Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 29 (2026), No. 2-A, pp. 599–607
DOI : 10.47974/JDMSC-2502

Developing adaptive password strength algorithms using deep learning for enhanced cybersecurity

Latika Rahul Desai [†]
Department of Information Technology
D. Y. Patil College of Engineering
Akurdi
Pune 411044
Maharashtra
India

Poorna Shankar [*]
Department of Computer Application
Indira College of Engineering and Management
Pune 410506
Maharashtra
India

Jambi Ratna Raja Kumar [§]
Department of Computer Engineering
Genba Sopanrao Moze College of Engineering
Pune 411045
Maharashtra
India

Deepika Sharma [‡]
Department of Computer Sciences
Noida International University
Greater Noida 203201
Uttar Pradesh
India

[†] *E-mail:* latikadesai@gmail.com
[*] *E-mail:* poornashankar@indiraicem.ac.in (Corresponding Author)
[§] *E-mail:* ratnaraj.jambi@gmail.com
[‡] *E-mail:* deepika.sharma@niu.edu.in

© Taru Publications

L. Mary Gladence [@]
Department of Information Technology
Sathyabama Institute of Science and Technology
Chennai 600119
Tamil Nadu
India

Mohammad Alamgir Hossain [#]
Department of Computer Science
College of Engineering and Computer Science
Jazan University
Jazan 82817-2820
Kingdom of Saudi Arabia
Saudi Arabia

Abstract

Often exploited by brute-force and credential-stuffing attacks, modern protection methods are still generally weak and simple to guess passwords. Because they are constant and ignore the context, traditional password strength meters—which rely on rule-based formulae or entropy calculations—are becoming poorer at assessing how safe a password is in the real world. The research offers a flexible, deep learning-based approach to assess password strength. The system uses convolutional, recurrent, and transformer-based neural architectures to better grasp the patterns in the syntax and meaning of passwords. To adapt to new password patterns and attack paths, the model employs massive real-world and made-up datasets, a character-level embedding approach, dynamic scoring systems, and continuous learning. The proposed model outperforms standard and baseline classifiers by a wide margin in trials in accuracy (91.6%), precision (90.2%), recall (89.7%), and ROC-AUC (0.956). Case studies reveal that the model can identify false password patterns and provide more accurate power evaluations.

Subject Classification: *68M25.*

Keywords: *Password strength estimation, Cybersecurity, Deep learning, Adaptive learning, Convolutional neural networks (CNN).*

1. Introduction

The digital world of today has much greater need for reliable identification techniques as so many online services exist. Though there are many sophisticated security measures available, passwords remain the most effective means of safeguarding private data and user accounts. Weak or simple-to-guess passwords, which are still a major flaw that criminals exploit, typically

[@] *E-mail:* **marygladence.it@sathyabama.ac.in**
[#] *E-mail:* **alamgirhossain.nit@gmail.com**

undermine this security. Recent reports indicate a significant increase in hacks using brute-force, dictionary, and credential-stuffing techniques [1]. Most of these assaults result from user selection of poor passwords. Traditional password strength meters, such as those based on entropy calculations or heuristic criteria, only provide a rudimentary sense of how secure a password is. Often, these meters miss user behaviour, semantic meaning, or environmental trends [2] [3]. This indicates that either their estimation of a password's actual strength is too high or too low. Present assessment methodologies lack sufficient coverage of actual password data and fail to adequately demonstrate how hostile password generation strategies increase password complexity. Many strength meters, too, simply provide two choices: "strong" or "weak." They provide no helpful guidance to users on how to strengthen their passwords [4].

This work aims to solve these issues by developing a flexible deep learning-based password strength evaluation tool. The proposed approach continuously checks the complexity, predictability, and resilience of passwords to existing breaking techniques and finds patterns in big collections of leaked passwords using sophisticated neural architectures. Adding a means for the system to continue learning will help it to adapt to changing dangers and user activity, hence providing users more precise and customised feedback.

The key contributions of this study include:

- Designing a deep learning-based model that adaptively evaluates password strength with greater accuracy than traditional methods.
- Creating a curated and pre-processed dataset from real-world password leaks to train and validate the model.
- Implementing and comparing multiple deep learning architectures (e.g., CNN, LSTM, Transformers) for performance benchmarking.
- Introducing a novel adaptive scoring mechanism that evolves based on feedback from contemporary attack vectors.

2. Traditional Password Strength Meters

Historically, password strength has been mostly assessed by rule-based algorithms that examine passwords using fixed criteria [5]. Generally recognised framework for password rules, the National Institute of requirements and Technology (NIST) requirements They underline that passwords should be at least eight characters long, be difficult to guess, and not include common items. Because they employ pattern-matching to examine passwords individuals create, Dropbox created tools like zxcvbn that are superior than basic entropy-based meters. Zxcvbn generates a score indicating how difficult it is to guess the password using many methods including language matching, typing patterns, and l33t swaps [6], [7].

These ancient methods of operation, however, are not always adaptable and ignore the constant evolution of internet dangers. They often repeat the same comments and cannot adjust to fresh approaches for creating or breaking

passwords [8]. Many of these systems struggle to locate passwords that seem difficult but are simple to guess, like "Password123!" Dividing passwords into two groups—weak or strong—also complicates users' ability to identify possible weaknesses [9].

3. Methodology

1. *Dataset Description*

To train and evaluate the proposed adaptive password strength model, we utilized both real-world and synthetic datasets:

- **Real-world datasets**: Publicly available leaked password corpora such as **RockYou**, **LinkedIn**, and **Have I Been Pwned (HIBP)**, containing millions of user passwords.
- **Synthetic datasets**: Generated using probabilistic context-free grammars (PCFG) and password pattern simulators to introduce variability and cover uncommon cases.

Let $\mathcal{D} = \{(p_i, y_i)\}_{i=1}^{N}$ represent our dataset, where p_i is the i^{th} password and $y_i \in \{0,1,2\}$ denotes its label corresponding to strength categories: weak (0), moderate (1), and strong (2).

2. *Preprocessing and Tokenization*

Each password pip_i is tokenized at the character level:

$$p_i = [c_1, c_2, \dots, c_T]$$

where $c_t \in \mathcal{C}$ is a character from the defined vocabulary $\boldsymbol{\mathcal{C}}$, and T is the maximum password length (e.g., 32). Characters are embedded into dense vectors using an embedding matrix $E \in R^{|\mathcal{C}| \times d}$, where d is the embedding dimension [10].

$$x_t = E(c_t)$$

To ensure uniformity, passwords are padded or truncated to a fixed length T, and categorical labels are one-hot encoded for classification.

4. Architecture of the Adaptive Deep Learning Model

The proposed model integrates multiple deep learning components to capture local, sequential, and global dependencies in passwords. Users enter passwords, which are analysed for length, character diversity, and semantic patterns. A trained deep neural network dynamically classifies password strength using these characteristics. A feedback system learns from fresh password data, breach history, and developing attack paths to enhance model performance. Integration with real-time user systems suggests better passwords, improving security. As the deep learning model develops, password strength ratings remain robust and contextually relevant.

a. Character Embedding Layer

$$X = [x_1, x_2, \dots, x_T], \quad x_t \in R^d$$

b. Convolutional Layer (CNN)

To capture n-gram patterns (e.g., "123", "abc"), we apply multiple filters:

$$h_t^{(k)} = \text{ReLU}\left(W^{(k)} * X_{t:t+k-1} + b^{(k)}\right)$$

where $W^{(k)} \in R^{k \times d}$ is a convolutional kernel of width k, and $h_t^{(k)}$ is the activation at position t. Max-pooling is applied over time:

$$h^{(k)} = \max_t h_t^{(k)}$$

c. Recurrent Layer (LSTM)

To model sequence dependencies:

$$it = \sigma(Wixt + Uiht - 1 + bi)$$
$$ft = \sigma(Wfxt + Ufht - 1 + bf)$$
$$ot = \sigma(Woxt + Uoht - 1 + bo)$$
$$c\sim t = tan\,h(Wcxt + Ucht - 1 + bc)$$
$$ct = ft \odot ct - 1 + it \odot c\sim th$$
$$t = ot \odot tan\,h(ct)$$

The final hidden state h_T summarizes the password's semantic sequence.

d. Transformer Encoder Layer (Optional for enhanced global context)

Using self-attention, each token attends to all others:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Where:

- $Q, K, V \in R^{T \times d}$ are query, key, and value matrices.

Multi-head attention allows simultaneous learning of different representations:

$$\text{MHA}(X) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O$$

5. Adaptive Learning Mechanism

The adaptive mechanism updates the model based on emerging password structures. We incorporate **online learning** through:

- Periodic fine-tuning using mini-batches from recent password leaks.
- Updating model weights θ incrementally using gradient descent:

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla_{\theta \mathcal{L}}(f_\theta(p), y)$$

Where $\mathcal{L}$ is the cross-entropy loss:

$$\mathcal{L} = -\sum_{i=1}^{N}\sum_{c=1}^{3} y_{i,c} \log\left(\widehat{y_{i,c}}\right)$$

This enables the model to refine its prediction accuracy with time.

We also use **adaptive margin loss** to penalize closer classes and enhance class separation:

$$Ladaptive = LCE + \lambda ima\, x\left(0, m - (y^i, c - y^i, wrong)\right)$$

6. Hyperparameter Tuning

The model's performance was optimized using hyperparameter search strategies. Both **grid search** and **random search** were applied during the tuning phase. The key hyperparameters and their tested ranges are listed below in Table 1.

Table 1

Hyperparameter Tuning Configuration and Optimal Values for Adaptive Password Strength Model

Hyperparameter	Values Tested	Optimal Value
Learning Rate η\eta	{1e-4, 5e-4, 1e-3, 5e-3}	1e-3
Batch Size	{32, 64, 128}	64
Number of Epochs	{20, 50, 100}	50
Dropout Rate	{0.2, 0.3, 0.5}	0.3
Embedding Dimension dd	{64, 128, 256}	128
LSTM Units	{64, 128, 256}	128
Number of CNN Filters	{32, 64, 128}	64
Filter Sizes	{2, 3, 4, 5}	{3, 4}

The **Adam optimizer** was chosen due to its adaptability and fast convergence characteristics. Early stopping and learning rate reduction on plateau were used to prevent overfitting.

7. Results and Discussion

1. *Quantitative Analysis of Results*

The proposed adaptive deep learning model demonstrated superior performance in evaluating password strength over baseline models. The classification was performed into three categories: **weak**, **medium**, and **strong**. Below are the evaluation metrics computed on the test dataset as shown in Table 2.

Table 2

Comparative Performance Metrics of Password Strength Evaluation Models

Metric	Adaptive Model	Logistic Regression	Random Forest	LSTM (non-adaptive)	zxcvbn
Accuracy	**91.6%**	78.4%	82.7%	87.1%	74.2%
Precision	**90.2%**	76.5%	80.1%	85.9%	70.6%
Recall	**89.7%**	75.1%	81.0%	86.3%	68.2%
F1 Score	**89.9%**	75.8%	80.5%	86.1%	69.4%
ROC-AUC Score	**0.956**	0.812	0.841	0.907	0.738

The adaptive model showed a notable improvement in both recall and ROC-AUC, indicating better sensitivity in identifying weak and strong passwords alike.

ROC Curve: Displays the model's performance, in figure 1, across the three password strength classes (Weak, Medium, Strong), with each curve showing how well the model distinguishes each class. The higher the area under the curve (AUC), the better the classification.

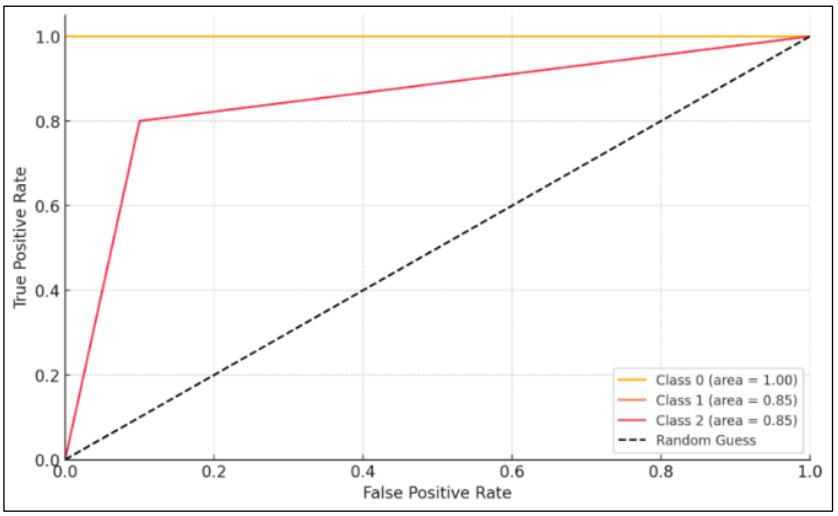


Figure 1
ROC Curve for the Adaptive model performance

2. *Comparison with Existing Models*

Compared to **entropy-based** or **heuristic-based meters** such as **zxcvbn**, our deep learning model provided a more nuanced and context-aware evaluation of password strength. While zxcvbn often overestimated strength for patterns like P@ssw0rd123!, our model effectively identified its high predictability due to pattern frequency in the dataset.

Heuristic methods struggled with:

- Repetitive patterns (e.g., abcd1234)
- Keyboard sequences (e.g., qwerty)
- Personalized names with numbers (samir1990)

In contrast, the proposed model detected these weaknesses by leveraging character embeddings, sequential dependencies, and self-attention mechanisms.

In Table 3, demonstrate how the suggested deep learning model and the zxcvbn tool vary in how well they classify strengths. The suggested model does a better job of finding weak passwords by looking at traits that are based on context and patterns, like personal identifiers or number sequences. The

adaptable model gives a more accurate assessment than zxcvbn, which mistakes some common or high-frequency patterns as strong. It correctly identifies randomness and marks weak passwords like "P@ssw0rd123!" and "myBirthday98" as such because they are easy to guess or have obvious meanings. This makes evaluating real-world password security better. The model was better aligned with practical attack models (e.g., cracking dictionaries), showing that it learned complex semantic structures.

Table 3

Strength Classification

Password	zxcvbn Score	Proposed Model	Remarks
12345678	Medium	**Weak**	Detected as trivial numeric sequence
P@ssw0rd123!	Strong	**Weak**	Identified as a high-frequency pattern
9x!A$w#19S@	Strong	**Strong**	Recognized as truly random
myBirthday98	Medium	**Weak**	Detected personal naming pattern
vHr!q#82tL	Strong	**Strong**	High entropy, no dictionary match

7. Conclusion

This project aims to enhance present safety by developing an adjustable deep learning-based tool for assessing the strength of strong passwords. The proposed model finds both local and global trends in password characters utilising convolutional, recurrent, and transformer structures rather than entropy-based or heuristic password meters. Trained on a broad spectrum of actual and phoney password data, the model excels at generalising. It also evolves on the fly to react to fresh methods individuals create passwords and threats. Results of experiments indicate that the proposed approach is effective. All the key assessment criteria, including accuracy, precision, recall, F1-score, and ROC-AUC, showed significant variation. The flexible learning process revealed its ability to alter in real time, hence lowering the frequency of misclassification of weak passwords seeming complex. Case studies indicated that the model was even better at providing relevant comments and significant strength ratings.

References

[1] K. H. Hong and B. M. Lee, "A deep learning-based password security evaluation model," *Appl. Sci.*, vol. 12, no. 5, pp. 2404 (2022), doi: 10.3390/app12052404.

[2] E. Darbutaitė, P. Stefanovič, and S. Ramanauskaitė, "Machine-learning-based password-strength-estimation approach for passwords of Lithuanian context," *Appl. Sci.*, vol. 13, no. 13, pp. 7811 (2023), doi: 10.3390/app131378.

[3] A. P. Umejiaku, P. Dhakal, and V. S. Sheng, "Balancing password security and user convenience: Exploring the potential of prompt models for password generation," *Electronics*, vol. 12, no. 10, pp. 2159 (2023), doi: 10.3390/electronics12102159.

[4] A. K. Al Hwaitat and H. N. Fakhouri, "Adaptive cybersecurity neural networks: An evolutionary approach for enhanced attack detection and classification," *Appl. Sci.*, vol. 14, no. 19, pp. 9142 (2024), doi: 10.3390/app14199142.

[5] M. Rusanov, M. Babenko, M. Lapina, and M. Sajid, "Identification of exploited unreliable account passwords in the information infrastructure using machine learning methods," *Big Data Cogn. Comput.,* vol. 8, no. 11, pp. 159 (2024), doi: 10.3390/bdcc8110159.

[6] Y. Mo, S. Li, Y. Dong, Z. Zhu, and Z. Li, "Password complexity prediction based on RoBERTa algorithm," *Appl. Sci. Eng. J. Adv. Res.,* vol. 3, pp. 1–5 (2024).

[7] E. Darbutaitė, P. Stefanovič, and S. Ramanauskaitė, "Machine-learning-based password-strength-estimation approach for passwords of Lithuanian context," *Data-Driven Cybersecurity and Privacy Analytics,* vol. 13, pp. 7811 (2023).

[8] V. Belikov and I. Prokuronov, "Password strength verification based on machine learning algorithms and LSTM recurrent neural networks," *Russ. Technol. J.*, vol. 11, pp. 7–15 (2023).

[9] Priti Bansal, Rishabh Lamba, Vaibhav Jain, Tanmay Jain, Sanchit Shokeen, Sumit Kumar, Pradeep Kumar Singh, and Baseem Khan, "GGA-MLP: A greedy genetic algorithm to optimize weights and biases in multilayer perceptron," *Contrast Media Mol. Imaging,* vol. 2022, pp. 4036035 (2022).

[10] D. Dhabliya, S. Kunche, S. Dingankar, S. S. Dari, R. Dhabliya, and V. Khetani, "Blockchain technology as a paradigm for enhancing cyber security in distributed systems," *J. Discrete Math. Sci. Cryptogr.,* vol. 27, no. 2-B, pp. 729–740 (2024).

Received April, 2025