electronics

MDPI

Article

Balancing Password Security and User Convenience: Exploring the Potential of Prompt Models for Password Generation

Afamefuna P. Umejiaku, Prastab Dhakal and Victor S. Sheng *

Computer Science Department, Texas Tech University, Lubbock, TX 79409, USA; afamefuna.p.umejiaku@ttu.edu (A.P.U.); prasdhak@ttu.edu (P.D.)
* Correspondence: victor.sheng@ttu.edu

Abstract: With the increasing prevalence of cyber attacks and data breaches, the importance of strong passwords cannot be overstated. Password generating software has been widely used to generate complex passwords that are difficult to crack, but it has its limitations. One of the main problems with this kind of software is that it often generates passwords that are difficult to remember, leading to users write them down or reuse them across multiple accounts. In recent years, prompt models such as ChatGPT have emerged as a promising solution for generating strong and memorable passwords. By leveraging machine learning algorithms, these models can generate unique and complex passwords tailored to individual users' preferences, making them easier to remember and more secure. However, the use of prompt models to generate passwords also raises concerns about exposing vulnerable passwords. Hackers can potentially use these models to predict passwords by analyzing a user's online activity and personal data. Additionally, the constant need to change passwords to stay secure poses a challenge for both password generating software and prompt models. As technology continues to evolve, finding a balance between password security and user convenience remains a complex issue. While prompt models such as ChatGPT can offer a promising solution, it is essential to consider the potential risks and challenges associated with their use, including the constant need for password changes and the potential vulnerability of the generated passwords.

Keywords: prompt models; password; security; ChatGPT

check for updates

Citation: Umejiaku, A.P.; Dhakal, P.; Sheng, V.S. Balancing Password Security and User Convenience: Exploring the Potential of Prompt Models for Password Generation. *Electronics* **2023**, *12*, 2159. https://doi.org/10.3390/electronics12102159

Academic Editors: Juan M. Corchado and Flavio Canavero

Received: 30 March 2023
Revised: 30 April 2023
Accepted: 7 May 2023
Published: 9 May 2023

1. Introduction

In today's digital world, text-based passwords are a popular and cost-effective authentication method on the web due to their simplicity and usability [1–3]. However, passwords pose a major security risk in information systems due to user behavior, such as choosing weak passwords and reusing them, which researchers in the password authentication field refer to as "human factor problems" [3,4]. To ensure secure protection against unauthorized access, it is crucial that passwords meet reasonable standards. As the number of accounts that users need to create passwords for continues to grow—with some estimates as high as 23 [5]—studies have shown that people have adopted several strategies to handle the demand, often using unsecured techniques, such as using a single password for multiple accounts, documenting passwords in unsecured locations [6], using common words or phrases, personal identifiers [7–9], and using password managers [10,11]. Yildirim et al. [12] encourage users to create strong passwords that are memorable by complying with best practices and password policies.

Several guidelines have been proposed, and some organizations enforce these guidelines to mitigate the potential impact of compromised passwords. In 2006, the National Institute of Standards and Technology (NIST) recommended using complex and lengthy passwords and changing them every 90 days (or 180 days for passphrases) for sound password policies [13]. NIST used a heuristics method to measure the strength and efficiency of a password restriction policy, using Shannon's Entropy to calculate password uncertainty [14]. However, research suggests that password restriction policies, especially those based on entropy, may not enhance password security. Studies have shown that enforcing complex and strict password policies can lead to insecure password practices by users, aid attackers in guessing passwords by reducing the number of potential options, and come at a significant cost to usability [15,16]. They have been proved to be less beneficial than expected [4,17,18], as they make passwords difficult to remember and type, causing users to resort to insecure coping strategies. Grawmeyer and Johnson [19] found that highly secure passwords were actually insecure single-word passwords, highlighting the need for password guidelines based on a theoretical understanding of user behavior [20]. Florencio and Herley [21] found that users only tolerate complex policies when there is no other choice, and Inglesant and Sasse [17] suggest that password policies should be designed with Human-Computer Interaction (HCI) principles to help users set strong passwords in specific contexts of use [17,22]. NIST updated their guidelines in 2017, dropping complexity requirements and recommending users to create a new password only in the case of a potential threat or suspected unauthorized access, as frequent password changes can irritate users and lead to workarounds [4,17,18]. They emphasized the importance of password length and recommended the use of long passwords and passphrases [23,24].

The strength of a password refers to its ability to withstand guessing attacks. There are several well-known methods for evaluating password strength, including Probabilistic Context-Free Grammar (PCFG) [25], feature matching [26], and neural networks trained using real-world passwords on a massive scale [27]. In a study by Shay et al. [28], it was found that password strength and usability are not always negatively correlated. An appropriate password creation strategy [12] can ensure the security of a password without sacrificing usability. Schweitzer et al. [29] suggested that users use keyboard patterns to create passwords. Shay et al. [30] suggested the use of blocklists to check user passwords against leaked or easily guessed passwords and providing real-time password strength estimates using strength meters. They also suggested that composition policies should not require specific character classes and that a minimum password length of eight characters should be set based on recommendations by Lee et al. [31] in 2022. Murray and Malone [32] discussed the use of mnemonic passwords as proposed by Barton and Barton [33] and password chunking [34] to create passwords. They also discussed the use of password meters to measure password strength [32] and password checkers used to prevent users from creating simple and common passwords [35]. Weak passwords can be enhanced using several strategies, such as composition-based password enhancement [36], analyze–modify password enhancement [35], password enhancement based on semantic transformation [37], and conversion-based password enhancement [38]. For example, vowels can be converted to consonants or removed from proposed passwords [8].

The growth of artificial intelligence in the last decade has led to several new possibilities, as models can be used to guess passwords exposing users to new security challenges, as models using natural language could detect even more words that ordinarily will not be in a password dictionary, quickly analyze lots of information including past users passwords and personal information, and with prompt models such as ChatGPT capable of generating millions of unique passwords based on a specific pattern given very limited data in very short time, very cheaply exposing users to new challenges [39,40].

The objective of this research is to devise a practical strategy for organizations and system administrators to facilitate users in creating easily secured memorable passwords using prompt models. This research was carried out in four phases.

- Firstly, we collected a sample of user passwords and analyzed the decision-making process involved in creating them to assess their strength.
- Secondly, we applied a set of rules to modify these passwords and evaluated the factors that affect the memorability of the passwords after the modifications were made.
- Thirdly, we input user information and modified passwords into a prompt model, such as ChatGPT, to generate new passwords for the users.

- Finally, we evaluated the effectiveness of using prompt models to develop stronger and more memorable passwords by applying our rules to the passwords generated by the model and assessing their memorability.

2. Materials and Methods

2.1. Research Participants

As part of our research, we collected personal information from our participants, including their names, sex, and level of computer literacy. In addition, we asked for any other personal information that they were comfortable sharing, such as social media profiles, curriculum vitae, and academic work. We also requested that they provide us with five passwords that they had previously used.

A breakdown of the participants' demographics revealed that about 60% of the respondents who took part in the experiment were between the ages of 18 and 23. This could be attributed to the fact that most of the participants were college students. Males accounted for about 56% of the respondents, while females accounted for about 44%. All of the participants in the experiment were able to independently use computers and had been using them for several years. Moreover, they had previously used at least five different passwords. When asked about their understanding of how browsers work and how to stay secure, 63% of the respondents considered themselves to have an intermediate understanding, while only slightly above 5% considered themselves to be experts in the field.

2.2. Dataset

The study utilized three publicly available datasets from Kaggle. One of these datasets, compiled by Alkalay, contains a list of 142k compromised passwords. The second dataset consists of passwords used on the RockYou website before it was compromised, while the third dataset contains 10,000 commonly used passwords. The RockYou dataset is particularly valuable as passwords were stored unencrypted in plaintext format, making it easier to analyze.

To build our password database, we combined the three datasets mentioned above into a large database. We removed duplicated passwords and discarded any passwords longer than 20 characters to maintain a realistic human-provided password dataset. Noisy entries such as "N/A" or a single comma, semicolon, and passwords of a length less than eight characters were also removed.

2.3. Password Strength Evaluation

We used Passfault's Password Strength Tester in this study to evaluate the strength of passwords. Passfault is an open-source tool that measures password complexity and strength in a completely different way. It evaluates passwords and enforces password policy and was developed by The OWASP Foundation and is now maintained by MalwareFox. The original developers of Passfault discontinued the project, but the extensive wordlist of Passfault was integrated into ZXCVBN to offer the best password strength tester on MalwareFox [41,42].

ZXCVBN is a password strength estimator inspired by password crackers. It recognizes and weighs 30k common passwords, common names, and surnames according to U.S. census data, popular English words from Wikipedia and U.S. television and movies, and other common patterns such as dates, repeats (aaa), sequences (abcd), keyboard patterns (qwertyuiop), and l33t speak through pattern matching and conservative estimation. ZXCVBN is an algorithmic alternative to password composition policy and is more secure, flexible, and usable when sites require a minimal complexity score in place of annoying rules like "passwords must contain three of lower, upper, numbers, and symbols". It is based on a solid mathematical foundation [26].

2.4. Prompt Model

We utilized ChatGPT as the prompt model to analyze a given dataset containing passwords and other personal information, such as social media profiles, curriculum vitae, and academic work. In addition, we used ChatGPT to generate passwords based on the data [43].

ChatGPT is a deep learning neural network that has undergone extensive training on a massive amount of text data to produce responses that closely resemble human-like replies to various prompts and questions. Developed by OpenAI and based on the GPT-3.5 architecture, ChatGPT boasts six billion parameters, making it one of the largest language models available today [44].

ChatGPT's creation is part of a more comprehensive natural language processing (NLP) research trend that has witnessed considerable advancements in recent years, particularly with the integration of deep learning techniques [45]. These models have demonstrated their effectiveness in various NLP tasks, such as language translation, text summarization, and question answering [46]. The availability of large amounts of data, along with the evolution of deep learning techniques, has significantly contributed to ChatGPT's ability to generate natural language responses that are remarkably human-like.

2.5. Passwords Modifications

Just as weak or simple and easily guessable passwords can pose a significant security risk, user behavior when required to create multiple passwords or repeatedly change them could also be a challenge. To overcome this challenge, it is essential to implement strategies that can enhance password security and make them difficult to predict.

Several approaches have been devised to enhance password strength and security using rules. Here, we used simple rules aimed at increasing the randomness in passwords or adding mnemonic chunks based on a pattern. Here are the five rules we applied to modify passwords:

1. Substituting vowels with predetermined characters: The first rule involved replacing all vowels in the password with predetermined characters. This approach helps in enhancing password security by increasing the complexity of the password. For example, the word 'password' can be modified as 'p@ssw0rd'.
2. Substituting the most frequently occurring vowel: In the second rule, the most frequently occurring vowel in the password was substituted with a predetermined character. This approach further strengthens the password by introducing an additional level of complexity. For instance, the word predetermined can be modified as pr1d2t3rmin4d.
3. Adding a four-character mnemonic chunk: The third rule involved adding a three- to five-character mnemonic chunk to the password. This chunk is created based on a pattern, such as the first letter of each word in a phrase. This approach not only enhances password security but also makes it easier to remember. For example, the phrase 'My favorite color is blue' can be modified as' 'MfciB'.
4. Combining the mnemonic chunk with vowel substitution: In the fourth rule, the addition of a mnemonic chunk was combined with the substitution of all vowels to further enhance the password strength. This approach introduces multiple levels of complexity, making it extremely challenging to predict the password. For example, the phrase 'I love to play football' can be modified as '1Lv2plFtbll#1234'.
5. Removing all vowels: In the fifth and final rule, all vowels in the password were removed as an additional measure to improve password security. This approach increases the complexity of the password while making it more challenging to predict. For example, the word 'information' can be modified as 'nfrmtn'.

2.6. Memorability Evaluation

We evaluated both the modified passwords and the passwords generated using the ChatGPT prompt model to determine their level of memorability. Our main concern was how difficult these passwords were to remember, given that memorability is a critical concern for knowledge-based authentication systems [8], which are limited by human memory and can compromise system security. Passwords are particularly problematic, ref. [47] as users struggle to remember them and often resort to coping strategies to avoid forgetting or resetting them. Various studies have explored the memorability of passwords, revealing that users tend to create passwords that are easily connected to their accounts as a memory assistance strategy. Additionally, researchers have found a tradeoff between password security and memorability, with many users preferring convenience over security [48].

To improve password memorability, psychological literature suggests that repetition in learning is essential. We explained the techniques used to modify the passwords and asked our research participants to type them three times. Studies show that verifying passwords three times can increase password memorability from 42% to 70%, while increasing verification to two times can increase password memorability by 17% [7].

2.7. Study Sequence

In the first step, a sample of passwords was collected from a database, including both randomly selected passwords from our database and manually entered simple and complex passwords. This allowed for an initial evaluation of the Passfault algorithm's ability to accurately assess the strength of different types of passwords. In the second step, passwords collected from participants were modified using only the suggested modification techniques, and the strength of both the original and modified passwords was evaluated. This step allowed for an assessment of the effectiveness of the modification techniques in providing useful suggestions for improving password strength. In the third step, participants' demographic and personal information, as well as five of their old passwords, were used to generate new passwords. ChatGPT, an AI language model, suggested 50 passwords each time using one of the suggested modification techniques after suggesting 50 passwords with no conditions attached, and the strength of these passwords was evaluated. In the fourth step, participants were asked if they had likely used any of the generated passwords or if they had considered using them in the past. This step provided valuable insight into the usability and real-world applicability of the generated passwords. Finally, in the fifth step, participants were asked to select five of the passwords suggested by ChatGPT and type them three times, along with the five original modified passwords. Participants were then asked to rate the likelihood of using the generated passwords, providing important feedback on both the memorability and usability of the passwords suggested by ChatGPT. This step was particularly important in assessing the effectiveness of using prompt models in suggesting passwords that are both secure and human-like.

Figure 1 below is a flow chart that illustrates the steps taken in the study. The flow chart is divided into two parts: Step 1 focuses on evaluating the Passfault algorithm. We used Passfault to estimate the strength of the passwords users provided, passwords generated by ChatGPT, and the passwords modified using our suggested password modification techniques.

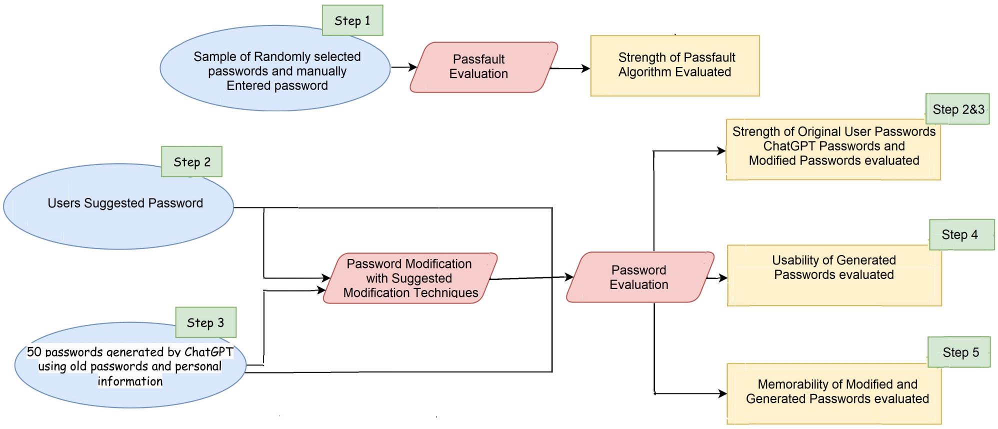

Step 1
Sample of Randomly selected passwords and manually Entered password
Passfault Evaluation
Strength of Passfault Algorithm Evaluated
Step 2&3
Strength of Original User Passwords ChatGPT Passwords and Modified Passwords evaluated
Step 2
Users Suggested Password
Password Modification with Suggested Modification Techniques
Password Evaluation
Step 4
Usability of Generated Passwords evaluated
Step 3
50 passwords generated by ChatGPT using old passwords and personal information
Step 5
Memorability of Modified and Generated Passwords evaluated

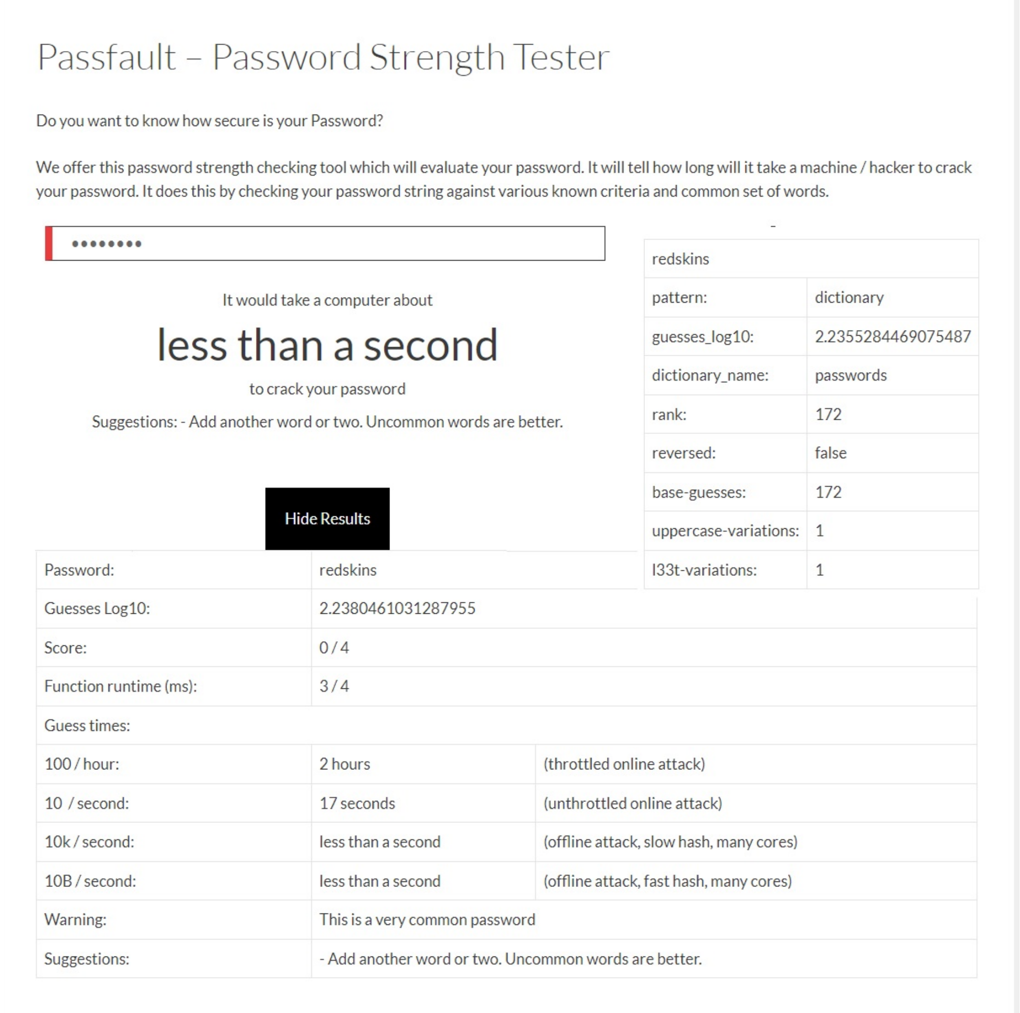
Figure 1. Flow Chart.

In the second part of the flow chart, Steps 2 to 5, we evaluated the strength, usability, and memorability of three types of passwords: the original user passwords, passwords generated by ChatGPT, and passwords modified using our suggested techniques. We determined the effectiveness of our password modification techniques in enhancing the strength, usability, and memorability of the passwords.

3. Findings

3.1. Password Strength

The Passfault password strength tester proved to be a highly effective algorithm in testing the strength of passwords and providing users with useful feedback to improve their password decisions, as seen in Figure 2. In our evaluation, we randomly selected 1000 passwords from our database and tested them using Passfault. Passfault indicated that all of the randomly selected passwords from our database of leaked passwords would be cracked within a second. Furthermore, Passfault was also able to accurately assess the strength of simple passwords that contained dictionary words or were short in length, indicating that such passwords could also be easily cracked in a matter of seconds. On the other hand, Passfault was able to recognize the strength of more complex passwords, indicating that they could take years to centuries to crack. Passfault's accurate evaluation of password strength and its valuable feedback make it a powerful tool for users.



Passfault – Password Strength Tester

Do you want to know how secure is your Password?

We offer this password strength checking tool which will evaluate your password. It will tell how long will it take a machine / hacker to crack your password. It does this by checking your password string against various known criteria and common set of words.

••••••••

It would take a computer about

less than a second

to crack your password

Suggestions: - Add another word or two. Uncommon words are better.

redskins	
pattern:	dictionary
guesses_log10:	2.2355284469075487
dictionary_name:	passwords
rank:	172
reversed:	false
base-guesses:	172
uppercase-variations:	1
l33t-variations:	1

Password:	redskins	
Guesses Log10:	2.2380461031287955	
Score:	0 / 4	
Function runtime (ms):	3 / 4	
Guess times:		
100 / hour:	2 hours	(throttled online attack)
10 / second:	17 seconds	(unthrottled online attack)
10k / second:	less than a second	(offline attack, slow hash, many cores)
10B / second:	less than a second	(offline attack, fast hash, many cores)
Warning:	This is a very common password	
Suggestions:	- Add another word or two. Uncommon words are better.	

Figure 2. Passfault—Password Strength Tester.

After validating Passfault for evaluating passwords, we used it to assess the passwords provided by users, as well as the passwords modified using our suggestions and those generated by ChatGPT. Passwords that could be cracked in less than 3 months were considered weak, those that could be cracked in less than 1 year but more than 3 months were considered okay, and those that would take years to centuries to crack were considered strong.

As expected, many of the passwords submitted by participants were weak or okay, accounting for 59% of the total. The modified passwords performed significantly better indicating our password enhancement techniques had a positive effect. Changing all the vowels resulted in 80% of the passwords being classified as strong, while changing a single vowel resulted in 77% being considered strong. Removing all vowels resulted in 73% being strong, and adding a mnemonic chunk also improved the password strength. Combining a mnemonic chunk with vowel changes resulted in 100% of the passwords being classified as secure. Additionally, 99% of the passwords suggested by the prompt model were considered strong. The results can be seen in Figure 3.

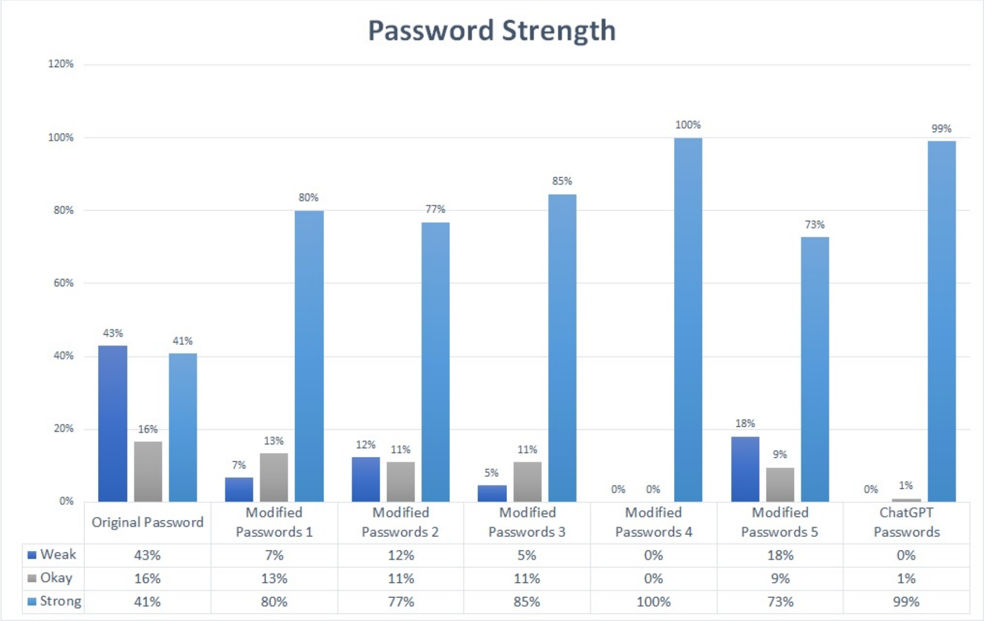


Figure 3. Password Strength.

3.2. *Usability and Current Exposure*

In order to better understand the practicality of passwords generated by prompt models, participants were asked if they had ever used or considered using any of the generated passwords in the past. As presented in Figure 4, it was found that a prompt model such as ChatGPT was able to suggest passwords that 90% of the respondents agreed were either used or likely to be used in the future, with 66% of the participants strongly agreeing that they would likely have used or would use at least one of the 50 suggested passwords. When respondents were presented with passwords generated with certain conditions attached, they generally did not relate well to them. Passwords modified by altering only the vowels performed similarly ok. For criteria 1, only 8% strongly agreed that at least one of the 50 suggested passwords has been used or would be used, for criteria 2, there was a strong prediction probability of 12%, and for criteria 5, 16% strongly agreed that at least one of the 50 suggested passwords has been used or would be used. Adding a mnemonic chunk was not as effective in improving the password's resistance to attacks, with 20% strongly agreeing that at least one of the 50 suggested passwords has been used or would be used, and a further 19% agreeing on average that at least one of the 50 suggested passwords has been used or would be used. We found that by adding a mnemonic chunk and manipulating the vowels in the password, we were able to produce passwords that were much more difficult for prompt models to predict. None of the participants strongly or even averagely agreed that they would have generated or used the password on their own.

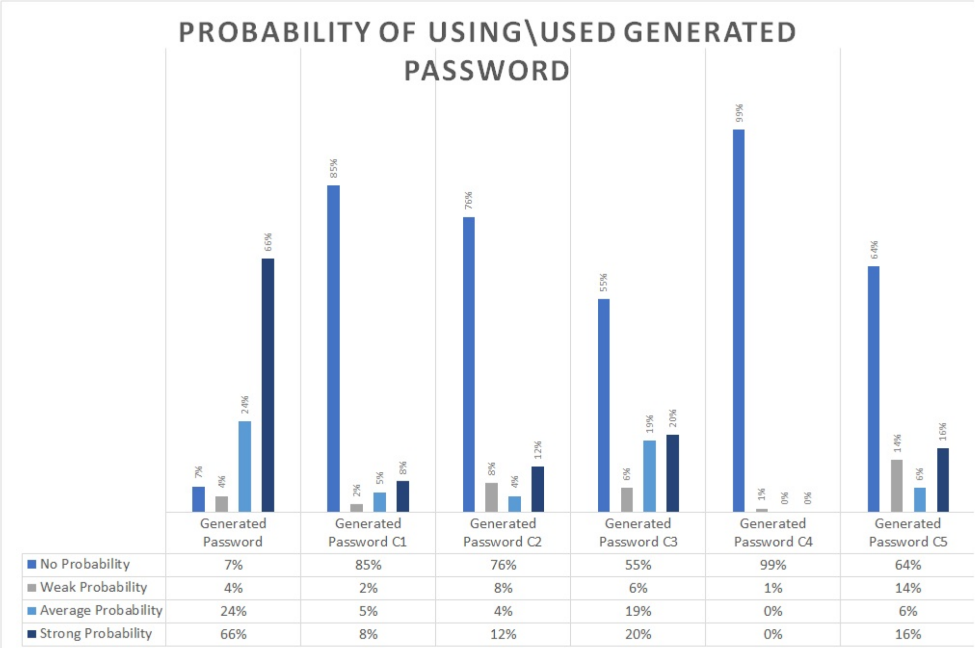
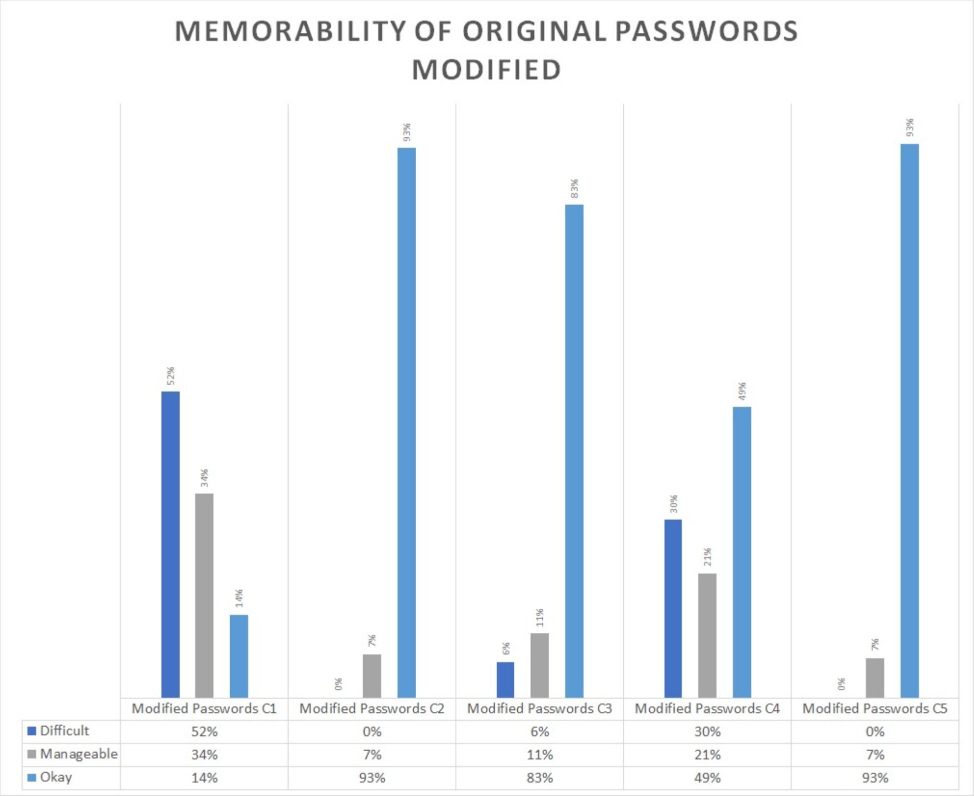


Figure 4. Probability of using or having used a password generated by ChatGPT.

These results show that prompt model-generated passwords were concerning from a security standpoint. The fact that 90% of the respondents agreed that at least one of the 50 suggested passwords was likely to be used by them indicates that these passwords were not particularly strong or unique to an attack that poses a small amount of personal information and knowledge of the passwords we have used in the past. This means that if a hacker gained access to the list of passwords generated by a prompt model, there is a high likelihood that they could successfully use one of them to gain unauthorized access to an individual's accounts.

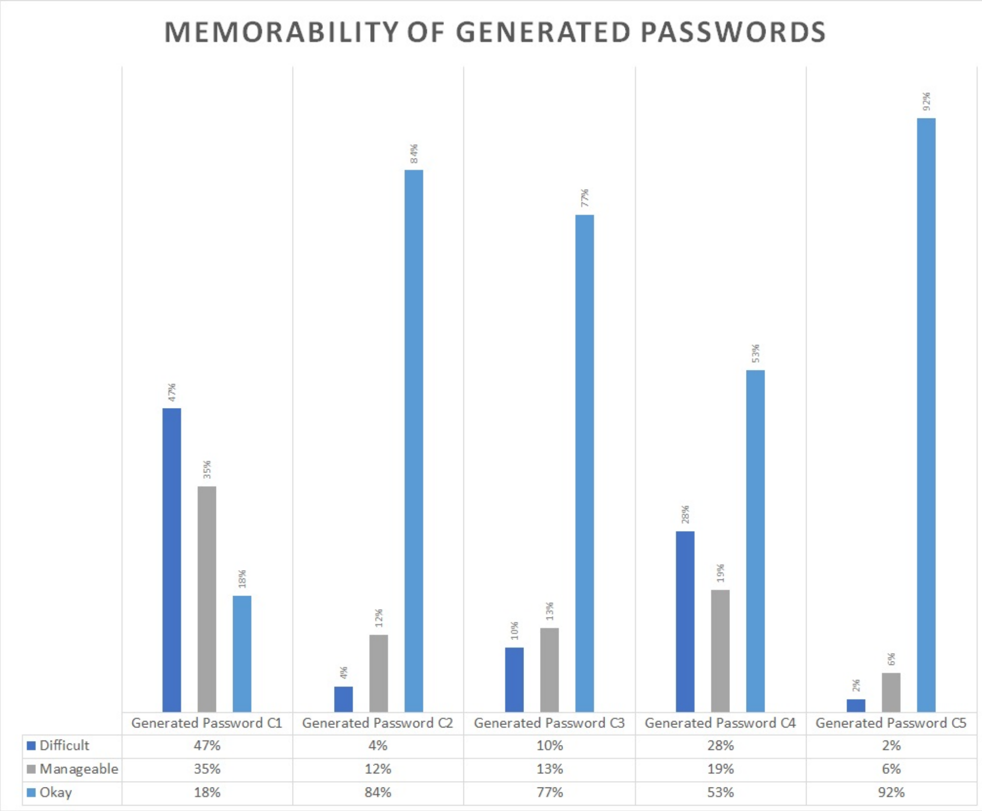
3.3. Memorability

As the users' ability to quickly and easily memorize suggested passwords is a critical concern, as explained in Section 2.6 the respondents were made aware of how the passwords were manipulated. They were asked to type down both the modified passwords and the password suggested by the prompt model three times to increase their capacity to memorize the passwords. As seen in Figures 5 and 6, the respondents encountered the most difficulty in recalling their passwords that were modified based on criteria 1, which involves changing all vowels. On the other hand, passwords that were modified based on criteria 2, which is changing a single vowel, were easier. Criteria 3, adding a mnemonic chunk, and criteria 5, deleting all vowels, were largely easy for the respondents to remember. However, it was also a challenge for users to recall passwords when a mnemonic chunk was added along with changing the vowels. Although the original modified passwords proved to be a little more memorable than those suggested by the models, the results did indicate that passwords generated by prompt models performed just as well and could be used as an effective way of generating memorable passwords.



	Modified Passwords C1	Modified Passwords C2	Modified Passwords C3	Modified Passwords C4	Modified Passwords C5
■ Difficult	52%	0%	6%	30%	0%
■ Manageable	34%	7%	11%	21%	7%
■ Okay	14%	93%	83%	49%	93%

Figure 5. Likely to recall password.



	Generated Password C1	Generated Password C2	Generated Password C3	Generated Password C4	Generated Password C5
■ Difficult	47%	4%	10%	28%	2%
■ Manageable	35%	12%	13%	19%	6%
■ Okay	18%	84%	77%	53%	92%

Figure 6. Likely to recall password.

4. Limitations and Further Recommendations

It is crucial to take into account the context of any user study to accurately interpret the results. In this particular study, it is important to note that our participants were primarily composed of undergraduate and graduate students, which may not be representative of the general population. Due to their higher education levels, they may have a greater awareness of security best practices and be more cautious with their passwords. Moreover, the passwords we collected were no longer in use and were not used to protect high-value accounts. As a result, these passwords may not fully represent current password practices, and the study's findings may not be applicable to the password habits of the general population or those protecting high-value accounts.

Furthermore, our study specifically focused on automated offline text-based password-guessing attacks, which means that other types of password schemes or real-life threats to password security, such as phishing and shoulder surfing, were not considered. It is important to recognize that there are many potential avenues for attackers to exploit password security beyond automated guessing attacks, and future studies should consider a broader range of threats to fully understand the effectiveness of password protection.

Finally, we suggest that future research explores the development of a prompt model focused on generating passwords. While our study was able to generate passwords using ChatGPT, a general-purpose language model, there may be room for improvement if a specialized prompt model is developed specifically for generating passwords. A specialized model may be better equipped to provide more secure and unique password suggestions that are tailored to individual users and their unique requirements.

5. Conclusions

While text-based passwords remain a widely used method of authentication, they pose a significant security risk due to common human behaviors such as using weak passwords, reusing passwords across multiple accounts, and storing passwords in unsecured locations. To mitigate this risk, it is essential to establish reasonable password standards that ensure secure protection against unauthorized access. However, implementing overly complex and stringent password policies can lead to user frustration, reduce usability, and ultimately result in insecure password practices. To address this challenge, password policies should be designed based on an understanding of user behavior and with HCI principles in mind. By taking into account how users interact with password authentication systems, password policies can help users set strong passwords while maintaining usability. For example, password policies can encourage the use of passphrases, which are longer and easier to remember than traditional passwords. In addition to establishing password policies, it is also essential to evaluate the effectiveness of independently created passwords. Our evaluation found that even seemingly secure passwords could be vulnerable to prompt models fed with relevant, often publicly known information. To address this issue, we utilized prompt models to generate passwords that were both memorable and secure. When we set different password composition policies, our users found the resulting passwords memorable once they understood the enhancement policies and readily accepted them. Prompt models can generate passwords based on any desired password composition policy using information that users can easily relate to, making them a promising solution for balancing password security and user convenience. By generating passwords that users can easily memorize, prompt models can help to reduce the incidence of insecure password practices such as password reuse and storing passwords in unsecured locations. In conclusion, establishing reasonable password standards and designing password policies based on an understanding of user behavior and HCI principles are critical for ensuring secure protection against unauthorized access while maintaining usability. Prompt models can serve as a useful tool for generating secure passwords that users can easily memorize, reducing the incidence of insecure password practices. By adopting a holistic approach to password security, organizations can effectively protect their systems while also ensuring a positive user experience.

Author Contributions: Conceptualization, A.P.U. and V.S.S.; Methodology, A.P.U.; Validation, P.D.; Formal analysis, A.P.U.; Investigation, A.P.U. and P.D.; Data curation, A.P.U.; Writing—original draft, A.P.U.; Writing—review & editing, A.P.U.; Visualization, A.P.U.; Supervision, V.S.S.; Project administration, V.S.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data available on request due to restrictions.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bonneau, J.; Herley, C.; Oorschot, P.C.; Stajano, F. The quest to replace passwords: A framework for comparative evaluation of web authentication schemes. In Proceedings of the 2012 IEEE Symposium on Security and Privacy, San Francisco, CA, USA, 20–23 May 2012. [CrossRef]
2. Forget, A. A World with Many Authentication Schemes. Ph.D. Thesis, Carleton University, Ottawa, ON, USA, 2012. [CrossRef]
3. Herley, C.; Van Oorschot, P.C.; Patrick, A.S. Passwords: If we're so smart, why are we still using them? In *Financial Cryptography and Data Security*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 230–237. [CrossRef]
4. Summers, W.C.; Bosworth, E. Password Policy: Proceedings of the Winter International SYNPOSIUM on Information and Communication Technologies. January 2004. Available online: https://dl.acm.org/doi/10.5555/984720.984724 (accessed on 28 March 2023).
5. McAfee. The Past, Present, and Future of Password Security. September 2021. Available online: https://www.mcafee.com/blogs/internet-security/security-world-password-day/ (accessed on 28 March 2023).
6. Alomari, R.; Thorpe, J. On password behaviours and attitudes in different populations. *J. Inf. Secur. Appl.* **2019**, *45*, 79–89. [CrossRef]
7. Woods, N.; Siponen, M. Improving password memorability, while not inconveniencing the user. *Int. J. -Hum.-Comput. Stud.* **2019**, *128*, 61–71. [CrossRef]
8. Alhamed, A.; Bhatia, S. VowPass: Novel method to generate secure and memorable passwords. In Proceedings of the 2021 4th International Conference on Signal Processing and Information Security (ICSPIS), Virtually, 24–25 November 2021. [CrossRef]
9. Woods, N.; Siponen, M. Too many passwords? how understanding our memory can increase password memorability. *Int. J. -Hum.-Comput. Stud.* **2018**, *111*, 36–48. [CrossRef]
10. Pearman, S.; Zhang, S.A.; Bauer, L.; Christin, N.; Cranor, L.F. Why People (Don't) Use Password Managers Effectively: Proceedings of the Fifteenth USENIX Conference on Usable Privacy and Security. August 2019. Available online: https://dl.acm.org/doi/10.5555/3361476.3361500 (accessed on 28 March 2023).
11. Alkaldi, N.; Renaud, K.; Mackenzie, L. Encouraging Password Manager Adoption by Meeting Adopter Self-Determination Needs. In Proceedings of the 52nd Hawaii International Conference on System Sciences (HICSS), Maui, HI, USA, 8–11 January 2019; pp. 4824–4833.
12. Yıldırım, M.; Mackie, I. Encouraging users to improve password security and memorability. *Int. J. Inf. Secur.* **2019**, *18*, 741–759. [CrossRef]
13. Burr, W.; Dodson, D.; Polk, W. Electronic Authentication Guideline. April 2006. Available online: https://csrc.nist.gov/publications/detail/sp/800-63/archive/2006-04-30 (accessed on 28 March 2023).
14. Shannon, C.E. Prediction and entropy of printed english. *Bell Syst. Tech. J.* **1951**, *30*, 50–64. [CrossRef]
15. Adams, A.; Sasse, M.A. Users are not the enemy. *Commun. ACM* **1999**, *42*, 40–46. [CrossRef]
16. Keith, M.; Shao, B.; Steinbart, P.J. The usability of passphrases for authentication: An empirical field study. *Int. J. -Hum.-Comput. Stud.* **2007**, *65*, 17–28. [CrossRef]
17. Inglesant, P.G.; Sasse, M.A. The true cost of unusable password policies. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Atlanta, GA, USA, 10–15 April 2010. [CrossRef]
18. Komanduri, S.; Shay, R.; Kelley, P.G.; Mazurek, M.L.; Bauer, L.; Christin, N.; Cranor, L.F.; Egelman, S. Of passwords and people. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Vancouver, BC, Canada, 7–12 May 2011. [CrossRef]
19. Grawemeyer, B.; Johnson, H. Using and managing multiple passwords: A week to a View. *Interact. Comput.* **2011**, *23*, 256–267. [CrossRef]
20. Weir, M.; Aggarwal, S.; Collins, M.; Stern, H. Testing metrics for password creation policies by attacking large sets of revealed passwords. In Proceedings of the 17th ACM Conference on Computer and Communications Security, Chicago, IL, USA, 4–8 October 2010. [CrossRef]
21. Florêncio, D.; Herley, C. Where do security policies come from? In Proceedings of the Sixth Symposium on Usable Privacy and Security, Redmond, WA, USA, 14–16 July 2010. [CrossRef]

22. Anon. Cracking Passwords in the Cloud: Insights on Password Policies. Available online: http://news.electricalchemy.net/2009/10/password-cracking-in-cloud-part-5.html (accessed on 28 March 2023).
23. Grassi, P.; Garcia, M.; Fenton, J. Digital Identity Guidelines. March 2020. Available online: https://csrc.nist.gov/publications/detail/sp/800-63/3/final (accessed on 28 March 2023).
24. Anon. Password Policy Best Practices for Strong Security in AD. Available online: https://www.netwrix.com/password_best_practice.html (accessed on 28 March 2023).
25. Weir, M.; Aggarwal, S.; de Medeiros, B.; Glodek, B. Password cracking using probabilistic context-free grammars. In Proceedings of the 2009 30th IEEE Symposium on Security and Privacy, Oakland, CA, USA, 17–20 May 2009. [CrossRef]
26. Wheeler, D.L. Zxcvbn: Low-Budget Password Strength Estimation. In Proceedings of the 25th USENIX Security Symposium (USENIX), Austin, TX, USA, 10–12 August 2016; pp. 157–173.
27. Melicher, W.; Ur, B.; Segreti, S.M.; Komanduri, S.; Bauer, L.; Christin, N.; Cranor, L.F. Fast, Lean, and Accurate: Proceedings of the 25th USENIX Conference on Security Symposium. August 2016. Available online: https://dl.acm.org/doi/10.5555/3241094.3241109 (accessed on 28 March 2023).
28. Shay, R.; Komanduri, S.; Durity, A.L.; Huh, P.S.; Mazurek, M.L.; Segreti, S.M.; Ur, B.; Bauer, L.; Christin, N.; Cranor, L.F. Designing password policies for strength and usability. *ACM Trans. Inf. Syst. Secur.* **2016**, *18*, 1–34. [CrossRef]
29. Schweitzer, D.; Boleng, J.; Hughes, C.; Murphy, L. Visualizing Keyboard Pattern passwords. In Proceedings of the 2009 6th International Workshop on Visualization for Cyber Security, Atlantic City, NJ, USA, 11 October 2009. [CrossRef]
30. Shay, R.; Komanduri, S.; Durity, A.L.; Huh, P.S.; Mazurek, M.L.; Segreti, S.M.; Ur, B.; Bauer, L.; Christin, N.; Cranor, L.F. Can long passwords be secure and usable? In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Toronto, ON, Canada, 26 April–1 May 2014. [CrossRef]
31. Lee, K.; Sjöberg, S.; Narayanan, A. Password policies of most top websites fail to follow best practices. In Proceedings of the 2022 ACM Conference on Computer and Communications Security (CCS '22), Virtual Event, 24–28 October 2022; pp. 1075–1090.
32. Murray, H.; Malone, D. Evaluating password advice. In Proceedings of the 2017 28th Irish Signals and Systems Conference (ISSC), Killarney, Ireland, 20–21 June 2017. [CrossRef]
33. Barton, B.F.; Barton, M.S. User-friendly password methods for computer-mediated information systems. *Comput. Secur.* **1984**, *3*, 186–195. [CrossRef]
34. Miller, G.A. The magical number seven, plus or minus two: Some limits on our capacity for processing information. 1956. Available online: https://pubmed.ncbi.nlm.nih.gov/8022966/ (accessed on 28 March 2023).
35. Yang, S.; Ji, S.; Beyah, R. DPPG: A dynamic password policy generation system. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 545–558. [CrossRef]
36. Furnell, S.; Khern-am-nuai, W.; Esmael, R.; Yang, W.; Li, N. Enhancing security behaviour by supporting the user. *Comput. Secur.* **2018**, *75*, 1–9. [CrossRef]
37. He, D.; Yang, X.; Zhou, B.; Wu, Y.; Cheng, Y.; Guizani, N. Password enhancement based on Semantic Transformation. *IEEE Netw.* **2020**, *34*, 116–121. [CrossRef]
38. Kakarla, T.; Mairaj, A.; Javaid, A.Y. A real-world password cracking demonstration using Open source tools for instructional use. In Proceedings of the 2018 IEEE International Conference on Electro/Information Technology (EIT), Rochester, MI, USA, 3–5 May 2018. [CrossRef]
39. Li, H.; Chen, M.; Yan, S.; Jia, C.; Li, Z. Password guessing via neural language modeling. In Proceedings of the Machine Learning for Cyber Security: Second International Conference, ML4CS 2019, Xi'an, China, 19–21 September 2019; pp. 78–93. [CrossRef]
40. Schroeder, W. DeepPass-Finding Passwords with Deep Learning. June 2022. Available online: https://posts.specterops.io/deeppass-finding-passwords-with-deep-learning-4d31c534cd00 (accessed on 28 March 2023).
41. Anon. Passfault—Password Strength Tester. March 2023. Available online: https://www.malwarefox.com/passfault/ (accessed on 28 March 2023).
42. Rodrigues, B.; Paiva, J.; Gomes, V.; Morris, C.; Calixto, W. Passfault: An Open Source Tool for Measuring Password Complexity and Strength. In Proceedings of the 8th International Multi-Conference on Complexity, Informatics and Cybernetics (IMCIC), Orlando, FL, USA, 21–24 March 2017.
43. Anon. Introducing Chatgpt. Available online: https://openai.com/blog/chatgpt (accessed on 28 March 2023).
44. Anon. GPT-3 Powers the Next Generation of Apps. Available online: https://openai.com/blog/gpt-3-apps/ (accessed on 28 March 2023).
45. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to Sequence Learning with Neural Networks: Proceedings of the 27th International Conference on Neural Information Processing Systems—Volume 2. December 2014. Available online: https://dl.acm.org/doi/10.5555/2969033.2969173 (accessed on 28 March 2023).
46. Vaswani, A.; Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need: Proceedings of the 31st International Conference on Neural Information Processing Systems. December 2017. Available online: https://dl.acm.org/doi/10.5555/3295222.3295349 (accessed on 28 March 2023).

47. Vu, K.L.; Proctor, R.W.; Bhargav-Spantzel, A.; Tai, B.; Cook, J.; Schultz, E.E. 2007. Improving password security and memorability to protect personal and organizational information. *Int. J. -Hum.-Comput. Stud.* **2007**, *65*, 744–757. [CrossRef]
48. Chiasson, S.; Forget, A.; Stobert, E.; van Oorschot, P.C.; Biddle, R. Multiple password interference in text passwords and click-based graphical passwords. In Proceedings of the 16th ACM Conference on Computer and Communications Security, Chicago, IL, USA, 9–13 November 2009. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.