MULTI-CLASS CLASSIFICATION OF GOOD DISTRIBUTION PRACTICE (GDP) INSPECTION FINDINGS USING MACHINE LEARNING

FRANCIS TENG CHUEN CHUING

UNIVERSITI KEBANGSAAN MALAYSIA

MULTI-CLASS CLASSIFICATION GOOD DISTRIBUTION PRACTICE (GDP) INSPECTION FINDINGS USING MACHINE LEARNING

FRANCIS TENG CHUEN CHUING

PROJECT SUBMITTED IN PARTIAL FULFILMENT FOR THE DEGREE OF MASTER OF DATA SCIENCE

FACULTY OF INFORMATION SCIENCE AND TECHNOLOGY
UNIVERSITI KEBANGSAAN MALAYSIA
BANGI

2024

MULTI-CLASS CLASSIFICATION OF GOOD DISTRIBUTION PRACTICE (GDP) INSPECTION FINDINGS USING MACHINE LEARNING

FRANCIS TENG CHUEN CHUING

PROJEK YANG DIKEMUKAKAN UNTUK MEMENUHI SEBAHAGIAN
DARIPADA SYARAT MEMPEROLEHI IJAZAH
SARJANA SAINS DATA

FAKULTI TEKNOLOGI DAN SAINS MAKLUMAT
UNIVERSITI KEBANGSAAN MALAYSIA
BANGI
2024

DECLARATION

I hereby declare that the work in this thesis is my own except for quotations and summaries, which have been duly acknowledged.

25 June 2024

FRANCIS TENG CHUEN CHUING
P119163

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to all those who have supported me throughout this journey. Firstly, I would like to thank my family, especially my spouse, Tham Su Ann, for their unwavering support, encouragement, and belief in me. Without them, this accomplishment would not have been possible.

I would like to express my deepest gratitude to my research supervisor, Assoc. Prof. Dr. Nazlia Omar, for her invaluable guidance, advice, and support throughout the project. Her knowledge and expertise were instrumental in helping me complete this study and achieve my goals.

Additionally, I extend my heartfelt appreciation to my friends and colleagues who have offered their encouragement, support, and words of wisdom throughout the process. Their unwavering support has been a source of strength and motivation for me.

I also would like to give appreciation to National Pharmaceutical Regulatory Agency (NPRA) for providing me invaluable Good Distribution Practice inspection reports. Without these reports I wouldn't be able to generate the dataset for the study. Finally, I want to extend my appreciation to Ministry of Health for granting me scholarship to pursue my master's studies.

ABSTRAK

Rantaian bekalan farmaseutikal adalah kompleks dan sering melibatkan pelbagai pihak seperti pengilang, pengimport, pemborong, pengedar dan peniaga runcit. Bahagian Regulatori Farmasi Negara (NPRA) di bawah Kementerian Kesihatan Malaysia, bertanggungjawab untuk memastikan kesihatan awam dilindungi dengan mengawasi keseluruhan rantaian bekalan farmaseutikal di Malaysia. NPRA telah menerbitkan Garispanduan Amalan Pengedaran Baik (AEB) untuk memastikan standard kualiti yang tinggi dan integriti proses pengedaran produk. Pemeriksa NPRA akan menjalankan pemeriksaan AEB untuk mengesahkan status pematuhan pemegang lesen berdasarkan keperluan yang dinyatakan di bawah garispanduan AEB dan menghasilkan laporan pemeriksaan AEB dalam tulisan Bahasa Melayu. Terdapat pelbagai kajian pengelasan berdasarkan topik, seperti mengkategorikan artikel berita kepada segmen-segmen seperti Sains-Teknologi, Perniagaan-Kewangan, Sukan, dan Gaya Hidup-Rekreasi. Walau bagaimanapun, tidak ada kajian pengelasan teks berasaskan pembelajaran mesin yang berkaitan dengan pengelasan penemuan audit AEB. Objektif kajian ini adalah untuk mengenal pasti pewakilandokumen dan pengelas yang sesuai untuk mengkategorikan isu ketidakpatuhan dalam laporan pemeriksaan AEB Bahasa Melayu dengan tepat dan melakukan kajian perbandingan tentang prestasi pengelas mesin yang berbeza. Set data yang digunakan dalam kajian ini terdiri daripada 1700 komen penemuan yang diekstrak daripada 104 laporan pemeriksaan. Set data tersebut kemudian dilabel mengikut bab garispanduan AEB dan menjalani beberapa operasi pra-pemprosesan, termasuk tokenisasi, dan penghapusan kata henti. Ia kemudian diubah melalui teknik pewakilan dokumen seperti Bag of Words, TF-IDF, dan Bigrams, diikuti dengan membandingkan kombinasi pelbagai pewakilan dokumen dengan pengelas yang berbeza (Naïve Bayes, Regresi Logistik, Mesin Vektor Sokongan, dan k-Nearest-Neighbor). Hasil kajian menunjukkan bahawa ciri yang paling teguh untuk mengklasifikasikan penemuan pemeriksaan AEB adalah TF-IDF dan pengelas yang paling teguh untuk mengklasifikasikan penemuan pemeriksaan AEB adalah Regresi Logistik. Ini mungkin bermakna bahawa gabungan Regresi Logistik dengan TF-IDF boleh menjadi pilihan model untuk tugas klasifikasi ini. Sebaliknya, hasil kajian menunjukkan bahawa gabungan Bag of Words dengan Regresi Logistik memberikan klasifikasi yang paling tepat bagi penemuan pemeriksaan AEB Bahasa Melayu di mana model ini mencapai ketepatan 0.92, ketepatan 0.93, kejituan 0.92, dan ukuran F1 0.92 dan mengalahkan model asas (BoW + NB) dalam semua metrik.

ABSTRACT

The pharmaceutical supply chain is complex, often involving multiple stakeholders like manufacturers, importers, distributors, wholesalers, and retailers. The National Pharmaceutical Regulatory Agency (NPRA), which operates under the Ministry of Health Malaysia, is responsible for ensuring public health is protected by overseeing Malaysia's entire pharmaceutical supply chain. NPRA has published the "Guideline on Good Distribution Practice (GDP)" to ensure high quality standards and the integrity of distribution processes. NPRA inspector will conduct a GDP inspection to verify the compliance status of the license holder based on the requirements stated under the GDP guidelines and produce a Malay texted GDP inspection report. There are numerous topic-based classification studies, such as categorising news articles into segments like Science-Technology, Business-Finance, Sports, and Lifestyle-Leisure. However, there are no machine learning-based text classification studies related to GDP audit finding classification. The study's objectives are to identify suitable features and classification algorithms for accurately categorizing non-compliance issues in Malay GDP inspection reports and performing a comparative study on the performance of machine learning classifiers. The dataset used in this study consisted of 1700 non-conformance comments extracted from 104 inspection reports. The dataset was then labelled according to the GDP guideline's chapter and underwent several pre-processing operations, including character processing, tokenisation and stopword removal. It was then transformed through feature extraction techniques (Bag of Words, TF-IDF, and Bigrams), followed by comparing the combination of various features with different classifiers (Naïve Bayes, Logistic Regression, Support Vector Machine, and k-Nearest-Neighbor). Results show that the most robust feature for classifying GDP inspection findings is TF-IDF and the most robust classifier for classifying GDP inspection findings is Logistic Regression. This may imply that a combination of Logistic Regression with TF-IDF could be the model choice for this classification task. On the contrary, the study showed that combining Bag of Words with Logistic Regression yielded the most accurate classification of Malay GDP inspection findings. This model achieved an accuracy of 0.92, precision of 0.93, recall of 0.92, and F1 score of 0.92 and outperformed the baseline model (BoW + NB) in all the metrics.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

LIST OF ABBREVIATIONS

BOW	Bag of Words
EDA	Exploratory Data Analysis
EEA	European Economic Area
EMEA	European Medicines Agency
GDP	Good Distribution Practice
GMP	Good Manufacturing Practice
ISO	International Organization for Standardization
KNN	k Nearest Neighbor
LR	Logistic Regression
MHRA	Healthcare Products Regulatory Agency
NB	Naïve Bayes
NLP	Natural Language Processing
NPRA	National Pharmaceutical Regulatory Agency
PIC/S	Pharmaceutical Inspection Co-operation Scheme
POS	Part of Speech
QMS	Quality Management System
SVM	Support Vector Machine
TF-IDF	Term Frequency-Inverse Document Frequency
UKM	Universiti Kebangsaan Malaysia
USFDA	U.S. Food and Drug Administration
WHO	World Health Organization

CHAPTER I

INTRODUCTION

1.1 RESEARCH BACKGROUND

In recent years, the worldwide pharmaceutical market has undergone substantial expansion. The total valuation of the global pharmaceutical market for the year 2022 was approximated at USD 1.48 trillion (Mikulic 2023). It is expected to reach USD 1.5 trillion by 2025, driven by factors like an ageing population, rising healthcare costs, and increasing prevalence of chronic diseases.

The pharmaceutical supply chain is complex, often involving multiple stakeholders like manufacturers, distributors, wholesalers, and retailers. Distribution plays a vital role in integrated supply chain management amid various challenges, such as counterfeiting and temperature control issues, especially for biologics and vaccines, which require strict temperature control during transportation and storage. Besides that, with the globalisation of the pharmaceutical industry, the distribution network often extends across borders, necessitating adherence to diverse regulatory requirements and logistical complexities. It becomes crucial to have effective control over the entire supply chain, from manufacturing to end-user delivery of pharmaceutical products, to ensure product quality and integrity throughout the supply chain. Therefore, regulatory bodies worldwide have established guidelines like the World Health Organization (WHO) Good Distribution Practice (GDP) guideline and Pharmaceutical Inspection Co-operation Scheme (PIC/S) Guide to Good Distribution Practice For Medicinal Products for practising consistent standards for the pharmaceutical industry.

Malaysia's pharmaceutical industry comprises over 275 licensed pharmaceutical manufacturers, with 64 per cent (176 manufacturers) focusing on traditional medicine and health supplements, 32 per cent (88 manufacturers) engaged in pharmaceutical production, and 4 per cent (11 manufacturers) specialising in veterinary products, 474 licensed importers and 1206 licensed wholesalers (Mida 2022). Licensed manufacturers within this industry offer a diverse range of pharmaceuticals, encompassing novel drug formulations, biologics, generic and over-the-counter medications, health supplements, traditional medicines, and food supplements. Local pharmaceutical companies play a crucial role, primarily in producing generic drugs, traditional medicines, and herbal supplements. Additionally, they serve as contract manufacturers for multinational companies. These manufacturers possess the capacity and expertise to produce various dosage forms, including sterile injections, sterile eye drops, tablets, hard capsules, soft gelatin capsules, and time-release products.

Some major players in the local pharmaceutical manufacturing industry include Pharmaniaga Manufacturing Berhad, Duopharma Biotech Berhad, Kotra Pharma (M) Sdn. Bhd., and Hovid Berhad. The production of generic drugs, including antibiotics, painkillers, health supplements, and injectables, is a core focus for these companies. Besides that, there are foreign pharmaceutical manufacturers in the country include Biocon Sdn. Bhd. (a subsidiary of India's Biocon Ltd.), Novugen Pharma (Malaysia) Sdn. Bhd. (part of UAE's Scitech International), Y.S.P. Industries (M) Sdn. Bhd. (Taiwan), Sterling Drug (M) Sdn. Bhd. (the manufacturing arm of the UK's Haleon), Ranbaxy (M) Sdn. Bhd. (a division of India's Sun Pharmaceutical Industries Ltd.), Xepa-Soul Pattinson (M) Sdn. Bhd. (Singapore), and SM Pharmaceutical Sdn. Bhd. (India) (MIDA 2022).

The drug distribution landscape is quite diverse, with both large multinational companies and smaller local players involved in the distribution network. Generally, major global pharmaceutical industry players act as licensed importers and distribute their branded drugs through locally incorporated entities. Some of the notable examples of such players include Pfizer Inc. (US), Schering-Plough, Eli Lilly & Co., AstraZeneca plc (UK), and Novartis International AG (Switzerland) (MIDA 2022).

The National Pharmaceutical Regulatory Agency (NPRA), which operates under the Ministry of Health Malaysia, is responsible for ensuring public health is protected by overseeing Malaysia's entire pharmaceutical supply chain. NPRA has published the Guideline on Good Distribution Practice (GDP) to maintain high quality standards and ensure the integrity of distribution processes. This guideline provides a comprehensive framework for all stakeholders involved in the supply chain. This guideline establishes fundamental principles for various stakeholders involved in the supply chain, including manufacturers of active pharmaceutical ingredients, product/cosmetic manufacturers, packaging/repackaging operations, importers, exporters, wholesale distributors, logistics providers, freight forwarders, pharmacies (such as retail, compounding, and hospital pharmacies), and healthcare professionals responsible for storing products prior to dispensing or administering to patients (NPRA 2018).

NPRA inspector will conduct a GDP inspection to verify the compliance status of the license holder based on the requirements stated under the GDP guidelines to ensure the maintenance of high standards of quality assurance and integrity of the distribution processes (NPRA 2020).

After the GDP inspection is performed, NPRA will issue a GDP inspection report for the auditee. The inspector will write down the conformance and non-conformance (findings) in the report for the auditee based on GDP guidelines, together with the compliance level of the premise. Besides that, NPRA will determine the inspection frequency for a premise based on risk factors such as compliance level, product ranges and the company size. The inspection frequency varies from once a year to once every five years (NPRA 2020).

NPRA conducts regular GDP inspections to assess compliance and identify non-conformances or deviations from GDP requirements. A total of 613 GDP inspections were conducted in 2022 involving 149 importers and 430 wholesalers (NPRA 2023). Analysing and categorising GDP inspection findings is a challenging and time-consuming task due to the large number of reports generated and the need for subject matter experts. Manual analysis often suffers from subjectivity and limitations in

scalability. Hence, leveraging text mining techniques and machine learning capability can provide automated, objective, and efficient methods to extract insights and categorise non-conformances from the reports, supporting NPRA's efforts to improve pharmaceutical distribution practices in Malaysia.

This study aims to utilise Natural Language Processing (NLP) techniques such as text classification to extract critical information and categorise non-compliance from GDP inspection reports. Automated analysis has significant potential in the inspection industry. However, the accuracy of the developed approach may be impacted by the variability in inspector writing styles and the complexity of non-conformance descriptions. Thus, it is crucial to account for the potential inconsistencies in writing styles and the non-conformance descriptions to ensure the accuracy of automated analysis. Consequently, it is essential to develop algorithms that can account for the unique writing styles of inspectors and the complexities of non-conformance descriptions to enhance the accuracy of automated analysis. This study has the potential to significantly improve the inspector’s efficiency in identifying and categorising the non-conformances from inspection reports. It can provide insights and ultimately improve the quality management system of the pharmaceutical supply chain in Malaysia.

1.2 PROBLEM STATEMENT

Malay language, characterised by distinct morphological and syntactic differences from its counterparts, remains underrepresented in text classification research (Nazratul Naziah Mohd et al. 2021). Most text classification field research has predominantly revolved around English and other extensively examined languages, with limited attention given to the Malay language. Therefore, limited availability of dataset like labelled datasets for use in Malay text classification. Features used in other Malay text classification might not fit well for GDP audit findings. Furthermore, the use of advanced features such as word embeddings is not as feasible for Malay amid limited availability of pre-trained model. There is lack of standardized benchmarks and evaluation metrics specifically designed for Malay text classification. This makes it difficult to compare the performance of different models and approaches effectively.

There are numerous topic-based classification studies, such as categorising news articles into segments like Science-Technology, Business-Finance, Sports, and Lifestyle-Leisure. However, there are no AI-based text classification studies related to GDP audit finding classification, and the existing literature primarily revolves around the ISO 9001:2015 Quality Management System (Corpuz 2021; Tarnate & Devaraj 2019). Using an existing established classifier for a news categorisation might not fit well for a new domain like GDP audit findings. Hence, a specific classifier for the domain needs to be evaluated.

Therefore, this scarcity of research on text classification in Malay and GDP inspection findings further underscores the need to develop suitable text mining techniques and machine learning algorithms to classify Malay GDP inspection findings accurately.

1.3 RESEARCH OBJECTIVES

The objectives of this study are as below:

1. To identify suitable features and classifiers for classifying Malay GDP inspection findings.

2. To perform a comparative study on the performance of machine learning classifiers.

1.4 RESEARCH SCOPE

The scope of this study was to identify an optimal combination of feature extraction methods and machine learning algorithms for the accurate categorisation of Malay GDP inspection results. Various feature extraction techniques, including Bag of Words (BoW), TF-IDF, and Bigrams, are investigated on the dataset (GDP inspection findings in 2022). BoW is effective in capturing word frequency and presence, providing a baseline for comparison with other features. It is also computationally less intensive, making it suitable for initial analysis. TF-IDF enhances the BoW approach by reducing the impact of common words that may not be informative while highlighting more

unique terms that are likely to be more important for each chapter in Good Distribution Practice guideline. By considering word pairs, bigrams can provide information on relationships between words, potentially improving the model's ability to classify more complex or context-dependent inspection findings. N-grams like 3-Grams and 4-Grams are excluded in the research scope because the inspection findings comments are short texts. In short texts, the occurrence of specific 3-grams or 4-grams can be very sparse. This might causes classifier prone to overfitting. Subsequently, supervised machine learning algorithms such as Naïve Bayes (NB), Logistic Regression (LR), Support Vector Machine (SVM), and k Nearest Neighbor (KNN) will be used in this study because the dataset is to be labelled according to chapters. NB is well-suited for text classification tasks due to its efficiency and relatively good performance with high-dimensional data. LR is a widely used linear model for binary classification, but it can be extended to multi-class classification through softmax regression. It models the probability of class membership directly, making it interpretable. SVM is particularly effective in high-dimensional spaces and for datasets where classes are well-separated whereas k-NN can capture local patterns in the data that other models might miss. Its simplicity and different approach compared to other classifiers provide a valuable comparison consideration. The outcomes of these combinations of feature extraction and machine learning algorithms were compared to determine the most effective model for classifying Malay GDP inspection findings. It is important to note that the proposed model is designed explicitly for classifying data based on the 11 chapters in the GDP guideline, excluding Annex 1. This model is intended to assist GDP inspectors in the automated and unbiased analysis of GDP inspection reports. Besides that, this model can also assist GDP inspectors in categorising inspection findings accurately for GDP report writing.

1.5 SIGNIFICANCE OF PROJECT

The successful development and implementation of an automated text mining solution for Malay GDP inspection findings classification will have significant implications for the regulatory authority. Through automated classification, it allows for the efficient and rapid processing of a large volume of inspection reports. This efficiency can help

regulatory authorities and organisations save time and resources in managing and analysing non-conformance findings.

Text mining techniques also ensure that the analysis is automated and unbiased, reducing the potential for human error due to manual classification entry for analysis.

1.6 ORGANIZATION PROJECT

The methodology illustrates the mechanism of performing this study. Figure 1.1 demonstrates a methodology and its various stages, which can be summarised as follows:

1. Chapter I: This chapter introduces the foundational aspects of the study, including the study's background description. It also formulates the research gap within the problem statement, emphasising the specific challenge the research aims to address. Furthermore, the study's objectives, which outline the methodologies to overcome this challenge, are articulated clearly.

2. Chapter II: This chapter provides an extensive review of the relevant literature pertaining to the text classification of the Malay language and various document classification methods. The chapter elucidates the methodologies employed in document classification, explores the domain of Malay language text mining, and examines the machine learning algorithms applied in the context of text classification. Furthermore, a comprehensive critical analysis of the pertinent prior research is presented in this chapter.

3. Chapter III: This chapter delineates the research methodology employed in this study, encompassing aspects such as the dataset's source, pre-processing procedures, feature extraction techniques, and the machine learning algorithms incorporated in the analysis.

4. Chapter IV: This chapter elaborates on the results of the comparative study combining feature extraction techniques with different machine learning algorithms.

5. Chapter V: The final chapter serves as the culmination of the study, delivering a concise summary of the research's findings and contributions. Additionally, it offers valuable insights into potential areas for future exploration and exploitation.

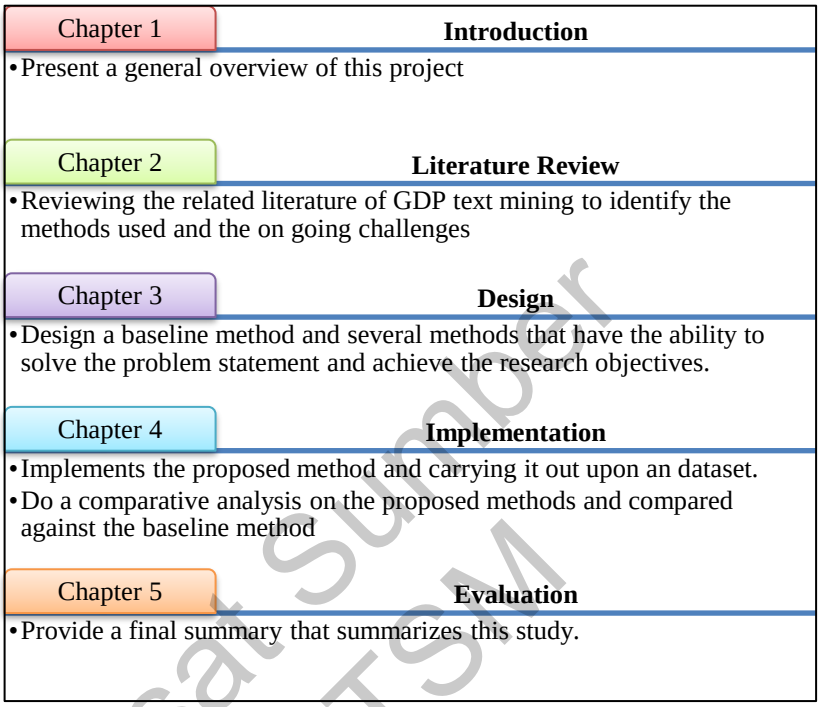


Figure 1.1 Research methodology stages

CHAPTER II

LITERATURE REVIEW

2.1 INTRODUCTION

Good Distribution Practice (GDP) is a crucial pharmaceutical and healthcare supply chain aspect. It ensures the safe and efficient distribution of medicinal products from the manufacturer to the end user. GDP involves a process of inspection, monitoring, and regulation to maintain the integrity and quality of pharmaceuticals throughout the supply chain from manufacturer to the end user.

A huge amount of unstructured text data is present in GDP inspection reports, including non-conformance findings and comments provided by inspectors. It provides an opportunity as well as a challenge to gain insights efficiently, especially since the comments and findings are written in Malay language. Text mining, a rapidly growing natural language processing and data analysis subfield, provides a powerful tool for extracting meaningful information from unstructured textual data. By leveraging this technique, researchers and organisations can gain valuable insights, improve decision-making processes, and ultimately drive more positive outcomes. Specifically, the application of text mining techniques to classify and categorise Malay Language GDP inspection findings has the potential to revolutionise the way to approach regulatory oversight, compliance, and risk management within the pharmaceutical supply chain.

This literature review aims to explore the existing studies related to text classification of inspection findings and related to the Malay language using machine learning. This study will examine the various approaches, tools, and best practices employed to leverage machine learning techniques in this specialised area of GDP inspection.

2.2 GOOD DISTRIBUTION PRACTICE

World Health Organization (WHO) defined Good Distribution Practice (GDP) as "a part of quality assurance that ensures the quality of a medical product is maintained by means of adequate control of the numerous activities that occur during the trade and distribution process, as well as providing a tool to secure the distribution system from falsified, unapproved, illegally imported, stolen, substandard, adulterated and or misbranded medical products" (Mccormick & Sanders 2022). Hence, WHO published a technical report series, TRS 1025 – Annex 7: Good Storage and Distribution Practices for Medical Products, in June 2020 with the intention to be applicable to all entities engaged in any aspect of storage and distribution of medical products. This encompasses activities ranging from the manufacturing premises to agents, individuals dispensing, or directly providing medical products to patients. It includes all participants in different stages of the medical product supply chain, including manufacturers, wholesalers, brokers, suppliers, distributors, logistics providers, traders, transport companies, forwarding agents, and their respective employees. The primary objective is to aid in meeting responsibilities across various supply chain stages and to prevent the entry of substandard and falsified products into the market.

The Pharmaceutical Inspection Co-Operation Scheme (PIC/S) published the PIC/S Guide to Good Distribution Practice For Medicinal Products in June 2014. It is based on the EU Guidelines on Good Distribution Practice (GDP) of Medicinal Products for Human Use (2013/C 343/01). It defined that "wholesale distribution does not depend on whether that distributor is established or operating in specific customs areas, such as free zones or warehouses. Distributors engaged in activities related to wholesale distribution, including importing, exporting, holding, or supplying, are subject to the same obligations regardless of their location". It was published to "ensure the maintaining of high standards of quality assurance and the integrity of the distribution processes of medicinal products, to promote uniformity in licensing of wholesaling of medicinal products and to further facilitate the removal of barriers to trade in medicinal products".

National Pharmaceutical Regulatory Agency (NPRA) under the Ministry of Health Malaysia defined GDP as "the measures that need to be considered in the storage, transportation and distribution of any registered product / notified cosmetics and its related material such that the nature and the quality of the intended use is preserved when it reaches the consumer" (NPRA 2020). It is a requirement for all license holders who are involved in the supply chain of any registered product / notified cosmetics to comply with the requirements stated under the current Good Distribution Practice Guideline when conducting their activities while ensuring the maintenance of high standards of quality assurance and integrity of the distribution processes. It is also regulated under several laws and regulations, which are Regulation 7 (1) (b) Control of Drug and Cosmetics Regulations 1984 and Section 12 (1) of the Sale of Drugs Act 1952. This GDP guideline consisted of 11 chapters and Annex 1 as shown in Table 2.1 for compliance requirements.

Table 2.1 GDP Guideline chapter description

Chapter	Description
Chapter 1	Quality System
Chapter 2	Personnel
Chapter 3	Premises and Equipment
Chapter 4	Stock Handling and Stock Control
Chapter 5	Transportation
Chapter 6	Products / Cosmetics Complaints
Chapter 7	Products / Cosmetics Recalls
Chapter 8	Substandard and Falsified Products / Cosmetics
Chapter 9	Outsourced Activities
Chapter 10	Self-Inspection
Chapter 11	Management of Documentation and Records
Annex 1	Management of Time and Temperature Sensitive Products (TTSP)

2.2.1 Inspection Findings Analysis Methods Used by Regulatory Authorities

A retrospective study was conducted on twenty-five Good Manufacturing Practice (GMP) inspection reports from March 2017 to December 2018 by Uche et al. (2021). This study done by Uche et al. (2021) focused on evaluating the proficiency of inspectors from the Drug Inspectorate in West Africa in the skill of inspection report

writing. The reports were categorised into several categories: Excellent Report, Good Report, Needs Improvement Report and Unacceptable Report. The analysis showed that 1 of 2 good reports had 51-75% of observed deficiencies correctly cited to the right GMP guideline, whereas another report had between 26-50% of observed deficiencies correctly cited according to GMP guideline or regulation. For the Needs Improvement Report categories, only 36.4% of the reports have more than 75% of observed deficiencies that are adequately referred to the correct regulatory citation. 36.4% of the reports had below 50% of the observed deficiencies adequately cited to the right regulations or guidelines. For unacceptable reports [11 of 25 (44%) total inspection reports], only 1 of them had between 26-50% of observed deficiencies properly cited to the right GMP guideline, whereas the rest of the unacceptable reports had 0-25% of observed deficiencies been cited correctly to the GMP guideline. This indicated that there is a knowledge gap in the classification of finding to the right GMP context for inspectors in the study. The author concluded that to enhance knowledge sharing and improve regulators' performance in drafting inspection reports, it has been suggested to provide training on various quality attributes. These include technical content related to the Quality Management System (QMS) and Site, utilising objective evidence, assigning risk levels to GMP violations, and referencing relevant laws, regulations, and guidelines to substantiate GMP observations. The goal is to bolster understanding and proficiency in these areas, ultimately contributing to more effective and comprehensive inspection reporting.

European Medicines Agency (EMEA) classifies GMP inspection deficiencies according to a list defined by the Medicines and Healthcare Products Regulatory Agency (MHRA) of 40 categories as shown in Table 2.2, which is not based on chapters, paragraphs and annexes of the EU GMP guide. Microsoft Access GMP Database is used as a data management tool for keeping all deficiencies data. However, from a statistical perspective, this system is impractical for use in analysis because the categories used in the system might refer to multiple different references in the GMP guide (EMEA 2007). The system also will not be able to assist inspector to correctly cite the deficiency according to the requirements listed in the guideline. Besides that, it requires competent / experience inspector for categorising the deficiency according to these 40 categories.

Table 2.2 List of categories of deficiencies used in EMEA GMP database

No	Category of GMP deficiency	No	Category of GMP deficiency
1	Analytical validation	21	Housekeeping - cleanliness, tidiness
2	Batch release procedures	22	In-process controls - control and monitoring of production operations
3	Calibration of measuring and test equipment	23	Intermediate and bulk product testing
4	Calibration of reference materials and reagents	24	Investigation of anomalies
5	Cleaning validation	25	Line clearance, segregation and potential for mix-up
6	Complaints and product recall	26	Personnel issues: Duties of key personnel
7	Computerised systems – documentation and control	27	Personnel issues: Hygiene/Clothing
8	Computerised systems - validation	28	Personnel issues: Training
9	Contamination, chemical/physical - potential for	29	Process validation
10	Contamination, microbiological - potential for	30	Production planning and scheduling
11	Design and maintenance of equipment	31	Regulatory issues: Non-compliance with manufacturing authorisation
12	Design and maintenance of premises	32	Regulatory issues: Non-compliance with marketing authorisation
13	Documentation - manufacturing	33	Regulatory issues: Unauthorised activities
14	Documentation - quality system elements/procedures	34	Sampling - procedures and facilities
15	Documentation - specification and testing	35	Self-inspection
16	Environmental control	36	Starting material and packaging component testing
17	Environmental monitoring	37	Status labelling - work in progress, facilities and equipment
18	Equipment qualification	38	Sterility Assurance
19	Finished product testing	39	Supplier and contractor audit and technical agreements
20	Handling and control of packaging components	40	Warehousing and distribution activities

This analytical framework employed within the EMEA serves as a pivotal tool for systematically monitoring consistency across diverse parameters, including inspectors and manufacturers engaged in various regions or activities. The utilisation of the deficiency database yields several advantages. Firstly, it facilitates the monitoring of variations among different groups of manufacturers, allowing for the identification and spotlighting of commonly encountered deficiencies within the industry. Secondly, the analysis aids industry stakeholders in gaining insights by comparing deficiencies on an industry-wide scale with those observed during internal audits and official inspections. Thirdly, the tool provides management within EU National Competent Authorities with a valuable metric to gauge the consistency of GMP inspection standards, indicating areas where additional training for inspectors and providing technical advice to the industry may prove beneficial. Furthermore, it can be utilised within an EU National Competent Authority to monitor the uniformity in deficiency reporting among inspectors. Additionally, the generated information is also helpful for regulatory authorities while revising the EU guidelines by focusing on the areas that need improvement. This analytical approach not only supports comparisons of industry-wide deficiencies with those identified during national inspections, fostering discussions for quality improvements but also identifies manufacturing practices of paramount concern to European Economic Area (EEA) competent authorities and manufacturers. The deficiency database emerges as a valuable instrument for EEA Inspectorates' management to assess the consistency of GMP inspection standards and determine areas necessitating additional training and technical advice for the industry. The insights garnered from this analysis may contribute to the contemplation of revisions to aspects of the EU guides to GMP, emphasising areas requiring heightened attention.

The United States Food and Drug Administration (USFDA) conducts inspections and assessments of regulated facilities to ascertain the compliance of a firm with relevant laws and regulations, including the Food, Drug, and Cosmetic Act and its related Acts. USFDA has been incorporating digital tools like Compliance Dashboards to bolster its regulatory oversight and analytical capabilities. These dashboards serve as powerful instruments for monitoring, analysing, and visualising compliance data and providing insights into the compliance status of the premise in a manner that is

accessible and user-friendly. The data dashboard's content is derived mainly from FDA compliance and enforcement data that has been authorised for public access. A closer look at the dataset supplied by the FDA, it was a compilation of inspection deficiencies from 2008 until the current date, with a total record count of 39,784 deficiencies for drug product inspection. These inspection deficiencies are categorised according to their Act/CFR Number based on the deficiencies written in the description by USFDA (USFDA 2023).

Table 2.3 shows the comparison of classification system used by different regulatory authorities.

Table 2.3 Comparison of classification system used by regulatory authorities

Aspect	EMEA	USFDA	NPRA
Classification System	40 categories defined by the MHRA not based on EU GMP guide chapters/paragraphs.	Act/CFR Number based on deficiencies written in the description.	Classification based on 11 chapters of GDP Guideline.
Database model	MS Access GMP Database.	Compliance Dashboard.	Not indicated.
Advantages	Systematically monitoring and managing deficiencies across diverse parameters	Classification based on Act/CFR Numbers, directly linking deficiencies to specific regulatory requirements.	Classification based on Chapter requirements in GDP guideline. Practical for statistical analysis and easily refers by auditee.
Limitation	Categories not directly linked to EU GMP guide thus requires experienced inspectors or expertise for correct categorisation. Impractical for statistical analysis due to multiple references to different sections of the GMP guide.	Classification is restricted to the specific regulations and may not fully capture contextual compliance issues.	Classification may be too broad to provide insight into specific deficiency.

A retrospective study was conducted by Stoimenova et al. (2019) to analyse the regulatory inspection findings of pharmaceutical wholesalers in Bulgaria in 2017, comparing the results with findings from other EU member-states. It provides valuable insights into the regulatory inspection findings of pharmaceutical wholesalers in Bulgaria and offers a comparative analysis with other EU member-states. It was

conducted by manually reviewing all the GDP inspection reports of all pharmaceutical wholesalers inspected during 2017. The non-conformities (NCs) identified in Bulgarian pharmaceutical wholesalers were compared to those found in other EU member-states in the study. The authors found that in Bulgaria, 17 NCs were documented during the inspections in 2017, with six deficiencies classified as major, and no critical deficiencies identified. It was significantly lower than those reported by inspectors in other EU member-states, such as the United Kingdom and Malta, where a higher percentage of wholesale dealers were assigned major findings. Additionally, the study highlighted that 56 major findings were recorded for 35 inspections in the Netherlands in 2016, indicating a higher frequency of major deficiencies than in Bulgaria. The authors emphasise that any departure from GDP compliance could risk the quality, effectiveness, and safety of medicines. Therefore, non-conformities (NCs) must be documented, investigated, and corrective actions implemented by the wholesalers. Based on the comparison, it was found that Bulgarian pharmaceutical wholesalers have a lower number of major deficiencies in comparison to other EU member-states. This indicates a high level of compliance with GDP requirements in Bulgaria, which is supported by the low occurrence of major deficiencies and the absence of critical deficiencies.

2.3 TEXT CLASSIFICATION

95% of data was unstructured or semi-structured (Gandomi & Haider 2015). It is necessary to organise and structure this unstructured textual data to harness this data for informed decision-making. Given the substantial volume of data, manual processing is laborious and time-consuming and might incur errors in data processing due to human error. Text classification, also known as document classification or text categorisation, is a supervised machine learning task that automatically categorises the natural language, be it in text, sentence, or document form, to one or more predefined categories or labels based on their content and semantics (Dogra et al. 2022; Hacohen-Kerner et al. 2020). Text classification is one of the essential components in various research domains, such as information extraction, text indexing, text mining, information retrieval, and word sense disambiguation (Hacohen-Kerner et al. 2020).

The goal of text classification is to enable the automated organisation, sorting, and categorisation of textual data into distinct classes or categories. Text classification was divided into two different types of text classification: namely topic-based classification, where a given document is categorised into its respective document category, and stylistic classification, where a document is classified according to writing style (Hacohen-Kerner et al. 2018).

An example of topic-based classification involves categorising news articles into segments like Business-Finance, Sports, Science-Technology, and Lifestyle-Leisure. On the other hand, stylistic classification pertains to classifying content according to various literary genres, such as action, fantasy, comedy, historical, crime, political, saga, and science fiction.

Both types of text classification tasks usually require different types of features for better performance in machine learning classifiers. The stylistic classification relies on linguistic attributes like quantitative features, part of speech (POS) tags, orthographic features, function words, and vocabulary richness features. In contrast, topic-based classification primarily utilises unigrams and/or n-grams (where n > 2) for its classification process (Hacohen-Kerner et al. 2020).

However, there are very limited machine learning based text classification studies related to audit finding classification (Corpuz 2021; Tarnate & Devaraj 2019). Most of the studies are associated with the classification of ISO 9001:2015 Quality Management System non-conformance findings.

The automated text classification methods can be divided into three groups namely data-driven method, rule-based method, and hybrid method. Data-driven methods such as machine learning algorithms acquire the ability to classify based on past data observations. When provided with labelled training data, it can discern the inherent connections between text segments and their corresponding labels. This method can unveil hidden patterns within the data, offering greater adaptability and applicability to diverse tasks. In contrast, rule-based methods classify text into various categories by using a set of predefined rules. For example, a document with fruit words

such as “apple”, “orange”, or “grape” will be labelled as “fruit”. This technique requires a thorough knowledge of the domain, and it will be a challenge to maintain the system. The hybrid method combines both the data-driven method and the rule-based method for making classification (Dogra et al. 2022).

Most of the machine learning methods follow the common 2-step method, where feature extraction is done from the text documents in the first step, and the features are fed to the classifier in the second step for classification purposes (Dogra et al. 2022). The common feature representation techniques are bag-of-words (BoW), which associates a text with a vector indicating the number of occurrences of each word in the training corpus, and term frequency-inverse document frequency (TF-IDF), whereas the common classifiers used are Naive Bayes (NB), K-nearest neighbor (KNN), SVM, decision trees and random forest.

Typically, the text classification process can be divided into 4 phases, which are feature extraction, dimension reductions, classifier selection and performance evaluation (Dogra et al. 2022; Kowsari et al. 2019). The initial pipeline takes a raw text dataset as input. Typically, text datasets consist of sequences of text organised into documents, denoted as D = {X1, X2..., Xn} where Xi represents a data point, such as a document or text segment, containing s number of sentences, and each sentence contains ws words with lw letters. Every data point is given with a class label from a set of k different discrete value indices (Kowsari et al. 2019).

2.3.1 Text Preprocessing

Before performing any step of data preprocessing, it is necessary to perform data exploration on the raw dataset and attempt to correct the errors. Then, the dataset will undergo pre-processing steps to transform the data into a suitable format for further processing with text mining methods (Misra & Yadav 2019). A relatively balanced distribution of dataset usually produce a better classification result (Sun et al. 2009). The inspection dataset tends to display an imbalanced class label. Predictive data mining algorithms are very sensitive to imbalanced dataset in which some chapters are less represented than others. Therefore, analysis of the dataset would yield less reliable results, and this can be rectified through proper Exploratory Data Analysis (EDA). EDA

is a technique used to explore datasets in order to extract useful, actionable information, identify relationships among the explanatory variables, detect errors, and preliminarily select appropriate machine learning models (Aldera et al. 2021). It uses descriptive statistics and visualisation tools to develop an understanding of the data (Pramanik et al. 2019).

Pre-processing is a crucial step for text classification. Prior to the text classification process, the raw data need to be pre-processed, and the selection of preprocessing methods can improve the accuracy results of text classification (Hacohen-Kerner et al. 2020). The major steps in data preprocessing include data 'cleaning' (such as correction of spelling errors, reduction of replicated characters, and disambiguation of ambiguous acronyms), data integration (multiple data sources may be combined into one), data reduction (obtaining the data with reduced presentations while producing the same results on analysis) and data transformation (converting the data into a suitable format for the mining algorithms) (Han J, 2012).

The data cleaning process is a crucial step because the presence of a high percentage of noise and unnecessary features in the training data set and/or the testing data set can have adverse effects on the performance of statistical and probabilistic learning algorithms, thus producing a less reliable data mining model (Kowsari et al. 2019). Besides that, Decision trees and distance-based algorithms (like the KNN algorithm) are known to be susceptible to noise (García S 2015). Preprocessing methods such as stop words removal, punctuation mark removal, special character removal, word stemming, and word lemmatisation may also be needed so that they can improve the quality of the dataset for the text classification model (Hacohen-Kerner et al. 2020).

Tokenisation is a critical step in tasks like text classification, sentiment analysis, machine translation, and information retrieval. It is also a pre-processing method that involves breaking down a piece of text, such as a sentence or a paragraph, into smaller units known as tokens. These tokens can be words, phrases, symbols or other meaningful elements (Kowsari et al. 2019).

Stop words are the most frequently used words. Examples of Malay stop words like “pun, sahaja, telah, ia, iaitu, ialah, ini, sama, yang, walau, walaupun” and so on. These stop words are worthless in text mining. However, using a stop word list doesn’t improve performance in most text classification applications, and it commonly makes use of the entire vocabulary for text processing (Daniel Jurafsky 2023; Mohammed & Omar 2020). A study was conducted by Hacohen-Kerner et al. (2018) to find out the performance of machine learning classification by removing different percentages of unigram (stop words) and claimed that it improved the accuracy of classifiers’ performance. However, the performance table shown in the study showed that accuracy is highest without any stop word removal (Hacohen-Kerner et al. 2018).

Text and documents often exhibit variations in capitalisation within sentences. This can pose challenges, especially when dealing with large documents during classification tasks. To ensure consistent capitalisation, it is common to convert all letters to lowercase. This unifies the feature space for words in the text and documents.

Noise removal is the process of refining words and removing special characters, numbers, and symbols (Alshalabi et al. 2017). Text and document datasets often include numerous unnecessary characters, such as punctuation and special symbols. It can be problematic for classification algorithms (Kowsari et al. 2019) because the presence of unnecessary punctuation and special characters may interfere with the algorithm's ability to classify the text accurately. Therefore, it necessitates the preprocessing steps to remove or handle them appropriately.

Spelling correction is an optional pre-processing step (Kowsari et al. 2019), aiding in the reduction of duplicate words. For the Indonesian language, the Finite State Automata (FSA) and Levenshtein Distance Method have been employed for spelling correction to address non-word errors. The FSA method proves effective in reducing the spelling correction processing time. Additionally, bigrams have shown higher correction hit rates compared to unigrams and trigrams (Christanti Mawardi et al. 2018).

Stemming and lemmatisation were also used as pre-processing steps. Lemmatisation converts words based on their root despite their surface differences. The words am, are, and is have the shared lemma be; the words dinner and dinners both have the lemma dinner (Daniel Jurafsky 2023). In comparison, stemming is the crude chopping of affixes on the word. For example, automates, automatic and automation all were reduced to automat.

2.3.2 Feature Extraction

Once the pre-processing step is done, the text will undergo the feature extraction phase. The feature extraction phase is an essential stage for text analysis, as creating a comprehensive model for all text data would be a challenging task. Therefore, the current approach in text analysis is to represent the text document by reducing its text structure complexity and simplifying the text documents (Ahmed et al. 2023). Vector space model (VSM) and probabilistic models like N-Gram are commonly used for feature extraction. There are several features for text presentation, such as Bag-of-Words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF), and Word Embeddings. It is a process of converting textual data into numerical representations that can be used as feature input for machine learning algorithms (Mohammed & Omar 2020).

BoW is text representation that describes the occurrence of words within a document. It is widely used in NLP tasks such as text classification, sentiment analysis, and document clustering, due to its simplicity and efficiency, where it uses simple words or phrases as features to represent text (Ahmed et al. 2023). It converts a piece of text (such as a sentence or a document) into a numerical vector by considering the frequency of words present in the text, regardless of their order (sequence). Therefore, it does not have the notion of syntactic/semantic similarity or, more formally, the distances between words.

Furthermore, BOW cannot process complex word meaning differences, such as synonyms and polysemy. For example, consider the words "cuci" and "bersih". These words are synonyms, meaning they have a similar meaning of “cleaning”. However, in a BoW model, they are treated as separate words, and their similarity is not captured.

Similarly, the word "rekod" can have multiple meanings, such as the action of writing a note or referring to a log book of record. In a BoW model, these different meanings are not distinguished, and the model cannot capture the nuances of language. This will result in high-dimensional feature space, hence an increase in sparseness in the text representation besides slang and misspelt words. Examples of features extracted through BoW are unigram and bigrams (Hacohen-Kerner et al. 2018).

Bigram features, unlike unigram, capture some contextual information by considering pairs of consecutive words. A bigram feature represents text as a sequence of word pairs, which may help in distinguishing between different meanings of words and identifying relationships between words hence able to show some word-to-word coherence (Daniel Jurafsky 2023). Therefore, the bigram feature can capture certain aspects of language semantics more effectively than unigram. For example, in a bigram feature, the word pair "audit dalaman" and "pemetaan suhu" would be treated as a distinct unit, allowing the model to capture some semantic differences between the words "audit" and "dalaman" as well as between "pemetaan" and "suhu".

Bigram features offer improvements over unigram by capturing some contextual information through word pairs, but they may still have limitations in fully addressing the complexities of language semantics. More advanced techniques, such as word embeddings and deep learning models, are often employed to further enhance the understanding of language semantics in natural language processing tasks. Although bigram will give a better improvement in sentiment classification because it can capture modified verbs and nouns, it is not a commonly used text classification task due to mixed performance in topical text classification (Wang & Manning 2012).

TF-IDF captures the importance of words to a document in a corpus. It gives scoring to the importance of a word in a document based on the lexical and morphological properties of the text. However, it does not capture position in text, semantics, and co-occurrences in different documents. In study by Mohammed and Omar (2020), the authors used Term Frequency-Inverse Document Frequency based on Part-Of-Speech (TFPOS-IDF) so that higher word weightage can be assigned according to part of speech such as verbs, nouns and so on.

Word embedding is a feature learning technique in which each word or phrase from the vocabulary is mapped to an N-dimension vector of real numbers. Unlike the vectors in other feature extraction techniques discussed earlier, embeddings are short, with a number of dimensions d ranging from 50-1000, and the vectors produced are dense, thus reducing sparsity. The key advantage of word embeddings is that they capture the syntactic or semantic relationships between words based on the distance of word vectors (Pintas et al. 2021). Popular word embedding algorithms like Word2Vec, GloVe (Global Vectors for Word Representation), and FastText use neural network architectures to learn word representations from vast amounts of text dataset.

Feature selection techniques were employed to identify the most discriminative terms (features) or keywords associated with each category from a large number of possibly noisy terms to reduce the dimensionality of the feature space in text classification, thereby improving the accuracy and efficiency of the classification process (Alshalabi et al. 2017; Wang et al. 2019).

Feature selection techniques can be categorised into two groups: information theory ranking methods, such as chi-square and mutual information, and information retrieval ranking methods, such as document frequency and odd ratio (Alshalabi et al. 2017). In text classification, feature selection involves selecting the most important features based on their scores obtained through various scoring techniques. These techniques can include info gain (IG), mutual information (MI), TF-IDF, principal component analysis (PCA), or statistical functions. Once the features have been scored, all other features with lower scores are removed.

A study was conducted Jaafar et al. (2016) to categorise Indonesian and Malay news documents into four categories, which are ‘economy’, ‘sport’, ‘entertainment’, and ‘technology’. Jaafar et al used a combination of feature selection techniques, including chi-square, information gain, and document frequency, to develop the category classification algorithm. These feature selection techniques were employed to identify the most relevant features or keywords associated with each category, thereby improving the accuracy and efficiency of the classification process. The k-NN classifier

is then used to classify the news document into one of the predefined categories based on the selected features.

To overcome the challenges of rapid data growth and high computational time, the author used the top-n feature selection method to improve the category classification model's performance. This method selects the top-n most relevant features for each category, reducing the number of features used in the classification process and improving the algorithm's efficiency. The integrated text classification algorithm is proven to produce a good result with high accuracy rates for category classification. The accuracy rates achieved in the experiments conducted using Indonesian and Malay online news corpora were up to 97.50% for category classification.

A study was conducted by Hacohen-Kerner et al. (2018) to determine the effects of removing a certain percentage of common unigrams (feature reduction) in topic-based classification. The author intentionally removes the common unigrams with high frequency from the training models in addition to the removal of stopwords. This is a departure from traditional approaches that often focus on using the most frequent words or expressions as strong indicators for classification.

In traditional methods, the focus is often on using the most frequent words or expressions as strong indicators for classification (Hacohen-Kerner et al. 2018). This means that commonly occurring words are given significant weight within the classification models and are typically considered crucial for determining the category or class to which a document belongs. However, the proposed method of intentionally removing these most frequent unigrams challenges this traditional approach. It suggests that, at times, these highly frequent unigrams may not be necessary for accurate classification and could even have a detrimental effect on the models' accuracy. By intentionally removing these common unigrams, the approach aims to uncover the value contributed by less popular unigrams, which may have been overshadowed by the dominance of the frequent ones.

Hacohen-Kerner et al. (2018) conducted experiments to test the effectiveness of unigram unmasking in the context of content classification. Their experiments found that intentionally removing the most commonly occurring words (unigrams) from the training models often improved the accuracy of classifying textual content. However, this improvement only lasted until a certain percentage of common unigrams were removed. Beyond that point, eliminating additional terms negatively impacted the accuracy of the models.

The authors' method involves intentionally removing a percentage of common unigrams from the training models, a process they refer to as "unigram unmasking." They conducted experiments using a dataset of texts from five topic categories and classified them using the 5000 most frequent unigrams. The intentional removal of up to nearly 24% of the most frequent unigrams from the training set often had no reduction in the models' classification accuracy based on the types of classifiers used. In contrast, the authors found that the Naïve Bayes' accuracy after removing the first 200 most common unigrams (4%) was 90.12%, as opposed to an accuracy of 89.12% with no unigrams removal. In addition to this finding, the study also found that the Naïve Bayes classifier is the most stable, with an accuracy of 89.89% still being achieved after 80% of its unigrams were removed.

Textual documents usually contain many irrelevant terms that lead to excessive computational complexity and poor text classification performance (Wang et al. 2019). However, in text categorisation, there are only very few irrelevant features. (Joachims 1998). Feature selection attempts to determine these irrelevant features, but it is noted that even features ranked lowest still contain considerable information and are somewhat relevant. This suggests that a good classifier should combine many features and that vigorous feature selection may result in a loss of important information (Joachims 1998). Hence, although feature selection techniques can be used for identifying and picking important features, it is an optional step for text processing.

2.3.3 Text Classification Model

There are two types of text, which are long text, such as paragraphs, and short text, such as news titles. A paragraph is considered a long text as it contains several sentences consisting of topic sentences, support sentences and a conclusion (Adhi et al. 2019).

Among the machine learning algorithms, Naïve Bayes (NB), Support Vector Machine (SVM) and K-Nearest Neighbor (k-NN) algorithms are the most commonly used classifiers for text classification (Palanivinayagam et al. 2023; Rostam & Malim 2021) and most of the studies are related to the classification of newsgroup (Palanivinayagam et al. 2023).

SVM is a supervised machine learning algorithm that can be used for classification tasks. SVMs are based on the Structural Risk Minimization principle from computational learning theory (Joachims 1998). It finds a hyperplane with the largest margin (distance between the hyperplane and the closest data points from each class), which maximally separates the different classes in the training data. This remarkable property of SVM enables its ability to learn can be independent of the dimensionality of the feature space as it measures the complexity of hypotheses based on the margin with which they separate the data but is not dependent on the number of features (Joachims 1998). Besides that, document vectors in text categorisation are sparse, containing only a few entries that are not zero. SVMs are well suited for problems with dense concepts and sparse instances, providing theoretical evidence that they should perform well for text categorisation. With the ability to generalise in high-dimensional feature spaces, SVM also eliminates the need for feature selection.

It is much less prone to overfitting of dataset (Rostam & Malim 2021). It thus can perform well, especially in handling datasets with interconnected elements and multi-label effects (Rostam & Malim 2021). Additionally, term weighting has demonstrated better performance than ensemble methods in addressing interconnected elements in the data (Rostam & Malim 2021). However, classification accuracy depends on the correct selection and optimisation of the kernel (Palanivinayagam et al. 2023). It also provides a compact description of the learned model and can be used for numeric prediction as well as classification.

NB classifier is a probabilistic classifier based on Bayes theorem. It can predict class membership probabilities, such as the probability that a given tuple belongs to a particular class. NB classifier assumes that the effect of an attribute value on a given class is independent of the values of the other attributes. This assumption is called class-conditional independence. Multinomial NB is widely used for text classification (Palanivinayagam et al. 2023) and NB works very well on short documents (Wang & Manning 2012).

kNN's classification is a non-parametric and instance-based algorithm. It classifies data points based on the majority class of their k-nearest neighbours in the feature space. The choice of k (number of neighbours) is a crucial parameter that influences the model's performance and is preferably an odd number to avoid race conditions (Palanivinayagam et al. 2023). However, it is difficult to determine the exact k-value for study as it does not have a specific rule except pre-determined experimentally (Jaafar et al. 2016).

LR is a linear model used for binary classification and also can be extended for multiclass classification. Both the logistic function and sigmoid function can be used for binary values. It applies the logistic function to the linear combination of input features, mapping the output to the range [0, 1]. Hence, it models the probability that a given input belongs to a particular class. There are three types of logistic regression models, which are defined based on categorical output (Hassan et al. 2022).

Binary logistic regression: Only two types of values are possible for the dependent variable. It has only two possible outcomes (e.g. 0 or 1, win or lose). Examples of its use include email spam detection or whether a customer will churn or not.

Multinomial logistic regression: the dependent variable has three or more possible outcomes that have no specified order (i.e., types have no quantitative significance), such as blood type of "type A" versus "type O" versus "type B" versus "type AB" (Hassan et al. 2022).

Ordinal logistic regression: This model is used when the response variable exhibits three or more potential outcomes, and in this scenario, these values possess a distinct and defined order. Examples of ordinal responses include assessment scores that can be categorised as ‘very good’, ‘good’, ‘poor, and ‘very poor’. These categories also can be given a rating scale from 1 to 5.

2.4 RELATED WORK

Due to a lack of resources for managing Malay text classification, very limited studies have been carried out on Malay text classification. In the Malay language context, there are text classification studies on the criminal domain, Malay news categorisation, sentiment analysis, and Malay tweets classification. However, none was conducted on the Malay GDP inspection findings classification.

Jaafar et al. (2016) used the k-NN algorithm to classify Malay and Indonesian news documents into four categories. The authors modified the value of k for the k-NN algorithm to get the best performance of category classification. Once the best value is obtained, the author performed top-n feature selection prior to classification. The authors found that accuracy increases when the value of k increases until a certain threshold, and the accuracy will decrease if the value of k exceeds the threshold. The study concurred that the classifier would produce a bad performance due to a smaller value of k, which will obtain less information from the training, whereas exceeding the threshold (too many neighbours to be used) will lead to the occurrence of noises to the information obtained from the dataset. The top-n feature selection method was also performed to reduce the system performance speed and in the hope of better accuracy. The n value is the number of words in the news documents, which is used as the representative keywords for the documents, and the study considers the highest-weight word in the document as a keyword. The study found that reducing or increasing the 'n' value from 60% decreases classifier accuracy. This phenomenon occurs because selecting fewer than 60% of words as keywords results in poorly described documents, whereas choosing more than 60% of keywords introduces additional noise. This study reported a high accuracy rate of 97.50% for the news classification. However, the author

did not evaluate the machine learning model's performance through other standard measures such as precision and F1 score.

Another study by Alshalabi et al. (2017) compared the performance of ensemble and base classifiers in Malay text. The study compared the performance of three machine learning models, namely NB, k-NN and N-gram, with two feature selection methods, the Gini Index and Chi-square, which were applied to reduce the feature space dimension on Malay text classification. Besides that, a combination of two classifiers through voting combination and stacking combination has also been evaluated. The voting combination achieved its highest performance, reaching 95.84%, when 500 features were selected using the GI method. Conversely, its lowest performance, at 92.14%, was observed when 100 features were selected using the Chi-square method. The stacking combination achieved its best performance, at 94.39%, when 300 features were selected using the GI method. In contrast, its lowest performance, at 91.23%, was observed when 400 features were selected using the Chi-square method. Based on the findings, the GI method is more effective in achieving higher performance when conducting feature selection operations. Moreover, the results indicate that while the stacking combination algorithm outperforms individual classifiers, the voting combination yields better outcomes than the stacking combination.

The research found that ensemble techniques outperformed base classifiers alone. Specifically, the meta-classifier ensemble framework, which combines the results of multiple base classifiers using a meta-classifier, performed better than the best individual classifiers on the tested datasets. This suggests that combining the strengths of multiple classifiers can improve classification performance, as the strengths of others can compensate for the weaknesses of individual classifiers. Therefore, ensemble techniques can be a valuable approach for improving the accuracy and efficiency of text classification in the Malay language. The study concluded that the NB classifier achieved the best performance in terms of macro-F1, but the voting combination method produced better performance than NB.

A comparative study of K-Nearest Neighbour and Naïve Bayes performances on Malay text classification was conducted by Nazratul Naziah Mohd et al. (2021), aimed to evaluate the effectiveness of these classification methods in handling Malay text data. The study utilised a dataset of housebreaking crime records from 2010 to 2013, obtained from a closed domain dataset collected from the Royal Police Department of Malaysia. The corpus contains 100,383 Malay crime reports, and 1000 crime reports were randomly chosen and screened according to five distinct modus operandi classes: cara (method), peranan (role), keganjilan (oddity), senjata (weapon), and tempat (location).

The authors found that representing the text data using 4-grams resulted in higher classification accuracy compared to lower-order n-grams such as 1-gram, 2-grams and 3-grams. This suggests that the use of 4-grams contributed to improved classification performance for the specific task of classifying Malay housebreaking crime reports. The text further underwent tf-idf, and the study reported a high accuracy rate of 97.86%, precision rate of 98.03% and recall rate of 97.86% for Naïve Bayes, indicating its effectiveness in accurately predicting the class of modus operandi for housebreaking crime documents. In contrast, the K-Nearest Neighbour algorithm achieved a lower accuracy rate of 88.43%. Additionally, the study highlighted the timely execution of the Naïve Bayes algorithm, taking only 9 seconds to complete the classification task, compared to 48 seconds for the K-Nearest Neighbour algorithm. These findings demonstrate the superior performance of the Naïve Bayes algorithm in terms of accuracy and execution time for classifying crime reports, particularly in the context of Malay text data.

Al-Saffar et al. (2018) conducted a study on Malay sentiment analysis classification. They used various classification methods, including Naïve Bayes, SVM, Deep Belief Network, and a combination of these methods. The researchers selected four subsets of features, which included the presence and frequency of sentiment words, sentence level, sentiment word polarity features, and subjective words conditional probability features. It concluded that the combination method is able to achieve an F-measure of 94.48% for sentiment analysis.

An experiment study by Tiun (2017) was conducted on Malay short text classification using three different types of classifiers, namely KNN, SVM and NB. These classifiers were employed to examine the usability of various features such as bag-of-words (BOW), TF-IDF, TF-IDF's variants, including smoothed TF-IDF, and ITC (sublinear TF-IDF). It aims to identify the best model for classifying binary classification in Malay short text. The evaluation metrics used to assess the performance of the classifiers were Precision, Recall, and F1-Score. The results indicated that the SVM classifier achieved the highest Precision, Recall, and F1 score when using ITC (sublinear TF-IDF) as the feature, all at 95%. This means that the model was able to identify 95% of the relevant instances correctly and had a low rate of false positives and false negatives. Therefore, the study recommended using SVM with ITC as the preferred Malay short text classification model. However, the author also stated that this study is based on binary class classification, and the number of class labels to be used plays a significant role in deciding the model to be used.

A study was conducted by Corpuz (2021) to classify ISO 9001:2015 Quality Management System audit findings using SVM and long short-term memory neural network. The author found that, in practice, there is an issue of ambiguity in the audit findings, which were interpreted with multiple clauses of the ISO standard. The incorrect interpretation can lead to wrong root cause analysis and, thus, ineffective corrective actions. Besides that, the wrong categorisation might cause misunderstandings between auditors and auditees. The author explores the causal relationship between dataset, holdout, and NumWords as independent variables with training accuracy and timeliness as dependent variables. This study reveals that the dataset and holdout percentage were the predictors of accuracy, with the dataset positively influencing both accuracy and timeliness. In contrast, the holdout percentage had a negative impact on accuracy. Whereas dataset and NumWords were the predictors of timeliness, which negatively affected training time.

The SVM and LSTM classification models were trained by increasing the number of datasets and holdout percentage, which had an effect on the resulting NumWords or vocabulary of words learned by the models as well as on the training accuracy and timeliness performance. It also revealed that LSTM generally exhibited

superior performance in terms of training accuracy on larger datasets compared to SVM (97.54% [best rating] for LSTM versus 94.74% [best rating] for SVM). The author further employed a statistical t-test to compare the effects of these parameters with the classification models and found that the difference between them was not statistically significant. However, the study did reveal that SVM performed significantly faster than LSTM in any dataset size. Overall, the study suggests that both SVM and LSTM are effective in classifying ISO audit findings and standard requirements.

Tarnate and Devaraj (2019) conducted a study to predict ISO 9001:2015 Quality Management System audit reports according to its major clauses using several RNN models. The authors developed a deep neural networks model with a combined word representation model (word encoding plus an embedding dimension layer) to classify audit reports according to the major clauses of the ISO 9001:2015 QMS Requirement. They used the "doc2sequence" n-gram model to convert the text data into vectors. It reduced the dimensionality of the data by assigning a fixed number or length of attributes through word embedding. The authors divided the datasets into three states: 70% for the training of the model, 15% for the validation of the model, and another 15% for the testing of RNN(s) of the model. The authors built a two deep-layered LSTM and Bi-LSTM neural networks with a total number of 225 hidden units and compared the performance of those models to the traditional LSTM and Bi-LSTM models. They achieved an average Classification Accuracy of 91.10% and a Cross-Entropy Loss of 1.59%. Therefore, the authors concluded that the Deep-Bidirectional LSTM outperformed the other three RNN models based on the average classification accuracy of 91.10%.

During the validation stage, the traditional LSTM model performed better than the two-layered Deep-LSTM model, with an average classification accuracy of 90.2%. According to the study results, incorporating an extra neural network layer into deep neural networks may not yield a considerable accuracy enhancement due to the deep neural networks' vanishing gradient descent issue. This suggests that the deeper architecture did not effectively capture more complex patterns in the data, leading to limited performance gains.

The study conducted by Tarnate et al. (2020) aimed to classify ISO 9001:2015 Quality Management System audit reports, specifically focusing on major clauses using various types of recurrent neural networks (RNN). The researchers compared the impact of different methods, including word encoding, word embedding, and a combination of both, on the learning performance of the classification model. The results showed that the bi-directional long short-term memory (Bi-LSTM) outperformed LSTM, especially when the combined word encoding and word embedding optimisation technique was utilised. Overall, all the LSTM methods can achieve above 87% accuracy.

Table 2.4 shows the summary of related works on Malay text classification. From the summary table, it clearly shown that the common features used in Malay text classification are N-grams and TF-IDF and these features also shown to be able to give high accuracy rate for classifier. Therefore, N-grams such as unigram (Bag of word), bigrams and TF-IDF will be chosen for multi-class classification of GDP audit finding in Malay language. 3-grams and 4-grams will be excluded in the study as the audit findings are short texts thus would not give better performance than unigram and bigrams.

The common classifier used in the related works are NB, SVM, k-NN and ANN. NB, SVM and k-NN are selected for performance comparison in this study. Besides that, multinomial logistic regression classifier is also included in this study as this classifier can be used for multi-class classification using softmax function (Daniel Jurafsky 2023). Although deep learning network was commonly used in related works, it was excluded in this study due to the size of dataset used in this study is relatively small which is unsuitable for deep learning model.

Table 2.4 Summary of related works on text classification.

Study	Dataset	Classifier	Features	Performance	Key Findings
A category classification algorithm for Indonesian and Malay news documents. (Jaafar et al. 2016)	Malay and Indonesian news documents	k-NN	TF-IDF with Top-n feature selection	Accuracy: 97.50%	k-NN performance improves with optimal k value; top-n feature selection effective; did not evaluate precision or F1 score
A comparative study of the ensemble and base classifiers performance in Malay text categorization. (Alshalabi et al. 2017)	Malay news document	NB, k-NN, Ensemble (Voting, Stacking)	N-gram, Feature selection (Gini Index, Chi-square)	Accuracy: Voting: 95.84%, Stacking: 94.39%	GI method more effective; ensemble methods outperformed individual classifiers; voting combination superior
Comparative Study of K-Nearest Neighbour and Naïve Bayes Performances on Malay Text Classification. (Nazratul Naziah Mohd et al. 2021)	Malay crime reports	Naïve Bayes, k-NN	N-grams (4-grams), TF-IDF	NB: Accuracy: 97.86%, Precision: 98.03%, Recall: 97.86%	NB outperformed k-NN; 4-grams feature improved classification accuracy; NB faster execution time

to be continued…

…continuation

Malay sentiment analysis based on combined classification approaches and Senti-lexicon algorithm. (Al-Saffar et al. 2018)	Malay Review Corpus	NB, SVM, Deep Belief Network, Ensemble	Malay sentiment lexicon	F-measure: 94.48%	Ensemble methods achieved highest F-measure for sentiment analysis
Experiments on Malay short text classification. (Tiun 2017)	Malay tweets	KNN, SVM, NB	N-grams (BOW), TF-IDF, smoothed TF-IDF, ITC (sublinear TF-IDF)	SVM (with ITC): Precision, Recall, F1: 95%	SVM with ITC performed best; binary classification focus
ISO 9001:2015 Quality Management System Requirements and Audit Findings Classification Using Suppor Vector Machine and Long Short-Term Memory Neural Network: An Optimization Method. (Corpuz 2021)	ISO 9001:2015 QMS audit reports	SVM, LSTM	BOW for SVM, word encoding for LSTM	LSTM: Accuracy: 97.54%, SVM: Accuracy: 94.74%	LSTM performed better on larger datasets; SVM faster execution time

to be continued…

…continuation

Prediction of ISO 9001:2015 Audit Reports According to its Major Clauses using Recurrent Neural Networks. (Tarnate and Devaraj 2019)	ISO 9001:2015 QMS audit reports	RNNs (LSTM, Bi-LSTM, Deep-LSTM, Deep-Bi-LSTM)	Word encoding, word embedding	Deep-Bi-LSTM: Accuracy: 91.10%, Cross-Entropy Loss: 1.59%	Deep-Bi-LSTM outperformed other RNN models
Overcoming the vanishing gradient problem of recurrent neural networks in the ISO 9001 quality management audit reports classification. (Tarnate et al. 2020)	ISO 9001:2015 QMS audit reports	RNNs (Bi-LSTM, LSTM	Word encoding, word embedding, combined optimisation technique	Bi-LSTM: Accuracy: >87%	Bi-LSTM with combined optimisation performed best

2.5 SUMMARY

This chapter provides a comprehensive review of the existing literature on text classification techniques, focusing on feature extraction methods and classifier performance for Malay language and multi-class classification domain. The methodologies used in previous studies are compared, including Naive Bayes, Logistic Regression, Support Vector Machines (SVM), and K-Nearest Neighbors (K-NN). The performances of each approach are highlighted. The review identifies several gaps in the literature, including the need for more comprehensive comparisons of feature extraction methods and classifiers across diverse datasets. This literature review establishes the foundation for the current study, which aims to systematically compare the performance of various feature extraction methods and classifiers. In conclusion, this chapter has reviewed the existing literature on text classification, identified key methodologies, and highlighted gaps that the current study aims to fill. The next chapter will detail the research methodology used to conduct this study.

CHAPTER III

METHODOLOGY

3.1 INTRODUCTION

This chapter functions as a comprehensive guide for executing the research plan, delineating the systematic procedures and methodologies applied to fulfil the research objectives. Section 3.2 delineates the research design, wherein the successful implementation of this design is pivotal to attaining the outlined research goals. The subsequent Section 3.3 provides an overview of the dataset utilised in the experiment, offering detailed insights into its characteristics. Section 3.4 delves into the preprocessing phase, elucidating specific tasks such as lowercase conversion and tokenisation. Section 3.5 illuminates the representation of terms through n-gram analysis, bag of words, and TF-IDF. The subsequent Section 3.6 expounds upon the machine learning algorithms incorporated in the study. Finally, Section 3.8 outlines the evaluation methodology employed, delineating the criteria and processes used to assess the proposed method's efficacy.

3.2 RESEARCH DESIGN

Figure 3.1 below shows the research methodology framework of text classification for this study. It consists of five phases: data collection, text pre-processing, feature extraction, machine learning classification and performance evaluation.

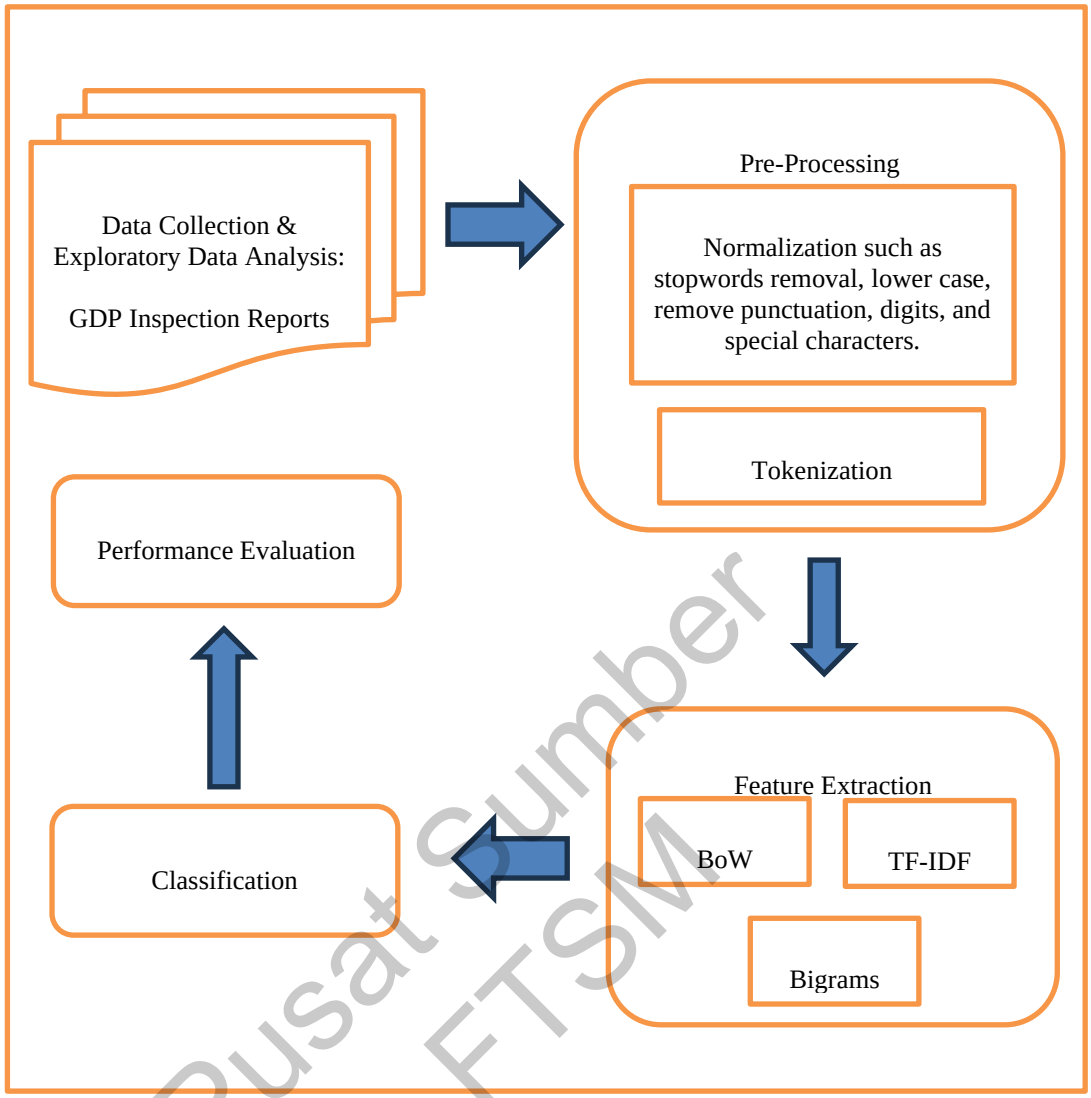


Figure 3.1 Research framework

3.3 DATASET COLLECTION

The initial phase involves the acquisition of relevant data from the National Pharmaceutical Regulatory Agency, Section GDP, which oversees the pharmaceutical good distribution practice in Malaysia. With retrieval of Good Distribution Practice (GDP) inspection reports for the year 2022, the non-conformance findings written in the Malay language in the comment section were extracted and manually labelled according to the chapter in GDP guidelines to form the dataset for analysis.

There was a total of 104 inspection reports (excluding cold chain inspection reports), with 1700 non-conformance comments manually extracted for the dataset. After the extraction, the non-conformance findings related to the Annex in GDP

guideline were filtered out and excluded from the text categorisation. The comment can be a paragraph, sentence, or subparagraph. Those comments were extracted and keyed into Microsoft Excel. Each comment was then labelled based on the chapters outlined in the GDP guideline. This results in a dataset with two attributes and one class attribute with a total of 1383 instances. The class label indicates the chapter on GDP guidelines. Figure 3.2 shows the sample of the raw dataset extracted from the GDP inspection reports.

1	Report	sentence	chapter
2	22.001	*Pihak syarikat telah mengambil tindakan pembetulan terhadap penemuan-penemuan lepas. Walau bagaimanapun, terdapat penemuan yang masih berulang (rujuk Bab 1: 1.4; Bab 2: 2.3; Bab 9: 9.2, Bab 10: 10.1 & 10.3).*	1
3	22.001	Rekod penilaian penerima kontrak tidak dapat dikemukakan semasa pemeriksaan dijalankan (rujuk Bab 9: 9.2). (penemuan berulang	1
4	22.001	Prosedur / program latihan masih belum disediakan oleh pihak syarikat. (penemuan berulang)	2
5	22.001	Rekod penilaian kompetensi ke atas penerima kontrak iaitu syarikat Agility Logistics Sdn. Bhd. tidak dapat dikemukakan semasa pemeriksaan dijalankan. (penemuan berulang)	9
6	22.001	Prosedur pemeriksaan dalaman masih belum diwujudkan. (penemuan berulang)	10
7	22.001	Pemeriksaan dalaman yang dijalankan masih terhad ke atas semakan stok sahaja namun tiada sebarang laporan dikeluarkan. (penemuan berulang)	10
8	22.002	Pihak syarikat belum mewujudkan keperluan / dokumen yang berkaitan dengan pelupusan produk.	4
9	22.002	Pihak syarikat belum menyediakan keperluan / dokumen yang berkaitan dengan aktiviti pengendalian produk *substandard* / tiruan.	8
10	22.003	Nombor pendaftaran produk tidak dinyatakan dalam rekod borong.	11
11	22.003	Nombor pendaftaran produk tidak dinyatakan dalam rekod import.	11
12	22.004	Pihak syarikat belum mewujudkan sistem untuk mengawal dan menilai penerima kontrak (rujuk Bab 9: 9.2).	1
13	22.004	Kompetensi penerima kontrak seperti syarikat Viva Pharmaceutical Inc. dan syarikat Seutic Pack Sdn. Bhd. b	9
14	22.004	Pihak syarikat belum pernah menjalankan pemeriksaan dalaman.	10
15	22.004	Nombor pendaftaran produk tidak dinyatakan dalam rekod borong.	11

Figure 3.2 Raw dataset of GDP inspection findings

3.3.1 Exploratory Data Analysis

Exploratory Data Analysis (EDA) is conducted to gain a comprehensive understanding of the dataset before delving into text mining. Through EDA, patterns, frequencies, and distributions of non-compliance issues across different chapters of GDP will be unveiled. Table 3.1 shows a list of attributes of its original data type with a description of the dataset, and Figure 3.3 shows the distribution of sentences based on the chapter on GDP guidelines for the raw dataset.

Table 3.1 Attributes and its original data type with description

Attributes	Original Data Type	Description
report number	Nominal	GDP inspection report number.
sentence	String	Non-conformance finding in the comment section of the GDP report.
chapter	Nominal	Chapters based on GDP guideline.

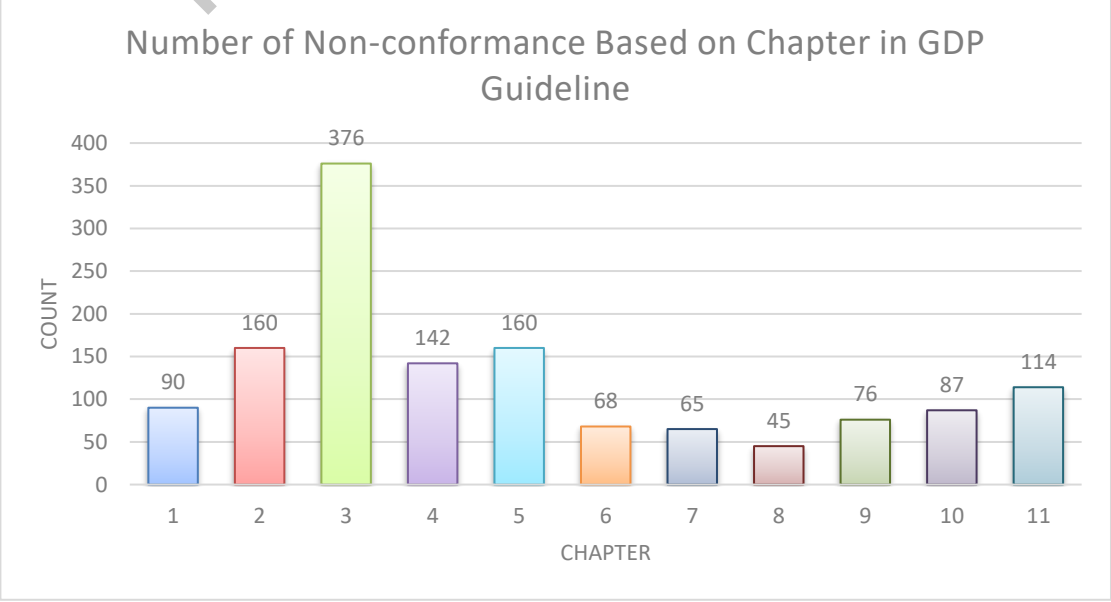


Figure 3.3 Number of non-conformances based on chapter in GDP guideline

Prior to the pre-processing phase, any duplicate sentences identified through EDA will be removed to reduce the redundancy of the dataset. After the removal of duplicate sentences, the total number of non-conformances is 1258.

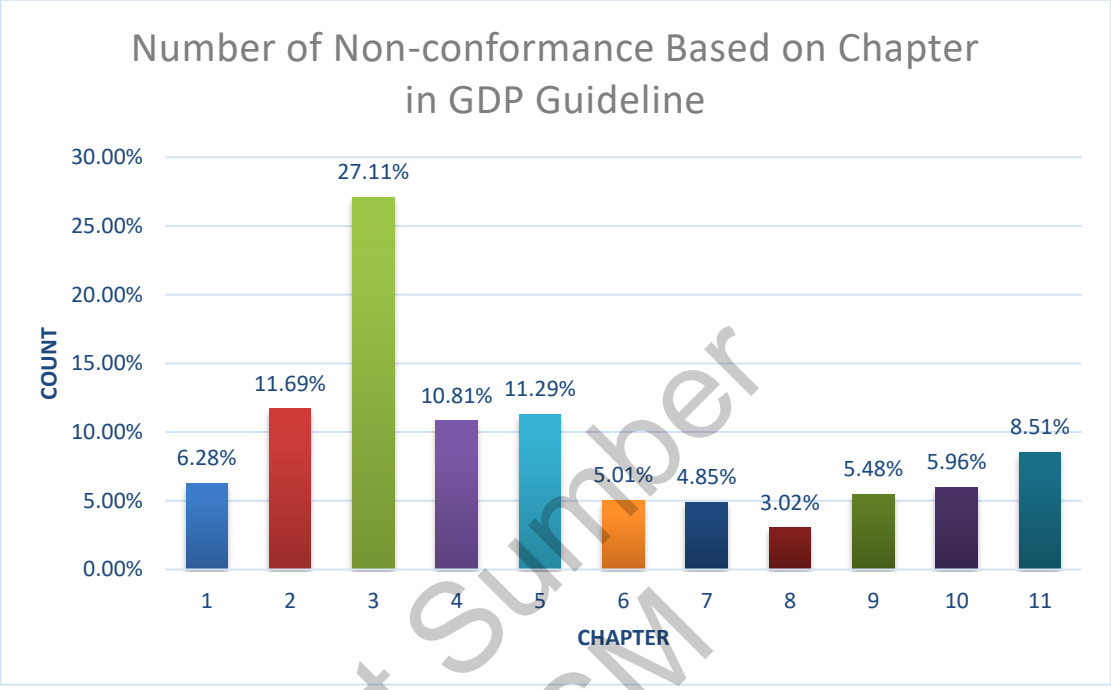


Figure 3.4 Number of non-conformances after removing duplicates

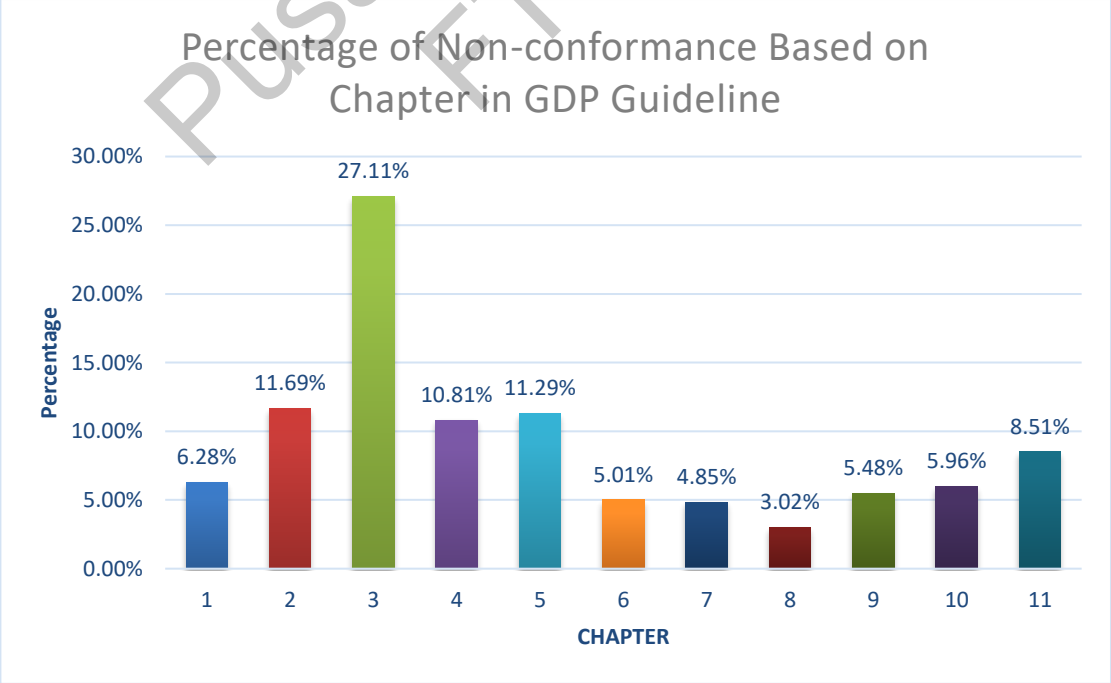


Figure 3.5 Distribution of non-conformances among chapters in GDP guideline

As can be seen in Figure 3.4 and Figure 3.5, the distribution of the sentences among chapters is somewhat skewed, leading to an imbalanced dataset. Therefore, stratified random sampling methods will be used to ensure that all chapters of the sentences are represented in roughly the same proportion of samples in the training set and test set as the original dataset, thus reducing the bias. This is crucial for an imbalanced dataset as shown in Figure 3.4, because it prevents the model from being biased towards the majority class and neglecting the minority classes.

The sentences will undergo the pre-processing phase, which is pivotal in refining and structuring the raw textual data extracted from the GDP inspection reports. The primary objectives of the pre-processing stage are to standardise the text, remove noise, and ensure a consistent and meaningful representation for effective text mining. This critical phase involves a series of tasks aimed at enhancing the quality of the dataset and preparing it for the text classification model.

3.3.2 Lowercase, Remove Punctuation, Digits and Special Characters

The first task in the pre-processing stage involves converting all text to lowercase. It standardised the representation of words, and this is important to ensure consistency and prevent the model from treating identical words with different cases as separate entities. This is crucial in reducing the complexity of data and significantly improving the consistency of expected output (Nazratul Naziah Mohd et al. 2021). Lowercasing is particularly useful in subsequent analyses as it simplifies the process and helps the model accurately capture patterns and relationships within the text.

In order to improve the accuracy of text mining processes and make the dataset cleaner and more meaningful, the next step involves removing punctuation, digits, and special characters from the text. This pre-processing stage helps to focus on the linguistic content and reduces unnecessary noise. It is an important step that eliminates non-alphabetic elements.

3.3.3 Tokenisation

Tokenisation is a crucial process in natural language processing, where the text is divided into a sequence of sentences, and then each sentence is converted into a sequence of tokens, which are individual words. The purpose of this process is to identify and isolate words from the text corpus, creating a structured representation of the text that can be used for feature extraction methods such as unigrams and bigrams. Tokenisation is essential in transforming continuous text into a format that can be effectively utilised by machine learning algorithms.

3.3.4 Stopword Removal

Common and non-informative words, known as stopwords, are often prevalent in text but carry little semantic meaning. Stopwordremoval is employed to filter out these redundant terms, further refining the dataset. By eliminating stopwords, the pre-processing stage focuses on retaining words that are more indicative of the content and context of the inspection findings. Stopword removal will be done by using the Malaya toolkit.

3.4 WORD CLOUD ANALYSIS

Word cloud analysis is performed thus can provide insight to the word usage patterns for each chapter. As shown in Figure 3.6, the significant word usage patterns for each chapter can be identified. Besides that, the analysis also shown that 'syarikat' is a common term used in inspection finding across all chapters though it does not provide any indicative information. It could be due to GDP findings address to the particular company.

Figure 3.6 Word cloud based on GDP guideline chapters

By removing the common word ‘syarikat’ through the code in Figure 3.7, word cloud analysis will show a more specific frequent words for each chapter as shown in Figure 3.8.

```
[26] # remove word 'syarikat'

     data["sentence2"] = data["sentence"].apply(lambda x: ' '.join([word for word in x.split() if word.lower() != "syarikat"]))
```

```
[27] class_texts = {}
     for chapter in data['chapter'].unique():
         class_texts[chapter] = " ".join(data[data['chapter'] == chapter]['sentence2'])
```

Figure 3.7 Code for removing 'syarikat' word

Figure 3.8 Word cloud after removal of 'syarikat' word

Table 3.2 listed the frequent words used in each chapter of GDP guideline. The frequent words in Chapter 1 - Quality system consists of frequent words from different chapters. This shows that the quality system of a company is evaluated as the whole distribution activity of company. Besides that, it also revealed the similarity of frequent words in chapter 4 and chapter 11. Consequently, the accuracy of classifier in classifying inspection findings for chapter 1, 4 and 11 could be potentially affected by it.

Table 3.2 Frequent words for each chapter

Chapter	Frequent Words
Chapter 1	Produk, pemeriksaan, menilai penerima, prosedur, penerima kontrak
Chapter 2	Personel, latihan, dokumen, rekod latihan, deskripsi tugas
Chapter 3	Kelembapan relatif, pemantauan suhu, pemetaan suhu, kawasan penstoran, kajian pemetaan
Chapter 4	Dokumen, prosedur, semakan, pelupusan produk
Chapter 5	Kenderaaan, insiden penyimpangan, pengendalian penyiasatan
Chapter 6	Aduan produk, aduan diterima, pengendalian aduan
Chapter 7	Panggil, produk, recall
Chapter 8	Substandard, tiruan
Chapter 9	Kompetensi, penerima, kontrak
Chapter 10	Pemeriksaan dalaman, laporan pemeriksaan
Chapter 11	Dokumen, rekod, prosedur

3.5 FEATURE EXTRACTION

Feature extraction is a crucial step in the text mining process, where the pre-processed textual data is transformed into a numerical format (vectorisation), creating a structured representation that machine learning models can effectively utilise (Dogra et al. 2022). In this research, various feature extraction techniques are employed to capture different aspects of the information present in the Malay language Good Distribution Practice (GDP) inspection findings.

3.5.1 Bag of Words (BoW)

BoW provides a simple yet effective way to represent the inspection findings, capturing the frequency information of words across the entire dataset. BoW is text representation

that describes the occurrence of words within a document. It converts a piece of text (such as a sentence or a document) into a numerical vector by considering the frequency of words present in the text, regardless of their order (sequence).

BoW method represents the text as a matrix, where each row corresponds to a document, and each column corresponds to a unique word in the entire corpus. The matrix entries contain the frequency of each word in the respective documents. Below is the example of BoW for inspection finding sentences corpus shown in Table 3.3.

Table 3.3 Example of sentences in the corpus

Documents	Sentences
D1	*belum menjalankan kajian pemetaan suhu*
D2	*laporan pemeriksaan dalaman belum diwujudkan*
D3	*borang aduan produk belum diwujudkan*

Table 3.4 Vocabulary of the corpus

Vocabulary (unique words in the corpus)
['belum', 'menjalankan', 'kajian', 'pemetaan', 'suhu', 'laporan', 'pemeriksaan', 'dalaman', 'diwujudkan', 'borang', 'aduan', 'produk']

Table **3.4** shows that there are 12 unique words in the vocabulary (assuming the whole corpus consists of D1, D2, and D3 documents only). Thus, we can use 12 12-dimension vector to represent each sentence, and the number 0-n indicates the frequency of the word that appeared in the particular sentence, as shown in Table 3.5.

Table 3.5 Bag of Word representation

Documents	Vector
D1	*[1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0]*
D2	*[1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0]*
D3	*[1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1]*

With the code shown in Figure 3.9, the total number of unique words in the vocabulary for this dataset can be retrieved. The dataset used in this study consists of 2432 unique words; hence, the dimensionality (length) for the vector is 2432.

```
#View the total number of unique words in the vocabulary created by Bag of Word

BOW = CountVectorizer()
BOW.fit(data['sentence'])
bow_vectors = BOW.transform(data['sentence'])
print(len(BOW.vocabulary_))

#shape of the BoW matrix
print(bow_vectors.shape)
```

```
2432
(1258, 2432)
```

Figure 3.9 Python code to retrieve total number of unique words in a vocabulary and dimensionality of the vector

3.5.2 TF-IDF (Term Frequency-Inverse Document Frequency)

One challenge with scoring word frequency is that common words, which are frequently used across various documents, tend to overshadow rarer, potentially domain-specific words in terms of importance. To address this issue, TF-IDF is implemented to adjust the word frequencies by considering their prevalence across all documents. This rescaling helps mitigate the influence of highly frequent but less informative words, such as "the," by penalising them based on their commonality across the entire dataset. TF-IDF is a feature extraction technique widely used in natural language processing and information retrieval. It aims to quantify the importance of a within a specific document relative to its occurrence across the entire corpus word by assigning weights. TF-IDF is calculated based on two components: Term Frequency (TF) and Inverse Document Frequency (IDF).

Term Frequency (TF) measures the frequency of a term (t) occurrence within a specific document (d). It is calculated by using the formula:

$$TF(t,d) = \frac{Number\ of\ times\ term\ t\ appears\ in\ a\ document}{Total\ number\ of\ terms\ in\ document\ d} \quad \ldots(3.1)$$

In simpler terms, TF is the ratio of the number of times a term appears in a document to the total number of terms in that document. However, the relevancy of a term does not increase proportionally with term frequency in a document (Daniel Jurafsky 2023).

Example: Consider the sentence: “Pihak syarikat didapati belum menjalankan kajian pemetaan suhu ke atas kawasan penstoran.” If want to calculate the TF for the term “pemetaan” in this sentence: TF (“pemetaan”, sentence) = 1/12.

The IDF component in TF-IDF plays a crucial role in discriminating documents by assigning higher weights to less common terms across the entire collection. This is based on the rationale that terms occurring in only a few documents are more discriminative and carry more information about the specific content of those documents. Rare terms, or those with low document frequency, are considered more valuable because they are unique to specific documents. These terms are indicative of the specialised content of certain documents, making them the effective features for distinguishing one document from another. Conversely, terms that occur frequently across the entire collection are likely to be common words and may not contribute significantly to the understanding of individual documents. The IDF term in TF-IDF effectively down-weights these common terms, reducing their impact on the overall score. This helps focus attention on the distinctive and discriminative terms, allowing for a more nuanced representation of the inspection findings.

$$idf_t = log_{10}\left(\frac{N}{df_t}\right) \quad \ldots(3.2)$$

where N is the total number of sentences in the corpus, and where df_t is the number of sentences in which the term t occurs.

For example, consider the term "syarikat" in a collection of sentences related to pharmaceutical inspections. If "syarikat" frequently appears in most sentences, its IDF will be lower, and its contribution to the TF-IDF score for any particular sentence will be diminished. However, suppose a term like "pemetaan" is less common and appears

in only a few sentences. In that case, its IDF will be higher, making it a more influential feature for those specific sentences.

TF-IDF is a combination TF component and IDF component with the equation below:$W_{t,d} = tf_{t,d}\ X\ IDF_t$ …(3.3)

where $tf_{t,d}$ refers to the term frequency t in document d, and IDF refers to the inverse document frequency of term t. For example, consider these three non-conformance findings (i.e. documents) D1, D2 and D3 as a corpus, and each has a sentence as shown in Table 3.6.

Table 3.6 Example of sentences in the corpus

Documents	**Sentences**
D1	*belum menjalankan kajian pemetaan suhu*
D2	*laporan pemeriksaan dalaman belum diwujudkan*
D3	*borang aduan produk belum diwujudkan*

To calculate the TF for each term in the entire corpus, the terms in the whole corpus are listed in Table 3.7.

Table 3.7 Vocabulary (term) frequency

Term	**D1**	**D2**	**D3**
Belum	1	1	1
Menjalankan	1	0	0
Kajian	1	0	0
Pemetaan	1	0	0
Suhu	1	0	0
laporan	0	1	0
Pemeriksaan	0	1	0
Dalaman	0	1	0
Diwujudkan	0	1	1
Borang	0	0	1
Aduan	0	0	1
Produk	0	0	1

As shown in Table 3.7, number (1) is the frequency of the word present in the sentence, while (0) means the absence of the word in the given sentence. IDF will be calculated for each term corresponding to the whole corpus where N = total number of documents, which is 3, and DF_t is the number of term appearances in the three documents. Based on equation (3.2), the calculation of IDF for each term is shown in Table 3.8.

Table 3.8 IDF calculation

Term	**DF_t**	**IDF**
Belum	3	$\log(3/3) = 0$
Menjalankan	1	$\log(3/1) = 0.477$
Kajian	1	$\log(3/1) = 0.477$
Pemetaan	1	$\log(3/1) = 0.477$
Suhu	1	$\log(3/1) = 0.477$
laporan	1	$\log(3/1) = 0.477$
Pemeriksaan	1	$\log(3/1) = 0.477$
Dalaman	1	$\log(3/1) = 0.477$
Diwujudkan	2	$\log(3/2) = 0.176$
Borang	1	$\log(3/1) = 0.477$
Aduan	1	$\log(3/1) = 0.477$
Produk	1	$\log(3/1) = 0.477$

With the value from both Table 3.7 and Table 3.8, the TF-IDF for each term can be obtained, as shown in Table 3.9.

Table 3.9 TF-IDF value for each term

Term	D1	D2	D3
Belum	0	0	0
Menjalankan	0.477	0	0
Kajian	0.477	0	0
Pemetaan	0.477	0	0
Suhu	0.477	0	0
laporan	0	0.477	0
Pemeriksaan	0	0.477	0
Dalaman	0	0.477	0
Diwujudkan	0	0.176	0.176
Borang	0	0	0.477
Aduan	0	0	0.477
Produk	0	0	0.477

In summary, the IDF component in TF-IDF is a crucial mechanism for highlighting terms that provide discriminatory power and unique information about the content of documents. It effectively addresses the challenge of distinguishing documents based on the rarity and uniqueness of terms, contributing to the overall effectiveness of TF-IDF as a feature extraction technique in text mining and document classification tasks.

3.5.3 Bigrams

Bigrams feature extraction considers pairs of consecutive words as features. This approach captures the contextual relationships between words, allowing the model to understand not only individual terms but also the interactions between adjacent terms. For example, a pair of words like “audit dalaman” or “panggil balik” can be identified as a feature instead of “audit” and “dalaman” or “panggil” and “balik” which does not show any relationship between them. Bigrams may enhance the model's ability to recognise phrases within the inspection findings.

3.6 CLASSIFICATION

The dataset is divided into a training set and a test set with an 80:20 ratio through stratified random sampling methods. This ensures the distribution of class labels is preserved in both the training and test sets. Supervised machine learning algorithms such as NB, LR, SVM and KNN are used in this study to classify non-conformance GDP inspection findings.

The first method used for classification is Naïve Bayes. NB is a probabilistic classifier. It means that for a document d, out of all classes c ∈ C. The classifier returns the class c, which has the maximum posterior probability given the document computed through the equation below (Daniel Jurafsky 2023):

$$P(c|d) = \frac{P(d|c)P(c)}{P(d)} \quad \ldots(3.4)$$

Logistic Regression is also a probabilistic classifier. It estimates the probability of an event occurring, such as voting or not voting, based on a given dataset of independent variables. It can be used to classify a sentence into of many classes by using multinomial logistic regression (also called softmax regression).

Support Vector Machine (SVM) is a type of supervised machine learning algorithm that can classify both linear and non-linear data. SVM employs a nonlinear mapping technique to transform the original training data into a higher dimension. In this new dimension, SVM looks for an optimal linear separating hyperplane between the classes. By using an appropriate nonlinear mapping to a sufficiently high dimension, SVM can always separate the data of two classes using a hyperplane (Han J, 2012). The SVM finds this hyperplane using support vectors and margins through the equation below:

$$f(\vec{x}) = sgn\big((\vec{x}\ X\ \vec{w}) + b\big) = \pm 1 \quad \ldots(3.5)$$

$$+1: (\vec{x}\ X\ \vec{w}) + b > 0$$

$$-1: Otherwise$$

It maps the optimum hyperplane with the optimum margin. Assume a positive and negative data instance partitioned by a hyperplane and the shortest path p+(p-) is lying between the nearest positive and nearest negative instances. In this case, the margin of this hyperplane is given as $p_+ + p_-$.

KNN is a classifier that predicts the class of an instance based on its nearest neighbour. The classifier determines the similarity between the existing and new data, assigning the new data to the category with the highest similarity. It is also called "lazy learner" algorithm because KNN doesn't learn from the training data immediately but rather makes decisions at the time of classification (Hassan et al. 2022). The value of k to be used in this study is set as 11.

3.7 EVALUATION

The classifiers' performance is evaluated through a confusion matrix to derive accuracy, precision, recall and F1 score. Figure 3.10 shows an example of a confusion matrix.

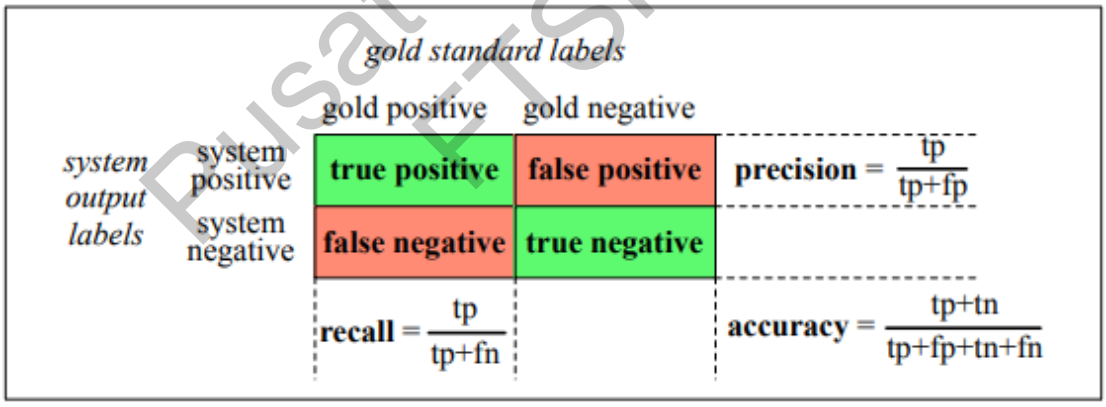


Figure 3.10 Confusion matrix

Gold standard labels are the human-defined chapter for each non-conformance sentence, whereas system output labels are the prediction of the chapters for each sentence classified by the classifier. For example, if the researcher would like to classify the sentences into Chapter 1 of the GDP guideline, true positive (TP) means the classifier correctly classifies the Chapter 1 sentences into the right chapter. False negative (FN) is defined as the classifier incorrectly classifying chapter 1 sentence into other chapters. False positive (FP) means the classifier incorrectly classifies non-

chapter 1 sentences as chapter 1 sentences, whereas true negative (TN) means the classifier correctly classifies non-chapter 1 sentences into other chapters.

With the value from the confusion matrix, metrics such as accuracy, precision, recall and F1 score can be derived.

Accuracy measures the percentage of all non-conformances sentences correctly labelled by the classifier. However, accuracy is not a good metric for performance measurement for this dataset as the dataset consists of unbalanced classes (Daniel Jurafsky 2023).

Precision measures the percentage of sentences the classifier detected (classified as Chapter 1) that are indeed Chapter 1 sentences. It is defined as:

$$Precision = \frac{true\ positives}{true\ positives + false\ positives} \quad \ldots(3.6)$$

Recall measures the percentage of sentences present in the input correctly identified by the classifier. It is defined as:

$$Recall = \frac{true\ positives}{true\ positives + false\ negatives} \quad \ldots(3.7)$$

F-measure is the harmonic mean of precision and recall. It gives equal weight to precision and recall. It is defined as:

$$F1 = \frac{2\ X\ Precision\ X\ Recall}{Precision + Recall} \quad \ldots(3.8)$$

This study aims to classify the GDP non-conformance finding into more than two classes (multi-class classification). Therefore, macroaveraging of precision and recall will be performed in which the performance of each class will be computed and then averaged over classes as this is more appropriate when performance on all classes is equally important (Daniel Jurafsky 2023).

3.8 SUMMARY

This chapter outlined the research methodology used in this study by identifying the research design framework and the 5 stages of system development and evaluation. The next chapter will describe the results of the study based on the methodology.

CHAPTER IV

RESULTS AND DISCUSSION

4.1 INTRODUCTION

This chapter discusses the experimental results based on the methodology in chapter 3. Section 4.2 explains the experiment setting used in the study and the dataset to be used after extracted from GDP inspection reports. Section 4.3 shows the results of classification through a combination of bag of word features with NB, LR, SVM and KNN. Section 4.4 shows the results of classification through a combination of TF-IDF features with NB, LR, SVM and KNN. Section 4.5 shows the results of classification through a combination of Bigrams features with NB, LR, SVM and KNN. Section 4.6 shows the comparative analysis of the results between features and classifiers, and section 4.7 is the comprehensive discussion of the comparative study.

4.2 EXPERIMENT SETTING

The experiments were conducted in Python 3.6 using Google Colab as the primary integrated development environment (IDE). Google Colab offers a cloud-based platform that facilitates collaborative coding and easy access to powerful computing resources.

The following Python libraries were utilised to implement and conduct the experiments, as detailed in Table 4.1.

Table 4.1 Python libraries

Library	Explanation
Pandas	Used for data manipulation and analysis. Pandas offers data structures such as data frames, making it efficient for handling structured data.
Matplotlib	Employed for data visualisation to create clear and informative plots and charts.
Scikit-learn	A machine learning library that provides tools for data mining and data analysis. It includes a stratified random sampling method, NB, SVM, LR, KNN classifiers and performance evaluation metrics.
Malaya	Natural Language toolkit library for Bahasa Malaysia, powered by Tensorflow and PyTorch.

4.2.1 Dataset Generation

The dataset used in this study was generated through manual extraction of GDP inspection reports' non-conformance findings for the year 2022, as detailed in Chapter 3. This results in a dataset with two attributes and 1 class attribute with a total of 1383 instances before removal of duplicate sentences. After the removal of duplicate sentences, it consisted of a total of 1258 instances, and all the sentences were written in Malay language.

4.3 BAG OF WORDS RESULTS

Table 4.2 shows the performance of classifiers combined with the Bag of Words feature extraction method. The results are presented in terms of accuracy, precision (macro-averaged), recall (macro-average) and F1 score (macro-average). However, the F1 score will be used as the final result for comparing the performance between the combinations.

Table 4.2 Classification performance of Bag of Words with Different Classifiers

Feature	Classifier	Accuracy	Precision (macro-averaged)	Recall (macro-averaged)	F1 Score (macro-averaged)
Bag of Words	Naïve Bayes	0.87	0.89	0.84	0.86
	Logistic Regression	0.92	0.93	0.92	0.92
	SVM	0.87	0.91	0.83	0.86
	KNN	0.71	0.76	0.70	0.70

4.3.1 Bag of Words with Naïve Bayes

Although Naïve Bayes and Support Vector Machine models are frequently used as baselines in text categorisation and sentiment analysis, the performance of Naive Bayes (NB) and Support Vector Machines (SVM) varies depending on the length of the document (Wang & Manning 2012). Specifically, for short snippet sentiment tasks, NB demonstrates better performance than SVMs. This suggests that NB is more effective when dealing with shorter text snippets. On the other hand, for longer documents, SVMs tend to outperform NB. This indicates that SVMs are more suitable for longer pieces of text. Additionally, the Multivariate Bernoulli NB (BNB) usually performs worse than Multinomial NB (MNB) and is less stable than MNB. Furthermore, Wang and Manning (2012) show that bag of features models are still strong performers on snippet sentiment classification tasks, with NB models generally outperforming the sophisticated, structure-sensitive models explored in recent work. The Malay non-conformance GDP findings dataset used in this study mostly consisted of short sentences. Therefore, Bag of Words (Unigrams) with multinomial naïve bayes classifier will be chosen as the baseline combination for performance comparison. In this study, although BoW with NB classifier is considered as a simple and basic combination, it can achieve good metrics score with an accuracy rate of 87%, precision (89%) and recall (84%) with an F1 score of 86%.

4.3.2 Bag of Words with Logistic Regression

Logistic regression shows that it outperformed the other classifiers in every metrics when used with the Bag of Words feature extraction method. This model achieved the highest accuracy rate of 92%, highest precision rate of 93%, highest recall rate of 92%, and F1 score of 92% in Bag of Words representation.

4.3.3 Bag of Words with SVM

SVM with Bag of Words combination achieved accuracy (87%), precision (91%), recall (83%) and F1 score (86%). Although it has a high precision score (91%), overall, its performance is on par with the baseline combination due to a lower recall rate.

4.3.4 Bag of Words with KNN

KNN exhibited the lowest metrics among all classifiers in the Bag of Words representation. Although it has moderate precision (76%), it performed average in accuracy (71%) and recall (70%), thus resulting in the lowest F1 score (70%). Therefore, KNN might not be as effective as other classifiers for BoW features in the Malay GDP inspection findings classification.

4.4 TERM FREQUENCY-INVERSE DOCUMENT FREQUENCY RESULTS

Table 4.3 shows the performance of classifiers combined with the TF-IDF feature extraction method.

Table 4.3 Classification performance of TF-IDF with different classifiers

Feature	Classifier	Accuracy	Precision (macro-averaged)	Recall (macro-averaged)	F1 Score (macro-averaged)
TF-IDF	Naïve Bayes	0.79	0.89	0.70	0.76
	Logistic Regression	0.89	0.93	0.86	0.88
	SVM	0.88	0.93	0.86	0.89
	KNN	0.89	0.88	0.89	0.88

4.4.1 TF-IDF with NB

TF-IDF with NB combination does not perform well in classifying Malay GDP inspection findings as it has the lowest accuracy (79%) and recall (70%) among the classifiers with the TF-IDF feature. Besides that, it doesn't outperform the baseline combination although having a similar precision rate of 89%.

4.4.2 TF-IDF with LR

TF-IDF with LR combination obtained the highest accuracy of 89% among the classifier with TF-IDF. It has a 93% precision rate and 86% recall rate. It is not the best performance classifier with TF-IDF, as its F1 score is 88%. However, it outperformed the baseline combination in every metrics.

4.4.3 TF-IDF with SVM

TF-IDF with SVM combination achieved 88% of accuracy in classifying Malay GDP inspection findings with high precision (93%), indicating it has a low false-positive rate. Besides that, it also performs well in terms of recall (86%) and F1 score (89%). SVM classifier has the best performance among other classifiers when using TF-IDF as a feature.

4.4.4 TF-IDF with KNN

TF-IDF with KNN combination demonstrates a balanced performance across all metrics. It achieves an accuracy of 89% with consistency in precision (88%) and recall (89%), thus resulting in a consistent F1 score (88%).

4.5 N-GRAMS RESULT

Table 4.4 shows the performance of classifiers combined with the n-grams feature extraction method.

Table 4.4 Classification performance of n-grams with different classifiers

Feature	Classifier	Accuracy	Precision (macro-averaged)	Recall (macro-averaged)	F1 Score (macro-averaged)
Bigrams	Naïve Bayes	0.87	0.88	0.85	0.86
	Logistic Regression	0.83	0.92	0.79	0.84
	SVM	0.72	0.93	0.67	0.75
	KNN	0.58	0.82	0.54	0.59
Trigrams	Naïve Bayes	0.76	0.81	0.72	0.75
	Logistic Regression	0.67	0.89	0.62	0.71
	SVM	0.50	0.94	0.44	0.52
	KNN	0.41	0.70	0.31	0.36

4.5.1 Bigrams

Bigrams with NB achieved comparable performance with baseline combination as both have similar metrics scores and the same F1 score. This showed that bigrams do not provide the additional benefit of creating meaningful sequence words for the classification. Bigrams with LR also achieved lower accuracy (0.83), recall (0.79) and F1 score (0.84) compared to the baseline combination. This also suggests that bigrams might not be as beneficial for Logistic Regression. Bigrams with SVM achieved the lowest accuracy (0.72), recall (0.67) and F1 score (0.75) among all SVM combinations. This indicates that bigrams might not be compatible with SVM for the classification of GDP inspection findings classification. Bigrams with KNN achieved the worst performance among all combinations. It has the lowest accuracy (0.58), recall (0.54) and F1 score (0.59). Recall is higher for bigrams compared to trigrams across all classifiers. This indicates that bigrams are better at capturing most of the relevant instances in the dataset. Naïve Bayes consistently shows the best recall for both bigrams and trigrams.

4.5.2 Trigrams

Across all classifiers, the performance are worsen with trigrams in terms of accuracy when comparing with bigrams. Naïve Bayes and Logistic Regression show the highest accuracy with bigrams, indicating that these classifiers might better capture the information encoded in bigrams compared to trigrams.

4.6 WORD EMBEDDING RESULT

Table 4.5 shows the performance of classifiers using word embedding such as Word2Vec as feature. Overall, the performance of all classifiers are the worse compared with other features. Word2Vec might not be capturing the necessary information for this classification task and this could be due to the dataset might be too small thus not have enough data to effectively train the embeddings method. Therefore, word embedding method should not be used in this study.

Table 4.5 Classification performance using Word2Vec as feature

Feature	Classifier	Accuracy	Precision (macro-averaged)	Recall (macro-averaged)	F1 Score (macro-averaged)
Word2Vec	Naïve Bayes	0.15	0.13	0.17	0.11
	Logistic Regression	0.27	0.03	0.09	0.04
	SVM	0.27	0.02	0.09	0.04
	KNN	0.46	0.49	0.33	0.42

4.7 COMPARISON ANALYSIS

The comparative analysis of classification results provides a detailed examination of the performance of different classifiers across distinct feature extraction methods. This analysis aims to identify a suitable combination for the task of classifying Malay language Good Distribution Practice (GDP) inspection findings.

4.7.1 Feature Comparison

Table 4.6 shows the overview result of text classification performance based on the three features which are Bag of Words, TF-IDF and Bigrams.

Table 4.6 Overview of text classification performance based on feature

Feature	Classifier	Accuracy	Precision (macro-averaged)	Recall (macro-averaged)	F1 Score (macro-averaged)
Bag of Words	Naïve Bayes	0.87	0.89	0.84	0.86
	Logistic Regression	0.92	0.93	0.92	0.92
	SVM	0.87	0.91	0.83	0.86
	KNN	0.71	0.76	0.70	0.70
TF-IDF	Naïve Bayes	0.79	0.89	0.70	0.76
	Logistic Regression	0.89	0.93	0.86	0.88
	SVM	0.88	0.93	0.86	0.89
	KNN	0.89	0.88	0.89	0.88
Bigrams	Naïve Bayes	0.87	0.88	0.85	0.86
	Logistic Regression	0.83	0.92	0.79	0.84
	SVM	0.72	0.93	0.67	0.75
	KNN	0.58	0.82	0.54	0.59

Table 4.7 shows the average score of each performance metrics for each feature. The average scores provide an overall summary of the performance of each feature representation across the different evaluation metrics, facilitating comparison and selection of the most suitable feature representation in this study.

Table 4.7 Average score of performance metrics for each feature

Feature	Accuracy	Precision	Recall	F1 Score
Bag of Words	0.84	0.87	0.82	0.84
TF-IDF	0.86	0.91	0.83	0.85
Bigrams	0.75	0.89	0.71	0.76

Figure 4.1 shows the performance comparison among features in the average score of every performance metrics. The Bag of Words feature extraction method achieved an average accuracy of 0.84, precision of 0.87, recall of 0.82, and F1 score of

0.84. This suggests that the Bag of Words method is effective for capturing relevant information for Malay language inspection findings classification and suitable for use across all classifiers.

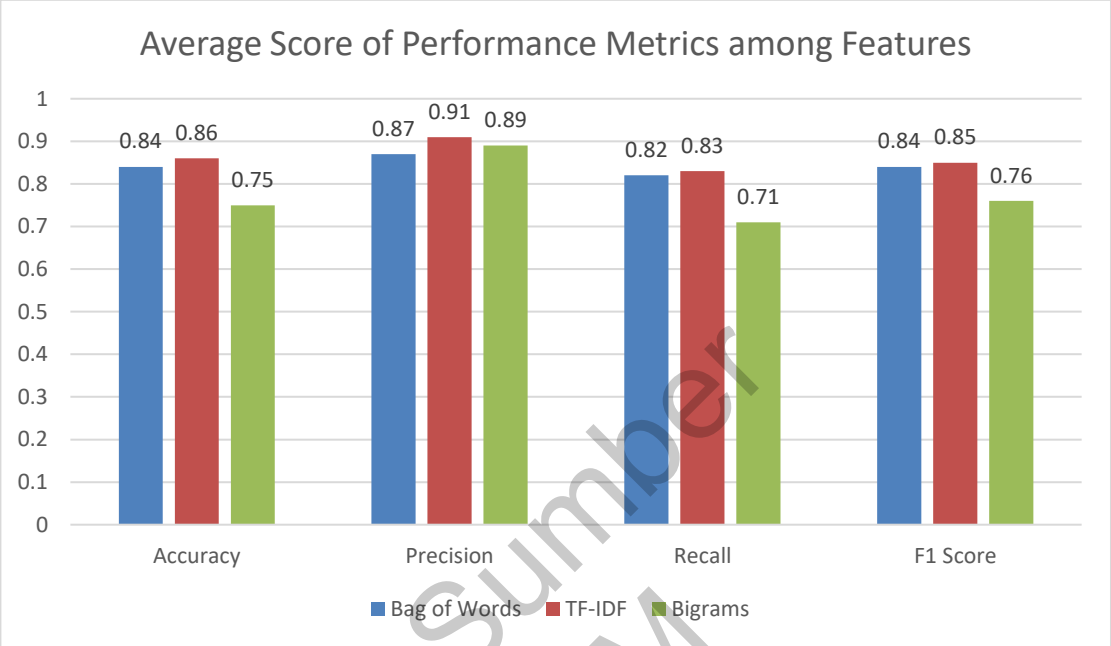


Figure 4.1 Performance comparison among features

The TF-IDF feature extraction method achieved an average accuracy of 0.86, precision of 0.91, recall of 0.83, and F1 score of 0.85. This suggests that the TF-IDF feature is also as effective for capturing relevant information for Malay language inspection findings classification as the Bag of Words feature and robust feature for use with different classifiers.

The Bigrams feature extraction method achieved an average accuracy of 0.75, precision of 0.89, recall of 0.71, and F1 score of 0.76. It shows that bigrams did not provide a consistent improvement over BoW and even led to a worse performance with some classifiers. This suggests that bigrams might not be as beneficial for this specific task as they might introduce noise and redundancy. This results also coincides with the study by Wang and Manning (2012) which stated bigrams is not commonly used in topical text categorization due to mixed performance.

4.7.2 Classifier Comparison

Table 4.8 shows the performance metrics of different classifiers using Bag of Words, TF-IDF, and Bigrams feature representations. The metrics include accuracy, precision, recall, and F1 score, all of which are evaluated using macro-averaging.

Table 4.8 Overview of text classification performance based on classifier

Classifier	Feature	Accuracy	Precision (macro-averaged)	Recall (macro-averaged)	F1 Score (macro-averaged)
Naïve Bayes	Bag of Words	0.87	0.89	0.84	0.86
	TF-IDF	0.79	0.89	0.7	0.76
	Bigrams	0.87	0.88	0.85	0.86
Logistic Regression	Bag of Words	0.92	0.93	0.92	0.92
	TF-IDF	0.89	0.93	0.86	0.88
	Bigrams	0.83	0.92	0.79	0.84
SVM	Bag of Words	0.87	0.91	0.83	0.86
	TF-IDF	0.88	0.93	0.86	0.89
	Bigrams	0.72	0.93	0.67	0.75
KNN	Bag of Words	0.71	0.76	0.7	0.7
	TF-IDF	0.89	0.88	0.89	0.88
	Bigrams	0.58	0.82	0.54	0.59

Table 4.9 shows the average score of performance metrics of different classifiers across accuracy, precision, recall, and F1-score. The average scores give an overall summary of the classifiers' performance across the metrics.

Table 4.9 Average score of performance metrics based on classifier

Classifier	Accuracy	Precision	Recall	F1 Score
Naïve Bayes	0.84	0.89	0.80	0.83
Logistic Regression	0.88	0.93	0.86	0.88
SVM	0.82	0.92	0.79	0.83
KNN	0.73	0.82	0.71	0.72

Figure 4.2 shows the performance comparison among classifiers in average score of every performance metrics. The Naïve Bayes classifier achieved average accuracy (0.84), precision (0.89), recall (0.80) and F1 score (0.83).

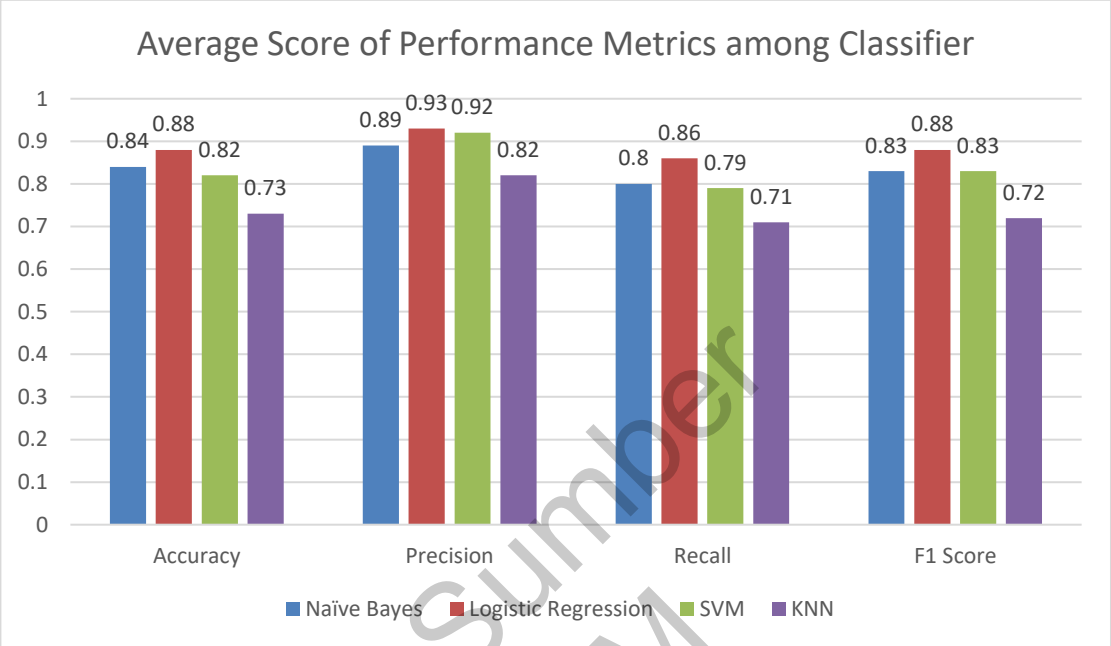


Figure 4.2 Performance comparison among classifiers

The Logistic Regression classifier achieved the highest average accuracy (0.88), precision (0.93), recall (0.86), and F1 score (0.88). This suggests that the Logistic Regression classifier is the best performing classifier and is robust to be used with different types of features.

The SVM classifier achieved an average accuracy of 0.82, precision of 0.92, recall of 0.79, and F1 score of 0.83. This suggests that the SVM classifier is as effective as the NB classifier.

The KNN classifier achieved the lowest average accuracy (0.73), precision (0.82), recall (0.71), and F1 score (0.72). This suggests that the KNN classifier is the least effective classifier for this classification. This is also probably due to the wrong selection of the k-value for this classification and its fundamental of dependency on the majority of members of the class, hence unsuitable for sentences with highly diverse data (Jaafar et al. 2016).

4.7.3 Confusion Matrix

Figure 4.3 shows the confusion matrix for using BoW as feature. From the confusion matrix, the baseline model (BoW + NB) struggles slightly with chapter 4 and 8, indicating potential difficulty in distinguishing these classes. Whereas BOW + LR only show slightly lower performance on clasifying chapter 4 and chapter 11. This could be due to similarity of common words between both chapters as shown in the result from word cloud analysis. BOW + SVM also suffered similar issues in which the precision rate for chapter 4 and chapter 11 is below 68%. It indicated high number of findings are incorrectly predicted as chapter 4 and chapter 11. The combination of BoW + kNN shows bad performances for identifying chapter 4 and chapter 10 audit findings. It only correctly classifying chapter 4 half of the time while 65% of chance to wrongly classify findings into chapter 10.

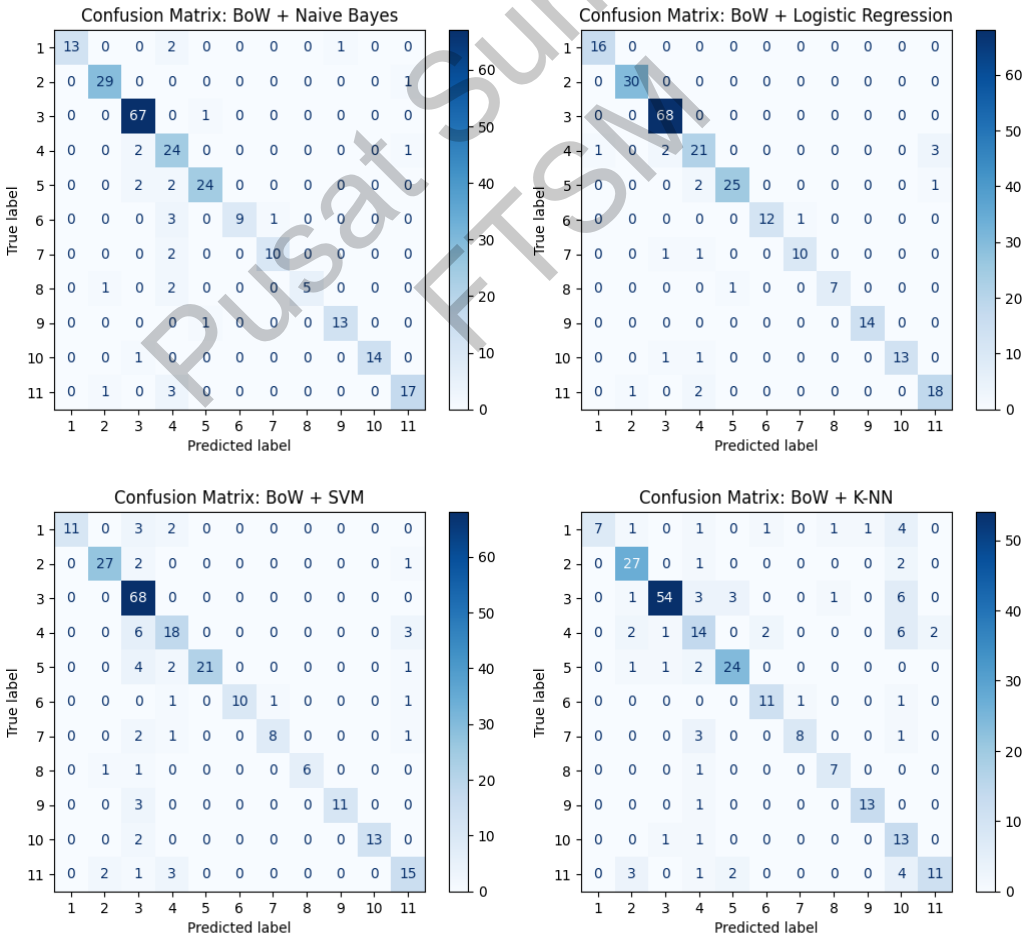


Figure 4.3 Confusion matrix with BoW as feature

Figure 4.4 shows the confusion matrix for using TF-IDF as feature. TF-IDF + NB shows difficulties in correctly classifying chapter 1, 6, 7, 8 and 11 especially chapter 8 with recall rate of 0.25 only, leading to lower overall performance. Logistic Regression with TF-IDF shows strong performance, although slightly lower than with BoW. It maintains good precision and recall across most classes, making it a robust choice. TF-IDF with SVM shows moderate performance in correctly classify chapter 8 probably due to low support (lowest instance for training set and test set) whereas TF-IDF with KNN shows moderate performance in classifying chapter 11.

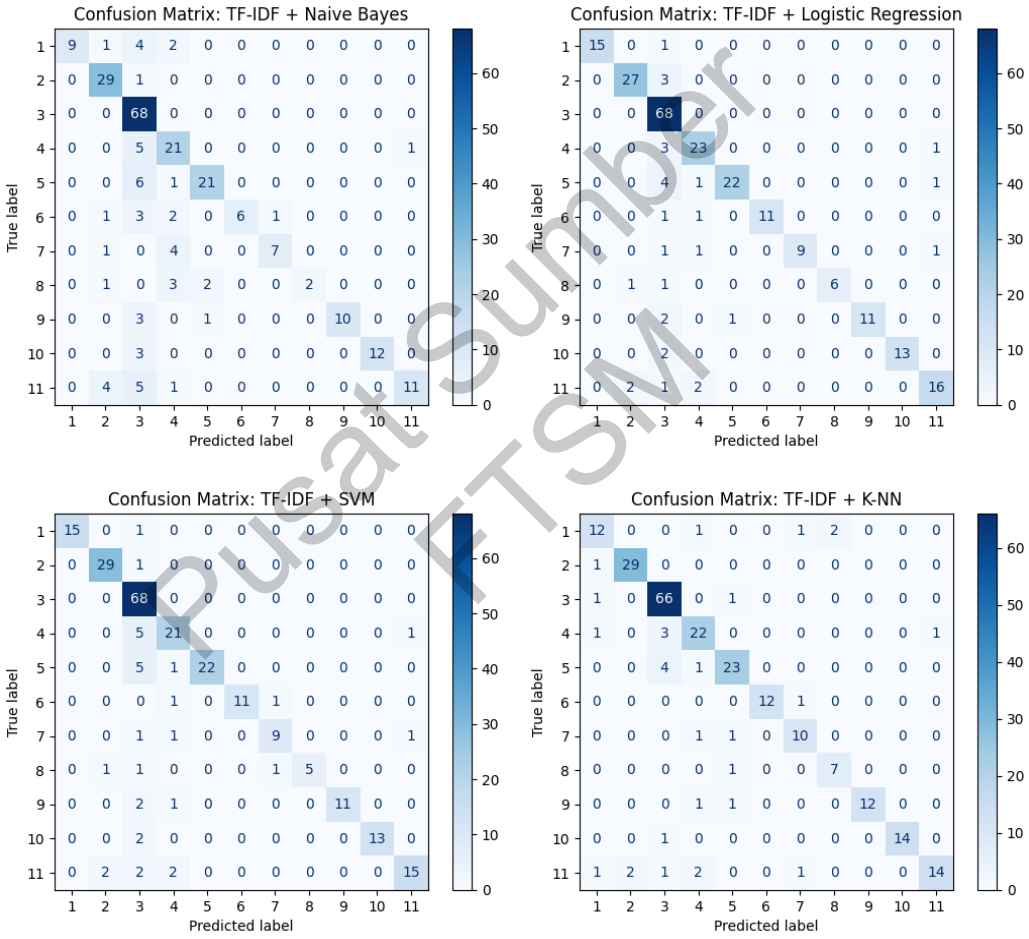


Figure 4.4 Confusion matrix with TF-IDF as feature

Figure 4.5 show the confusion matrix for using bigrams as feature. Naive Bayes with bigrams performs well, with good precision and recall for most chapters. However, it has low recall rate of 0.62 for Chapter 1 and 6, leading to lower overall performance compared to Logistic Regression with BoW. Logistic Regression with bigrams has lower recall rate for several classes such as Chapter 1, 4, 6 and 11, indicating that bigrams may not capture features as effectively. Further examination on the SVM with bigrams and KNN with bigrams shows deteriorated in recall rate for most of the classes. Especially bigrams with KNN could not classify Chapter 4 and 11 at all with recall rate of 0.07 and 0.04. It also shows that 2/3 of Chapter 4 findings are wrongly classified as Chapter 3 (18/27 instances). Therefore, bigrams should not be use as feature for inspection finding classification.

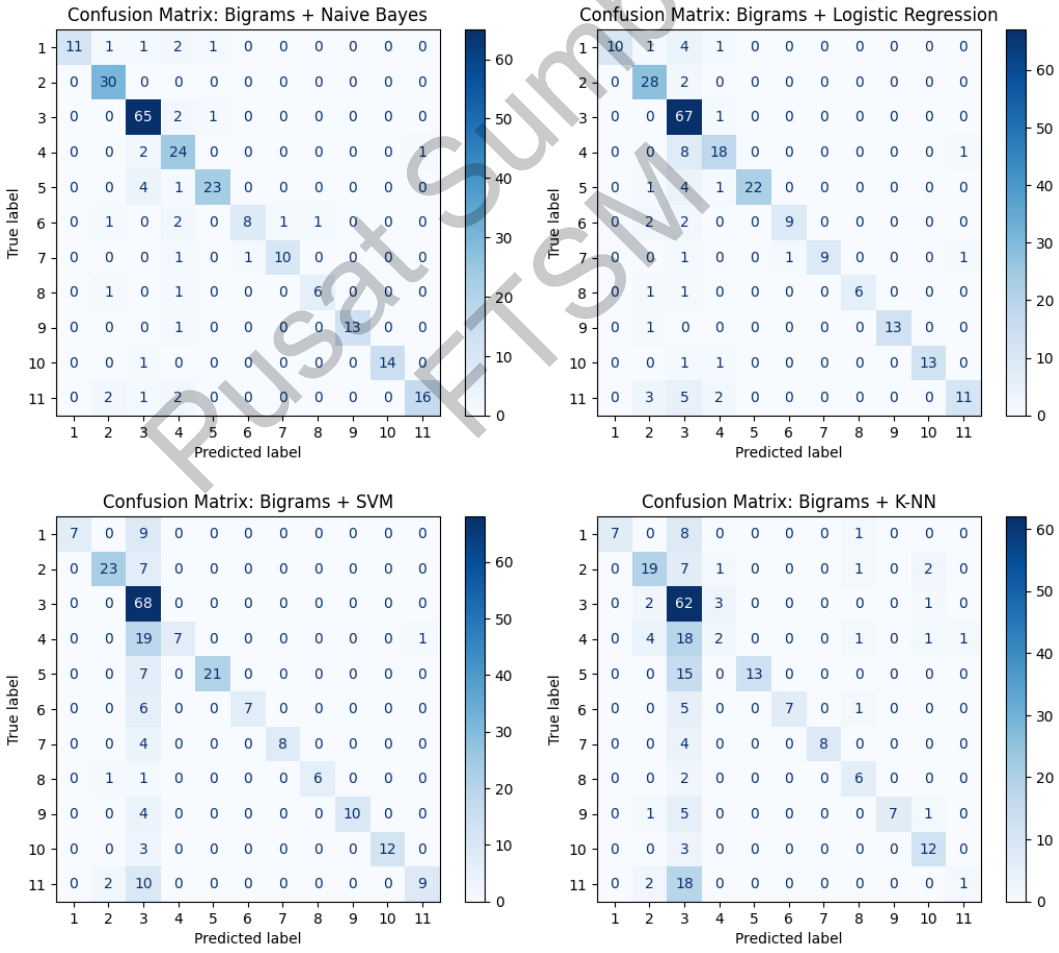


Figure 4.5 Confusion matrix with Bigrams as feature

4.7.4 Multi-class Comparison

Table 4.10 shows the performance metrics of models for Chapter 1 – Quality System. Logistic Regression with BoW and TF-IDF, SVM with TF-IDF are the best models for classifying Chapter 1 (F1-score = 0.97). While KNN with Bow and Bigrams, SVM with Bigrams have the lowest recall rate (0.44) and lowest F1-score (0.61).

Table 4.10 Performance metrics for Chapter 1 - Quality System

Chapter	Classifier	Feature	Precision	Recall	F1-score	Support
1	NB	BoW	1	0.81	0.9	16
		TF-IDF	1	0.56	0.72	16
		Bigrams	1	0.69	0.81	16
	LR	BoW	0.94	1	0.97	16
		TF-IDF	1	0.94	0.97	16
		Bigrams	1	0.62	0.77	16
	SVM	BoW	1	0.69	0.81	16
		TF-IDF	1	0.94	0.97	16
		Bigrams	1	0.44	0.61	16
	KNN	BoW	1	0.44	0.61	16
		TF-IDF	0.75	0.75	0.75	16
		Bigrams	1	0.44	0.61	16

Table 4.11 shows the models' performance metrics for Chapter 2 – Personnel. Generally, all models performed very well in classifying chapter 2 except KNN with bigrams (F1-score = 0.66). Logistic Regression with BoW is the best performer model with F1-score of 0.98.

Table 4.11 Performance metrics for Chapter 2 - Personnel

Chapter	Classifier	Feature	Precision	Recall	F1-score	Support
2	NB	BoW	0.94	0.97	0.95	30
		TF-IDF	0.78	0.97	0.87	30
		Bigrams	0.86	1	0.92	30
	LR	BoW	0.97	1	0.98	30
		TF-IDF	0.90	0.90	0.90	30
		Bigrams	0.76	0.93	0.84	30
	SVM	BoW	0.9	0.9	0.9	30
		TF-IDF	0.91	0.97	0.94	30
		Bigrams	0.88	0.77	0.82	30
	KNN	BoW	0.77	0.9	0.83	30
		TF-IDF	0.94	0.97	0.95	30
		Bigrams	0.68	0.63	0.66	30

Table 4.12 shows the models' performance metrics for Chapter 3 – Premises and Equipment. The models perform similarly compared with classifying Chapter 2, in which KNN with Bigrams is still the worst model (F1-score = 0.58). LR with BoW performed particularly well with an F1-score of 0.97.

Table 4.12 Performance metrics for Chapter 3 – Premises and Equipment

Chapter	Classifier	Feature	Precision	Recall	F1-score	Support
3	NB	BoW	0.93	0.99	0.96	68
		TF-IDF	0.69	1	0.82	68
		Bigrams	0.88	0.96	0.92	68
	LR	BoW	0.94	1	0.97	68
		TF-IDF	0.78	1	0.88	68
		Bigrams	0.71	0.99	0.82	68
	SVM	BoW	0.74	1	0.85	68
		TF-IDF	0.77	1	0.87	68
		Bigrams	0.49	1	0.66	68
	KNN	BoW	0.95	0.79	0.86	68
		TF-IDF	0.88	0.97	0.92	68
		Bigrams	0.42	0.91	0.58	68

Table 4.13 shows the models' performance metrics for Chapter 4 – Stock Handling and Stock Control. Generally, all the models have moderate performance in classifying Chapter 4, with an F1-score between 0.67 to 0.84 except Bigrams with KNN and SVM and Bow with KNN. KNN with Bigrams shows difficulties in classifying Chapter 4 with recall rate of merely 0.07.

Table 4.13 Performance metrics for Chapter 4 – Stock Handling and Stock Control

Chapter	Classifier	Feature	Precision	Recall	F1-score	Support
4	NB	BoW	0.63	0.89	0.74	27
		TF-IDF	0.62	0.78	0.69	27
		Bigrams	0.67	0.89	0.76	27
	LR	BoW	0.78	0.78	0.78	27
		TF-IDF	0.82	0.85	0.84	27
		Bigrams	0.75	0.67	0.71	27
	SVM	BoW	0.67	0.67	0.67	27
		TF-IDF	0.78	0.78	0.78	27
		Bigrams	1	0.26	0.41	27
	KNN	BoW	0.5	0.52	0.51	27
		TF-IDF	0.79	0.81	0.8	27
		Bigrams	0.33	0.07	0.12	27

Table 4.14 shows the models' performance metrics for Chapter 5 – Transportation. All of the models performed very well in classifying Chapter 5 with precision rate above 0.83 and F1-scores above 0.81, except KNN with Bigrams with low recall rate of 0.46.

Table 4.14 Performance metrics for Chapter 5 – Transportation

Chapter	Classifier	Feature	Precision	Recall	F1-score	Support
5	NB	BoW	0.92	0.86	0.89	28
		TF-IDF	0.88	0.75	0.81	28
		Bigrams	0.92	0.82	0.87	28
	LR	BoW	0.96	0.89	0.93	28
		TF-IDF	0.96	0.79	0.86	28
		Bigrams	1	0.79	0.88	28
	SVM	BoW	1	0.75	0.86	28
		TF-IDF	1	0.79	0.88	28
		Bigrams	1	0.75	0.86	28
	KNN	BoW	0.83	0.86	0.84	28
		TF-IDF	0.85	0.82	0.84	28
		Bigrams	1	0.46	0.63	28

Table 4.15 shows the models' performance metrics for Chapter 6 – Products / Cosmetics Complaints. LR with BoW and KNN with TF-IDF have the highest F1-score of 0.96.

Table 4.15 Performance metrics for Chapter 6 – Products / Cosmetics Complaints

Chapter	Classifier	Feature	Precision	Recall	F1-score	Support
6	NB	BoW	1	0.69	0.82	13
		TF-IDF	1	0.46	0.63	13
		Bigrams	0.89	0.62	0.73	13
	LR	BoW	1	0.92	0.96	13
		TF-IDF	1	0.85	0.92	13
		Bigrams	0.90	0.69	0.78	13
	SVM	BoW	1	0.77	0.87	13
		TF-IDF	1	0.85	0.92	13
		Bigrams	1	0.54	0.70	13
	KNN	BoW	0.79	0.85	0.81	13
		TF-IDF	1	0.92	0.96	13
		Bigrams	1	0.54	0.70	13

Table 4.16 shows the models' performance metrics for Chapter 7 – Products / Cosmetics Recalls. The best model for this chapter is NB with either BoW or Bigrams and also LR with BoW. Overall the models' performance for classifying Chapter 7 is quite similar among each other with f1-scores in between 0.70 and 0.87.

Table 4.16 Performance metrics for Chapter 7 – Products / Cosmetics Recalls

Chapter	Classifier	Feature	Precision	Recall	F1-score	Support
7	NB	BoW	0.91	0.83	0.87	12
		TF-IDF	0.88	0.58	0.70	12
		Bigrams	0.91	0.83	0.87	12
	LR	BoW	0.91	0.83	0.87	12
		TF-IDF	1	0.75	0.86	12
		Bigrams	1	0.75	0.86	12
	SVM	BoW	0.89	0.67	0.76	12
		TF-IDF	0.82	0.75	0.78	12
		Bigrams	1	0.67	0.8	12
	KNN	BoW	0.89	0.67	0.76	12
		TF-IDF	0.77	0.83	0.80	12
		Bigrams	1	0.67	0.80	12

Table 4.17 shows the models' performance metrics for Chapter 8 – Substandard and Falsified Products / Cosmetics. NB with TF-IDF has the worst performance with recall of 0.25 and leads to lowest f1-score (0.40) in classifying Chapter 8. The only outstanding model is LR with BoW (f1-score = 0.93).

Table 4.17 Performance metrics for Chapter 8 – Substandard and Falsified Products / Cosmetics

Chapter	Classifier	Feature	Precision	Recall	F1-score	Support
8	NB	BoW	1	0.62	0.77	8
		TF-IDF	1	0.25	0.40	8
		Bigrams	0.86	0.75	0.80	8
	LR	BoW	1	0.88	0.93	8
		TF-IDF	1	0.75	0.86	8
		Bigrams	1	0.75	0.86	8
	SVM	BoW	1	0.75	0.86	8
		TF-IDF	1	0.62	0.77	8
		Bigrams	1	0.75	0.86	8
	KNN	BoW	0.78	0.88	0.82	8
		TF-IDF	0.78	0.88	0.82	8
		Bigrams	0.6	0.75	0.67	8

Table 4.18 shows the models' performance metrics for Chapter 9 – Outsourced Activities. Most of the models have perfect precision (precision = 1.00) in classifying Chapter 9. However, KNN with Bigrams seems to struggle in recall score thus reducing its overall performance to f1-score of 0.67.

Table 4.18 Performance metrics for Chapter 9 – Outsourced Activities

Chapter	Classifier	Feature	Precision	Recall	F1-score	Support
9	NB	BoW	0.93	0.93	0.93	14
		TF-IDF	1	0.71	0.83	14
		Bigrams	1	0.93	0.96	14
	LR	BoW	1	1	1	14
		TF-IDF	1	0.79	0.88	14
		Bigrams	1	0.93	0.96	14
	SVM	BoW	1	0.79	0.88	14
		TF-IDF	1	0.79	0.88	14
		Bigrams	1	0.71	0.83	14
	KNN	BoW	0.93	0.93	0.93	14
		TF-IDF	1	0.86	0.92	14
		Bigrams	1	0.5	0.67	14

Table 4.19 shows the models' performance metrics for Chapter 10 – Self-Inspection. NB with either Bow or Bigrams and KNN with TF-IDF have the highest f1-score (0.97) whereas KNN with BoW does not have good performance due to low precision (0.35).

Table 4.19 Performance metrics for Chapter 10 – Self-Inspection

Chapter	Classifier	Feature	Precision	Recall	F1-score	Support
10	NB	BoW	1	0.93	0.97	15
		TF-IDF	1	0.8	0.89	15
		Bigrams	1	0.93	0.97	15
	LR	BoW	1	0.87	0.93	15
		TF-IDF	1	0.87	0.93	15
		Bigrams	1	0.87	0.93	15
	SVM	BoW	1	0.87	0.93	15
		TF-IDF	1	0.87	0.93	15
		Bigrams	1	0.8	0.89	15
	KNN	BoW	0.35	0.87	0.5	15
		TF-IDF	1	0.93	0.97	15
		Bigrams	0.71	0.8	0.75	15

Table 4.20 shows the models' performance metrics for Chapter 11 – Management of Documentation and Records. The results show a high variation in performance among the models. NB with BoW and LR with BoW show a more consistent score for precision, recall and f1-score in the range of 80s. KNN with Bigrams is not effective in classifying Chapter 11 with f1-score of merely 0.09.

Table 4.20 Performance metrics for Chapter 11 – Management of Documentation and Records

Chapter	Classifier	Feature	Precision	Recall	F1-score	Support
11	NB	BoW	0.89	0.81	0.85	21
		TF-IDF	0.92	0.52	0.67	21
		Bigrams	0.94	0.76	0.84	21
	LR	BoW	0.82	0.86	0.84	21
		TF-IDF	0.84	0.76	0.8	21
		Bigrams	0.85	0.52	0.65	21
	SVM	BoW	0.68	0.71	0.7	21
		TF-IDF	0.88	0.71	0.79	21
		Bigrams	0.9	0.43	0.58	21
	KNN	BoW	0.85	0.52	0.65	21
		TF-IDF	0.93	0.67	0.78	21
		Bigrams	0.5	0.05	0.09	21

Naive Bayes presents a pattern of high precision across all chapters except Chapter 4. It also constantly shows lowest f1-scores across all chapters when using TF-IDF as feature compare to Bow and Bigrams. This indicates that N-grams feature is more suitable for NB in classifying GDP inspection findings.

Overall, Logistic Regression (LR) consistently demonstrates robust performance across most classes, characterized by high precision and recall, leading to strong F1-scores. LR excels in classes 1, 2, 3, 5, 6, 8, 9, and 10, where its F1-scores generally exceed 0.90. However, it performs moderately in classes 4, 7, and 11, with F1-scores ranging from 0.81 to 0.87. In most chapters, LR with BoW achieves the highest f1-scores, suggesting it effectively captures relevant features for classification.

The Support Vector Machine (SVM) also exhibits high precision but occasionally suffers from lower recall in some classes, such as classes 1, 4, 6, 7, 8 and 11, especially when using Bigrams as a feature.

KNN's performance is notably weaker when combined with Bigrams, especially in classes 4 and 11, where the F1-scores are below 0.12 due to bad recall rate. However, KNN benefits significantly from TF-IDF, making it a suitable choice if KNN is preferred.

Table 4.21 shows the models' average score of precision, recall and f1-score for classification of each chapter. The results shows that Chapter 1, 4, 8, 11 have f1-score lower than 0.80. Another observation is chapters with low support such as Chapter 6, 7 and 8 tend to have lower recall rate. Perhaps the performance can be improved by increasing size of dataset for these chapters.

Table 4.21 Average performance metrics for classifying each Chapter

Chapter	Precision (average)	Recall (average)	F1-score (average)	Support
1	0.97	0.69	0.79	16
2	0.86	0.91	0.88	30
3	0.77	0.97	0.84	68
4	0.70	0.66	0.65	27
5	0.94	0.78	0.85	28
6	0.97	0.73	0.82	13
7	0.92	0.74	0.81	12
8	0.92	0.72	0.79	8
9	0.99	0.82	0.89	14
10	0.92	0.87	0.88	15
11	0.83	0.61	0.69	21

Table 4.22 summarized the best and worst models for each chapter's classification.

Table 4.22 Summarised table of the best and poor models for each chapter

Chapter	Best Model	F1-score	Poor Model	F1-score	Support
1	LR with BoW LR with TF-IDF SVM with TF-IDF	0.97	KNN with BoW KNN with Bigrams SVM with Bigrams	0.61	16
2	LR with BoW	0.98	KNN with Bigrams	0.66	30
3	LR with BoW	0.97	KNN with Bigrams	0.58	68
4	LR with TF-IDF	0.84	KNN with Bigrams	0.12	27
5	LR with BoW	0.93	KNN with Bigrams	0.63	28
6	LR with BoW KNN with TF-IDF	0.96	NB with TF-IDF	0.63	13
7	LR with BoW NB with BoW NB with Bigrams	0.87	NB with TF-IDF	0.70	12
8	LR with BoW	0.93	NB with TF-IDF	0.40	8
9	LR with BoW	1.00	KNN with Bigrams	0.67	14
10	NB with BoW NB with Bigrams KNN with TF-IDF	0.97	KNN with BoW	0.50	15
11	NB with BoW	0.85	KNN with Bigrams	0.09	21

4.8 DISCUSSION

As the result shown in Table 4.7, the most robust feature to be used for classifying GDP inspection findings is TF-IDF while Table 4.9 shown that the most robust classifier to be used for classifying GDP inspection findings is Logistic Regression. This may imply that combination of Logistic Regression with TF-IDF could be the model choice for this classification task. On contrary, the result shown otherwise in Table 4.6, the best performing model is the Logistic Regression model with Bag of Words feature extraction. This model achieved an accuracy of 0.92, precision of 0.93, recall of 0.92, and F1 score of 0.92 and also outperformed the baseline model (BoW + NB) in all the metrics.

BoW represents text as simple counts of word occurrences. This simplicity can be advantageous on small dataset, where more complex representations might capture noise. It helps in preventing overfitting in Logistic Regression because in overfitting, the learning method also learns from noise, which when included in the document representation, can leads to increased classification errors on new, unseen data.

TF-IDF introduces additional complexity by weighting terms based on their frequency across sentences. This added complexity could amplify the impact of rare terms, which might not be representative of the class but occur due to noise. Therefore, the term weighting in TF-IDF can lead to overfitting in small datasets by giving too much importance to rare words and creating more complex decision boundaries in Logistic Regression that not generalize well.

When comparing with NB, BoW creates a high-dimensional feature space by considering the frequency of all words in the vocabulary while logistic regression is more robust to correlated features than NB and it generally performs better on larger documents or datasets than NB (Daniel Jurafsky 2023). The dataset used in this study probably is considered as relatively larger dataset hence LR outperformed NB in this study.

The choice of feature significantly influences model performance, with Bag of Words proving particularly effective in this task when combined with logistic regression classifier, whereas Bigrams does not provide improvement over BoW and even led to worse performance with some classifiers which could not classifying chapter 4 and 11. This is probably due to some topic keywords being indicative alone, such as ‘aduan’ indicates chapter 6 of GDP guideline.

TF-IDF feature can be as effective as the BoW feature for the classification of Malay GDP inspection findings. However, it needs to be used with a suitable classifier to get better performance.

For each chapter's classification, analysis of the result from Table 4.22 revealed that Logistic Regression with BoW is the most consistent top performer across multiple chapters, whereas KNN with Bigrams often underperforms other models in this task. Therefore, KNN with Bigrams model is not suitable for GDP inspection finding classification. Naive Bayes performs well with BoW and Bigrams, showing it beneficial from N-grams features. Chapter 1, 4 and 11 have average f1-score of less than 0.80. It could be due to high similarity of frequent words as revealed in word cloud analysis.

In chapters with lower support, such as Chapter 4, 8 and 11, variability in performance is more pronounced, further confirming the need for sufficient data is required to boost the classification performance.

4.9 SUMMARY

The bag of Words feature was able to capture important information from the non-conformance finding for classification, and the Logistic Regression classifier was able to gain more information from this feature to make accurate predictions.

This concludes that the Logistic Regression classifier with Bag of Words feature extraction is the best performing model for Malay GDP inspection finding classification. This model achieved an accuracy of 0.92, precision of 0.93, recall of 0.92, and F1 score of 0.92 and outperformed the baseline combination (BoW + NB) in all metrics.

CHAPTER V

CONCLUSION AND FUTURE WORKS

5.1 RESEARCH SUMMARY

This study utilises machine learning to categorise inspection findings of Good Distribution Practice (GDP) in the Malay language. By examining feature extraction methods and machine learning algorithms, the study has revealed valuable insights into the potential for automating the analysis of GDP inspection reports. Additionally, it has laid the groundwork for future researchers to enhance the effectiveness and precision of GDP inspections by implementing advanced data science methodologies.

It encompasses a comprehensive investigation into feature extraction techniques and machine learning algorithms, aiming to identify the most suitable combination for analysing Good Distribution Practice Inspection Reports issued by NPRA inspectors. The study culminates in a comparative analysis of the performance of various feature extraction techniques and machine learning algorithms, shedding light on their effectiveness in classifying and interpreting GDP inspection findings. The study not only contributes to the field of data science but also holds implications for optimising regulatory compliance processes within the pharmaceutical industry.

5.2 OBJECTIVE ACHIEVEMENT

This study successfully accomplished the research objective of identifying suitable features and classifiers for the classification of Malay GDP inspection findings.

This study thoroughly investigates and compares various feature extraction techniques, including Bag of Words (BoW), TF-IDF, and Bigrams, in combination with classifiers such as Naïve Bayes (NB), Logistic Regression (LR), Support Vector

Machine (SVM), and k Nearest Neighbor (KNN). Through this exploration, the research identifies the most suitable combination of feature extraction and classification methods for accurately categorising Malay GDP inspection findings.

The study's second objective was to conduct a comprehensive comparative analysis of the performance of machine learning classifiers with different feature extraction methods. This comparative study provides valuable insights into the strengths and weaknesses of each combination, ultimately fulfilling the objective of performing a thorough evaluation of machine learning classifiers in the context of the Malay GDP inspection findings classification.

Therefore, this study has effectively achieved the research objectives by systematically identifying suitable features and classifiers, as well as conducting a rigorous comparative study to evaluate the performance of machine learning classifiers in the classification of Malay GDP inspection findings.

5.3 LIMITATIONS

The limitations of the study on the classification of Malay Language GDP Inspection through Text Mining include:

Data Availability and Generalisability: It is important to acknowledge that the findings of this study may be limited by the availability of datasets for Malay language GDP inspection. The study's classification models were developed based on the available data, which might not fully represent the diversity and complexity of the language. As a result, the models' robustness and the generalizability of the findings could be limited to Malay GDP inspection reports issued by NPRA’s inspectors. Adaptation to other languages or beyond the GDP domain might need further adaptation and validation.

Language-Specific Challenges: The distinctive morphological and syntactic features of the Malay language present notable challenges for text classification. Furthermore, the language's limited representation in research significantly impedes the availability of established methodologies and benchmarks for comparative analysis.

Inspector Variability: The variability in inspector writing styles and the complexity of non-conformance descriptions may impact the accuracy of the developed approach, potentially introducing biases or limitations in the automated analysis of GDP inspection reports.

Model Interpretability: The interpretability of the developed classification models, particularly those based on complex feature extraction techniques and machine learning algorithms, may be limited, potentially hindering the understanding of the decision-making process behind classification outcomes.

5.4 RESEARCH CONTRIBUTION

This study has proposed a Logistic Regression classifier with Bag of Words feature extraction as the best performing model for Malay GDP inspection finding classification. It demonstrates the effective application of text mining techniques to analyse and classify non-conformance findings in GDP inspection reports. This showcases the potential of automated methods in extracting valuable insights from inspection reports.

Besides that, this study identifies the potential for integrating the developed model into existing regulatory processes, providing an automated and objective method for classifying inspection findings. This could contribute to more efficient decision-making within regulatory bodies.

5.5 FUTURE WORK

This research project can be extended for future research in which some recommendations can be used to enhance this research. This section will discuss these recommendations, which can be illustrated as follows:

1. Leveraging domain knowledge: Develop a comprehensive lexicon of Malay-specific keywords and phrases related to GDP violations and good practices. Thereby utilise the lexicon to create n-gram features capturing important word combinations for improved classification accuracy.

2. Partner with GDP inspectors and domain experts to develop rule-based systems for identifying specific patterns and phrases indicative of different findings. Integrate these rules into the classification model for enhanced interpretability.

3. Implement a deep learning model, such as a CNN or RNN, to automatically learn features from the text data, which may improve performance and handle complex relationships in the language.

4. Explore using the proposed feature and classifier to classify Malay Good Manufacturing Practice inspection findings.

5. Explore usage of new features such as Doc2Vec, variants of TF-IDF as text representation for classifying the GDP inspection findings.

6. Experiment with stemming, lemmatisation, and part-of-speech tagging to enhance feature representation and model performance. Explore named entity recognition and dependency parsing for extracting additional structural and semantic information from the text.

7. Design and implement an ETL pipeline to extract relevant data from the inspection reports efficiently. This may involve parsing structured fields, including company name and date while extracting unstructured text findings for analysis.

8. Conduct sentiment analysis on the text to identify positive feedback and differentiate criticality of deficiencies mentioned in the reports. This can provide valuable insights for regulators.

9. Implement bias detection and mitigation techniques to ensure the text mining models are not biased towards specific industries, businesses, or types of findings. Employ diverse training data and consider ethical considerations regarding data privacy and security.

10. Predictive modelling: Develop predictive models to identify businesses at higher risk of non-compliance based on their inspection history and other relevant data. This can help regulators prioritise their resources and target their inspections more effectively.

Exploring these future works can be a valuable step for researchers and practitioners to enhance the effectiveness and impact of text mining for classifying Malay language GDP inspection findings. This can play a significant role in promoting regulatory oversight, ensuring compliance with good distribution practices, and ultimately providing quality medicinal products which are safe and efficacious to end users.

REFERENCES

Adhi, B. P., Saskiah, D. & Widodo, W. 2019. A Systematic Literature Review of Short Text Classification on Twitter. *KnE Social Sciences*: 625-635.

Ahmed, M. H., Tiun, S., Omar, N. & Sani, N. S. 2023. Short Text Clustering Algorithms, Application and Challenges: A Survey. *Applied Sciences* 13(1): 342.

Al-Saffar, A., Awang, S., Tao, H., Omar, N., Al-Saiagh, W. & Al-Bared, M. 2018. Malay sentiment analysis based on combined classification approaches and Senti-lexicon algorithm. *PLOS ONE* 13(3): 1-18.

Aldera, S., Emam, A., Al-Qurishi, M., Alrubaian, M. & Alothaim, A. 2021. Exploratory Data Analysis and Classification of a New Arabic Online Extremism Dataset. *IEEE Access* 9: 161613-161626.

Alshalabi, H. A., Sabrina, T. & Nazlia, O. 2017. A comparative study of the ensemble and base classifiers performance in Malay text categorization.

Christanti Mawardi, V., Susanto, N. & Santun Naga, D. 2018. Spelling Correction for Text Documents in Bahasa Indonesia Using Finite State Automata and Levinshtein Distance Method. *MATEC Web Conf.* 164: 01047.

Corpuz, R. S. A. 2021. ISO 9001:2015 Quality Management System Requirements and Audit Findings Classification Using Support Vector Machine and Long Short-Term Memory Neural Network: An Optimization Method. *Science and Technology* - Mindanao Journal of Science and Technology, Philippines [e-ISSN: 2449-3686] 19: 197-223.

Daniel Jurafsky, J. H. M. 2023. *Speech and Language Processing.*Draft.

Dogra, V., Verma, S., Kavita, Chatterjee, P., Shafi, J., Choi, J. & Ijaz, M. F. 2022. A Complete Process of Text Classification System Using State-of-the-Art NLP Models. *Computational Intelligence and Neuroscience* 2022: 1883698.

European Medicines Agency. 2007. *Good Manufacturing Practice: An analysis of regulatory inspection findings in the centralised procedure.*

Gandomi, A. & Haider, M. 2015. Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management* 35(2): 137-144.

García S, L. J., Herrera F. 2015. *Data Preprocessing in Data Mining.* Springer.

Hacohen-Kerner, Y., Miller, D. & Yigal, Y. 2020. The influence of preprocessing on text classification using a bag-of-words representation. *PLOS ONE* 15(5): e0232525.

Hacohen-Kerner, Y., Rosenfeld, A., Sabag, A. & Tzidkani, M. 2018. Topic-based Classification through Unigram Unmasking. *Procedia Computer Science* 126: 69-76.

Han J, K. M., Pei J. , 2012. *Data Mining Concepts and Techniques* 3rd edition Ed.Elsevier Inc.

Hassan, S. U., Ahamed, J. & Ahmad, K. 2022. Analytics of machine learning-based algorithms for text classification. *Sustainable Operations and Computers* 3: 238-248.

Jaafar, J., Indra, Z. & Zamin, N. 2016. A category classification algorithm for Indonesian and Malay news documents. *Jurnal Teknologi* 78(8-2): 121-132-132.

Joachims, T. 1998. Text categorization with Support Vector Machines: Learning with many relevant features. *Machine Learning: ECML-98*, pp.137-142.

Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L. & Brown, D. 2019. Text Classification Algorithms: A Survey. *Information* 10(4): 150.

Mccormick, K. & Sanders, J. H. 2022. Chapter 14 - Good distribution practice. In Mccormick, K. & Sanders, J. H. (ed.). *Quality (Second Edition)*, pp.327-344. Butterworth-Heinemann.

Malaysian Investment Development Authority. 2022. *Malaysia's Pharmaceutical Industry: A Fast Growing Force*. 16/1/2024.

Mikulic, M. 2023. Pharmaceutical market: worldwide revenue 2001-2022. https://www.statista.com/statistics/263102/pharmaceutical-market-worldwide-revenue-since-2001/ [3/1/2024].

Misra, P. & Yadav, A. 2019. Impact of Preprocessing Methods on Healthcare Predictions. *SSRN Electronic Journal*:

Mohammed, M. & Omar, N. 2020. Question classification based on Bloom's taxonomy cognitive domain using modified TF-IDF and word2vec. *PLOS ONE* 15(3): e0230442.

Nazratul Naziah Mohd, M., Rosmayati, M., Noor Maizura Mohamad, N. & Zulaiha Ali, O. 2021. Comparative Study of K-Nearest Neighbour and Naïve Bayes Performances on Malay Text Classification, Research Synergy Foundation. 1**:** 50-60.

National Pharmaceutical Regulatory Agency. 2018. *Guideline on Good Distribution Practice*

National Pharmaceutical Regulatory Agency. 2020. *Frequently Asked Questions (FAQs) : Good Distribution Practice (GDP).*

National Pharmaceutical Regulatory Agency. 2023. *Annual Report 2022.*

Palanivinayagam, A., El-Bayeh, C. Z. & Damaševičius, R. 2023. Twenty Years of Machine-Learning-Based Text Classification: A Systematic Review. *Algorithms* 16(5): 236.

Pintas, J. T., Fernandes, L. a. F. & Garcia, A. C. B. 2021. Feature selection methods for text classification: a systematic literature review. *Artificial Intelligence Review* 54(8): 6149-6200.

Pramanik, J., Samal, A. K., Sahoo, K. & Pani, D. S. 2019. Exploratory Data Analysis using Python. *International Journal of Innovative Technology and Exploring Engineering* 8: 4727-4735.

Rostam, N. a. P. & Malim, N. H. a. H. 2021. Text categorisation in Quran and Hadith: Overcoming the interrelation challenges using machine learning and term weighting. *Journal of King Saud University - Computer and Information Sciences* 33(6): 658-667.

Stoimenova, A., Kirilov, B. & Zaykova, K. 2019. Analysis of good distribution practice inspection deficiency data of pharmaceutical wholesalers in Bulgaria. *Pharmacia* 66: 85-89.

Sun, Y., Wong, A. K. C. & Kamel, M. S. 2009. Classification of Imbalanced Data: A Review. *International Journal of Pattern Recognition and Artificial Intelligence* 23(04): 687-719.

Tarnate, K. J. & Devaraj, M. 2019. Prediction of ISO 9001:2015 Audit Reports According to its Major Clauses using Recurrent Neural Networks. 8:

Tarnate, K. J. M., De Goma, J. C. & Devaraj, M. 2020. Overcoming the vanishing gradient problem of recurrent neural networks in the ISO 9001 quality management audit reports classification. *International Journal of Scientific and Technology Research* 9(3): 6683-6686-6686.

Tiun, S. 2017. Experiments on malay short text classification. *2017 6th International Conference on Electrical Engineering and Informatics (ICEEI),* pp.1-4.

Uche, C. B. N., Ekeocha, Z., Byrn, S. R. & Clase, K. L. 2021. Retrospective Study of Inspectors Competency in the Act of Writing GMP Inspection Report.

United States Food and Drug Administration. 2023. Inspections Citations Details. FDA Compliance Dashboards.[6/11/2023]

Wang, H., Hong, M. & Lau, R. Y. K. 2019. Utility-based feature selection for text classification. *Knowledge & Information Systems* 61(1): 197-226.

Wang, S. & Manning, C. 2012. Baselines and Bigrams: Simple, Good Sentiment and Topic Classification. pp.90-94.

APPENDIX A

SAMPLE OF DATASET

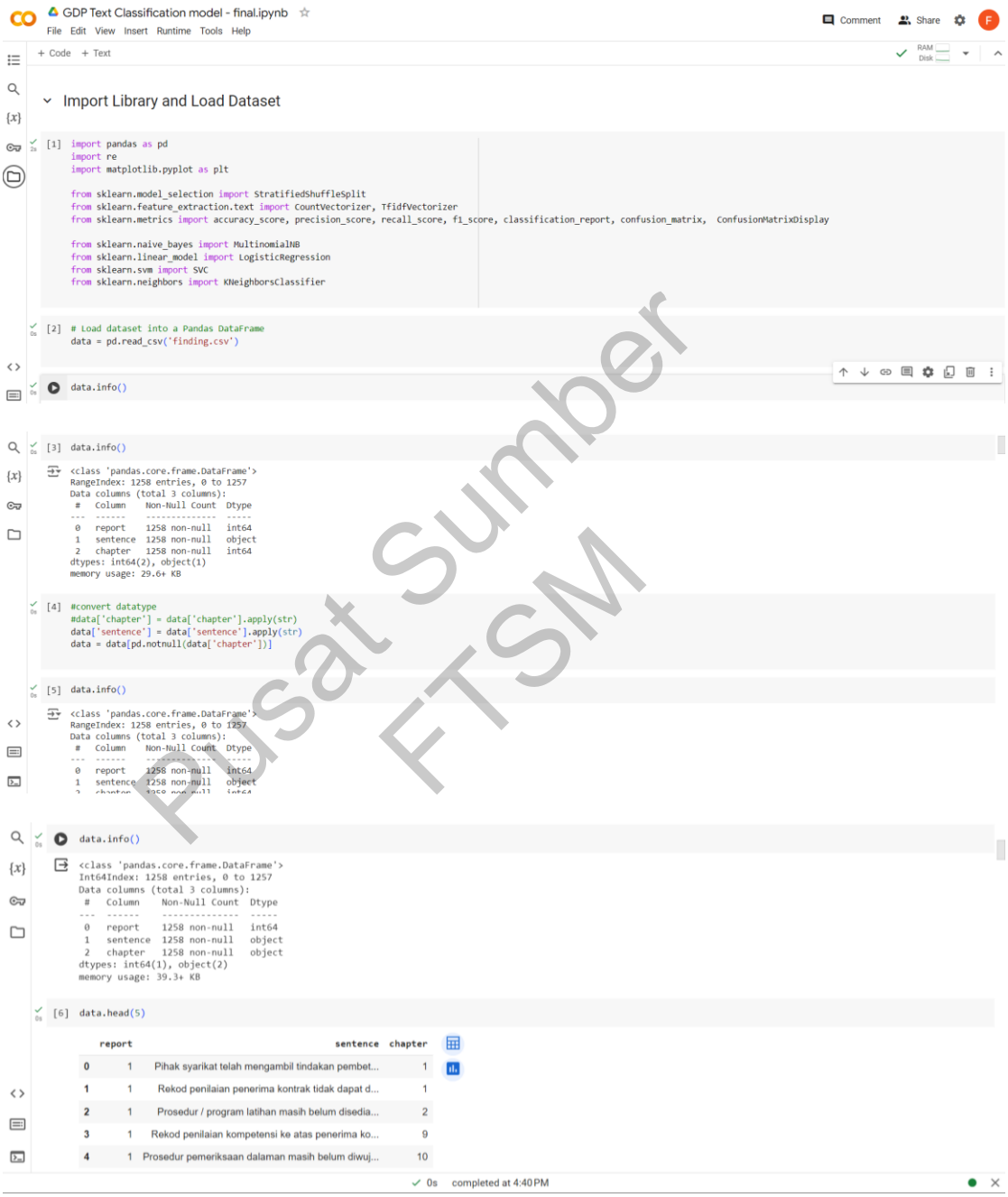
CHAPTER	INSPECTION FINDINGS
1	Pihak syarikat telah mengambil tindakan pembetulan terhadap penemuan-penemuan lepas. Walau bagaimanapun, terdapat penemuan yang masih berulang (rujuk Bab 1: 1.4; Bab 2: 2.3; Bab 9: 9.2, Bab 10: 10.1 & 10.3).
1	Syarikat masih belum mempunyai sistem untuk mengawal dan menilai penerima kontrak (rujuk para 9.2).
1	Pihak syarikat belum menyediakan sistem untuk memastikan produk / kosmetik dibekalkan oleh pembekal yang diluluskan dan diedarkan oleh entiti yang diluluskan.
1	Syarikat masih belum melaksanakan CAPA dengan menyeluruh hasil dari pemeriksaan yang lepas
1	Pemeriksaan mendapati kelemahan – kelemahan berkaitan penjejakan nombor kelompok produk yang diedarkan secara dokumentasi
2	Prosedur / program latihan masih belum disediakan oleh pihak syarikat. (penemuan berulang)
2	Pemeriksaan ambil maklum bahawa pihak pengurusan syarikat hanya terdiri daripada dua (2) personel sahaja. Walau bagaimanapun. deskripsi tugas bagi kedua-dua personel masih belum diwujudkan lagi sejak pemeriksaan lepas. (Nota: Penemuan berulang daripada pemeriksaan 2020)
2	Pihak syarikat tidak menyediakan prosedur latihan dan program latihan.
2	Tiada program latihan bagi tahun 2022 disediakan
2	Pihak syarikat didapati tidak menjalankan penilaian kefahaman ke atas personel yang dilatih.
3	Pihak syarikat belum menyediakan kawasan pengasingan yang sewajarnya bagi produk berdaftar dengan status: lulus, kuaratin, dipulang dan ditolak.
3	Pihak syarikat belum menyediakan kemudahan seperti palet atau sistem rak untuk aktiviti penstoran produk berdaftar.
3	Semakan ke atas Temperature Monitoring Log mendapati spesifikasi suhu dan kelembapan relatif yang perlu dipantau tidak ditetapkan.
3	Pihak syarikat telah menyediakan sebuah thermohygrometer (Brand: Brannan) bagi aktiviti pemantauan suhu dan kelembapan relatif bagi kawasan penstoran, namun thermohygrometer berkenaan tidak dikalibrasi.
3	Kesesuaian lokasi peralatan thermohygrometer dalam stor tidak dapat dikenal pasti oleh pihak syarikat.
4	Pihak syarikat belum mewujudkan keperluan / dokumen yang berkaitan dengan pelupusan produk.
4	Semakan dokumen SOP for Order for Order Receipt, Storage and Dispatch of Finished Product tidak menyatakan tatacara pengendalian produk berdaftar termasuk prosedur penerimaan produk dengan sewajarnya.
4	Pihak syarikat belum mewujudkan prosedur berkaitan pengendalian produk berdaftar yang dipulangkan.
4	Prosedur pengedaran produk kepada pelanggan yang dijalankan oleh pihak syarikat tidak memperincikan proses yang dijalankan di ibu pejabat dan rekod dan dokumen pengedaran (contohnya: consignment note syarikat kurier) produk tidak dapat dikemukakan semasa pemeriksaan.
4	Senarai pembekal yang diluluskan belum diwujudkan
4	Pihak syarikat tidak menyertakan sijil analisa produk untuk setiap kelompok produk yang diimport.

5	Prosedur pengendalian dan penyiasatan insiden penyimpangan daripada keadaan penstoran.
5	Prosedur operasi / penyelenggaraan / pembersihan / kawalan makhluk perosak melibatkan kenderaan belum diwujudkan.
5	Prosedur pengendalian dan penyiasatan insiden penyimpangan daripada keadaan penstoran belum diwujudkan.
5	Pihak syarikat memaklumkan bahawa pemandu syarikat / penghantaran pihak ketiga akan memaklumkan kepada pihak syarikat dan memulangkan semula produk sekiranya produk gagal dihantar pada hari yang sama atau dalam masa 24 jam. Walau bagaimanapun, prosedur pengendalian dan penyiasatan insiden penyimpangan ini masih belum diwujudkan
5	Semakan dokumen Custom Brokerage and Logistic Services Agreement (Schedule 2: Quality Agreement for Transportation) yang telah ditandatangani oleh wakil syarikat pada 23 April 2019 dan wakil syarikat ejen pengangkutan mendapati terdapat keperluan seperti berikut 'in the event that a curcumstance arises making it impossible to maintain the required temperature, Service Provider shall notify Sanofi within one hour of such event'. Walau bagaimanapun, tiada prosedur pengendalian dan penyiasatan insiden penyimpangan daripada keadaan penstoran dikemukakan termasuk perincian keperluan ejen pengangkutan memaklumkan insiden / penyimpangan kepada pemberi kontrak diwujudkan oleh pihak syarikat
6	Menurut SOP for Handling of Market Complaints, Supply Chain Manager telah dilantik untuk meluluskan dan menutupi kes-kes aduan yang diterima. Walau bagaimanapun, jawatan tersebut adalah tidak wujud dalam carta organisasi syarikat.
6	Skop prosedur termasuk pengendalian aduan bagi produk berdaftar dan kosmetik tidak dikemaskini sepertimana penjelasan dikemukakan pihak syarikat.
6	Maklumat berhubung personel yang bertanggungjawab tidak dinyatakan dalam prosedur aduan produk
6	Pihak syarikat belum melantik personel yang bertanggungjawab untuk menguruskan aktiviti pengendalian panggil balik produk berdaftar.
6	Fail / rekod aduan produk / kosmetik belum diwujudkan.
7	Pihak syarikat tidak menyediakan prosedur pengendalian panggil balik produk.
7	Pihak syarikat belum melantik personel yang bertanggungjawab untuk menguruskan aktiviti pengendalian panggil balik produk berdaftar.
7	Pihak syarikat belum melantik personel yang bertanggungjawab untuk menguruskan aktiviti pengendalian panggil balik produk berdaftar.
7	Tahap dan paras panggil balik produk tidak dinyatakan dalam dokumen Products / Cosmetics Recall (No. Dokumen: CHAPTER 7, Ver. 1; Tarikh Efektif: 01 Januari 2018).
7	Pihak syarikat ada mewujudkan Product Recalls Procedure (SOP/PRODUCT RECALLS/004, Tarikh Efektif: 1 Jun 2021) Walau bagaimanapun, prosedur ini tidak menyatakan jenis tahap dan paras panggil produk untuk proses panggil balik produk.
8	Syarikat masih belum menyediakan dokumen berkenaan pengendalian tiruan / substandard dengan sewajarnya.
8	Pihak syarikat belum menyediakan keperluan / dokumen yang berkaitan dengan aktiviti pengendalian produk substandard / tiruan.
8	Dokumen Handling of Suspect Product Counterfeits (EA-GO-006, Rev. B; Tarikh: 30 Oct 2018) ada disediakan. Walau bagaimanapun, ia didapati tidak disemak semula oleh pihak syarikat mengikut keperluan semakan berkala sekurang-kurangnya setiap tiga tahun yang ditetapkan dalam dokumen Document Change Management (GQP-04-02, Rev. I; Tarikh: 30 Sep 2021).

8	Prosedur bertulis pengendalian produk substandard/ tiruan masih belum disediakan sejak pemeriksaan terakhir termasuk keperluan merekod tindakan yang akan diambil oleh pihak syarikat serta pelaporan kepada agensi yang bertanggungjawab sekiranya terdapat insiden tersebut
8	Pihak syarikat telah menyediakan dokumen bertajuk Handling of Counterfeit Medical Device (No. Dokumen: PGW/SOP/32/3, Traikh Efektif: 1 Januari 2021, Rev. No.: 3). Walau bagaimanpun, dokumen tersebut lebih khusus untuk pengendalian produk peranti perubatan tiruan.
9	Dokumen kontrak antara pihak syarikat dan 3PL belum disediakan.
9	Pihak syarikat belum menyediakan prosedur penilaian vendor.
9	Pihak syarikat tidak dapat mengemukakan prosedur atau apa-apa dokumen berkenaan tatacara menjalankan Outsourced Vendor Evaluation termasuk frekuensi, tindakan susulan penilaian seperti pembentangan dalam 'management meeting' dan sebagainya.
9	Pihak syarikat ada mewujudkan kontrak perkhidmatan aktiviti kawalan makhluk perosak dengan Rentokil Initial Sdn. Bhd. Walau bagaimanapun, pihak syarikat tidak mewujudkan kontrak bertulis dengan pembekal dan juga pengedar.
9	Penilaian kompetensi ke atas penerima kontrak masih belum dijalankan. Pihak syarikat juga tidak dapat mengemukakan sebarang prosedur ataupun pelan untuk menjalankan penilaian kompetensi tersebut.
10	Prosedur pemeriksaan dalaman masih belum diwujudkan. (penemuan berulang)
10	Pemeriksaan dalaman yang dijalankan masih terhad ke atas semakan stok sahaja namun tiada sebarang laporan dikeluarkan. (penemuan berulang)
10	Pihak syarikat belum pernah menjalankan pemeriksaan dalaman.
10	Pihak syarikat telah menyediakan dokumen Internal Audit Procedures (No. Dokumen: 013, Date Adopted: 1 September 2015, Revision Date: 1 Januari 2021). Walau bagaimanapun, dokumen tersebut tidak memperincikan tatacara pengendalian audit, keperluan personel yang menjalankan audit bersifat 'independent' dan frekuensi audit.
10	Pemeriksaan dalaman terakhir yang dijalankan adalah pada 7 April 2022 dan laporan telah disediakan menggunakan format Annual Internal Audit Report / Checklist. Walau bagaimanapun, skop pemeriksaan tidak merangkumi keseluruhan aktiviti syarikat yang tertakluk di bawah keperluan Amalan Pengedaran Baik.
11	Syarikat telah menyediakan beberapa prosedur dan dokumen berkaitan AEB. Namun, terdapat dokumen dan prosedur bertulis tidak dilengkapkan dengan nombor dokumen, tarikh kuat kuasa dan tanda tangan pelulus.Contohnya carta organisasi syarikat, Order And Delivery, Training SOP dan Prosedur Produck Recall SOP.
11	Kesan penggunaan 'liquid paper' ditemui pada dokumen RDM (Title: RDM, Location: BR).
11	Rekod penggunaan hologram belum diwujudkan oleh pihak syarikat.
11	Dokumen prosedur telah disediakan dengan format yang selaras, dilengkapi dengan tajuk, nombor dokumen, tarikh dan kelulusan sewajarnya. Kesemua dokumen prosedur kecuali prosedur pengendalian aduan telah dikemas kini pada 8 Disember 2021. Dokumen prosedur terkini yang telah dikemas kini pada 8 Disember 2021 disimpan oleh personel Senior Regulatory Affairs Manager. Namun, dokumen prosedur terkini tidak dikongsikan kepada personel yang menjalankan aktiviti berkaitan dengan setiap prosedur.
11	Pemeriksaan mengambil maklum pihak syarikat ada membuat aktiviti penampalan hologram pada produk yang diedarkan. Pihak syarikat juga ada membuat perekodan aktiviti penggunaan hologram dengan jelas. Aktiviti penampalan hologram akan dibuat sebelum produk tersebut diedarkan. Walau bagaimanapun, semakan mendapati tiada prosedur tatacara pengambilan dan pengurusan hologram diwujudkan.

APPENDIX B

GOOGLE COLAB



Preprocessing

```
[7] #load Malaya
    !pip install malaya
    import malaya
```

```
[8] #malay stopword removal
    malay_stopwords = malaya.text.function.get_stopwords()

    data["sentence"] = data["sentence"].apply(lambda x: ' '.join([word for word in x.split() if word.lower() not in malay_stopwords]))
```

```
[9] data.head()
```

	report	sentence	chapter
0	1	syarikat mengambil tindakan pembetulan penemua...	1
1	1	Rekod penilaian penerima kontrak dikemukakan p...	1
2	1	Prosedur / program latihan disediakan syarikat...	2
3	1	Rekod penilaian kompetensi penerima kontrak sy...	9
4	1	Prosedur pemeriksaan dalaman diwujudkan. (pene...	10

✓ 0s completed at 4:40 PM

```
# Lowercase, punctuation, digit removal

data['sentence'] = data['sentence'].str.lower()  # Lowercasing
data['sentence'] = data['sentence'].str.replace(r'[^\w\s]', '')  # Remove punctuation, special symbols, and non-word characters.
data['sentence'] = data['sentence'].apply(lambda x: re.sub(r'\d', '', x)) # Remove digit
```

Exploratory Data Analysis

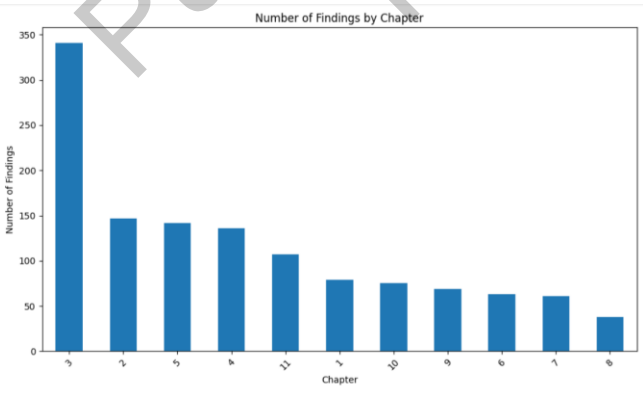
```
[11] data.head()
```

	report	sentence	chapter
0	1	syarikat mengambil tindakan pembetulan penemua...	1
1	1	rekod penilaian penerima kontrak dikemukakan p...	1
2	1	prosedur program latihan disediakan syarikat ...	2
3	1	rekod penilaian kompetensi penerima kontrak sy...	9
4	1	prosedur pemeriksaan dalaman diwujudkan penemu...	10

```
chapter_counts = data['chapter'].value_counts()

plt.figure(figsize=(10, 6))
chapter_counts.plot(kind='bar')
plt.title('Number of Findings by Chapter')
plt.xlabel('Chapter')
plt.ylabel('Number of Findings')
plt.xticks(rotation=45)  # Rotate x-axis labels for readability

# Show the plot
plt.tight_layout()
plt.show()
```



```
chapter_counts.columns = ['Chapter', 'Number of Findings']

# Display the table
print(chapter_counts)
```

```
3     341
2     147
5     142
4     136
11    107
1      79
10     75
9      69
6      63
7      61
8      38
Name: chapter, dtype: int64
```

Word Cloud Analysis

```
class_texts = {}
for chapter in data['chapter'].unique():
    class_texts[chapter] = " ".join(data[data['chapter'] == chapter]['sentence'])
```

```
from wordcloud import WordCloud

def plot_word_cloud(text, title):
    wordcloud = WordCloud(width=800, height=400, background_color='white').generate(text)
    plt.figure(figsize=(6, 3))
    plt.imshow(wordcloud, interpolation='bilinear')
    plt.title(title)
    plt.axis('off')
    plt.show()

for chapter, text in class_texts.items():
    plot_word_cloud(text, f'Word Cloud for Chapter {chapter}')
```

GDP Text Classification model - final.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

```
#View the total number of unique words in the vocabulary created by Bag of Word

BOW = CountVectorizer()
BOW.fit(data['sentence'])
bow_vector = BOW.transform(data['sentence'])
print(len(BOW.vocabulary_))

#shape of the BoW matrix
print(bow_vector.shape)
```

```
2432
(1258, 2432)
```

Feature extraction and classifier Evaluation

```
# Split data into features (X) and labels (y)
X = data['sentence']
y = data['chapter']

# Define the Stratified Shuffle Split
stratified_splitter = StratifiedShuffleSplit(n_splits=1, test_size=0.2, random_state=42)

# Split the data into training and testing sets using stratified sampling
for train_index, test_index in stratified_splitter.split(X, y):
    X_train, X_test = X[train_index], X[test_index]
    y_train, y_test = y[train_index], y[test_index]
```

```
# Check class distribution in the original dataset
print("Original Data Class Distribution:")
print(y.value_counts())

# Check class distribution in the training set
print("\nTraining Set Class Distribution:")
print(y_train.value_counts())

# Check class distribution in the testing set
print("\nTesting Set Class Distribution:")
print(y_test.value_counts())
```

```
Original Data Class Distribution:
3     341
2     147
5     142
4     136
11    107
1      79
10     75
9      69
6      63
7      61
8      38
Name: chapter, dtype: int64
```

```
Training Set Class Distribution:
3     273
2     117
5     114
4     109
11     86
1      63
10     60
9      55
6      50
7      49
8      30
Name: chapter, dtype: int64

Testing Set Class Distribution:
3     68
2     30
5     28
4     27
11    21
1     16
10    15
9     14
6     13
7     12
8      8
Name: chapter, dtype: int64
```

```python
# Feature Extraction
vectorizers = {
    'BoW': CountVectorizer(),
    'TF-IDF': TfidfVectorizer(),
    'Bigrams': CountVectorizer(ngram_range=(2, 2)),
    '3-grams': CountVectorizer(ngram_range=(3, 3))

}


# Classifier Selection
classifiers = {
    'Naive Bayes': MultinomialNB(),
    'Logistic Regression': LogisticRegression(),
    'SVM': SVC(),
    'K-NN': KNeighborsClassifier(n_neighbors=11)
}
```

```python
# Compare different combinations
for vectorizer_name, vectorizer in vectorizers.items():
    X_train_features = vectorizer.fit_transform(X_train)
    X_test_features = vectorizer.transform(X_test)

    for classifier_name, classifier in classifiers.items():
        classifier.fit(X_train_features, y_train)
        y_pred = classifier.predict(X_test_features)

        # Model Evaluation
        accuracy = accuracy_score(y_test, y_pred)
        report = classification_report(y_test, y_pred)

        # Confusion Matrix
        cm = confusion_matrix(y_test, y_pred, labels=sorted(data['chapter'].unique()))
        disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=classifier.classes_)
        disp.plot(cmap=plt.cm.Blues)
        plt.title(f"Confusion Matrix: {vectorizer_name} + {classifier_name}")
        plt.show()
        #print(f"Confusion Matrix:\n{cm}")


        print(f"Feature Extraction: {vectorizer_name}, Classifier: {classifier_name}")
        print("Accuracy:", accuracy)
        print("F1 score:", f1_score(y_test, y_pred, average='macro'))
        print("Classification Report:\n", report)
        print()
```

```
Feature Extraction: 3-grams, Classifier: Naive Bayes
Accuracy: 0.7619047619047619
F1 score: 0.7526754884568898
Classification Report:
              precision    recall  f1-score   support

           1       0.89      0.50      0.64        16
           2       0.81      0.83      0.82        30
           3       0.68      0.94      0.79        68
           4       0.73      0.59      0.65        27
           5       0.88      0.75      0.81        28
           6       0.75      0.69      0.72        13
           7       0.89      0.67      0.76        12
           8       0.78      0.88      0.82         8
           9       0.92      0.79      0.85        14
          10       0.85      0.73      0.79        15
          11       0.71      0.57      0.63        21

    accuracy                           0.76       252
   macro avg       0.81      0.72      0.75       252
weighted avg       0.78      0.76      0.76       252


Feature Extraction: 3-grams, Classifier: Logistic Regression
Accuracy: 0.6746031746031746
F1 score: 0.7060796659480869
Classification Report:
              precision    recall  f1-score   support

           1       1.00      0.50      0.67        16
           2       0.82      0.60      0.69        30
           3       0.48      1.00      0.65        68
           4       0.83      0.37      0.51        27
           5       1.00      0.57      0.73        28
           6       0.89      0.62      0.73        13
           7       1.00      0.58      0.74        12
           8       1.00      0.75      0.86         8
           9       1.00      0.71      0.83        14
          10       1.00      0.67      0.80        15
          11       0.82      0.43      0.56        21

    accuracy                           0.67       252
   macro avg       0.89      0.62      0.71       252
weighted avg       0.80      0.67      0.68       252
```

```
Feature Extraction: 3-grams, Classifier: SVM
Accuracy: 0.503968253968254
F1 score: 0.5203974203374503
Classification Report:
              precision    recall  f1-score   support

           1       1.00      0.44      0.61        16
           2       1.00      0.17      0.29        30
           3       0.35      1.00      0.52        68
           4       1.00      0.04      0.07        27
           5       1.00      0.43      0.60        28
           6       1.00      0.38      0.56        13
           7       1.00      0.50      0.67        12
           8       1.00      0.75      0.86         8
           9       1.00      0.50      0.67        14
          10       1.00      0.40      0.57        15
          11       1.00      0.19      0.32        21

    accuracy                           0.50       252
   macro avg       0.94      0.44      0.52       252
weighted avg       0.83      0.50      0.47       252


Feature Extraction: 3-grams, Classifier: K-NN
Accuracy: 0.4126984126984127
F1 score: 0.3560368883868091
Classification Report:
              precision    recall  f1-score   support

           1       1.00      0.38      0.55        16
           2       0.31      0.33      0.32        30
           3       0.33      0.93      0.49        68
           4       0.00      0.00      0.00        27
           5       1.00      0.32      0.49        28
           6       1.00      0.15      0.27        13
           7       1.00      0.17      0.29        12
           8       1.00      0.62      0.77         8
           9       1.00      0.14      0.25        14
          10       1.00      0.33      0.50        15
          11       0.00      0.00      0.00        21

    accuracy                           0.41       252
   macro avg       0.70      0.31      0.36       252
weighted avg       0.55      0.41      0.35       252
```

Save the model

```
# Assuming X and y represent entire dataset
bow_vectorizer = CountVectorizer()
X_train_bow = bow_vectorizer.fit_transform(X_train)

lr_classifier = LogisticRegression()
lr_classifier.fit(X_train_bow,y_train)
```

```
LogisticRegression()
```

```
[19] import joblib

# Save the vectorizer and classifier to files
joblib.dump(bow_vectorizer, 'bow_vectorizer.pkl')
joblib.dump(lr_classifier, 'lr_classifier.pkl')
```

```
['lr_classifier.pkl']
```

Apply on new data

```
[20] import joblib

# Load the vectorizer and classifier
bow_vectorizer = joblib.load('bow_vectorizer.pkl')
lr_classifier = joblib.load('lr_classifier.pkl')
```

```
# Assuming 'new_data' is a DataFrame with a 'sentence' column

# Load dataset into a Pandas DataFrame
new_data = pd.read_csv('new_finding.csv')


new_data.head(10)
```

	report_no		sentence	chapter
0	NaN	1.2\tPihak syarikat didapati tidak menjalankan...		NaN
1	NaN	1.4\tPihak syarikat belum menyediakan sebarang...		NaN
2	NaN	2.5	Pihak syarikat tidak dapat mengemukaka...	NaN
3	NaN	2.6	Aktiviti penilaian kefahaman tidak dij...	NaN
4	NaN	2.7	Pihak syarikat telah mengemukakan doku...	NaN
5	NaN	3.9	Pihak syarikat telah melantik Midah Pe...	NaN
6	NaN	3.13	Pihak syarikat telah menyediakan 3 bua...	NaN
7	NaN	3.14	Kesesuaian lokasi peralatan termohygrome...	NaN
8	NaN	3.15	Aktiviti kajian pemetaan suhu ke atas ka...	NaN
9	NaN	Jadual penyelenggaraan bagi peralatan penstora...		NaN

```
[22] malay_stopwords = malaya.text.function.get_stopwords()

     new_data["sentence"] = new_data["sentence"].apply(lambda x: ' '.join([word for word in x.split() if word.lower() not in malay_stopwords]))
     new_data['sentence'] = new_data['sentence'].str.lower()
     new_data['sentence'] = new_data['sentence'].str.replace(r'[^\w\s]', '')
     new_data['sentence'] = new_data['sentence'].apply(lambda x: re.sub(r'\d', '', x))
```

```
new_data.head(10)
```

	report_no	sentence	chapter
0	NaN	syarikat menjalankan tindakan pembetulan sewa...	NaN
1	NaN	syarikat menyediakan sebarang sistem menilai ...	NaN
2	NaN	syarikat mengemukakan sebarang bukti personel...	NaN
3	NaN	aktiviti penilaian kefahaman dijalankan perso...	NaN
4	NaN	syarikat mengemukakan dokumen kad pesakit hos...	NaN
5	NaN	syarikat melantik midah pest control menjalan...	NaN
6	NaN	syarikat menyediakan buah thermohygrometer m...	NaN
7	NaN	kesesuaian lokasi peralatan termohygrometer s...	NaN
8	NaN	aktiviti kajian pemetaan suhu kawasan stor di...	NaN
9	NaN	jadual penyelenggaraan peralatan penstoran dik...	NaN

```
[24] X_new = new_data['sentence']

     X_new_bow = bow_vectorizer.transform(X_new)
```

```
# Predict using the loaded LR classifier
y_new_pred = lr_classifier.predict(X_new_bow)

# y_new_pred now contains the predicted labels for the new data
```

```
[26] print("Predicted Labels for New Data:")
     print(y_new_pred)

     Predicted Labels for New Data:
     ['3' '1' '2' '2' '2' '3' '3' '3' '3' '3']
```

```
[27] new_data = ["Pihak syarikat didapati tidak mewujudkan carta organisasi", "audit dalaman tidak dijalankan dengan sewajarnya."]

     # Preprocess and vectorize the new data using the loaded BoW vectorizer
     X_new_bow = bow_vectorizer.transform(new_data)

     # Step 4: Make Predictions on New Data
     y_new_pred = lr_classifier.predict(X_new_bow)

     # Print the predictions for the new data
     print("Predictions for the new data:")
     print(y_new_pred)

     Predictions for the new data:
     ['2' '10']
```

```
[48] import pandas as pd
     import re
     import matplotlib.pyplot as plt
     import numpy as np

     from sklearn.model_selection import StratifiedShuffleSplit
     from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, classification_report, confusion_matrix, ConfusionMatrixDisplay

     from sklearn.naive_bayes import GaussianNB
     from sklearn.linear_model import LogisticRegression
     from sklearn.svm import SVC
     from sklearn.neighbors import KNeighborsClassifier

     from gensim.models import Word2Vec
```

Word Embedding (Word2Vec)

```
#load Malaya
!pip install malaya
import malaya
```

Show hidden output

```
[54] # Load the data
     data = pd.read_csv('finding.csv')
     data['sentence'] = data['sentence'].apply(str)
     data = data[pd.notnull(data['chapter'])]

     # Define stopwords
     malay_stopwords = malaya.text.function.get_stopwords()

     # Preprocess the text
     data["sentence"] = data["sentence"].apply(lambda x: ' '.join([word for word in x.split() if word.lower() not in malay_stopwords]))
     data['sentence'] = data['sentence'].str.lower()
     data['sentence'] = data['sentence'].str.replace(r'[^\w\s]', '', regex=True)
     data['sentence'] = data['sentence'].apply(lambda x: re.sub(r'\d', '', x))
```

```python
# Tokenize sentences
sentences = [sentence.split() for sentence in data['sentence']]

# Train Word2Vec model
word2vec_model = Word2Vec(sentences, vector_size=100, window=5, min_count=1, workers=4)

# Function to get sentence vector by averaging word vectors
def get_sentence_vector(sentence, model):
    words = sentence.split()
    vector = np.mean([model.wv[word] for word in words if word in model.wv], axis=0)
    return vector

# Apply the function to get vectors for all sentences
data['vector'] = data['sentence'].apply(lambda x: get_sentence_vector(x, word2vec_model))

# Filter out sentences that result in NaN vectors
data = data[~data['vector'].isna()]

# Split the data into train and test sets
X = np.array(data['vector'].tolist())
y = data['chapter']

sss = StratifiedShuffleSplit(n_splits=1, test_size=0.2, random_state=0)
for train_index, test_index in sss.split(X, y):
    X_train, X_test = X[train_index], X[test_index]
    y_train, y_test = y[train_index], y[test_index]
```

```python
# Initialize classifiers
classifiers = {
    "Gaussian Naive Bayes": GaussianNB(),
    "Logistic Regression": LogisticRegression(max_iter=1000),
    "Support Vector Machine": SVC(),
    "K-Nearest Neighbors": KNeighborsClassifier()
}

# Function to evaluate and print results
def evaluate_classifier(clf, X_train, y_train, X_test, y_test):
    clf.fit(X_train, y_train)
    y_pred = clf.predict(X_test)
    print(f"Classifier: {clf.__class__.__name__}")
    print(f"Accuracy: {accuracy_score(y_test, y_pred)}")
    print(f"Precision: {precision_score(y_test, y_pred, average='macro')}")
    print(f"Recall: {recall_score(y_test, y_pred, average='macro')}")
    print(f"F1 Score: {f1_score(y_test, y_pred, average='macro')}")
    print(classification_report(y_test, y_pred))
    cm = confusion_matrix(y_test, y_pred)
    ConfusionMatrixDisplay(cm).plot()
    plt.show()

# Evaluate each classifier
for name, clf in classifiers.items():
    evaluate_classifier(clf, X_train, y_train, X_test, y_test)
```

```
Classifier: GaussianNB
Accuracy: 0.15079365079365079
Precision: 0.13161177446891734
Recall: 0.17231640908111495
F1 Score: 0.10891343783126915
              precision    recall  f1-score   support

           1       0.07      0.38      0.11        16
           2       0.00      0.00      0.00        30
           3       0.48      0.19      0.27        68
           4       0.00      0.00      0.00        27
           5       0.33      0.21      0.26        28
           6       0.25      0.08      0.12        13
           7       0.00      0.00      0.00        12
           8       0.08      0.50      0.14         8
           9       0.10      0.07      0.08        14
          10       0.13      0.47      0.21        15
          11       0.00      0.00      0.00        21

    accuracy                           0.15       252
   macro avg       0.13      0.17      0.11       252
weighted avg       0.20      0.15      0.14       252
```

```
[54] Classifier: LogisticRegression
     Accuracy: 0.2698412698412698
     Precision: 0.025129342202512936
     Recall: 0.09090909090909091
     F1 Score: 0.039374638100752 75
                   precision    recall  f1-score   support

                1       0.00      0.00      0.00        16
                2       0.00      0.00      0.00        30
                3       0.28      1.00      0.43        68
                4       0.00      0.00      0.00        27
                5       0.00      0.00      0.00        28
                6       0.00      0.00      0.00        13
                7       0.00      0.00      0.00        12
                8       0.00      0.00      0.00         8
                9       0.00      0.00      0.00        14
               10       0.00      0.00      0.00        15
               11       0.00      0.00      0.00        21

        accuracy                           0.27       252
       macro avg       0.03      0.09      0.04       252
    weighted avg       0.07      0.27      0.12       252
```