

## CONTENTS

|                                                          |           |
|----------------------------------------------------------|-----------|
| <b>Approval Page</b>                                     | <b>2</b>  |
| <b>Halaman Pengesahan</b>                                | <b>3</b>  |
| <b>Plagiarism Statement</b>                              | <b>4</b>  |
| <b>Foreword</b>                                          | <b>5</b>  |
| <b>Contents</b>                                          | <b>6</b>  |
| <b>List of Tables</b>                                    | <b>8</b>  |
| <b>List of Figures</b>                                   | <b>9</b>  |
| <b>Abstract</b>                                          | <b>10</b> |
| <b>Intisari</b>                                          | <b>11</b> |
| <b>1. Introduction</b>                                   | <b>12</b> |
| 1.1 Background                                           | 12        |
| 1.2 Research Problem                                     | 13        |
| 1.3 Research Scope                                       | 13        |
| 1.4 Research Objective                                   | 14        |
| 1.5 Research Advantages                                  | 14        |
| 1.7 Research Methodology                                 | 15        |
| 1.8 Thesis Organisation                                  | 15        |
| <b>2. Literature Review</b>                              | <b>16</b> |
| <b>3. Basic Theories</b>                                 | <b>20</b> |
| 3.1 Detection on Cyberattacks                            | 20        |
| 3.2 Apache Spark Technologies                            | 21        |
| 3.3 KDD CUP 99 Dataset                                   | 23        |
| 3.4 Data Preprocessing                                   | 24        |
| 3.4.1 Taking care of Missing Values and Categorical Data | 25        |
| 3.4.2 Normalisation and Standardisation                  | 25        |
| 3.5 Machine Learning Algorithms for Detection            | 27        |
| 3.6 Model Evaluation with Confusion Matrix               | 28        |
| <b>4. Analysis and Design</b>                            | <b>30</b> |
| 4.1 Research Description                                 | 30        |
| 4.2 Research Workflow                                    | 31        |

|                                                               |           |
|---------------------------------------------------------------|-----------|
| 4.3 Dataset Preprocessing . . . . .                           | 31        |
| 4.4 Application and Model Design and Implementation . . . . . | 35        |
| 4.5 Model Evaluation Plan . . . . .                           | 40        |
| <b>5. Implementation . . . . .</b>                            | <b>42</b> |
| 5.1 Tools and Materials . . . . .                             | 42        |
| 5.2 Visualising the Dataset . . . . .                         | 43        |
| 5.3 Import Libraries . . . . .                                | 44        |
| 5.4 Extract Data . . . . .                                    | 45        |
| 5.5 Dataset Preprocessing . . . . .                           | 46        |
| 5.5.1 Taking care of Missing Values . . . . .                 | 46        |
| 5.5.2 <i>Normalisation</i> . . . . .                          | 46        |
| 5.5.3 Z-Score Transformation . . . . .                        | 47        |
| 5.6 Test and Train Data . . . . .                             | 48        |
| 5.7 Implementation of the SVM Model . . . . .                 | 48        |
| 5.8 Implementation of the Decision Tree Model . . . . .       | 49        |
| 5.9 K-Fold Cross Validation . . . . .                         | 50        |
| 5.10 Implementation Without PySpark . . . . .                 | 50        |
| <b>6. Evaluation and Results . . . . .</b>                    | <b>51</b> |
| 6.1 Extract Data . . . . .                                    | 51        |
| 6.2 Evaluation and Result SVM Model . . . . .                 | 52        |
| 6.2.1 Precision, Recall and F1-Score Result . . . . .         | 53        |
| 6.3 Evaluation and Result Decision Tree Model . . . . .       | 54        |
| 6.3.1 Precision, Recall and F1-Score Result . . . . .         | 55        |
| 6.4 Result of Application without PySpark . . . . .           | 56        |
| 6.4.1 Precision, Recall and F1-Score Result . . . . .         | 58        |
| 6.5 Comparison of Results . . . . .                           | 59        |
| <b>7. Conclusion and Suggestion . . . . .</b>                 | <b>61</b> |
| 7.1 Conclusion . . . . .                                      | 61        |
| 7.2 Future Work . . . . .                                     | 62        |
| <b>References . . . . .</b>                                   | <b>63</b> |

## LIST OF TABLES

|                                                                                |    |
|--------------------------------------------------------------------------------|----|
| Table 2.1 : Comparison table of different research projects using Spark. . . . | 18 |
| Table 3.1 : Example of feature description from dataset . . . . .              | 23 |
| Table 3.2 : Confusion Matrix . . . . .                                         | 28 |
| Table 4.1 : Initial table of most_frequent method used in dataset. . . . .     | 32 |
| Table 4.2 : Result table of most_frequent method used in dataset. . . . .      | 32 |
| Table 4.3 : Example of First Normal Form used in dataset. . . . .              | 33 |
| Table 4.4 : Proposed Model Evaluation. . . . .                                 | 41 |
| Table 6.1 : Result Confusion Matrix of SVM Model . . . . .                     | 54 |
| Table 6.2 : Result Confusion Matrix of Decision Tree Model . . . . .           | 56 |
| Table 6.3 : Confusion Matrix of SVM Model without PySpark . . . . .            | 58 |
| Table 6.4 : Confusion Matrix of Decision Tree Model without PySpark . . . . .  | 58 |
| Table 6.5 : Average of Results using k-fold of Comparison . . . . .            | 59 |

## LIST OF FIGURES

|                                                                                                     |    |
|-----------------------------------------------------------------------------------------------------|----|
| Figure 3.1 : Example of dataframe used to create a dataset in PySpark . . . . .                     | 22 |
| Figure 3.2 : Example of Confusion Matrix used in the iris dataset . . . . .                         | 29 |
| Figure 4.1 : Research Workflow. . . . .                                                             | 33 |
| Figure 4.2 : Z-score conversion used . . . . .                                                      | 34 |
| Figure 4.3 : Data flow diagram of SparkContext . . . . .                                            | 37 |
| Figure 4.4 : Decision Tree Model . . . . .                                                          | 39 |
| Figure 4.5 : Diagram of K-fold cross validation result analysis . . . . .                           | 40 |
| Figure 5.1 : Applications in Jupyter Notebook . . . . .                                             | 43 |
| Figure 5.2 : Dataset input in DBeaver . . . . .                                                     | 44 |
| Figure 5.3 : Importing basic <i>machine learning</i> libraries in the application . . . . .         | 44 |
| Figure 5.4 : Importing libraries from <i>PySpark</i> MLlib . . . . .                                | 45 |
| Figure 5.5 : Importing and extracting the dataset information . . . . .                             | 46 |
| Figure 5.6 : Initiating the <i>normalisation</i> process . . . . .                                  | 47 |
| Figure 5.7 : Assign categories in feature ' <i>Normal or Attack</i> ' to <i>normalise</i> . . . . . | 47 |
| Figure 5.8 : Attempting z-score conversion on 1 feature . . . . .                                   | 47 |
| Figure 5.9 : z-score conversion performed in the first 10 features . . . . .                        | 48 |
| Figure 5.10 : Implementation to split <i>data</i> into test and train . . . . .                     | 48 |
| Figure 5.11 : Code Implementation of the SVM Model . . . . .                                        | 49 |
| Figure 5.12 : Code Implementation of the Decision Tree Model . . . . .                              | 50 |
| Figure 5.13 : Code Implementation of the Cross Validation . . . . .                                 | 50 |
| Figure 6.1 : Dataset visualised in DBeaver . . . . .                                                | 52 |
| Figure 6.2 : Dataset import results in the application . . . . .                                    | 52 |
| Figure 6.3 : Confusion Matrix result for SVM model . . . . .                                        | 53 |
| Figure 6.4 : Confusion Matrix result for the Decision Tree model . . . . .                          | 55 |
| Figure 6.5 : Confusion Matrix of SVM Model without PySpark . . . . .                                | 56 |
| Figure 6.6 : Confusion Matrix of Decision Tree Model without PySpark . . . . .                      | 57 |