Postprint of a Password Attack Algorithm Based on Conditional Variational Autoencoder

Authors: Duan Dagao, Zhao Zhendong, Liang Shaohu, Han Zhongming

Date: 2019-01-03T00:00:00+00:00

Abstract

Passwords are a widespread approach for data encryption and user authentication. Since user-chosen passwords lack complete randomness, they are highly vulnerable to attacks from password cracking tools. The use of password guessing algorithms constitutes an effective method for evaluating password strength and security. This paper proposes PassCVAE, a password guessing algorithm based on Conditional Variational Autoencoder. The algorithm leverages the Conditional Variational Autoencoder model, employing user personal information as conditional features to train a password attack model. At the encoder side, a bidirectional GRU recurrent neural network and a TextCNN text convolutional neural network are respectively utilized to encode password sequences and user personal information while performing abstract feature extraction; at the decoder side, a two-layer GRU neural network decodes user personal information and latent encodings of password data to generate password sequences. The algorithm can effectively model the distribution of password data and character combination patterns, producing high-quality guessed password data. Multiple sets of experimental results demonstrate that the proposed PassCVAE algorithm outperforms existing mainstream password guessing algorithms.

Full Text

Password Cracking Algorithm Using Conditional Variational Autoencoders

Duan Dagao[1][2], Zhao Zhendong[1], Liang Shaohu[1], Han Zhongming[1][2]
[1]School of Computer & Information Engineering, Beijing Technology & Business University [2]Beijing Key Laboratory of Big Data Technology for Food Safety, Beijing 100048, China

Abstract: Passwords are universal methods for data encryption and user authentication. The passwords set by users are not completely random, making

them vulnerable to password cracking tools. Using password guessing algorithms is an effective way to assess password strength and security. This paper proposes PassCVAE, a password guessing algorithm based on conditional variational autoencoders. The algorithm employs user personal information as conditional features to train a password attack model. At the encoder side, bidirectional GRU recurrent neural networks and TextCNN convolutional neural networks are used to encode password sequences and extract abstract features from personal information. At the decoder side, a two-layer GRU neural network decodes the latent encoding of password data combined with user personal information to generate password sequences. This algorithm can effectively fit the distribution of password data and learn character combination rules, generating high-quality password guessing data. Extensive experimental results demonstrate that the proposed PassCVAE algorithm outperforms existing mainstream password guessing algorithms.

Keywords: conditional variational autoencoders; password guessing algorithm; password cracking

0 Introduction

With the rapid development and widespread adoption of mobile internet technology, an increasing number of users are learning, working, and entertaining themselves through mobile terminals. Effective and secure user authentication is crucial for network information security and the protection of user privacy data. Although new technologies such as fingerprint recognition and facial recognition have been proposed, user passwords remain the most common authentication method due to their simple implementation, good user experience, and low software development cost. Unfortunately, multiple password database leaks have shown that users tend to choose easily guessable passwords composed primarily of common strings and numbers, with many password creation rules incorporating various combinations of personal information. Consequently, these passwords are vulnerable to password cracking algorithms.

Therefore, verifying whether user passwords are securely configured represents a critical security issue. Active online password guessing detection techniques are commonly used to evaluate password strength. Many researchers have proposed online password guessing algorithms based on probabilistic statistical models to verify password security. Reference [1] systematically evaluated numerous probabilistic password models, including Markov models with various normalization and smoothing techniques. Reference [2] proposed a new Markov model-based password cracker that significantly improved the guessing speed of existing algorithms. Reference [3] introduced a method for generating password structures in descending probability order, which first trains an automatic context-free grammar creation based on existing public datasets, then generates word modification rules according to the learned grammar for producing guessed passwords. However, these traditional statistical methods cannot accurately learn users' password setting habits and require substantial computational resources

and time, making them unsuitable for real-time password strength evaluation. Moreover, most existing password security detection algorithms only consider the probability distribution of characters in password datasets without incorporating user personal information (such as email addresses, usernames, etc.) as conditional features, even though such personal information often exhibits strong correlations with passwords.

In recent years, deep learning [4,5] has achieved remarkable success in artificial intelligence. Deep learning can extract abstract features and possesses powerful fitting capabilities for high-dimensional data, proving highly effective for sequence generation tasks [6,7]. This paper leverages conditional variational autoencoders (CVAE) from deep learning, using user personal information as conditional features for password generation tasks, and proposes the PassCVAE password attack algorithm. Through comparative experiments with multiple existing models on large-scale datasets, the results demonstrate that the proposed PassCVAE algorithm outperforms traditional password guessing models, better fitting data distributions and generating higher-quality guessed passwords.

1 Related Work

The simplest password attack method is brute-force cracking, which performs an exhaustive search of all possible combinations. Due to its prohibitively high time cost, this approach is generally infeasible. As an improvement over brute-force attacks, dictionary attacks try possible passwords (words or phrases) from a user-defined dictionary one by one. Unlike brute-force cracking, which attempts all possible password combinations, dictionary attacks use a predefined word list of likely passwords. While this simple method may sometimes work, it misses numerous password combinations not based on existing dictionaries and cannot accurately exploit users' password setting habits.

To address this limitation, many researchers have introduced machine learning into password guessing attack models to better identify high-probability password character combinations and learn reasonable distributions of password text data. Reference [8] proposed the TarGuess password attack framework, which established grammar-adaptive rules for user personal information using context-free grammars and achieved certain research results through Bayesian optimization. Reference [9] employed multi-layer LSTM recurrent neural networks [10,11] to implement a probabilistic language model for generating guessed passwords, also achieving good attack effectiveness. Reference [12] introduced Markov models into dictionary attacks, significantly reducing the password search space and proposing efficient algorithms for enumerating the remaining password space. Reference [13] utilized syntactic and semantic tags to build context-free models that capture the semantic essence of password samples. Reference [14] added pinyin rules to context-free grammar models for Chinese passwords, improving algorithm effectiveness. Reference [15] proposed new evaluation metrics to replace Shannon entropy and guessing entropy through statistical analysis of 70 million anonymized Yahoo passwords. Reference [16] proposed

adaptive password strength evaluation rules using Markov models, greatly improving the accuracy of password strength estimation. Reference [17] introduced a new method based on Monte Carlo methods to estimate the number of guesses required by modern attack methods, offering advantages such as low resource consumption and easy convergence. Reference [18] measured password strength based on Brute-Force Markov (BFM), a hybrid between brute-force cracking and n-gram models that can more accurately calculate the required number of guesses. Reference [19] proposed PassGAN, a novel method for enhancing password generation based on adversarial neural networks. By training on existing leaked password data, PassGAN can approximate the distribution of the password training dataset, thus potentially matching passwords that have not yet been leaked.

2.1 Conditional Variational Autoencoders

Variational Autoencoders (VAE) are generative models based on a regularized version of standard autoencoders. The model imposes a prior distribution on the latent variable z, where $p(z)$ is a regular geometric form (commonly a standard Gaussian distribution), enabling the model to generate samples closer to the original data distribution. VAE replaces the encoder in standard autoencoders with a learned posterior recognition model $q(z|x)$, parameterizing the posterior distribution of latent variables to approximate the imposed prior distribution (standard Gaussian distribution). VAE has two learning objectives: (a) minimizing the reconstruction loss of samples, and (b) minimizing the KL divergence between the encoded latent variable z and the standard Gaussian distribution.

The model' s loss function is shown in Equation (1), where $KL(q(z|x)||p(z))$ represents the KL divergence between $q(z|x)$ and $p(z)$, measuring the similarity between two distributions. When the two distributions are more similar, the KL divergence is smaller. $p(x|z)$ represents the reconstruction loss of data samples by the decoder, whose learning objective is to reconstruct the real data as accurately as possible.

Conditional Variational Autoencoders (CVAE) [20,21] are conditional probability extensions of VAE. While VAE cannot control the data generation process, CVAE enables generation of data under specific conditions by adding generation conditions to the model. The loss function is shown in Equation (2), where y is the conditional variable, and the decoder generates specific data under condition y. In this paper' s password guessing model, the generation condition y is user personal information, including username, email address, and phone number.

2.2 Password Attack Model

The overall framework of the model is shown in Figure 1 [Figure 1: see original paper]. The encoder consists of two parts: a two-layer bidirectional GRU and a CNN convolutional network. The GRU encoder encodes user password sequences, while the CNN encoder encodes user personal information.

As shown in Equation (3), $x_{1:t}$ is the user' s password sequence, processed by a bidirectional GRU recurrent neural network. Taking its final time-step output state h_t, two fully connected layers generate μ and σ (Equations (4) and (5)). In Equation (6), randn represents a random vector sampled from the standard normal distribution with the same dimension as μ, which after reparameterization yields the intermediate encoding z'. In Equation (7), y is user personal context data, $y = \{\text{username}, \text{email}, \text{phone number}\}$, which concatenates user personal information as a string and encodes it through a CNN convolutional network to generate the conditional encoding vector g. Equation (8) concatenates the intermediate encoding z' with the conditional encoding g to form the final latent encoding z.

The decoder is implemented by a two-layer unidirectional GRU. As shown in Equation (9), the hidden state at each time step incorporates the latent encoding z and the conditional encoding vector g, generating the password guessing sequence x'.

During training, the model controls the latent encoding z generated by the encoder through the standard Gaussian prior distribution using KL divergence, making it approach the prior Gaussian distribution. The encoder CNN network takes user personal information (email address, username, and phone number) as input to generate the user' s conditional encoding vector g. Finally, the latent variable z and conditional encoding vector g are concatenated together as the initial state of the decoder to generate password sequences.

After model training, latent variables z are randomly sampled from the standard Gaussian distribution, and user personal information is encoded by the decoder CNN network to generate the conditional encoding vector g. By inputting the latent variable z and conditional encoding vector g into the decoder, the guessed password sequence for this user can be generated.

Under the control of the prior distribution, password sequences from the dataset are abstractly encoded and embedded in a high-dimensional Gaussian distribution space. When generating passwords, latent variables sampled from the prior distribution $p(z)$ will conform to the real encoding distribution of the training data, and combined with the conditional encoding vector g, this enables generation of user password guessing sequences.

2.3 Algorithm Implementation Steps

Based on the conditional variational autoencoder and password attack model described above, the implementation process of the PassCVAE algorithm can be organized as follows.

Algorithm: Password Attack Algorithm Based on Conditional Variational Autoencoder

1. Initialize parameters of the GRU and CNN encoder modules, and initialize the GRU decoder module parameters.

2. For each iteration $i = 1, 2, \ldots, M$ do:
3. Sample a password sample sequence $x_{1:t}$ with user context information y.
4. Generate the password sequence hidden state sequence h_t according to Equation (3).
5. Generate variational parameters μ and σ based on h_t according to Equations (4) and (5).
6. Obtain the password sequence encoding latent variable z' according to Equation (6).
7. Obtain the user context feature variable g according to Equation (7).
8. Generate the final latent variable z according to Equation (8).
9. Obtain the generated password sequence x' according to Equation (9).
10. Compute the reconstruction loss and KL divergence loss.
11. Update parameters of GRU, CNN, and GRU modules using gradient descent.
12. End for.

3.1 Dataset

The experimental evaluation dataset in this paper consists of three large-scale real-world password datasets, primarily leaked through hacker attacks or insiders and publicly available on the Internet. The detailed dataset descriptions are shown in Table 1 .

The 12306 dataset contains leaked password data from a Chinese internet train ticket booking platform, including relatively complete user personal information such as user email, phone number, ID number, and name (pinyin). The CSDN dataset contains leaked user password data from an IT community platform, including password, username, and email information. The Renren dataset contains leaked user password data from a Chinese social networking platform, including password and email information.

3.2 Experimental Setup

To verify the effectiveness of the proposed method, this paper selects four password guessing algorithms for comparison: PCFG [3], OMEN [2], PassGAN [17], and PassLSTM. PCFG and OMEN are based on traditional statistical methods, PassGAN employs generative adversarial networks from deep learning, and PassLSTM is based on LSTM recurrent neural network language models.

The experimental models are trained on the training samples from each dataset to produce password generation models. On the test sets, the number of cracking attempts per password is limited to no more than a specified threshold—successful cracking within 1000, 2000, 3000, 4000, and 5000 attempts is considered a successful attack. The calculation process is shown in Equations (10)-(12). In Equation (10), X' represents the n generated guessing sequences. The success rates of each model under different guessing counts are statistically analyzed on the test samples of different datasets. The experimental results are

shown in Tables 2 –4.

In the 12306 dataset, users have more complete personal information, allowing the proposed PassCVAE model to extract more conditional information and demonstrating significantly better performance than other models. However, since the 12306 dataset contains only over 100,000 samples, the cracking success rates of all models are not as high as those on the CSDN and Renren datasets. The PassCVAE model proposed in this paper achieves the best results on all three datasets, proving the effectiveness of conditional embedding of user personal information for password cracking generation.

Tables 5 –9 show experimental results on different datasets under the same cracking attempt limits. It can be observed that the proposed PassCVAE achieves better results under different cracking attempt conditions. As the number of cracking attempts increases, reducing the randomness of generated cracked passwords, PassCVAE demonstrates even greater advantages over other comparison algorithms, achieving higher cracking success rates.

4 Conclusion

Users often tend to include personal information when setting passwords, making such passwords more susceptible to password attack algorithms. This paper proposes a password attack model based on conditional variational autoencoders, using user personal information (email address, username, phone number, etc.) as conditional features for training. At the encoder side, bidirectional GRU recurrent neural networks and CNN text convolutional neural networks are used to encode password sequences and extract abstract features from user personal information. At the decoder side, a two-layer GRU neural network decodes user personal information and latent encoding of password data to generate password sequences. The model can effectively fit the distribution of password sequences under the condition of user personal information. Experiments demonstrate that the proposed PassCVAE model outperforms existing mainstream password attack algorithms.

References

[1] Ma J, Yang Weining, Min Luo, et al. A study of probabilistic password models [C]//Proc of IEEE Symposium on Security and Privacy. San Jose: IEEE Press, 2014: 689-704.

[2] Dürmuth M, Angelstorf F, Castelluccia C, et al. OMEN: faster password guessing using an ordered Markov enumerator[C]//Proc of International Symposium on Engineering Secure Software and Systems. Berlin: Springer International Publishing, 2015: 119-132.

[3] Weir M, Aggarwal S, Medeiros B D, et al. Password cracking using probabilistic context-free grammars [C]//Proc of IEEE Symposium on Security & Privacy. Washington DC: IEEE Computer Society, 2009: 391-405.

[4] LeCun Y, Bengio Y, Hinton G. Deep learning [J]//Nature, 2015,521(7553): 436-436.

[5] Van Hasselt H, Guez A, Silver D. Deep reinforcement learning with double Q-learning [EB/OL].(2015-12-10).https://arxiv.org/pdf/1509.06461.pdf.

[6] Yu Lantao, Zhang Weinan, Wang Jun, et al. SeqGAN: sequence generative adversarial nets with policy gradient [EB/OL].(2017-08-28). https://arxiv.org/pdf/1609.05473.pdf.

[7] Lin K, Li Dianqi, He Xiaodong, et al. Adversarial ranking for language generation [C]//Advances in Neural Information Processing Systems. Cambridge: MIT Press, 2017: 3155-3165.

[8] Ding Wang, Zhang Zijian, Wang Ping, et al. Targeted online password guessing: An underestimated threat [C]//Proc of ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2016:1242-1254.

[9] Bauer L, Melicher W, Ur B, et al. Fast, lean, and accurate: Modeling password guessability using neural networks [C]//Proc of the 25th USENIX Security Symposium. Berkeley: USENIX Press, 2016:175-191.

[10] Hochreiter S, Schmidhuber J. Long Short-Term Memory [J]. Neural Computation, 1997, 9(8): 1935-1780.

[11] Karim F, Majumdar S, Darabi H, et al. LSTM fully convolutional networks for time series classification [J]. IEEE Access, 2018, 6(99): 1662-1669.

[12] Narayanan A, Shmatikov V. Fast dictionary attacks on passwords using time-space tradeoff[C]//Proc of ACM Conference on Computer and Communications Security. New York: ACM Press, 2005:364-372.

[13] Veras R, Collins C, Thorpe J. On the Semantic Patterns of Passwords and their Security Impact [C]//Proc of USENIX Networked and Distributed System Security Symposium. Berkeley, CA: USENIX Press, 2014:286-301.

[14] Li Zhigong, Han Weili, Xu Wenyuan. A large-scale empirical analysis of Chinese web passwords [C]//Proc of USENIX Conference on Security Symposium. Berkeley, CA: USENIX Press, 2014:559-574.

[15] Bonneau J. The science of guessing: analyzing an anonymized corpus of 70 million passwords [C]//Security and Privacy. San Francisco: IEEE Press: 2012, 538-552.

[16] Castelluccia C, Dürmuth M, Perito D. Adaptive password-strength meters from markov models [C]//Proc of Usenix Networked and Distributed System Security Symposium. Berkeley, CA: USENIX Press: 2012, 143-156.

[17] Dell'Amico M, Filippone M. Monte Carlo Strength Evaluation: Fast and Reliable Password Checking [C]//Proc of ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2015: 183-195.

[18] Ur B, Kelley P G, Komanduri S, et al. How does your password measure up? the effect of strength meters on password creation [C]//Proc of USENIX Security Symposium. Berkeley, CA: USENIX Press: 2012, 5-5.

[19] Hitaj B, Gasti P, Ateniese G, et al. PassGAN: a deep learning approach for password guessing [J]. arXiv preprint arXiv: 1709. 00440, 2017.

[20] Sohn K, Yan X, Lee H. Learning structured output representation using deep conditional generative models [C]//Proc of International Conference on Neural Information Processing Systems. Cambridge: MIT Press, 2015: 3483-3491.

[21] Bao Jianmin, Chen Dong, Wen Fang, et al. CVAE-GAN: fine-grained image generation through asymmetric training [C]//Proc of IEEE International Conference on Computer Vision. Washington DC: IEEE Computer Society, 2017: 2764-2773.

Note: Figure translations are in progress. See original paper for figures.