., Song, D., & Zhu, H. (2020). An Attention-Based LSTM Model for Stock Price Trend Prediction Using Limit Order Books. IEEE Access, 8, 192112-192122. https://ui.adsabs.harvard.edu/link_gateway/2020JPhCS1575a2124L/doi:10.1088/1742-6596/1575/1/012124