Measuring Usability and User Experience with Eye-Tracking: Predicting Pragmatic and Hedonic Quality using Machine Learning

Fabian Engl, Timur Ezer, Jürgen Mottok
Software Engineering Laboratory for Safe and Secure Systems
OTH Regensburg
Regensburg, Germany
email: {fabian.engl | timur.ezer | juergen.mottok}@oth-regensburg.de

***Abstract*—This paper compares six different Machine Learning (ML) algorithms — the k-nearest neighbor algorithm, a Support Vector Machine, a Multi-Layer Perceptron, a Random Forest, Gradient Boosting, and Adaptive Boosting — in their ability to classify users based on their usability and user experience (UX) ratings, using only eye-tracking data. A study was designed using three different websites from German drinking water providers, with the corresponding usability and UX ratings based on the User Experience Questionnaire (UEQ) and the AttrakDiff questionnaire. In total, 104 participants, contributing over 18 hours of eye-tracking data, took part in the study. The results indicate that Machine Learning models trained on smaller datasets, such as those in the field of eye-tracking, often achieve reasonable F1-scores without the need for extensive hyperparameter tuning. A comparison of random and Bayesian optimization approaches reveals that especially tree-based models benefit from Bayesian optimization. Among all models, the Support Vector Machine and Multi-Layer Perceptron perform the best, averaging F1-scores in the 90 % range, and demonstrating that usability and UX can be predicted using similar approaches across different websites within the same domain. Additionally, no significant difference was found between the usability and UX definitions of the UEQ and the AttrakDiff, suggesting that both are equally suitable for UUX predictions based on Machine Learning and eye-tracking.**

Keywords-Machine Learning; Eye-Tracking; Usability; User Experience; UX.

I. Introduction

With the advancing digitization, everyday tasks are progressively shifting towards the digital realm. Relevant information can often only be found in digital form, and users are required to fulfill more tasks by themselves online, ranging from financial transactions to travel arrangements. Websites are among the most typical and widely used digital products in today's society, making it essential to design them with the users needs in mind. Usability and User Experience (UX) play a significant role in digital product design [1], making their assessment during development an important consideration. UUX — the combination of usability and user experience — is typically assessed in more traditional ways, relying on questionnaires or qualitative methods, such as think-aloud protocols [2].

Despite advancements in technologies like Electroencephalogram (EEG) and eye-tracking, which have become more accessible and affordable in recent years [3], they are rarely employed to assess UUX. The vast amount of output data, with sensors producing hundreds of measurements per second, and the required expertise can discourage their use [4]. With the rapidly growing trend of Machine Learning, a new question arises: Can both technologies be combined to address UUX in a more data-driven manner?

This paper addresses the generalizability of UUX measurability when combining Machine Learning models with eye-tracking, training them solely on eye movements. To investigate this, an eye-tracking study was designed, comparing three websites from German drinking water providers. A total of 104 participants took part in the study, resulting in over 18 hours of eye-tracking data and UUX ratings based on the User Experience Questionnaire (UEQ) and AttrakDiff questionnaires. Six commonly used Machine Learning models — k-nearest Neighbors (KNN), Support Vector Machine (SVM), Multi-Layer Perceptron (MLP), Random Forest (RF), Gradient Boosting (GB) and Adaptive Boosting (ADA) Models — were trained to separately classify the users into those who rated the usability and UX of the websites as low and those that rated them high.

This paper is structured as follows: First, Section II addresses related work in the field of eye-movement- and machine-learning-based UUX predictions, with shortcomings, research questions, and hypotheses elaborated in Section III. Following this, Section IV provides a brief introduction to the study design, used questionnaires, demographic information about the participants, and the technical setup. Sections V, Section VI and VII introduce common eye movement metrics, data preperation steps and Machine Learning model evaluation methods, respectively. Section VIII presents the classification results and answers the previously formulated research questions. Finally, Sections IX, X, and XI summarize the results, address the limitations of the current study, and offer an outlook for future research while discussing the implications of the findings.

II. Related Work

Usability and UX have been studied using eye-tracking before, even Machine Learning approaches are nothing new. Here, websites play a significant role. However, when looking at previous publications most of the time only one of the two UUX dimensions is analyzed.

Koonsanit *et al.* [5] for example study the effect of strong and weak signifies in URLs, which consist of differently highlighted links, with and emphasize on usability. They analyze how the level of highlighting helps participants to identify linked sites [5]. Instead of developing features from different types of eye movements, they train their model purely based on heatmaps, which they aggregate using a principal component analysis. They train different Machine Learning models to detect which users were looking at websites with strong and which were looking at those with weak signifies. They compare data from eleven participants and report accuracy results peaking at 90 % for the best model [5].

Cao *et al.* [6] take a similar approach and compare different website prototypes using eye-tracking data. 30 users had to find specific products on four versions of an e-commerce platform. Cao *et al.* train Machine Learning models to predict usage intention, splitting the participants based on their interest in using the website again [6]. They report accuracy, recall and precision metrics ranging from 71.7 % to 85.0 %; 57.5 % to 93.0 % and 62.5 % to 87.0 % respectively, with the deep neural network performing the best [6].

Wang *et al.* [7] studied search engines in particular, measuring how certain eye movement could be used to predict satisfaction levels. In total, eye-tracking data was collected from 48 participants, which were asked to find four different publications in the *Web of Science* database. Satisfaction was measured using a seven-point Likert scale and the predicted using both regression and a classification, which differentiated between the two groups rating the satisfaction low-to-medium of high. Their models achieved accuracies roughly between 64% and 68% in classification and R^2 scores between 0.02 and 0.75 for regression [7].

Pappas *et al.* [8] study whether visual appeal can be predicted based on eye-tracking, further more focusing on how much eye-tracking data is actually required to make valid predictions. They use a questionnaire from [9], which differentiates four different aspects of visual appeal including simplicity, diversity, colorfulness and craftsmanship [9]. Based on data from 23 participants they show that using a random forest regression, 15 to 20 seconds of recording data were enough not make predictions with a Normalized Root Mean Squared Error (NRMSE) of 0.1 to 0.14 across all four different categories. 25 seconds of available eye-tracking data only improved the data marginally, while 10 seconds lead to a noticeable decline in prediction error up to a NRMSE of 0.17 [8].

In addition to aesthetic, Öder *et al.* [10] demonstrate that it is possible to classify and differentiate between users which have previously visited a website and new visitors. While not directly linked to usability or UX their results show that it is possible to distinguish both groups using Machine Learning. As one of few they also report precision, recall and f-scores in addition to the classical accuracy metric. Their F1-Scores range from 0.382 to 0.836 depending on the task, differentiating between simple browsing and searching tasks [10].

Typical eye-tracking metrics used by the different studies consist of fixations, saccades, blinks and even pupil data [6] [8]. Most of the aforementioned papers use features out of multiple categories, with the majority using at least fixations and saccades. The Machine Learning models also varied from simple k-nearest Neighbors algorithms to random forests, support vector machines and even deep learning approaches [6] [8] [10].

III. Research Questions

The screened literature analyzes websites in many different ways, often focusing on specific partial aspects, such as usage intention, visual appeal, differentiating user groups or even usability as a whole. However, they show two severe shortcomings:

First of all, almost none of the papers use their own un-validated questionnaires, readily breaking down the concepts usability and UX. This both makes it difficult to compare individual studies and often fails to depict scientifically accepted UUX models in a broader sense.

Second of all, none of the studies analyze multiple different websites, but rather use eye-tracking as a technology for UUX evaluation with Machine Learning models being used as a tool to analyze the vast and often huge eye-tracking datasets. It remains unclear whether the results reported by the researchers are one-time observations or whether usability and UX can be measured using the same features and Machine Learning models across multiple different websites. For this reason, this paper tries to predict both dimensions using three different websites within the same domain. Further details about the study design and the used labels can be found in Section IV.

Having addressed this current research gap this papers aims at filling the gap regarding these gaps in the research area of machine-learning-based UUX predictions using eye-tracking data. To do so, the following three Research Questions (RQ) and corresponding Hypotheses (H) are presented:

RQ1 How much hyperparameter tuning do ML models require to optimize classification performance based on eye-tracking data?

RQ2 Which Machine Learning models are most suited for predicting UUX using only eye-tracking metrics?

RQ3 How do the Machine Learning predictions differ for usability and UX?

H1 ML models trained on comparatively small datasets, require fewer hyperparameter adjustments to reach near-optimal classification performance.

H2 More complex models, such as neural networks are better in detecting patterns in the eye-tracking data compared to more simpler models, such as decision-tree-based approaches.

H3 The ML models can classify usability more accurately compared to UX.

The selection of the Machine Learning models used in this study - consisting of a k-nearest Neighbors (KNN) algorithm, a Support Vector Machine (SVM), a Multi-Layer Perceptron (MLP), a Random Forest (RF), Gradient Boosting (GB) and Adaptive Boosting (ADA) — was guided by existing literature, with these models being commonly employed in similar studies. It is worth mentioning that there are many other types of ML algorithms available, which are out of scope for this study.

IV. Study design

This eye-tracking study examines three websites from German drinking water and non-alcoholic beverage manufacturers, selected to represent varying design quality and UX levels. Similar to a study conducted by Hassenzahl *et al.* who uses websites from liquor brands [11], water producers were chosen as a more neutral topic, avoiding biases and influences by factors, such as religious views. All sites were fully interactive and pre-downloaded to ensure consistent content during the recording session.

During the study participants completed tasks of varying difficulty, including finding company founding dates or drink ingredients, with the intent of ensuring varying usability ratings. Each participant had 30 seconds to explore the site freely, followed by a three-minute task. If completed early, they continued browsing to ensure sufficient eye-tracking data. Tasks and questionnaires were mouse-only to keep participants focused on the screen.

A. Usability and UX Questionnaires

As previously mentioned, this paper aims to address usability and UX from a general perspective. These two dimensions are typically measured using questionnaires, with the User Experience Questionnaire (UEQ) and AttrakDiff being the most commonly used in the research field [2]. Both are based on Hassenzahl *et al.*'s model of Pragmatic (also referred to as Ergonomic) and Hedonic Quality. According to this model, a software's appeal is determined by its Pragmatic Quality (PQ), which represents usability, and its Hedonic Quality (HQ), which reflects user experience.

Both the UEQ and AttrakDiff assess usability and UX using bipolar word pairs, such as "boring" and "exciting" on a seven-point Likert scale. As the full versions with 26 and 28 pairs would have made the study too long, their validated short versions with 8 pairs each were used to keep the eye-tracking session manageable.

B. Participants

In total, 104 participant took part in the study, with 43.3% identifying as female ($n = 45$) and 56.7% as male ($n = 59$). Among them, 35.6% ($n = 37$) wore glasses or contact lenses during the study. The average age at the time of the study was 29.4 years ($min = 18$, $max = 67$, $sd = 11.18$).

Regarding education, 57 participants were students, 38 were employees or self-employed, two were retired, and one selected *other*. Among the students, five were also working at least part-time and were therefore counted in both the student and employed categories. Further looking at work experience, 31 participants had less than one year of full-time work experience, 19 had one to two years, 25 had two to five years, 10 had five to ten years, and 19 had more than ten years of experience.

C. Eye-Tracking Setup and Data Collection

The study was conducted using Tobii Pro Lab software (Version 1.232.52758) with up to nine mobile Tobii Pro Fusion eye-trackers operating simultaneously. The eye-trackers recorded at 250 Hz (Firmware Version D3417769DB, Driver Version: 2.10.7.0) and were attached to a 21-inch Full-HD (1920×1080) monitor with a 60 Hz refresh rate. Participants were positioned approximately 65 cm from the screen and instructed to remain still during the study. Whenever possible, direct light was minimized by turning off the ceiling lights and closing the blinds. These methodological choices align with the recommendations of Ezer *et al.* [12] [13].

All participants were briefed and signed a consent form approved by the Joint Ethics Committee of the Bavarian Universities (GEHBa). Participation was voluntary, and anonymous identifiers were used to ensure data privacy. To further maintain data quality, two quality thresholds were specified: a calibration threshold of 0.75, based on prior Tobii Pro Fusion studies [14], with participants excluded if unmet, and a missing data threshold excluding stimuli with over 5 % missing data.

V. Eye Movement Metrics

This section provides an overview of common eye movement metrics and explains how they can be utilized as Machine Learning features for data-driven UUX predictions.

Fixation duration: A fixation is defined as an eye movement where the eye is relatively still for a period of time. The fixation duration describes the time in milliseconds for how long the fixation lasts [15, pp. 526-527].

K-Nearest Fixations: Some UUX studies also explore more complex fixation metrics, such as k-nearest fixations [16]. This concept is typically used for calculating saliency maps and determining the probability that a random fixation falls within a specific area [17], [18]. In this study, k-nearest fixations are not calculated in relation to predefined areas but rather to other fixations, with the goal of identifying spatially more closely viewed areas, as suggested by Yin *et al.*

Fixation Grid: This metric calculates the distribution of total fixations on a stimulus across 50 uniform areas of the screen. These areas are created by placing a 10x5 grid - roughly based on the screen ratio - over the stimulus. Each area on the stimulus can then be assigned a percentage of the fixations it contains, both in relation to the total fixation count as well as fixation duration [19].

Saccade Length: Saccade length is the distance of a saccade from its start to end point [15, p. 448]. The distances between fixations are a coarse approximation of saccade lengths. As done in this study, they are typically calculated as the Euclidean distance between fixation points [15, p. 448]. However, it is worth mentioning that also other implementations, such as the length of the saccade polyline, exist [15, pp. 447-448].

Saccade Velocity: This metric describes the average velocity of a saccade. It can be seen as an approximation of the first derivative of gaze position data with respect to time. [15, p. 463] In this paper, it is calculated by dividing the saccade length by the duration of the corresponding saccade.

Saccade Direction: The saccade direction describes the angle between a saccade and the horizontal axis in the coordinate system of the stimulus. Hereby, the saccade direction represents an idealistic straight line from the start to the end point of a saccade. It does not account for the curvatures of saccades [15, pp. 440-441].

NGRAMs: Similar to k-nearest fixations, NGRAMs represent a more complex saccade metric that quantifies saccade sequences by encoding their direction and length as upper- and lowercase character sequences. To achieve this, all possible saccade directions are divided into eight sections (see Figure 1), with the lowercase letter threshold set to the $Q_{0.25}$ for saccade length based on all saccades of the respective participant on that stimulus. The resulting strings are then transformed into Machine Learning features by extracting all recurring sequences using a sliding window approach, which counts the occurrences of each sequence. This procedure is illustrated in Figure 2. For this study, the sliding window was set to a size of two characters. However, both the window size and the number of sections can be adjusted arbitrarily, with more sections requiring a higher total number of saccades to ensure adequate sequence distribution.

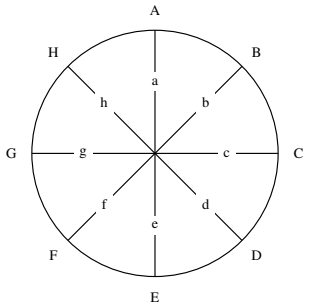

Figure 1. NGRAM Sections; Adaption based on the concept of Bulling *et al.* [20].

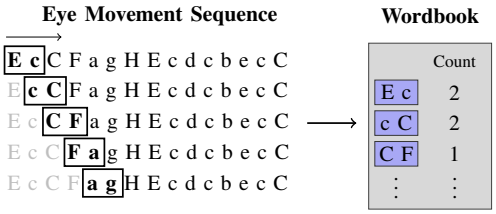

Figure 2. NGRAM to feature conversion; Visulization based on Bulling *et al.* [20].

VI. Data Preparation

Before training the Machine Learning models, the aforementioned eye-tracking metrics were transformed into features. This step, known as feature engineering, is crucial as it aggregates the (semi-)raw data consisting of multiple eye movements into a structured format that Machine Learning models can more easily process to make predictions. Feature engineering typically involves different forms of data representation, such as distributions, vectors, or other aggregation methods [21, pp. 21–25]. Afterwards, correlation-based feature selection was applied to make the dataset more interpretable for ML models, as suggested by Hall [22]:

$$\text{Merit}_s = \frac{k \cdot \overline{r_{cf}}}{\sqrt{k + k(k-1) \cdot \overline{r_{ff}}}}$$

where k is the number of selected features, $\overline{r_{cf}}$ is the average feature-class correlation and $\overline{r_{ff}}$ is the average feature-feature correlation. This merit ensures that

only features with a high class- and low inter-feature-correlation remain in the final dataset [22]. Afterwards all features were normalized using a Standard Scaler.

A. Labels

To assess whether Machine Learning can distinguish between low and high UUX ratings, the seven-point Likert scale was split into two classes: < 4 (low) and > 4 (high). Neutral ratings ($= 4$) were excluded to ensure a clear separation between groups. Table VI-A summarizes the final class distributions.

TABLE I. CLASS DISTRIBUTIONS FOR WEBSITES AND LABELS.

Label	Website 1		Website 2		Website 3	
	< 4	> 4	< 4	> 4	< 4	> 4
PQ UEQ	23	65	33	56	14	77
PQ AttrakDiff	22	67	28	58	13	81
HQ UEQ	55	26	50	36	13	91
HQ AttrakDiff	39	47	39	47	7	81

While the first two websites show rather balanced classes, Website 3 deviates notably, especially in Hedonic Quality based on the AttrakDiff Questionnaire. This will be discussed further in Section VIII.

VII. MACHINE LEARNING MODEL EVALUATION

Various metrics can be used to assess the classification performance of Machine Learning algorithms. These metrics allow not only for the comparison of different algorithms but also for the evaluation of the same algorithm under varying hyperparameter settings. In the following subsections, the F1 score is introduced as a key evaluation metric, followed by an exploration of different approaches to hyperparameter tuning.

A. Evaluation Metrics

For two-class problems, multiple metrics can be used to quantify the classification performance of algorithms. Metrics, such as accuracy, precision, F1-score, Cohen's Kappa, and Matthew's Correlation Coefficient, among others, are suitable for this purpose [23] [24]. However, in the present work, we utilize the F1-score as a performance measure, as it is the most commonly used metric [25]. It can be calculated following [26] as:

$$\text{F1-Score} = \frac{2 \cdot \text{TP}}{2 \cdot \text{TP} + \text{FP} + \text{FN}}$$

where TP, FP, and FN represent the number of true positive, false positive, and false negative predictions, respectively. The higher the F1-score, the better the classification performance, with F1-score $\in [0; 1]$.

B. Hyperparameter Tuning of the ML Models

Unlike model parameters, which are learned during training, hyperparameters are predefined and remain unchanged throughout the learning process. Thus, it is essential to optimize these hyperparameters in order to improve the classification performance. Several approaches exist for hyperparameter tuning, like random search, grid search, and Bayesian optimization:

Grid Search: Grid search employs a brute-force method for model selection through cross-validation. It systematically explores predefined sets of hyperparameter values, training a model for each combination. The model achieving the highest performance score is chosen as the optimal one. [27, pp. 210-211]

Random Search: An alternative to grid search's exhaustive search is selecting a fixed number of random hyperparameter combinations from user-defined parameter ranges. This method, known as randomized search, samples hyperparameter values randomly and without replacement for the provided distribution. [27, pp. 212-213]

Bayesian Search: Bayesian search works similarly to random search in that it also relies on a fixed number of iterations rather than evaluating all possible parameter combinations. However, instead of selecting parameters completely at random, it considers past classification performance to guide future selections. It chooses hyperparameters based on expected improvement or the upper Gaussian confidence bound, focusing on well performing hyperparameter ranges within the provided search space. By doing so, Bayesian search refines the parameter range iteratively, potentially leading to more efficient optimization requiring fewer iterations. [28] [29]

Since the number of hyperparameters varies between models and no universally applicable set of hyperparameters exists, this paper utilizes only random and Bayesian search to optimize the Machine Learning models. The effectiveness of both approaches is compared in the next section.

VIII. RESULTS

Starting with RQ1, the focus is first set on optimizing the Machine Learning models, specifically analyzing the convergence of F1-scores over time when comparing random search and Bayesian search. It is essential to differentiate between these two approaches, as Bayesian search by default is set to have fewer iterations [29]. To account for this discrepancy, all models were optimized using 100, 500, 1,000, 2,500, and 5,000 iterations for random search, while 10, 50, 100, 150, and 250 iterations were used for Bayesian search. The chosen scaling increases steeply to illustrate F1-score development in relation to computational complexity.

The detailed results of both search methods are presented in Table II and visualized in Figure 3, with iteration as 1 representing a completely untrained model.

Both show that the two optimization approaches generally yield similar final results, with F1-scores increasing sharply at the beginning of the optimization process before gradually plateauing as the number of iterations grows. Examining individual models, Figure 3 reveals a clear trend: models with fewer hyperparameters to tune — such as KNN, SVM, MLP, and ADA — tend to reach an optimization ceiling earlier, with only marginal improvements beyond a certain point. Between 1,000 and 5,000 iterations for random search and 100 and 250 iterations for Bayesian search, these four models show only slight F1-score improvements, ranging from 0.5 % for KNN to 0.9 % for AdaBoost.

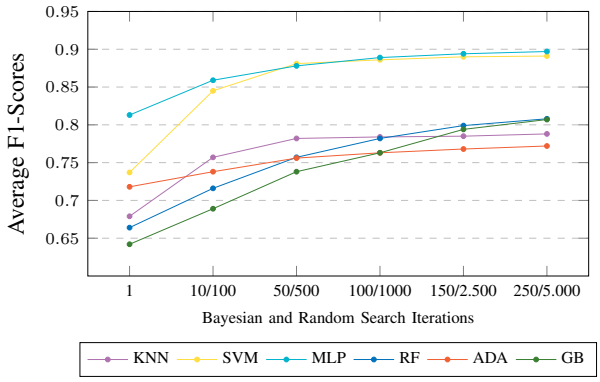

Figure 3. Average F1-Score Development of all Machine Learning Models with increasing Hyperparameter Tuning Iterations based on all websites and labels.

In contrast, algorithms with a larger hyperparameter space, such as Random Forest and Gradient Boosting, continue to show notable improvements even beyond 100/1,000 iterations. The F1-score for Random Forest improved by an average of 2.6 % exceeding these iterations, while Gradient Boosting sees an even greater gain of 4.4 % over the same range. A more detailed view in Figure 4 shows that these gains are particularly pronounced for Bayesian search, which outperforms random search by 2.4 % for the Random Forest and 4.8 % for the Gradient Boosting model. This is likely due to the higher inter-dependencies within the larger hyperparameter space, which are more effectively aligned by Bayesian optimization than by random sampling.

While this effect is primarily evident for Random Forest and Gradient Boosting, Bayesian search also slightly outperforms random search on average across all models (see Figure 5). However, the difference between the two approaches is most pronounced at lower iteration counts, particularly for fewer than 1.000 iterations in random search.

Following up with RQ2, a clear trend emerges when analyzing the performance of different Machine Learning models across all three websites. Figures 6, 7, and 8 present the F1-scores for all models and the four

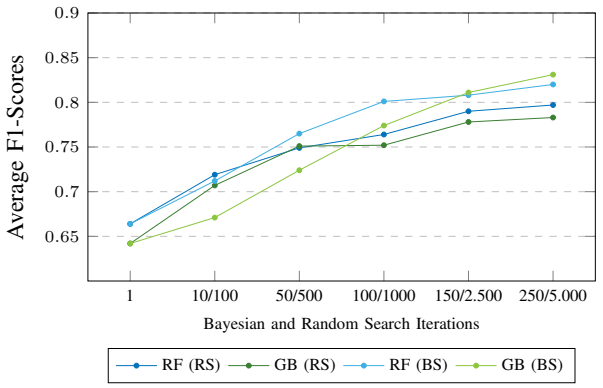

Figure 4. Average F1-Score Development of the Random Forest (RF) and the Gradient Boosting Model (RB) comparing Random Search (RS) and Bayesian Search (BS) based on all websites and labels.

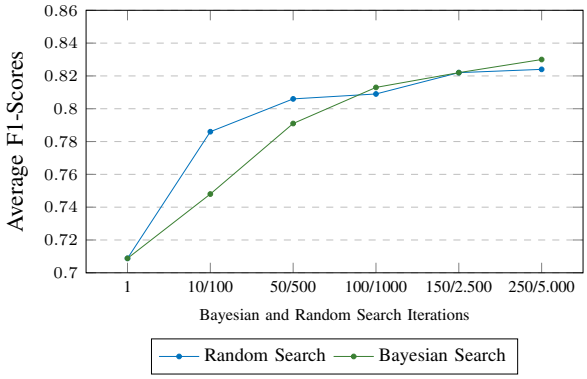

Figure 5. Average F1-Score Development comparing Random and Bayesian Search based on all Machine Learning models, websites and labels.

classification labels across all three websites. Regardless of the website, the SVM and the MLP stand out, as they consistently achieve F1-scores close to or above 90%. On the first two websites, MLP outperforms SVM, though at times only by a negligible lead. However, on the third website, SVM takes the lead, with MLP following closely behind, showing a similar performance trend across all labels.

The three tree-based models — Random Forest, AdaBoost, and Gradient Boosting — demonstrate comparable performance, with Random Forest generally achieving the highest F1-scores out of the three Machine Learning models. Nevertheless, there are exceptions: for the UEQ Pragmatic Quality label on Website 1 and both Pragmatic Quality labels on Website 3, Gradient Boosting outperforms Random Forest by 5.8 % and 5.9 %, respectively. Aside from these cases, Random Forest maintains a slight advantage. Among the tree-based models, AdaBoost consistently underperforms compared to both Random Forest and Gradient Boosting.

Lastly, the k-nearest Neighbor (KNN) algorithm shows inconsistent classification performance across all

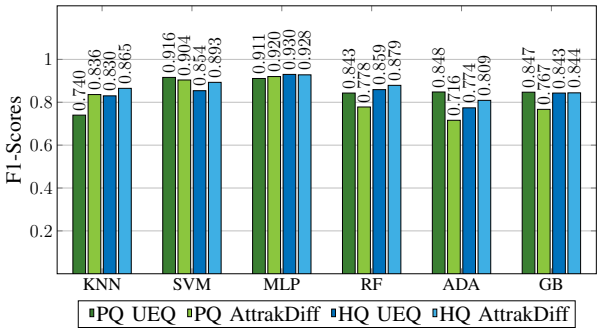

Figure 6. F1-Scores of all Machine Learning Models for the Labels on Website 1.

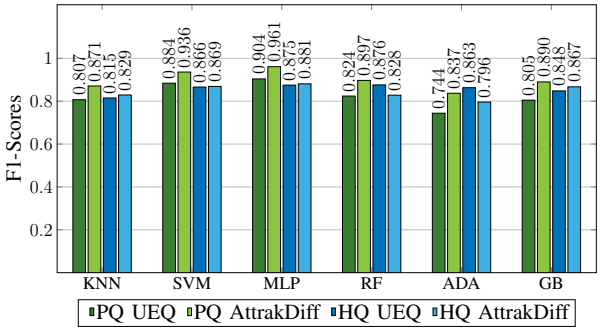

Figure 7. F1-Scores of all Machine Learning Models for the Labels on Website 2.

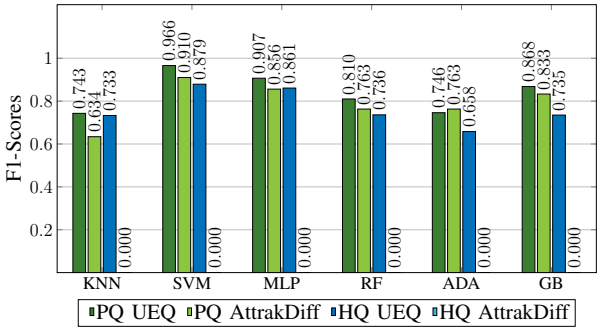

Figure 8. F1-Scores of all Machine Learning Models for the Labels on Website 3.

labels. While it occasionally outperforms the tree-based models, such as for the AttrakDiff Pragmatic Quality label on Website 1, it falls behind in most cases, achieving the lowest F1-scores in four out of eleven labels.

Considering these results, H2 cannot be refuted, as the more complex MLP model frequently outperforms the other models. Yet, in general, both MLP and SVM prove to be adequate choices for classifying participants' UUX ratings based on eye movement data. In contrast, KNN and AdaBoost perform the worst in this study, likely due to their simpler evaluation mechanisms, which rely on spatial distances between data points or single parameter thresholds within individual features (decision stumps). This suggests that while these models can differentiate UUX labels to some degree, more eye movement features are needed at a time to effectively differentiate between the UUX labels.

RQ3 shifts the focus from the performance of the individual models to the broader question of whether usability (Pragmatic Quality) and UX (Hedonic Quality) labels, as defined by the UEQ and AttrakDiff questionnaires, can be reliably predicted. Averaging the F1-scores across all Machine Learning models shows similar results compared to the individual websites including both questionnaires. Thus, proofing their ability to classify the UUX ratings accurately (see Figure 9).

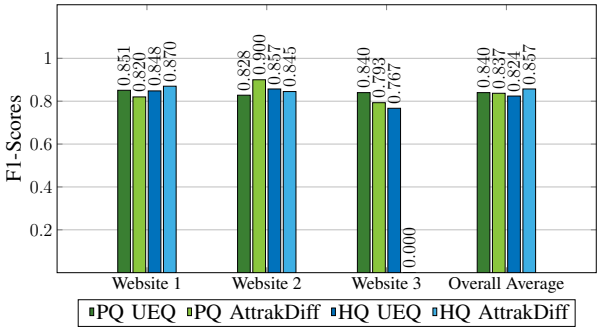

Figure 9. Average F1-Scores of all Machine Learning Models by Websites and Labels.

However, when examining the two questionnaires in detail, some differences emerge across websites. For example, on Website 2, the Pragmatic Quality label from the AttrakDiff questionnaire (89.9 %) was predicted with an average F1-score 7.1 % higher than that of the UEQ (82.8 %). In contrast, on Website 3, this trend reverses, with the UEQ Pragmatic Quality label (83.8 %) outperforming the AttrakDiff (79.3 %) equivalent by 4.7 %. This suggests that the predictability of the two questionnaires may change depending on the underlying stimuli and, therefore, changing eye movement patterns.

However, when averaging results across all websites, these differences become minimal, as shown in Figure 9. For Pragmatic Quality, the difference in classification performance between the questionnaires is only 0.2 %. Although the difference for Hedonic Quality is larger at 3.3 %, it's important to note that AttrakDiff's Hedonic Quality could not be calculated for Website 3 — where all models performed worse overall — due to too few data points. Assuming Website 3 would have followed the same trend as the other sites, Hedonic Quality predictions are likely to even out as well. This is supported by results from Websites 1 and 2, where Hedonic Quality scores between UEQ and AttrakDiff differ by just 0.5 %.

Due to this limitation, H3 can neither be rejected nor supported. Following the aforementioned assumption, both usability (Pragmatic Quality) and UX (Hedonic

Quality) can be classified with nearly identical certainty. Although UX predictions are slightly less accurate than usability predictions, the difference is too small to draw definite conclusions about whether this is a generalizable finding or simply a study-specific artifact. To answer this question, more websites would have to be included in the study.

IX. Summary of results

This study showed that usability (Pragmatic Quality) and UX (Hedonic Quality), as defined by the UEQ and AttrakDiff, can be effectively predicted using Machine Learning models trained solely on eye-movement features. Comparing random and Bayesian hyperparameter tuning, both approaches produced similar results, though tree-based models particularly benefited from Bayesian search, likely due to their complex hyperparameter space. However, performance gains across all models plateaued — around 150 iterations for Bayesian search and 2,500 for random search.

Among all models, the SVM and MLP performed the best, consistently reaching F1-scores in the 90 % range, reaching these scores with even minimal hyperparameter tuning. Out of the tree-based models, Random Forest performs best, followed by Gradient Boosting, while AdaBoost and KNN show the lowest classification performance.

Finally, comparing UEQ and AttrakDiff reveals no differences in predictive performance. Both usability and UX are equally predictable from eye-tracking data, regardless of which questionnaire is used, with average results across all labels and websites showing negligible differences.

X. Limitations

This study has two main limitations. First, it is difficult to determine whether random or Bayesian hyperparameter tuning is superior, as both methods would likely converge to similar results over more iterations. Thus, this comparison should be seen as a general guideline rather than a strict conclusion, particularly for similar eye-tracking datasets. Its applicability to other datasets remains to be tested.

Second, the size of the dataset raises the question of whether the amount of collected data is sufficient. This is a common challenge in eye-tracking research, as data collection is both time-consuming and complex. However, this study includes a relatively large participant pool and diverse stimuli compared to similar studies. Future research could strengthen these findings by incorporating more participants and websites.

Additionally, class imbalances in the dataset may have influenced classification performance. As noted in Section VII, using additional metrics, such as Cohen's Kappa or Matthew's Correlation Coefficient alongside the F1-score would provide a more comprehensive evaluation of model performance in handling imbalanced data.

XI. Conclusion and Future Work

This paper contributes to the growing field of data-driven UUX research based on eye-tracking, demonstrating within a broader context what previous studies have shown in more narrow use cases: both usability and UX, as defined by commonly used UUX questionnaires, such as the UEQ and AttrakDiff, can be predicted by training Machine Learning models on eye movement data. The findings suggest that this predictability is not limited to a single product but extends across a range of similar digital products within the same domain. This supports the assumption that specific eye movement patterns are systematically linked to participants' perception of a product's usability and UX.

Building on these findings, future studies could explore several aspects of UUX. One next step could include a broader range of websites to further test the generalizability of eye-tracking-based UUX predictions. Another key question is whether trained Machine Learning models can identify patterns across multiple websites rather than being limited to one. If so, datasets could be expanded by aggregating data from different websites, improving both hyperparameter tuning and prediction robustness.

Additionally, the labels could be examined in more detail. This study classified only between low and high UUX ratings. Future research should explore whether models can distinguish between low, neutral, and high scores, moving beyond binary classification. Success in this area could enable regression models to predict continuous UUX scores, allowing for more nuanced assessments of usability and UX.

Despite these opportunities for future work, the present results are already highly promising, with the highest F1-scores among existing literature in this research field.

Acknowledgments

This study was conducted as part of the EDIH *Digital Innovation Ostbayern (DInO)*, which is funded by the European Union and the European Funds for Regional Development (EFRE) (References: 101083427 and 20-3092.10-THD-105) as well as by the 'German Federal Ministry of Education and Research' (BMBF) through the granting of the 'Bavarian State Budget' (ZD.B) (FKZ: 16-1541).

The study was approved by the Joint Ethics Committee of the Bavarian Universities (GEHBa) (Reference: GEHBa-202312-V-155-R).

We acknowledge the use of DeepL Write (DeepL SE, https://www.deepl.com/write) and ChatGPT (OpenAI,

TABLE II. OVERVIEW OF F1-SCORES FOR ALL MODELS, WEBSITES, LABELS, AND HYPERPARAMETER OPTIMIZATION METHODS. F1-SCORES THAT NO LONGER SHOW IMPROVEMENT ARE GRAYED OUT. THE BEST RESULTS ARE MARKED BASED ON THE OPTIMIZATION METHOD THAT ACHIEVED THE HIGHEST SCORE: RS = RANDOM SEARCH, BS = BAYESIAN SEARCH, X = IDENTICAL RESULTS.

Website	Label	Algorithm	Untrained Model	Random Search (RS)					Bayesian Search (BS)					Best Result
				100	500	1.000	2.500	5.000	10	50	100	150	250	
Website 1	PQ UEQ	KNN	0.664	0.699	0.723	0.723	0.734	0.740	0.684	0.723	0.723	0.723	0.723	0.740 (RS)
		SVM	0.720	0.905	0.905	0.916	0.916	0.916	0.916	0.916	0.916	0.916	0.916	**0.916 (X)**
		MLP	0.799	0.890	0.897	0.897	0.897	0.897	0.872	0.897	0.911	0.911	0.911	0.9111 (BS)
		RF	0.644	0.736	0.736	0.750	0.792	0.823	0.763	0.769	0.823	0.843	0.843	0.843 (BS)
		ADA	0.742	0.808	0.822	0.822	0.848	0.848	0.799	0.799	0.811	0.811	0.811	0.848 (RS
		GB	0.593	0.755	0.807	0.807	0.807	0.807	0.745	0.757	0.794	0.847	0.847	0.847 (BS)
	PQ AttrakDiff	KNN	0.701	0.831	0.831	0.836	0.836	0.836	0.779	0.831	0.831	0.831	0.831	0.836 (RS)
		SVM	0.737	0.900	0.900	0.900	0.900	0.904	0.855	0.855	0.904	0.904	0.904	0.904 (X)
		MLP	0.853	0.919	0.919	0.919	0.920	0.920	0.851	0.887	0.906	0.906	0.919	**0.920 (RS)**
		RF	0.663	0.666	0.744	0.744	0.762	0.762	0.684	0.723	0.778	0.778	0.778	0.778 (BS)
		ADA	0.713	0.702	0.716	0.716	0.716	0.716	0.698	0.698	0.713	0.713	0.716	0.716 (X)
		GB	0.582	0.702	0.702	0.702	0.716	0.731	0.680	0.680	0.742	0.767	0.767	0.767 (BS)
	HQ UEQ	KNN	0.693	0.778	0.830	0.830	0.830	0.830	0.774	0.830	0.830	0.830	0.830	0.830 (X)
		SVM	0.800	0.837	0.854	0.854	0.854	0.854	0.822	0.837	0.842	0.854	0.854	0.854 (X)
		MLP	0.889	0.907	0.915	0.930	0.930	0.930	0.835	0.838	0.869	0.881	0.895	**0.930 (RS)**
		RF	0.708	0.748	0.764	0.816	0.832	0.832	0.784	0.784	0.849	0.851	0.859	0.859 (BS)
		ADA	0.718	0.745	0.770	0.770	0.772	0.774	0.745	0.759	0.759	0.770	0.770	0.774 (RS)
		GB	0.617	0.764	0.812	0.812	0.826	0.826	0.675	0.811	0.819	0.819	0.843	0.843 (BS)
	HQ AttrakDiff	KNN	0.769	0.865	0.865	0.865	0.865	0.865	0.821	0.865	0.865	0.865	0.865	0.865 (X)
		SVM	0.858	0.892	0.892	0.892	0.893	0.893	0.812	0.881	0.881	0.881	0.881	0.893 (RS)
		MLP	0.893	0.917	0.927	0.928	0.928	0.928	0.880	0.893	0.917	0.917	0.928	**0.928 (X)**
		RF	0.743	0.715	0.775	0.775	0.801	0.814	0.737	0.802	0.868	0.868	0.879	0.879 (BS)
		ADA	0.719	0.773	0.809	0.809	0.809	0.809	0.771	0.784	0.795	0.795	0.798	0.809 (RS)
		GB	0.742	0.729	0.751	0.751	0.805	0.805	0.642	0.739	0.832	0.844	0.844	0.844 (BS)
Website 2	PQ UEQ	KNN	0.690	0.793	0.807	0.807	0.807	0.807	0.755	0.755	0.788	0.788	0.793	0.807 (RS)
		SVM	0.800	0.865	0.884	0.884	0.884	0.884	0.800	0.884	0.884	0.884	0.884	0.884 (X)
		MLP	0.874	0.874	0.888	0.888	0.904	0.904	0.847	0.871	0.874	0.874	0.888	**0.904 (RS)**
		RF	0.711	0.730	0.758	0.769	0.792	0.792	0.708	0.772	0.777	0.794	0.824	0.824 (BS)
		ADA	0.725	0.738	0.741	0.744	0.744	0.744	0.713	0.738	0.738	0.738	0.738	0.744 (RS)
		GB	0.663	0.729	0.738	0.740	0.788	0.795	0.692	0.732	0.732	0.752	0.805	0.805 (BS)
	PQ AttrakDiff	KNN	0.708	0.871	0.871	0.871	0.871	0.871	0.861	0.871	0.871	0.871	0.871	0.871 (X)
		SVM	0.766	0.934	0.934	0.934	0.934	0.936	0.843	0.914	0.914	0.916	0.916	0.936 (RS)
		MLP	0.922	0.961	0.961	0.961	0.961	0.961	0.838	0.917	0.961	0.961	0.961	**0.961 (X)**
		RF	0.755	0.826	0.826	0.826	0.862	0.862	0.749	0.785	0.861	0.882	0.897	0.897 (BS)
		ADA	0.795	0.798	0.798	0.803	0.809	0.835	0.767	0.795	0.835	0.837	0.837	0.837 (BS)
		GB	0.805	0.746	0.815	0.815	0.855	0.855	0.635	0.742	0.797	0.863	0.890	0.890 (BS)
	HQ UEQ	KNN	0.755	0.815	0.815	0.815	0.815	0.815	0.776	0.815	0.815	0.815	0.815	0.815 (RS)
		SVM	0.772	0.848	0.866	0.866	0.866	0.866	0.735	0.865	0.865	0.866	0.866	0.866 (RS)
		MLP	0.823	0.856	0.874	0.874	0.874	0.875	0.784	0.848	0.857	0.857	0.874	**0.875 (RS)**
		RF	0.741	0.672	0.742	0.807	0.814	0.818	0.682	0.844	0.844	0.844	0.876	0.876 (BS)
		ADA	0.818	0.850	0.852	0.852	0.852	0.852	0.804	0.850	0.850	0.863	0.863	0.863 (BS)
		GB	0.769	0.746	0.796	0.796	0.817	0.817	0.698	0.713	0.809	0.841	0.848	0.848 (BS)
	HQ AttrakDiff	KNN	0.757	0.829	0.829	0.829	0.829	0.829	0.779	0.803	0.803	0.803	0.803	0.829 (RS)
		SVM	0.821	0.845	0.857	0.857	0.857	0.869	0.795	0.834	0.834	0.834	0.845	0.869 (RS)
		MLP	0.846	0.868	0.880	0.880	0.880	0.880	0.843	0.843	0.880	0.880	0.881	**0.881 (BS)**
		RF	0.723	0.751	0.750	0.751	0.783	0.783	0.720	0.752	0.792	0.809	0.828	0.828 (BS)
		ADA	0.722	0.770	0.775	0.782	0.782	0.796	0.737	0.768	0.768	0.768	0.768	0.796 (RS)
		GB	0.615	0.699	0.776	0.776	0.779	0.779	0.651	0.768	0.811	0.867	0.867	0.867 (BS)
Website 3	PQ UEQ	KNN	0.584	0.741	0.741	0.741	0.743	0.743	0.631	0.739	0.741	0.743	0.743	0.743 (BS)
		SVM	0.584	0.932	0.947	0.952	0.966	0.966	0.810	0.947	0.966	0.966	0.966	**0.966 (X)**
		MLP	0.584	0.891	0.907	0.907	0.907	0.907	0.818	0.838	0.854	0.874	0.874	0.907 (RS)
		RF	0.547	0.742	0.795	0.795	0.795	0.795	0.672	0.797	0.797	0.797	0.810	0.810 (BS)
		ADA	0.689	0.742	0.746	0.746	0.746	0.746	0.689	0.746	0.746	0.746	0.746	0.746 (X)
		GB	0.540	0.669	0.762	0.762	0.778	0.781	0.756	0.756	0.798	0.815	0.868	0.868 (BS)
	PQ AttrakDiff	KNN	0.583	0.632	0.634	0.634	0.634	0.634	0.609	0.634	0.634	0.634	0.634	0.634 (X)
		SVM	0.675	0.890	0.890	0.910	0.910	0.910	0.764	0.877	0.877	0.877	0.877	**0.910 (RS)**
		MLP	0.730	0.848	0.848	0.848	0.856	0.856	0.820	0.832	0.832	0.832	0.832	0.856 (RS)
		RF	0.559	0.658	0.679	0.679	0.763	0.763	0.638	0.691	0.691	0.691	0.691	0.763 (RS)
		ADA	0.568	0.711	0.763	0.763	0.763	0.763	0.691	0.712	0.712	0.712	0.744	0.763 (RS)
		GB	0.553	0.674	0.674	0.674	0.711	0.730	0.582	0.646	0.667	0.793	0.833	0.833 (BS)
	HQ UEQ	KNN	0.569	0.669	0.718	0.718	0.733	0.733	0.666	0.669	0.669	0.669	0.733	0.733 (X)
		SVM	0.569	0.826	0.826	0.826	0.879	0.879	0.770	0.810	0.810	0.821	0.821	**0.879 (RS)**
		MLP	0.729	0.841	0.841	0.841	0.861	0.861	0.736	0.794	0.822	0.855	0.855	0.861 (RS)
		RF	0.516	0.667	0.667	0.691	0.697	0.719	0.696	0.696	0.729	0.729	0.736	0.736 (BS)
		ADA	0.687	0.595	0.599	0.599	0.658	0.658	0.580	0.590	0.652	0.652	0.652	0.658 (RS)
		GB	0.585	0.567	0.633	0.633	0.670	0.688	0.624	0.624	0.718	0.718	0.735	0.735 (BS)
	HQ AttrakDiff	—	—	—	—	—	—	—	—	—	—	—	—	—

GPT-4, https://chatgpt.com/) to assist in the formulation of this document. These tools were used only for language refinement or during the coding processes, not to generate content or ideas. Those originated solely from the authors or are based on the cited literature.

DATA

If you have any questions regarding the dataset, eye movement metric calculations or the python sklearn Machine Learning implementation, feel free to contact Fabian Engl using the contact information provided.

REFERENCES

[1] M. Hassenzahl, A. Platz, M. Burmester, and K. Lehner, "Hedonic and *ergonomic quality aspects determine a software's appeal*," en, in *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, ACM, Apr. 2000, pp. 201–208.

[2] E. Mortazavi, P. Doyon-Poulin, D. Imbeau, M. Taraghi, and J.-M. Robert, "Exploring the landscape of ux subjective evaluation tools and ux dimensions: A systematic literature review (2010–2021)," *Interacting with Computers*, vol. 36, no. 4, pp. 255–278, 2024.

[3] J. Š. Novák, J. Masner, P. Benda, P. Šimek, and V. Merunka, "Eye tracking, usability, and user experience: A systematic review," *International Journal of Human–Computer Interaction*, vol. 40, no. 17, pp. 4484–4500, 2024.

[4] R. Zemblys, D. C. Niehorster, and K. Holmqvist, "Gazenet: End-to-end eye-movement event detection with deep neural networks," *Behavior Research Methods*, vol. 51, no. 2, pp. 840–864, Apr. 2019.

[5] K. Koonsanit, T. Tsunajima, and N. Nishiuchi, "Evaluation of strong and weak signifiers in a web interface using eye-tracking heatmaps and machine learning," in *Computer Information Systems and Industrial Management*, K. Saeed and J. Dvorský, Eds., Cham: Springer International Publishing, 2021, pp. 203–213.

[6] Y. Cao *et al.*, "Detecting users' usage intentions for websites employing deep learning on eye-tracking data," *Information Technology and Management*, vol. 22, no. 4, pp. 281–292, Dec. 1, 2021.

[7] P. Wang, H. Yang, J. Hou, and Q. Li, "A machine learning approach to primacy-peak-recency effect-based satisfaction prediction," *Information Processing & Management*, vol. 60, no. 2, p. 103 196, Mar. 1, 2023.

[8] I. O. Pappas, K. Sharma, P. Mikalef, and M. N. Giannakos, "How quickly can we predict users' ratings on aesthetic evaluations of websites? employing machine learning on eye-tracking data," in *Responsible Design, Implementation and Use of Information and Communication Technology*, M. Hattingh *et al.*, Eds., Cham: Springer International Publishing, 2020, pp. 429–440.

[9] M. Moshagen and M. T. Thielsch, "Facets of visual aesthetics," *International journal of human-computer studies*, vol. 68, no. 10, pp. 689–709, 2010.

[10] M. Öder, Ş. Eraslan, and Y. Yeslida, "Automatically classifying familiar web users from eye-tracking data: A machine learning approach," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 30, no. 1, pp. 233–248, Jan. 1, 2022.

[11] M. Hassenzahl, M. Burmester, and F. Koller, "AttrakDiff: Ein Fragebogen zur Messung wahrgenommener hedonischer und pragmatischer Qualität," de, in *Mensch & Computer 2003: Interaktion in Bewegung*, G. Szwillus and J. Ziegler, Eds., Wiesbaden: Vieweg+Teubner Verlag, 2003, pp. 187–196.

[12] T. Ezer, M. Greiner, L. Grabinger, F. Hauser, and J. Mottok, "Eye tracking as technology in education: Data quality analysis and improvements," in *16th annual International Conference of Education, Research and Innovation*, ser. ICERI 2023, Seville, Spain: IATED, Nov. 2023, pp. 4500–4509.

[13] T. Ezer, L. Grabinger, F. Hauser, S. Staufer, and J. Mottok, "Eye tracking as technology in education: Further investigation of data quality and improvements," in *18th International Technology, Education and Development Conference*, ser. INTED 2024, Valencia, Spain: IATED, Mar. 2024, pp. 2955–2961.

[14] M. Kristen, F. Engl, and J. Mottok, "Enhancing phishing detection: An eye-tracking study on user interaction and oversights in phishing emails," in *SECURWARE 2024, The Eighteenth International Conference on Emerging Security Information, Systems and Technologies*, 2024.

[15] K. Holmqvist, *Eye Tracking: A Comprehensive Guide to Methods and Measures*. Oxford University Press, 2017, pp. 440–527.

[16] Y. Yin, M. P. McGuire, Y. Alqahtani, J. H. Feng, and J. Chakraborty, "Classification of information display types using graph neural networks," in *2023 International Conference on Computational Science and Computational Intelligence (CSCI)*, Dec. 2023, pp. 130–136.

[17] G. Kootstra, B. de Boer, and L. R. Schomaker, "Predicting eye fixations on complex visual stimuli using local symmetry," *Cognitive computation*, vol. 3, pp. 223–240, 2011.

[18] P. Blignaut, "Fixation identification: The optimum threshold for a dispersion algorithm," *Attention, Perception, & Psychophysics*, vol. 71, pp. 881–895, 2009.

[19] M. Millecamp, C. Conati, and K. Verbert, "Classifeye: Classification of personal characteristics based on eye tracking data in a recommender system interface," in *Joint Proceedings of the ACM IUI 2021 Workshops*, CEUR Workshop Proceedings, vol. 2903, 2021.

[20] A. Bulling, J. A. Ward, H. Gellersen, and G. Tröster, "Eye movement analysis for activity recognition using electrooculography," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 4, pp. 741–753, 2010.

[21] A. Jung, *Machine Learning: The Basics*. Springer Nature, 2022, pp. 21–25.

[22] M. A. Hall, "Correlation-based feature selection for discrete and numeric class machine learning," Ph.D. dissertation, University of Waikato, Department of Computer Science, 1999.

[23] D. Chicco and G. Jurman, "The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, no. 1, p. 6, Jan. 2020.

[24] D. Chicco, M. J. Warrens, and G. Jurman, "The matthews correlation coefficient (mcc) is more informative than cohen's kappa and brier score in binary classification assessment," *IEEE Access*, vol. 9, pp. 78 368–78 381, May 2021.

[25] D. J. Hand, P. Christen, and N. Kirielle, "F*: an interpretable transformation of the f-measure," *Machine Learning*, vol. 110, no. 3, pp. 451–456, 2021.

[26] M. Startsev and R. Zemblys, "Evaluating eye movement event detection: A review of the state of the art," *Behavior Research Methods*, vol. 55, no. 4, pp. 1653–1714, Jun. 2023.

[27] C. Albon, *Python Machine Learning Cookbook: Practical Solutions from Preprocessing to Deep Learning*. 2018, pp. 210–213.

[28] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," *Advances in Neural Information Processing Systems*, vol. 4, pp. 2951–2959, 2012. arXiv: 1206.2944.

[29] F. Nogueira, *Bayesian Optimization: Open source constrained global optimization tool for Python*, 2014.