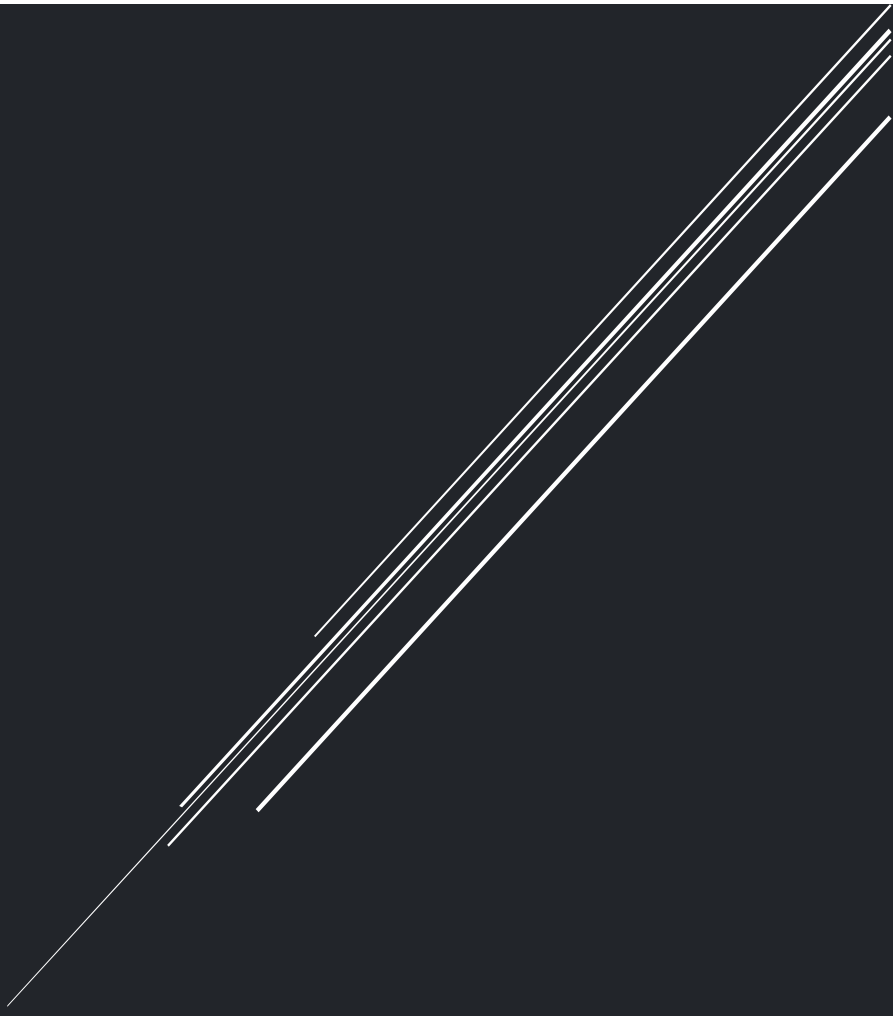
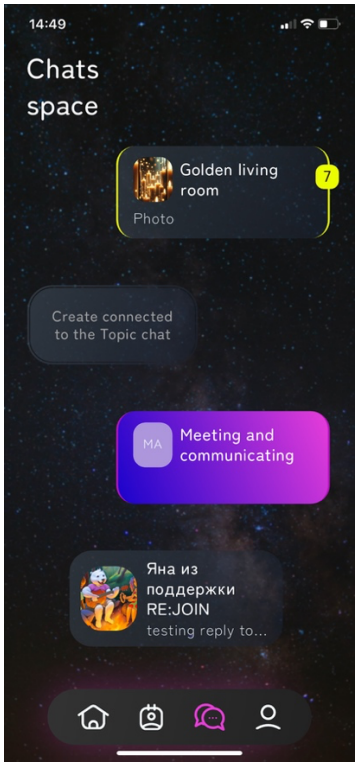


Portfolio

Advit Ahuja

Table of Contents

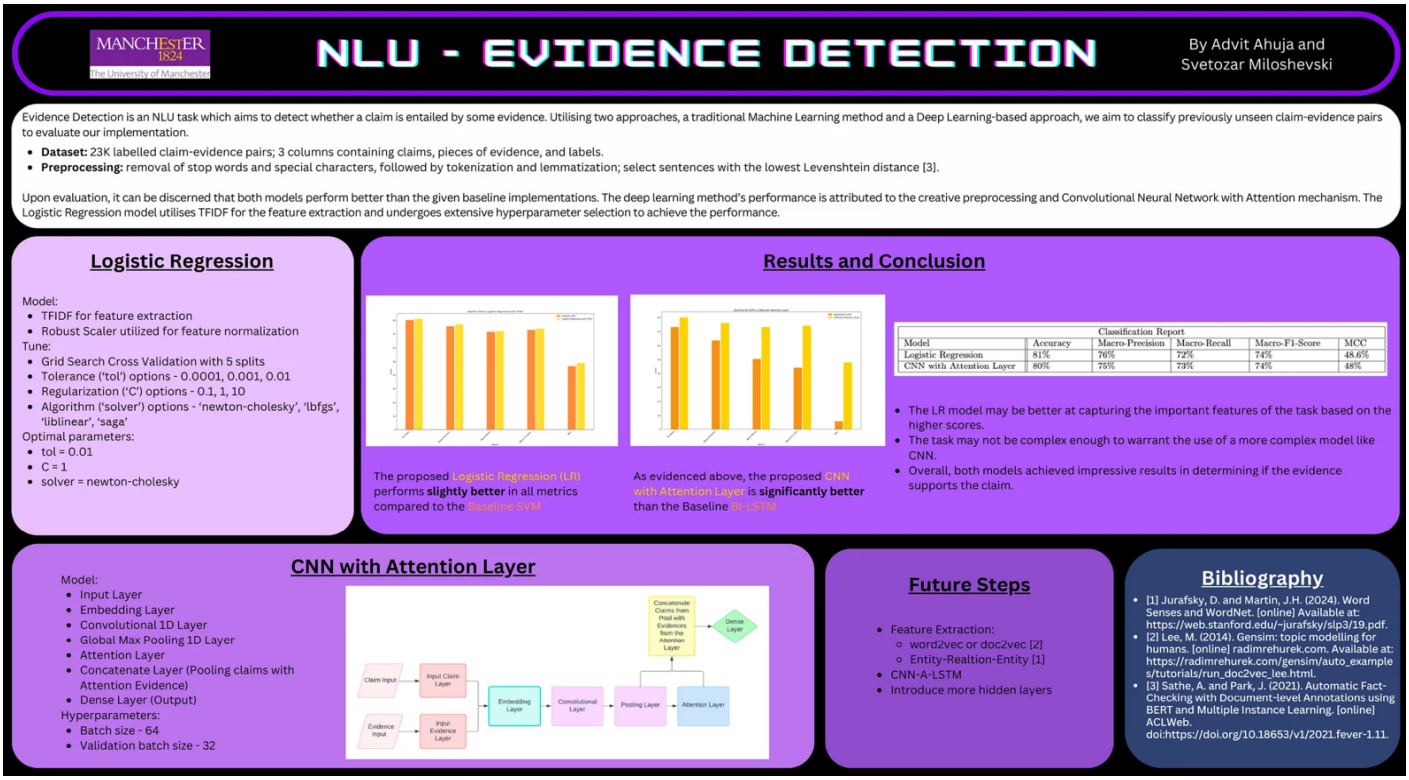
RE:JOIN - Mobile App Developer



During my final year, I was also working with a start-up called RE:JOIN. We were creating a platform for content creators to gain more insight into their customers and would use similar tools to yourselves to increase customer engagement and boost revenue.

My experience in RE:JOIN led me to create a chat room between creators and clients, primarily working on front-end programming to display key screens for the creator and user. Moreover, I continuously pushed production code, code reviewed, and added new branches to the pipeline, accelerating the efficiency of the small team. Working closely with the design team and back-end team, we collaborated on our ideas to help one another as a team to reach our goals.

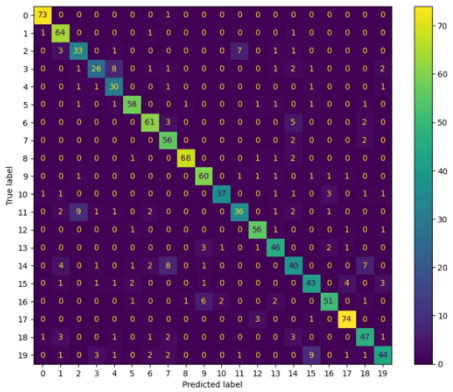
Natural Language Understanding - Claim Evidence Detection



Cognitive Robotics - Image Classification

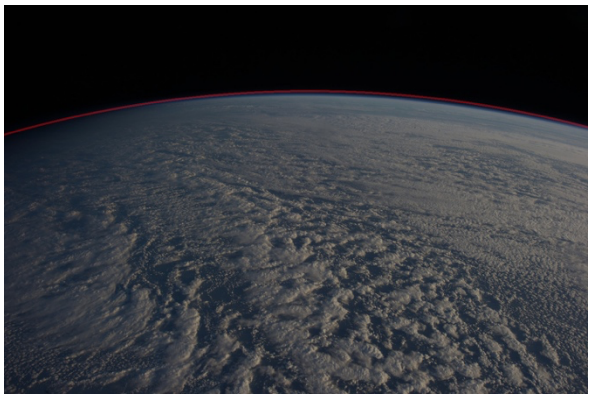
Experimented on a Convolution Neural Network (CNN) to classify images from the CIFAR-100 dataset. Modelled a CNN architecture to initially classify unseen images. Plotted the accuracy against the epochs for different dropout rates, pool sizes, kernel sizes, and Learning Rates to tune the CNN. Being given a model CNN, I tested, iterated, and then built again, which resulted in a 14% increase in accuracy in comparison to the baseline.

Summary of results: In conclusion, the overall best accuracy (0.54) and loss (1.71) come from the following hyperparameter values: pool size, (2,2), kernel size, (3,3) and learning rate, 0.001, dropout, 0.2. Observing the confusion matrix, we can see that the model is better at classifying some images compared to others. This could be due to some classes being similar in shape to others, such as a car versus a van.



Visual Computing - Horizon Detection

Developed a computer vision pipeline to detect the horizon given an image as input using OpenCV with C++. Pre-processed via grayscale conversion, binary thresholding (Otsu coupled with a manual trackbar), and Canny edge detection for feature extraction. Tuned Hough Line parameters to identify and filter horizontal lines in conjunction with a polynomial regression on filtered line coordinates to locate and plot a curve over the horizon.



Predicting Neutron and Muon Source Reliability at ISIS using Machine Learning

The ISIS Neutron and Muon Source is a world-leading scientific facility that unlocks the secrets of materials at the atomic level. By using powerful particle accelerators, ISIS generates neutron and muon beams that scientists use to probe the structure and behavior of materials, leading to breakthroughs in everything from medicine to energy. The entire process requires a series of individual subsystems to work continuously for many weeks in a cycle. Since the particle accelerator is constantly under high-performance operation, it can also lead to downtime through maintenance issues, impacting ongoing scientific endeavours. The particle accelerator hosts a vast series of sensors that record the state of the machine. While monitoring sensor readings can help identify some failing components, machine learning (ML) offers a more powerful approach. ML excels at analyzing complex datasets, allowing it to predict potential failures before they occur. This study specifically investigates the potential of using ML to predict failures within the ion source and uncover patterns within the readings.

Methodology

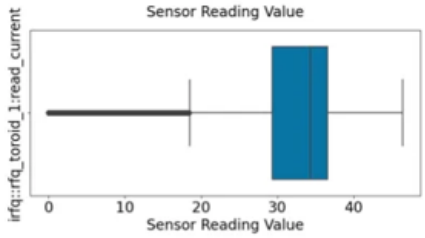
Preprocessing of Sensors

Using Python libraires such as Pandas DF, I was able le to format the given sensor readings using a time window expansion to prepare the data to be trained on.

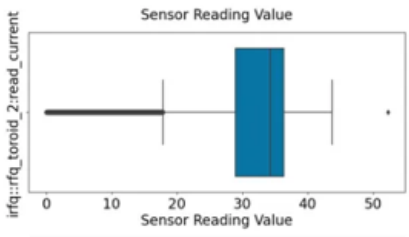
Labelled Sensor Readings per User Run		
Column Name	**Data Type**	**Description**
FaultDateTime	DateTime	date and time of fault: %Y-%m-%d %H:%M:%S
Sensor Name 1	Unit	Sensor reading given the time
...	...	...
Sensor Name n	Unit	Sensor reading given the time
Label	Int	0 or 1 for representing an ion source failure

Exploratory Data Analysis (EDA)

Evaluated visual statistical methods for feature extraction. Examples include Box and Whisker Plots, Heat Maps, Hierarchical Clustering, KDE, PCA. By finding sensors that behave similarly, we can eliminate redundant features for efficient training.



(a) Box and Whisker Plot of irfq:irq_toroid_1:read_current.



(b) Box and Whisker Plot of irfq:irq_toroid_2:read_current.

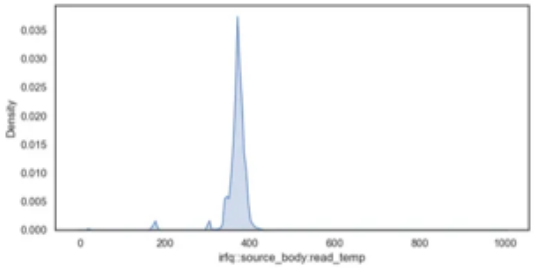
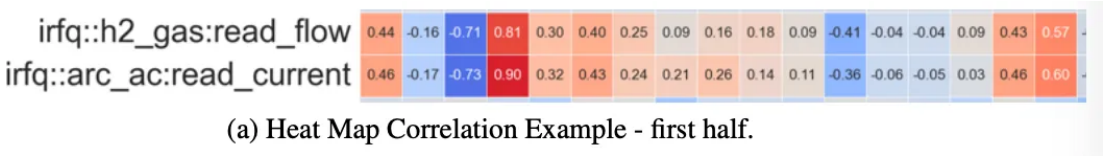


Figure 4.4: Kernel Density Estimate of irfq::source_body:read_temp.



(a) Heat Map Correlation Example - first half.

Machine Learning Pipelines

Tuned, trained, and tested against multiple supervised ML algorithms, including: Logistic Regression, Decision Trees, Random Forest, AdaBoost, and XGBoost. Researching a variety of models expanded my breadth for a solution and aided my interpretation of the results.

Results

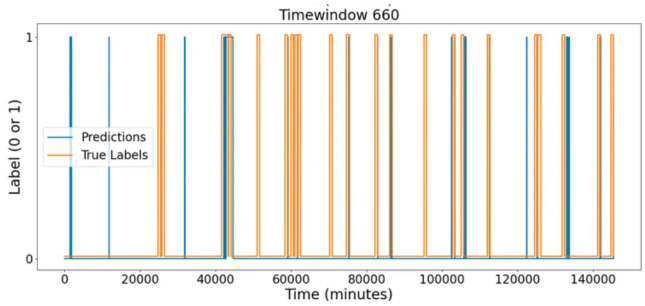


Figure 4.9: Prediction Visualisation of Logistic Regression - for time window expanded to 660 minutes.

Time Window	660				
MCC (3 d.p)	0.066				
Tabular Rectified Recall	128335 (TN)	1006 (FP)			
	12 (FN)	10 (TP)			
Classification Report		**Precision**	**Recall**	**F1-Score**	**Support**
	0	1.00	0.99	1.00	129341
	1	0.01	0.45	0.02	22
	Accuracy			0.99	129363
	Macro Average	0.5	0.72	0.51	129363
	Weighted Average	1.00	0.99	1.00	129363

Table 4.11: Summary Report for Logistic Regression with Time Window 660.

Based on the data, the most promising results for future work could be to use Logistic Regression and AdaBoost models, which are better 'regressors', separating the fine line between true failures and false positives. Whereas Decision Trees, Random Forest, and XGBoost, which are all propitious for finding repeated patterns, cannot comparatively accurately predict failures. In summary, we can show that the best models and respective time windows to use would be Logistic Regression with 660 and AdaBoost with 120, given the experiments. After evaluating several models and discussing the theory of how each model predicts a failure, the analysis reveals that there are some repeated patterns that result in failures.

By pioneering into this specific unexplored field of research and achieving initial breakthroughs, we have successfully presented the future potential of this project to use machine learning models to predict failures within the particle accelerator at ISIS; decreasing downtime.