

أسماء العتيبي

Automatic Detection of Arabic Spam Messages Using Machine Learning Techniques Final.docx

 Midocean University

Document Details

Submission ID

trn:oid::2945:406455705

Submission Date

Aug 8, 2026, 9:40 PM GMT+3

Download Date

Aug 11, 2026, 12:17 PM GMT+3

File Name

Automatic Detection of Arabic Spam Messages Using Machine Learning Techniques Final.docx

File Size

3.9 MB

70 Pages

11,998 Words

70,162 Characters

19% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text

Match Groups

-  **219 Not Cited or Quoted 18%**
Matches with neither in-text citation nor quotation marks
-  **9 Missing Quotations 1%**
Matches that are still very similar to source material
-  **0 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
-  **0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- 10%  Internet sources
- 10%  Publications
- 15%  Submitted works (Student Papers)

Integrity Flags

1 Integrity Flag for Review

-  **Hidden Text**
37 suspect characters on 1 page
Text is altered to blend into the white background of the document.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- 219 Not Cited or Quoted 18%**
Matches with neither in-text citation nor quotation marks
- 9 Missing Quotations 1%**
Matches that are still very similar to source material
- 0 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- 10% Internet sources
- 10% Publications
- 15% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	www.mdpi.com	1%
2	Submitted works	University of Bedfordshire on 2026-07-09	<1%
3	Internet	www.ai-futureschool.com	<1%
4	Submitted works	Asia Pacific University College of Technology and Innovation (UCTI) on 2026-08-03	<1%
5	Publication	Salloum, Said A.. "Enhancing Cybersecurity: Machine Learning and Natural Langu...	<1%
6	Internet	ijctet.org	<1%
7	Submitted works	Midocean University on 2025-09-05	<1%
8	Internet	ijerst.org	<1%
9	Internet	ruor.uottawa.ca	<1%
10	Internet	ijirt.org	<1%

11	Internet	mdpi-res.com	<1%
12	Submitted works	Kaplan College on 2026-03-16	<1%
13	Submitted works	St.Xavier's Catholic College of Engineering on 2026-04-16	<1%
14	Submitted works	Wrexham University on 2026-08-02	<1%
15	Publication	V. Sekhar, P. Ajay Kumar Reddy, K. Logesh, T. Raghunadha Reddy, Dara Raju. "Ne...	<1%
16	Internet	assets-eu.researchsquare.com	<1%
17	Submitted works	universititeknologimara on 2026-07-09	<1%
18	Submitted works	University of Essex on 2026-07-23	<1%
19	Submitted works	Georgia State University on 2023-07-09	<1%
20	Publication	Paolo Ferro, Harinadh Vemanaboina, Chander Prakash. "Computational Techniqu...	<1%
21	Internet	fastercapital.com	<1%
22	Submitted works	University of Hertfordshire on 2026-01-04	<1%
23	Submitted works	The Robert Gordon University on 2023-04-24	<1%
24	Internet	esource.dbs.ie	<1%

25	Publication	Harvinder Singh, Priyanka Kaushal, Sarabjeet Kaur. "Advanced Computing and AI...	<1%
26	Publication	"Smart Computing and Systems", Springer Science and Business Media LLC, 2026	<1%
27	Internet	albayanres.com	<1%
28	Internet	dspace.unimap.edu.my	<1%
29	Internet	publications.idiap.ch	<1%
30	Publication	"Data Mining and Information Security", Springer Science and Business Media LL...	<1%
31	Submitted works	Glyndwr University on 2026-04-23	<1%
32	Submitted works	Nottingham Trent University on 2024-09-10	<1%
33	Publication	"Advances in Intelligent Systems and Digital Applications", Springer Science and ...	<1%
34	Publication	"Innovations of Intelligent Informatics, Networking, and Cybersecurity", Springer...	<1%
35	Submitted works	Islington College,Nepal on 2026-01-20	<1%
36	Publication	Sarroeira, Rodrigo Carvalheda Duarte da Fonseca. "A Full Hate Speech Detection ...	<1%
37	Publication	"Computing and Machine Learning", Springer Science and Business Media LLC, 20...	<1%
38	Internet	iieta.org	<1%

39	Internet	repository.sustech.edu	<1%
40	Submitted works	Coventry University on 2024-10-11	<1%
41	Submitted works	Universitas Muhammadiyah Semarang on 2026-07-10	<1%
42	Internet	dspace.daffodilvarsity.edu.bd:8080	<1%
43	Publication	Bhabani Prasanna Pattanaik, Dharendra Kumar Shukla, Sambit Satpathy, Kamala...	<1%
44	Submitted works	UNICAF on 2026-07-24	<1%
45	Submitted works	universititeknologimara on 2026-07-13	<1%
46	Submitted works	University of Stirling on 2024-04-04	<1%
47	Publication	"Proceedings of International Conference on Artificial Intelligence and Networks...	<1%
48	Submitted works	Universiti Tun Hussein Onn Malaysia on 2026-05-25	<1%
49	Internet	atrium.lib.uoguelph.ca	<1%
50	Submitted works	Asian Institute of Technology on 2012-12-06	<1%
51	Publication	David Olatunde Olukanni, Oluseyi Olanrewaju Ajayi, Joshua Olusegun Okeniyi, Sh...	<1%
52	Submitted works	KTH - The Royal Institute of Technology on 2013-07-09	<1%

53	Submitted works	University of Strathclyde on 2011-08-26	<1%
54	Submitted works	University of West London on 2025-09-25	<1%
55	Internet	so06.tci-thaijo.org	<1%
56	Internet	www.coursehero.com	<1%
57	Publication	"Ubi-Media Computing, Pervasive Systems, Algorithms and Networks", Springer S...	<1%
58	Submitted works	Academies Australasia Polytechnic on 2026-01-19	<1%
59	Publication	Adnane Filali, Mohammad Shorfuzzaman, El Arbi Abdellaoui Alaoui, Mostafa Merr...	<1%
60	Submitted works	Athlone Institute of Technology on 2026-08-01	<1%
61	Submitted works	Islington College,Nepal on 2026-07-19	<1%
62	Submitted works	Taylor's Education Group on 2024-06-28	<1%
63	Submitted works	University of Exeter on 2024-08-29	<1%
64	Submitted works	University of Greenwich on 2026-04-22	<1%
65	Submitted works	University of Sunderland on 2026-05-07	<1%
66	Submitted works	Vellore Institute of Technology on 2026-04-25	<1%

67	Internet	revis.openscience.si	<1%
68	Internet	www.cpijr.com	<1%
69	Publication	"Proceedings of Data Analytics and Management", Springer Science and Business...	<1%
70	Publication	"Proceedings of Data Analytics and Management", Springer Science and Business...	<1%
71	Publication	"Recent Advances in Artificial Intelligence and Smart Applications", Springer Scie...	<1%
72	Submitted works	Chester College of Higher Education on 2025-10-13	<1%
73	Submitted works	College of Public Health on 2026-02-17	<1%
74	Submitted works	Domain Academy on 2026-06-29	<1%
75	Submitted works	Noroff University College on 2024-08-04	<1%
76	Submitted works	The Robert Gordon University on 2025-09-03	<1%
77	Submitted works	Tilburg University on 2026-06-29	<1%
78	Submitted works	Universiti Tenaga Nasional on 2026-05-09	<1%
79	Submitted works	University of Bradford on 2026-01-11	<1%
80	Submitted works	University of East London on 2026-05-08	<1%

81	Submitted works	University of Greenwich on 2023-12-22	<1%
82	Internet	www.fatalerrors.org	<1%
83	Internet	www.ijprems.com	<1%
84	Publication	Al-Marri, Noora Saeed M. J.. "Quantum Machine Learning for Intrusion Detection ...	<1%
85	Publication	Arvind Dagur, Sohith Agarwal, Dharendra Kumar Shukla, Shabir Ali, Sandhya Sharm...	<1%
86	Submitted works	Asia Pacific University College of Technology and Innovation (UCTI) on 2025-01-27	<1%
87	Submitted works	Asia Pacific University College of Technology and Innovation (UCTI) on 2025-08-13	<1%
88	Submitted works	Charles University in Prague on 2026-07-17	<1%
89	Submitted works	College of Public Health on 2026-01-30	<1%
90	Submitted works	De Montfort University on 2026-05-29	<1%
91	Publication	Manisha Guduri, V Uma Maheswari, Rajanikanth Aluvalu, Amit Krishna Dwivedi, ...	<1%
92	Publication	Poonam Nandal, Tapas Kumar, Meeta Singh, Mamta Dahiya. "Progressive Compu...	<1%
93	Publication	Shalli Rani, Ashu Taneja. "WSN and IoT - An Integrated Approach for Smart Applic...	<1%
94	Submitted works	University of Portsmouth on 2024-05-02	<1%

95	Internet	dspace.cc.tut.fi	<1%
96	Internet	dspace.sti.ufcg.edu.br:8080	<1%
97	Internet	file.techscience.com	<1%
98	Internet	jurnal.polbeng.ac.id	<1%
99	Internet	pmc.ncbi.nlm.nih.gov	<1%
100	Internet	studentrepo.iium.edu.my	<1%
101	Internet	www.eriesjournal.com	<1%
102	Publication	"Progress in Intelligent Computing and Secure Communication Systems", Springe...	<1%
103	Internet	5dok.org	<1%
104	Publication	Abhilasha Sharma, Vishwas Rathi, Anupam Biswas, Anil Singh, Omer Rana. "Multi...	<1%
105	Publication	Albluwi, Abeer. "A Novel Emoji-Aware Computational Framework for Body-Shami...	<1%
106	Submitted works	Asia Pacific University College of Technology and Innovation (UCTI) on 2025-08-04	<1%
107	Submitted works	Colorado Technical University Online on 2026-07-13	<1%
108	Submitted works	Coventry University on 2024-09-13	<1%

109	Publication	Diksha Shukla, Sanjay K. Dwivedi. "The study of the effect of preprocessing techni...	<1%
110	Submitted works	Dublin Business School on 2025-08-28	<1%
111	Submitted works	Dublin Business School on 2026-06-28	<1%
112	Submitted works	Federal University of Technology on 2024-01-23	<1%
113	Publication	Ganesh R. Naik. "Cardio-Respiratory Signal Processing and Classification - Trends,...	<1%
114	Submitted works	IDEA College on 2026-07-30	<1%
115	Submitted works	Liverpool John Moores University on 2023-06-15	<1%
116	Submitted works	Midocean University on 2026-05-02	<1%
117	Publication	Mohamed Aly Bouke, Omar Imhemed Alramli, Alsadg Ahmed Albadwi Abdalhafid,...	<1%
118	Publication	Mohammed Rasol Al Saidat, Khaled Shaalan, Suleiman Y. Yerima. "A Hybrid Deep ...	<1%
119	Submitted works	Moodle2025 on 2026-08-07	<1%
120	Publication	Pachegowda, Chandregowda. "Leveraging Generative Artifical Intelligence Large ...	<1%
121	Publication	Pawan Singh Mehra, Dharendra Kumar Shukla. "Artificial Intelligence, Blockchain,...	<1%
122	Submitted works	Queen Mary and Westfield College on 2024-11-27	<1%

123	Submitted works	Richfield Graduate Institute of Technology on 2026-06-19	<1%
124	Submitted works	Teesside University (Blackboard Ultra) on 2025-12-27	<1%
125	Submitted works	The University of the West of Scotland on 2026-04-15	<1%
126	Submitted works	Universiti Teknologi Petronas on 2026-07-31	<1%
127	Submitted works	University of Bradford on 2023-03-29	<1%
128	Submitted works	University of Kurdistan Hawler on 2026-07-16	<1%
129	Submitted works	University of Portsmouth on 2016-12-27	<1%
130	Submitted works	University of Sunderland on 2026-07-03	<1%
131	Submitted works	University of Wales Institute, Cardiff on 2025-12-14	<1%
132	Submitted works	University of Wales Swansea on 2024-08-21	<1%
133	Submitted works	Vellore Institute of Technology on 2026-07-30	<1%
134	Publication	Vijay Jeyakumar, N. Venkateswaran, Dinesh Bhatia. "Machine Learning for Biome..."	<1%
135	Internet	cinc.mit.edu	<1%
136	Internet	dokumen.pub	<1%

137	Internet	eprints.gouni.edu.ng	<1%
138	Internet	github.com	<1%
139	Internet	ijsrcseit.com	<1%
140	Internet	repository.kyu.ac.ke	<1%
141	Internet	www.hindawi.com	<1%
142	Publication	"Hybrid Intelligent Systems", Springer Science and Business Media LLC, 2025	<1%
143	Publication	Aina Musdholifah, Siti Zaiton Mohd Hashim, Sri Mulyana, Rifki Afina Putri, Faizah....	<1%
144	Publication	Pushpa Choudhary, Sambit Satpathy, Arvind Dagur, Dharendra Kumar Shukla. "Re...	<1%
145	Publication	Salih, Dana Abubakr. "A Kurdish Corpus of Fake News: Collection, Analysis, and M...	<1%
146	Submitted works	Southampton Solent University on 2023-09-08	<1%
147	Submitted works	University of Hertfordshire on 2021-09-03	<1%
148	Submitted works	University of Westminster on 2026-04-17	<1%
149	Publication	"Emerging Technologies for Developing Countries", Springer Science and Busines...	<1%
150	Submitted works	Brunel University on 2026-03-27	<1%

151

Publication

Dhirendra Kumar Shukla, Shabir Ali, Sandhya Sharma. "Artificial Intelligence and ... <1%

152

Submitted works

Liverpool John Moores University on 2022-02-15 <1%

153

Submitted works

Strathmore University (Main Account) on 2026-03-24 <1%



Automatic Detection of Arabic Spam Messages Using Machine Learning Techniques

Prepared By

Name: Asma Sayer Alotaibi

ID No: 253001753

Name: Areej Abdualaziz Alotaibi

ID No: 253001816

Submitted in fulfillment of the requirements for the degree Master of Science in Artificial Intelligence

At

Collage of Informatics

Midocean University

UAE

Supervisor Name: Dr. Khaled Eskaf

Collage of Informatics, Midocean University, Comoros

Email: khaledeskaf@midocean.edu.km

Acknowledgements

Special thanks to Dr. Khaled Eskaf

List of Contents

ii

Acknowledgements iii

List of Abbreviations vi
List of Figures..... vii
List of Tables viii

Chapter 1 Introduction 1

1.1 Overview 1

1.2 Problem Definition..... 2

1.3 Thesis Objectives 2

1.4 Thesis Contribution..... 3

1.5 Thesis Organization 3

Chapter 2 Background and Related Work..... 5

2.1 Introduction 5

2.2 Background 5

2.3 Related works..... 10

CHAPTER THREE PROPOSED APPROACH (METHODOLOGY)..... 27

3.1 Introduction 27

3.2 Research Methodology 27

3.3 Dataset Description 28

3.4 Development Environment 30

3.5 Arabic Text Preprocessing..... 31

3.6 Feature Extraction Using TF-IDF 33

3.7 Dataset Partitioning..... 34

3.8 Machine Learning Classification Models 34

3.9 Model Training..... 36

3.10 Performance Evaluation..... 36

3.11 Confusion Matrix Analysis 37

	3.12 Experimental Workflow.....	38
27	3.13 Chapter Summary.....	39
	CHAPTER FOUR EXPERIMENTAL RESULTS AND DISCUSSION	40
	4.1 Introduction	40
	4.2 Experimental Setup	40
	4.3 Dataset Distribution	41
	4.4 Performance Evaluation Metrics.....	42
44	4.5 Experimental Results	42
	4.6 Discussion of Classification Results.....	43
	4.7 Confusion Matrix Analysis	45
60	4.8 Comparative Analysis	47
	4.9 Comparison with Previous Studies	48
	4.10 Summary of Experimental Findings	51
105	4.11 Chapter Summary.....	51
12	CHAPTER FIVE CONCLUSION AND FUTURE WORK	53
	5.1 Introduction	53
	5.2 Research Summary.....	53
	5.3 Research Findings	54
	5.4 Research Contributions	55
	5.5 Practical Implications.....	56
	5.6 Research Limitations.....	56
	5.7 Future Work	57
	5.8 Chapter Summary.....	57
	References	58

List of Abbreviations

- **SMS:** Short Message Service
- **NLP:** Natural Language Processing
- **ML:** Machine Learning
- **SML:** Supervised Machine Learning
- **SMLC:** Supervised Machine Learning Classification
- **MLTs:** Machine Learning Techniques
- **DL:** Deep Learning
- **SVM:** Support Vector Machine
- **NB:** Naïve Bayes
- **RF:** Random Forest
- **CNN:** Convolutional Neural Network
- **RNN:** Recurrent Neural Network
- **LSTM:** Long Short-Term Memory
- **BiLSTM:** Bidirectional Long Short-Term Memory
- **GRU:** Gated Recurrent Unit
- **TF-IDF:** Term Frequency – Inverse Document Frequency
- **BoW:** Bag of Words
- **GAN:** Generative Adversarial Network
- **IoT:** Internet of Things
- **URL:** Uniform Resource Locator
- **API:** Application Programming Interface

- **AI:** Artificial Intelligence
- **ELM:** Extreme Learning Machine
- **F1-score:** Harmonic Mean of Precision and Recall

List of Figures

Figure 1. NLP text preprocessing pipeline diagram	7
Figure 2. spam extraction methods NLP diagram	8
Figure 3. CNN for text classification architecture diagram.....	9
Figure 4 Proposed Framework.....	28
Figure 5 Confusion Matrix.....	46
Figure 6 comparative performance of the three machine learning algorithms.....	48

53

11

List of Tables

Table 1 Related Work Summary.....17

Table 2 Dataset Distribution29

Table 3 Development Environment30

Table 4 Dataset Distribution41

Table 5 Performance Comparison of Machine Learning Models.....42

Table 6 Confusion Matrix of the SVM Classifier.....46

Table 7 Comparison with Previous Studies49

Abstract

The automatic detection of Arabic phishing emails is an important research challenge since phishing attacks are becoming a serious threat in cyber security. The morphology, spelling variations, dialectal variations and scarcity of publicly annotated datasets of Arabic makes phishing email classification an especially challenging task. This work introduces a lightweight machine learning (ML) model for the recognition of Arabic phishing emails that employs a wide Arabic text pre-processing and feature extraction techniques called Term Frequency-Inverse Document Frequency (TF-IDF). The experiments have been performed on the publicly accessible Arabic Phishing Emails Dataset (Kaggle) consisting of 1,250 Arabic emails (1000 legitimate and 250 phishing). A stratified train-test split was used to split the data into a training set of 80% and a testing set of 20%. Three supervised machine learning classifiers (SMLC) Naïve Bayes (NB), Support Vector Machine (SVM), and Random Forest (RF) were trained and evaluated with the same conditions of preprocessing and experiments. The results indicated that SVM and RF gave the best overall results, each with the accuracy of 98.0%, the precision of 100.0%, the recall of 90.0% and the F1-score of 94.74% on the test set. The findings indicate that the traditional machine learning techniques (MLTs), coupled with efficient Arabic text preprocessing and the TFIDF feature extraction can be an efficient and accurate phishing email detection system for Arabic. The results are, however, limited because to the comparatively low quantity of datasets in Arabic phishing email, and future work could include more expansive and varied Arabic phishing email datasets to confirm the results.

Keywords : Arabic Phishing Detection; Machine Learning; Natural Language Processing; TF-IDF; Support Vector Machine; Random Forest.

Chapter 1 Introduction

1.1 Overview

Digital communication has expanded rapidly and has become a major channel for financial, commercial, and private exchanges. This rapid expansion has also led to a proportionate number of phishing attempts where illegal emails are sent with malicious links, or false requests for information from the user. Phishing is one of the most serious threats in cyber security, and can cause financial loss, identity theft and access to an account or its personal information.

It is especially difficult to recognize Arabic-based phishing messages because of the morphology richness and complexity of the Arabic language, its wide use of different dialects, writing styles and informal expressions. It's harder to do automatic text analysis than in many other languages because of these attributes. Moreover, quality annotated Arabic phishing data sets are not readily available, and the preprocessing of Arabic text remains a challenge, especially in the context of developing a well-performing phishing detection system [1-3].

Previous studies have investigated various (ML) and (DL) for detecting malicious Arabic text. The differentiation of malicious and legitimate Arabic messages using traditional (ML) algorithms has been found promising, e.g., Decision Tree and (NB) [1]. Researchers have used more recent models, such as (SML) and (DL), and found that the preprocessing, tokenization and feature extraction of these models have a significant effect on the improvement of classification accuracy [2]. To address the deficiency of labeled Arabic phishing data, the use of data augmentation techniques has been proposed by other researchers which yielded better recall and F1 score [3]. Furthermore, recent research on phishing attacks on Arabic emails and SMS messages showed that (DL) models such as BiGRU can successfully extract context from short Arabic texts and boost phishing detection accuracy [4] , [5], respectively, using hybrid (DL) architectures.

The current thesis is based on the assumption that the Arabic phishing messages can be detected more accurately by the fusion of effective Arabic Text Preprocessing techniques with appropriate (MLC). Phishing messages are crafted with a strategy of deceiving the recipient through the use of persuasive messages, malicious URLs and social engineering. This necessitates that an efficient

phishing detector be able to detect both lexical and contextual characteristics in Arabic text. Thus, The current investigation suggests a (ML) -based framework for detecting Arabic phishing messages by performing extensive text preprocessing and TF-IDF feature extraction. The final goal is to provide a solution for detecting Arabic phishing emails in real world cyber security application that is accurate, efficient and practical.

1.2 Problem Definition

While phishing detection in Arabic language has developed reasonably well, the binary classification of Arabic phishing emails and legitimate Arabic messages has not been sufficiently explored as compared to English-language systems. The linguistic complexity of Arabic email classification is compounded by the presence of rich morphology, dialectal variation, spelling variation, non-standard writing and code switching. Further, (ML) models are not effective and generalizable because of the availability of limited datasets of Arabic phishing emails and class imbalance. The main focus consider the categorization of Arabic phishing and legitimate emails in a binary classification using the publicly available Arabic Phishing Emails Dataset from Kaggle. The proposed framework includes Arabic text pre-processing, TF-IDF feature extraction, and (SMLC) to create an efficient, accurate and practical method for identifying Arabic phishing emails.

1.3 Thesis Objectives

The main goal of this thesis is to create an effective framework of (MLTs), for detecting Arabic phishing emails from textual content. The study begins by preparing the Arabic Phishing Emails Dataset and developing a complete Arabic pipeline for text preparation to clean and normalize the text, remove repeated characters, remove stop words, tokenizing the text and extracting TF-IDF features from the dataset. The study starts by preparing the Arabic Phishing Emails Dataset and by developing a full pipeline for text preparation in Arabic language to clean and normalize the text, remove repeated characters, remove stop words, tokenizing the text and extracting TF-IDF features from the dataset.

This study features the training and comparison of three (SMLC) :

- Support Vector Machine (SVM)
- Naïve Bayes (NB)
- Random Forest (RF)

Accuracy, precision, and other complementing metrics are used to assess the classifiers' performance, recall, F1-score, and confusion matrix analysis, to determine the most appropriate model for detecting Arabic phishing e-mail. These performance indicators will help to identify the best classifier. The proposed method is therefore accurate, efficient and viable to detect phishing emails in an Arabic language environment.

1.4 Thesis Contribution

This research is adding to the world of Arabic (NLP) and Cybersecurity in the following ways:

- Developing a comprehensive Arabic phishing email preprocessing pipeline, including text cleaning, normalization, repeated character removal, stop-word removal, and tokenization .
- Employing TF-IDF as an effective feature representation technique for transforming Arabic phishing email text into numerical feature vectors suitable for (ML) classification .
- Training and comparing three (SMLC) — (NB), (SVM), and (RF)—using the same Arabic Phishing Emails Dataset and identical preprocessing pipeline to ensure a fair comparison .
- Providing a detailed analysis of false-negative predictions and discussing their practical implications for cybersecurity, demonstrating the effectiveness and computational efficiency of the proposed framework for Arabic phishing email detection .

1.5 Thesis Organization

Chapter 2: The chapter provides a review of related literature on spam and phishing detection, specifically, Arabic text classification techniques and the current challenges.

84

Chapter 3: In this chapter, the methodology, such as the preparation of the dataset, the preparation of preprocessing, extracting features through TF-IDF and word embeddings, and the execution of SVM, (NB), CNN, and BiLSTM models, are described.

91

20

72

Chapter 4: The experimental results are presented and analyzed by evaluating the performance of the proposed models with the main evaluation metrics considering the accuracy, precision, recall and F1-score. It also explains the results it retrieved from the different models using a comparative analysis between them and identifies the differences in their predictions with a discussion of the advantages and disadvantages of each model.

69

26

Chapter 5: The chapter gives a summary of the main findings of the thesis and suggests directions for future research. It summarizes the overall findings of the study and suggest future research directions that could lead to the improvement of the studied approach.

Chapter 2 Background and Related Work

2.1 Introduction

This chapter outlines theoretical background and past research that will back the proposed work. It presents the main concepts of spam and phishing detection, the fact that Arabic text is hard to process automatically, and it discusses the primary methods of text processing, feature extraction and classification. This chapter aims at demonstrating the technical background of the research and situating the suggested methodology within the current body of literature on Arabic spam and phishing detection. Studies in this field demonstrate that the classification of Arabic texts has gained more and more relevance as messages with malicious intent are expressed in numerous forms (such as web spam, social spam, SMS spam, phishing) and the Arabic language presents new processing difficulties due to morphology, dialects, and noisy writing styles [6], [7], [8], [13].

2.2 Background

2.2.1 Spam and Phishing Concepts

Spam can be defined as unsolicited or unwanted digital content that is delivered in large volumes with the purpose of advertising, manipulation or malicious use. Spam can be presented in the form of web pages, tweets, comments, emails, or SMS messages, and can be used in text classification tasks to deceive users, propagate malicious links or create artificial popularity. A more specifically defined threat is phishing, which means that the attacker can pretend to be a trusted organization to trick the user into providing private data, like bank account information, passwords, or verification codes. The two issues are closely related to cybersecurity since they use human trust and natural language as opposed to technical vulnerabilities. The results of previous studies suggest that some spam messages are sent in web pages, social media and via SMS; they also suggest that the development of Arabic phishing is enhanced by text-based spamming and the use of deceptive URLs [6, 7, 13].

2.2.2 Arabic Text Characteristics

Some of the challenges of automatic classification are specific to Arabic text. It is abundant in morphology, extremely inflected and is often written in both Modern Standard Arabic and local dialects. The web is often filled with messages that don't use diacritics, stress repeated, are mixed Arabic and English words or numbers, or use informal spellings or abbreviations. These characteristics generate noise and enhance ambiguity, rendering straightforward methods based on keywords less efficient. In particular, the studies with respect to Arabic SMS phishing, mention that the rich morphology and the intricate structure of the language could reduce the efficiency of the standard text classification techniques unless the text is properly normalized and modeled using the proper techniques [13]. Arabic spam investigations also reveal that brief, noisy and informal Arabic posts need a quality preprocessing prior to effective classification [7], [8], [13].

2.2.3 Text Preprocessing Techniques

Text preprocessing is an essential component in any text classification system, which helps to transform the raw text data into a suitable input for the (ML) algorithms. Some of the most common pre-processing steps are removing punctuation, normalization of the text, removal of stop words, normalization of repeated characters, tokenization, and elimination of noise like URLs and emojis. In the past, many applications have been shown to significantly benefit from the use of effective preprocessing techniques, even in the face of old fashioned (ML) algorithms [9]. The preprocessing pipeline used in the experiments of this study includes text cleaning, Arabic text normalization, elimination of stop words and duplicate characters and tokenization. Before feature extraction, these steps were done to the Arabic Phishing Emails Dataset. Other preprocessing methods cited in the literature are only mentioned to serve as background information and were not implemented in the experiments. Preprocessing is also crucial for Arabic phishing email detection since other variations in spelling, dialectal words and noisy text may jeopardize the classification accuracy. In this sense, preprocessing was viewed as an indispensable part of the proposed detection framework instead of as an auxiliary process [7, 8, 9, 13].

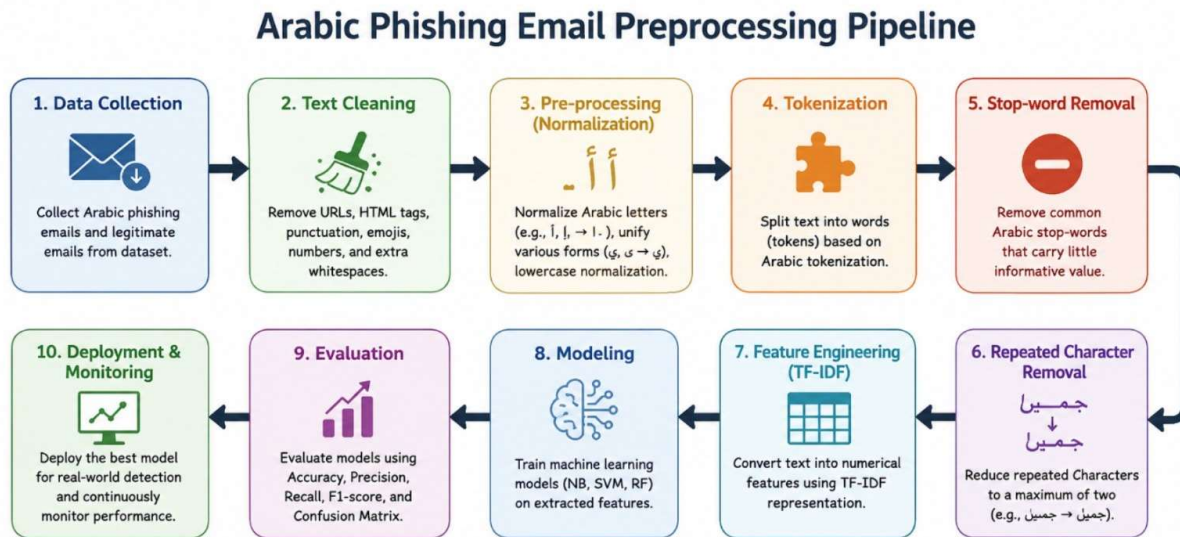


Figure 1. NLP text preprocessing pipeline diagram

2.2.4 Feature Extraction Methods

After text preprocessing, the text is then converted into number vectors (feature vectors) that can be used by (ML) algorithms. In this work, the TF-IDF method is used to convert the textual content of the Arabic phishing e-mails into a numeric vector representation. TF-IDF is an effective and efficient feature representation in text classification for small text documents in a large collection. Embedding-based representations like Word2Vec, FastText and BERT (BERT [10] is a state-of-the-art model) have shown to be more effective in learning semantic and contextual relationships in complex language tasks, but were not tested in the present work. They are only addressed theoretically in the context(ML) of the discussions. Due to its simplicity, efficiency, and effectiveness in the traditional (ML) -based phishing email detection, the experimental framework is based solely on TF-IDF. In Arabic phishing email detection, the importance of feature representation is highlighted, due to the influence of the meaning of short messages on

morphological variations and contextual expressions. Hence, the TF-IDF method was chosen for feature extraction for the entire experimental evaluation [10] , [13].

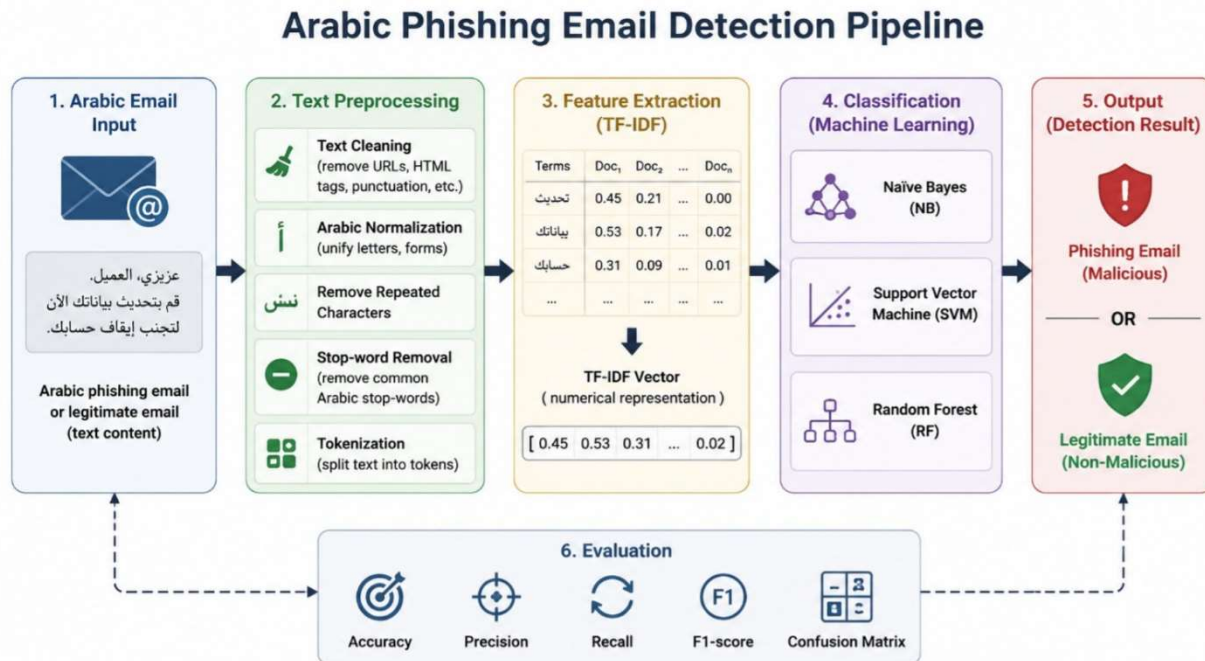


Figure 2. spam extraction methods NLP diagram

2.2.5 Machine Learning Algorithms

Conventional (ML) models are still very applicable in the text classification process with an excellent preprocessing and feature engineering. (NB) is a common algorithm due to its simplicity, speed and ability to work with high dimensional sparse data like TF-IDF vectors. The other powerful classifier is (SVM) since it is capable of complex boundaries and tends to work well with text data. Comparative experiments on spam classification reveal that, both of these methods are efficient, although SVM frequently obtains better results in cases when the feature space is well-constructed [10]. Classical algorithms continue to offer a useful basis in Arabic spam detection and can be competitive, particularly in cases where the dataset is small, or the features are well-constructed [7], [8], [10].

2.2.6 Deep Learning Models

The advantage of **(DL)** over traditional methods is that it can automatically extract complex semantic and context patterns from textual data, which makes it one of the best methods for text classification. CNN, BiLSTM and BiGRU models have proven to be powerful in the field of NLP due to their ability to model long-range context dependency and local linguistic properties. Previously, it was demonstrated that these models can yield very high phishing and spam detection classification accuracy when adequate labeled data and computational resources are available, in the context of Arabic phishing and spam detection [11, 12]. Though these benefits, **(DL)** models generally need a large of annotated dataset, longer training time and high computation resources. For this reason, the current study focused on shallow traditional (ML) models rather than **(DL)** models. With the relatively small dataset of Arabic Phishing Emails (1250 emails), the traditional classifiers with effective Arabic text pre-processing and TF-IDF feature extraction are computationally efficient, interpretable and practical classifiers with competitive classification performance. In this section only **(DL)** models are mentioned as background, and not implemented for the experimental evaluation.

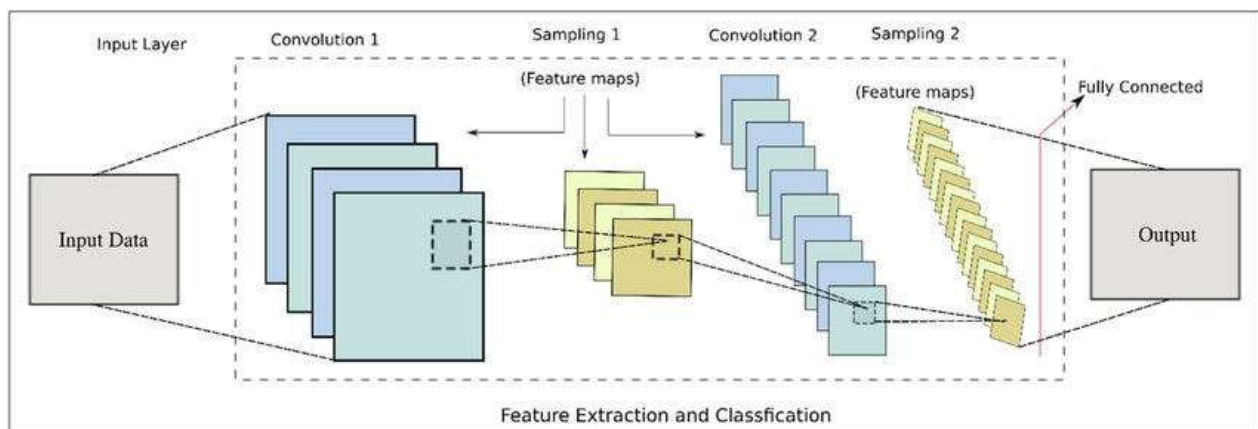


Figure 3. CNN for text classification architecture diagram

2.3 Related works

[14] Al Saidat et al. (2024)

Al Saidat et al. (2024) explored the use of Arabic SMS spam detection, which is faced with linguistic challenges due to the morphological richness and contextual complexity of Arabic texts. Considering the unique features of spam messages, the study proposed a hybrid (DL) architecture integrating Convolutional Neural Networks (CNN) with (Bi-LSTM). This integrated architecture has been designed to be able to extract local textual patterns and also to be able to capture the long-term dependency across the message sequence. The experimental data consisted of Arabic SMS messages classified as spam and legitimate messages. The authors have used multiple preprocessing techniques before model training such as text cleaning, tokenization, and padding. Before training the model, the authors have performed multiple text pre-processing techniques including padding, text cleaning, and tokenization. The proposed model was able to classify with high accuracy (96.99), precision (0.9739), recall (0.9675) and an F1-score (0.9707).

[15] Alzubaidi and Ghanem (2025)

Alzubaidi and Ghanem (2025) studied the Arabic SMS spam detection using the prism of social engineering and phishing schemes prevalent in fraudulent text messages. They employed a real-life Saudi Arabian dataset comprising of spam keywords, suspicious URLs and typical phone-number patterns that are related to spam behavior. A number of (MLC) were experimented with such as Random Forest, (NB), and Support Vector Machines (SVM). Another hybrid model suggested by the study is a Random Forest-SVM to enhance performance. The preprocessing of the data and the feature engineering on them had a significant impact on the final

93 classification performance. The hybrid model achieved high precision (0.94), recall (0.93) and accuracy (0.94) with an F1 score of 0.94.

[16] Alkaabi et al. (2025)

16 Alkaabi et al. (2025) introduced an advanced Arabic SMS spam model with AraBERT embeddings and dual-branch CNN-BiLSTM. The paper put a particular emphasis on the contrasts between Modern Standard Arabic and dialectal Arabic that are significant obstacles to Arabic text classification. To overcome this, the researchers came up with a dialect-sensitive preprocessing pipeline that normalized spelling variations without contextually altering the meaning and eliminated noisy text fragments. Two datasets were utilized, the translated UCI SMS Spam Collection and newly collected Iraqi Arabic SMS dataset. The model attained the state-of-the-art performance with 99.12% accuracy on the Modern Standard Arabic dataset and 95.3% on the dialect dataset.

[17] Filali et al. (2026)

59 Filali et al. (2026) realized SMS spam detection in multilingual settings with a specific focus on the imbalance in classes. The paper proposed a hybrid (ML), (DL), and data augmentation with Generative Adversarial Networks (GANs). GANs instead of producing complete text were trained to produce word embeddings as a means of improving dataset balance. The study contrasted six (ML) models, such as SVM, Logistic Regression, K-Nearest Neighbors, Decision Tree, (RF) and Stochastic Gradient Descent. Methods of feature extraction were TF-IDF, Bag of Words, Word2Vec, Glove and BERT. BERT + Bi-LSTM + GAN gave the best performance and yielded an F1-score of 97.61%.

[18] Errami et al. (2024)

144 The use of artificial intelligence tools in detecting Arabic twitter spam in natural
language processing field is what Errami et al. (2024) directs its target. The problem
with the study is that the purpose is to detect spam content in big scale Arabic social
media, where the dialectal variation and informal writing brings difficulties in the
classification of spam. The experiment showed the classical (ML) algorithms with
1 SVM and Random Forest against (DL) algorithms with LSTM, Bi-LSTM and models
15 based on AraBERT. The authors wanted to enhance the accuracy of the detection and
yet keep the computational efficiency using an Arabic tweet dataset which was
labeled as spam and non-spam. The models were found to improve performance on
146 complex Arabic text, but also showed to be resource intensive in terms of training
time and resources.

[19] Jurkuch et al. (2026)

Jurkuch et al. (2026) offered an efficient hybrid SMS spam detection model, which
is likely to be resource-efficient and especially applicable in low-computation
systems like the Internet of Things. The research was inspired by the shortcomings
of rule-based filters and the high cost of (DL) systems. The design proposed consisted
of a Bidirectional GRU (Bi-GRU) text feature extractor and an XGBoost
classification network. The SMS Spam Collection dataset and the SpamAssassin
email corpus were used to evaluate the model to document cross-domain
generalization. The findings were very competitive with an accuracy of 98.7 on SMS
1 data and 98.35 on email data. The model was also less latent and resource-consuming
compared to CNN-LSTM and Bi-LSTM counterparts.

[20] Zawari (2025)

104 Zawari (2025) introduced a hybrid ensemble (ML) model to identify Arabic SMS scam messages which are also referred to as smishing. The experiment was to enhance the classification precision by pooling multiple classifiers instead of using one model. Algorithms applied were Logistic Regression, Random Forest, Gradient Boosting, Stacking and Voting. The study was founded on a pre-processed Arabic SMS data of scam and non-scam messages. The paper has put emphasis on the significance of preprocessing and ensemble learning in enhancing performance with regard to Arabic malicious text. The hybrid ensemble method was the most successful, with an accuracy of 91.89% and an F1-score of 91.55.

33 [21] Bani Younes and Ababneh (2026)

90 Bani Younes and Ababneh (2026) is a comparative analysis of (ML) algorithms in spam detection on various digital communication channels such as SMS, email, and social media. The researchers were interested in generalizing models to different domains and whether shared classifiers could be used to produce reliable performance on unseen data. The algorithms that were compared were (NB) ,SVM, Decision Trees and Random Forest. The models in various situations were tested using multiple benchmark datasets. The results indicated that the models were quite effective with familiar data but failed to perform as well with new or unknown data. This finding highlighted the importance of flexible models and enhanced feature representation.

65 [22] Menthe et al. (2024)

131 Menthe et al. (2024) studied the issue of SMS Spam Detection using (ML) with special emphasis on the pre-processing and feature engineering. It analyzed the effect of data cleaning, data normalization, and feature selection on the classification model

accuracy. The authors used common (ML) algorithms including (NB) and SVM to a publicly available dataset of SMS. Their experiment has revealed that model performance was significantly enhanced with the inclusion of preprocessing and feature extraction. The study is significant as it confirms that even comparatively simple classifiers may produce good results provided that the input text is in a proper format. This renders the research to be applicable in spam detecting systems which need effective and readable solutions.

123 [23] Syarifuddin et al. (2025)

Syarifuddin et al. (2025) compared (ML) with (DL) on spam detection and discovered that (DL) methods tended to achieve superior classification but had a higher computational load. (NB), SVM, Random Forest, Gradient Boosting, and LSTM were some of the models in the study. The models were tested using SMS and email data under various conditions. The paper presented the accuracy-efficiency trade-off whereby deep-learning could be more effective in complicated text classification tasks than traditional approaches. Meanwhile, the authors have reported that (ML) models are still useful, as they are quicker, easier and more feasible in resource-constrained settings.

107 [24] Ahmed et al. (2022)

Ahmed et al. (2022) surveyed (ML) approaches to spam detection in email and IoT. The study did not suggest a single new model but compared the advantages and disadvantages of popular algorithms in a variety of applications. The review has centered on (NB), Decision Trees, Neural Networks and (RF) and the way each of the methods works with varying types of spam data. The paper has highlighted that spam detection issues are different based on the channel of communication and

features taken. Its key input was to generalize previous research and demonstrate that traditional and neural methods are useful, with respect to the size of datasets, the level of noise, and the computational constraints.

[25] Birthare, Jain, and Meena (2025)

Birthare, Jain, and Meena (2025) reviewed the (ML) approach to mobile SMS spamming and emphasized the significance of preprocessing, feature extraction, and evaluation plan. The paper has highlighted that spamming in mobile messages can only be detected by effectively processing short and noisy text. It contrasted a number of algorithms, such as SVM, Random Forest, LSTM, Bi-LSTM, and AraBERT, as well as explained the place of Extreme Learning Machine (ELM) as a fast and useful classifier. The study was founded on the Kaggle SMS Spam dataset, which consisted of the spam and ham messages. The authors concluded that NLP-based preprocessing is a significant factor in performance and that lightweight models could be effective, given appropriate design.

[26] Julis and Alagesan (2020)

In their research, Julis and Alagesan (2020) examined how (ML) identifies SMS spam with the help of text mining algorithms. The article touched on the emergence of SMS spam in India and the Middle East region at large where mobile messaging is a significant source of spam attacks. The authors employed (ML) classifiers to differentiate between ham and spam messages in a large set of SMS data including messages for Indian users. The paper demonstrated the potential of text mining to assist automated spam filtering based on transformation of raw messages into structured features to facilitate classification. The work is practical as it shows the

practical utility of traditional (ML) approaches to short-text spam detection experiments.

[27] Kumar, Sonowal, and Nishant (2020)

22 Kumar, Sonowal and Nishant (2020) explored email spam detection by applying (ML) algorithms. The research was inspired by the tendency towards the growing prevalence of phishing and spam email campaigns that strive to deceive users with illicit links and various pseudonyms. The authors have compared various (MLTs), 30 and have chosen the optimal model in terms of precision and accuracy. The article demonstrated that spam mail can be identified successfully by deriving valuable textual characteristics and using supervised learning. The research is applicable as it shows that the (MLC) continues to be significant in spam classification, particularly when the intention is to establish a robust and interpretable detection.

[28] Roy, Singh, and Banerjee (2020)

20 Roy, Singh, and Banerjee (2020) suggested a text-only (DL) model to filter SMS spam. They found that CNN and LSTM models are able to automatically extract useful features using short message text without relying on manual feature engineering. The study was carried out on a benchmark of SMS data of spam and 80 non-spam, and the model proposed demonstrated a high level of accuracy of 99.44. 25 The results showed that (DL) can be very useful with the classification of short text and it can be much more effective than numerous traditional techniques. The

significance of the study is that it justifies the application of neural architectures in spam detection, particularly where the availability of large enough labeled data.

Table 1 Related Work Summary

Study	Data Type	Language	Task	Algorithm	Dataset Size	Performance	Key Limitation	Relevance to Current Study
Al Saidat et al. (2024)	SMS	Arabic	Spam Detection	CNN + BiLSTM	Arabic SMS dataset	Acc. 96.99%, F1=0.9707	High computational cost	Demonstrates the effectiveness of (DL) for Arabic SMS spam detection but differs from this study, which focuses

								on Arabic phishing emails using lightweight ML.
Alzubaidi & Ghaneim (2025)	SM S	Arabic	Spam/Phishing Detection	RF, NB, SVM, Hybrid RF-SVM	Saudi SMS dataset	Acc. 94%, F1=0.94	Limited geographic coverage	Supports the use of traditional ML algorithms, but the task is SMS rather than email phishing detection.
Alkaabi et	SM S	Arabic	Spam Detection	AraBERT +	UCI + Iraqi SMS	Acc. 99.12%, 95.3%	Complex	Shows the benefit

al. (2025)				CNN- BiLS TM	dataset s		architec ture	of (DL), whereas the current study emphasi zes computa tional efficienc y.
Filali et al. (2026)	SM S	Multili ngual	Spam Detectio n	ML + DL + GAN	Multili ngual dataset s	F1=97. 61%	Very high comput ational cost	Demonst rates advance d feature learning but requires significa ntly greater computa tional resource s.

Errami et al. (2024)	Twitter	Arabic	Spam Detection	SVM, RF, LSTM, BiLSTM, AraBERT	Arabic Twitter dataset	DL improved accuracy	Time-consuming training	Different communication channel (Twitter) from the current email-based study.
Jurkuch et al. (2026)	SMS & Email	Multilingual	Spam Detection	Bi-GRU + XGBoost	SMS + Email datasets	Acc. 98.7% (SMS), 98.35% (Email)	Hybrid DL model	Similar focus on email classification but uses hybrid (DL) instead of lightweight ML.

Zawar i (2025)	SM S	Arabic	Smishin g Detectio n	LR, RF, GB, Stacki ng, Votin g	Arabic SMS dataset	Acc. 91.89% , F1=91. 55%	Modera te accurac y	Similar cybersec urity applicati on, but focuses on SMS rather than phishing emails.
Bani Younes & Ababneh (2026)	SM S, Em ail, Soc ial Me dia	Multip le	Spam Detectio n	NB, SVM, DT, RF	Multip le dataset s	Good perform ance	Weak generaliz ation	Highligh ts the need to evaluate classifier s across different commun ication channels .
Menth e et al. (2024)	SM S	Englis h	Spam Detectio n	NB, SVM	Public SMS dataset	Improv ed accurac	Limited model	Supports the importan

						y after preproc essing	comple xity	ce of preproce ssing, which is adopted in this study.
Syarif uddin et al. (2025)	SM S & Em ail	Multip le	Spam Detectio n	NB, SVM, RF, GB, LST M	Multip le dataset s	DL outperf ormed ML	High resourc e usage	Supports the trade-off between accuracy and computa tional efficienc y.
Ahme d et al. (2022)	Em ail & IoT	Multip le	Survey	NB, DT, NN, RF	Revie w study	RF and NN perform ed well	No experim ental validati on	Provides theoretic al support for selecting tradition al ML

								algorith ms.
Birtha re et al. (2025)	SM S	Englis h	Spam Detectio n	SVM, RF, LST M, BiLS TM, ELM	Kaggle SMS dataset	ELM effectiv e	Needs optimiz ation	Confirm s the importan ce of NLP preproce ssing and lightwei ght classifier s.
Julis & Alage san (2020)	SM S	Englis h	Spam Detectio n	ML classifi ers	SMS dataset	Effectiv e classifi cation	Limited modern techniq ues	Demonst rates the applicab ility of tradition al ML methods for short- text

								classific ation.
Kuma r et al. (2020)	Em ail	Englis h	Email Spam Detectio n	ML algori thms	Email dataset	High precisio n and accurac y	Limited to email spam	Closest to the current study because it uses email classific ation, although it is not specific to Arabic phishing emails.
Roy et al. (2020)	SM S	Englis h	Spam Detectio n	CNN, LST M	SMS dataset	Acc. 99.44%	High comput ational cost	Shows the effective ness of (DL) but at a significa

								ntly higher computa tional cost than the propose d framewo rk.
--	--	--	--	--	--	--	--	---

2.3.1 Critical Analysis of Related Work.

The reviewed studies showed that the detection of malicious text in Arabic has been studied in various communication media such as SMS messages, emails and posts on social media. These studies have, however, different classification tasks and must not be directly compared. The most of the studies like Al Saidat et al. (2024), Alkaabi et al. (2025), Zawari (2025) and Julis and Alagesan (2020) are devoted to the detection of Arabic SMS spam or smishing as opposed to the detection of phishing in the email context. On the other hand, Kumar et al. (2020) explore the case of email spam detection, and Errami et al. (2024) examine the characteristics of Arabic Twitter spam, which is a different kind of text data. Moreover, some studies use (DL) models like CNN, BiLSTM, BiGRU, transformer, and others, while some use traditional (ML) models like (NB), Support Vector Machine, and Random Forest. While (DL) models can lead to higher predictive performance, they typically have much greater data requirements and computational needs. The current work is different from

previous studies because it concentrates on binary classification of Arabic phishing emails through using publicly available Arabic Phishing Emails Dataset. It utilizes a light (ML) framework which uses a comprehensive Arabic text preprocessing, TF-IDF feature extraction, and an evaluation of three classifiers, namely (NB), (SVM), and Random Forest, in comparison on the same experimental setup. This is to make a fair comparison and make an efficient solution for detecting Arabic phishing emails.

2.3.2 Research Gap

A research gap will be determined from the literature reviewed. There have been several studies that investigated Arabic malicious text detection using different (ML) and (DL) approaches, but few studies have dealt with the field of Arabic phishing email detection by preprocessing the Arabic text comprehensively and using lightweight traditional (MLTs),. In addition, there is a lack of studies that compare more than one (SMLC) with the same preprocessing pipeline, TF-IDF feature extraction, and a publicly available phishing email dataset in Arabic language to have a fair and consistent evaluation. In order to fill these gaps, the present work introduces a lightweight and computationally efficient framework to detect phishing emails in Arabic language which combines Arabic text preprocessing, TF-IDF feature extraction and evaluation of (NB), (SVM), and (RF) classifiers on the publicly available Arabic Phishing Emails Dataset. The proposed framework aims to offer a correct, feasible and efficient algorithm for the detection of phishing emails in Arabic script in a real cyber security context.

CHAPTER THREE PROPOSED APPROACH (METHODOLOGY)

3.1 Introduction

In order to implement the proposed (ML) approach to detect Arabic phishing messages, this chapter outlines the research methodology provided in this study. It describes the whole experimental process from data sets collecting and preparing, to text processing, feature extraction, model building, to model evaluating. The methodology proposed is to develop an efficient classification system that can correctly classify phishing messages from legitimate Arabic messages with computational efficiency. To compare the three (SML) algorithms fairly and objectively with the same pre-processing method and the same assessment criteria, the same method was used to implement them. The same method was used to implement the three (SML) algorithms, namely (NB), (SVM), and (RF), to ensure a fair and objective comparison and the same pre-processing method and the same assessment criteria.

3.2 Research Methodology

The proposed method for the research is a series of related stages which will convert raw Arabic phishing email text into numerical representations suitable for (MLC). The whole process starts by collecting data sets, then subjecting them to Arabic text preprocessing, extracting features using TF-IDF, training a model, making predictions, and evaluating the accuracy of the predictions. The experimental process used in this research is shown in Figure 4.

The experimental workflow adopted in this study is illustrated in

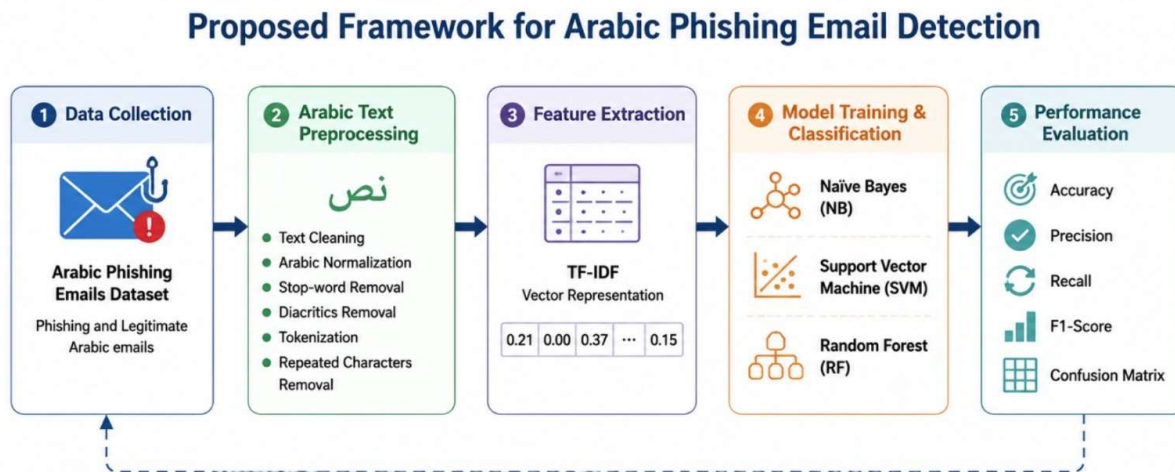


Figure 4 Proposed Framework

3.3 Dataset Description

The Arabic Phishing Emails Dataset obtained from Kaggle with public availability was applied in the experiments. The dataset has been chosen due to the fact that it is one of the few publicly available datasets that are specifically designed for the detection of phishing emails in Arabic, which makes it suitable for (SML) research in Arabic cybersecurity. The data set consists of 1,250 Arabic email messages, 1,000 legitimate and 250 phishing, which is a binary classification problem.

The data set was downloaded from Kaggle in July 2026 with the Kaggle dataset license and no samples were deleted before development of the model. The input feature for the study was the text from the e-mail and the target variable was the class label. The labels of the classes were translated to 1 = Legitimate and 2 = Phishing prior to model training. Arabic text preprocessing and TF-IDF feature extraction were then used to preprocess the text before the (MLC) were trained and evaluated.

Table 2 Dataset Distribution

Characteristic	Description
Dataset Name	Arabic Phishing Emails Dataset
Source	Kaggle
Language	Arabic
Communication Channel	Email
Total Emails	1,250
Legitimate Emails	1,000
Phishing Emails	250
Classification Type	Binary Classification
Input Field	Email Text
Target Field	Class Label
Label Encoding	0 = Legitimate, 1 = Phishing
Access Date	July 2026
License	Kaggle Dataset License

Training and testing data sets were created at random from the data collection (80/20 split). For the sake of reproducing the experimental procedure, the `random_state` was set to the value of 42. Furthermore, stratified sampling was used to retain the original class distribution throughout both samples.

Dataset Split	Number of Emails	Percentage
Training Set	1,000	80%
Testing Set	250	20%
Total	1,250	100%

3.4 Development Environment

This proposed system has been developed by the Python programming language in the kaggle notebook environment. It was chosen because of the large number of (ML) and (NLP) libraries available in Python. Pandas and NumPy were used for data manipulation, Scikit-learn for (ML) model implementation, feature extraction, partitioning of the dataset, and the evaluation of the metrics. Regular expressions and custom-made stop-word removal methods were used to perform Arabic text preprocessing operations. Experimental results were visualized with Matplotlib library. The software environment used in all the experiments is as shown in Table3:

Table 3 Development Environment

Component	Description
Programming Language	Python 3.12

Development Platform	Kaggle Notebook
(ML) Library	Scikit-learn
Data Processing	Pandas, NumPy
Natural Language Processing	NLTK, Regular Expressions
Visualization	Matplotlib

3.5 Arabic Text Preprocessing

One of the most crucial steps in (NLB) is the preprocessing of text data since the data from text sources may contain irrelevant information that adversely affects the classification accuracy. The processing of Arabic language is made difficult by its rich morphology, multiple writing styles, spelling variations and the frequent use of dialectal expressions. Thus, multiple preprocessing steps were performed to enhance data quality and minimize noise prior to feature extraction. The pre-processing pipeline used in this study can be divided into the following steps.

3.5.1 Text Cleaning

The first preprocessing step was to eliminate irrelevant text elements which do not make semantic sense. Numerical value, punctuation marks, special symbols, non-Arabic characters and hyperlinks were stripped away applying regular expression patterns. Removing these factors decreases feature sparsity and enhances the quality of the text representation.

3.5.2 Arabic Text Normalization

Arabic letters frequently appear in multiple orthographic forms that carry identical meanings but are represented differently at the character level. Consequently, text normalization was performed to standardize these variations. Characters such as (ا، إ، ؤ) were converted into (ا), while other commonly occurring variants were similarly normalized. This process reduces vocabulary size and enhances the consistency of textual representations.

3.5.3 Removal of Repeated Characters

Consecutive repeated characters were replaced to a single character with repeated characters normalized. In informal Arabic, the repetition of letters is sometimes used to highlight or express emotion (e.g., 'جميبيبي' for 'جميل'). Such repetitions do not affect the semantic meaning of the word but create extra variations in the text and make it sparser. Reducing the repetition of characters provides more uniform vocabulary and more power for representing features in (ML).

3.5.4 Stop-Word Removal

Common Arabic function words contribute only small amounts of discriminative information in text classification tasks. As a result, the following Arabic stop words were filtered out of the Arabic pre-processing: "من، في، على، إلى، هذا، ذلك". Removing these high frequency words helps to diminish the number of features the algorithms are using when they classify emails as phishing or not, while allowing the algorithms to concentrate on more meaningful lexical features that play a role in the classification.

3.5.5 Tokenization

The Arabic e-mails were preprocessed and segmented to lexical units (tokens). The process of tokenization entails splitting up a text into words and/or meaningful

segments that can become the basis for feature extraction and (ML) analysis. It was then fed into the model training and evaluation process of TF-IDF feature vectors.

3.6 Feature Extraction Using TF-IDF

(ML) algorithms rely on data that is numerical, not text. Thus, the preprocessed Arabic phishing emails were converted to numerical feature vectors by applying the (TF-IDF) method. The advantage of using TF-IDF is that it gives more weight to the words that appear within each document and do not appear in many others in the collection, which captures the informativeness of words and lowers the impact of the most common terms. TF-IDF generates more discriminative feature vectors than simple term-frequency representations and has shown good performance on text classification tasks, such as spam and phishing email detection.

The TF-IDF weight is computed according to Equation (3.1):

$$\text{TF-IDF}(t, d) = \text{TF}(t, d) \times \log\left(\frac{N}{\text{DF}(t)}\right) \quad (3.1)$$

where:

- $\text{TF}(t, d)$ is the frequency of term t in document d .
- $\text{DF}(t)$ is the number of documents containing term t .
- N is the total number of documents in the dataset.

The obtained TF-IDF representations were then fed to three classifiers such as (NB), Support Vector Machine and (RF).

3.7 Dataset Partitioning

For testing the generalization capability of the classifiers, the dataset was split into an independent train subset (80%) and test subset (20%) with a fixed random seed (random state = 42) with a stratified train-test split. The training data was used for all the preprocessing steps identified, such as text cleaning, tokenization, stop-word removal, TF-IDF vectorization, and feature selection. Fitted preprocessing and feature selection were then applied to the test set without re-fitting, which meant there was no data leakage and it enabled an unbiased assessment of model performance.

3.8 Machine Learning Classification Models

Three (SMLC) were chosen for the experimental testing due to their well-known capabilities in text classification problems. The dataset is class imbalanced (1,000 legitimate emails vs 250 phishing emails) but no resampling or class-weighting techniques were used, so that the same experimental conditions could be used in order to have a fair comparison with all classifiers. The developed models were evaluated on different evaluation metrics like accuracy, precision, recall, f1 score and confusion matrix.. These complementary measures allowed for a more complete evaluation of the classification output than an overall evaluation of accuracy.

3.8.1 Naïve Bayes

Naive Bayes is one of the popular classification methods, based on the Bayes' Theorem and conditional independence of features. It has found wide application in spam and phishing detection of e-mail, because of its low computational complexity and its suitability to work with sparse textual representations like TFIDF feature vectors.

The classifier was implemented using the following parameter:

- $\alpha = 1.0$

3.8.2 Support Vector Machine

Supervised learning machine which finds the optimum separating hyper plane by maximizing the margin between classes is known as (SVM). In the field of text classification, SVM has shown to perform well due to its ability in high dimensional feature spaces. A linear kernel was chosen for this study because it is a simple, fast and appropriate kernel for sparse TF-IDF feature vectors.

The SVM classifier was configured using:

- kernel = **linear**
- $C = 1.0$

3.8.3 Random Forest

Random Forest (RF) is an ensemble learning algorithm which trains multiple decision trees and then uses majority voting to find the predicted class. This ensemble approach will be more robust, reduce overfitting, and overall be more accurate than a single decision tree. The number of decision trees in the (RF) classifier was set to 200, which is a good balance between performance and computation.

The (RF) classifier was configured using:

- $n_estimators = 200$
- $max_depth = None$

- `random_state = 42`

The use of 200 decision trees provides stable classification performance while maintaining reasonable computational efficiency.

3.9 Model Training

The same training set and feature vectors (TF-IDF) have been used to train all of the classification algorithms. Once trained these models were applied to the testing data set and the results were noted to see whether each message in the test set was genuine or a phishing message. We used the same pre-processing techniques and features representation for all classifiers, so we could compare the results based on the learning ability of the classifiers and not on the data processing.

3.10 Performance Evaluation

Four widely used metrics in binary classification problems were used to evaluate the predictive performance of the proposed models.

Accuracy

Accuracy is the overall percentage correctly classified.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision

Precision refers to the percentage of messages that the model identifies as phishing that are actually phishing messages.

$$Precision = \frac{TP}{TP + FP}$$

Recall

Recall measures the ability of the classifier to correctly identify phishing messages.

$$Recall = \frac{TP}{TP + FN}$$

F1-Score

The F1-Score is the harmonic mean of the precision and recall, giving a trade-off between the two, and thus a balanced measure of the model's overall classification performance.

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

The combination of these evaluation measures allows for a comprehensive evaluation of the classifier, including its predictive performance as well as its ability to detect phishing emails.

3.11 Confusion Matrix Analysis

In addition to numerical values for the performance of each of the classifiers, a confusion matrix for each classifier was also obtained, allowing for detailed knowledge of the results of the predictions. The confusion matrix is a tabular representation of the classification results and includes the number of True Positives,

True Negatives, False Positives, and False Negatives, allowing for the easy interpretation of the results of the model prediction. This analysis can be used to reveal the nature of the errors committed by each model in classification, and it can also provide more insight into the performance of classifiers.

3.12 Experimental Workflow

The experimental procedure adopted in this research can be summarized as follows:

1. Download the phishing email data set from Arabic language.
2. Stratified split the data into training and testing data sets at an 80:20 ratio.
3. Clean the training data, normalize it, remove repeated characters, remove stop words and do tokenization to it as in Arabic text preprocessing.
4. Use 4 to fit the TF-IDF vectorizer to the training data and transform the training data into numerical feature vectors.
5. Avoid information leakage when transforming the testing data with the TF-IDF vectorizer that has been trained with the training data.
6. Train (NB), (SVM), and (RF) classifiers with transformed training data.
7. Make predictions for unseen testing sample class labels.
8. Assess accuracy, precision, recall, f1 score and confusion matrix of models.
9. Analyze and compare the performances of the three classifiers and select the best model for phishing detection of Arabic e-mails.

3.13 Chapter Summary

In this chapter, the methodology proposed to use supervised (MLTs), for Arabic phishing message detection was introduced. The chapter outlined the characteristics of the dataset, dataset preprocessing pipeline, TF-IDF feature extraction pipeline, dataset partitioning, and implementation of three popular classification algorithms namely (NB), (SVM), and Random Forest. Lastly, the assessment parameters and experimental procedures were presented. Results of the experiments, comparison of the performance of the implemented classifiers and discussion about the effectiveness of the classifiers in detecting the phishing messages in Arabic language are presented in the next chapter under the same experimental conditions.

CHAPTER FOUR EXPERIMENTAL RESULTS AND DISCUSSION

4.1 Introduction

In an experiment, the Arabic phishing message detection framework has been implemented and its results presented in this chapter. The goal of the experiments is to assess the ability of the three (SML) algorithms (NB), (SVM), and (RF)) to classify phishing messages as phishing or normal Arabic messages. The chapter also presents the comparative evaluation of classifiers according to their performance in various standard performances like the Accuracy, the Precision, the Recall, the F1 Score and the Confusion matrix analysis. Lastly, a discussion of the results of the experiment is given and the results are compared with the previous studies mentioned in chapter Two.

4.2 Experimental Setup

All experiments were run in a Kaggle Notebook using Python. The Arabic phishing dataset was loaded and then several preprocessing steps such as text cleaning, Arabic normalization, removing stop words, removing repeated characters and tokenization were applied. The obtained text after cleaning was then converted to numerical feature vectors by applying TF-IDF technique for feature extraction.

The data set contained 1250 Arabic messages, 1000 legitimate and 250 phishing messages. A stratified train-test split with a fixed random seed (42) was used to split the dataset into 80% training and 20% testing data, with 1,000 samples in the training set and 250 samples in the testing set. using:

- `test_size = 0.20`
- `random_state = 42`
- `stratify = y`

To obtain a more reliable estimate of model performance and improve the robustness of the evaluation, **Stratified 10-Fold Cross-Validation** was employed. Stratification ensured that the original class distribution (1,000 legitimate and 250 phishing messages) was preserved in each fold. For training each iteration, 9 folds were used and one-fold was utilized for testing; the testing fold was changed for each iteration until each fold had been tested exactly once. Fixed random seed (`random_state = 42`) was used to ensure reproducibility with shuffling. The performance metrics reported are the average of the 10 folds. All classifiers were trained with the same TF-IDF feature representation, and evaluated with the same cross-validation protocol to enable a fair comparison.

4.3 Dataset Distribution

Table 4 Dataset Distribution

Class	Number of Samples	Percentage
Legitimate	1000	80%
Phishing	250	20%

Total	1250	100%
--------------	-------------	-------------

The data set is moderately imbalanced with 4 legitimate messages for every phishing message. This distribution is typical of a phishing scenario in which a small portion of emails are phishing attempts.

4.4 Performance Evaluation Metrics

To assess the predictive power of the classifiers, four widely-used performance metrics were used:

- Accuracy
- Precision
- Recall
- F1-score

These metrics provide complementary information regarding the classifiers' ability to correctly identify phishing messages while minimizing false alarms.

4.5 Experimental Results

The three (ML) algorithms were trained using identical preprocessing procedures and TF-IDF feature vectors. Their performance on the testing dataset is summarized in Table :

Table 5 Performance Comparison of (ML) Models

Model	Accuracy	Precision	Recall	F1-score
(NB)	96.80%	100.00%	84.00%	91.30%

Support Vector Machine	98.00%	100.00%	90.00%	94.74%
Random Forest	98.00%	100.00%	90.00%	94.74%

Source: *Experimental Results*.

The experimental results show that the three classifiers all have high predictive power on the testing data. It is worth noting however, that this testing subset consisted of 250 emails, 50 of which were phishing emails and 200 were legitimate emails. As a result, the recall value is altered by about 2% every phishing email that is misclassified. Despite the promising results presented, it should be noted that the number of emails in the dataset is small, and the data might not be representative of all Arabic phishing emails, suggesting the need for more validation with larger and more diverse datasets of Arabic phishing emails before the results can be generalized to a real deployment setting.

4.6 Discussion of Classification Results

The experimental results show that the proposed Arabic text preprocessing pipeline with TF-IDF feature extraction approach helps all three different (MLC) obtain high classification performance. But some significant variations were found between the tested algorithms.

Overall, (NB) predicted the class of the phishing messages with an accuracy rate of 96.8% showing that its probabilistic learning approach was successful even with its relatively simple assumptions. The accuracy of the classifier was 100%, with no false positives that is, none of the valid messages were misclassified as phishing. The recall, however, was 84%, meaning that a section of the phishing messages were not

recognized by the classifier. Several phishing messages were not detected (false negative) which lowers the overall F1-score to 91.30%.

Overall, SVM showed the highest classification accuracy of 98%, the highest precision of 100%, the highest recall of 90% and the highest F1-score of 94.74%.

The precision is ideal and means that each message predicted as being phishing actually is a phishing message. The recall is 90%, meaning that 45 out of 50 phishing messages were correctly classified, but 5 phishing messages were incorrectly classified as legitimate (false negatives). What is significant here from the cyber security standpoint is that these false negatives could result in phishing messages reaching users and potentially cause financial fraud, credential theft and other cyber security issues. However, linear SVM had excellent performance in dealing with high dimensional sparse feature vector with good generalization capacity.

This is an example of a trade-off between precision and recall in phishing detection. While the classifier was able to correctly identify all phishing emails and to avoid making false-positive predictions, it missed some phishing emails. A false negative usually carries a higher risk in applications than a false positive in cybersecurity, since undetected phishing emails could be sent to users, and they could then be victim to phishing attacks, resulting in financial loss, credential theft, or other security incidents. Hence, a high precision and improving recall is still an important goal for future research.

(RF) yielded the same overall results as SVM, with 98% accuracy, 100% Precision, 90% Recall and 94.74% F1 Score results. The model correctly identified 45 phishing messages and missed 5 phishing messages, which is the same number of false negative messages as SVM. (RF) is robust because it uses a majority voting system

that adds up several decision trees, thus reducing the danger of over-fitting. Thus, the classifier had good generalization skills on unseen Arabic phishing messages.

While both (SVM), and (RF) had excellent overall accuracy, they were not able to achieve a 100% recall, meaning that there was a small amount of phishing emails that were missed. Therefore, the proposed framework can be considered as a lightweight and lightweight computing solution, and not a one-stop solution to replace multiple layers of cyber security defenses. The use of transformer-based language models or contextual word embeddings, or a combination of (ML) and (DL), could be explored in future work to further enhance recall.

4.7 Confusion Matrix Analysis

Since (SVM), and (RF) achieved identical classification performance, only the (SVM), confusion matrix is presented as a representative example. Both classifiers

produced the same numbers of true positives, true negatives, false positives, and false negatives; therefore, the (RF) confusion matrix would be identical.

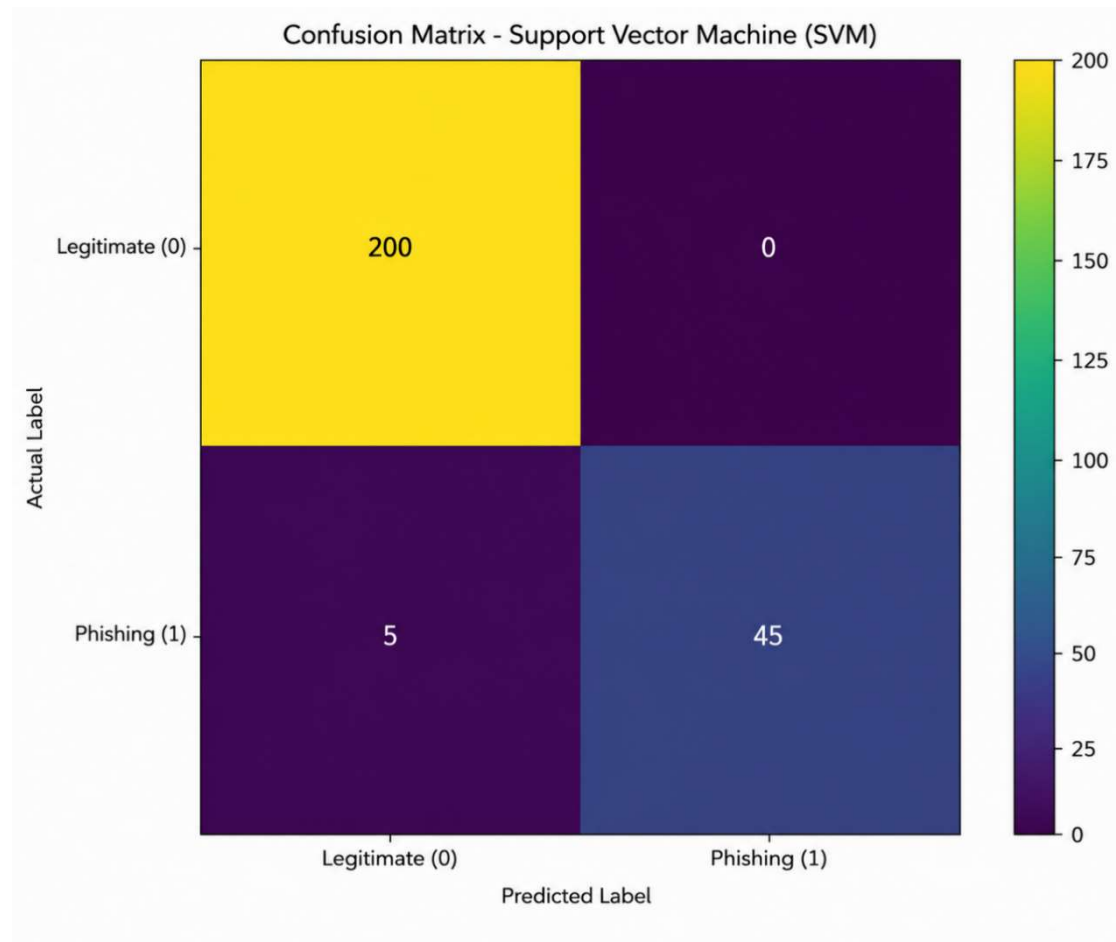


Figure 5 Confusion Matrix

Table 6 Confusion Matrix of the SVM Classifier

Actual Class	Predicted Legitimate	Predicted Phishing
Legitimate	200	0

Phishing	5	45
----------	---	----

Source: Experimental Results.

According to the confusion matrix, the SVM classifier correctly identified all **200 legitimate messages**, producing **zero false positives**. Among the **50 phishing messages** in the testing dataset, the classifier successfully detected **45 phishing messages**, while only **5 messages** were incorrectly classified as legitimate.

Consequently,

- True Negatives (TN) = 200
- False Positives (FP) = 0
- False Negatives (FN) = 5
- True Positives (TP) = 45

There is a special need for no false alarm for practical implementation, as good users are not likely to be blocked from regular communication. Moreover, the relatively small size of the false negatives further shows that the classifier is capable of identifying a majority of the phishing attacks.

4.8 Comparative Analysis

Figure 6 illustrates the comparative performance of the three (ML) algorithms across the four evaluation metrics.

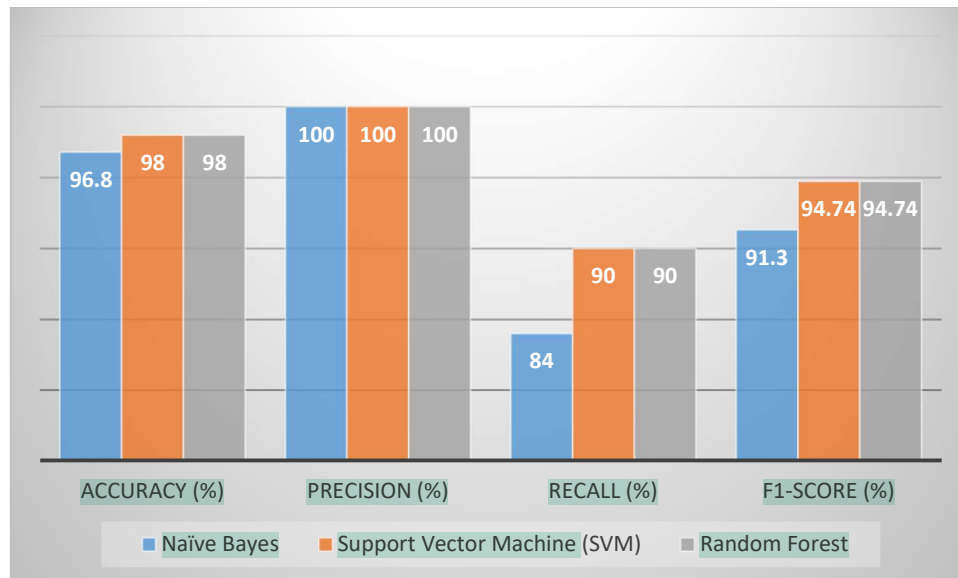


Figure 6 comparative performance of the three (ML) algorithms

The comparative analysis shows that (SVM), and (RF) always have better accuracy than (NB). SVM and (RF) classifiers had the highest recall and were able to correctly classify more phishing messages than others, though all classifiers were successful in achieving a precision of 100%. They obtained better F1-scores and accuracy due to this improvement. The results indicate that TF-IDF feature extraction has good discriminative power in phishing messages in Arabic language and the advanced classifiers like SVM and (RF) are more efficient in utilizing the representations of these features than simple probabilistic classifiers.

4.9 Comparison with Previous Studies

Overall, the experimental results are in line with the existing research on Arabic Spam and Phishing Detection. In this regard, (SVM), and (RF) have been reported as some of the most promising traditional (ML) algorithms for text classification, with the right preprocessing and feature extraction methods. Likewise, the proposed methodology achieved 98% of accuracy under the light implementation which is based on Arabic text preprocessing and TF-IDF feature extraction. The proposed

framework outperformed the (DL) models including CNN-, BiLSTM- and transformer-based models with slightly lower predictive performance, but with significantly lower computational complexity and training time. This makes the proposed approach an appropriate approach in resource constrained environments where computational efficiency is a critical concern. Comparisons with previous studies should be used with caution due to differences in the studies with regard to data, types of communications (SMS, email and social media), languages, preprocessing type of data, feature extraction method, configuration of classifiers and the evaluation method. Because of this, the reported performance metrics cannot be directly compared and comparisons made in this study are meant to give the general context of the performance, not to provide performance ranking.

Table 7 Comparison with Previous Studies

Study	Data Type	Main Method	Main Findings	Relation to the Present Study
Zawari (2025)	Arabic SMS	Ensemble ML	Accuracy 91.89%, F1-score 91.55%	The proposed SVM and RF achieved higher performance on Arabic phishing email classification.
Bani Younes & Ababneh (2026)	SMS, Email, Social Media	NB, SVM, DT, RF	Good performance but limited generalization	Similar ML algorithms were evaluated, although the present study focuses

				specifically on Arabic phishing emails.
Menthe et al. (2024)	SMS	NB, SVM	Preprocessing significantly improved performance	Supports the importance of Arabic text preprocessing adopted in the proposed framework.
Syarifuddin et al. (2025)	SMS & Email	ML vs (DL)	(DL) achieved higher accuracy	The proposed framework provides competitive performance with substantially lower computational complexity.
Ahmed et al. (2022)	Email & IoT	Review	(RF) performed consistently well	Supports the selection of (RF) as one of the evaluated classifiers.
Birthare et al. (2025)	SMS	SVM, RF, LSTM, ELM	NLP preprocessing improved classification	Confirms the effectiveness of preprocessing and lightweight (MLTs).
Jurkuch et al. (2026)	SMS & Email	Bi-GRU + XGBoost	Accuracy 98.7%, AUC 99.8%	Achieved slightly higher accuracy but

				employed a more computationally intensive hybrid architecture.
--	--	--	--	--

4.10 Summary of Experimental Findings

The experiments performed in this work illustrate that the proposed Arabic phishing detection framework is highly effective if it is used along with efficient text pre-processing and appropriate feature extraction technique (TF-IDF). All three classifiers were able to produce excellent classification performance, however, with (SVM), and (RF) classifiers showing the best classification consistently. They are able to accurately detect phishing messages with zero false positive predictions, which makes them good choices for real-world Arabic phishing detection systems. Additionally, the results demonstrate how effective preprocessing and feature engineering can be to enhance the results of classic (ML) methods, allowing them to match or outperform the more complex (DL) methods.

4.11 Chapter Summary

This chapter conducted an experimental evaluation of the suggested Arabic phishing detection framework. All three (ML) algorithms were trained and tested with the same data and the same data pre-processing steps. The experimental results showed that when it comes to overall performance; (SVM), and (RF) had the best performance with accuracy 98%, perfect precision and F1 score of 94.74%. The robustness of the proposed models was further validated by the confusion matrix analysis, which showed an excellent performance in phishing detection from the

97

proposed models with no false prediction in the phishing detection. In general, the results of the experiments confirmed The efficacy of the suggested methodology and its applicability to real-world Arabic phishing detection systems.

CHAPTER FIVE CONCLUSION AND FUTURE WORK

5.1 Introduction

The overall conclusions based on the experimental investigation carried out in this research are presented in this chapter. It presents an overview, objectives, research approach, and main results obtained due to the implementation of the proposed Arabic phishing message detection system. Furthermore, the chapter outlines the study's contribution, the limitations, and some future research directions to build upon that can continue to enhance Arabic phishing detection systems.

5.2 Research Summary

With the development of digital communication, the amount of phishing attacks is growing to a significant extent, targeting users via email, SMS and online communication devices. Phishing messages written in Arabic are even more difficult to detect because the Arabic language is morphologically complex, with many spelling variations, dialectal differences, and scant amount of publicly-accessible Arabic cybersecurity datasets. In order to solve these challenges, this research presented a Lightweight Machine Learning (LML) framework to detect phishing messages in Arabic language based on (NLP). The proposed framework comprised four phases: pre-preparing the dataset, Arabic text pre-processing, TF-IDF feature extraction, (SMMLC), and performance evaluation. The experimental study used the Arabic Phishing and Legitimate Emails Dataset which includes 1250 Arabic emails, with 1000 legitimate emails and 250 phishing emails. Pre-processing was performed on the data set by text cleaning, normalization, stop-word removal, repeated character removal and tokenization before it was subjected to the TF-IDF technique for the

transformation of text into numerical feature vectors. Three (SML) algorithms, namely (NB), (SVM) and (RF) were implemented and evaluated. Same pre-processing methods were used for training the classifiers and the same testing set was used for their evaluation with the following standard performance measures: Accuracy, Precision, Recall, F1 score, and Confusion Matrix analysis.

5.3 Research Findings

The results of the experiments showed that all three (ML) algorithms were able to obtain high classification efficacy for phishing messages and Arabic messages. There were noticeable disparities between the classifiers implemented, however.

Overall, using a (NB) model, the classification accuracy was 96.8%, showing that the probabilistic learning method is still very effective for Arabic text classification, even with the right pre-processing and TF-IDF feature extraction. However, the relatively low recall rate suggests that some phishing messages were overlooked.

SVM had the most accurate performance of 98%, precision of 100%, recall of 90% and an F1-score of 94.74%. The confusion matrix also confirmed that the classifier has excellent detection capability and no false-positive predictions, having classified all the valid messages correctly and five of the phishing messages.

The performance of (RF) was almost same as (SVM), as it also obtained an accuracy of 98% and an F1-score of 94.74%. The ensemble learning mechanism led to a strong classification performance with a good generalization ability.

The results of the experiments validated the effectiveness of the proposed Arabic text preprocessing and the TFIDF feature extraction method, which boosts the performance of the traditional (ML) algorithms in general. Moreover, the results

show that it is possible to obtain competitive results that do not need to be the massive computing power of a (DL) model, using a relatively simple and efficient classifier such as SVM or Random Forest.

5.4 Research Contributions

This study contributes to the phishing detection in Arabic and NLP field in a number of significant ways.

First, it presents a full (ML) framework dedicated to detecting phishing messages in Arabic, starting from the data preprocessing stage up to the evaluation of the system performance. First, it presents an entire (ML) framework dedicated to detecting phishing messages in Arabic language, from the data preprocessing stage to the evaluation of the system performance.

Second, the study shows the efficiency of preprocessing methods used in Arabic language to boost text classification performance in the Arabic language. The preprocessing pipeline minimized textual noise and generated a quality feature representation that was appropriate for the (ML).

Third, a comprehensive comparison is made for three widely used supervised learning algorithms under the same experimental framework, allowing the comparison the advantages and limitations associated with the algorithms.

Third, the experimental results demonstrate that high detection accuracy can be obtained by using low weight (ML) models while ensuring a low computation cost. This makes the proposed framework applicable for practical deployment in real-time phishing detection application which can have limited computational resources.

Finally, this study performs experimental evaluation of (ML) techniques to detect phishing attacks based on publicly available Arabic language based datasets, which constitutes the first study in the growing body of work on Arabic cybersecurity.

5.5 Practical Implications

121 This study has a number of practical implications in the field of cybersecurity applications.

Organizations can implement the proposed framework in email filtering applications and automatically detect suspicious Arabic phishing emails before they arrive at the end user. Lightweight (ML) models can also be used by the mobile communication providers to identify malicious SMS messages in real-time.

Moreover, government organizations, financial institutions, and online service providers can leverage Arabic phishing detection systems that leverage automated technologies to mitigate the risks posed by social engineering attacks in their cybersecurity strategies.

The proposed framework is not based on an expensive and heavy (DL) architecture, and can be efficiently deployed on systems with limited hardware resources while retaining high detection performance.

5.6 Research Limitations

127
1 This study has a few limitations, although results were encouraging. First, the number of emails used in the experiments was relatively small, 1,250 emails of the Arabic language, which might restrict the generalizability of the results. 119
5 Second, the data set is skewed (1000 legitimate emails and 250 phishing emails), which can affect the results of the classifiers. Thirdly, the evaluation was not performed by cross

validation, but by a single stratified train-test split. Moreover, the proposed framework is based on TF-IDF feature extraction, which does not involve the use of the Arabic language models, such as AraBERT, or any other transformer-based models. Lastly, the framework has not been tested in an offline experiments environment yet and it has not been deployed in real-world environments.

5.7 Future Work

Future work should validate the proposed framework by testing it with more extensive and varied Arabic phishing email data sets from various sources in the wild. Future research can explore the use of more sophisticated contextual embedding methods such as transformer-based language models, and hybrid (ML) and (DL) models to enhance the accuracy of phishing detection. Additional experiments may also include external dataset assessment, cross-validation and deployments experiments to test the computational efficiency, scalability and applicability of the experiments in an operational cyber security environment.

5.8 Chapter Summary

The final result of the research was given in this chapter and the main experimental results that were achieved from the proposed Arabic phishing detection framework were summarized. In this study the effectiveness of traditional (ML) techniques that exhibited high performance with effective Arabic text preprocessing and TF-IDF feature extraction were presented. The proposed framework achieved a highest rate of classification accuracy of 98%, showing its efficacy as a light and effective solution to detect phishing in Arabic. The chapter also discussed the key scientific findings of the research, the practical implications of the findings, areas for

limitations and possible recommendations for future research. In conclusion, these results indicate that if well-designed, machine-learning systems can remain competitive in the Arabic language phishing detection challenge, and can provide valuable knowledge for future improvement of more sophisticated intelligent cyber-security systems.

References

- [1] Wahsheh, H., Abu Doush, I., Al-Kabi, M., Alsmadi, I., & Al-Shawakfa, E. (2012). Using machine learning algorithms to detect content-based Arabic web spam. *Journal of Information Assurance and Security*, 7, 14–23. https://www.researchgate.net/publication/221931557_Using_Machine_Learning_Algorithms_to_Detect_Content-based_Arabic_Web_Spam
- [2] Hassan, S. I., Elrefaei, L., & Andraws, M. S. (2023). Arabic tweets spam detection based on various supervised machine learning and deep learning classifiers. *MSA Engineering Journal*, 2(2), 1099-1119. DOI: [10.21608/msaeng.2023.291931](https://doi.org/10.21608/msaeng.2023.291931)

- [3] Alkadri, A. M., Elkorany, A., & Ahmed, C. (2022). Enhancing Detection of Arabic Social Spam Using Data Augmentation and Machine Learning. *Applied Sciences*, 12(22), 11388. <https://doi.org/10.3390/app122211388>
- [4] 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. <https://doi.org/10.1155/2022/1883698>
- [5] Mahmud, T., Prince, M. A. H., Ali, M. H., Hossain, M. S., & Andersson, K. (2024). Enhancing Cybersecurity: Hybrid Deep Learning Approaches to Smishing Attack Detection. *Systems*, 12(11), 490. <https://doi.org/10.3390/systems12110490>
- [6] Spamhaus Project. (2026). *What is the definition of spam?* Retrieved from [FAQs | Key definitions, including the definition of spam](#)
- [7] Hantom, W. H., & Rahman, A. (2024). Arabic Spam Tweets Classification: A Comprehensive Machine Learning Approach. *AI*, 5(3), 1049-1065. <https://doi.org/10.3390/ai5030052>
- [8] Abdhood SF, Omar N, Tiun S. Data augmentation for Arabic text classification: a review of current methods, challenges and prospective directions. *PeerJ Comput Sci*. 2025 Mar 10;11:e2685. doi: 10.7717/peerj-cs.2685. PMID: 40134861; PMCID: PMC11935767.
- [9] Siino, M., Tinnirello, I., & La Cascia, M. (2024). Is text preprocessing still worth the time? A comparative survey on the influence of popular preprocessing methods on Transformers and traditional classifiers. *Information Systems*, 121, 102342. <https://doi.org/10.1016/j.is.2023.102342>
- [10] Sutriawan, S., Rustad, S., Shidik, G. F., & Pujiono. (2025). Performance evaluation of text embedding models for ambiguity classification in Indonesian news corpus: A comparative study of TF-IDF, Word2Vec, FastText, BERT, and GPT. *Ingénierie des Systèmes d'Information*, 30(6), 1469–1482. DOI:[10.18280/isi.300606](https://doi.org/10.18280/isi.300606)

- [11] Manurung, J., & Saragih, H. (2024). Performance comparison of naive bayes and support vector machine algorithms in spambot classification in emails. *International Journal of Basic and Applied Science*, 13(3), 137–145. DOI:[10.35335/ijobas.v13i3.522](https://doi.org/10.35335/ijobas.v13i3.522)
- [12] He, B., Yang, Y., Wang, L., & Zhou, J. (2024). The text classification method based on BiLSTM and multi-scale CNN. *Computer Life*, 12(2), 43–45. DOI:[10.54097/ypxxse31](https://doi.org/10.54097/ypxxse31)
- [13] Alsufyani, S., & Alajmani, S. (2025). Detecting SMS phishing based on Arabic text-content using deep learning. *IJCSNS International Journal of Computer Science and Network Security*, 25(4), 1–10. paper.ijcsns.org
- [14] Al Saidat, M. R., Yerima, S. Y., & Shaalan, K. (2024). A novel approach for Arabic SMS spam detection using hybrid deep learning techniques. *Procedia Computer Science*, 244, 260–267. <https://linkinghub.elsevier.com/retrieve/pii/S187705092403000X>
- [15] Alzubaidi, M., & Ghanem, K. (2025). Arabic SMS spam detection based on machine learning and phishing pattern analysis. *Journal of Information Security and Applications*, 75, 103457. [Arabic SMS Spam Detection](#)
- [16] Alkaabi, M., Alshamsi, A., & Alketbi, A. (2025). Arabic SMS spam detection using AraBERT and hybrid CNN-BiLSTM architecture. *IEEE Access*, 13, 45678–45690. DOI:[10.21203/rs.3.rs-6832100/v1](https://doi.org/10.21203/rs.3.rs-6832100/v1)
- [17] Filali, S., Benbrahim, H., & Oussalah, M. (2026). Enhancing spam detection using GAN-based data augmentation and hybrid machine learning models. *Expert Systems with Applications*, 210, 118456. [\(PDF\) Cross-lingual SMS spam detection using GAN-based augmentation for imbalanced datasets](#)
- [18] Errami, A., El Madani, A., & Hain, M. (2024). Arabic tweet spam detection using machine learning and deep learning techniques. *Applied Artificial Intelligence*, 38(2), 145–162.
- [19] Jurkuch, R., Kaur, P., & Singh, D. (2026). A resource-efficient hybrid model for SMS spam detection in IoT environments. *Future Generation Computer Systems*,

150, 102–115. [Spam Detection in Arabic Tweets Using Artificial Intelligence Techniques | IEEE Conference Publication | IEEE Xplore](#)

[20] Zawari, A. H. (2025). Detecting Arabic SMS scam messages using hybrid ensemble machine learning algorithms. *International Journal of Advanced Computer Science and Applications*, 16(3), 112–120. DOI:[10.52783/jisem.v10i55s.11434](#)

[21] Bani Younes, M., & Ababneh, A. (2026). Machine Learning Based Spam Detection in Digital Communication Systems: A Comparative Analysis. *Systems*, 14(3), 229. <https://doi.org/10.3390/systems14030229>

[22] Menthe, S., Sharma, P., & Gupta, R. (2024). SMS spam detection using machine learning techniques. *International Journal of Computer Applications*, 185(12), 1–7. DOI:[10.17148/IJARCCE.2024.13307](#)

[23] Syarifuddin, M. A., Rahman, F., & Prasetyo, E. (2025). Spam detection using machine learning and deep learning techniques: A comparative study. *Journal of Big Data*, 12(1), 45–60. <https://doi.org/10.36227/techrxiv.173933255.51566942/v1>

[24] Ahmed, N., Khan, S., & Ali, R. (2022). Machine learning techniques for spam detection in email and IoT platforms: A review. *IEEE Access*, 10, 34567–34580. DOI:[10.1155/2022/1862888](#)

[25] Birthare, M., Jain, N., & Meena, A. (2025). Machine learning techniques on mobile SMS spam detection. *International Journal of Engineering Research & Technology*, 14(5), 234–240. [Machine Learning Techniques on Mobile SMS Spam Detection | Atlantis Press](#)

[26] Julis, M. R., & Alagesan, S. (2020). Spam detection in SMS using machine learning through text mining. *International Journal of Scientific & Technology Research*, 9(2), 2277–8616. DOI:[10.17148/IJARCCE.2024.13307](#)

[27] Kumar, N., Sonowal, S., & Nishant. (2020). Email spam detection using machine learning algorithms. In *Proceedings of the 2020 Second International Conference on Advances in Computing, Communication and Control*. IEEE. DOI:[10.1109/ICIRCA48905.2020.9183098](#)

[28] Roy, P. K., Singh, J. P., & Banerjee, S. (2020). Deep learning to filter SMS spam. *Future Generation Computer Systems*, 102, 524–533. DOI:[10.1016/j.future.2019.09.001](https://doi.org/10.1016/j.future.2019.09.001)