

Project

Course Title: Research Methodology in Computer Science
Course Code: AI800
Lecturer: Dr. Mohammed Anbar

A Hybrid and Time-Aware Machine Learning Framework for Detecting Previously Unseen Phishing Websites

Research Project Proposal

Albara Almugrn

Table of Contents

Abstract

Chapter 1: Introduction and Research Foundation

1.1 Introduction

1.2 Background

1.3 Problem Statement

1.4 Research Goal and Objectives

1.5 Research Scope

Chapter 2: Literature Review

2.1 Foundational Studies

2.2 Extended Studies

2.3 Synthesis and Research Gap

Chapter 3: Research Methodology

3.1 Research Design

3.2 Data Collection and Preparation

3.3 Model Development

3.4 Experimental Design and Analysis

3.5 Ethical Considerations

3.6 Limitations

3.7 Expected Outcomes

3.8 Research Milestones and Gantt Chart

References

Abstract

Phishing websites remain a persistent cybersecurity threat because attackers can rapidly create new URLs, imitate legitimate services, and modify their infrastructure to evade reactive blacklist-based defenses. Existing machine-learning and deep-learning studies report high detection performance, but they frequently differ in datasets, feature sources, evaluation protocols, and operational assumptions. This makes direct comparison difficult and leaves uncertainty about whether reported performance generalizes to previously unseen phishing campaigns. This research proposes a hybrid and time-aware phishing website detection framework. A lightweight URL-based model will perform initial screening, while selected complementary page and contextual features will be used for uncertain cases. The study will construct a labeled dataset of legitimate and phishing websites, preserve collection timestamps, and use chronological training, validation, and test partitions to reduce temporal leakage. Conventional machine-learning and deep-learning baselines will be evaluated under the same protocol. Performance will be measured using accuracy, precision, recall, F1-score, false-positive rate, area under the ROC curve (AUC), and average detection time. The expected contribution is a reproducible framework that balances detection effectiveness, computational efficiency, and generalization to later phishing campaigns, together with evidence on when richer features provide sufficient benefit to justify their additional processing cost.

Keywords: phishing website detection, machine learning, deep learning, malicious URL, cybersecurity, temporal evaluation, hybrid detection.

Chapter 1: Introduction and Research Foundation

1.1 Introduction

Phishing is a social-engineering attack in which an adversary impersonates a trusted organization or service to persuade a victim to disclose credentials, financial information, or other sensitive data. Websites and URLs are central to many phishing campaigns because they provide the interface through which victims are redirected and deceived. Conventional defenses such as blacklists remain useful for known malicious sites, but they are inherently reactive and may fail when attackers register new domains, generate new URL paths, or compromise legitimate infrastructure. This limitation has motivated extensive research into machine-learning and deep-learning techniques capable of learning discriminative patterns from URLs, webpage content, visual layout, and contextual information.

The research literature shows substantial progress. PhishStorm introduced real-time URL classification based on intra-URL relatedness; URLNet learned character- and word-level URL representations; page-layout methods exploited CSS similarity; multidimensional deep-learning methods combined heterogeneous indicators; PDRCNN combined recurrent and convolutional learning; and feature-selection approaches demonstrated that conventional models can remain competitive. More recent work has explored transformer architectures, temporal convolutional networks, and systematic comparisons of machine-learning and deep-learning approaches. Nevertheless, a major methodological challenge remains: high performance on a fixed benchmark does not necessarily demonstrate robust detection of future phishing campaigns.

1.2 Background

Phishing detection approaches can be broadly organized into list-based, heuristic, visual-similarity, machine-learning, and deep-learning methods. List-based approaches are computationally efficient but depend on prior observation. Heuristic methods encode expert knowledge into rules or engineered features. Machine-learning methods learn decision boundaries from features such as URL length, domain characteristics, HTML properties,

redirects, and security indicators. Deep-learning methods can reduce manual feature engineering by learning representations directly from character or token sequences.

A recurring pattern in the literature is the extensive use of public data sources. PhishTank is commonly used for phishing samples, while popular-domain lists have often supplied legitimate examples. Systematic reviews have noted this concentration of data sources and the dominance of supervised learning. Such datasets enable reproducible experimentation but can introduce sampling bias. A second issue is evaluation design. Random train-test splitting may place related URLs from the same campaign or time period in both partitions, potentially producing optimistic estimates of future performance. A time-aware evaluation, in which models are trained on earlier data and tested on later data, better reflects the operational question of whether a detector can recognize new campaigns.

The proposed research therefore focuses not merely on maximizing benchmark accuracy, but on balancing four operational properties: phishing recall, low false-positive rate, inference efficiency, and generalization to later data. A hybrid architecture is selected because URL-only analysis is fast and safe, whereas page and contextual evidence can help when the URL alone is ambiguous.

1.3 Problem Statement

Phishing website detection is a cybersecurity problem concerned with identifying fraudulent web resources before users disclose sensitive information. Reactive blacklist systems are unable to guarantee detection of newly generated phishing URLs. Machine-learning and deep-learning approaches address this limitation, but the literature reveals continuing trade-offs. URL-only models provide rapid early-stage decisions and avoid visiting suspicious pages, yet they can have limited evidence when attackers use compromised legitimate domains or carefully constructed benign-looking URLs. Page-layout and multidimensional approaches can provide richer evidence but increase data-collection latency, computational cost, and implementation complexity. Deep models reduce manual feature engineering but can be difficult to interpret, while feature-based models may depend heavily on the dataset and feature definitions used.

The significance of this problem is operational as well as academic. A detector that produces excessive false positives can disrupt legitimate browsing and reduce trust in security controls, while false negatives expose users and organizations to credential theft. The six studies reviewed in Assignments 1 and 2 report promising performance, but they use different datasets and experimental designs. Recent surveys similarly show substantial variation in data sources, algorithms, and evaluation practices. Therefore, comparing headline accuracy values alone does not establish which approach will remain effective when deployed against later phishing campaigns.

The key research gap is the limited evidence that high-performing phishing detectors generalize over time under a consistent evaluation protocol. There is a need for an approach that combines the efficiency of URL-based learning with richer evidence only when necessary, and that is evaluated using chronological data partitions and common operational metrics. Accordingly, the central research problem is the lack of a phishing website detection framework that simultaneously provides high detection effectiveness, low false-positive rates, practical computational efficiency, and demonstrable generalization to previously unseen phishing campaigns.

1.4 Research Goal and Objectives

Primary research goal: To design and evaluate a hybrid, time-aware machine-learning framework for phishing website detection that improves detection of previously unseen phishing websites while maintaining a low false-positive rate and practical computational efficiency.

- To identify and categorize URL, page, and contextual features used in the reviewed phishing-detection literature and map their limitations to the proposed model design.

- To construct and preprocess a labeled phishing/legitimate website dataset with collection timestamps and chronological training, validation, and testing partitions.
- To develop representative baseline models, including a conventional feature-based classifier and a URL-based deep-learning classifier.
- To design a two-stage hybrid model in which URL-based classification is performed first and selected complementary features are invoked for low-confidence cases.
- To evaluate all models under the same experimental protocol using accuracy, precision, recall, F1-score, false-positive rate, AUC, and average detection time.
- To assess temporal generalization by comparing performance on later-period test data with performance obtained from a conventional random split.
- To determine whether the hybrid model provides a statistically and operationally meaningful improvement over the selected baselines.

1.5 Research Scope

The study is limited to binary classification of phishing and legitimate websites. The primary detection object is the URL, supplemented by selected webpage and contextual indicators for uncertain cases. The research will not attempt to identify the attacker, perform malware analysis, or address voice phishing and purely text-based phishing emails that contain no web link. The experimental work will use publicly obtainable research datasets or feeds and will avoid interaction with live phishing pages wherever possible.

The scope includes feature extraction, preprocessing, model training, threshold selection, chronological evaluation, baseline comparison, and statistical analysis. The proposed framework is intended as a research prototype rather than a production security product. Its conclusions will therefore focus on comparative detection performance and generalization within the collected datasets.

Chapter 2: Literature Review

2.1 Foundational Studies from Assignment 1

Marchal et al. (2014) – PhishStorm

PhishStorm proposes proactive real-time phishing URL detection using intra-URL relatedness. Search-engine query information is used to derive relationships among URL words, which are then supplied to machine-learning classifiers. The study evaluated 96,018 URLs and reported 94.91% correct classification with a 1.44% false-positive rate. Its major strength is its explicit real-time orientation and substantial dataset. Its dependence on external search services, however, can affect latency, reproducibility, and long-term maintainability.

Le et al. (2018) – URLNet

URLNet replaces much manual lexical feature engineering with learned URL representations. Character-level and word-level convolutional neural networks capture local and sequential patterns directly from raw URLs. Large-scale experiments showed gains over conventional lexical baselines. The study is important because it demonstrates end-to-end representation learning, although its scope covers malicious URLs broadly rather than phishing exclusively and the learned model is less interpretable than conventional feature-based classifiers.

Mao et al. (2019) – Page Layout Features

Mao et al. use CSS page-layout properties to learn whether a suspicious page resembles a known legitimate target. SVM, Decision Tree, AdaBoost, and Random Forest classifiers are evaluated on more than 490 phishing

pages targeting 46 legitimate pages, producing more than 20,000 test samples. The method contributes a complementary page-level signal, but it requires page retrieval and may be affected by dynamic layouts, redesigns, or phishing pages that avoid close visual imitation.

Yang et al. (2019) – Multidimensional Deep Learning

Yang, Zhao, and Zeng investigate phishing detection using multidimensional features and deep learning. The approach argues that heterogeneous evidence can improve robustness compared with relying on a narrow feature family. The advantage is broader evidence; the cost is increased collection and processing complexity together with lower explainability.

Wang et al. (2019) – PDRCNN

PDRCNN combines recurrent and convolutional neural networks to learn global and local URL patterns without depending on search engines or DNS services. Using nearly 500,000 Alexa and PhishTank samples, the authors report approximately 97% accuracy, about 99% AUC, and sub-millisecond detection. This makes the model attractive for high-volume screening, although URL-only evidence can be insufficient for compromised legitimate domains and dataset-source separation can create sampling bias.

Ubing et al. (2019) – Feature Selection and Ensemble Learning

Ubing et al. study feature selection and ensemble/conventional machine learning for phishing classification, reporting accuracy up to approximately 95%. The work demonstrates that deep learning is not the only path to competitive performance and that compact feature sets can be practical. However, selected features may be dataset-specific, and accuracy alone does not characterize false-positive behavior or temporal robustness.

2.2 Extended Literature: Additional Studies

Tang and Mahmoud (2021)

This survey provides a structured overview of machine-learning-based phishing website detection, covering data collection, feature extraction, model building, and performance evaluation. Its importance to the proposed research lies in showing that phishing detection quality is shaped by the complete experimental pipeline rather than the classifier alone. The survey also supports systematic comparison of feature-based and learned approaches. As a review, however, it synthesizes heterogeneous studies rather than experimentally resolving dataset and temporal-generalization issues.

Aljofey et al. (2021)

The authors investigate lightweight URL-based phishing detection for mobile devices and evaluate neural networks together with transformer models such as BERT and ELECTRA. Their results show that transformer representations can classify phishing using URL text without requiring a visit to the suspicious website. This strengthens the case for URL-first screening. The work also highlights a trade-off relevant to this proposal: URL-only approaches improve safety and deployability but may sacrifice evidence available in webpage content.

Catal et al. (2022)

Catal and colleagues conduct a systematic literature review of deep-learning applications for phishing detection. Forty-three journal articles are synthesized. The review reports widespread use of supervised learning, URL-related data, website content, and third-party information; DNN, CNN, and RNN/LSTM models are prominent, and hybrid deep-learning models show strong performance. PhishTank is identified as the most common dataset.

This evidence supports the need to control for dataset-source dependence and to evaluate hybrid approaches under a unified protocol.

Wang and Chen (2022) – TCURL

TCURL combines a convolution branch with a transformer branch to capture both local and global URL-character relationships. The model reports accuracies of 96.92%, 99.77%, and 89.73% on three sampled datasets. The variation across datasets is especially relevant: it illustrates that performance can depend strongly on data composition. The hybrid architecture supports the proposed project's use of complementary representations, while the cross-dataset variability reinforces the need for careful generalization testing.

Aljabri et al. (2022) – Temporal Convolutional Network

This study applies a Temporal Convolutional Network with word embedding to phishing URL detection and reports 98.95% accuracy with approximately 98% precision, recall, and F1-score. It demonstrates that modern sequence architectures can achieve strong URL-only performance. Nevertheless, the dynamic nature of phishing emphasized by the authors means that strong benchmark metrics still need validation against later campaigns.

Systematic Literature Review on Phishing Website Detection Techniques (2023)

A 2023 systematic review analyzes 80 studies and compares list-based, visual-similarity, heuristic, machine-learning, and deep-learning techniques. It reports that machine learning was the most common approach and that PhishTank and Alexa were heavily used as phishing and legitimate data sources. This concentration is a critical observation for the present proposal because repeated use of the same sources can create common biases across apparently independent studies. The review therefore strengthens the justification for transparent dataset construction and time-aware evaluation.

2.3 Synthesis and Research Gap

The extended literature reveals three consistent findings. First, learned phishing detection is superior in flexibility to static blacklist-only defenses because it can classify previously unseen strings or pages. Second, representation quality matters: researchers have progressively moved from manually engineered URL features toward CNNs, recurrent networks, transformers, and hybrid architectures. Third, richer evidence can improve discrimination but increases collection and computation costs.

The literature also exposes methodological weaknesses. Public feeds and popular-domain lists are repeatedly reused; evaluation protocols differ; and high headline accuracy is often emphasized more than temporal robustness, false-positive cost, or inference latency. The reported performance variation of newer models across datasets further demonstrates that a single benchmark score is not sufficient evidence of generalization.

The proposed project addresses this gap by combining URL-first detection with selective complementary analysis and by evaluating the models chronologically. The literature review directly informs the objectives: URLNet, PDRCNN, transformers, and TCN models motivate the URL baseline; page-layout and multidimensional studies motivate complementary features; feature-selection research motivates limiting costly features; and systematic reviews motivate a unified and transparent evaluation protocol.

Chapter 3: Research Methodology

3.1 Research Design

The research will use a quantitative experimental design supported by a structured literature review. The independent experimental factors will include model type, feature set, decision threshold, and data-splitting

strategy. The dependent measures will be accuracy, precision, recall, F1-score, false-positive rate, AUC, and average detection time. The principal comparison will be between a conventional feature-based baseline, a URL-only deep-learning baseline, and the proposed two-stage hybrid model.

3.2 Data Collection and Preparation

Phishing samples will be obtained from a reputable public phishing feed such as PhishTank or an equivalent archived research source. Legitimate samples will be collected from a reputable ranked-domain or benign-URL source. Each observation will include a label, URL, collection timestamp where available, and the selected complementary attributes. Duplicate URLs will be removed and exact overlaps between classes will be investigated. Data will be normalized without destroying discriminative URL structure.

To reduce temporal leakage, the primary experiment will use chronological partitions: earlier samples for training, a subsequent period for validation and threshold selection, and the latest period for final testing. A secondary random stratified split will be retained only as a comparison so that the effect of evaluation strategy can be quantified.

3.3 Feature Engineering and Model Development

The URL component will use raw character/token representations and a compact set of lexical indicators such as URL length, number of subdomains, digit and special-character ratios, suspicious token patterns, and path/query complexity. Complementary features will be restricted to indicators that can be obtained safely and consistently, for example selected HTML metadata, redirect behavior from archived records, or domain/context attributes where available.

Baseline 1 will use a conventional classifier such as Random Forest or Gradient Boosting on engineered features. Baseline 2 will use a URL sequence model such as a CNN-based architecture. The proposed hybrid will first calculate a URL-based phishing probability. High-confidence decisions will terminate at stage one; samples within a predefined uncertainty interval will proceed to stage two, where complementary features will be fused with the URL representation or classified by a second model.

3.4 Experimental Procedure and Data Analysis

- Preprocess the dataset and document class distribution, missing values, duplicates, and source composition.
- Train all baseline models using only the training partition; tune hyperparameters using the validation partition.
- Select the hybrid model's confidence thresholds on validation data without using final test labels.
- Evaluate every model on the same chronological test set and repeat the analysis using a random stratified split for comparison.
- Calculate confusion matrices and the full metric set: accuracy, precision, recall, F1-score, false-positive rate, AUC, and mean/median inference time.
- Use bootstrap confidence intervals for key performance differences and, where predictions are paired, apply an appropriate paired test such as McNemar's test to assess whether error differences are statistically meaningful.
- Conduct an ablation analysis by removing complementary feature groups to determine which groups materially improve the hybrid model.
- Report both predictive gains and computational cost so that improvements are not judged by accuracy alone.

Python will be used for preprocessing and analysis. Suitable libraries include pandas and NumPy for data preparation, scikit-learn for conventional machine learning and metrics, and TensorFlow/Keras or PyTorch for

deep-learning experiments. Experiments will use fixed random seeds where applicable and configuration details will be documented to improve reproducibility.

3.5 Mapping Research Objectives to Methods

Objective	Method	Evidence / Measure
Identify feature categories and limitations	Structured literature synthesis and feature taxonomy	Feature-to-limitation matrix
Construct time-aware dataset	Data collection, cleaning, timestamp preservation, chronological split	Dataset profile and split statistics
Develop baselines	Random Forest/Gradient Boosting and URL deep model	Validation and test metrics
Design hybrid model	Confidence-gated two-stage architecture	Hybrid architecture and threshold rules
Evaluate consistently	Common test set and metrics	Accuracy, precision, recall, F1, FPR, AUC, latency
Assess temporal generalization	Chronological versus random-split experiment	Performance degradation/gain across split strategies
Determine meaningful improvement	Paired error analysis, confidence intervals, ablation	Statistical significance and operational cost

3.6 Ethical Considerations

The project will minimize interaction with active phishing infrastructure. Where possible, analysis will use URLs, archived metadata, or safely collected attributes rather than manually browsing live malicious pages. No stolen credentials or personal victim data will be collected. Research data will be stored securely and used only for defensive analysis. Any publicly shared dataset derived from the project will exclude sensitive personal information and will follow the terms of the original data sources.

3.7 Limitations

The study may be limited by the availability and accuracy of timestamps, the representativeness of public phishing feeds, class-label errors, and the ability to obtain consistent complementary features. The proposed framework is evaluated as a research prototype; production deployment may introduce network latency, browser behavior, adversarial adaptation, and infrastructure constraints not fully represented in the experiment.

3.8 Expected Outcomes

- A clearly documented, time-aware phishing/legitimate dataset suitable for comparative experiments.
- A reproducible comparison of conventional feature-based, URL deep-learning, and hybrid detection approaches.
- Evidence showing how much performance changes when evaluation moves from random splitting to chronological testing.
- A two-stage hybrid framework that aims to preserve URL-only speed for confident cases while using richer evidence for ambiguous cases.
- Quantified trade-offs among phishing recall, false-positive rate, AUC, and detection latency.
- Identification of the complementary feature groups that provide the greatest benefit through ablation analysis.
- Practical recommendations for designing and evaluating phishing detectors intended to operate against evolving campaigns.

3.9 Research Milestones and Gantt Chart

The following plan covers the project period from the hand-out date to the submission date. Some activities overlap because literature extension, data preparation, and writing are iterative.

Milestone	Start	End
Finalize research questions and extended literature review	18 Jul	22 Jul
Dataset acquisition and cleaning	20 Jul	26 Jul
Feature extraction and preprocessing	24 Jul	30 Jul
Baseline model implementation	28 Jul	3 Aug
Hybrid model design and implementation	1 Aug	6 Aug
Model training and hyperparameter tuning	4 Aug	8 Aug
Chronological evaluation and statistical analysis	7 Aug	10 Aug
Ablation analysis and interpretation	9 Aug	11 Aug
Draft Chapters 1–3 and integrate results plan	10 Aug	13 Aug
Editing, citation check, formatting and final QA	13 Aug	14 Aug
Final submission	15 Aug	15 Aug

Weekly Gantt View

Activity	18–24 Jul	25–31 Jul	1–7 Aug	8–14 Aug	15 Aug
Literature / research questions	■	■			
Data acquisition / cleaning	■	■			
Feature extraction		■			
Baseline models		■	■		
Hybrid model			■		
Training / tuning			■	■	
Evaluation / statistics			■	■	
Ablation / interpretation				■	
Report writing / revision				■	
Submission					■

References

- Aljabri, M., Altamimi, H. S., Albelali, S. A., Al-Harbi, M., Alhuraib, H. T., Alotaibi, N. K., Alahmadi, A. A., Alhaidari, F., Mohammad, R. M. A., & Salah, K. (2022). Detection of phishing URLs using temporal convolutional network. *Procedia Computer Science*, 212, 74–82. <https://doi.org/10.1016/j.procs.2022.10.209>
- Aljofey, A., Jiang, Q., Qu, Q., Huang, M., & Niyigena, J.-P. (2021). An effective phishing detection model based on character level convolutional neural network from URL. *Electronics*, 9/related literature; and lightweight URL-based phishing detection using NLP transformers for mobile devices, *Procedia Computer Science*, 191, 127–134. <https://doi.org/10.1016/j.procs.2021.07.040>
- Catal, C., Giray, G., Tekinerdogan, B., Kumar, S., & Shukla, S. (2022). Applications of deep learning for phishing detection: A systematic literature review. *Knowledge and Information Systems*, 64, 1457–1500. <https://doi.org/10.1007/s10115-022-01672-x>
- Le, H., Pham, Q., Sahoo, D., & Hoi, S. C. H. (2018). URLNet: Learning a URL representation with deep learning for malicious URL detection. *arXiv:1802.03162*.
- Mao, J., Bian, J., Tian, W., Zhu, S., Wei, T., Li, A., & Liang, Z. (2019). Phishing page detection via learning classifiers from page layout feature. *EURASIP Journal on Wireless Communications and Networking*, 2019, 43. <https://doi.org/10.1186/s13638-019-1361-0>
- Marchal, S., François, J., State, R., & Engel, T. (2014). PhishStorm: Detecting phishing with streaming analytics. *IEEE Transactions on Network and Service Management*, 11(4), 458–471. <https://doi.org/10.1109/TNSM.2014.2377295>
- Tang, L., & Mahmoud, Q. H. (2021). A survey of machine learning-based solutions for phishing website detection. *Machine Learning and Knowledge Extraction*, 3(3), 672–694. <https://doi.org/10.3390/make3030034>
- Ubing, A. A., Jasmi, S. K. B., Abdullah, A., Jhanjhi, N. Z., & Supramaniam, M. (2019). Phishing website detection: An improved accuracy through feature selection and ensemble learning. *International Journal of Advanced Computer Science and Applications*, 10(1). <https://doi.org/10.14569/IJACSA.2019.0100133>
- Wang, C., & Chen, Y. (2022). TCURL: Exploring hybrid transformer and convolutional neural network on phishing URL detection. *Knowledge-Based Systems*, 258, 109955. <https://doi.org/10.1016/j.knosys.2022.109955>
- Wang, W., Zhang, F., Luo, X., & Zhang, S. (2019). PDRCNN: Precise phishing detection with recurrent convolutional neural networks. *Security and Communication Networks*, 2019, 2595794. <https://doi.org/10.1155/2019/2595794>
- Yang, P., Zhao, G., & Zeng, P. (2019). Phishing website detection based on multidimensional features driven by deep learning. *IEEE Access*, 7, 15196–15209. <https://doi.org/10.1109/ACCESS.2019.2892066>
- A systematic literature review on phishing website detection techniques. (2023). *Journal of King Saud University - Computer and Information Sciences*, 35(2), 590–611. <https://doi.org/10.1016/j.jksuci.2023.01.004>

Assignment 1

Course Title: Research Methodology in Computer Science
Course Code: AI800

Assignment Report: Literature Review

Machine Learning and Deep Learning Techniques for Phishing Website Detection

Prepared for: AI800 - Research Methodology in Computer Science

Submission Type: Assignment 1 - Literature Review of Six Research Papers

Academic Literature Review

Table of Contents

1. Research Title and Scope	3
2. Summary of the Selected Research Papers	3
3. Critical Analysis	7
4. Connections Between the Findings	9
5. Conflicts and Differences Between the Findings	10
6. Synthesis of the Literature	11
7. Synthesis Matrix for Literature Review	12
8. Conclusion	14
References	15

1. Research Title and Scope

Selected research title: Machine Learning and Deep Learning Techniques for Phishing Website Detection

Phishing remains a persistent cybersecurity problem because attackers can rapidly create new websites and URLs that imitate legitimate services. Traditional blacklists are useful for known threats but are inherently reactive. The six papers reviewed in this report investigate data-driven alternatives, including lexical URL analysis, page-layout similarity, feature selection and ensemble learning, and deep neural networks. The review focuses on how each study defines the detection problem, the data and features it uses, the learning method employed, the reported findings, and the limitations that affect practical deployment.

2. Summary of the Selected Research Papers

Paper 1: PhishStorm: Detecting Phishing With Streaming Analytics (Marchal et al., 2014)

Objectives and methods. The objective of PhishStorm is to provide proactive, real-time phishing URL detection rather than relying only on reactive blacklists. The authors introduce the concept of intra-URL relatedness, which measures relationships between words in the registered domain and other URL components. Features are derived using search-engine query data and then supplied to machine-learning classifiers. The system is also designed for streaming implementation and can interface with email servers or HTTP proxies.

Key findings. The evaluation used 96,018 phishing and legitimate URLs. The reported correct classification rate was 94.91% with a 1.44% false-positive rate. The paper also discusses a phishingness rating extension with high-confidence decisions and implementation patterns based on streaming/big-data architectures.

Critical evaluation. A major strength is the emphasis on real-time operation and a substantial real-world URL dataset. The work also moves beyond simple lexical counts by modeling semantic relatedness within a URL. However, the approach depends on external search-engine query information, which can introduce availability, latency, reproducibility, and long-term maintainability concerns. Because phishing tactics and search ecosystems change over time, performance measured on the original dataset may not represent present-day traffic without retraining and revalidation.

Paper 2: URLNet: Learning a URL Representation with Deep Learning for Malicious URL Detection (Le et al., 2018)

Objectives and methods. URLNet addresses limitations of manually engineered lexical features by learning URL representations directly from raw URL strings. The framework applies convolutional neural networks to character-level and word-level URL representations and jointly learns embeddings that capture sequential and semantic patterns. It also introduces techniques for handling rare or previously unseen URL words.

Key findings. The authors report extensive experiments on a large-scale malicious-URL dataset and show that URLNet improves performance over traditional machine-learning approaches based on bag-of-words and expert-designed lexical features. Ablation experiments are used to examine the contribution of model components.

Critical evaluation. The principal strength is end-to-end representation learning, which reduces dependence on hand-crafted feature engineering and improves handling of previously unseen patterns. The main limitation for this review is scope: URLNet targets malicious URLs broadly, not phishing websites exclusively, so some malicious classes may differ from phishing behavior. Deep models also require more training resources and can be less interpretable than feature-based classifiers, which may complicate operational explanation of alerts.

Paper 3: Phishing Page Detection via Learning Classifiers from Page Layout Feature (Mao et al., 2019)

Objectives and methods. This study investigates whether machine learning can automate phishing detection from visual page-layout similarity. The method extracts Cascading Style Sheets (CSS) layout properties, creates comparison vectors for pairs of pages, and trains classifiers to determine whether a suspicious page is visually similar to a known target. Four classifiers are evaluated: Support Vector Machine, Decision Tree, AdaBoost, and Random Forest.

Key findings. The evaluation used more than 490 phishing pages from PhishTank that mimicked 46 target pages and produced more than 20,000 testing samples. The experiments demonstrate that learned page-layout similarity can support effective phishing detection and show how classifier performance is affected by dataset size and sample distribution.

Critical evaluation. The study is strong because it uses real phishing pages and explores a feature family that is different from URL-only detection. It also compares multiple classifiers rather than reporting a single model. Its main weakness is operational cost: a page generally has to be retrieved and its CSS/layout processed, which is slower and potentially riskier than URL-only screening. The approach may also be challenged by highly dynamic pages, script-generated layouts, legitimate redesigns, or phishing pages that deliberately avoid close visual similarity.

Paper 4: Phishing Website Detection Based on Multidimensional Features Driven by Deep Learning (Yang, Zhao, & Zeng, 2019)

Objectives and methods. The paper proposes a deep-learning phishing detection approach intended to reduce reliance on prior expert knowledge in feature engineering. Rather than using only one evidence source, the method combines multidimensional information associated with phishing websites and uses deep learning to learn discriminative representations.

Key findings. The authors report that the multidimensional deep-learning approach achieves strong detection performance and is designed to improve robustness compared with methods that rely on a narrow, manually selected feature set. The study demonstrates the value of combining heterogeneous phishing indicators instead of treating the URL,

page content, or other signals in isolation.

Critical evaluation. A key strength is feature diversity: combining several dimensions can make evasion more difficult than attacking a single feature family. The trade-off is complexity. Multidimensional collection may require more processing, more dependencies, and more complete page access than URL-only methods. The model can also inherit dataset-specific biases, and deep-learning decisions are harder to explain. These issues matter in enterprise environments where latency, explainability, and reliable feature availability are important.

Paper 5: PDRCNN: Precise Phishing Detection with Recurrent Convolutional Neural Networks (Wang et al., 2019)

Objectives and methods. PDRCNN proposes a phishing detection model that uses only URL information and does not depend on search engines or DNS services. The architecture combines a recurrent neural network, used to capture global sequential information, with a convolutional neural network, used to extract local URL patterns. This design aims to obtain both speed and accuracy while avoiding external information sources.

Key findings. The study constructed a large dataset from Alexa and PhishTank with nearly 500,000 samples. It reports approximately 97% accuracy, an AUC of about 99%, and detection time below 0.4 milliseconds per URL. The authors compare the model with baseline approaches and report better phishing URL detection than the tested machine-learning and n-gram methods.

Critical evaluation. The paper's strongest feature is the combination of large-scale data, URL-only operation, and very low reported inference time. This makes the approach attractive for gateways and high-volume screening. However, URL-only detection can miss attacks hosted on compromised legitimate domains or URLs deliberately designed to appear benign. The dataset sources can also create sampling bias: Alexa represents popular legitimate sites, while PhishTank represents reported phishing, which may make the two classes more separable than difficult real-world traffic.

Paper 6: Phishing Website Detection: An Improved Accuracy through Feature Selection and Ensemble Learning (Ubing et al., 2019)

Objectives and methods. This research studies phishing website classification using feature selection and ensemble learning. The authors investigate whether selecting a more informative subset of phishing indicators and combining learning models can improve predictive accuracy while avoiding unnecessary features. The study considers common classifiers including Random Forest and Logistic Regression within its experimental comparison.

Key findings. The proposed approach reports an accuracy of up to approximately 95%, showing that carefully selected features and ensemble learning can provide competitive phishing detection without requiring a deep neural architecture.

Critical evaluation. The study's strength is its practical focus on reducing the feature set and using conventional learning algorithms that are generally easier to train and explain than deep models. However, reported accuracy alone does not fully characterize

deployment quality; false positives, class balance, temporal drift, and performance on newly emerging phishing campaigns are equally important. Feature-selection results may also be dataset-specific, so a feature subset that performs well on one benchmark may not transfer unchanged to another environment.

3. Critical Analysis

Across the six studies, the most important methodological difference is the source of evidence used to decide whether a website is malicious. PhishStorm uses URL semantics enriched by search-engine information; URLNet and PDRCNN learn directly from URL strings; Mao et al. focus on CSS-based visual layout; Yang et al. combine multidimensional features; and Ubing et al. emphasize feature selection and ensemble classification. This diversity is valuable because it demonstrates that phishing detection is not a single-feature problem.

Strengths of the literature. The reviewed work shows a clear progression from manually defined features toward representation learning. Several studies use large datasets, and PDRCNN and URLNet explicitly target scalable URL processing. Mao et al. provide a complementary perspective by detecting imitation at the page-layout level, while Yang et al. argue for combining multiple evidence dimensions. Together, these studies provide several defensible detection strategies rather than a single algorithmic answer.

Methodological weaknesses. A recurring weakness is that datasets are often constructed from convenient public sources. PhishTank is useful because it contains reported phishing URLs, but reported samples may not represent unreported, short-lived, targeted, or newly created campaigns. Likewise, popular-site lists can produce a legitimate class that is cleaner than ordinary web traffic. This can inflate measured separability. Random train-test splits may also place highly related URLs from the same campaign in both sets, making generalization appear stronger than it would be in a time-separated deployment.

Evaluation metrics. Accuracy is widely reported, but accuracy alone can be misleading when class distributions are imbalanced. For operational anti-phishing systems, false-positive rate, recall for phishing, precision, AUC, latency, resource consumption, and performance under temporal drift are all important. PhishStorm reports false positives, and PDRCNN reports both accuracy and AUC together with inference speed, which makes their evaluations more operationally informative.

Reproducibility and deployment. URL-only methods are easier to deploy at email gateways, secure web gateways, and browser filtering layers because they can make a decision before fully loading a page. Layout and multidimensional approaches can provide richer evidence but introduce additional data collection and processing. Deep-learning methods reduce manual feature engineering but increase model complexity and can reduce interpretability. Therefore, the highest laboratory accuracy is not automatically the best enterprise solution; deployment constraints and the cost of false positives must be considered.

4. Connections Between the Findings

The papers share several consistent findings. First, all challenge the sufficiency of blacklist-only defenses for previously unseen phishing. Second, they show that machine learning can extract useful patterns from either URLs or webpage characteristics. Third, feature quality is central to performance: PhishStorm models intra-URL relationships, URLNet and PDRCNN learn URL representations, Mao et al. learn layout similarity, Yang

et al. combine multiple dimensions, and Ubing et al. select informative features. Fourth, the studies generally support automated detection as a complement to conventional reputation and blacklist systems rather than as a purely manual security process.

There is also a conceptual connection between feature engineering and deep learning. The earlier and conventional approaches explicitly define indicators and then train a classifier, whereas URLNet and PDRCNN attempt to learn useful representations from raw strings. The research trend therefore moves from 'design the features, then learn the classifier' toward 'learn both the representation and the classifier.'

5. Conflicts and Differences Between the Findings

The main conflict is not that one paper proves another wrong; rather, the studies make different trade-offs. URL-only approaches favor speed and early blocking, while layout-based and multidimensional methods favor richer context. PhishStorm uses external search information, whereas PDRCNN explicitly avoids third-party services. URLNet and PDRCNN favor deep representation learning, while Ubing et al. show that conventional classifiers with feature selection can still achieve strong results. Mao et al. assume that visual similarity to a target is a valuable signal, but URL-focused systems do not require a page to imitate a known visual layout.

The papers also differ in what they optimize. Some emphasize predictive accuracy, some emphasize real-time detection, and others emphasize robustness through richer features. These differences mean that direct ranking by a single accuracy percentage would be methodologically weak because datasets, class definitions, feature availability, and evaluation protocols are not identical.

6. Synthesis of the Literature

Synthesizing the six papers suggests that no single evidence source is sufficient for all phishing scenarios. URL-only deep models are attractive as a first-stage filter because they are fast and can operate before page retrieval. However, difficult cases - such as phishing on compromised domains or carefully crafted benign-looking URLs - may require page-level, layout, reputation, or contextual evidence. A practical research direction is therefore a layered detector: a lightweight URL model performs initial screening, while uncertain cases are escalated to richer multidimensional or visual analysis.

The literature also exposes an important research gap: generalization over time. Most reported experiments evaluate a fixed dataset, but phishing campaigns evolve rapidly. Future work should use temporally separated training and testing, cross-source validation, explicit measurement of concept drift, and evaluation on previously unseen campaigns. This would provide stronger evidence that high laboratory performance translates into durable real-world protection.

7. Synthesis Matrix for Literature Review

Study	Year	Authors	Methodology	Key findings	Weaknesses / limitations
PhishStorm: Detecting Phishing With Streaming Analytics	2014	Marchal, François, State & Engel	Intra-URL relatedness; search-engine query features; ML classification; streaming architecture.	96,018 URLs; 94.91% correct classification; 1.44% false positives; designed for real-time use.	Depends on external search information; temporal drift and reproducibility concerns.
URLNet: Learning a URL Representation with Deep Learning for Malicious URL Detection	2018	Le, Pham, Sahoo & Hoi	End-to-end CNN using character- and word-level URL representations and learned embeddings.	Large-scale experiments show gains over conventional lexical ML and hand-crafted features.	Broader malicious-URL scope, not phishing-only; deep model interpretability and training cost.
Phishing Page Detection via Learning Classifiers from Page Layout Feature	2019	Mao et al.	CSS layout features; comparison vectors; SVM, Decision Tree, AdaBoost and Random Forest.	>490 phishing pages, 46 target pages and >20,000 testing samples; effective layout-similarity detection.	Requires page/CSS retrieval; dynamic or redesigned pages may reduce reliability.
Phishing Website Detection Based on Multidimensional Features Driven by Deep Learning	2019	Yang, Zhao & Zeng	Deep learning over multidimensional phishing-related features.	Shows value of combining heterogeneous indicators and reducing dependence on narrow manual features.	Higher collection/processing complexity; explainability and feature availability concerns.
PDRCNN: Precise Phishing Detection with Recurrent Convolutional Neural Networks	2019	Wang et al.	URL-only hybrid RNN + CNN; no search-engine or DNS dependency.	Nearly 500,000 samples; ~97% accuracy; ~99% AUC; <0.4 ms detection per URL.	URL-only blind spots; Alexa/PhishTank sampling may not represent difficult production traffic.
Phishing Website Detection: Improved Accuracy through Feature Selection and Ensemble Learning	2019	Ubing et al.	Feature selection with conventional/ensemble ML, including Random Forest and Logistic Regression.	Reports accuracy up to ~95%; demonstrates competitive performance without deep learning.	Feature subset may be dataset-specific; accuracy alone is insufficient for deployment assessment.

8. Conclusion

The six reviewed studies demonstrate that machine learning and deep learning provide effective alternatives to purely reactive phishing blacklists. The evidence supports several viable strategies: semantic URL analysis, end-to-end URL representation learning, visual-layout classification, multidimensional feature learning, and feature-selected ensemble models. The strongest overall conclusion is that performance depends not only on the learning algorithm but also on the quality, representativeness, and freshness of the data.

For future research, a hybrid architecture appears especially promising: fast URL-based screening can handle large traffic volumes, while uncertain cases can be analyzed using richer page-level features. Such research should be evaluated using time-separated datasets and operational metrics, particularly false-positive rate, phishing recall, latency, resource use, and resilience to evolving attacks.

References

1. Le, H., Pham, Q., Sahoo, D., & Hoi, S. C. H. (2018). URLNet: Learning a URL representation with deep learning for malicious URL detection. arXiv:1802.03162.
2. Mao, J., Bian, J., Tian, W., Zhu, S., Wei, T., Li, A., & Liang, Z. (2019). Phishing page detection via learning classifiers from page layout feature. *EURASIP Journal on Wireless Communications and Networking*, 2019, 43. <https://doi.org/10.1186/s13638-019-1361-0>
3. Marchal, S., François, J., State, R., & Engel, T. (2014). PhishStorm: Detecting phishing with streaming analytics. *IEEE Transactions on Network and Service Management*, 11(4), 458-471. <https://doi.org/10.1109/TNSM.2014.2377295>
4. Ubing, A. A., Jasmi, S. K. B., Abdullah, A., Jhanjhi, N. Z., & Supramaniam, M. (2019). Phishing website detection: An improved accuracy through feature selection and ensemble learning. *International Journal of Advanced Computer Science and Applications*, 10(1). <https://doi.org/10.14569/IJACSA.2019.0100133>
5. Wang, W., et al. (2019). PDRCNN: Precise phishing detection with recurrent convolutional neural networks. *Security and Communication Networks*, 2019, 2595794. <https://doi.org/10.1155/2019/2595794>
6. Yang, P., Zhao, G., & Zeng, P. (2019). Phishing website detection based on multidimensional features driven by deep learning. *IEEE Access*, 7, 15196-15209. <https://doi.org/10.1109/ACCESS.2019.2892066>

Note on synthesis: Critical observations in this report are analytical evaluations of the cited studies, especially where they concern deployment trade-offs, dataset bias, temporal drift, and generalizability.

Assignment 2

Course Title: Research Methodology in Computer Science

Course Code: AI800

Weighting: 30%

Assignment 2

Machine Learning and Deep Learning Techniques for Phishing Website Detection

Report Components

Research Title - Problem Statement - Research Goal and Objectives - Literature
Review

Academic Report

Table of Contents

1. Research Title	3
2. Problem Statement	3
3. Research Goal and Objectives	4
4. Literature Review	5
5. Critical Analysis	9
6. Connections and Conflicts	10
7. Synthesis Matrix	11
8. Synthesis and Research Gap	13
References	14

1. Research Title

Machine Learning and Deep Learning Techniques for Phishing Website Detection

2. Problem Statement

Problem domain. Phishing website detection is a cybersecurity problem concerned with identifying fraudulent web resources that imitate legitimate services in order to obtain credentials, financial information, or other sensitive data. A major difficulty is that phishing websites can be created, modified, and removed quickly. Consequently, defenses that depend mainly on previously observed URLs and reactive blacklists can fail to identify newly generated phishing sites. Marchal et al. (2014) explicitly identify the short lifetime of phishing websites as a limitation of reactive URL blacklisting and motivate proactive, real-time detection. Le et al. (2018) similarly argue that blacklists cannot be exhaustive and cannot detect newly generated malicious URLs.

Description of the problem. Machine-learning approaches have improved proactive detection, but the reviewed literature shows that the problem is not fully resolved. Feature-engineered approaches require the designer to decide which URL, content, reputation, or page characteristics are important. Deep-learning approaches such as URLNet and PDRCNN reduce manual feature engineering by learning representations from URL strings, but they introduce model complexity and may be difficult to interpret. Other approaches use page-layout or multidimensional information, which can provide richer evidence but require additional page retrieval and processing. Therefore, phishing detection involves a continuing trade-off among detection performance, false positives, computational cost, feature availability, and ability to generalize to unseen attacks (Le et al., 2018; Mao et al., 2019; Wang et al., 2019).

Significance. The problem is significant because a phishing detector must identify malicious sites while avoiding unnecessary blocking of legitimate websites. In practical environments such as email gateways, secure web gateways, and browser protection, a high false-positive rate can disrupt legitimate activity, while false negatives can expose users to credential theft. The reviewed studies report strong classification results, including 94.91% correct classification with a 1.44% false-positive rate for PhishStorm and approximately 97% accuracy with an AUC of about 99% for PDRCNN, demonstrating the potential of machine learning while also showing that no method eliminates classification error (Marchal et al., 2014; Wang et al., 2019).

Existing solutions. Existing research addresses the problem through several strategies. PhishStorm uses intra-URL relatedness and machine-learning classification for real-time phishing detection (Marchal et al., 2014). URLNet applies convolutional neural networks at character and word levels to learn URL representations directly from raw strings (Le et al., 2018). Mao et al. (2019) use CSS page-layout features with classifiers such as SVM, Decision Tree, AdaBoost, and Random Forest. Yang et al. (2019) investigate multidimensional features driven by deep learning. PDRCNN combines recurrent and convolutional neural networks while using only URL information (Wang et al., 2019). Ubung et al. (2019) demonstrate a feature-selection and ensemble-learning approach using conventional machine-learning models.

Need for further research. Although these approaches achieve promising results, their reported performance is based on different datasets, feature sources, and evaluation protocols. This makes direct comparison difficult and leaves an important gap concerning generalization to new phishing campaigns. A model can perform well on a fixed dataset yet lose effectiveness when URL patterns, hosting infrastructure, page designs, or attacker strategies change. There is therefore a need to investigate a phishing detection approach that combines the speed of URL-based learning with carefully selected complementary features, and to evaluate it using consistent metrics and temporally separated data so that performance on previously unseen phishing campaigns can be measured.

Research problem. The central research problem is the lack of a phishing website detection approach that simultaneously provides high detection effectiveness, low false-positive rates, practical computational efficiency, and reliable generalization to previously unseen phishing campaigns.

3. Research Goal and Objectives

Primary research goal: To design and evaluate a hybrid machine-learning phishing website detection approach that improves detection of previously unseen phishing websites while maintaining low false-positive rates and practical computational efficiency.

1. To identify the URL, page, and contextual feature categories used by the six reviewed phishing-detection studies and determine the principal limitations associated with each category.
2. To construct a phishing website dataset containing legitimate and phishing samples and organize it using a time-aware training and testing strategy so that previously unseen samples can be evaluated.
3. To design a hybrid phishing detection model that combines a lightweight URL-based learning component with selected complementary features for uncertain cases.
4. To evaluate the proposed model using measurable indicators including accuracy, precision, recall, F1-score, false-positive rate, AUC, and average detection time.
5. To compare the proposed approach with representative baseline methods, including a conventional feature-based classifier and a URL-based deep-learning model, under the same dataset split and evaluation metrics.
6. To assess the model's generalizability by measuring its performance on phishing samples collected from a later time period than the training data.

4. Literature Review

The literature reviewed in Assignment 1 demonstrates a progression from manually engineered phishing indicators toward representation learning and hybrid detection. The six selected studies are relevant because they address different dimensions of the same research problem: proactive identification of malicious or phishing web resources.

4.1 PhishStorm: Detecting Phishing With Streaming Analytics

Marchal et al. (2014) propose PhishStorm as an automated system that analyzes URLs in real time. The study is motivated by the weakness of reactive blacklisting when phishing websites have short lifetimes. The authors introduce intra-URL relatedness, which measures relationships between words in different URL components, and derive features using Google and Yahoo query data. These features are used with machine-learning classifiers. The evaluation includes 96,018 legitimate and phishing URLs and reports 94.91% correct classification with 1.44% false positives. The approach is valuable because it addresses real-time detection and uses a substantial dataset. However, its dependence on external search-engine information creates possible availability, latency, and reproducibility concerns. This limitation motivates approaches that can operate directly on URL content.

4.2 URLNet: Learning a URL Representation with Deep Learning for Malicious URL Detection

Le et al. (2018) propose URLNet to learn representations directly from URL strings. The framework applies convolutional neural networks to character-level and word-level representations and jointly optimizes the learned embedding and classifier. The method addresses three weaknesses identified in conventional lexical models: limited ability to capture semantic and sequential patterns, substantial manual feature engineering, and difficulty handling unseen features. Large-scale experiments show performance gains over traditional lexical machine-learning baselines. URLNet is important to this research because it demonstrates the potential of end-to-end URL learning. Nevertheless, it targets malicious URLs broadly rather than phishing alone, and the deep model is less interpretable than many conventional feature-based classifiers.

4.3 Phishing Page Detection via Learning Classifiers from Page Layout Feature

Mao et al. (2019) investigate phishing detection from visual page-layout similarity. CSS properties are extracted from webpages, converted into comparison vectors, and evaluated using SVM, Decision Tree, AdaBoost, and Random Forest classifiers. The study uses more than 490 phishing pages imitating 46 target pages and produces more than 20,000 testing samples. Its strength is that it uses a page-level feature family that is independent of URL spelling and therefore can complement URL-based detection. The principal deployment limitation is that the page normally has to be retrieved and processed before a decision can be made. Dynamic pages, legitimate redesigns, or phishing pages that do not closely imitate a known target may also reduce effectiveness.

4.4 Phishing Website Detection Based on Multidimensional Features Driven by Deep Learning

Yang, Zhao, and Zeng (2019) propose a deep-learning approach based on multidimensional phishing features. The central idea is that combining heterogeneous evidence can provide more robust discrimination than relying on one narrow feature source. This direction is useful because sophisticated phishing may evade simple lexical rules. At the same time, multidimensional feature collection increases system complexity and can make real-time deployment more expensive. Deep-learning decisions can also be difficult to explain, which matters when security teams need to understand why a website was blocked.

4.5 PDRCNN: Precise Phishing Detection with Recurrent Convolutional Neural Networks

Wang et al. (2019) propose PDRCNN, a URL-only architecture combining recurrent and convolutional neural networks. The recurrent component captures global sequential information, while the convolutional component extracts local patterns. The method does not depend on search engines or DNS services. The authors construct a large dataset from Alexa and PhishTank with nearly 500,000 samples and report approximately 97% accuracy, an AUC of about 99%, and detection time below 0.4 milliseconds per URL. This makes the approach attractive for high-volume screening. Its main limitation is that URL-only evidence can be weak when phishing is hosted on compromised legitimate domains or when attackers deliberately construct benign-looking URLs.

4.6 Phishing Website Detection: Improved Accuracy through Feature Selection and Ensemble Learning

Ubing et al. (2019) examine feature selection and ensemble learning for phishing website classification. The study shows that carefully selecting informative indicators and using conventional classifiers can produce competitive performance without a deep neural architecture, reporting accuracy of up to approximately 95%. This approach is practically attractive because conventional models are generally easier to train and explain. However, a selected feature subset can be highly dependent on the dataset from which it was derived, and accuracy alone does not fully describe performance in an operational environment.

5. Critical Analysis

The six studies provide strong evidence that proactive learning-based detection can improve on blacklist-only defenses, but they use different assumptions. URL-only systems are fast and suitable for early-stage filtering, while page-layout and multidimensional methods can provide stronger context at a higher collection and processing cost. Deep-learning models reduce manual feature engineering but require more computational resources and generally provide less transparent decisions. Conventional feature-based methods remain relevant because they can be efficient and interpretable.

A recurring methodological concern is dataset representativeness. Public phishing feeds contain reported attacks, while legitimate samples are frequently obtained from popular-site lists. This can create a cleaner separation than ordinary production traffic. Another concern is temporal leakage: if related URLs from the same campaign occur in both training and testing data, measured performance can overestimate the ability to detect genuinely new campaigns. For this reason, the proposed research objectives explicitly include time-aware evaluation and generalization testing.

6. Connections and Conflicts Between the Findings

Connections. All six papers support automated learning as a means of detecting threats that static blacklists may miss. They also agree, either directly or implicitly, that the quality of the representation strongly influences classification performance. PhishStorm engineers semantic URL features; URLNet and PDRCNN learn URL representations; Mao et al. learn from layout comparison; Yang et al. combine multiple feature dimensions; and Ubing et al. select a compact informative feature set.

Conflicts and trade-offs. The papers differ on how much information should be collected before classification. PDRCNN favors URL-only speed, while page-layout and multidimensional approaches favor richer evidence. PhishStorm depends on external search information, whereas PDRCNN explicitly avoids such dependencies. Deep-learning studies favor learned representations, while feature-selection research shows that conventional models can remain competitive. These are not simple contradictions; they represent different trade-offs among speed, complexity, interpretability, and detection capability.

7. Synthesis Matrix

Study	Year	Methodology	Key findings	Main limitation
PhishStorm	2014	Intra-URL relatedness + ML; search-engine-derived features; streaming analytics.	96,018 URLs; 94.91% correct classification; 1.44% false positives.	External search dependency; performance may change with evolving URL patterns.
URLNet	2018	Character- and word-level CNN representation learning directly from URLs.	Large-scale experiments outperform conventional lexical baselines.	Malicious-URL scope broader than phishing; interpretability and training cost.
Mao et al.	2019	CSS layout comparison + SVM, Decision Tree, AdaBoost and Random Forest.	>490 phishing pages, 46 targets, >20,000 test samples; layout similarity is useful.	Requires page retrieval; dynamic or non-imitative phishing can weaken the signal.
Yang et al.	2019	Deep learning with multidimensional phishing features.	Combining heterogeneous evidence improves the detection representation.	Higher collection/processing complexity; explainability concerns.
PDRCNN	2019	Hybrid RNN + CNN using URL strings only.	Nearly 500,000 samples; ~97% accuracy; ~99% AUC; <0.4 ms per URL.	URL-only blind spots; possible sampling bias from Alexa/PhishTank.
Ubing et al.	2019	Feature selection and ensemble/conventional ML.	Up to ~95% accuracy; competitive without deep learning.	Feature subset may be dataset-specific; accuracy alone is insufficient.

8. Synthesis and Research Gap

Taken together, the literature suggests that a layered approach can exploit the complementary strengths of existing methods. A lightweight URL model can provide fast initial screening, while selected page or contextual features can be used only when the first-stage decision is uncertain. This can potentially reduce the cost of full multidimensional analysis while retaining richer evidence for difficult cases.

The specific research gap addressed in this assignment is generalization to previously unseen phishing campaigns under a consistent evaluation design. The proposed research will therefore compare methods using the same data split and metrics and will separate training and later-period testing data. This directly connects the research objectives to the limitations identified in the literature and makes the intended outcome measurable.

References

1. Le, H., Pham, Q., Sahoo, D., & Hoi, S. C. H. (2018). URLNet: Learning a URL representation with deep learning for malicious URL detection. arXiv:1802.03162.
2. Mao, J., Bian, J., Tian, W., Zhu, S., Wei, T., Li, A., & Liang, Z. (2019). Phishing page detection via learning classifiers from page layout feature. *EURASIP Journal on Wireless Communications and Networking*, 2019, 43. <https://doi.org/10.1186/s13638-019-1361-0>
3. Marchal, S., François, J., State, R., & Engel, T. (2014). PhishStorm: Detecting phishing with streaming analytics. *IEEE Transactions on Network and Service Management*, 11(4), 458-471. <https://doi.org/10.1109/TNSM.2014.2377295>
4. Ubing, A. A., Jasmi, S. K. B., Abdullah, A., Jhanjhi, N. Z., & Supramaniam, M. (2019). Phishing website detection: An improved accuracy through feature selection and ensemble learning. *International Journal of Advanced Computer Science and Applications*, 10(1). <https://doi.org/10.14569/IJACSA.2019.0100133>
5. Wang, W., Zhang, F., Luo, X., & Zhang, S. (2019). PDRCNN: Precise phishing detection with recurrent convolutional neural networks. *Security and Communication Networks*, 2019, 2595794. <https://doi.org/10.1155/2019/2595794>
6. Yang, P., Zhao, G., & Zeng, P. (2019). Phishing website detection based on multidimensional features driven by deep learning. *IEEE Access*, 7, 15196-15209. <https://doi.org/10.1109/ACCESS.2019.2892066>