Cognitive Workload Detection Using ML & Eye-tracking Data

A. Introduction

This task aims to identify the important components and actions needed to answer the research questions. To successfully classify cognitive interference using fixation and saccades features, involves defining the required resources, discussing appropriate research methodologies, and designing experiments. Additionally, ethical issues like participant privacy and data confidentiality are covered. Significant standards have also been established to monitor progress and ensure the timely conclusion of the research project.

B. Research Progress and Planning

1. Resources Required

a. Data

- Access to existing eye-tracking datasets containing fixation and saccades-related data.
- Access to databases and academic journals such as IEEE Xplore, Deakin Library, Research Gate, etc for data hunting.

b. Software

- For machine learning algorithms, Python packages like Scikit-learn.
- For visualization, Matplotlb and Seaborn are considered.
- For development, IDEs such as Jupyter Notebook or Google Colab are used.

c. Hardware

- High-performance processing power CPU/GPU for data processing and model training.

2. Relevant Research methods

In this study, eye-tracking data is used to characterize task-specific cognitive interference using a range of machine-learning models and approaches. To guarantee reliable performance and accurate categorization, the subsequent methods are applied-

a. Machine learning models

- Machine learning models such as Random Forest, Support Vector Machine (SVM), Logistic Regression, and ANN are utilized to categorize cognitive interference.

b. Feature sets

The following feature sets are used for classification

- Fixation features- These are characteristics that relate to the quantity of fixations, average duration of fixations, and regressions of fixations. During tasks, these characteristics improve in capturing focused attention and cognitive processing.
- Saccades features- Saccade frequency, length, velocity, and amplitude are examples of the rapid eye movements that occur between fixations. These aspects provide insight into movement related to workload and visual exploration.
- Fixation+Saccades features- To produce a more accurate representation of cognitive effort, fixation, and saccades properties are combined. To further ensure that the model concentrates on task-related differences rather than individual variability, normalization is also used within each subject to minimize variability.
- Fixation+Saccades Normalization- The mean of every task is subtracted to normalize the data for each subject. By doing this, it is ensured that task-specific properties are the focus of the classification models rather than inter-subject inequalities. This procedure improves generalization across subjects and streamlines task comparisons.

c. Multi-Class Classification

- In multi-class classification, a more detailed knowledge of the cognitive processes underlying various tasks is made possible by training each classifier (e.g., Random Forest, SVM) to distinguish between three or more task-specific cognitive effort categories.

d. Cross-validation

Five-fold cross-validation is utilized to make sure the models generalize effectively and offer objective performance estimates-

- Five folds are created from the dataset; four of the folds are used for training and the fifth fold is used for testing. Each fold serves as the test set once during the five repetitions of this operation.
- By assessing the model on several subsets of the data, cross-validation helps prevent overfitting by guaranteeing that the reported accuracy, F1-score, and other performance metrics are reliable and indicative of the model's performance on unobserved data.
- Due to its ability to maximize the use of available data for both training and evaluation, cross-validation is especially crucial in the case of small datasets.

To ensure a comprehensive and rigorous approach to answering the research questions, these machine learning models are combined with cross-validation. The study tries to determine which models and features, based on eye-tracking data, best represent task-specific cognitive workload by utilizing several feature sets and multi-class categorization.

3. Experiment Design and Dataset Requirement

Overview-

a. The study provides insights into how external distractions affect cognitive load and attention by examining eye-tracking data under both interference and non-interference scenarios. Given that task complexity varies in real-world applications, cognitive interference plays a significant role in workload estimating models.

b. To detect cognitive interference in tasks under different circumstances, the study looks into ways to make use of eye-tracking data. The classifiers conduct experiments to simulate how eye movement behavior is influenced by cognitive load and how different tasks (reading vs. naming) or task conditions (with/without interruption) affect it.

c. The capacity of distinguishing between these tasks indicates that specific patterns in eye movements can be induced by various task types or cognitive situations (interference vs. no interference).

Subsequent experiments aim to methodically assess the effects of different feature sets (fixation, saccades, and combined data(normalized) on the efficiency of machine learning models that categorize task-specific cognitive workload using eye-tracking data. The experiments try to find which model and feature set combination performs the best at differentiating between tasks by training and evaluating several classifiers.

Experiment 1- Classification of task-specific Cognitive Workload based Eye-tracking Data

This experiment aims to evaluate the optimal machine learning model for task-specific cognitive burden classification. We will evaluate how well various classifiers Random Forest, Logistic Regression, Support Vector Machine (SVM), and Artificial Neural Networks (ANN) perform when trained on a combination of eye-tracking features. The goal is to identify the model with the best F1 score and classification accuracy.

Tasks-

a. Using eye-tracking features such as fixation and saccades, perform classification to train the machine learning models such as Random Forest, Logistic Regression, Support Vector Machine, and Artificial Neural Network.

b. For each model, evaluate the accuracy, f1-score, and confusion matrix using 5-fold cross-validation. Next, compare the evaluation performance of all the models and determine the best model to classify the task-specific cognitive workload.

By using eye-tracking data to characterize task-specific cognitive demands, this experiment will identify the machine learning model that performs best in this regard. To capture the complex aspects of cognitive effort, the comparison will assist evaluate whether more complicated models, such as Random Forest and ANN, are needed or if simpler models, such as Logistic Regression, are sufficient.

Experiment 2- Classification Using Fixation Data

The goal of this experiment is to determine how well fixation-related variables can predict the cognitive burden associated with different tasks.
The fixation data offers information about the user's attentional focus areas and fixation durations, both of which are correlated with cognitive load.

Tasks-

a. Fixation features- Utilise the seven dependent features described by Rizzo et al. (2022)-
   i. n_fix- Number of fixations,
   ii. fix_mean, fix_max- Average and maximum fixation length.
   iii. norm_fix_mean, norm_fix_max- Normalized fixation length
   iv. x_regression, y_regression- Horizontal and vertical regressions.

b. Training the classifiers- Using this feature set, train the four classifiers- Random Forest, Logistic Regression, SVM, and ANN. The main task is to classify the cognitive workload according to tasks (NamingWITH/without interference, ReadingWITH/without interference).

c. Evaluation- To ensure reliable performance estimations, measure each classifier's accuracy and F1-score using 5-fold cross-validation.

Experiment 3- Classification Using Saccades Data

To gain insight into patterns of visual exploration, this experiment attempts to categorize cognitive workload solely based on saccades data, which records the quick eye movements between fixations.

Tasks-

a. Saccades features- Utilise the 22 dependent saccades features described by Rizzo et al. (2022)-
   i. up_freq, down_freq, left_freq, right_freq- Up/Down/Left/Right Frequency.
   ii. min_duration, avg_duration, max_duration- Minimum/Average/Maximum saccade duration.
   iii. min_vel, avg_vel, max_vel- Minimum/Average/Maximum saccade velocity.
   iv. min_ampl, avg_ampl, max_ampl- Minimum/Average/Maximum saccade amplitude.

v. min_angle, ave_angle, max_angle- Minimum/Average/Maximum saccade angle.
vi. min_distance, avg_distance, max_distance- Minimum/Average/Maximum saccade distance.
vii. min_slope, avg_slope, max_slope- Minimum/Average/Maximum saccade slope.

b. Training the classifiers- Same as Experiment 2, use the saccade-related feature se to train the four classifiers- Random Forest, Logistic Regression, SVM, and ANN with the same task classification labels

c. Evaluation- To ensure reliable performance estimations, measure each classifier's performance using 5-fold cross-validation.

Experiment 4 - Classification Using Combined Data (Fixation + Saccades Normalized)

This experiment attempts to categorize cognitive workload based on a combination of features (fixation+ saccades) data, which is also normalized to handle inter-subject differences.

Tasks-

a. Combined features- Use the features in Experiments 2(fixation) and Experiment 3(saccades).

b. Normalization- Apply normalization by deducting the mean of every variable in each subject's data across all tasks, following Rizzo et al. (2022). By reducing individual variability, this step guarantees that the focus of the classifiers are task-related patterns, rather than individual variations in gaze behavior,

c. Training the classifiers- Same as Experiment 2, use the combined feature set to train the four classifiers- Random Forest, Logistic Regression, SVM, and ANN with the same task classification labels

d. Evaluation- To ensure reliable performance estimations, measure each classifier's performance using 5-fold cross-validation.

Experiment 5- Evaluation of Classifier Performance for Feature Sets

In this experiment, three feature sets, fixation only, saccades, and combined normalized features are compared for classifier performance using Random Forest, Logistic regression, SVM, and ANN. The main goal is to observe how these different feature sets impact the performance of the ML models. We will compare the parameters to evaluate the overall performance of the classifiers.

Data Requirements

To perform the above experiments, the following public eye-tracking raw data is used. Rizzo et al. (2022)
https://tinyurl.com/EyeTrackData

a. A total of 64 university students participated in the experimental procedures. During the test, the participants had to perform two tasks- Naming and Reading. These tasks were both divided into two conditions- 'With Interference' and 'Without Interference'. Therefore, the four generated stimuli were composed by-
   1. Reading Without Interference(RWoI)
   2. Reading With Interference(RWI)
   3. Naming Without Interference(NWoI)
   4. Naming With Interference(NWI) Rizzo et al. (2022)

b. This dataset includes saccades and fixation data from several people engaged in cognitive workload activities that are task-specific (NamingWITHinterference, NamingWITHOUTinterference, ReadingWITHinterference, and ReadingWITHOUTinterference).

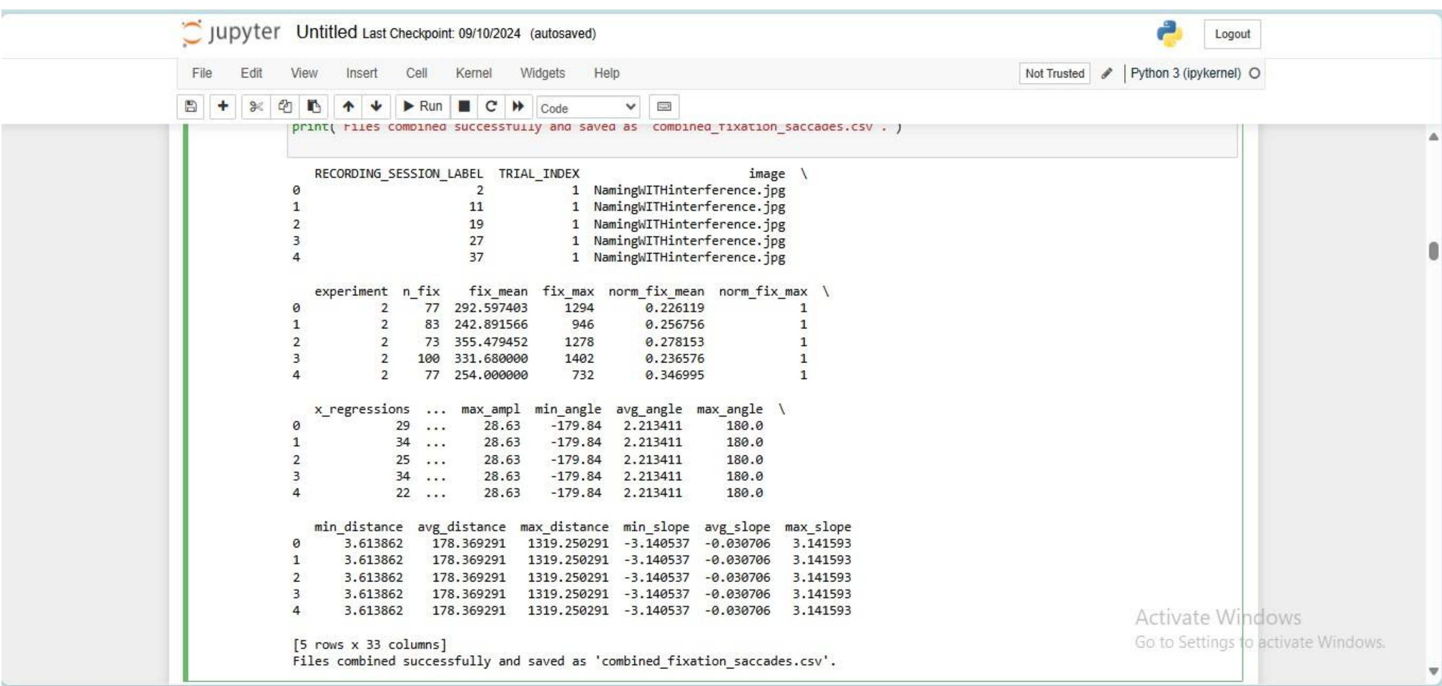
4. Implementation

a. Data collection and feature engineering

For machine learning, the eye-tracking data must be processed and prepared from the raw data available. Features' quality and relevance have a big influence on how well classifiers function, which is why feature engineering is so important. Three feature sets a combination feature set, saccades-only, and fixation-only will be the main emphasis of this study.

Steps to create the dataset-

1. The raw data has several files of fixation and saccades separately of each participant with each of the four experiments conducted.
2. Combine the files of each feature such as fixation and saccades into a CSV file.
3. Calculate the 7 fixation dependent variables and 22 saccades-related dependent variables.
4. Lastly, concatenating fixation and saccades dependent variables into a ‘combined_fixation_saccades.csv’ file based on grouping ‘experiment’, ‘’TRIAL_INDEX’, and ‘image’.
5. Therefore, ‘combined_fixation_saccades.csv’ is a preprocessed dataset with fixation and saccades features which can be used for model training.

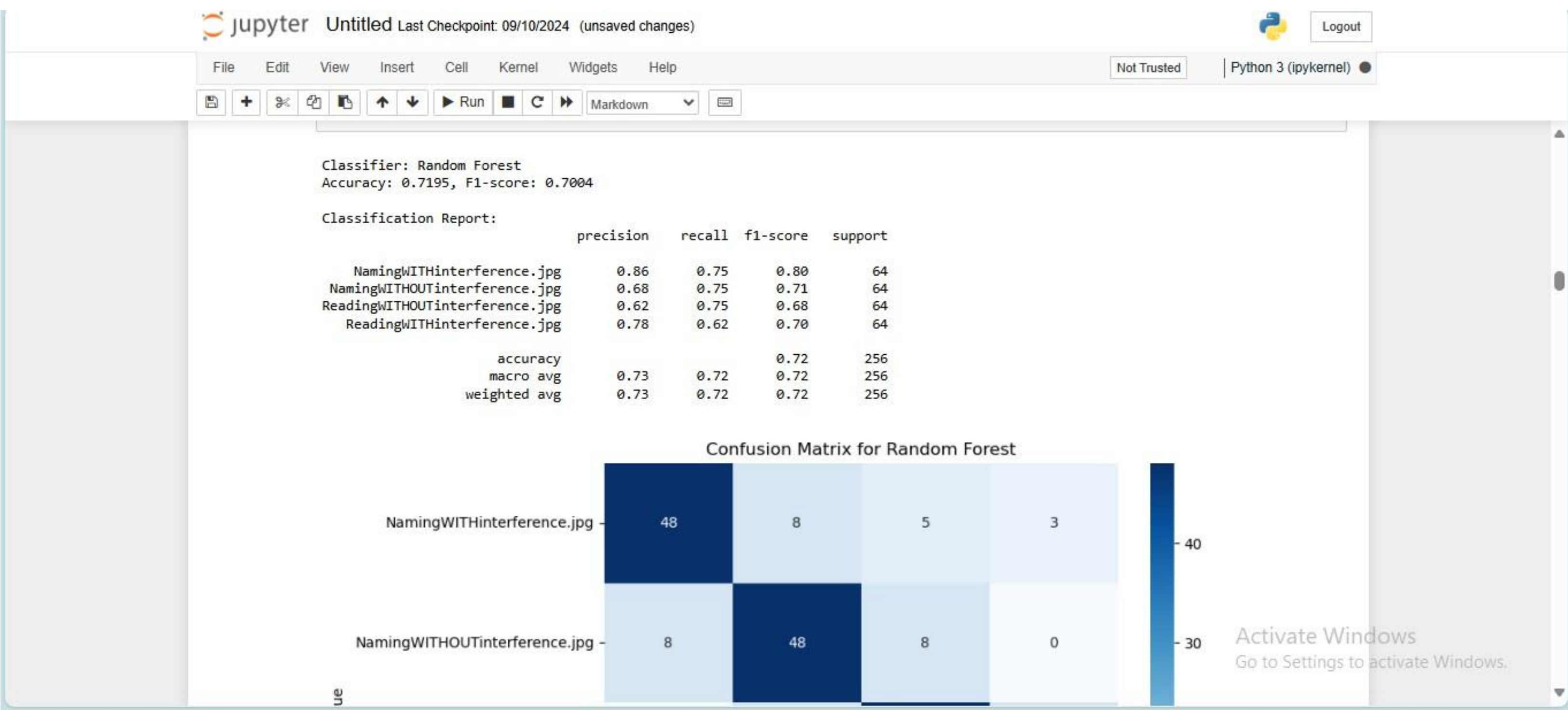


b. Classification and Data Analysis -
Now, we have the dataset with the required features, therefore, in this step, we will conduct classification experiments using these features.

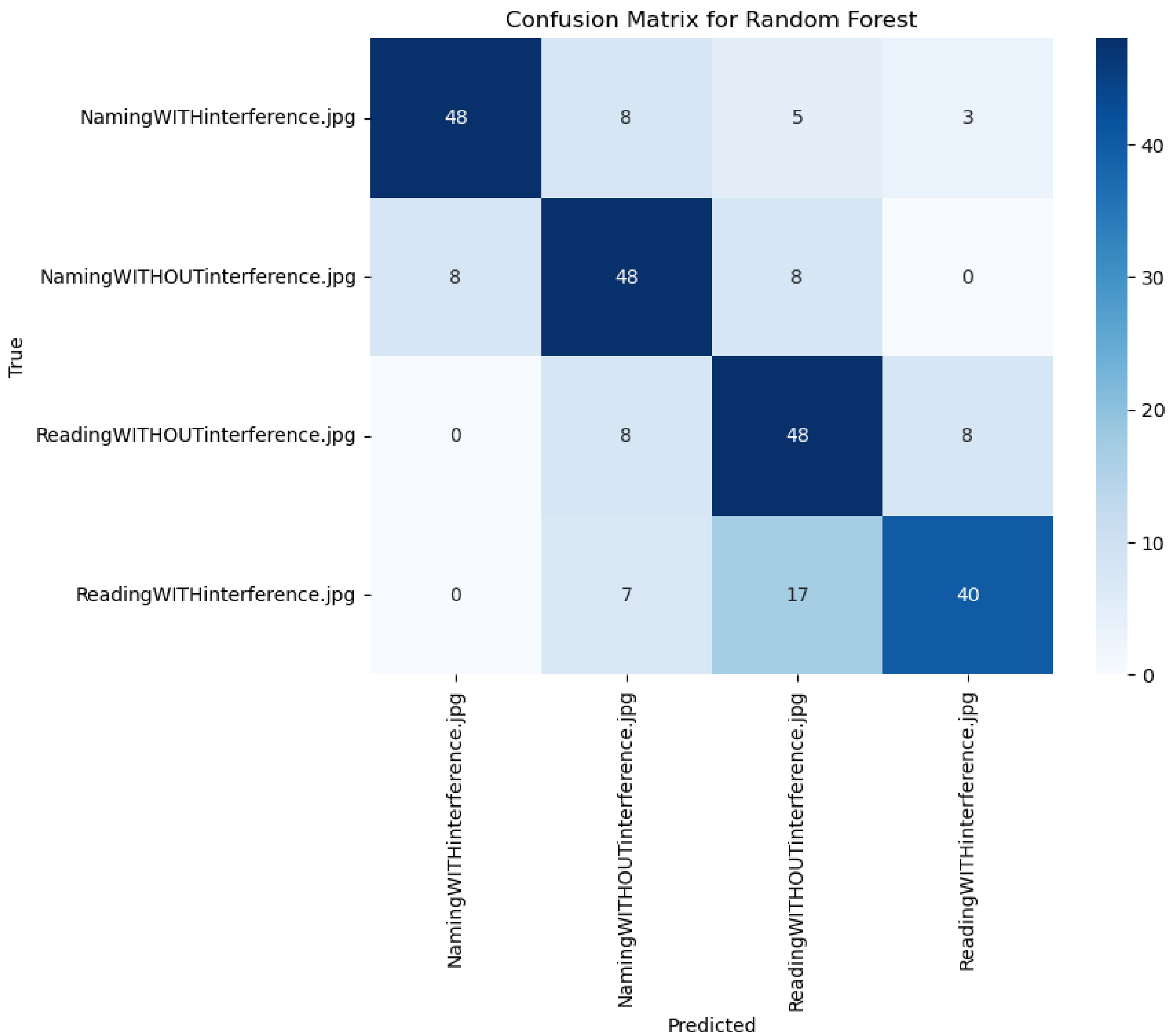
1. Classification of task-specific Cognitive Workload based Eye-tracking Data-
    a. In this classification, machine learning models such as Random Forest, Logistic Regression, SVM, and ANN are used to perform training and testing procedures using the eye-tracking data.
    b. Here we perform multi-class classification using 4 classes, NamingWITHInterference, NamingWITHOUTInterference, ReadingWITHOUTInterference, ReadingWITHInterference.
    c. 5-fold cross-validation is used to get a robust evaluation of each model's performance. To evaluate the performance, and accuracy, the F1-score and confusion matrix are calculated.

Evaluation of each model's performance-

1. Random Forest
   Accuracy - 0.71
   F1-score - 0.70



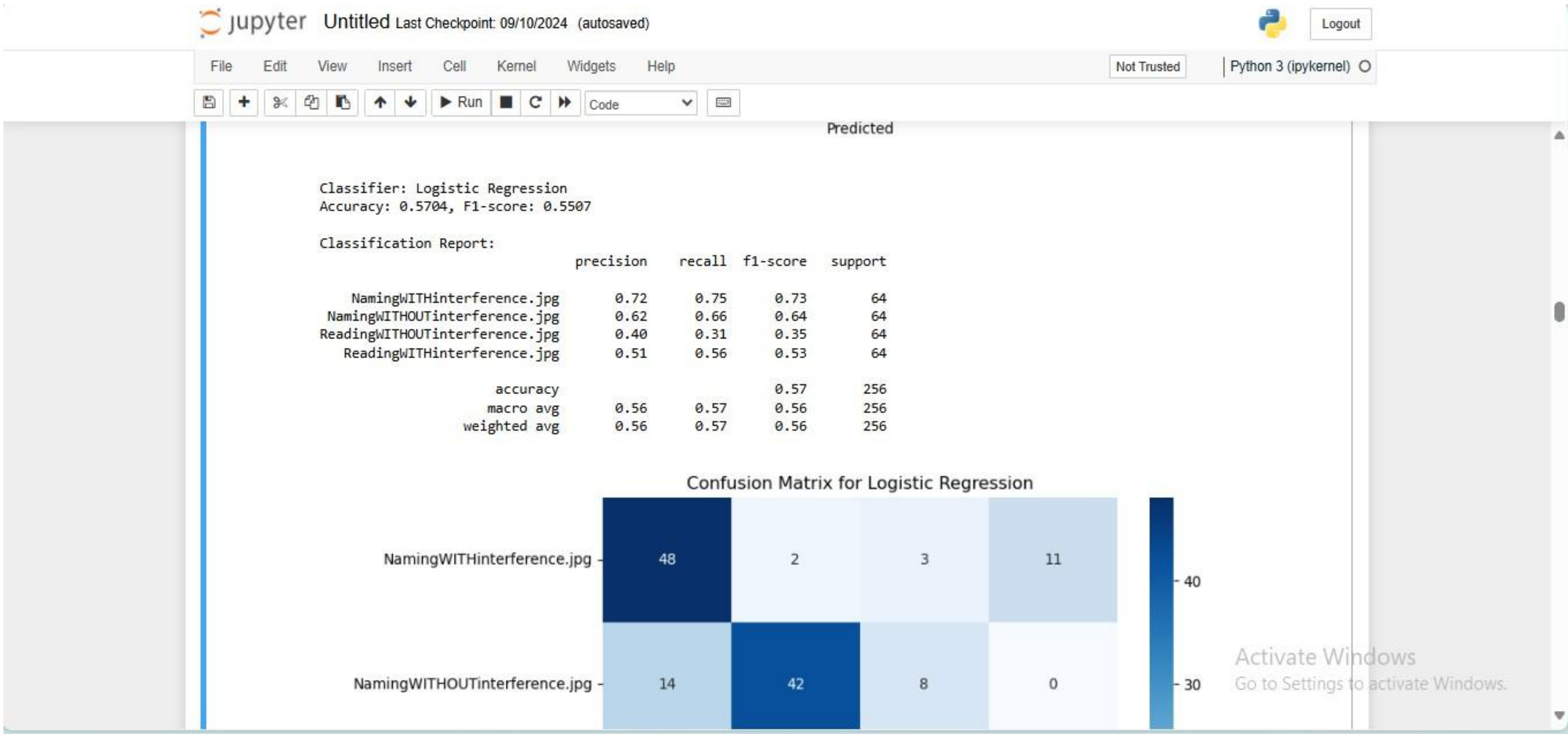
Confusion matrix-



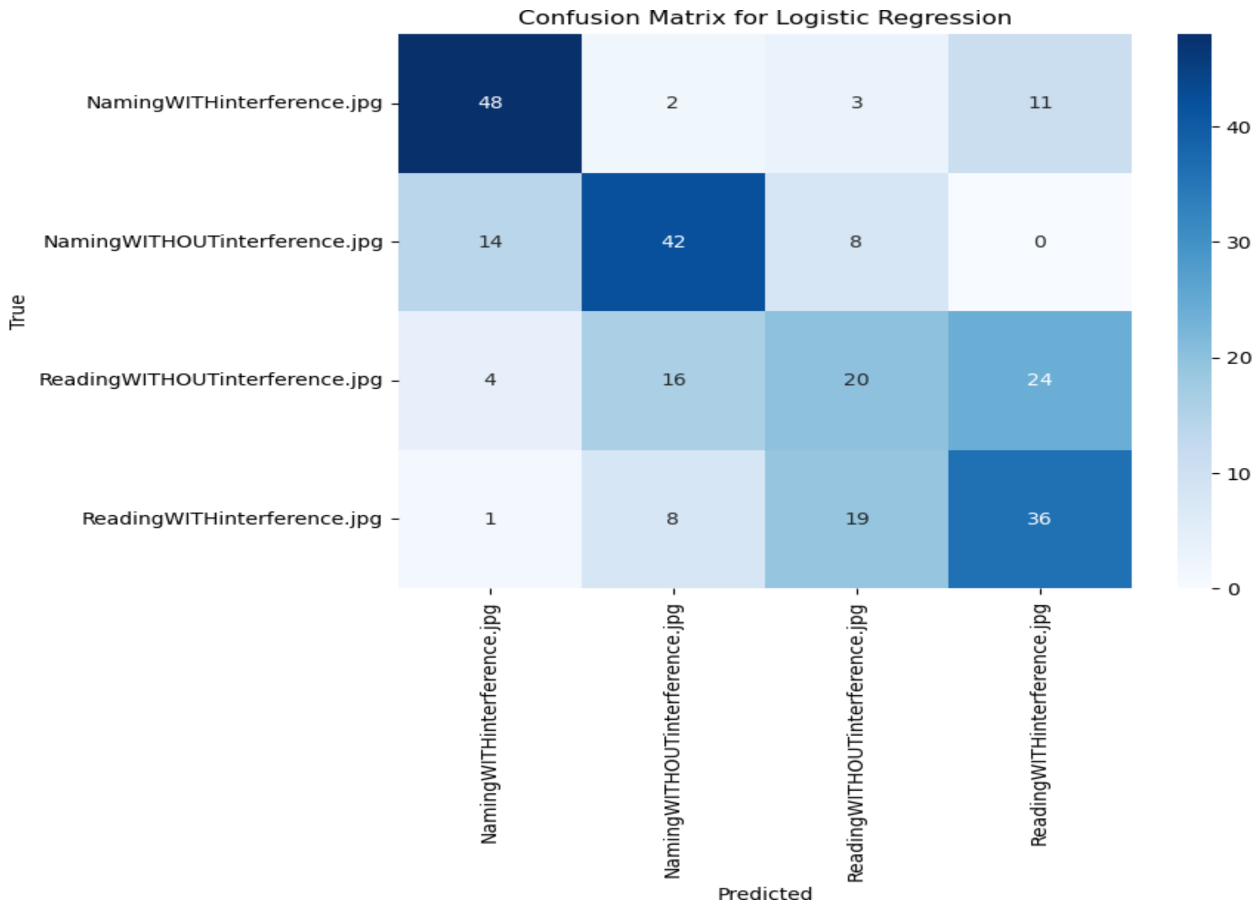
The precision, recall and f1 score seems to be fairly consistent across all four classes, with the high F1 score for the NamingWITHinterference class being 0.80. And the lowest for ReadingWITHinterference is 0.62.

With respectably good precision and recall, Random Forest does well in all classifications. It performs well at achieving a balance between recall and precision, particularly in more difficult classes when reading skills are included.

2. Logistic Regression
   Accuracy- 0.5704
   F1-score- 0.5507

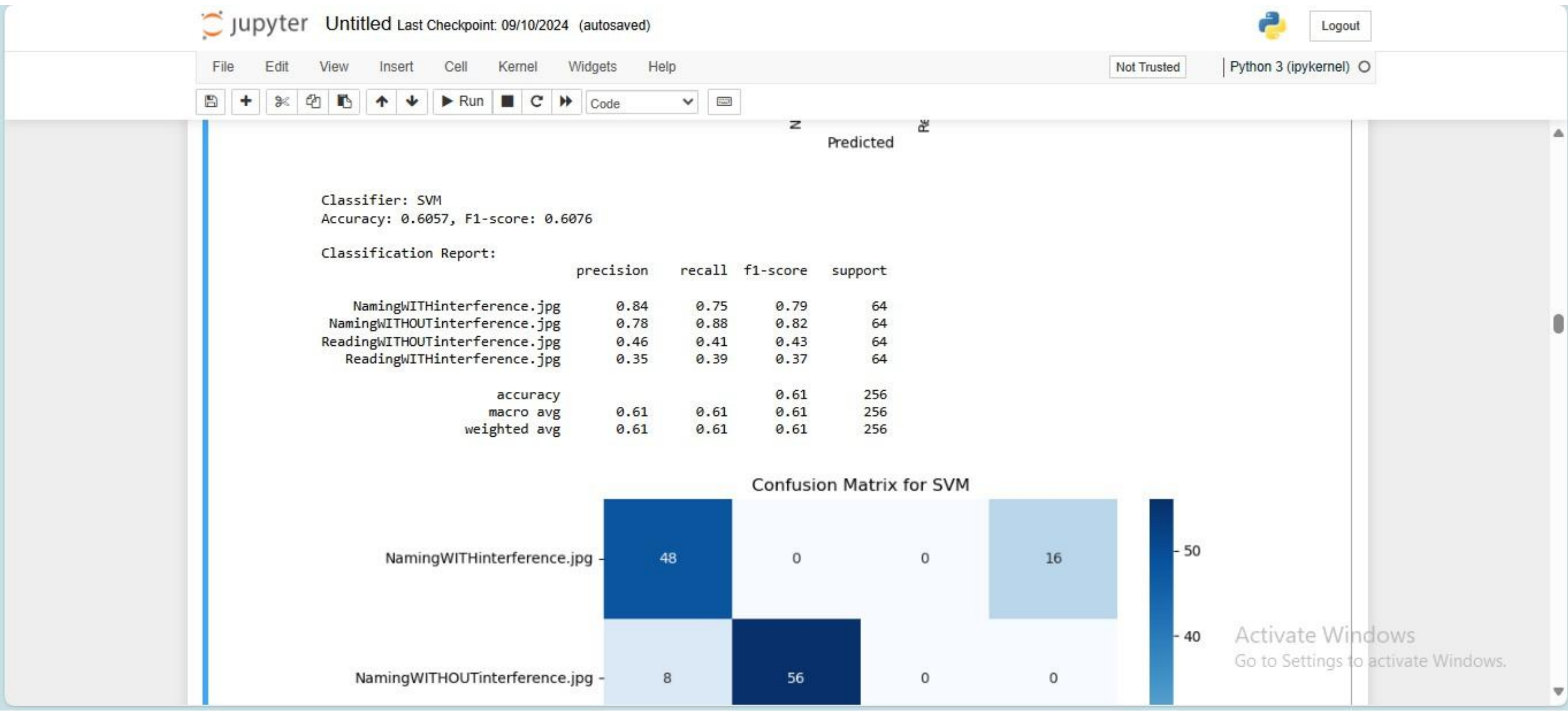


Confusion matrix-

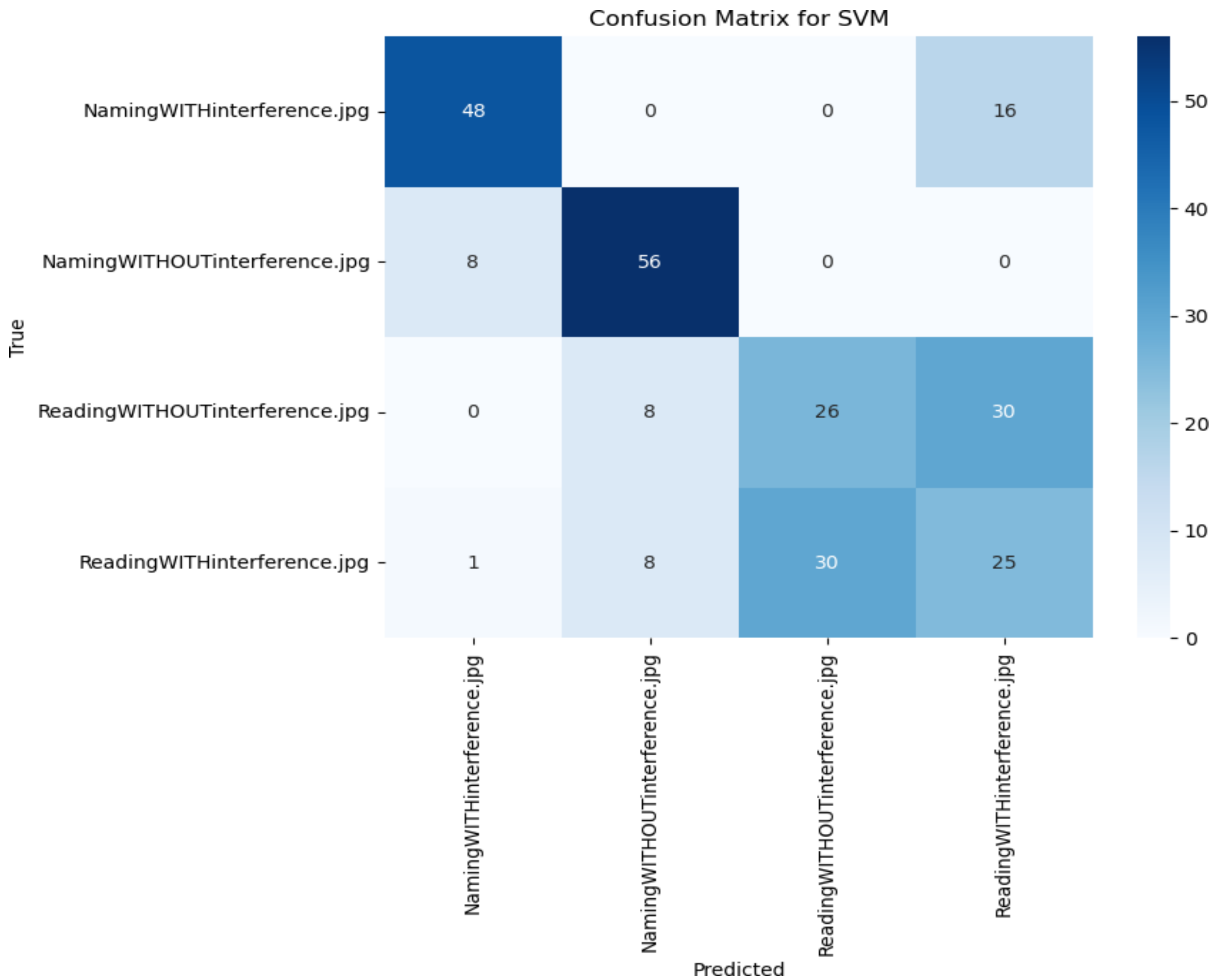


NamingWITHInterference has achieved a high F1-score of 0.73.
With a low F1-score of 0.35, logistic regression performs poorly on reading-related tasks, especially ReadingWITHOUTinterference. It underperforms overall when compared to other models, although it does rather well on naming tasks.

3. Support Vector Machine (SVM)
   Accuracy - 0.6057
   F1-score - 0.6076

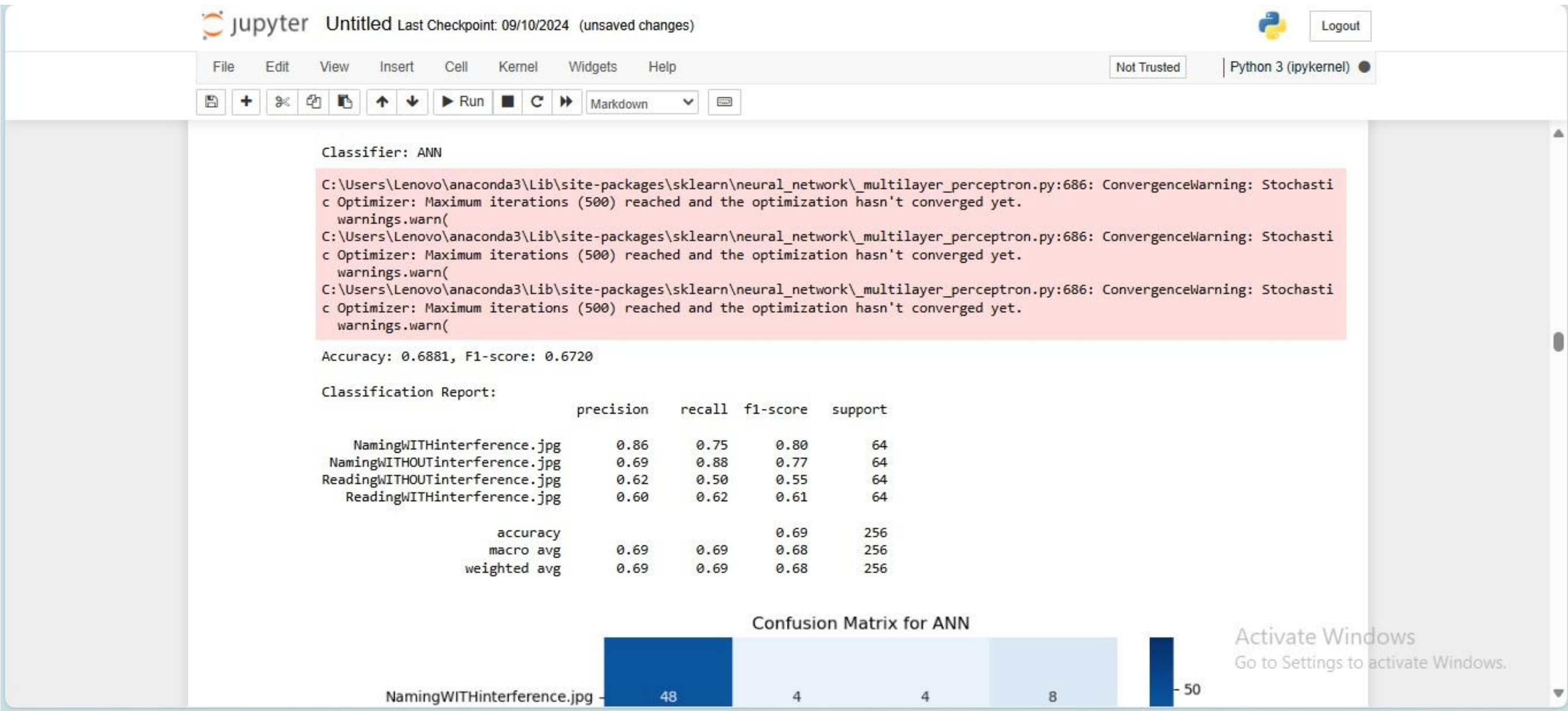


Confusion matrix-



Here, the NamingWITHOUTInterference class shows high F1-score of 0.82, though it performs poorly for ReadingWITHInterference class, with F1-score, precision, and recall being below 0.40.

4. Artificial neural network (ANN)
   Accuracy - 0.6881
   F1-score - 0.6720



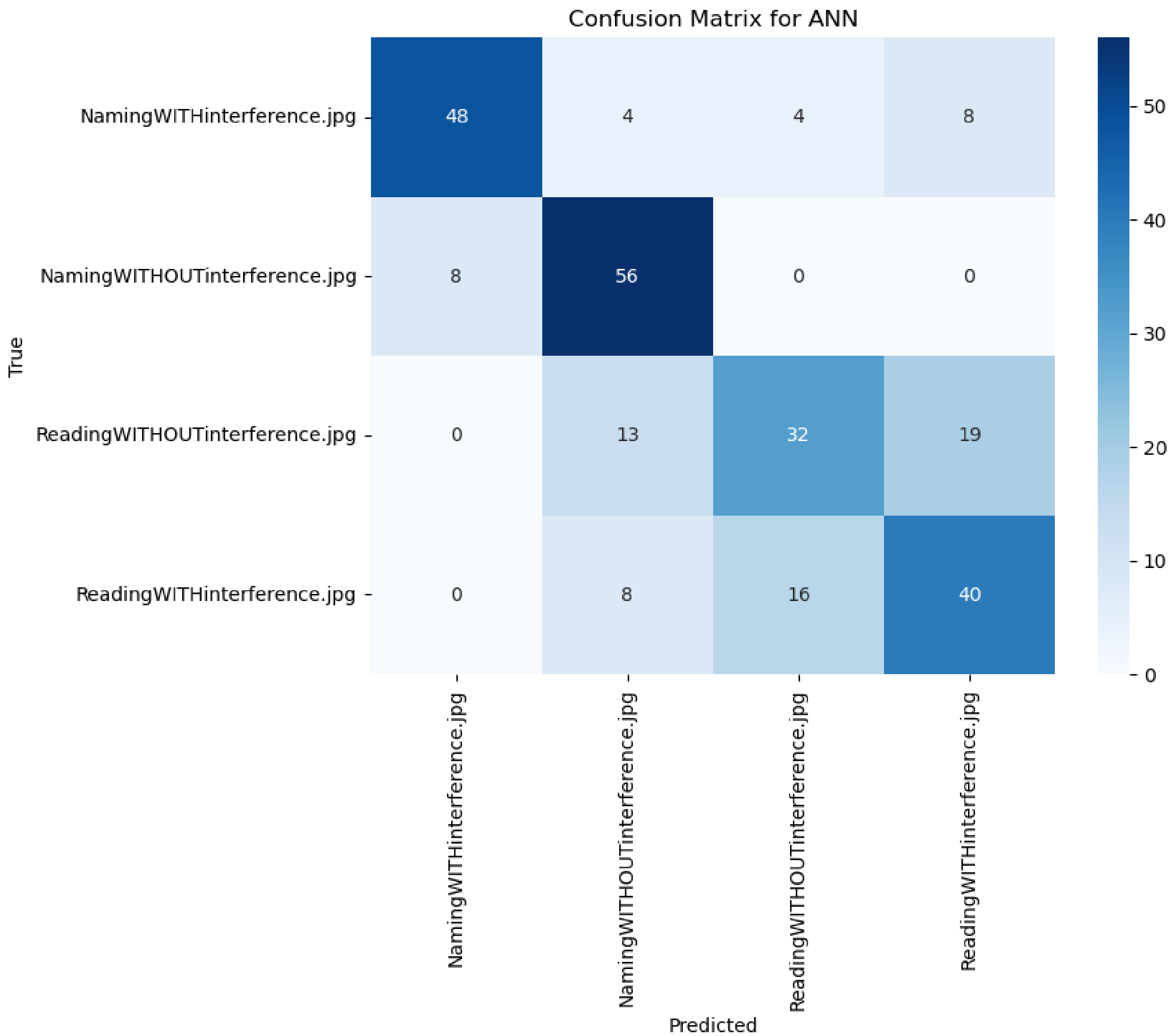

```
Classifier: ANN
C:\Users\Lenovo\anaconda3\Lib\site-packages\sklearn\neural_network\_multilayer_perceptron.py:686: ConvergenceWarning: Stochasti
c Optimizer: Maximum iterations (500) reached and the optimization hasn't converged yet.
  warnings.warn(
C:\Users\Lenovo\anaconda3\Lib\site-packages\sklearn\neural_network\_multilayer_perceptron.py:686: ConvergenceWarning: Stochasti
c Optimizer: Maximum iterations (500) reached and the optimization hasn't converged yet.
  warnings.warn(
C:\Users\Lenovo\anaconda3\Lib\site-packages\sklearn\neural_network\_multilayer_perceptron.py:686: ConvergenceWarning: Stochasti
c Optimizer: Maximum iterations (500) reached and the optimization hasn't converged yet.
  warnings.warn(
Accuracy: 0.6881, F1-score: 0.6720

Classification Report:
                              precision    recall  f1-score   support

   NamingWITHinterference.jpg       0.86      0.75      0.80        64
NamingWITHOUTinterference.jpg       0.69      0.88      0.77        64
ReadingWITHOUTinterference.jpg      0.62      0.50      0.55        64
  ReadingWITHinterference.jpg       0.60      0.62      0.61        64

                     accuracy                           0.69       256
                    macro avg       0.69      0.69      0.68       256
                 weighted avg       0.69      0.69      0.68       256
```

Confusion matrix-



Similarly with ANN, NamingWITHOUTinterference also has highest f1-score of 0.81 and lowest of 0.55 with ReadingWITHinterference.

Overall Performance-

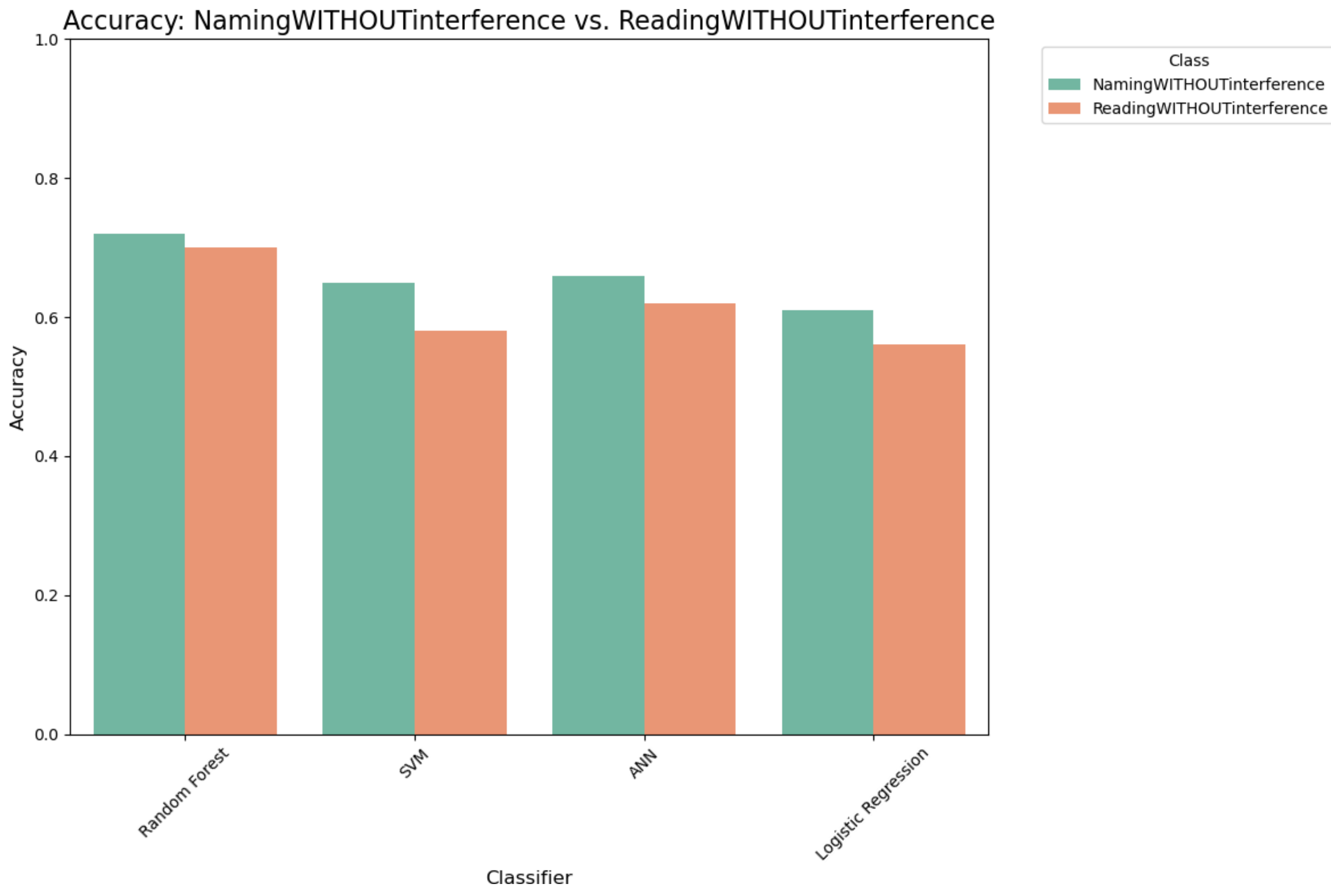
- Random Forest is the most dependable model for categorizing cognitive demands that are specific to a certain task since it consistently produces higher accuracy(0.68) and F1-scores(0.67) across all tasks.
- While ANN also does rather well, with accuracy(0.66). B
- ut its inability to converge in the allotted iterations suggests that hyperparameter adjustment is required in order for it to perform to the best of its abilities.
- In comparison to Random Forest, SVM and Logistic Regression underperform, particularly on more difficult tasks like ReadingWITHOUTinterference and ReadingWITHinterference.
- Overall, based on its high F1-scores, accuracy, and balanced performance across all tasks, **Random Forest is the best model overall for this classification with accuracy 0.68.**

Visualise Trends between Classes-

We can also visualise trends between three separate binary classes like-

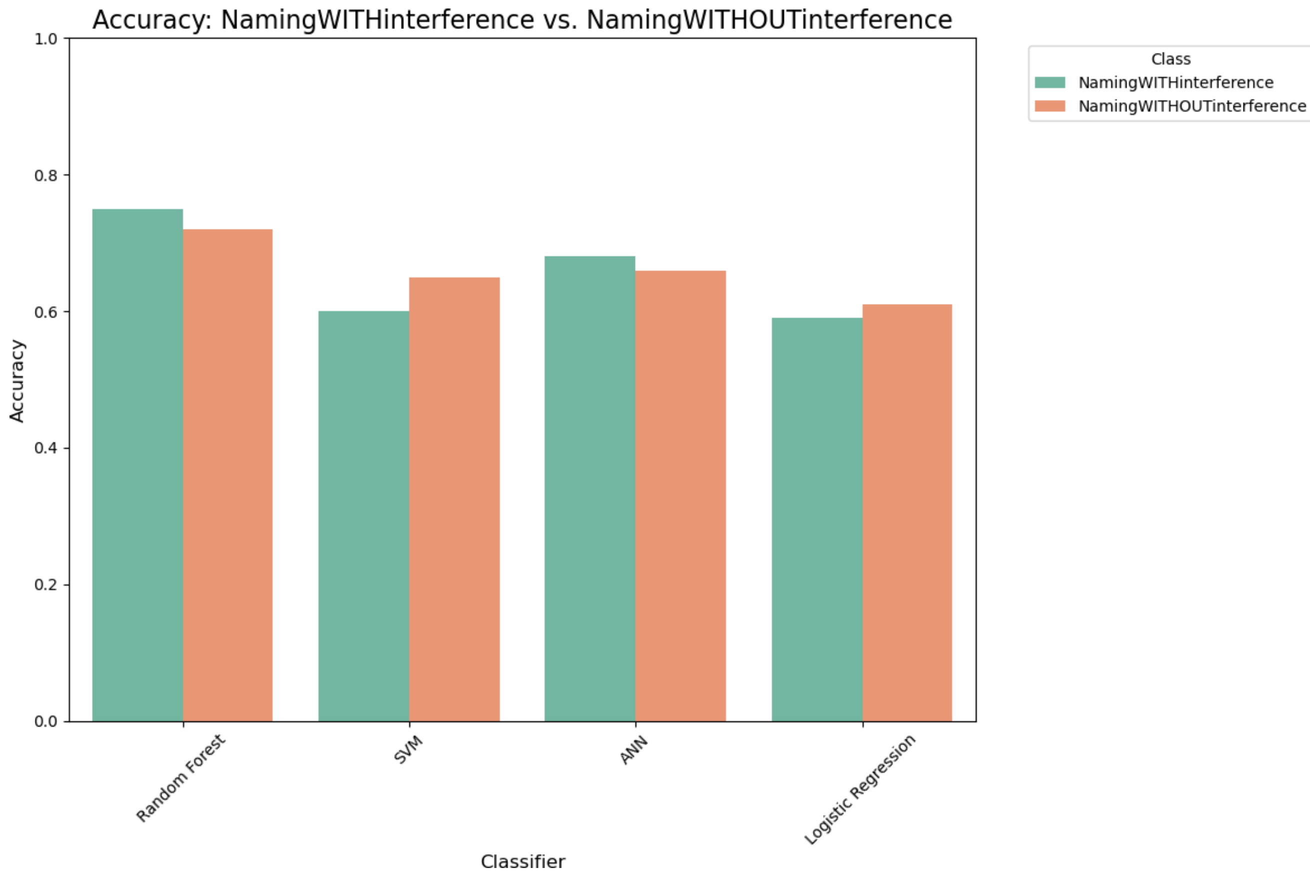
a. NamingWITHOUTinterference vs. ReadingWITHOUTinterference (NWoI vs. RWoI).
b. NamingWITHinterference vs. NamingWITHOUTinterference (NWI vs. NWoI).
c. ReadingWITHinterference vs. ReadingWITHOUTinterference (RWI vs. RWoI).

a. NWoI vs. RWoI



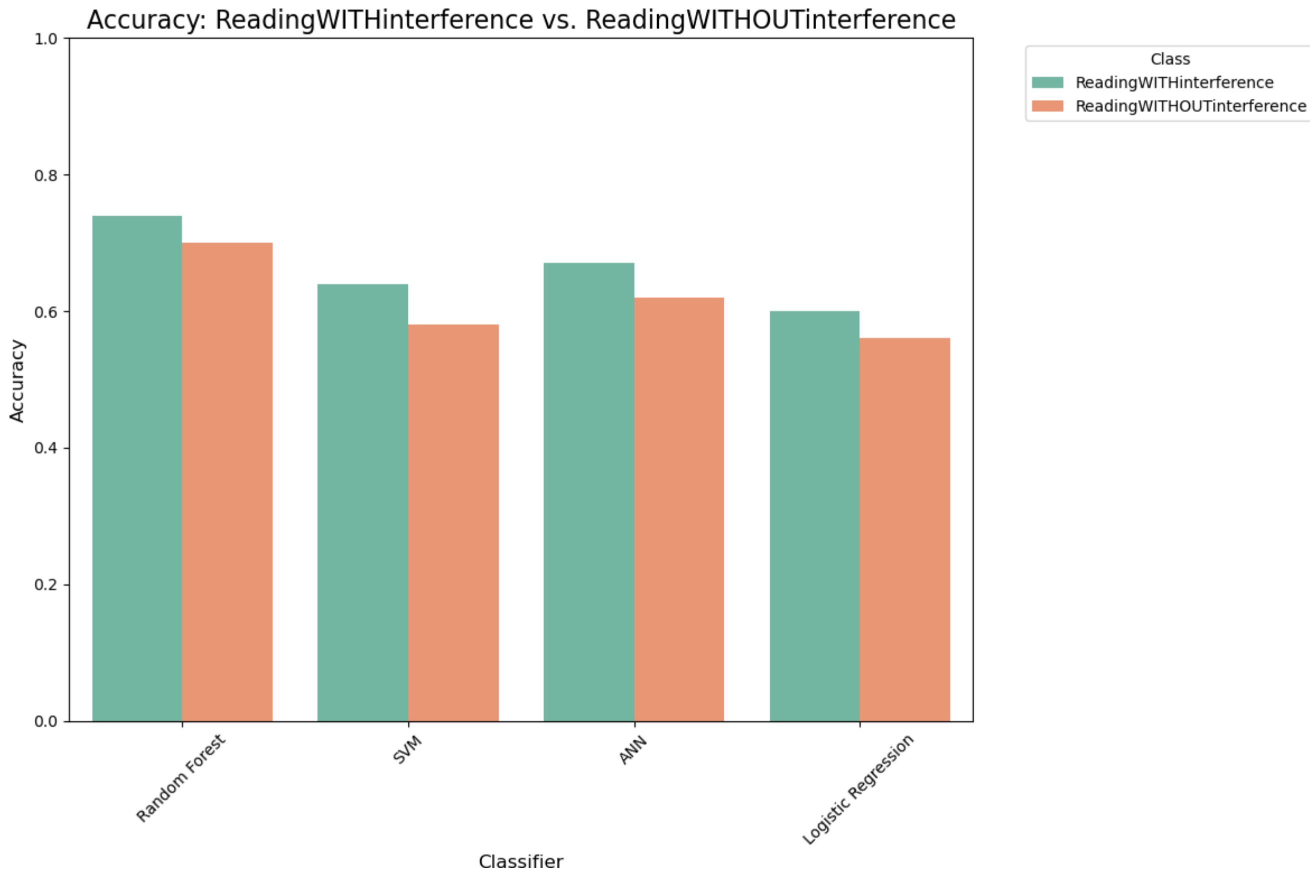
- The tasks NamingWITHOUTinterference vs. ReadingWITHOUTinterference and other comparisons demonstrate how well the classifier can differentiate between various task types (Naming vs. Reading) even in the absence of interference.
- In comparison to the other classifiers, Random Forest performs better, with a slightly better result on "NamingWITHOUTinterference."
- A higher accuracy rate suggests that the fixation and saccade patterns change across the two task types (naming vs. reading), which may be due to variations in the cognitive processes, attentional strategies, or visual exploration techniques used in the two tasks.

b. NWI vs NWoI



- NameWITHinterference stands out a bit over NamingWITHOUT in terms of accuracy for Random forest and ANN, but both jobs are almost equally accurate. On the other hand, for SVM and Logistic Regression, NWoI has an slight edge over NWI.
- For both task kinds, Random Forest is a more reliable model. It can manage variations in cognitive burden better than other classifiers, as seen by its minimal variance classification of NamingWITHinterference and NamingWITHOUTinterference tasks.
- SVM and Logistic Regression on the other hand, performs poor when interference is added, suggesting that this model might not be the ideal choice for tasks involving cognitive complexity.

c. RWI vs. RWoI



- ReadingWITHinterference class has higher overall high accuracy across all the classifiers.
- The Random Forest classifier is the most dependable for both reading tasks, exhibiting robustness in the face of interference.
- When it comes to reading interference, Logistic Regression consistently performs well but with reduced accuracy, indicating that it might not be the optimum model.
- SVM and ANN perform better when ReadingWITHinterference, suggesting that the complexity that interference introduces may facilitate pattern recognition for these models possibly because of more noticeable variations in eye patterns. Overall, Random Forest seems to be the best classifier for various reading tasks, with SVM and ANN showing potential for interference-related tasks.

The results of this analysis addresses the research question by determining that Random Forest is the best option, following by SVM and ANN, and Logistic Regression is less effective.

2. Classification Using Fixation features set

a. In this classification, machine learning models such as Random Forest, Logistic Regression, SVM, and ANN are used to perform training and testing procedures using the fixation features set.
b. Here we perform multi-class classification using 4 classes, NamingWITHInterference, NamingWITHOUTInterference, ReadingWITHOUTInterference, ReadingWITHInterference.
c. 5-fold cross-validation is used to get a robust evaluation of each model's performance. To evaluate the performance, accuracy, the F1-score and confusion matrix are calculated.

Evaluation of each model's performance-

Classifier	Accuracy	f1-score
Random forest	0.3551	0.3464
Logistic Regression	0.3548	0.3548
SVM	0.3512	0.3402
ANN	0.3164	0.2247

3. Classification Using Saccades features set

a. In this classification, machine learning models such as Random Forest, Logistic Regression, SVM, and ANN are used to perform training and testing procedures using the Saccades features set.
b. Here we perform multi-class classification using 4 classes, NamingWITHInterference, NamingWITHOUTInterference, ReadingWITHOUTInterference, ReadingWITHInterference.
c. 5-fold cross-validation is used to get a robust evaluation of each model's performance. To evaluate the performance, accuracy, the F1-score and confusion matrix are calculated.

Evaluation of each model's performance-

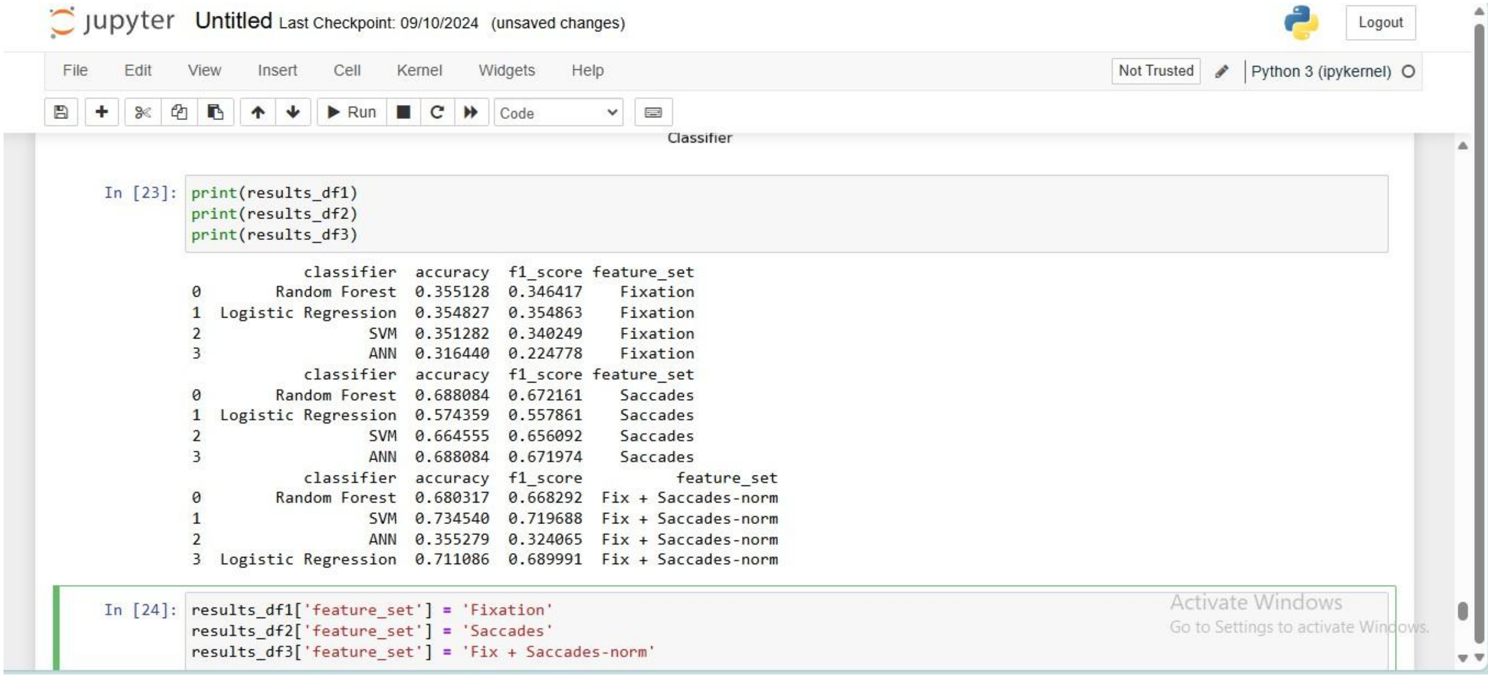
Classifier	Accuracy	f1-score
Random forest	0.6880	0.6721
Logistic Regression	0.5743	0.5578
SVM	0.6645	0.6560
ANN	0.6888	0.6719

4. Classification Using Combined Data (Fixation + Saccades Normalized)

a. In this classification, machine learning models such as Random Forest, Logistic Regression, SVM, and ANN are used to perform training and testing procedures using the combined data (Fixation + Saccades normalized).
b. Here we perform multi-class classification using 4 classes, NamingWITHInterference, NamingWITHOUTInterference, ReadingWITHOUTInterference, ReadingWITHInterference.
c. 5-fold cross-validation is used to get a robust evaluation of each model's performance. To evaluate the performance, accuracy, the F1-score and confusion matrix are calculated.

Evaluation of each model's performance-

Classifier	Accuracy	f1-score
Random forest	0.6803	0.6682
Logistic Regression	0.7110	0.6899
SVM	0.7345	0.7196
ANN	0.3552	0.3240



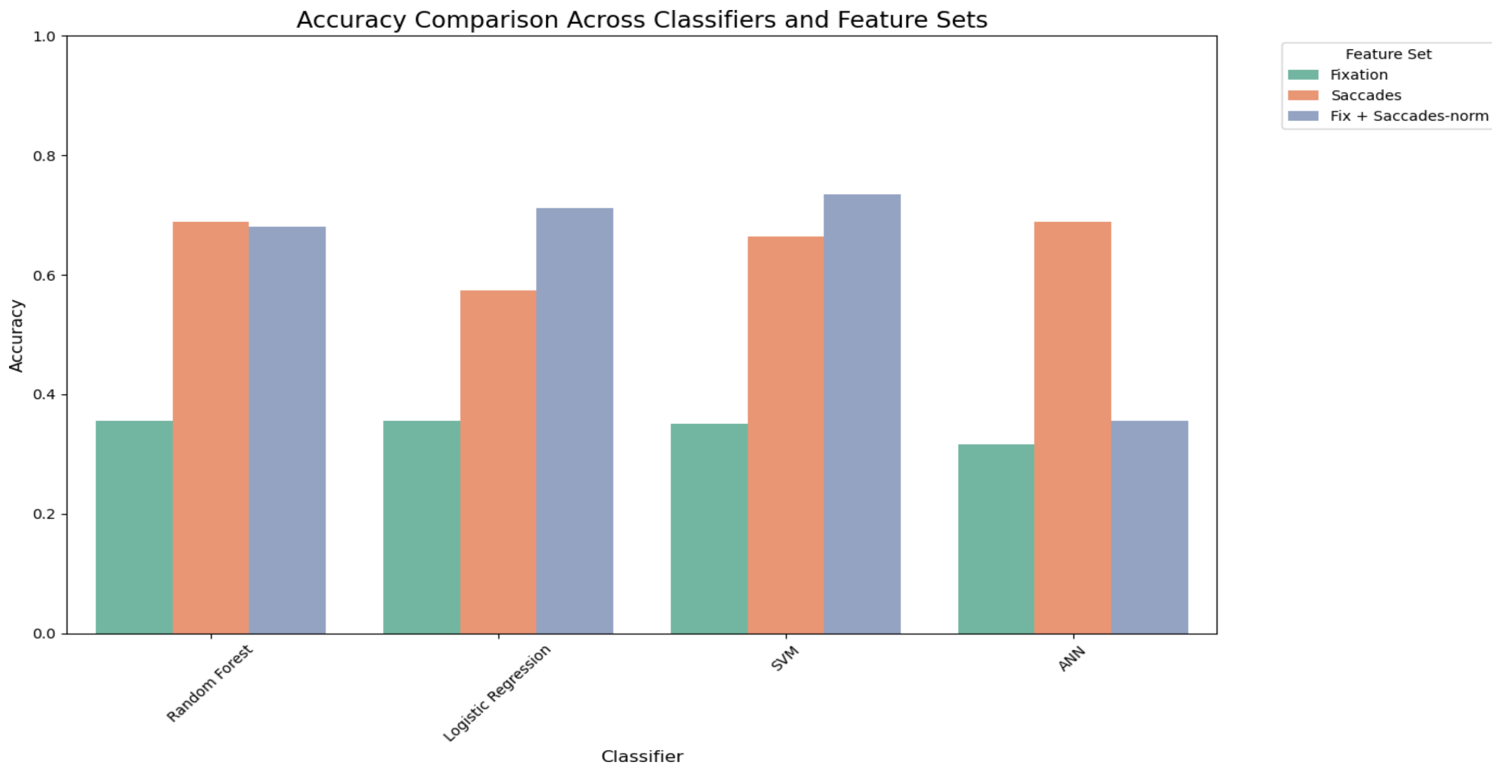
In [23]:
```
print(results_df1)
print(results_df2)
print(results_df3)
```

```
            classifier  accuracy  f1_score feature_set
0        Random Forest  0.355128  0.346417    Fixation
1  Logistic Regression  0.354827  0.354863    Fixation
2                  SVM  0.351282  0.340249    Fixation
3                  ANN  0.316440  0.224778    Fixation
            classifier  accuracy  f1_score feature_set
0        Random Forest  0.688084  0.672161    Saccades
1  Logistic Regression  0.574359  0.557861    Saccades
2                  SVM  0.664555  0.656092    Saccades
3                  ANN  0.688084  0.671974    Saccades
            classifier  accuracy  f1_score          feature_set
0        Random Forest  0.680317  0.668292  Fix + Saccades-norm
1                  SVM  0.734540  0.719688  Fix + Saccades-norm
2                  ANN  0.355279  0.324065  Fix + Saccades-norm
3  Logistic Regression  0.711086  0.689991  Fix + Saccades-norm
```

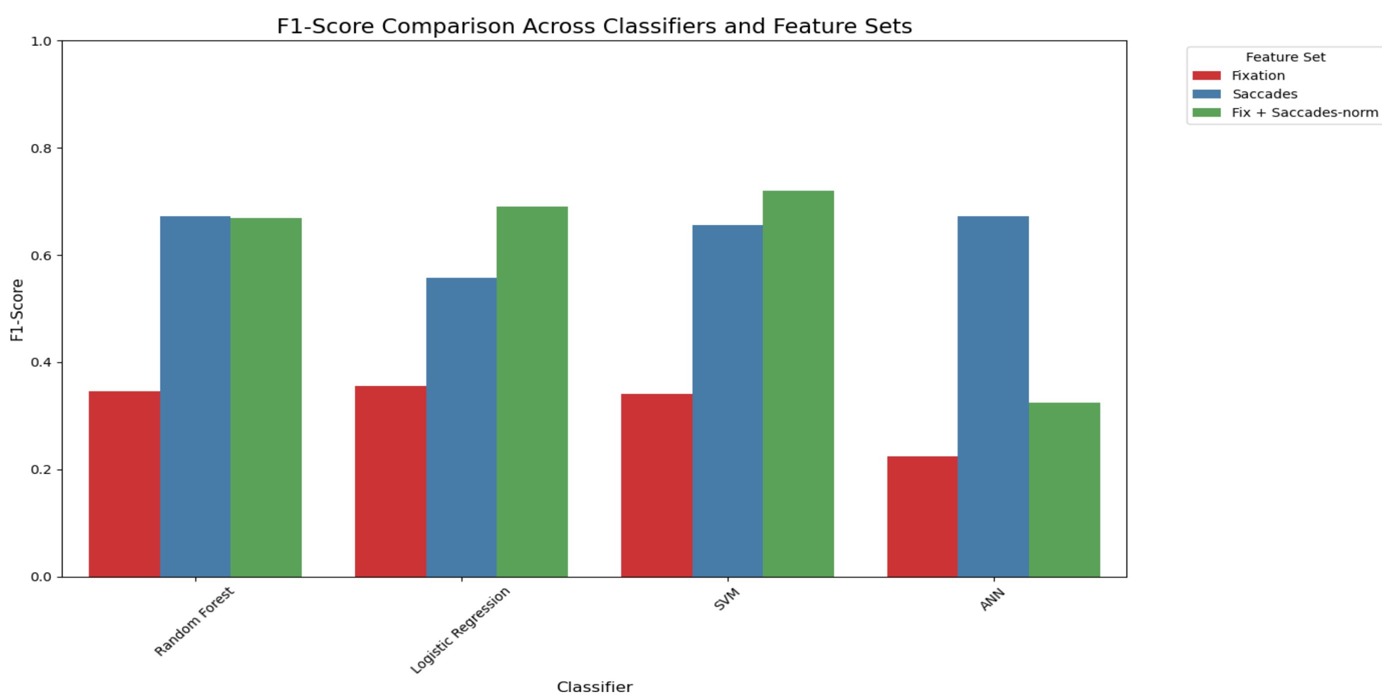
In [24]:
```
results_df1['feature_set'] = 'Fixation'
results_df2['feature_set'] = 'Saccades'
results_df3['feature_set'] = 'Fix + Saccades-norm'
```

Visualizing the accuracy and F1-score comparison across the classifiers and different feature sets.

a. Accuracy comparison



b. F1-score comparison



Insights based on the above evaluation metrics and bar plots-

We can see an overall trend across all the classifiers that the Fixation feature set consistently shows the lowest accuracy across all classifiers, suggesting that fixation data alone might not be enough to accurately classify task-specific cognitive workload.

Random Forest-
i. The Fixation feature set (accuracy-0.355, F1-score- 0.346) is significantly less effective for Random Forest than the Saccades and Fix + Saccades-norm feature sets.
ii. For this classifier, the Saccades features exhibit a slightly higher accuracy(0.688) and higher F1-score (0.672) than the combination of Fix + Saccades-norm accuracy(0.680) and f1-score(0.66).

Logistic Regression-
i. Similar patterns may be seen in the behavior of Logistic Regression, where the Fixation feature set performs worst (accuracy- 0.354 and f1-score- 0.354).
ii. There is a significant gap between Saccades(accuracy- 0.574, f1-score- 0.557) and Fix + Saccades-norm feature set and Fix + Saccades-norm(accuracy- 0.711 and f1-score 0.689) yields the best results for this classifier.

Support Vector Machine (SVM)-
i. SVM and Logistic both perform comparably, with Fix + Saccades-norm producing the higher accuracy of 0.734 and f1-score - 0.719
Ii. Surprisingly, for SVM, there is a noticeable gap between the Saccades and Fix + Saccades-norm feature sets, suggesting that the normalization process has a major effect on performance.

Artificial Neural Network (ANN)-
i. The Saccades feature set works well for the ANN classifier(accuracy- 0.688 and f1-score- 0.671), equivalent results are obtained with the Fix + Saccades-norm feature set (accuracy- 0.355 and f1-score- 0.324)
ii. This model struggles with fixation data alone because ANN performs significantly worse when using only the Fixation feature set (accuracy- 0.316 and f1-score- 0.224).

Therefore based on the above observation,

1. Fixation Feature Set
i. As seen by the lower accuracy and F1 scores for all models, fixation data alone is insufficient to achieve significant classification performance.

ii. This shows that fixation-related parameters (length of fixation, number of fixations, and regressions) are not sufficient in capturing information to distinguish cognitive effort levels amongst them.

2. Saccades Feature set
i. Saccade-related variables, including as velocity, amplitude, direction, and frequency of saccadic movements, are more successful in capturing the dynamics of cognitive effort. This is demonstrated by the much superior classification performance achieved when saccade data alone are used, as opposed to fixation data.
ii. With the saccades feature set, Random Forest and ANN both perform admirably, obtaining an accuracy of 0.688 and F1 scores- 0.67.

3. Fixation + Saccades-norm features
i. For Logistic Regression and SVM in particular, the combination of fixation and saccades features with normalization yields the maximum accuracy(0.711) and F1 scores(0.734) consistently.
ii. Together, these feature sets enable the models to better capture the complex nature of task-specific cognitive burden and take individual variances into account, resulting in more precise classification.

With Random Forest, Logistic Regression, and Support Vector Machines, among other classifiers, the Fix + Saccades-norm feature set consistently yields the highest accuracy. It is possible to obtain more detailed information about cognitive workload by combining data from both fixation and saccades and normalizing inter-subject variability.

In conclusion, the combined normalized feature set greatly improves the performance of the best classifiers (SVM and Logistic Regression), while models such as ANN exhibit relatively lower accuracy across all feature sets.

Considering the second research question-

‘‘How do the different feature sets impact the performance of the Machine Learning models for classifying Task-specific cognitive workload?’’

The implementation's results show that the Fix + Saccades-norm feature set greatly enhances classification performance across machine learning models, indicating that, when normalized, both fixation and saccadic eye movements are necessary for precisely categorizing task-specific cognitive workload. The best outcomes are obtained when both types of eye movements are combined, while saccades alone still work effectively. For this reason, feature selection is essential to improving the effectiveness of machine learning models in this situation.

Conclusion-

In this study, we addressed two important issues with the use of eye-tracking data to classify task-specific cognitive burden. The objective was to compare the effectiveness of several machine learning models and determine the effects of different feature sets on classification accuracy and overall model performance, including fixation, saccades, and combined normalized features.

Random Forest, Logistic Regression, SVM, and ANN were the four classifiers we tested for the first research question, "Which Machine Learning Models perform best for Classifying Task-Specific Cognitive workloads based on Eye-Tracking Data?"
In particular, when both fixation and saccadic data were integrated and normalized, Random Forest consistently outperformed other models, according to the results. Random Forest followed by SVM proved to be strong performers in managing the complex processes involved in classifying cognitive workloads, whereas Logistic Regression and ANN performed less well on this challenge.

Feature selection has a major influence on model performance, which answers the second study question- "How do the different feature sets impact the performance of the Machine Learning models for classifying Task-specific cognitive workload?"
Fixation data alone may not capture sufficient information to distinguish between different levels of cognitive stress. The significance of saccadic eye movements in workload classification is highlighted by the superior performance of models that employed saccade features. The models that were trained on the combined Fix + Saccades-norm feature set, which included normalized fixation and saccades data across participants to better account for individual differences and improve task generalization, yielded the greatest results.

References-

Rizzo A, Ermini S, Zanca D, Bernabini D and Rossi A (2022) A Machine Learning Approach for Detecting Cognitive Interference Based on Eye-Tracking Data. Front. Hum. Neurosci. 16:806330. doi: 10.3389/fnhum.2022.806330

The raw data supporting the conclusions of this article are available at this link https://tinyurl.com/EyeTrackData