Improving Prediction Accuracy of User Cognitive Abilities for User-Adaptive Narrative Visualizations

by

Alireza Iranpour

Prof: Cristina Conati

Course: CPSC 532C (Human-Centered AI)

THE UNIVERSITY OF BRITISH COLUMBIA

Department of Computer Science

Fall 2019

1. Introduction

Oftentimes, text and visualization are combined to describe complex information in a format known as Magazine Style Narrative Visualization (MSNV). However, working with two disparate sources of information, at the same time, can be somewhat challenging as attention will be split. Recent studies show that these difficulties are dependent on reader's level of cognitive abilities. As a result, predicting user cognitive skills can discover individual struggles and enable tailored support. This project would be an extension of a current research on this subject wherein eye-tracking data has been leveraged to predict user's cognitive abilities and task performance with the ultimate goal of providing personalized support with information visualization processing.

2. Related Work

2.1. Motivations

People with different cognitive abilities have different performance in processing different aspects of Information visualization. Hence, learning the level of each user's cognitive abilities can greatly empower personalized support. Given the perceptual nature of processing Information visualizations (InfoVis), eye movement patterns can reveal so much about user cognitive states and, therefore, can be leveraged to predict user cognitive abilities.

Cognitive abilities such as verbal working memory (VERWM), visualization literacy (VISLIT), and reading proficiency (READP) are used to identify how adaptation is performed while task performance measures such as speed and accuracy are used to determine when adaptation should take place.

Each cognitive ability determines the level of proficiency in processing particular aspects of narrative visualizations. Verbal working memory indicates ability in maintaining verbal information, so people with lower levels of this ability might benefit from simplification of the texts. Visual literacy, on the other hand, refers to the degree to which one can efficiently and confidently perceive data visualization, so people with low levels of this trait would appreciate visual cues that link corresponding elements in text and visualization. Last but not least, reading proficiency is the ability of processing labels and datapoints, so additional guidance would help those who are not proficient readers.

2.2. Solution

To gain insight into how eye movement patterns relate to cognitive abilities and processing performance, a study was conducted with 56 subjects wherein participants were asked to read 15 MSNV documents and after each respond to 3 comprehension questions to measure task accuracy and speed. The cognitive skills of each subject were also evaluated and scored in advance through a battery of well-established

psychological tests. During the study, as participants read MSNV documents, their eye movement patterns were tracked using a T120 remote eye-tracker. These raw data were then processed and represented in terms of the following features (feature set) over each of the 7 salient regions in a MSNV also known as areas of interest (AOIs): Gaze features including fixations (rate and duration) and saccades (duration, distance, and velocity), pupil size (width and dilation velocity), and head to display distance.

The results from the study including eye-tracking data and task performance together with cognitive ability scores were then used as training data to train a classification model that classifies cognitive abilities and performance measures of each user into high and low based on their eye-movement patterns. Predictions were made in two different ways. First, in order to learn how early predictions can be made in the absence of prior user information, the model would make a prediction every single second for 29 consecutive seconds within each task. Second, since user information can be logged from several interaction sessions, the usefulness of prior user data was examined by having the classification model make predictions at the end of each task based on accumulated data from all previous tasks. This time only cognitive abilities were predicted and performance measures were left out as performance is task dependent and is not a long-term state.

In this study, the following four classification algorithms were trained on the dataset and compared against a majority-class baseline: Logic Regression (LR), Random Forest (RF), Support Vector Machine (SVM), and Extreme Gradient Boosting (XGB). The evaluation of these algorithms was based upon 10-fold cross-validation over participants where highly correlated features were removed from the training folds in each iteration. To investigate the feasibility of predicting cognitive abilities, a one-way repeated measures ANOVA was run for each cognitive ability in both within and across task predictions. The main effect of classifier for each of the abilities was significant ($p < 0.01$). For within task predictions the best classifier for READP, VISLIT, and VERWM was LR, LR, and Base respective (table 1). However, for across task predictions, RF, SVM, and XGB had the best performance respectively (table 2). Finally, for task performance, RF was the best classifier for both accuracy and speed (table 3).

Table 1. Comparisons of the classifiers within task

Cognitive Ability	Ranking of Classifiers	Accuracy at Best Window
READP	**LR** > SVM > RF = XGB > Base	0.67
VISLIT	**LR** > Base = XGB > RF = SVM	0.58
VERWM	**Base** > SVM = LR > XGB = RF	-

Table 2. Comparisons of the classifiers across tasks

Cognitive Ability	Ranking of Classifiers	Accuracy at Best Window
READP	**RF** > SVM > XGB > LR > Base	0.67
VISLIT	**SVM** = **RF** > XGB = Base > LR	0.66
VERWM	**XGB** > RF > LR > SVM > Base	0.72

Table 3. Comparisons of the classifiers within task

User Performance	Ranking of Classifiers	Accuracy at Best Window
Accuracy	**RF** > XGB = LR > Base > SVM	0.67
Speed	**RF** = LR > XGB > SVM > Base	0.73

3. Problem

In this study, to avoid complexity, only 4 classification algorithms have been explored and evaluated. However, there are lots of other classifiers that might yield higher prediction accuracies. Moreover, the current classifiers have been assessed based on their default configurations and, therefore, did not reach and demonstrate their full potential. The main purpose of the study was to establish the feasibility of predicting cognitive abilities and performance based on eye-tracking data. As a result, not enough effort was put into fine-tuning the employed classifiers or exploring alternative ones.

4. Solutions

This project aims to extend the previous work on *User-Adaptive Narrative Visualizations* by focusing on the user modeling aspect and improving the prediction of user cognitive abilities during Information visualization processing. The primary goal of this project is to achieve higher prediction accuracies by fine tuning the currently employed classifiers as well as other viable classifiers that could possibly outperform the ones proposed by the paper.

4.1. Considerations

Optimization of hyper-parameters is a time consuming and computationally expensive process. In this regard, for each cognitive ability, the hyper-parameters of the proposed best classifier were fine-tuned based on the reported best window (window with highest predictive accuracy) found by the previous work. Moreover, the main focus was placed on the across tasks predictions where user cognitive abilities were predicted at the end of each task using accumulated eye-tracking data from all previous tasks as well as the current one. This was done because of several reasons. First of all, this form of prediction, as indicated by the results of the previous work, offers better performance than within task predictions. Although for reading proficiency, both yielded similar accuracies, for visual literacy and verbal working memory, across

tasks is clearly the better choice. Specifically, for verbal working memory, the baseline had the highest within task prediction accuracy. Second, given that user cognitive abilities are long-term states and that interaction data are often logged, it makes more sense to involve user interaction history in the prediction process. Finally, the dataset used for training the classifiers in the within task predictions contained 795 samples (15 samples for each of the 53 participants) for each of the 29 windows while in the across tasks predictions, only 53 samples (one sample for each participant) were used for each of the 15 windows. As a result, preforming the tuning process in the across tasks predictions would be more computationally tractable. For the above-mentioned reasons, in this project, classifiers have been tuned in the across tasks setting. However, the proposed solution can be easily applied to the within task setting as well.

4.2. Challenges

4.2.1. Mismatch between Evaluation Methods

In the previous work, classification accuracy was evaluated based on a repeated 10-fold cross-validation. Repetition was used to minimize the effect of lucky splits. However, when tuning the hyper-parameters using a nested structure, repetition would disarrange the test sets used in the outer loop and render the evaluation results for the selected hyper-parameters meaningless. In this regard, in order to resolve the mismatch between the evaluation methods and make for a just comparison, the proposed solution was to use leave-one-out cross-validation for the outer loop of the nested structure as well as for re-evaluating the default models from the previous work. Leave-one-out cross-validation completely eliminates the element of luck in the data splits as it places only one participant in each of the folds and, therefore, provides a more reliable approximation of the test error. Moreover, given the small size of the dataset, which consists of only 53 samples, Leave-one-out cross-validation leaves more of the data available for the classifiers to train on since it merely holds out one sample for testing. Finally, on top of providing better training quality and more stability, for this specific dataset, leave-one-out cross-validation is also computationally cheaper than repeating making it a far better evaluation approach than the one taken in the previous work.

4.2.2. High Dimensionality of Hyper-parameters

Some classification algorithms have multiple hyper-parameters to be set. In cases like this, searching over all combinations of possible values would be extremely expensive and often intractable. For instance, the classification algorithm, xgbTree, has 7 different hyper-parameters. That mean trying only 3 values for each of the hyper-parameters would result in 2,187 (3^7) possible models to be separately trained and evaluated. However, not all hyper-parameters have the same level of influence on the model's accuracy. In that regard, blindly tweaking all the values would be inefficient and ineffective. Hence, the proposed solution was to try more values for the hyper-parameters with highest expected influence on accuracy while keeping other

less influential ones fixed at default values. In order to identify the hyper-parameters to which the model's accuracy was most sensitive, empirical analysis was used based on the dataset for each cognitive ability. This approach, allows to come across and find better models both faster and cheaper.

4.3. Tuning Pipeline

In the previous work, the employed classifiers were trained and evaluated at their default configurations and no effort was made towards optimizing the classifiers to achieve the highest possible accuracy. In this section, I will describe the proposed tuning pipeline that was utilized to find the optimum values for the hyper-parameters as well as to provide an unbiased approximation of the generalization accuracy.

Trying a variety of hyper-parameter values on the whole dataset in an effort to find the best combination would make the results subject to optimization bias since the same dataset that is used for optimization is also used for evaluation. To resolve this bias and to provide a realistic approximation of the accuracy that the model would achieve on new and unseen data, a nested structure was employed where tuning is performed in the inner loop while evaluation takes place in the outer loop. As mentioned before, to eliminate the possibility of lucky groupings of the samples, for the outer iterations, a leave-one-out cross-validation was used to evaluate the selected optimum model for every single participant. However, in the inner iterations, 10-fold cross-validation was employed to evaluate candidate models in search for the best one. In the outer loop, at each iteration, one sample would be reserved for testing while the rest is used for tuning. Typical nested cross-validation is mostly used to narrow down the number of viable models and does not provide a single best as it selects a different model at each inner cross-validation. However, the proposed pipeline is designed to find the single best model for each cognitive ability. In order to this, at each outer iteration, rather than reporting the most accurate model as the best, the performance of each candidate model is evaluated and stored, and this procedure is performed for all 53 outer iterations. Next, the average accuracy is calculated for each candidate model, and the one with the highest is selected. Finally, the selected model is evaluated on the reserved test samples to provide an unbiased approximation of the accuracy that the model would likely achieve on new data.

5. Results

In this section, for each cognitive ability, I will discuss the re-evaluation results of the model used in the previous work as well as the results achieved by tuning the same model. Moreover, the best alternative models and the results of the tuned version will be discussed.

The best classifier for each cognitive ability, proposed by the previous work, was re-evaluated using leave-one-out cross-validation to allow for a fair comparison with the tuned version. Table 4 shows the results of this re-evaluation as well as the best window at which each classifier was re-assessed.

Table 4. Re-evaluation of the proposed classifiers across tasks

Cognitive Ability	Classifier	Window	Acc Overall	Acc Low	Acc High
READP	RF	4 tasks	0.66	0.65	0.67
VISLIT	RF	3 tasks	0.66	0.75	0.56
VERWM	XGB	10 tasks	0.70	0.73	0.67

5.1. Reading Proficiency

The first cognitive ability is reading proficiency which is the ability of processing labels and datapoints. In the previous work, for this cognitive ability, among the 4 candidate classifiers, Random Forest (RF) was proposed as the best performing classifier. The results of the re-evaluation are shown in table 4.

Random Forest has only one hyper-parameter "*mtry*" which determines the number of randomly selected predictors (features). The default value for this hyper-parameter is the square root of the total number of features. However, choosing this value may not always result in the highest performance. Hence, a grid search was performed on this hyper-parameter to find an optimum value. For reading proficiency, RF was tuned based on the best window reported by the previous work which consisted of eye-tracking data collected across the first 4 tasks. The possible range for this hyper-parameter was 1 to 167, so 10 different values in this range were randomly selected, and for each, a separate RF model was created. Figure 1 shows the average accuracy for each of the 10 candidate RF models. Increasing the number of predictors raises the accuracy of the algorithm. However, it reaches its pinnacle at 75. After that point, more predictors create redundancy and simply add noise.

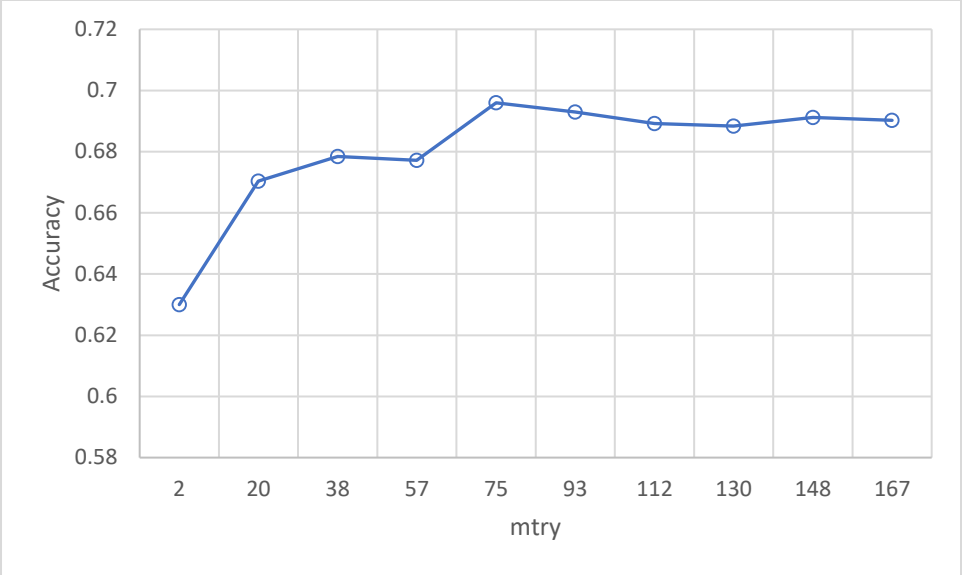

Figure 1: Random Forest Search Grid

Clearly, for this particular dataset, 75 seems to be the optimum value for the hyper-parameter of interest. Next, in order to approximate the performance of the selected model on new data, at each iteration of the outer loop, the model was retrained on the training set (52 participants) and then evaluated on the unseen left-out sample. Table 5 shows the generalization results of the selected model (tuned RF) and compares them with those of the default model. Tuning the classifier improved the overall accuracy, accuracy low, and accuracy high by 9%, 12.3%, and 4.5%, respectively. Moreover, In the default model, prediction accuracy for the low class was lower than that of the high class. However, tuning the classifier, not only improved the accuracy for both but it also did so even more for the low class which is important since those are the people who need adaptation the most.

Table 5. Comparison of the default and the tuned RF (READP)

Model	**Acc Overall** (SD)	**Acc Low** (SD)	**Acc High** (SD)
RF (default)	0.66 (0.478)	0.65 (0.485)	0.67 (0.480)
RF (tuned)	0.72 (0.455)	0.73 (0.452)	0.70 (0.465)

There were over 200 classification algorithms available in the Caret package in R. However, many of them were slightly altered algorithms from the same family of classifiers. In this regard, from each family of classifiers, the most viable model was tried on the dataset to estimate the potential accuracy achievable. Among the classification types that were tried on the dataset associated with reading proficiency, those based on decision trees demonstrated higher potential. As a result, in addition to RF, xgbTree was also considered for tuning. This classifier had 7 different hyper-parameters, and tweaking all of them would be

computationally expensive. Among these 7 hyper-parameters, 2 were kept at their default values while a random search was performed on the rest. Table 6 shows the results of the classifier after being tuned in comparison with the default RF proposed in the previous work. The overall accuracy, accuracy low, and accuracy high were respectively improved by 13.6%, 18.5%, and 10.4%. Although, xgbTree was previously considered for predicting reading proficiency, it was not set for maximum performance, thus RF was proposed as the better classifier. However, after being tuned, it outperformed both the default and the tuned version of the Random Forest making it a better choice for this cognitive ability.

Table 6. Comparison of the default RF and the tuned xgbTree model (READP)

Model	**Acc Overall** (SD)	**Acc Low** (SD)	**Acc High** (SD)
RF (default)	0.66 (0.478)	0.65 (0.485)	0.67 (0.480)
XGB (tuned)	0.75 (0.434)	0.77 (0.430)	0.74 (0.447)

Figure 2 demonstrates the performance of these models against the baseline across tasks for different windows. In addition to the best window (4 tasks), xgbTree performed slightly better also in the first two windows. Thus, depending on the number of tasks for which the system has collected eye-tracking data, this plot can be used to choose the right model. Nevertheless, the two classifiers have only been tuned based on the best window. In that regard, each classifier could also be tuned based on the first 3 windows to enable earlier predictions with even higher accuracies.

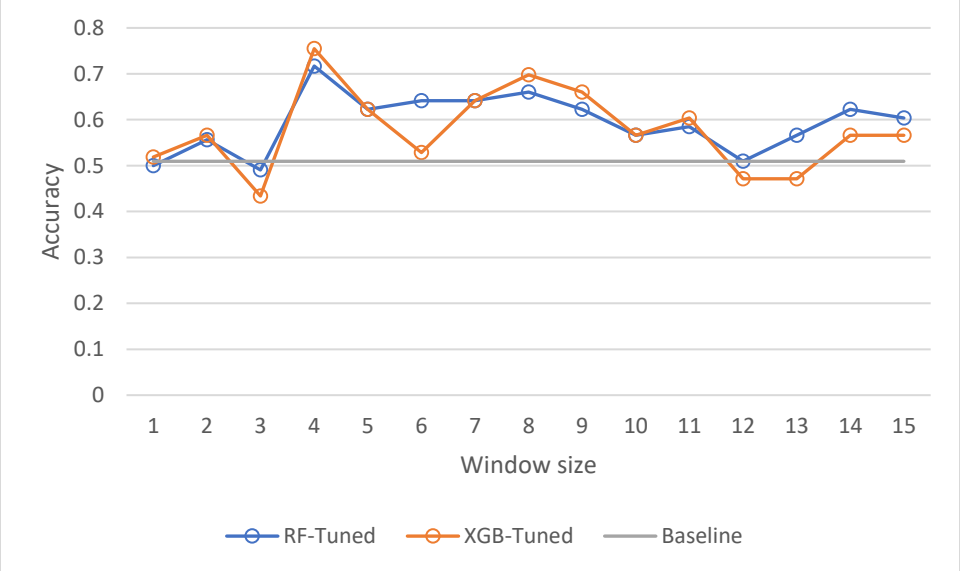

Figure 2: Comparison of the modified models and the baseline across tasks

5.2. Visual Literacy

The second cognitive ability is visual literacy which refers to the degree to which one can efficiently and confidently perceive data visualization (bar charts). For this cognitive ability, the proposed best classifier, was also Random Forest. The results of the re-evaluation are shown in table 4. Once again, in order to tune the classifier 10 different models were created to find the optimum value for the hyperparameter. As can be seen in figure 3, the value 21 yielded the highest average accuracy and was thus selected as the best model.

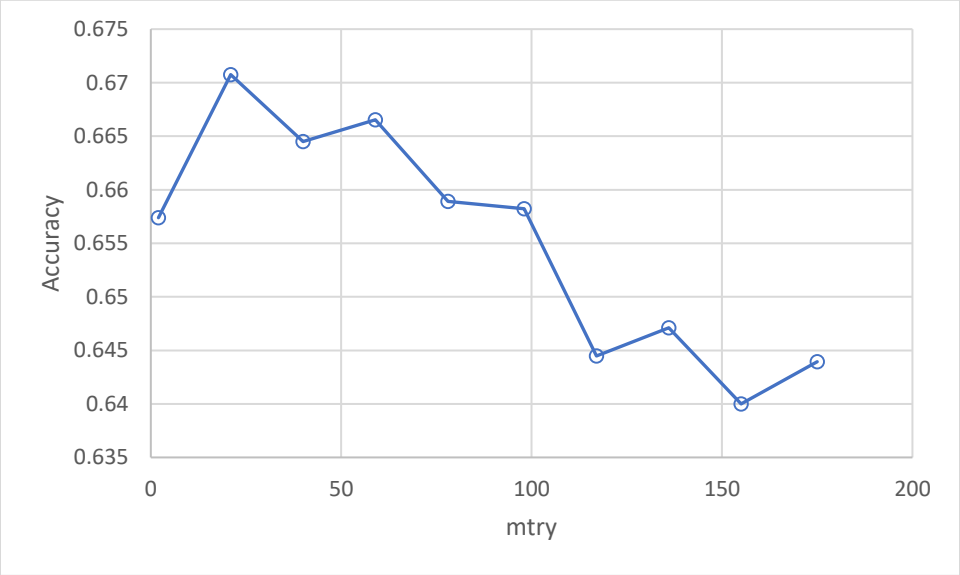

Figure 3: Random Forest Search Grid

Table 7 shows the generalization results of the selected model (tuned RF) and compares them with those of the default model. For visual literacy, tuning the classifier improved the overall accuracy, accuracy low, and accuracy high by 6.1%, 5.3%, and 7.1%, respectively.

Table 7. Comparison of the default and the tuned RF (VISLIT)

Model	**Acc Overall** (SD)	**Acc Low** (SD)	**Acc High** (SD)
RF (default)	0.66 (0.478)	0.75 (0.441)	0.56 (0.507)
RF (tuned)	0.70 (0.463)	0.79 (0.418)	0.60 (0.500)

Among the alternative classifiers that were tried on this dataset, Boosted Linear Model (BstLm) was able to achieve an even higher accuracy. BstLm had 2 hyper-parameters, namely mstop and nu which were, respectively, the number of boosting iterations and shrinkage. Based on empirical evidence, changing the value of shrinkage had almost no effect on the accuracy and, as a result, was kept constant at a default value

of 0.1. For the other hyper-parameter, 10 different values were tried, and, as can be seen in figure 4, 150 was discovered to be the optimum number of boosting iterations.

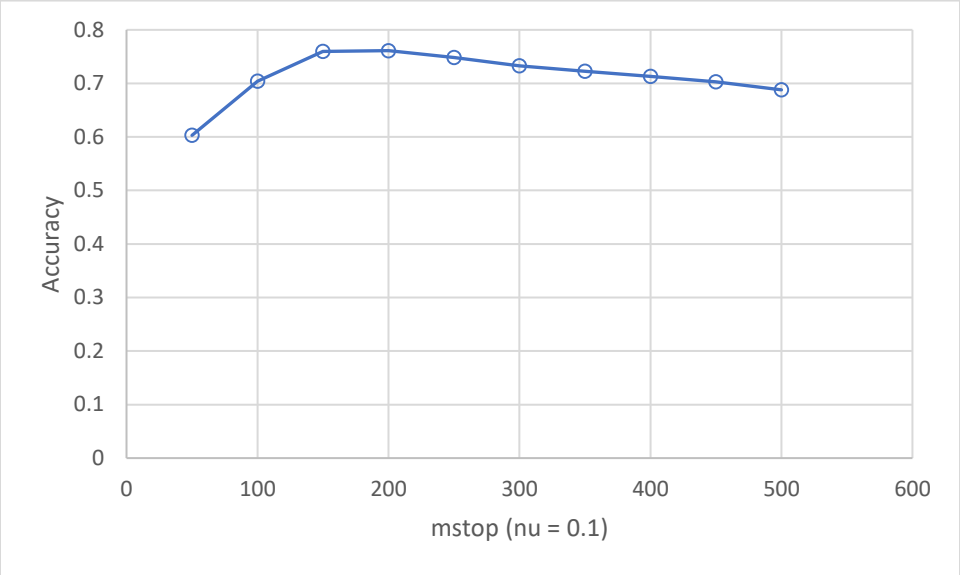

Figure 4: BstLm Search Grid

Table 8 shows the generalization results of the selected model (tuned BstLm) and compares them with those of the proposed default model. After being tuned, BstLm was able to improved the overall accuracy, accuracy low, and accuracy high by 12.1%, 5.3%, and 21.4%, respectively. Even though the low class is the one we are most concerned with, the default RF model was having a very poor performance on the high class, and the relatively high overall accuracy was mostly due to the high accuracy of the low class. Even tuning this classifier did not address the issue. However, with the BstLm, not only a higher overall accuracy was achieved but a better balance was also reached between the accuracies of the two classes.

Table 8. Comparison of the default RF and the tuned BstLm model (VISLIT)

Model	**Acc Overall** (SD)	**Acc Low** (SD)	**Acc High** (SD)
RF (default)	0.66 (0.478)	0.75 (0.441)	0.56 (0.507)
BstLm (tuned)	0.74 (0.445)	0.79 (0.418)	0.68 (0.476)

Figure 5 illustrates the performance of these models against the baseline for different windows across tasks. In addition to the best window (3 tasks), BstLm had better performance for the first window as well. According to the plot, collecting eye-tracking data after the first 3 tasks adds noise to the data and adversely affects the prediction accuracy.

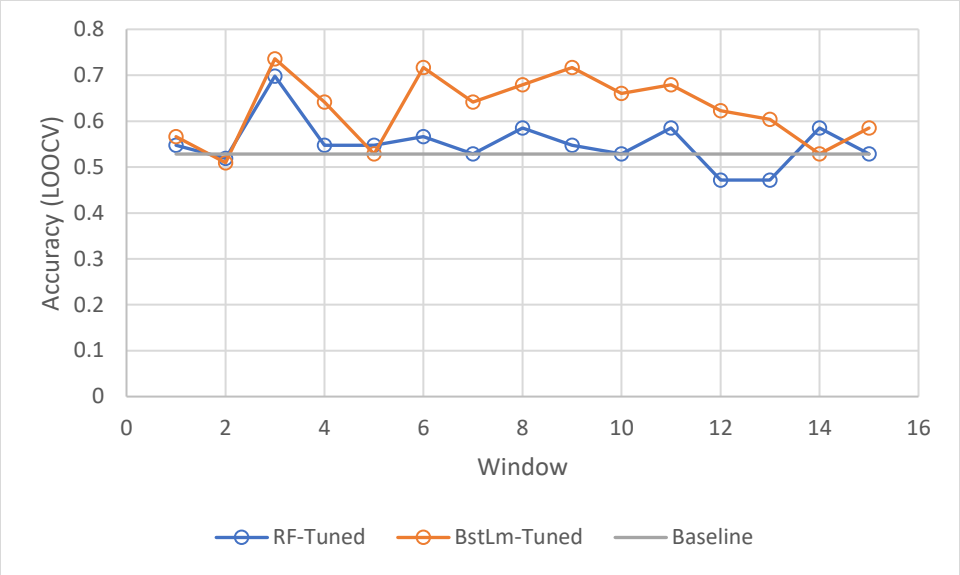

Figure 5: Comparison of the modified models and the baseline across tasks

5.3. Verbal Working Memory

The last cognitive ability is verbal working memory which indicates a person's ability in maintaining verbal information. In the previous work, xgbTree was proposed as the best classifier for this cognitive ability. The results of the re-evaluation are shown in table 4. In order to tune this classifier, two of the hyper-parameters remained unchanged while a grid search was performed on the other 5. Tuning this classifier raised the overall accuracy and accuracy high by 12.9% and 4.5%, respectively. More importantly, a considerable accuracy of 0.88 was achieved for the low class (20.5% improvement). Table 9 shows the generalization results of the selected model (tuned xgbTree) and compares them with those of the proposed default model.

Table 9. Comparison of the default and the tuned XGB (VERWM)

Model	**Acc Overall** (SD)	**Acc Low** (SD)	**Acc High** (SD)
XGB (default)	0.70 (0.463)	0.73 (0.452)	0.67 (0.480)
XGB (tuned)	0.79 (0.409)	0.88 (0.326)	0.70 (0.465)

Among the other available classifiers that were explored, none was able to come close to this accuracy. Figure 6 illustrates the performance of the tuned xgbTree against the baseline for different windows across tasks. To attain the highest level of accuracy, it is best to make predictions after collecting data from the first 10 tasks. However, an overall accuracy of 0.74 can also be achieved at the second window which is still higher than that of the default model at the tenth window. In other words, predictions can still be made with an accuracy of 0.74 as early as two tasks until the user has performed 10.

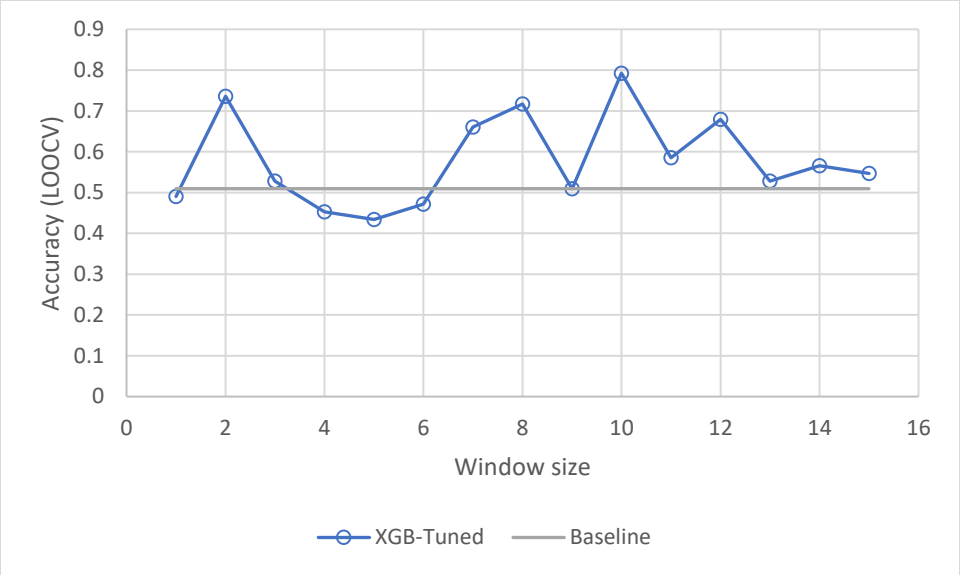

Figure 6: Comparison of the modified model and the baseline across tasks

6. Statistical Analysis

In order to determine the statistical significance of the achieved improvements, a paired samples t-test was conducted for each of the classification models. A one-sided test was used since the hypothesis was that tuning would improve accuracy. Table 10 shows the results for the tuned RF. According to the analysis, the improvements were found to be statistically non-significant. However, the effect size was large for the overall accuracy as well as for the high and low classes suggesting that the study should be repeated with a larger sample size.

Table 10. Comparison of the default and the tuned RF (READP)

Accuracy	t	p	r
Acc	1.14	0.130	0.70
Acc Low	1	0.163	0.65
Acc High	0.57	0.287	0.75

Table 11 shows the results for the tuned xgbTree. The improvements achieved by this model were also statistically non-significant. The effect size for the overall accuracy, accuracy low, and accuracy high were medium, large, and small, respectively.

Table 11. Comparison of the default RF and the tuned XGB (READP)

Accuracy	t	p	r
Acc	1.40	0.084	0.42
Acc Low	1.36	0.092	0.56
Acc High	0.70	0.245	0.30

For visual literacy, the improvements made by tuning the default RF model were found to be statistically insignificant (table 12). However, the large effect size indicates the need for another study with a larger sample size.

Table 12. Comparison of the default and the tuned RF (VISLIT)

Accuracy	t	p	r
Acc	1.43	0.080	0.92
Acc Low	1	0.163	0.90
Acc High	1	0.164	0.92

The higher accuracies achieved by the alternative model, BstLm, also failed to demonstrate statistical significance (table 13). Nevertheless, for the high class, there was a large effect size with a relatively small p-value.

Table 13. Comparison of the default RF and the tuned BstLm (VISLIT)

Accuracy	t	p	r
Acc	1.16	0.126	0.47
Acc Low	0.37	0.356	0.30
Acc High	1.36	0.093	0.60

As opposed to the previous cognitive abilities, for verbal working memory, tuning the proposed classifier did result in statistically significant improvements, specifically for the overall and the low-class accuracies (table 14). In addition, the effect size was large.

Table 14. Comparison of the default and the tuned XGB (VERWM)

Accuracy	t	p	r
Acc	2.33	0.012	0.78
Acc Low	2.13	0.022	0.60
Acc High	1	0.163	0.92

7. Conclusion

This project focused on extending the work of a current research on user-adaptive narrative visualizations where eye-tracking is leveraged to predict user cognitive abilities and task performance with the ultimate goal of providing tailored support with information visualization processing. Based on the findings of the previous work, which established the feasibility of predicting user cognitive abilities from eye-tracking data, this work attempted to improve the prediction accuracy of these cognitive properties by (a) fine tuning

the classification algorithms and (b) exploring further classification algorithms. The fine-tuning process was performed using a modified nested cross-validation structure which would optimize the hyper-parameter values for each classier as well as report the generalization accuracy of the tuned model on new data. For reading proficiency and visual literacy, tuning the proposed classifiers did result in improved accuracy. In fact, for reading proficiency, tuning the proposed classifier made the model more accurate for the low-class which was previously less accurate than the high-class. This is important as the low-class is of higher concern. Moreover, tuning alternative classifiers for each of the mentioned abilities lead to even higher prediction accuracies. Specifically, for visual literacy, the alternative model, BstLm, not only improved the overall accuracy, but it also addressed the large accuracy difference between the low and the high class. Finally, for verbal working memory, tuning the proposed classifier resulted in substantially higher accuracies. In addition, the tuned model was able to provide even more accurate predictions after only 2 tasks (0.74) than the default model did after 10 (0.70). For the first two cognitive abilities, namely reading proficiency and visual literacy, the results of the analysis revealed no statistical significance. However, the large size of the effects suggests the need for repeating the study with a larger sample size. On the other hand, for verbal working memory, the improvements in accuracy were found to be statistically significant with large effects.

8. Future Work

In this project, for each cognitive ability, the proposed and the alternative classifiers were fine-tuned based on the best window reported by the previous work. In this regard, as future work, these classifiers could also be tuned based on the rest of the windows to assess the possibility of earlier and more accurate predictions. Moreover, due to having higher practical potential and computational tractability, this work was mainly concentrated on the across tasks predictions. However, as future work, the proposed tuning pipeline can just as easily be applied to the within task predictions as well.

9. References

Eye-Tracking to Predict User Cognitive Abilities and Performance for User-Adaptive Narrative Visualizations