
The 23rd Winona Computer Science Undergraduate Research Symposium

April 25, 2023
9:00am to 11:00am
Watkins 102

Winona State University
Winona, MN

Sponsored by the Department of Computer Science
at Winona State University



Table of Contents

Title	Author	Page
<i>Comparing Performances of Real-time Communication Performance in Web sockets using Socket.io and Long-Polling using Ajax</i>	Ei Myatone Aung	1
<i>Gamification of Software Leads to Faster Learning</i>	Ruth Ann Carlson	5
<i>Extreme Gradient Boosting and Adaptive Boosting to Predict Mortality of Patients in ICU</i>	Sheikh Fahad	9
<i>Gamification of Recipe-Viewing Software Leads to an Enhanced User Experience</i>	Justin Fuller	13
<i>Investigating Ease-of-Use in GUIs vs. CLIs on User Feedback</i>	Matthew Jilk	17
<i>Investigating the Comprehension of GPT's Interpretation of Word Meanings</i>	Longge Yuan	24

Comparing Performances of Real-time Communication Performance in Web sockets using Socket.io and Long-Polling using Ajax

Ei Myatnoe Aung
Department of Computer Science
Winona State University
Winona, MN
507-703-3129
eimyatnoeaung98@gmail.com

ABSTRACT

Short polling, long polling, server-sent events, and web sockets are current techniques to accomplish real time communication in web applications. They are based on a client-server communication model. Among these techniques, long polling and web sockets are most frequently used for real life projects. Long polling keeps the connection alive until there is valid response from the server, and terminates the connection starts a new connection, whereas web sockets keep the connection open regardless until client or server leaves the connection. This study compares the performance of the two most popular real-time communication techniques, long polling and web sockets. Real-time data dashboard was implemented in two versions: one in web sockets using Socket.io library and another in long polling using Ajax. The database in the backend was constantly updated at random intervals and all the changes were sent back to client side at real time through the respective real time techniques. The performance of the two technologies were then compared based on number of requests made, total response time, and CPU utilization. The final testing results suggest that web sockets performance is comparable to that of long polling.

General Terms

Synchronous Communication, Realtime data, Performance, Memory, Message Frequency, CPU Usage

Keywords

Web Sockets, Long-Polling, Ajax, Sockets.io, CPU, Realtime data dashboard, HTTP, TCP, Latency

1. INTRODUCTION

As the internet has become an indispensable part of people's life, the services and requirements surrounding web products have changed immensely. In the past, static websites, routing, delayed responses were acceptable, and not much focus was placed on other alternatives regardless of immense waste in resources and overhead costs. However, as the technology evolved and more alternatives are created, web apps are including more dynamic and interactive updates and real time communication.

The traditional architecture of the web is built upon client-server model, where the client makes a request from server and server sends back the response [3]. Such communication is one-directional, where only client can send request and pull response from the server, and server sends back response only when requests are received. However, the challenge of the client-server model is that it does not allow the server to push updates to the clients, if there are any, in real time. This is especially challenging for web apps that require real time data updates multiple times per second. Examples of some web apps which require real time communication are chat applications, collaborative editing such as <https://www.google.com/docs/about/>, <https://www.figma.com/>, real time data dashboards, and so on. In these use cases, it is preferred that the server be allowed to push updates whenever updates become available, instead of waiting for the client to establish a connection. This whole process is called bidirectional or real time communication.

Investing in two-way communication not only enables real time updates, but also saves hardware cost. For example, polling, while implementing real time communication, could come with waste of HTTP headers and requests, thus causing network traffic. Server-Sent events are buffered by firewalls and proxy servers, could potentially cause connection timeouts, with HTTP headers adding to overheads. Therefore, technology where there are less requirements for both client and server and less overhead and load for server is preferred. Unlike these technologies, Web sockets is known to have minimal header overhead and reduces network traffic and latency. Therefore, it is important to compare what is the best technology to implement real time communication by looking at how much each technology consumes CPU and causes latency.

2. BACKGROUND KNOWLEDGE

Well-known technologies to implement bidirectional communication are Long/short polling (client pull), WebSocket (server push), and Server-Sent Events (Server push).

Long/short polling are client pull polling techniques where the client decides the time intervals to ask the server for data updates and the server returns the response through Hypertext Transfer Protocol (HTTP) request. The difference between long/short polling is that short polling is where the client first initiates a connection by sending a request to the server. Then, the server returns with a new update or empty response and ends the connection.

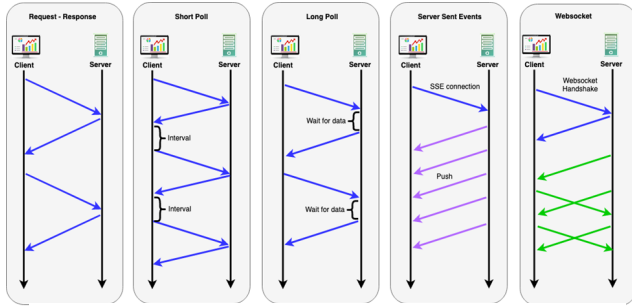


Figure 1. How each real time technology works

In contrast, the long polling is where the client makes a request to the server and the connection stays open until there is an update from the server [5]. Server push is a state where there is connection between the server and client where both server and the client can send/request responses.

Server-Sent events (SSE) pushes messages or events from server to client through HTTP request. When the client sends a request to the server for event stream from the server, HTTP connection is established. This connection is kept alive based on the header's components sent by the client and allows servers to push events when they are ready to the client side. SSE is restricted to transporting UTF-8 message format only, thus binary data is not supported. Web Sockets is a computer communication protocol that sends and receives Internet Protocol (IP) packets between client and server by establishing Transmission Control Protocol (TCP) connection. It allows full duplex (two-way communication) over the TCP connection, where client sends a request to the servers to open a connection, the server then agrees by sending back a switching protocol response, and once this handshake between server and client is complete, the TCP connection is established. After these steps, client and server can exchange messages back and forth until terminated by one of the parties. The difference between web sockets and server-sent events is that web sockets allow bi-directional communication through TCP connection where both clients and servers can push updates, whereas with server-sent events, server only can send updates in the form of a stream of event updates.

3. HYPOTHESIS

Web sockets perform better at implementing real time communication than Long Polling.

4. METHODOLOGY

The goal of the study is to compare which technique yields better performance (CPU usage, delays, etc.) when data is retrieved in real time from database and clients can communicate in real time. The simulation program used is a real-time data dashboard that shows the data updates in real time. Data updates are loaded from a database, which is automated to add new data at random intervals. Two versions of this program are developed: one is implemented in Web sockets using Socket.io and the other is implemented in long polling using Ajax

These two cases are chosen to test how the two technologies perform. Use case one is where 50, 100, 150, 200, 250 records are added to database at same intervals and monitor the total elapsed time and average elapsed time for each technology. Use case two is where random number records are added to database at random

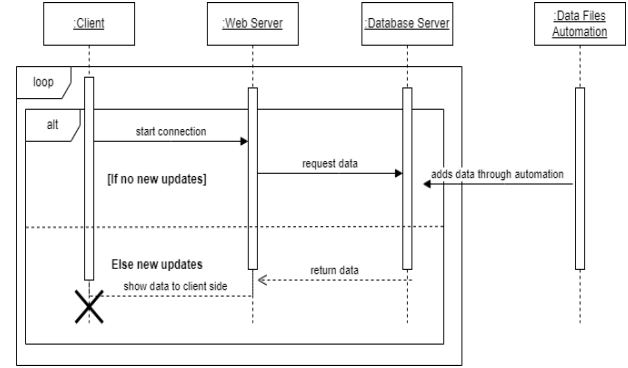


Figure 2. Design of Program (Long Polling) explains how server side (client) and web server(server) sides communicate in long polling and how data files are added to database server through automation script

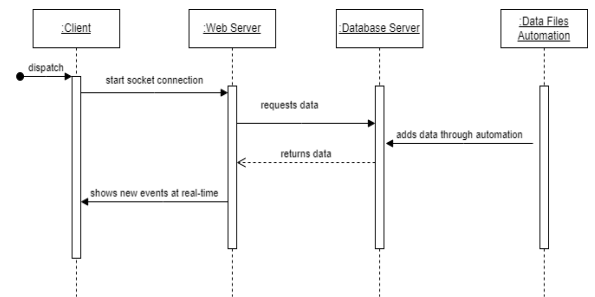


Figure 3. Design of Program (Web Sockets) explains how server side (client) and web server(server) sides communicate using web sockets and how data files are added to database server through automation script

time intervals to see how the random response of the two technologies in rendering the new records to client side. The results of the testing cases are then summarized for further analysis of the performance between Web sockets and Long Polling.

4.1 Testing

The testing metrics chosen to measure the performance of the simulation program are response time, number of requests, and CPU utilization. Response time measures the total time taken from the time client sends a request to the time the server returns the response to the client side, where it will be rendered on the interface. The best scenario is to have fast response time. Number of requests measures the HTTP requests made by client during the process to the server. The best scenario is to have as few requests as possible, thus reducing network traffic. CPU Utilization measures how web sockets and long polling consume CPU during the process. The best scenario is to have low CPU utilization since we want less work on the CPU for rendering.

Chrome Developer Tool is used to collect the necessary data for the testing metrics. Chrome Developer Tool is a built-in web developer tool that helps with debugging, logging, performance audit, networking statistics, and memory management.

4.1.1 Response Time

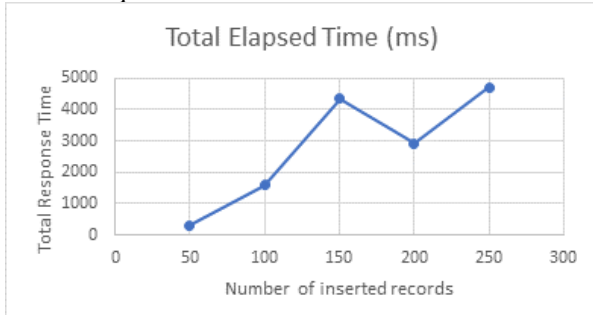


Figure 4. Response Time for Web sockets explores the relationship between where number of inserted records to database(x-axis) and total of response time (y-axis).

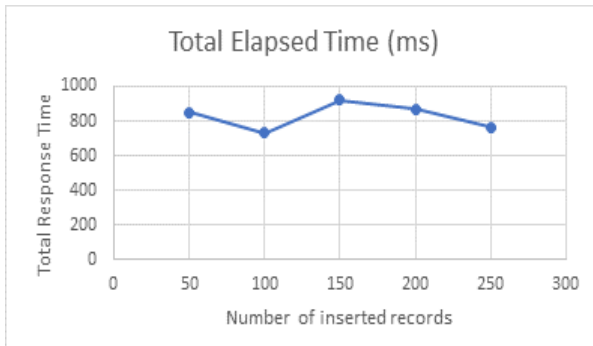


Figure 5. Response Time for Long Polling explores the relationship between number of inserted records to database (x-axis) and total of response time (y-axis).

According to Figure 4 and Figure 5, The program implemented using Socket.io has more fluctuations in terms of response time. As the number of new records rises, the total response time for web sockets version rises as well. In contrast, the long polling version returns more stable response time statistics. Regardless of the number of inserted records, the total response time stays around 800 milliseconds to 1000 milliseconds in long polling version. Based on these statistics, it is estimated that web sockets are more likely to be affected by amount of data because it uses packets and transforms into hypertext documents which consumes higher storage, while long polling relies on HTTP connection, which exchanges data transfer in terms of hypertext documents directly. To conclude, web sockets seem to take longer to return data updates than long polling does.

4.1.2 Number of Requests

According to Figure 6, In terms of number of requests made by each technology, it seems that web socket version has drastically lower number of requests than long polling. Long polling starts a request whenever it finishes rendering the first update and starts another request to listen to future updates, thus creating multiple requests, depending on the frequency of date updates. In contrast, web sockets only have one connection open, and all updates are communicated over the same connection until client or server terminates the connection. To conclude, long polling seems to cause busier traffic than web sockets.

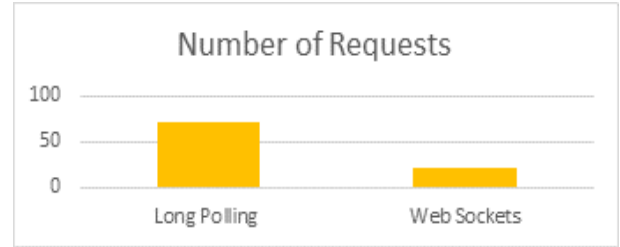


Figure 6. Number of Requests Long Polling vs Web sockets shows the number of requests for each real time technology

4.1.3 CPU Utilization

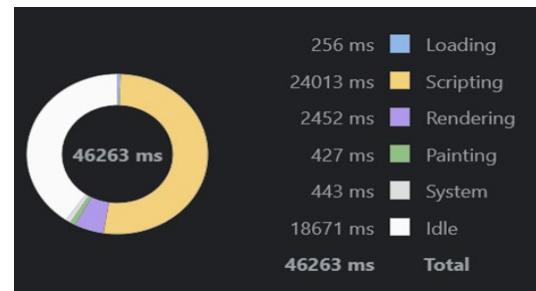


Figure 7. Web sockets CPU Utilization

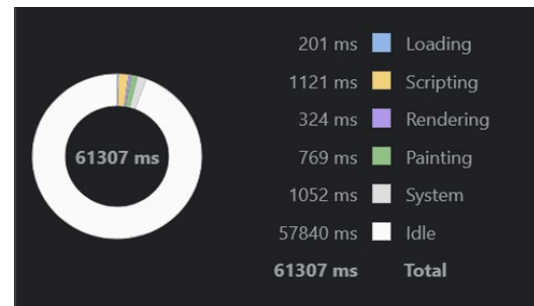


Figure 8. Long Polling CPU Utilization

According to Figure 7 and Figure 8, in terms of CPU Utilization, web sockets and long polling yield noticeably different numbers in four areas: scripting, rendering, system, and idle. Scripting means parsing and compiling JavaScript codes. Long polling version yields less scripting time than web sockets version. This could be due to the web sockets version implemented in JavaScript as server-side language. So, the amount of time for back-end operations is included in the scripting statistics, whereas the long polling version implemented in server-side scripting language, PHP, might not be reflected in the scripting statistics. Thus, web sockets could consume more CPU, compiling JavaScript code. This could result in CPU being more active when it comes to web sockets version. In terms of idle time, from start to finish, CPU was more active running the web socket dashboard version than the long polling version. It could be said that CPU resources are being used more for running the web sockets version than long polling version. To conclude, long polling consumes less CPU tasks, compared to web sockets.

5. RESULTS

Based on the results examined, Web sockets reduce network traffic by not having multiple requests for each update. However, for other metrics such as CPU utilization and response time, web sockets have higher yield. This could be because web sockets are made up of TCP protocol and are considered more secure and consistent connection. Moreover, the response time could take longer because it is a connection-oriented protocol. Thus, it must store some data frames. In contrast, long Polling has a much busier traffic than web sockets, proved by high number of requests. However, for other metrics, long polling returns much lower statistics compared to web sockets. This could be because long polling follows HTTP protocol, which is a connectionless protocol. Thus, long polling may not need to save memory, making it much lighter and faster than web sockets. Likewise, as long polling data packets are lighter, it does not consume much of CPU. Thus, apart from the number of

requests, Web sockets and Long Polling share similar performance statistics. Based on these statistics, it could be concluded that web sockets do not have better performance overall than long polling. Long polling is the more appropriate choice for implementing this simulation program. However, this is the pilot study, so further study should be implemented to better understand the two differences.

6. PROBLEMS & STRATEGIES

Chrome Developer Tools is the primary tool for performance testing. However, it does not report the details such as each data frame, data frame response time, and the network traffic. Currently, Chrome Developer supports Web sockets API the most. Thus, for now, simulate a frame and manually measure the timing of the frame. Thus, it is likely that current statistics for Web sockets

implementation using Socket.io. could be erroneous and may need reinvestigation later.

Another problem is that the program is implemented using two different languages, PHP and JavaScript. There could be an indirect consequence due to the different languages. Because it becomes harder to distinguish between the effects caused by different real time technologies and the effects caused by different languages.

To solve these issues, better performance audit tools that can properly measure web socket's performance should be introduced and the program should be re-implemented in same programming language to see the direct differences between web sockets and long polling.

7. REFERENCES

- [1] Zhangling, Yan & Mao, Dai. (2012). A Real-Time Group Communication Architecture Based on WebSocket. International Journal of Computer and Communication Engineering. 408-411. 10.7763/IJCE.2012.V1.100.
- [2] Paul Murley, Zane Ma, Joshua Mason, Michael Bailey, and Amin Kharraz. 2021. WebSocket Adoption and the Landscape of the Real-Time Web. In Proceedings of the Web Conference 2021 (WWW '21). Association for Computing Machinery, New York, NY, USA, 1192–1203. <https://doi.org/10.1145/3442381.3450063>
- [3] Performance study of JavaScript WebSocket frameworks Prestandajämförelse av JavaScript WebSocket ramverk Jakob Hansson Supervisor : George Osipov Examiner : Ola Leifle
- [4] Ma K, Sun R. Introducing WebSocket-Based Real-Time Monitoring System for Remote Intelligent Buildings. International Journal of Distributed Sensor Networks. 2013;9(12). doi:10.1155/2013/867693
- [5] Performance comparison of XHR polling, Long polling, Server sent events and Web sockets

Gamification of Software Leads to Faster Learning

Ruth Ann Carlson

Winona State University

175 W Mark St

Winona MN, 55987

(507)-457-5000

ruthann.carlson@winona.edu

ABSTRACT

Turning learning into a game is a standard tool in the digital age, and it can be found in many different areas, such as workplaces, stores, and classrooms. This is called gamification, adding game-like elements to non-game activities to increase user enjoyment and engagement. The goal of this research was to understand the impact of gamification on learning speeds and to review the hypothesis that “learning methods that utilize gamification lead to an increased learning speed when compared to learning methods that do not include gamified elements.”

The study was conducted as a survey where participants were asked to learn through four given methods and then take a quiz after each learning method. Participants were asked to only spend ten minutes on each learning method. There were two non-gamified learning methods, the first was two short readings, and the other was a ten-minute video. The other two learning methods were gamified. These gamified learning methods were a matching game and a Jeopardy-style game. After the participant had completed all learning methods and their corresponding quiz, they took an exit survey that asked about their preferred learning methods and to disclose any prior knowledge they had on any of the topics.

This study got responses from nine participants and found that quiz scores were similar through all learning methods except for the reading, which had lower results. It was discovered that there was inconclusive evidence to state that learning methods with gamified elements resulted in faster learning than their non-gamified learning counterparts.

General Terms

Design, Experimentation

Keywords

Gamification, Learning Software

Proceedings of the 23rd Winona Computer Science Undergraduate Research Seminar, April 25, 2023, Winona, MN, U.S.

1. INTRODUCTION

As electronics become a more accessible resource for students, it is important that research is done to understand the best ways that these tools can be used in order to effectively aid in learning and understanding of topics. One of the new tools that educators are utilizing to support students in their education is gamification. Gamification is when game-like elements are incorporated with other activities. Gamification can be demonstrated in many different ways, such as through point tracking, a level system, competitive leaderboards, challenges and awards [1,3].

The current knowledge in the field mostly resides in non-educational fields, such as using gamification for motivating employees or for increasing revenue. It hasn't been until more recent years that researchers have begun to look into the impact of virtual gamification from an educational standpoint [5]. In addition to this, a great number of studies on gamified education tend to rely on students learning programming languages, and not other topics [2,6]. This means that a gamification strategy on education hasn't been thoroughly explored for topics outside the scope of computer science.

The purpose of this study is to research the impact that gamification has on learning, with an overarching goal to see if there is a correlation between a gamified approach and learning the material faster. In this study, survey participants will learn topics through traditional learning approaches and through a gamified approach. The participants will get the same amount of time to learn with each method, and will take quizzes in order to see how much information they have retained through the different methods. This information will be analyzed through comparing the different quiz scores.

Through this research, it will help future educators know the impact of gamification on education, and give them a better background for making decisions as for whether or not they want to use gamification strategies in their teaching approach. In addition to this, it will help shed light on how well gamification applies to learning in fields outside of computer science, as the learning done by the survey takers in this experiment will be in other fields.

2. HYPOTHESIS

This study aims to prove the hypothesis that the gamification of learning leads to a faster understanding of topics.

3. METHODS

This project is tested through a survey. This survey has the goal of teaching short, simple topics to the survey takers who will then take quizzes to demonstrate how much knowledge they retained from the lessons. How fast they learn the information will be based on their quiz scores after spending the same amount of time on each lesson.

The participants in this study were college students, mostly students pursuing a degree in Computer Science.

3.1 Learning Methods

For this survey there are four different learning methods which are being utilized. Two of these learning methods are from traditional learning approaches and the other two are through gamified learning approaches. All of the topics for these learning approaches are centered around niche information about animals. They're all about animals to keep the information similar for studying purposes, but different as learning the same information repeatedly would invalidate the study.

The first traditional learning approach is reading. The survey takers are given two short readings each about different frogs. These readings are from National Geographic and are informative about the different frog species.

The second traditional learning approach is a lecture. This lecture is ten minutes and forty two seconds. The lecture discusses various ways that animals have been accessorized in order to aid scientists. This video comes from a very popular youtube channel called SciShow which aims to inform their viewers about various topics in science. This channel was chosen because it was started by the famous brothers Hank and John Green who are also known for their very similar youtube channel crash course. Crash course is a channel that aims to teach people many subjects that are taught in school, mostly American high school, and are commonly used by educators as part of their curriculum [4].

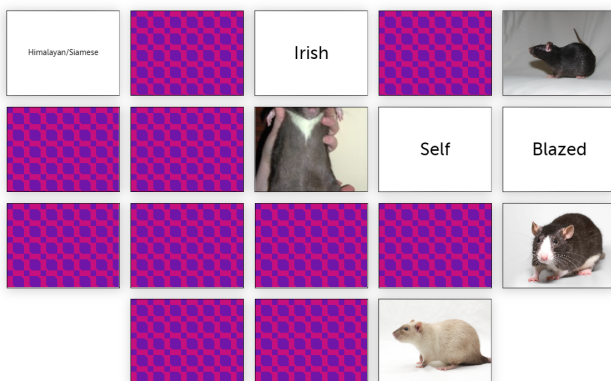


Figure 1. Matching game software

The first gamified learning approach, shown in figure 1, is a memory matching game created on the website puzzel.org. In this game, participants have to match the correct fancy rat coat type name to a picture of the coat type. When the play begins all of the cards are laid out with the terms and photos unable to be seen. The participant must flip over one card at a time and get a correct match. When two cards have been flipped over, if they match they remain facing up and if they do not match they face back down. The player has won the game when all the cards are facing up right as it means all of the matches have been correctly made.

Bears	Frogs	Seals	Raccoons	Whales
100	100	100	100	100
200	200	200	200	200
300	300	300	300	300
400	400	400	400	400
500	500	500	500	500

Figure 2. Jeopardy game software

The final gamified learning approach, shown in figure 2, is a virtual jeopardy game. This game is based on the popular tv show jeopardy and the game itself is on jeopardy.com. This game has five different subjects of questions, each with five questions of varying levels of difficulty. The topics are bears, frogs, seals, raccoons and whales. The questions are labeled with how difficult they are, with 100 being the easiest and 500 being the most difficult. When the user clicks on a number in a category they get a question about that category with the level of difficulty as indicated by the number.

3.2 Survey

This survey begins with a document that the participants are given. This document aims to give an overview of the entire survey for the participants to follow. The document begins with an introduction paragraph that briefly overviews the purpose of this study. The following four paragraphs in the study explain directions for the participant in each of the four learning methods. These directions tell the user to set a timer and only spend ten minutes on each learning activity except for the video lecture. These directions also tell the participant to take a quiz after each lesson. Each quiz asks five questions relevant to the subject matter. In addition to this, each quiz asks for a name or username for the participant, this is used to keep track of quiz scores for participants.

After the participants have completed the four lessons and quizzes, they are thanked for their time and given an exit survey. This survey asks them to rate how well they felt they could learn from each learning method on a scale of one to five, one being not well at all and five being the best way they learn. Participants are also asked for which learning method was their favorite, which was their least favorite, and if they had any prior knowledge for any of the learning topics. They are asked their thoughts on the learning methods to view how their feelings towards learning methods correspond to the grades they receive from learning

through different methods. They're also asked about their prior knowledge in order to consider possible ways prior knowledge could impact the results.

4. RESULTS

The survey conducted resulted in nine participants completing the survey. From these results, four participants stated that they had no prior knowledge of any information from any of the learning methods. Four participants stated that they had some prior knowledge on a few topics, but not much. And one participant stated that they had a decent amount of prior knowledge throughout the learning methods. None of these participants had their results disqualified as they all had rather limited amounts of knowledge and the participants who had more knowledge still resulted in similar quiz results to the participants who stated having no prior knowledge.

Through the results it is able to be analyzed how the different learning methods compare against each other. This is done through comparing the quiz scores of the participants and the participants' thoughts on different learning methods.

Through the quiz scores, three of the learning methods resulted in average quiz scores that were very similar, while one resulted in disproportionately low quiz scores. The learning method with the lowest average quiz score was learning method one, which was the readings. These had an average of 4.44 out of 7. This is much lower than the results of the other three learning methods, as seen in figure 3.

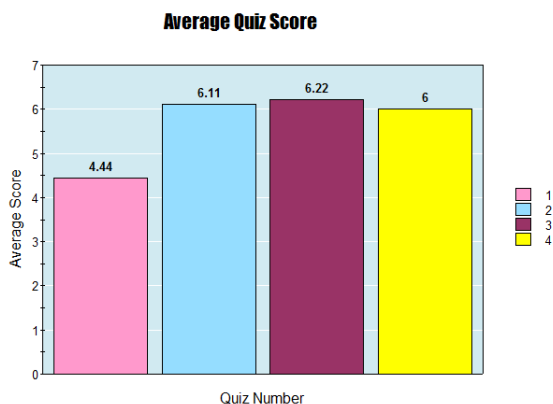


Figure 3. Average quiz results from survey takers. The column numbers correspond to the four different learning methods stated in the methods section.

When viewing the average quiz scores of survey takers, it is able to be noted how well they retained information from each of the learning methods after spending ten minutes learning with each method. Between the different learning methods, gamification methods tend to be similar or better than more traditional learning methods.

Similarly, participants were asked in the exit survey to rate each learning method on a scale of 0 to 5. Zero being they felt the learning method was not effective and five being they thought the learning method was very effective. These results are shown in figure 4.

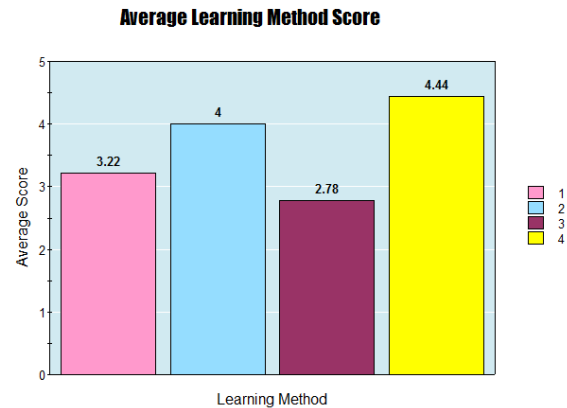


Figure 4. Average scores survey participants gave learning methods. The column numbers correspond to the four different learning methods stated in the methods section.

Demonstrated through figure 4, it is able to be known that participants tended to prefer the fourth learning method, which was a gamified learning approach through a game similar to Jeopardy. It is also shown that the participants tended to dislike learning method three the most, which was the other gamified learning method. Instead of being a Jeopardy game, this one was a matching game. This information has a heavy juxtaposition when compared to the quiz scores. This is because while the third learning method was rated by participants as the learning method they felt was least effective, this learning method also resulted in the highest average quiz scores of all of the learning methods.

Similarly these survey results display which learning methods were most liked and disliked. The most disliked learning methods were learning method one, reading, and learning method 3, the matching game. Four survey takers felt that the first learning method was the worst, and five survey takers felt that the third learning method was the worst. The reason users didn't like the matching game in learning method three was because they tended to find this learning method to be repetitive and time consuming. With other learning methods they only needed to go over the information once before taking the quiz. Yet for this learning method many participants felt they had to go through the game multiple times before remembering information and the repetition made it less enjoyable.

As for the learning method that the participants felt was their favorite, which is shown in figure 5, the majority of participants preferred the fourth learning method, which was the Jeopardy style game. This demonstrates that not only did participants find this learning method to be the most effective, despite the quiz results showing otherwise, but it also demonstrates that the

gamified learning method was a preferred way to learn. Participants stated they preferred this learning method because it aided them in identifying important information they should be remembering, it was more engaging and entertaining, and the participants tended to feel they better remembered the information through this method.

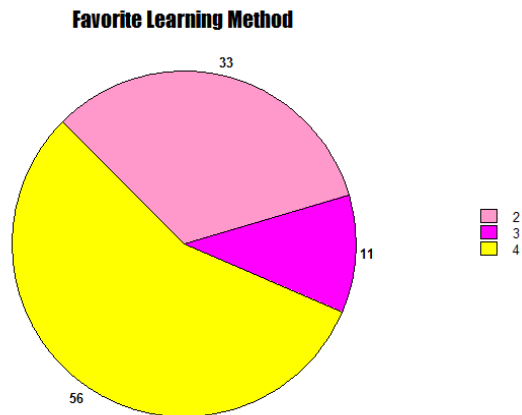


Figure 5. Survey participants vote on their favorite learning method. The section numbers correspond to the different learning methods stated in the methods section.

5. CONCLUSION

In conclusion, with the large scope of gamification, it is hard to pinpoint a direct answer as to how it impacts learning. This is because gamification is very broad and can be applied many different ways to many different forms of learning. In this survey, while a learning method utilizing gamification was found to be the most effective in a timed setting, it was not much more effective than some of the other learning methods. In addition to this, while the learning method resulted in high quiz results, the majority of participants did not enjoy the learning method and did not feel like they could effectively learn from it.

Yet, my hypothesis was supported to a degree. Both gamified learning methods resulted in higher quiz results than the non-gamified learning method, reading. Yet the learning method 2, the video, resulted in higher results than learning method 4, the Jeopardy game.

6. REFERENCES

- [1] A. P. U. of Gothenburg, A. Palmquist, U. of Gothenburg, and O. M. V. A. Metrics, “gamification was not the problem: Proceedings of the 24th international academic mindtrek conference,” ACM Other conferences, 01-Jun-2021. [Online]. Available: <https://dl.acm.org/doi/10.1145/3464327.3464347>. [Accessed: 23-Apr-2023].
- [2] Achilleas L. D. Buisman Radboud University Nijmegen, A. L. D. Buisman, R. U. Nijmegen, R. U. N. V. Profile, Marko C. J. D. van Eekelen Radboud University Nijmegen, Marko C. J. D. van Eekelen, V. University, and O. M. V. A. Metrics, “Gamification in educational software development: Proceedings of the Computer Science Education Research Conference,” ACM Other conferences, 01-Nov-2014.

[Online]. Available: <https://dl.acm.org/doi/10.1145/2691352.2691353>. [Accessed: 23-Apr-2023].

- [3] Gabriel Barata INESC-ID/IST/Technical University of Lisbon, G. Barata, I. N. E. S. C.-I. D. I. S. T. T. U. of Lisbon, Sandra Gama INESC-ID/IST/Technical University of Lisbon, S. Gama, Joaquim Jorge INESC-ID/IST/Technical University of Lisbon, J. Jorge, Daniel Gonçalves INESC-ID/IST/Technical University of Lisbon, D. Gonçalves, University of Ontario Institute of Technology, U. of Waterloo, and O. M. V. A. Metrics, “Improving participation and learning with gamification: Proceedings of the first international conference on gameful design, research, and applications,” ACM Other conferences, 01-Oct-2013. [Online]. Available: <https://dl.acm.org/doi/10.1145/2583008.2583010>. [Accessed: 23-Apr-2023].
- [4] H. Green and J. Green, “Who we are,” Hank and John. [Online]. Available: <https://www.hankandjohn.com/who-we-are/>. [Accessed: 23-Apr-2023].
- [5] Reza Hadi Mogavi Hong Kong University of Science and Technology, R. H. Mogavi, Hong Kong University of Science and Technology, Bingcan Guo Hong Kong University of Science and Technology, B. Guo, Yuanhao Zhang Hong Kong University of Science and Technology, Y. Zhang, Ehsan-Ul Haq Hong Kong University of Science and Technology, E.-U. Haq, Pan Hui Hong Kong University of Science and Technology & University of Helsinki, P. Hui, Hong Kong University of Science and Technology & University of Helsinki, Xiaojuan Ma Hong Kong University of Science and Technology, X. Ma, C. University, U. of Washington, N. Y. University, and O. M. V. A. Metrics, “When gamification spoils your learning: A qualitative case study of gamification misuse in a language-learning app: Proceedings of the Ninth ACM Conference on Learning @ Scale,” ACM Other conferences, 01-Jun-2022. [Online]. Available: <https://dl.acm.org/doi/10.1145/3491140.3528274>. [Accessed: 23-Apr-2023].
- [6] Ç. Ç. U. C. Dublin, Ç. Çubukçu, U. C. Dublin, B. W. U. C. Dublin, B. Wang, L. G. U. C. Dublin, L. Goodman, E. M. U. C. Dublin, E. Mangina, SUSTech, and O. M. V. A. Metrics, “Gamification for teaching java: Proceedings of the 10th EAI international conference on simulation tools and techniques,” ACM Other conferences, 01-Sep-2017. [Online]. Available: <https://dl.acm.org/doi/10.1145/3173519.3173538>. [Accessed: 23-Apr-2023].

7. APPENDIX

Survey:

https://docs.google.com/document/d/1un_vGOrx764-9LJqy822MwVPljRneRo2S1xH9E5UfgY/edit

Extreme Gradient Boosting and Adaptive Boosting to Predict Mortality of Patients in ICU

Sheikh Fahad
Winona State University
175 West Mark Street
Winona, Minnesota

sheikh.fahad@go.winona.edu

ABSTRACT

This paper presents a study that used electronic health record (EHR) data to predict the mortality of patients admitted to the ICU who are 40 years old or above. The study focused on comparing the performance of two popular machine learning algorithms, Adaptive Boost (AdaBoost) and eXtreme Gradient Boost (XGBoost), to predict the outcome of a binary classification problem. It used data collected from PhysioNet's MIMIC-IV dataset. Both algorithms were trained on the dataset and their performance was evaluated using accuracy, precision, F1-score, and AUC (area under the curve). The hyperparameters for both algorithms were tuned using a grid search cross-validation approach, and the best hyperparameters were selected based on the performance of the algorithm on the validation set. The findings suggest that the percentage of accuracy for XGBoost is at least 8% better than for AdaBoost, making it more effective and accurate in predicting mortality.

General Terms

Algorithms, Performance.

Keywords

Electronic Health Record (EHR), AdaBoost, XGBoost, Performance metric, Accuracy, Precision, Recall, F-1 score, Area under the curve.

1. INTRODUCTION

Electronic Health Record (EHR) data has been used to analyze and build machine learning models in the past decade. There have been several studies that have shown various models [1,2,3,4], to be beneficial in handling patients with various conditions in a hospital with EHR data. Some studies were based off how to help evaluate fall risk after emergency department visits [1], while other studies compared different machine learning algorithms and validated them [4]. Some studies predicted mortality while others did not. In this study, the focus was on two machine learning algorithms: XGBoost and AdaBoost. There have been various studies using XGBoost and AdaBoost [6,7,8], in the health sector. Previous studies and models built with EHR data to predict mortality Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

contained patient information outside the Intensive Care Unit (ICU) [6,8], and some studies have been done to predict mortality of patients inside the ICU [7]. This study aims to use information of patients who are 40 years old or older and admitted to the ICU.

The primary goal of this study is to build a model using eXtreme Gradient Boost (XGBoost) and Adaptive Boost (AdaBoost) on EHR data to predict mortality of patients who are 40 years or older admitted to the ICU. Specifically, the algorithm looks at the patient demographics to predict mortality.

The findings of this study can contribute to the understanding of the strengths and limitations of AdaBoost and XGBoost algorithms in predicting mortality with demographics information and EHR data to provide insights for future research in the field of machine learning. This would also be helpful to the doctors and nurses as they would be able to treat a patient with more focus based on their demographics to avoid mortality in the ICU.

2. Hypothesis

XGBoost will yield a higher accuracy of prediction of mortality than AdaBoost among patients older than 40 years old who are admitted to ICU based of demographics data.

3. Methodology

3.1 Data

The data was obtained through PhysioNet's MIMIC-IV data [5]. The data consists of tables from a relational database and hence the data required for the study needed to be extracted. The data file contains 4.2 million patient data. The study used the patients, admissions, and icustays data tables. From the patient's data table, the gender, anchor age, and date of death were selected as features. From the icustays data table, the first care unit and the last care unit were selected as features. From the admissions data table, the admission type, admission location, discharge location, insurance, language, marital status, and race were selected as features. Each of the features were cleaned and then joined to make a clean dataset. The dataset was then randomized to remove any continuity bias before the experimentation began.

To make the study go smoother and to be able to control features better, the study also used One-Hot Encoding technique. This technique allowed the research to go smoothly as all the columns and their data were turned into features and the data then just consisted of 0's or 1's.

Since the available compute power was limited, the study used only a subset of the data, which was approximately two hundred thousand data points. The subset was randomized through the usage of seeding while collecting the data. The data collected was of one

hundred thousand patients who perished and one hundred thousand patients who have not perished. Then the data was randomized again to avoid continuity between perished and non-perished patients. The data was separated in a 70-30 split for training and testing sets.

3.2 Algorithms

This study aims to compare the performance of two popular machine learning algorithms, Adaboost and XGBoost, in predicting the outcome of a binary classification problem.

Adaboost (Adaptive Boosting) is an iterative algorithm that creates a set of weak classifiers and combines them to form a strong classifier. It assigns weights to instances based on their difficulty to classify, so that instances that are misclassified in previous iterations are given more weight in the next iteration. This process is repeated until a satisfactory accuracy is achieved of about 75%.

XGBoost (eXtreme Gradient Boosting) is an optimized implementation of Gradient Boosting. It uses a decision tree-based model and implements several techniques to improve the speed and accuracy of the algorithm, such as regularization, parallel processing, and early stopping.

In this study, both Adaboost and XGBoost algorithms were trained on the same dataset and their performance were evaluated using common evaluation metrics such as accuracy, precision, recall, F1-score, and AUC (Area Under the Receiver Operating Characteristic Curve). The algorithms were trained using the scikit-learn and xgboost libraries in Python.

To ensure a fair comparison, the hyperparameters for both algorithms were tuned using a grid search cross-validation approach. The best hyperparameters were selected based on the performance of the algorithm on the validation set.

Finally, the accuracy, precision, recall, F1-score, and AUC of both algorithms were compared and discussed to determine which algorithm performs better for the given classification problem. The percentage of accuracy (and subsequently other performance metrics) for XGBoost model has the potential to perform at least 8% better than for AdaBoost model.

3.2.1 Pilot Study

The study was done initially with a smaller pilot study. The pilot study was done to check the code quality and to find bugs before starting the original study. The pilot study yielded the results shown in table 1.

Table 1. Comparison of model during pilot study

Metrics	AdaBoost	XGBoost
Accuracy	77.89	87.54
Precision	79.45	88.60
Recall	79.14	88.21
F-1 score	79.36	88.25
AUC	76.97	87.11

Table 1 shows the performance metrics of both the algorithms, AdaBoost and XGBoost. The performance metrics consisted of accuracy, precision, recall F-1 score and AUC. In the table it shows that AdaBoost underperformed XGBoost in all performance

metrics, but it did not satisfy the study since the metrics for XGBoost was expected to be higher.

After investigation, it was found that the dataset was heavily biased with 86% of the patients still alive and 14% of the patients deceased. The biased dataset was immediately fixed by randomly choosing the deceased patients 50% of the time and alive patients the other 50% of the time. This made sure that the dataset had a 50-50 split between the patients living or deceased in the ICU.

3.3 Steps Taken

The steps for this research were taken procedurally. The steps and explanations of the steps are described below:

1. Data retrieval from PhysioNet – a certification needed to be done to be able to handle confidential patient data.
2. Data collection – data was collected from 3 tables (patients, admissions, icustays) provided from the PhysioNet data. A subset of the data was collected which was about two hundred thousand patient information.
3. Data cleaning – data was cleaned and joined through the subject id of each patient. One hot encoding was done after that to make each column a feature for the algorithm.
4. Data split – 70% of the data was randomly split from the main table for training and 30% was for testing.
5. Pilot study – A pilot study was done to test the integrity of the data and the code written for training and testing. The pilot study showed flaws in the dataset as well as some minor bugs in the code which needed to be fixed.
6. Grid Search Cross Validation – Another subset of about one hundred thousand datapoints were taken for Grid Search Cross Validation to find the best parameters for fair comparison between the models.
7. AdaBoost – AdaBoost was implemented through the scikit-learn library in python. Grid Search Cross Validation hyperparameters were used. Average performance metrics were collected after five simulations and ten simulations. Performance metrics collected were accuracy, precision, recall, F1-score, and AUC (Area Under the Receiver Operating Characteristic Curve) scores.
8. XGBoost - XGBoost was implemented through the xgboost library in python. Grid Search Cross Validation hyperparameters were used. Average performance metrics were collected after five simulations and ten simulations. Performance metrics collected were accuracy, precision, recall, F1-score, and AUC (Area Under the Receiver Operating Characteristic Curve) scores.

4. Results

After the pilot study was completed, a random subset of 50000 data was selected from the new unbiased dataset to perform the Grid Search Cross Validation to find the best parameters. The best parameters were chosen for each model to ensure a fair comparison between the models. The final parameters used in the training of both models were:

AdaBoost: iterations = 2000 and learning rate = 1.0

XGBoost: max depth = 8, iterations = 500, subsample = 0.33, learning rate = 0.5, and early stopping rounds = 15

Both the models performed as per expectation. AdaBoost underperformed XGBoost in every performance metric. AdaBoost's iteration took a substantial amount of time (about 20 minutes) to run one iteration. The result from the model was as expected, below 85%. XGBoost's iterations took less time than AdaBoost (about 12-15 minutes). This boosting algorithm is industry standard and performed much higher than AdaBoost, as expected, with scores above 90%.

5. Analysis

When collecting the data, each of the models were simulated a total of ten times. The average of the model scores were done by hand, by taking the total scores and dividing by 5 after five runs and dividing by 10 after ten runs.

Table 2. Comparison of model after five simulations

Metrics	AdaBoost	XGBoost
Accuracy	76.09	94.59
Precision	77.87	93.62
Recall	83.28	93.92
F-1 score	80.48	93.77
AUC	81.47	94.06

Table 3. Comparison of model after ten simulations

Metrics	AdaBoost	XGBoost
Accuracy	79.01	95.94
Precision	80.29	96.54
Recall	83.39	95.29
F-1 score	81.81	95.91
AUC	83.06	96.11

As seen in tables 2 and 3, both the tables are looking at the performance metrics (accuracy, precision, recall, F-1 score, and AUC) for both the algorithms. In table 2 the difference in accuracy is larger than in table 3 for both the algorithms. Looking at all the other performance metrics, only the accuracy difference is reduced between the tables. All the other performance metrics from table 2 and table 3 had an increase in difference. Also, table 3 had higher values reported than table 2, which confirms that more simulations gave better results.

Regardless of that, both tables show that the XGBoost model scores were higher than the AdaBoost model. AdaBoost underperformed in all performance metrics.

It was important to check the receiver operating characteristic (ROC) curve for a better understanding of the predictions made by the algorithms. An ROC curve is a graph showing the performance of a classification model. This curve plots two parameters: True Positive Rate vs. False Positive Rate. Figures 1 and 2 show the ROC curve of AdaBoost and XGBoost, respectively.

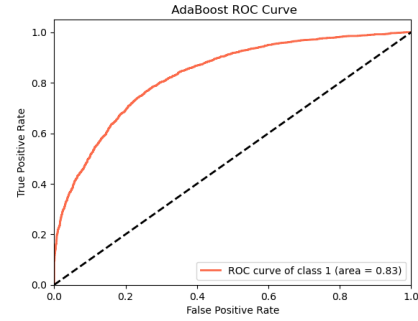


Figure 1. ROC Curve for the AdaBoost Algorithm

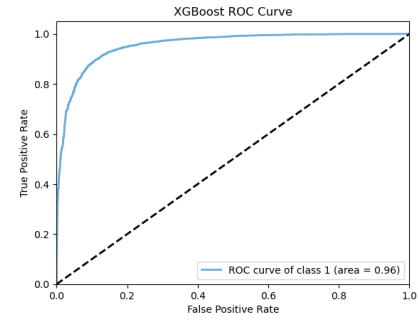


Figure 2. ROC Curve for the XGBoost Algorithm

The black dotted line in both figures is considered as the halfway line between predicting and guessing. If a curve is close to the dotted line, it is considered that the algorithm is guessing its predictions. If the curve is below the dotted line, then the algorithm is not predicting correctly, in other words, the predictions are mostly false positives. If the curve is above the dotted line, then the algorithm is predicting well. The steepness of the curve determines how well the algorithm is predicting.

In figure 1, it shows that the bend of the curve is less steep and close to the dotted line. Compared to the AdaBoost curve, the XGBoost curve, in figure 2, is much steeper and further away from the dotted line. XGBoost predicted better and had less false positive predictions than AdaBoost.

This confirmed the hypothesis of the study that XGBoost indeed yields higher accuracy of patient mortality in ICU who are 40-year-old and above than AdaBoost.

6. Conclusion

The study was done on PhysioNet's MIMIC-IV data. A subset of two hundred thousand patient records were used from 4.2 million patient information. AdaBoost and XGBoost were used to model data to predict mortality of patients in the ICU who are 40 years old or above. The hypothesis of the study was that XGBoost will yield higher accuracy of prediction than AdaBoost. Based off the research and findings of this study, it can be concluded that XGBoost does indeed produce higher accuracy of prediction for patient mortality in ICU who are 40 years old or above.

One limitation of the study was that it was done on demographics data and there are many other factors that were not considered. Where a person was born, immigration status, gender at birth, and so on were not taken into account while making predictions. In the

future, if a more complete dataset were to be built, a better prediction could be done.

Another limitation is that the study did not test the statistical significance of the difference in metrics. Future studies can be done to check whether the difference in metrics really impacts the algorithms capability to predict mortality of patients in the ICU.

Also, further study could be done to find out why XGBoost outperforms AdaBoost in all performance metrics. In addition, it would be beneficial to know if we remove the highest contributing feature, would the models perform the same, and what the primary cause of mortality is using just the ICU information without the demographics data.

7. ACKNOWLEDGMENTS

I would like to give a special thanks to Dr. Collin Engstrom for his help with the research. Dr. Engstrom's immense knowledge of machine learning and bioinformatics background provided a lot of value to this research. Also, I would like to thank Dr. Mingrui Zhang and Dr. Sudharsan Iyengar for reviewing and correcting the papers I wrote. I would also like to thank Dr. Gerald Cichanowski for the immense amount of knowledge he has given me over the last few years. Lastly, I want to thank the faculty and staff members of Computer Science, Mathematics & Statistics, and Physics departments, who have taught me so much and without them I would not be able to be here.

8. REFERENCES

- [1] Patterson BW, Engstrom CJ, Sah V, Smith MA, Mendonça EA, Pulia MS, Repplinger MD, Hamedani AG, Page D, Shah MN. Training and Interpreting Machine Learning Algorithms to Evaluate Fall Risk After Emergency Department Visits. *Med Care.* 2019 Jul;57(7):560-566. doi: 10.1097/MLR.0000000000001140. PMID: 31157707; PMCID: PMC6590914.
- [2] Engstrom CJ, Adelaide S, Liao F, Jacobsohn GC, Patterson BW. Operationalizing a real-time scoring model to predict fall risk among older adults in the emergency department. *Front Digit Health.* 2022 Oct 31;4:958663. doi: 10.3389/fdgth.2022.958663. PMID: 36405416; PMCID: PMC9671211.
- [3] Cowling TE, Cromwell DA, Bellot A, Sharples LD, van der Meulen J. Logistic regression and machine learning predicted patient mortality from large sets of diagnosis codes comparably. *J Clin Epidemiol.* 2021 May;133:43-52. doi: 10.1016/j.jclinepi.2020.12.018. Epub 2021 Jan 22. PMID: 33359319.
- [4] Faisal M, Scally A, Howes R, Beatson K, Richardson D, Mohammed MA. A comparison of logistic regression models with alternative machine learning methods to predict the risk of in-hospital mortality in emergency medical admissions via external validation. *Health Informatics J.* 2020 Mar;26(1):34-44. doi: 10.1177/1460458218813600. Epub 2018 Nov 29. PMID: 30488755.
- [5] Johnson, Alistair, et al. "MIMIC-IV" (version 2.2). *PhysioNet* (2023), <https://doi.org/10.13026/6mm1-ek67>.
- [6] Vaid A, et al. Machine Learning to Predict Mortality and Critical Events in a Cohort of Patients With COVID-19 in New York City: Model Development and Validation, *J Med Internet Res* 2020;22(11): e24018. <https://www.jmir.org/2020/11/e24018>. doi: 10.2196/24018
- [7] Choi, M.H., Kim, D., Choi, E.J. et al. Mortality prediction of patients in intensive care units using machine learning algorithms based on electronic health records. *Sci Rep* 12, 7180 (2022). <https://doi.org/10.1038/s41598-022-11226-4>
- [8] Cong Li, et al. Machine learning based early mortality prediction in the emergency department, *International Journal of Medical Informatics*, Volume 155, 2021, 104570, ISSN 1386-5056, <https://doi.org/10.1016/j.ijmedinf.2021.104570>. (<https://www.sciencedirect.com/science/article/pii/S1386505621001969>)

Gamification of Recipe-Viewing Software Leads to an Enhanced User Experience

Justin Fuller

Winona State University

175 W Mark St

Winona, MN 55987

justin.fuller@go.winona.edu

ABSTRACT

This paper explores the effect of gamification in a recipe-viewing application and its impact on user experience. Gamification incorporates game-style incentives into non-game activities. Two identical recipe applications were created, with one of them incorporating a game while the other does not. In these pieces of recipe software, the user can choose a recipe from four different categories: breakfast, lunch, dinner, and dessert. For each recipe the user selected, the user was prompted with the instructions, ingredients, and a youtube video on how to make the recipe. The gamified application has a game with a point system in which the user can unlock random recipes corresponding to the category of food they are in. The basic application is the same but without the game.

To assess user experience, the users were given an 11 question survey to express their feelings about certain parts of the application. The questions ranged from rating their overall experience using both applications, how effective the implementation of gamification was, the usefulness of the game, and how much they liked the game's design and implementation on a scale from one to five.

The results show that the gamified application provided users with a significantly better user experience. The gamified application had a higher overall experience rating, was preferred over the basic application, and was more effective at keeping the users interest than the basic application.

General Terms

Design

Keywords

Gamification, User Experience

Proceedings of the 23rd Winona Computer Science Undergraduate Research Seminar, April 25, 2023, Winona, MN, US

1. INTRODUCTION

Gamification incorporates game-style incentives into everyday or non-game activities. The process of gamifying products has been around since the late 1800s but, in recent years, has gained widespread popularity due to the rising increase in technology [6]. From professors using gamification to improve learning and engagement [1] to developers using gamification to boost a healthy lifestyle [3], gamification is here to stay.

This project gamified a recipe software that aims to make cooking meals at home more gratifying. Two recipe applications were created, and both were laid out similarly in the sense that recipes were divided into separate categories. When the user clicks on a category, they are given a list of possible recipes that correspond with their chosen category. After choosing a recipe, the user is led to a page that gives all of the necessary information to cook the recipe they chose, and at this point, the two software will be different.

The gamified software includes a point system in which users earn points based on the meals they create at home. After clicking on a recipe, they are rewarded with one point. When they reach 3 points, the game rewards them with a random recipe related to the category that they are currently in. For example, if the user gets 3 points in the "Dessert" category, they will be rewarded with a random Dessert recipe.

The objective of this research project was to survey how gamification affects the user's experience with the application that was made. This project aimed to learn the effects of gamification and how it affects user experience by receiving feedback on two different web applications, one gamified and one non-gamified.

This research was essential to the field of Computer Science and Gamification because as attention spans get lower and lower every year [4], Web Developers and Software Developers will need to find more engaging ways to keep the user on their websites for more extended periods. After conducting this study, gamification will be proven effective for these developers to provide a more engaging experience.

2. HYPOTHESIS

Software's gamification allows for a more engaging user experience through higher overall satisfaction rating, higher rates of gauging interest, and a higher preferred rate.

3. METHODOLOGY

To conduct this research, two basic web applications were created using HTML, CSS, and JavaScript, but one was 'gamified,' and the other was not. The user experienced two similar applications, but one application had a point system game element added to it. After the user experienced both web applications, they were given a short survey that asked them about how they liked the website, how they would rate their overall experience out of 5, what they liked and disliked about each application, and any other feedback the user would like to give.

In the survey, the user was prompted with 11 total questions; one question asked for the user's name if they wanted to provide it, four questions asked the user to rate their feelings about certain parts of the application (rating their overall experience, how effective the game aspect was, if they liked how the game was designed, the usefulness of the game) on a scale from 1-5. The users were then asked which version of the application they preferred, why they preferred it, and what made it better than the other version. Although mostly tailored to the applications' video game aspect, these questions can give a good and meaningful result as to whether or not adding the video game portion really enhanced the user experience. This survey was heavily influenced by Nova Eka Diana and Ocky Aditia Saputra's research on measuring user experience of a potential shipment tracking application. [2]

3.1 Screenshots of Software

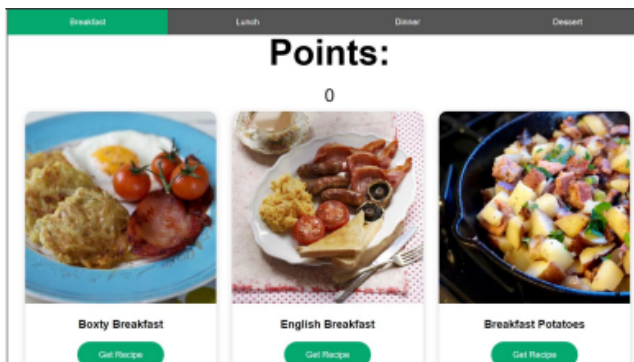


Figure 1. Gamified Software with point system

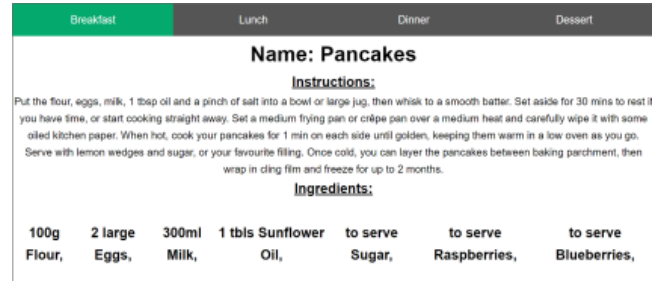


Figure 2. Example of Random Recipe after gaining 3 points

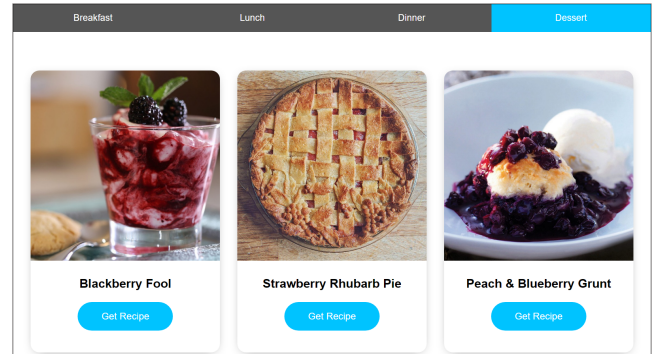


Figure 3. Non-gamified software without the point system

Figure 1 includes a picture of the gamified software with the point system. When you click on the green "Get Recipe" button underneath each food item, you are rewarded with one point. When you get 3 points, you are rewarded with a random recipe (Figure 2).

Figure 2 includes a picture of what the gamified software will look like when you gain 3 points. In the random recipe, it shows the name of the random recipe, the instructions of the recipe, and all the ingredients needed to cook that recipe.

Figure 3 includes a picture of the non-gamified software. The non-gamified software is laid out identically to the gamified software except the non-gamified software does not have the point system.

4. RESULTS

After conducting this experiment, the Gamified application had higher overall ratings in satisfaction (Figure 4), was preferred more than the non-gamified application, and had higher rates of gauging the user's interest than the non-gamified application (Figure 5).

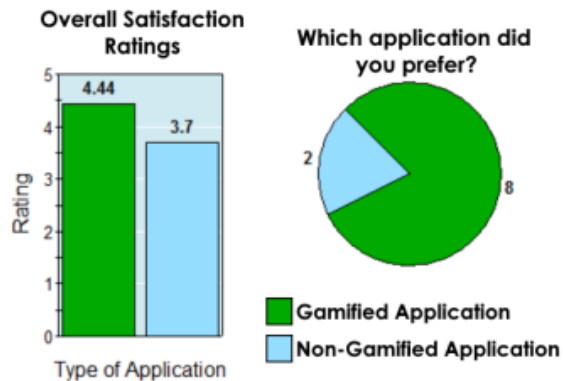


Figure 4. Overall Satisfaction Ratings and Preferred Application Results

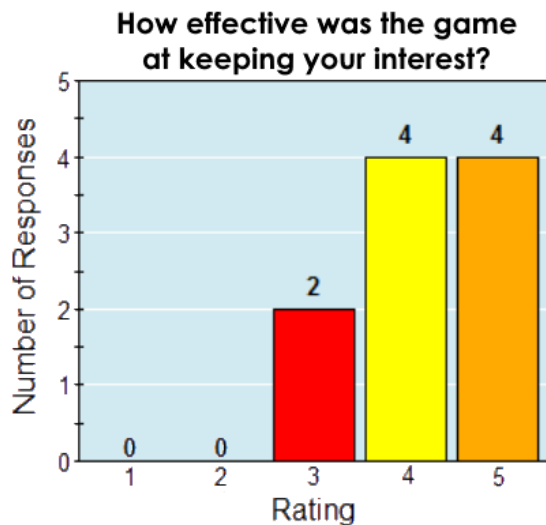


Figure 5. Effectiveness of the game at gauging interest results

Figure 4 shows two graphs. The graph on the left shows the Overall Satisfaction ratings on a scale from one to five. The users, on average, were more satisfied with using the gamified application than the non-gamified application. Overall satisfaction ratings were crucial to measure because if the users were not satisfied with how an application was, they would most likely want to use the application less. The graph on the right of Figure 4 shows which application the user preferred. Eight of the ten people surveyed said they preferred to use the application with the point system, which means that the implementation of the game helped boost the overall user experience.

Figure 5 shows the survey results when the users were asked, "How effective was the game at keeping your interest?" on a scale from one to five. The results show that, on average, the game made the users want to use the gamified software more. This question was crucial to this research because although the implementation of the game was basic, the game still made the

users want to use the gamified software more, which proves the hypothesis.

5. CONCLUSION

In conclusion, the hypothesis of whether or not gamifying software will improve the user experience was proven true. Data was collected from 10 users that took the survey. With the results of the survey, I was able to conclude that users preferred the gamified application more than the basic application.

There can be many extensions of this project if time allows. Ideally, the project would include a much more sophisticated and engaging video game added to it to make it a lot more engaging, such as a point leaderboard with which one can compete against their friends, a better-designed UI, better ingredient descriptions, better instructions, and the list goes on. Another great application choice for this type of research would be an application that could benefit people's health, such as a fitness or calorie tracking application.

In terms of the pool of users being surveyed, the next logical step for anyone wanting to continue this research would be to survey a broader range of people than Winona State students. A more extensive and diverse pool of participants would have solidified the evidence and strengthened the hypothesis even more because the only people with access to this research study are those with an internet connection and a working computer, so it will exclude people that do not have access to them. Unfortunately, for this project, this was a barrier that could not get solved due to time constraints.

6. ACKNOWLEDGEMENTS

I would like to thank my advisors, Dr. Mingrui Zhang, and Dr. Sudharsan Iyengar for guiding me through the research process and helping me with this project.

7. REFERENCES

- [1] Barata, G., Gama, S., Jorge, J., & Gonçalves, D. (2013). Improving participation and learning with gamification. *Proceedings of the First International Conference on Gameful Design, Research, and Applications*. <https://doi.org/10.1145/2583008.2583010>
- [2] Nova Eka Diana and Ocky Aditia Saputra. 2015. Measuring user experience of a potential shipment tracking application. In *Proceedings of the International HCI and UX Conference in Indonesia (CHIuXiD '15)*. Association for Computing Machinery, New York, NY, USA, 47–51. <https://doi-org.wsuproxy.mnpals.net/10.1145/2742032.2742032>

- [3] Oliver Vicente, Sandra Vicente, Daniel Martin, Miguel Angel Rodriguez-Florido, and Manuel Maynar. 2014. Health gamification. In *Proceedings of the 2014 Summer Simulation Multiconference (SummerSim '14)*. Society for Computer Simulation International, San Diego, CA, USA, Article 57, 1–7.
- [4] Phys.org. (2022, February 16). Are attention spans really collapsing? data shows UK public is worried, but also see technology benefits. Phys.org. Retrieved January 18, 2023, from <https://phys.org/news/2022-02-attention-spans-collapsing-uk-technology.html>
- [5] Sepehr, S., & Head, M. (2013). Competition as an element of gamification for learning. *Proceedings of the First International Conference on Gameful Design, Research, and Applications*. <https://doi.org/10.1145/2583008.2583009>
- [6] Shannon, J. (n.d.). *The history of gamification - journey from 1896 to the 21st Century*. Make a Video Game in Minutes Not Months. Retrieved January 18, 2023, from <https://www.gamify.com/gamification-blog/the-history-of-gamification>
- [7] Hadi Mogavi, R., Haq, E.-U., Gujar, S., Hui, P., & Ma, X. (2022). More gamification is not always better. *Proceedings of the ACM on Human-Computer Interaction*, 6(CSCW2), 1–32. <https://doi.org/10.1145/3555553>

Investigating Ease-of-Use in GUIs vs. CLIs Based on User Feedback

Matthew Jilk

Winona State University | B.S. in Computer Science

175 W Mark Street

Winona MN, United States

1 952-479-8752

matthewjilk@gmail.com

ABSTRACT

The Human-Computer Interaction subfield known as User Experience (UX) is a highly complex and notoriously difficult-to-study area of software design. One reason for this difficulty as well as the lack of concrete data on the subject is that researchers often look at examples of high-quality User Interfaces (UIs) and then subjectively explain why they think they're high-quality. We sought to solve this problem by conducting a more scientific and pragmatic study comparing ease-of-use in different UIs that offer identical functionality for the same piece of software. Our results were twofold: CLIs are difficult for most people to use and polls intended to measure UX should have detailed questions *as specific* as the competing design choices and justifications for choosing one over another.

Winona State University | CS 495 Computer Science Research Seminar

Measurement, Documentation, Design, Economics,
Experimentation, Human Factors

UI, UX, screen recording, heatmaps, graphics, golden rules,
subjective reporting vs. objective measurements, minimalism vs.
ease-of-use, aesthetics vs. utility

1. INTRODUCTION

UX occupies the intersection of UI design and human psychology. The quality of a user's experience depends on the quality of the User Interface (UI). A piece of software's UI is the auditory and visual space where the interactions between the user and the application occur. The terms "application", "program", and "website" will be used interchangeably throughout this paper since the same principles apply to each. The terms "programmer" and "developer" will also be used interchangeably. The term "element" will be used in this paper to refer to any abstractable component of a user interface beyond the simplest level containing things like single characters in a word, one pixel in a rectangle, etc.) Examples of elements include buttons, sliders, icons, headings, text input boxes, dialog boxes, and menus, which are commonly called "widgets".

UI design encompasses everything from practical considerations like the resolution of the typical user's screen and expected peripherals to more abstract and artistic aspects such as color theory and the "weight" of different sections of the interface.

Developers design based on how they expect users to interact with their software in theory, but the software's UX depends on how exactly the user interacts emotionally and cognitively with their hardware to start up, control, modify, view, reference, navigate, close, etc. the software in practice.

Designing UIs conducive to good UXs is of obvious importance since poorly designed UIs waste time, frustrate users, and may drive users away if they are aware of sufficiently advanced alternatives. Ineffective UX has efficiency, user satisfaction, and business performance implications ranging from not-worth-considering to death and suffering.

Although the experiment we conducted only looked at three versions of a webpage, the principles discovered and summarized apply to apps written for smartphones, laptops, desktops, smart watches, smart fridges, one-time-setup IoT devices, industrial control panels, etc. Our main discovery which might be more surprising to programmers than the general population is that GUIs often offer subjectively and objectively offer more effective UXs than CLIs for most users, at least according to our survey and timing data, respectively.

Well-designed UIs and well-anticipated UXs effectively serve as the junction between the software's functionality and the user's desire to take advantage of it by being intuitive (i.e., it functions how most users expect it to upon first impressions), efficient (i.e., common actions can be done with a minimal number of clicks/keypresses), and aesthetically pleasing to the greatest degree possible within the other more utility-focused constraints.

Generally accepted, well-justified design principles include consistency of the software, shortcuts for the convenience of frequent users, information feedback, usage of keyboard-navigable dialog boxes, recoverability, customization, and general minimalism.

The first consideration when designing a UI is whether it will be a command-line interface (CLI) or a graphical user interface (GUI). Most modern pieces of software used by the general public have a GUI, yet there is a commonly-held view among programmers that keyboard-navigable UIs such as CLIs offer superior UXs. This assumption is the main subject of the experiment described in this paper which involved pitting multiple UIs for the same website, including multiple GUIs and a CLI, against each other. These three simple UIs, referred to

throughout this paper as “Version 1”, “Version 2”, and “Version 3”, offer the same functionality but look and behave differently. Each version consisted of only a single webpage. Version 1, a GUI, encapsulated the inputs for each type of search into its tab, Version 2, another GUI, had all inputs visible at all times but enabled and disabled two or then depending on the state of a toggle switch, and Version 3, the CLI, had only a single text input field where a command is entered, and a “Submit” button.

Due to the unavoidably biased and subjective nature of single-researcher research on UX, all of the principles in the “Additional Principles” and potentially some in the “Seven Golden Rules” section will not apply to some or most typical users of a given piece of software.

2. HYPOTHESIS

The CLI version will be significantly slower and less preferred than the GUI versions.

3. PRINCIPLES OF EFFECTIVE UI DESIGN

Before explaining the specifics of the methodology and analyzing the results of the experiment, it will be useful to understand some general principles of good UI design. This may seem unnecessary for the more “common sense” principles, but what seems obvious to one developer may not be so clear to another, leading to disagreement and conflict. If a common sense principle under debate is truly a good principle, then it helps to be able to articulate why, and if it’s not, then attempting to justify it will reveal its issue(s).

One researcher[1] has proposed eight “golden rules” for UI design which have been condensed here to seven. These rules are described below with the names of the rules quoted verbatim in italics. All principles from the beginning of “Additional Recommendations” onwards were not present in this researcher’s paper, arguably due to having been too specific and/or already covered within the eight golden ones.

3.1 Seven golden rules

3.1.1 Consistency of the software

One of the broadest foundational principles is that UIs ought to be internally and externally consistent. Internal consistency refers to consistency across different screens, menus, views, windows, etc. within the application.

External consistency refers to consistency across different pieces of software on a given OS, browser, type of computer, input type, screen size, or platform. Given a particular new piece of software to develop, developers must have an awareness of and do research on its target user base and other programs they might be familiar with since, in addition to being a more worthwhile time investment, it also gives developers a non-arbitrary escape from the impossibility of making it consistent with all other software on the target platform. Some platforms have official guidelines for their UIs such as the “macOS design philosophy”[13] which recommends many of the broader principles described in this

paper such as simplicity, clarity, cleanliness, minimalism, and focus on the user[800].

If the software is cross-platform and the norms of two or more platforms conflict, the developer is faced with a dilemma. One way to resolve it is to make a choice that adheres to neither of the norms, thereby forcing new neural connections to be imprinted in the user’s brain that they will eventually instinctively rely on instead of relying on existing ones which they might habitually default to when they shouldn’t. Although this may cause minor frustrations during the initial learning stages, it will prevent frustrations that could arise as users switch between platforms, which some users will do frequently for the entire time they use your software on one or more of them.

3.1.2 Shortcuts for the convenience of frequent users

CLIs are entirely keyboard-based. Users *can* copy-paste text from the shell window, but CLIs weren’t designed for this. Non-CLI UI’s that take advantage of the keyboard beyond just typing out words, numbers, and arrow keys can make for highly smooth, intuitive, and effective UXs.

Both mouse-based and keyboard-based shortcuts should be present. Every mouse-based shortcut, whether in a menu, toolbar, floating window, or dialog box should have a corresponding keyboard shortcut that does the exact same thing, and the list of all possible actions of length f sorted by usage frequency for a typical user should be mapped directly to first f shortcuts in the list of all possible keyboard shortcuts sorted by the number of keys with the shortcuts with the fewest number of keys mapped to the most common actions with the exception of “risky” actions that can’t be undone such as deleting something, submitting a form, sending an email, etc., which should require at least two keys to be pressed simultaneously to prevent the accidental execution of the action.

Another important property of frequently used keybindings is that they should be easy to remember by having the key match the first letter of the action’s name. In Photoshop, for example, the keyboard shortcut for “clone stamp brush” is “c”, “marquee tool” is “m”, “brush” is “b”, and so on[3]. Designers following this convention will quickly realize the impossibility of adhering to it for all the keybindings (or even just all frequently used ones), but that doesn’t mean developers need to resort to selecting random keys. They can instead choose a key with a character that’s at the middle or end of the actions name (such as Photoshop’s “Move” being to “v”) or a character/symbol that visually resembles the action (such as Studio One’s “Add Crossfade” being bound to “x”[4]).

Having easy-to-remember keybindings results in the user taking advantage of more of them more often which leads to a more efficient and less painful UX as using the keyboard is faster and results in less hand/wrist strain than relying on the mouse[5].

3.1.3 Feedback information

When something goes wrong, the user needs to be notified of it. Users also need to be notified when an action finishes successfully. This is often done through dialog boxes that show some kind of red “x” for errors or a green check mark for non-errors. Although other colors could be used, the meaning of an icon with a red or green color is instantly clear to most users since most countries use these colors in stoplights where red means something is wrong and the action of pressing the brake pedal is required and green means nothing is wrong and no action is required. In the context of software, an error occurring typically requires one or more actions to resolve, whereas something executing successfully merely requires one “Okay” button to be clicked or nothing to be done at all.

3.1.4 Designing dialog boxes to show the status

Dialog boxes have a set of behavioral properties familiar to most users such as the inability to click on any UI elements except the dialog box, which is useful when something demands immediate attention because interacting with non-dialog-box elements before resolving it would cause problems. These properties lend naturally to useful visual/aesthetic properties such as allowing the software to present novel information in an attention-demanding manner without making any drastic, unpleasant changes to the normal layout such as transitioning to a completely different “screen” of the UI with different menus, content in the main section taking up most of the screen, and so on. Having an always-visible space for this information (for example, in a menu or toolbar) would both fail to grab the user’s attention and use the screen space efficiently.

3.1.5 Action reversal- undo

The undo action may be the single most universally recognized action across all GUIs. Unless a developer has good reason to do otherwise, this action should be bound to the Control + z for Windows and Linux and Command + z on Mac where the “+” sign represents the keys to the left and right of it needing to be pressed down simultaneously. The shortcut for the redo action should be identical except for the addition of a single modifier key such as Shift or Alt, requiring minimal hand movement to undo and redo. The Control + y shortcut used in many Windows applications is awkward and requires a relatively drastic repositioning of one or more of the user’s hands.

3.1.6 Design in a way that the user is in control of the software

For example, offer the ability to customize keyboard shortcuts as well as what options appear in the menu(s) on the software’s main screen to give advanced users the ability to optimize their workflow to whatever degree needed or desired. The Safari web browser, to take a specific example, gives users the ability to fully customize what buttons appear in the main toolbar on the top of the screen, including the ability to add and remove any of the possible buttons[2].

3.1.7 Minimize instructions and information to make it easy for the user to remember things.

Only a frequently used subset of the currently relevant elements should be visible and enabled. The rest should be hidden away in fixed-position or floating menus, and users should not be required to remember too many pieces of information at once. Software should be easy to use for new users who haven’t read unreasonably large portions of the documentation. For simple-to-medium-complexity-software intended for a large, wide, general audience, developers should assume that the typical user will start using the software without reading any of the documentation, so their UI should be as intuitive and minimalistic as possible. Ideally, all software would be completely self-explanatory and not require any documentation or tutorial to be read, watched, listened to, or interactively completed within the actual application/website. Due to their limited screen space, mobile phone UIs commonly have a button to show/hide a list of these options consisting of a few horizontal bars. The decrease in visual clutter and confusion for new users outweighs the small amount of extra time it takes to click something within a dropdown menu (or something similar).

3.2 Additional Principles

3.2.1 Awareness of user intelligence

To design an optimal UI for a population, developers must know what the population is. A piece of software’s typical user may have certain relevant characteristics that should inform certain UI design choices such as disabilities, the software they might be familiar with, usage environment, age, intelligence, etc. A common mistake developers make is to design UIs ideal for themselves with no consideration for what their users want or need. This is problematic because developers are not representative of the average population in several ways including intelligence and computer skills. The average IQ of a typical programmer could be between 1[7] and 1.5[8] standard deviations[9] above average for the general population, meaning that a UI with the optimal UX for programmers may not be the UI with the optimal UX for the user.

Smoother, more effective UXs require simpler UIs with fewer visual elements on any given screen/menu/view at a given time, especially for less intelligent users, but developers should prioritize simplicity and minimalism regardless of the target user’s intelligence since even highly intelligent users would prefer to use a smaller portion of an already relatively small portion of their mental effort to use an application assuming no loss of functionality or efficiency.

3.2.2 Awareness of user skill and experience

Furthermore, programmers will always have more of a certain class of computer skills and experience than non-programmers due to the nature of their profession. Included and potentially most relevant among these are those directly related to using the software they develop. Most programmers who spend days implementing a feature will test that feature many times by using it as the user would, and in the most extreme case where one programmer develops all of an application’s features, they will be

an expert in using that application by the time it's finished. But becoming proficient in an application's entire feature set by developing it makes it impossible for the developers to have a true first-time-user experience since users probably become proficient in some subset of the application's feature set in a much shorter time than it took to develop it. For this reason, people who weren't involved in the creation of a piece of software should be consulted regarding its UX. It is for these reasons that developers must maintain an ongoing awareness of who is using their UIs and how they are using them.

3.2.3 Usage of vector as opposed to bitmap graphics

Either vector graphics or low-detail, low-complexity, low-information bitmap images should be used for menu bar items, sidebars, buttons, logos, etc. This means having graphics with a large radius for circular shapes such as circles, Bézier curves, and other curves, and sharp, zero-pixel transitions between colors so that graphics will look the same at any viewing distance and users won't spend too much time looking at the details. To make the transitions between colors more appealing, anti-aliasing should be used, so the transitions between colors might be a few pixels rather than zero.

The opposite of this “keep detail low” principle in the worst-case-scenario might involve having a one-pixel by one-pixel medium grey rectangle within an 80-pixel by 40-pixel medium-light gray button or an 80-pixel by 1-pixel line within an 80-pixel by 80-pixel button. While serifs can aid in reading speed due to the higher level of detail making it easier to know what a letter is by only looking at it for a very small fraction of a second, users only need to use an element a few times before memorizing its exact location, at which point they no longer need to read its text, so developers looking for a more modern look and feel should prefer sans-serif fonts since their lower level of detail gives them a cleaner look and feel and fewer detail for elements viewed elements results in less visual clutter. For example, the buttons and options in the top bar (also known as the “Ribbon”) in Microsoft Word[15] (which use a sans-serif font) are rarely used since the user spends most of their time looking at the text they are writing.

4. METHODOLOGY

The following steps were taken in the order listed to set up and execute the UX experiment. The website's name “YTCL” will be used throughout the remainder of this text.

4.1 Programming the software

The common YTCL backend for searching/filtering YouTube videos and comments was programmed, which crucially included code to record the user's browser window while they were on the website. This was done since knowledge of being recorded might interfere with how naturally and calmly they use it if they are, for example, nervous about making errors and performing poorly.

3 distinct versions of the single-webpage UIs were created and made publicly available at <http://ytcl.xyz/v.html> where v is either “1”, “2”, or “3” respective to each version.

4.1.1 UI explanations, notes, and screenshots

The calendar icon to the right of the “mm/dd/yyyy” text pops up a date selection widget.

The video or comment results, depending on what the user inputs into the form fields, are displayed below what is visible on the screenshots inside another blue background box with an equally horizontally sized, identically colored, and identically positioned and colored header using the same font as the first box's header. It wouldn't necessarily be “wrong” for the bottom background box to have a different color in this case because it contains a different class of elements: the top contains inputs while the bottom contains outputs, although this approach of color-coding different elements by class makes more sense when there are at least two discrete elements in each class.

Version 1 organizes and encapsulates the elements into tabs. Some elements are duplicated in ways that could be avoided like they are in Version 2 and Version 3 such as the “Start Date” and “End Date” date inputs[11] and their corresponding label.

Figure 1. Version 1

Version 2 displays all form inputs at all times but the “Video URL” text box, is disabled by default. The toggle switch flips between enabling this and the “Channel Name” text box.

Figure 2. Version 2

Version 3 is the CLI version. “Text Patterns” allow users to exclude videos that don’t match a regular expression. This term was used since most non-developers are probably not familiar with regular expressions as a term or concept, the latter of which served as reason for including a “help” hyperlink to a document with a half-page explanation of what “Text Patterns” are, some simple examples, and why they are useful.

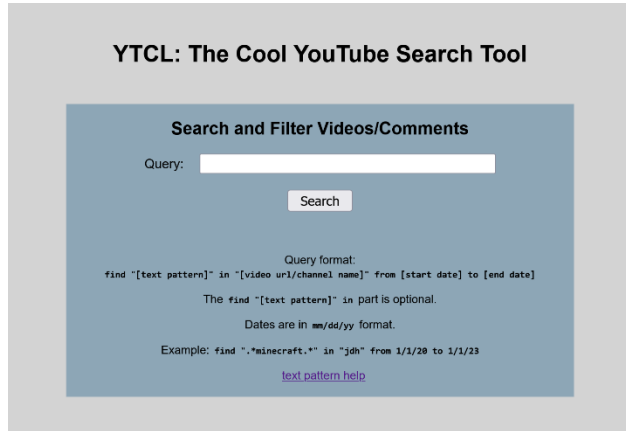


Figure 3. Version 3

4.2 Having test subjects use it

A simple poll with these hyperlinks and the questions “Which version of this website do you prefer?” and “Which version do you least prefer” were created. Flyers containing QR code hyperlinks to this survey were distributed throughout the Winona State University campus on bulletin boards and the surrounding Winona residential area on randomly selected residents’ doorsteps.

4.3 Analyzing how users used it

The screen recordings were then quantified by measuring the time it took between the user entering the webpage to clicking the “submit” button and subtracting loading time for non-invalid queries.

5. RESULTS

5.1 Quantitative Data

The reason 83 respondents voted on the “Which version do you most prefer?” question while only 23 voted on the “Which do you least prefer?” one is because the latter was added to the survey after some respondents had already completed the one-question version. Mouseflow[16], the API used to record users’ screens, has an additional feature of creating a heatmap showing aggregates of user mouse positions (in other words, the total sum of time the cursors of each user spent at each pixel), which was used to more qualitatively analyze YTCL’s UX.

Table 1. Times taken to execute search queries

Version	Average time to perform search (seconds)	Median time to perform search (seconds)
1	29.8	25.0
2	13.5	12.8
3	51.1	42.7

1	29.8	25.0
2	13.5	12.8
3	51.1	42.7

The “timer” started when the page that would submit the successful query (submitting a query reloads the webpage) loads and ends when the user clicks “submit”.

Table 2. Most and least preferred versions

Version	% of respondents who preferred it most (80 respondents)	% of respondents who preferred it the least (23 respondents)
1	40	20
2	40	28
3	20	52

Table 3. Percentage of Queries That Were Invalid Queries

	Version 1	Version 2	Version 3
% of queries that were invalid	23	29	61

Queries in Versions 1 and 2 can be invalid if the channel name or video URL the user enters doesn’t exist or the end date comes after the start date.

5.2 Qualitative Observations

5.2.1 Version 1

Many users repeatedly clicked between two adjacent tabs very quickly after already having clicked on every tab at least once, presumably because they found doing so viscerally satisfying.

5.2.2 Version 2

There is a large red area on the mouse position heatmap over the toggle button showing that users were confused for a relatively substantial time as compared to v1 which made it instantly clear how it works for the vast majority of users.

5.2.3 Version 3

Users spent a relatively substantial amount of time reading the instructions before typing anything

5.2.4 General Notes

Most respondents preferred the CLI version the least, corroborating the common view that people prefer simpler and more intuitive UIs to ones that require reading and memorizing instructions when both offer identical features and functionality. People don’t enjoy reading instructions before gaining the ability to do what they want to do. UIs that are self-explanatory have an easier and faster learning curve and also result in less frustration since they disallow users from entering invalid input before they submit their query. Version 2 and 3’s usage of date input widgets makes it impossible to enter invalid dates such as May 35th and years such as those in the future and makes it obvious to the user

exactly the argument type and number of arguments required whereas the CLI version only informs the user of invalid, impossible, or improperly formatted dates after they click “Submit” which reloads the page and deletes the entire command (just as common CLIs such as OS shell applications do), further increasing frustration. It could have the common command line shell feature of using the arrow keys to select previous commands, but this still requires navigating back to the text cursor back to the problematic section(s), which is less convenient than simply not being allowed to click type an invalid date.

Having a greater capacity to submit erroneous queries resulted in more erroneous queries being submitted. The GUI versions, which disallow some percentage of all possible invalid inputs *before* clicking “Submit” as opposed to the CLI version which disallows zero percent, saw 72 percent fewer invalid queries submitted. An optimal UIs would have disallowed 100 percent.

Even excluding the time spent entering invalid queries, the GUIs saw superior query execution times (Table 1). Version 1 and 2’s average times are around 50 percent longer than Version 3 for both the median and mean times.

Keyboard-based UIs have superior ergonomics to mouse-based UIs. And while proponents of CLIs may cite the superior ergonomics[5] of never using the mouse such as substantially mitigated risk of repetitive strain injury, it’s possible to entirely avoid the issue of wrist strain and decreased speed and efficiency incurred from mouse usage by simply making the entire GUI accessible via the keyboard, as it was for YTCL by pressing the Tab key to focus the next input field and Enter key to submit the query. Furthermore, designers should aim to also make UIs as close to 100% accessible using only the mouse as is practical since it may be faster or more convenient for the user to keep their hand on the mouse if they only need to input a few clicks sandwiched between much longer periods of mouse input such as navigating websites. The awareness of where the user’s hands should be and more importantly where they probably will be for the typical user is crucial to crafting optimal UXs.

Despite the duplicate elements and longer query execution times in Version 1, respondents preferred it equally to Version 2, suggesting that mathematical properties such as absolute element count minimalism don’t necessarily lead to an effective UX. Version 2 took nearly twice long to execute successful queries than Version 1, but the aesthetic appeal of Version 1’s tab selector bar or the frustrating confusion of Version 2’s toggle switch may have been responsible for the users “misreporting” which version was most effective for them. The semi-superfluosity of aesthetic appeal demonstrates the importance of not relying too heavily on user feedback derived from polls with unspecific questions. If feedback is to be used to make a potentially important design decision, the question(s) should be *as specific* as the design choice being made and can include brief summaries of justifications for doing it one way over another.

6. LIMITATIONS

6.1 Suboptimal experiment

Although our experiment confirmed already known principle(s) of effective UI design, no serious conclusions can be drawn from it alone: the questions were unspecific, the sample size was small, the UIs were simple, and the CLI variant bore almost no resemblance to common CLIs which don’t run in a web browser, display the command syntax documentation directly beneath the command line, or allow users to specify optional arguments via command prefixes, and most users probably didn’t use it for long enough to accurately gauge which made for the objectively or subjectively best UX.

6.2 UI Version Inconsistency

One limitation of the specific way that these three versions were designed is that, although they offer the same functionality, they don’t all offer *exactly* the same functionality: Version 2 and 3 offer the same functionality, but Version 1 lacks the ability to specify both a date range and a regular expression to filter by. This may have made users less inclined to select Version 1, although it’s highly likely that most didn’t realize this since most of them didn’t use the “Text Pattern” feature.

7. CONCLUSION

7.1 GUIs vs. CLIs

As software becomes increasingly advanced to the point where several high-quality, well-designed, and often free applications exist for any category of application for a given purpose, the sometimes significant but sometimes not-worth-considering UX aspects such as awareness of the user, consistency, shortcuts for the convenience of frequent users, feedback, dialog boxes, action reversal, user control and customizability, minimalism, vector graphics, and an awareness of user demographics are becoming increasingly essential to set one application apart from another. As one researcher succinctly stated, “A system will always be judged by its users not by its functionality, but by its interface design”[2]. In nearly all of these aspects, CLIs are inferior to GUIs and should thus be avoided most of the time. By default, they offer less helpful feedback, more visual clutter and room for error, action reversal, dialog boxes, or shortcuts. While it was not particularly important for YTCL to have a maximally good UX for its typical user, this isn’t always the case: consequences of bad UIs and bad UXs range from wasted time to negative perceptions of a company to mass hysteria[12] to death[14].

Most new programs should only be CLI-only if they’re intended for programmer or non-human use. A program intended for human use being CLI only, especially if intended for use by most of the general population, must be rationalized by certain extraneous or extenuating characteristics of its typical user, development circumstance, or purpose.

7.2 Diminishing Returns on advanced feature investments

The economic law of diminishing returns [17] states that the vast majority of a technology’s utility comes from its most basic

functionality and has implications across all fields of design with UI being no exception: there's little point in reserving always-visible screen space for rarely used elements. This principle is also known as the "80-20 Rule"[10], stating that 80 percent of users only use 20 percent of a piece of software's features. Although every conceivable advanced feature *could* be thought of, designed, implemented, and hidden away in an "Advanced Features" menu, doing so is not feasible or possible within nearly all practical, actual, or desirable time, business, effort, human, or success constraints.

8. REFERENCES

- [1] Ravi Chandra Chaitanya Guntupalli. University West. User interface design – methods and qualities of a good user interface (2008). Retrieved from <http://hv.diva-portal.org/smash/get/diva2:215020/FULLTEXT01.pdf>.
- [2] Apple Support. Customize toolbars on Mac. Retrieved from <https://support.apple.com/guide/mac-help/customize-toolbars-mchlb294260e/mac>.
- [3] Photoshop Support. Default keyboard shortcuts (2022). Retrieved from <https://helpx.adobe.com/photoshop/using/default-keyboard-shortcuts.html>.
- [4] Knowledge Base | Presonus. Studio One 3: Editing. Retrieved from <https://support.presonus.com/hc/en-us/articles/210050333-Studio-One-3-Editing>.
- [5] Proceedings of the Human Factors and Ergonomics Society Annual Meeting. Keyboard Shortcut users: They are faster at more than just typing (2009). Retrieved from https://www.researchgate.net/publication/249955485_Keyboard_Shortcut_Users_They_Are_Faster_at_More_than_Just_Typing.
- [6] Marc Hassenzahl and Noam Tractinsky. Taylor & Francis. User experience - a research agenda. Retrieved from <https://www.tandfonline.com/doi/abs/10.1080/014492905000330331>.
- [7] Robert M. Hauser. University of Wisconsin-Madison Center for Demography and Ecology - Gwern. Retrieved from <https://gwern.net/doc/iq/ses/2002-hauser.pdf>.
- [8] Statistic Brain. IQ estimates by college major. Retrieved from <https://www.statisticbrain.com/iq-estimates-by-intended-college-major>.
- [9] Linda S. Gottfredson. APA PsycNet. Logical fallacies used to dismiss the evidence on intelligence testing (2009). Retrieved from <https://psycnet.apa.org/record/2008-17662-001>.
- [10] Kelly-Bootle, ACM Digital Library. A persistent rule of thumb in the programming trade (2006). Retrieved from dl.acm.org.
<https://dl.acm.org/doi/pdf/10.1145/1122674.1122696>.
- [11] Mozilla Developer Network Docs. HTML: HyperText Markup Language (2023). Retrieved from <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input/date>
- [12] Amy B. Want. The Washington Post. Hawaii missile alert: How one employee 'pushed the wrong button' and caused a wave of panic (2018). Retrieved from <https://www.washingtonpost.com/news/post-nation/wp/2018/01/14/hawaii-missile-alert-how-one-employee-pushed-the-wrong-button-and-caused-a-wave-of-panic>.
- [13] Apple Developer. Designing for macOS. <https://developer.apple.com/design/human-interface-guidelines/platforms/designing-for-macos/>.
- [14] Nancy Leveson and Clark S. Turner. IEEE. An Investigation of Therac-25 Accidents (1993). Retrieved from https://www.cse.msu.edu/~cse470/Public/Handouts/Therac/Therac_1.html.
- [15] Microsoft 365 Support. Customize the ribbon in Word. Microsoft Support. Retrieved from <https://support.microsoft.com/en-us/office/customize-the-ribbon-in-word-c4b6051a-7a70-43c8-a527-932917dec682>.
- [16] Mouseflow. Behavior analytics for optimal website UX (2023). Retrieved from <https://mouseflow.com>.
- [17] Encyclopædia Britannica. Diminishing Returns. Retrieved from <https://www.britannica.com/money/diminishing-returns>.

Investigating the Comprehension of GPT's Interpretation of Word Meanings

Longge Yuan
Winona State University
Department of Computer Science
Winona, Minnesota
Longge.yuan@go.winona.edu

ABSTRACT

ChatGPT is a large-scale language model developed by OpenAI. Its purpose is to start a conversation with people and provide them with information ranging from simple facts to more complex topics. Trained on a large amount of text data, ChatGPT has shown us that it can understand human language and respond in a meaningful way. In this paper, we show a way to judge whether ChatGPT can accurately understand the various meanings of words and use them based on the output of the answer by ChatGPT. The five dimensions, each scored up to five points, are word explanation, etymology, synonyms and antonyms, sentence construction, and explanation of words in sentences. The results of the testing are Word Explanation, Etymology, Sentence Structure, and Interpretation all scored 4 or above. Then, the tests show lower scores for synonyms and antonyms with an average score of 1.86. The average overall score is 20.44 in 25. Although ChatGPT is slightly inferior in synonyms and antonyms, the results show that ChatGPT understands the meaning of words well.

1. INTRODUCTION

With the further maturity of artificial intelligence technology and the continuous increase of government and industry investment, the application of artificial intelligence is developing rapidly. The popular ChatGPT was born in recent years. In recent years, the development of natural language processing can be described as advancing by leaps and bounds [1]. Since ancient times, human writing has usually been assisted and supported by other tools, from traditional tools such as pen and paper to computers and software. With the advent of various artificial bits of intelligence such as Google Translate. Humans are increasingly using artificial intelligence to help them with writing tasks. Therefore, as a popular product, ChatGPT is very important to be able to explain the differences in the way of text well.

Proceedings of the 23rd Winona Computer Science Undergraduate Research Seminar, April 25, 2023, Winona, MN, US.

2. Keyword

OpenAI, ChatGPT, Artificial Intelligence, Large Language Model

3. BACKGROUND

ChatGPT is an AI-autonomous chat website created by OpenAI. The model it uses is the language model GPT created by OpenAI itself, Generative Pre-Training (<https://en.wikipedia.org/wiki/GPT-3>). GPT is based on the Transformer architecture. The GPT model first performs unsupervised pre-training on a large-scale corpus and then performs fine-tuning for specific tasks on a much smaller supervised data set. First, train a general model, and then adjust it for each task. This kind of model design technique that does not rely on individual tasks can achieve good performance in multiple tasks at once [2]. These large batches allow increasing amounts of computers to be efficiently poured into the training of a single model and are an important enabler of the fast growth in AI training computers.

3.1 CHATGPT

ChatGPT relies on this large-scale learning model, which makes it much ahead of other applications. ChatGPT was created by training a GPT-3.5 series model through reinforcement learning from human feedback (RLHF) [2]. ChatGPT is a conversational AI model that is based on the GPT architecture, which stands for "Generative Pre-trained Transformer." It is a type of deep learning model that uses an unsupervised learning method called pre-training to learn how to generate human-like responses to text input. "This model has been trained on a massive amount of data, allowing it to understand and generate human-like text with remarkable accuracy [6]."

Pre-training is a technique where the model is trained on large amounts of data, such as books, articles, and websites, to learn how language is used and to capture the patterns and structures of human language. This is done by training the model to predict the next word in a sentence, given the previous words. The goal is to create a language model that can generate text that is similar to human-generated text. Once the model has been pre-trained, it can be fine-tuned on

a specific task, such as conversational AI. Fine-tuning involves training the model on a smaller dataset that is specific to the task and adjusting the weights of the pre-trained model to fit the new data.

4. METHODOLOGY

4.1 Hypothesis/Question

As a chat tool, ChatGPT is extremely important to understand people's conversations. Sentences are made up of individual words. Many of these words do not have unique meanings. A word can have many meanings. Being able to understand the meaning of each word in different contexts is to prove that ChatGPT is an advanced AI performance. Thus, this research focuses on whether ChatGPT AI understands and explains the meaning of words.

4.2 Methods

Inspired by [1], I evaluate whether ChatGPT can understand the words. Considering this factor, I collected some comparative words, which contain polysemy and ambiguity. I believe such comparative terms can be a valuable and interesting source for studying the nature of human language and language models. Compare the answer from the ChatGPT with Oxford dictionary [3], A Complete Dictionary of Synonyms, Antonyms or Synonyms and Words of Opposite Meaning [4] and A Concise Etymological Dictionary of the English Language [5].

There are 5 judging dimensions. Each dimension has 5 points. Total of 25 points

4.2.1 (5 points) Evaluate the meaning of the words given by the AI response.

Compare the given word with its meaning in the dictionary. Test ChatGPT with some polysemous words. Then score it based on the answers given by ChatGPT. Finding 5 key words of the dictionary in the answer, each key word has 1 point.

4.2.2 (5 points) Evaluate find the etymology.

Compare ChatGPT's answer with given words' etymology in dictionary or paper. If ChatGPT can find the etymology. 1 point for determining the etymology. 1 point for correct etymology. Finding 3 key words of the dictionary in the answer, each key word has 1 point.

4.2.3 (5 points) Whether to answer synonyms and antonyms for given words.

Ask ChatGPT to give 5 synonyms and antonyms for each word. Compare the synonyms and antonyms of the given word with its synonyms and antonyms in the Oxford Thesaurus dictionary. Each synonym has 0.5 points. Total of 2.5 points. Each antonym has 0.5 points. Total of 2.5 points.

4.2.4 (5 points) Whether to make 5 sentences with the given words.

Use ChatGPT to make 5 sentences for each word. For each sentence, 0.5 points for sentences conforming to English grammar, 0.5 points for correct use of words in a sentence. 5 sentences, total of 5 points.

4.2.5 (5 points) Whether to interpret words in context.

Input 5 sentences containing the given word and let ChatGPT interpret the given word in context. 1 point for explaining the correct meaning of a word in context. Total of 5 points.

In this research, choosing 15 common words, 5 rare words and 5 professional words for ChatGPT to answer. Then score according to the 5 judging schemes listed. After scoring, make a five-dimensional diagram for average points. According to the scores of the five-dimensional diagram, analyze and summarize them. Draw conclusions about the credibility of ChatGPT for word interpretations.

4.3 Word List

Table 1 Table of words

	Meaning	Etymology	Synonyms /Antonyms	Meaning in Text	Sentence
Aim	5	5	2.5	5	5
Key	5	5	0	5	5
Blue	4	4	1.5	5	5
Border	5	5	5	5	5
Can	4	5	0	5	5
Cast	5	5	5	5	5
Character	5	5	5	5	5
Power	5	5	4.5	5	5
Last	5	5	4	5	5
Tear	5	5	2	5	5
Spare	5	5	2	5	5
Recommend	4	5	2	5	5
Strip	5	5	2.5	5	5
Mole	4	5	0	5	5
Way	5	5	0	5	5
Draconian	5	5	2.5	5	5
Convivial	5	5	1.5	5	5
Egregious	2.5	5	0.5	5	5
Imperious	2.5	5	1.5	5	5
Visceral	2.5	5	0	5	5

Architecture	5	5	1.5	5	5
Cluster	5	5	2	5	5
Compile	5	5	2	5	5
Distribute	3	5	1	5	5
Dynamic	4	5	0.5	5	5

4.4 Example of Question:

4.4.1 Question

I will give you a word. Please answer the following questions.

1. Answer all the meaning of the given word.
2. find the etymology.
3. Give me five synonyms and five antonyms for the given word.
4. make 5 sentences with the given words.

4.4.2 Aim

Question: Aim

Answer: Sure! Here are the answers to your questions about the word "aim":

Meanings of "aim":

To direct one's efforts or attention towards a particular goal or objective.

To intend or try something towards a particular target or direction.

The goal or purpose that one is trying to achieve.

The direction or trajectory towards which something is aimed.

Etymology of "aim":

The word "aim" comes from the Old French word "esmer", which meant "to estimate or guess". It is derived from the Latin word "aestimare", which means "to value or estimate".

Synonyms and antonyms of "aim":

Synonyms: target, focus, direct, point, aspire, strive, endeavor, goal, objective, purpose, intention. Antonyms: distract, deviate, wander, miss, veer, neglect, abandon, give up, disregard, ignore.

Sentences using "aim":

My aim is to become a successful writer.

He aimed his rifle at the target and pulled the trigger.

The company aims to reduce its carbon emissions by 50% within the next decade.

Her aim was off and she missed the bullseye.

The coach encouraged the team to aim high and strive for excellence.

Question: I will give you 5 sentences, please answer me the meaning of "aim" in each sentence:

1. My aim is perfect!
2. He was free, he had nothing but his aim to consider, and he reached it.
3. She took careful aim and squeezed off a shot.

4. He grasped a musket and took aim at the French.
5. In the long run men hit only what they aim at.

Answer: Sure, here are the meanings of "aim" in each of the sentences you provided:

My aim is perfect! In this sentence, "aim" refers to a person's ability to accurately hit a target or achieve a desired outcome. It suggests that the speaker is very skilled and precise in their actions.

He was free, he had nothing but his aim to consider, and he reached it. In this sentence, "aim" refers to a person's goal or objective. It suggests that the person in question had a clear sense of purpose and was able to achieve their goal despite any obstacles.

She took careful aim and squeezed off a shot. In this sentence, "aim" refers to the act of pointing a weapon or object towards a specific target. It suggests that the person in question was very deliberate and precise in their actions.

He grasped a musket and took aim at the French. In this sentence, "aim" again refers to the act of pointing a weapon or object towards a specific target. It suggests that the person in question was preparing to engage in combat and was taking careful aim at an enemy target.

In the long run men hit only what they aim at. In this sentence, "aim" refers to a person's goal or objective. It suggests that people are more likely to achieve their desired outcome if they have a clear sense of purpose and direction.

4.4.3 Grading rules example of word "Aim":

Evaluate the meaning of the words given by the AI response:

In Oxford dictionary, the meanings of word "Aim" are:

Aim —v. 1 intend or try; attempt (aim at winning; aim to win). 2 (usu. Foll. By at) direct or point (a weapon, remark, etc.). 3 take aim. —n. 1 purpose or object. 2 the directing of a weapon etc. At an object. take aim direct a weapon etc. At a target. [latin aestimare reckon]

The ChatGPT's answers are:

To direct one's efforts or attention towards a particular goal or objective.

To intend or try something towards a particular target or direction.

The goal or purpose that one is trying to achieve.

The direction or trajectory towards which something is aimed.

Evaluation: In definition as a verb, ChatGPT's answer includes "intend or try" and "direct" key words. In definition as a noun, ChatGPT's answer includes "object" and "target" key words. Thus, the score is 5 points because the answer includes all the key words of "Aim".

Find the etymology.

Answer from A Concise Etymological Dictionary of the English Language:

Aim, to endeavour after. (F. -L.) M.E. semen. From confusion of (i) A.F. esmer, from L.astimare. to estimate. Aim at, intend; and (2) O.F. aesmer, from L. ad-astimare, comp. with prefix ad-, to. See Esteem.

The ChatGPT's answers is:

Etymology of "aim": The word "aim" comes from the Old French word "esmer", which meant "to estimate or guess". It is derived from the Latin word "aestimare", which means "to value or estimate".

Evaluation: The ChatGPT's answer includes key words" esmer", "aestimare" and "Latin word". Thus, the score is 5 points because the answer includes all the key words of "Aim".

Give me five synonyms and five antonyms for the given word:

Answer from A Complete Dictionary of Synonyms and Antonyms:

KEY: Aim \n.\

SYN: Tendency, intent, aspiration, bent, drift, object, scope, goal, purpose, mark, end, design, intention.

ANT: Shunning, disregarding, disaffecting, overlooking, avoiding.

KEY: Aim \v.\

SYN: Seek, level, propose, design, affect, intend, mean, purpose.

ANT: Shun, disregard, disaffect, ignore, overlook, avoid.

ChatGPT's answer are:

Synonyms and antonyms of "aim": Synonyms: target, focus, direct, point, aspire, strive, endeavor, goal, objective, purpose, intention.

Antonyms: distract, deviate, wander, miss, veer, neglect, abandon, give up, disregard, ignore.

Evaluation: The ChatGPT's answer of synonyms include key words" objective", "purpose", "goal" and "intention". In antonyms include words" disregard". Thus, the score is 2.5 points because the answer includes 5 key words of "Aim".

Make 5 sentences with the given words.

ChatGPT's answers:

My aim is to become a successful writer.

He aimed his rifle at the target and pulled the trigger.

The company aims to reduce its carbon emissions by 50% within the next decade.

Her aim was off and she missed the bullseye.

The coach encouraged the team to aim high and strive for excellence.

Evaluation: In that case, every sentence conforms to English grammar, and the meaning of the words in the sentence is used correctly. Thus, the score is 5 points.

Whether to interpret words in context.

The answers about each sentence:

My aim is perfect!

He was free, he had nothing but his aim to consider, and he reached it.

She took careful aim and squeezed off a shot.

He grasped a musket and took aim at the French.

In the long run men hit only what they aim at.

The ChatGPT's answers:

My aim is perfect! In this sentence, "aim" refers to a person's ability to accurately hit a target or achieve a desired outcome. It suggests that the speaker is very skilled and precise in their actions.

He was free, he had nothing but his aim to consider, and he reached it. In this sentence, "aim" refers to a person's goal or objective. It suggests that the person in question had a clear sense of purpose and was able to achieve their goal despite any obstacles.

She took careful aim and squeezed off a shot. In this sentence, "aim" refers to the act of pointing a weapon or object towards a specific target. It suggests that the person in question was very deliberate and precise in their actions.

He grasped a musket and took aim at the French. In this sentence, "aim" again refers to the act of pointing a weapon or object towards a specific target. It suggests that the person in question was preparing to engage in combat and was taking careful aim at an enemy target.

In the long run men hit only what they aim at. In this sentence, "aim" refers to a person's goal or objective. It suggests that people are more likely to achieve their desired outcome if they have a clear sense of purpose and direction.

Evaluation:

First sentence, "Aim" means target or goal.

Second sentence, "Aim" means target or goal.

Third sentence, "Aim" means the directing of a weapon.

Fourth sentence, "Aim" means the directing of weapon to a target.

Fifth sentence, "Aim" means the goal or objective.

In that case, ChatGPT correctly interprets the word "aim" in all sentences. Thus, the score is 5 points.

4.5 Five-Dimensional Diagram

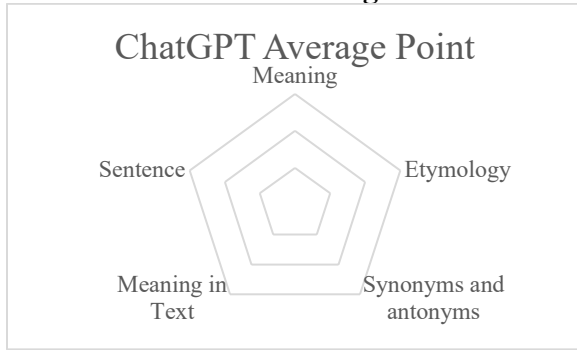


Figure 1 Example of 5 dimensions graphic

The five-dimensional diagram is created by Meaning, Etymology, Synonyms and antonyms, Text, Sentence.

4.5.1 Example of diagram:

Meaning corresponds to the judgment **dimension 1** --- Evaluate the meaning of the words given by the AI response.

Etymology corresponds to the judgment **dimension 2** --- Evaluate find the etymology.

Synonyms and antonyms correspond to the judgment **dimension 3** --- Whether to answer synonyms and antonyms for given words.

Meaning in Text corresponds to judgment **dimension 4** --- Whether to interpret words in context.

Sentences correspond to judgment **dimension 5** --- Whether to make 5 sentences with the given words.

5. RESULTS AND ANALYSIS

Table 2 Table of the All words

	Meaning	Etymology	Synonyms /Antonyms	Meaning in Text	Sentence
Aim	5	5	2.5	5	5
Key	5	5	0	5	5
Blue	4	4	1.5	5	5
Border	5	5	5	5	5
Can	4	5	0	5	5
Cast	5	5	5	5	5
Character	5	5	5	5	5
Power	5	5	4.5	5	5
Last	5	5	4	5	5
Tear	5	5	2	5	5
Spare	5	5	2	5	5
Recommend	4	5	2	5	5

Strip	5	5	2.5	5	5
Mole	4	5	0	5	5
Way	5	5	0	5	5
Draconian	5	5	2.5	5	5
Convivial	5	5	1.5	5	5
Egregious	2.5	5	0.5	5	5
Imperious	2.5	5	1.5	5	5
Visceral	2.5	5	0	5	5
Architecture	5	5	1.5	5	5
cluster	5	5	2	5	5
Compile	5	5	2	5	5
Distribute	3	5	1	5	5
Dynamic	4	5	0.5	5	5

Table 3 Table of Score of Total words

	Meaning	Etymology	Synonyms and antonyms	Meaning in Text	Sentence	Total
Average	4.22	4.76	1.86	4.8	4.8	20.44

Table 4 Table of Common Words

	Meaning	Etymology	Synonyms /Antonyms	Meaning in Text	Sentence
Aim	5	5	2.5	5	5
Key	5	5	0	5	5
Blue	4	4	1.5	5	5
Border	5	5	5	5	5
Can	4	5	0	5	5
Cast	5	5	5	5	5
Character	5	5	5	5	5
Power	5	5	4.5	5	5
Last	5	5	4	5	5
Tear	5	5	2	5	5
Spare	5	5	2	5	5
Recommend	4	5	2	5	5
Strip	5	5	2.5	5	5
Mole	4	5	0	5	5
Way	5	5	0	5	5

Table 4 Table of Score of Common Words

	Meaning	Etymology	Synonyms and antonyms	Meaning in Text	Sentence	Total
Average	4.4	4.6	2.23	4.67	4.67	20.57

Table 5 Table of Rare Words

	Meaning	Etymology	Synonyms /Antonyms	Meaning in Text	Sentence
Draconian	5	5	2.5	5	5
Convivial	5	5	1.5	5	5
Egregious	2.5	5	0.5	5	5
Imperious	2.5	5	1.5	5	5
Visceral	2.5	5	0	5	5

Table 6 Table of Score of Rare Words

	Meaning	Etymology	Synonyms and antonyms	Meaning in Text	Sentence	Total
Average	3.5	5	1.2	5	5	19.7

Table 7 Table of Professional Words

	Meaning	Etymology	Synonyms /Antonyms	Meaning in Text	Sentence
Architecture	5	5	1.5	5	5
cluster	5	5	2	5	5
Compile	5	5	2	5	5
Distribute	3	5	1	5	5
Dynamic	4	5	0.5	5	5

Table 8 Table of Score of Professional words

	Meaning	Etymology	Synonyms and antonyms	Meaning in Text	Sentence	Total
Average	4.4	5	1.4	5	5	20.8

5.1 Analysis

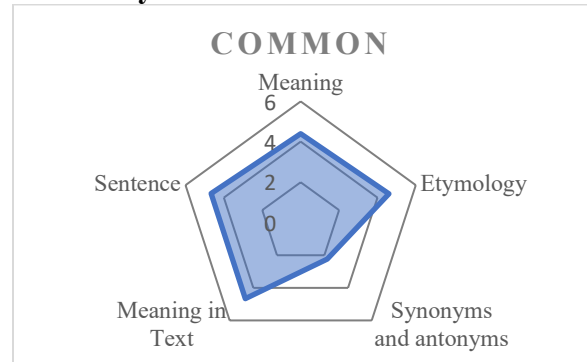


Figure 9 Common Word 5-dimensional Graphic

Based on the provided data, for common words, ChatGPT received a score of 4.4 out of 5 for evaluating the meaning of the words provided by the AI response, 4.6 out of 5 for finding the etymology, 2.23 out of 5 for answering synonyms and antonyms for given words, 4.67 out of 5 for making five sentences with the given words, and 4.67 out of 5 for interpreting words in context.

These results suggest that ChatGPT is proficient in finding the etymology of common words, making sentences with them, and interpreting them in context. However, its ability to understand the meaning of common words and answer synonyms and antonyms for given words appears to be a weaker area that may require further development.

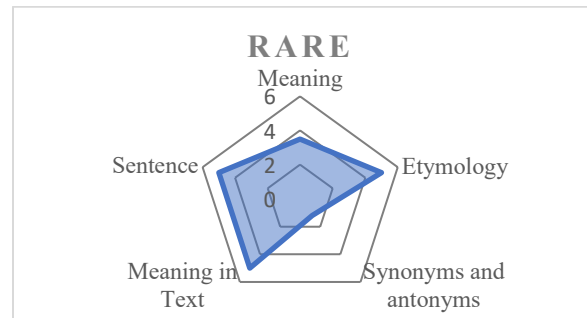


Figure 10 Rare Word 5-dimensional Graphic

Based on the data provided, for the rare word evaluated, ChatGPT received a score of 3.5 out of 5 for evaluating the meaning of the words provided by the AI response, 5 out of

5 for finding the etymology, 1.2 out of 5 for answering synonyms and antonyms for given words, 5 out of 5 for making five sentences with the given words, and 5 out of 5 for interpreting words in context.

These results suggest that ChatGPT exhibits a high level of proficiency in finding the etymology of rare words, as well as the ability to make sentences and interpret words in context. However, its ability to understand the meaning of rare words and answer synonyms and antonyms for given words appears to be weaker.

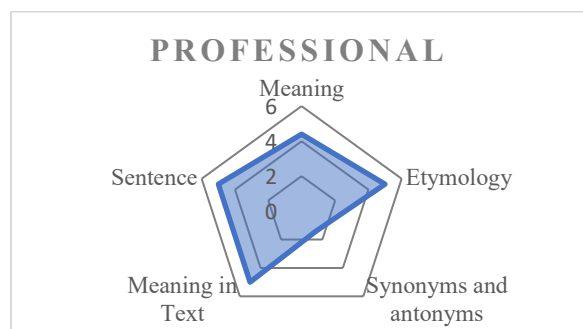


Figure 11 Professional Word 5-dimensional Graphic

According to the provided data, for professional words, ChatGPT received a score of 4.4 out of 5 for evaluating the meaning of the words provided by the AI response, 5 out of 5 for finding the etymology, 1.4 out of 5 for answering synonyms and antonyms for given words, 5 out of 5 for making five sentences with the given words, and 5 out of 5 for interpreting words in context.

These results indicate that ChatGPT is proficient in finding the etymology of professional words, making sentences with them, and interpreting them in context. However, its ability to understand the meaning of professional words and answer synonyms and antonyms for given words appears to be a weaker area.

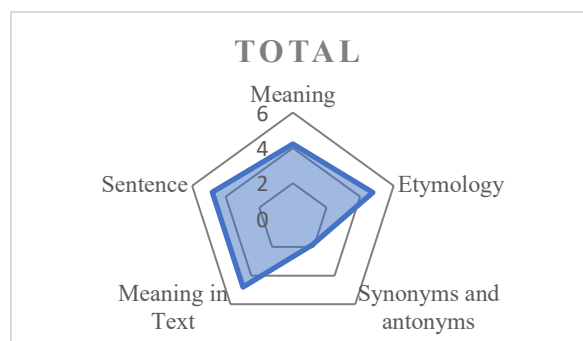


Figure 13 Total Word 5-dimensional Graphic

Based on the provided data, the evaluation of ChatGPT's performance on a total set of words yielded a score of 4.22 out of 5 for evaluating the meaning of the words provided by the AI response, 4.76 out of 5 for finding the etymology, 1.86 out of 5 for answering synonyms and antonyms for given words, 4.8 out of 5 for making five sentences with the given words, and 4.8 out of 5 for interpreting words in context.

These results suggest that ChatGPT exhibits a relatively high level of proficiency in comprehending the meaning of words and finding their etymology. It also demonstrates a solid ability to make sentences with the given words and interpret them in context. However, its ability to answer synonyms and antonyms for given words appears to be a weaker area that may require further development.

5.2 Conclusion

In terms of results, as a weak artificial intelligence, ChatGPT's natural language processing can well identify the meaning of each word. It can carry on a conversation and provide helpful answers to various questions and tips. However, it is worth noting that ChatGPT scored lower in answering synonyms and antonyms. One reason is that ChatGPT's training set contains more synonyms and antonyms that are not in the dictionary. ChatGPT's answer is difficult to accurately find the corresponding answer from the dictionary. ChatGPT's language skills are strong. In actual teaching, in scenarios where precision and logic requirements are not very strict, it can already play a certain auxiliary role.

6. Reference

- [1] Kohen, T. (2019) "The Ai Revolution," XRDS: Crossroads, The ACM Magazine for Students, 25(3), pp. 38–40. Available at: <https://doi.org/10.1145/3313159>.
- [2] Christiano, P. et al. (2023) Deep reinforcement learning from human preferences, arXiv.org. Available at: <https://arxiv.org/abs/1706.03741> (Accessed: April 20, 2023).
- [3] Hony, H.C., Iz, F. and Alderson, A.D. (1992) The Oxford Turkish Dictionary. Oxford.
- [4] Skeat, W.W. (1948) A concise etymological dictionary of the English language, by the rev. Walter W. Skeat .. Oxford: Clarendon Press.
- [5] FALLOWS, S.A.M.U.E.L. (2021) Complete dictionary of synonyms and Antonyms, or synonyms and words of opposite meaning. S.I.: MJP PUBLISHERS.
- [6] "CHATGPT: Open possibilities" (2023) Iraqi Journal for Computer Science and Mathematics, pp. 62–64. Available at: <https://doi.org/10.52866/20ijcsm.2023.01.01.0018>