COMP0053: Affective Computing and Human Robot Interaction - Group Project Report

Agledahl, Sondre Burlacu, Sebastian-Valentin Cheng, Wing Lam

Chu, Ringo Winkelhake, Carlo

Chapter 1

Creating a dataset

1.1 Aims of our affect recognition system

The idea behind our affect recognition system came from Bobby Fisher, one of the greatest chess players of all time. An enigmatic, yet highly intelligent individual, he was not shy of giving controversial but insightful quotes about his life as a brilliant top chess figure. One of his most notable ones "I like the moment I break a man's ego."[1], is what inspired this team's search of defining, finding and analysing this precise point in a chess game.
It is a known fact that when two opponents face each other in this sport, a battle of egos is unleashed over the board. Both players try to employ any tactic to gain some advantage over their opponent, and this is done by vigorously studying each other's playstyle, analysing previous games and more. So when is the moment one's ego gets broken during a chess game? Going through the most critical phases of such a match by reviewing live chess games and observing top rated players exhibiting different types of behaviour, our team concluded that this point in time is equivalent to when a player realises that he lost.

When watching live games of some of the most current well-known players, a high proportion of them show different levels of frustration, anguish, relief or exhilaration[2][3][4][5]. Most of the time, these affective facial states were amplified in high-intensity games such as blitzes (when the players usually have each five minutes with or without time increment per move). Even though an outsider might see chess as just a simple game, the emotional and physiological signs of these top-rated players indicate otherwise. This aspect is further supported by an article that describes the physical strains a body of such a player goes through during a prestigious tournament[6]. A brief description is given about the amount of physical exercise performed by these players to stay fit and be able to cope with all the stress during a game. It also introduces the example of the 1984 World Chess Championships which were called off after five months and 48 games because defending champion Anatoly Karpov had lost 22 pounds. Burning calories is correlated with an increased heart rate and consequently, the overall body temperature.

Therefore, our team concluded that studying the heart rate and facial expressions of chess players is a good starting point in the journey of discovering when one's ego has been broken. Our experimental pipeline is composed of the heart rate Empatica[1] sensor and a facial analysis library.

[1] `https://www.empatica.com/`

The chess environment is supported by Lichess.com[2], an online free chess platform where a player can be matched against another using an invitation URL. Using two laptops, two Empatica sensors and recorded videos using the front cameras of the laptops, we started collecting data from different people who were chess enthusiasts and agreed to take part in our experiments.

Our team was able to conduct four such experiments (four games) most of which were attended by players with similar levels. Each player had five minutes and a few seconds of increment per move. This time constraint ensured that the games were highly intensive, which was needed to amplify the emotional states of the players. At the start of each game, two members of our team were starting and synchronising the Empatica sensors as well as the video recordings. The moves of the matches were recorded directly on the chess platform and later stored for the analysis and labelling part of our experimental pipeline.

1.2 Labelling data

1.2.1 Heart rate monitor

We processed heart rate data from the Empatica sensor to visualise a timelapse of heart rate for each player as it related to the self-reported "loss point" for the losing player. As an additional source of information, we analysed each match via the chess engine Stockfish to relate the player's perceived loss point with the point at which the engine thought the match shifted in the winning player's favour. This data is summarised in figure 1.1.

[2] `https://lichess.org`

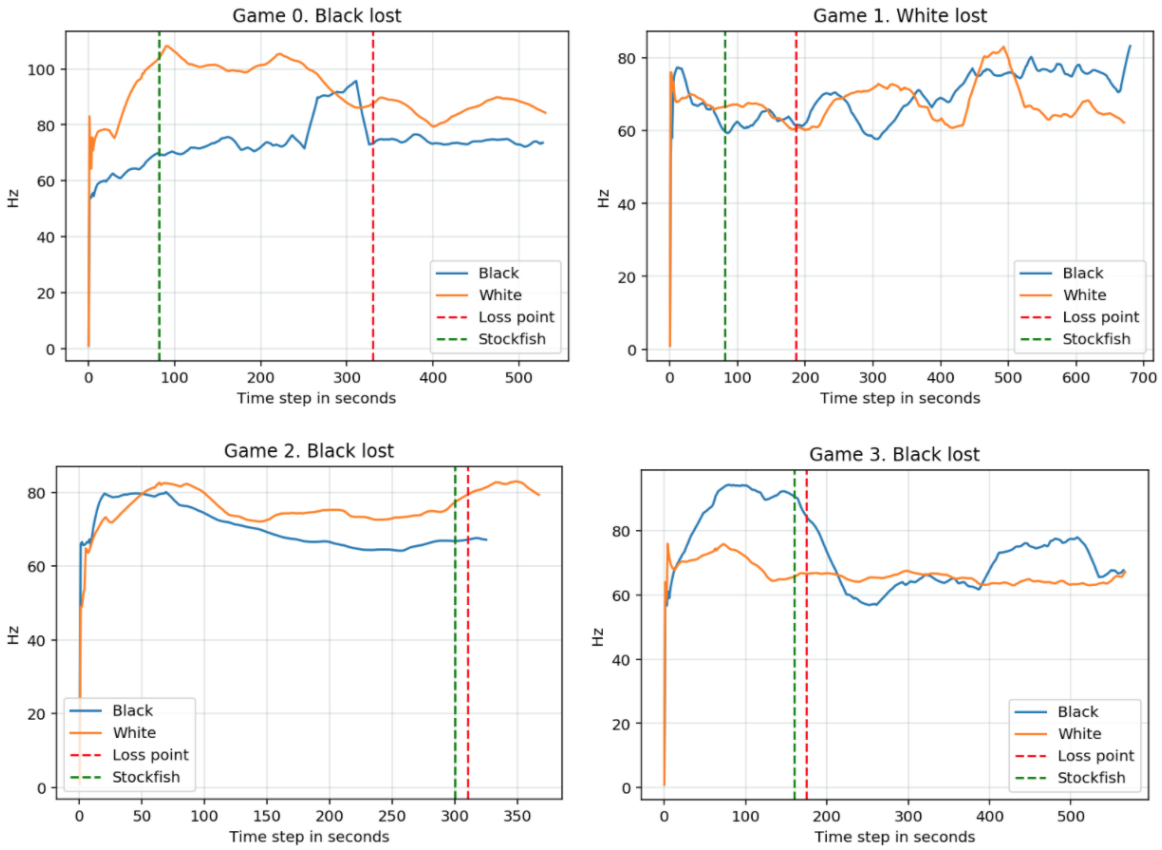

Figure 1.1: Heart rate data for each player for each game, indicating the self-reported loss point.

No clear pattern in heart rate surrounding the loss point appears to reveal itself from this small data set. From looking at the graphs, the closest thing to a correlation seems to be a prominent drop in heart rate immediately following the loss point in Game 3. Recognising that there may be small inaccuracies in our synchronisation of timestamps, the sharp downward-sloping spike in the losing player's heart rate immediately preceding the loss point in Game 0 appears to be similar. Because this pattern does not recur in the other two games, however, this can only be speculative.

The close proximity of the reported and Stockfish's projected "loss point" for games 2 and 3 is nevertheless worth noting. This consistency suggests that players are self-aware enough that their reporting constitutes a valid source of information about the point of interest – and that if a relationship between heart rate and the realisation of having lost a match exists, then such a self-reporting system would be an effective way to find it.

1.2.2 Action unit analysis

In order to extract data about which action units (AUs) were demonstrated most prominently during the labelled loss point, we used the Facial landmark detection[7][8] and Facial Action Unit

detection[9] provided in OpenFace[3], an open-source facial behavior analysis toolkit[10]. Taking the raw video recording of each losing player's face, this provided as output for every frame, the degree of activation of 17 different AUs (on a scale from 0 to 5). We parsed this data, and aggregated the AU output for all frames in the five seconds immediately following the "loss point", to try to ascertain a pattern in facial expressions. These results are summarised in figure 1.2.

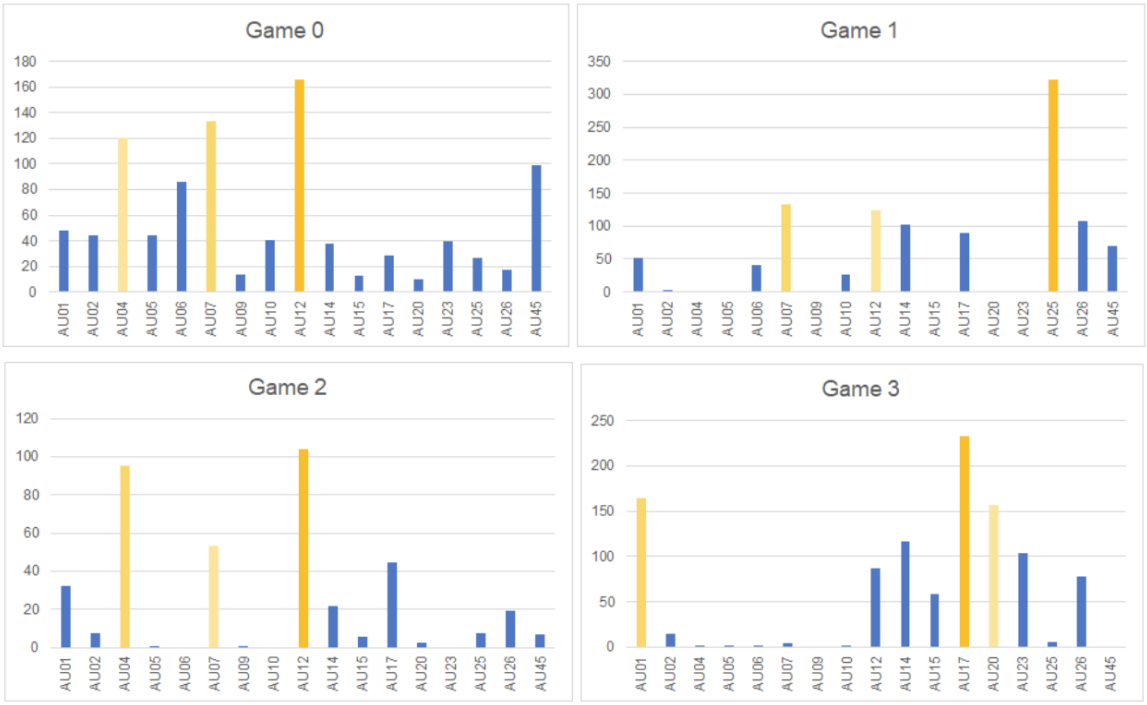

Figure 1.2: Sum of AU activation for the losing player for all frames of our video recording in the five seconds following the "loss point". The three most prominent AUs are highlighted in shades of orange.

The only discernible similarity here seems to be between games 0 and 2, with AUs 4, 7 and 12 being the most prominent AUs. This is worth noting especially since the recording was of two different players.

Using the EMFACS system of mapping AU activation to emotions[11], we can attempt an analysis of the emotional reaction these correspond to. Considering games 0 and 2, we note that AU4 and AU7 are involved in expressions of both anger and fear; activation AU12 can be associated with contempt as well. The data for the other two games do not appear to indicate a consistent emotion (either independently or considered together).

Ultimately, however, our sample size is far too small to discern a valid pattern from. If we had been able to perform the experiment more widely and proceeded to use the data gathered to train

[3] `https://github.com/TadasBaltrusaitis/OpenFace`

a recognition model, we would have used the heart rate data and AU activation in the few seconds immediately following the loss point as input features.

Chapter 2

Creating the ARS

2.1 Data Description

For part two of the mini group project, our team has taken the Movement Behaviour Classification task from the EmoPain 2020 challenge. The aim of the Affect Recognition System is to classify and detect protective behaviours, such as guarding and stiffness, of people who are experiencing chronic lower back pain (CLBP)[12][13].

Our team is given training and validation data , which contain multimodal movement data collected from healthy and CLBP participants performing physical activity[14]. The training data is collected from 16 CLBP participants and 7 controlled participants, and the validation data consists of data from 5 controlled participants and 7 CLBP patients. The movement data for each participant is stored in a separate data file represented in a TxF matrix, where T denotes the number of dataframe for the file, and F denotes the number of columns in the file. The feature corresponding to each column is described in table 1. The ground truth of the data is the continuous binary protective behaviour label professionally annotated by physiotherapists and psychologists.

Our code is available at: `https://github.com/chengmar09/comp0053`

2.2 Class Imbalance

A first inspection of the two datasets suggests a significant class imbalance with a bias towards the non-protective (0) class. This problem will influence various decisions about the training procedure. Firstly, the choice of algorithm may reinforce the found bias. Secondly, proper evaluation metrics have to be selected. To mitigate the effects of this class imbalance, the dataset was pre-processed.

The team experimented with two pre-processing techniques: 1) removing the controlled datasets; 2) rebalancing the dataset by truncating the start and the end of the individual session files. Since all features start and end with a protective behaviour of 0, this is a simple method to address the imbalance without having to inspect each session file individually. The pre-processing technique is applied only to the training data. Prior to pre-processing the number of batches with protective labels , the proportion of protective batches is 0.1870 (6656 protective frames out of 35608 frames). After the pre-processing, the proportion increases to 0.3252 (6656 out of 20471).

Another attempt to balance the dataset was made through synthetic data generation. At first, the distributions for each individual feature was calculated. Sampling from these distributions provides a simple way to generate data. However, this approach does not take into account possible dependencies between features. To address this issue a multivariate gaussian distribution was fitted on the whole feature set. A test run using the original data and an additional 10000 positive-labelled samples did not provide any increase in performance, though. In this scenario the data is not treated as sequential. Thus, the quality of the synthetically generated samples is poor.

To properly generate data for this problem, the sequentiality of the data has to be taken into account. Common architectures for this approach are GANs and LSTMs. As a starting point, the baseline stacked bidirectional Long Short Term Memory model was modified. The results of this experiment can be seen in the appendix. The synthetic data from this model does not seem to represent the original data well. Further experiments into sequential data generation are needed to utilise them for training.

Following the EmoPain Challenge 2020 implementation, the team uses the macro average F1 score and the F1 score for each class (i.e. protective and non- protective to assess the performance[2]. Due to the imbalance of the data, metrics such as accuracy are not representative, as the algorithm may learn to simply predict the more common class and still obtain high accuracy. F1 score is able to evaluate the class-wise performance by considering both recall and precision.

2.3 Feature selection methods

Three feature selection methods were investigated, namely the Principal Component Analysis (PCA), Analysis of Variance (ANOVA), and Random Forest Classifier.

The PCA[1] orthogonally transforms the original data coordinates into principal components. Succeeding principal components have the greatest possible variance, with the limit that they are uncorrelated with the preceding components. In our experiment, the number of components n is selected to keep 99% of the variance of each feature. Keeping only the first n components reduces the data dimensionality while retaining most of the information. Prior to using PCA, features are standardised such that the features are rescaled to have a standard normal distribution where mean μ is 0 and standard deviation σ is 1.

Analysis of Variance (ANOVA) is a statistical method for evaluating whether a continuous feature has an impact on the categorical label. Using scikit-learn[2], this approach selects the k top features that have the highest score from the one way ANOVA F-test. These features are statistically concluded to have an impact on the label and will be included for the model training. The team used empirical analysis to find out the best k for each combination of features.

Random Forest Classifier[3] is an ensemble-based estimator for computing the feature importance score; it is calculated using feature usage statistics in the forest- if a feature is often selected as best

[1] https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html
[2] https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectKBest.html
[3] https://scikit-learn.org/stable/modules/feature_selection.html

split, it is more likely an informative feature. The team experimented with constructing the forest with different numbers of trees (10,30,and 100), and they have consistent selection of features.

The team explores four combinations of modalities: 1) Joint angles, joint energies and sEMG, which contains 30 initial features; 2) Joint angles and joint energies, with 26 original features; 3) Joint energies and sEMG, which contains 17 features; 4) Joint angles and sEMG which also contains 17 features. During each experiment, a fusion of modality is chosen. Feature selection is then applied to reduce the dimensionality of the input features and selects the most relevant ones for model training. The experiment is run multiple times and the average performance across the runs is recorded.

2.4 Baseline Model

The team implemented a stacked Long Short Term Memory model following the baseline model described by Wang et al.[15][13]. The LSTM is a variant of recurrent neural networks (RNN) which is able to process long temporal and sequential data, and learn long term dependencies. The memory cells in the hidden layer have a recurrent weight of 1 that prevents the vanishing gradient problem[16]. Following the results from Wang et al's experiment on the optimal number of LSTM layers and hidden units, we implemented a 3-layer network with 32 hidden units in each and a dropout rate of 0.5 to prevent the overfitting. As suggested by the baseline paper, in our implementation a sliding window of size 180 with an overlapping ratio of 0.75 is chosen. The ground truth of the window segment is found by majority voting. Since no test set was given, to train the model, we set the validation_split as 0.2 such that fractions of the training data will be used as validation data and be used for evaluating the loss and other metrics at the end of each epoch[4]. The given validation set was used for testing.

2.4.1 Baseline Performance

The baseline performances for each modality is described in table 2.1. No feature selection, data augmentation or pre-processing techniques were applied onto the data, with the exception of feature normalisation using min-max scaling. Our baseline differs from the Emopain Challenge 2020 because we split the data into batches and applied normalisation before feeding them into the LSTM model. Furthermore, we used a bi-directional LSTM rather than a stacked LSTM. Table 3 describes the performances of different combinations of modalities and feature selection.

[4] `http://ee104.stanford.edu/lectures/validation.pdf`

Modality	Class	Accuracy	F1 Score
Joint Angles, Energies and sEMG	Non-protective		0.9664
	Protective		-
	Average	0.4675	0.4850
Joint Angles and Energies	Non-protective		0.9664
	Protective		-
	Average	0.4675	0.4850
Joint Angles and sEMG	Non-protective		0.9653
	Protective		0.0088
	Average	0.4665	0.49
Joint Energies and sEMG	Non-protective		0.9665
	Protective		-
	Average	0.4674	0.485

Table 2.1: Baseline performances (accuracy and f1 score) of different combinations of modality.

2.5 Results Evaluation

In the following section, the team explores the differences between the features selected via each selection method, and their impact on the model performances.

2.5.1 Joint Angles, Energies and sEMG

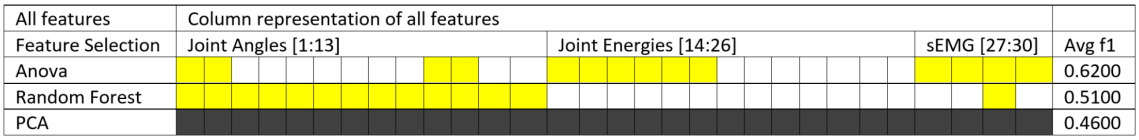

Figure 2.1: The figure illustrates the selected features (yellow) in the experiment using the Joint Angles, Energies and sEMG modalities. The average F1 score for the different feature selection methods is shown on the right.

Due to the changes in network architecture, the new measured baseline average f1-score for the Joint Angles, Energies and sEMG modality (30 features) is 0.4850. When the described pre-processing techniques are applied, it increases to 0.5200. It is observed that the non-protective class performs significantly better (0.9038) compared to the protective class (0.1373). The Anova classifier gives the best overall performances, with the average F1-score increasing to 0.6200. It selects features from all three modalities, including all of the sEMG data, six from Joint Energies and four from joint angles. Interestingly, the random forest selects 13 features, all of which are from the joint angles. Its performance is worse than when we use the features selected by Anova.

2.5.2 Joint Angles and Energies

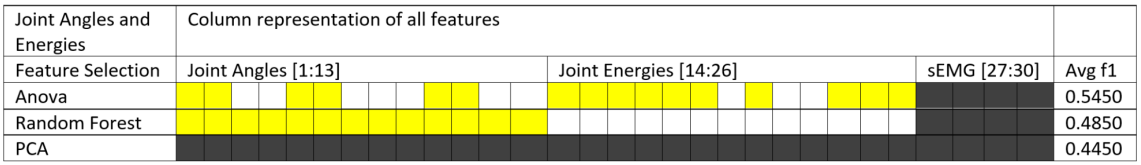

Figure 2.2: The figure illustrates the selected features (yellow) in the experiment using the Joint Angles and Energies modalities. The average F1 score for the different feature selection methods is shown on the right.

The new measured baseline average f1-score for the moCAP data is 0.4850, which is the same as when sEMG data is also taken into consideration. After preprocessing, the average f1 score increases to 0.5000. Again the non-protective class performs significantly better than the protective class, due to the imbalance of the dataset. The Anova classifier is the best feature selection method. It selects 16 features out of the 26 features available. In comparison, the Random Forest classifier selects all of its features from the first 13 joint angles columns, and the average f1-score slightly worse than the Anova classifier.

2.5.3 Joint Angles and sEMG

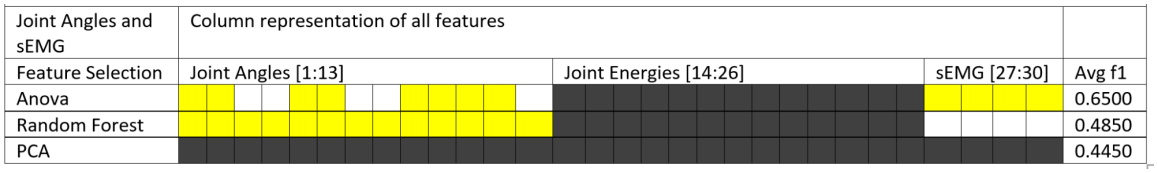

Figure 2.3: The figure illustrates the selected features (yellow) in the experiment using the Joint Angles and sEMG modalities. The average F1 score for the different feature selection methods is shown on the right.

This input set contains 17 features from Joint Angles and sEMG. It has a comparable baseline performance to when all 30 features are used, prior to and after the pre-processing. Before pre-processing, the baseline average f1 score is 0.4900, and after preprocessing, it increases to 0.5350. The best performance is obtained when Anova is applied ; the dimensionality is reduced to twelve features, with eight features from the Joint Angles and four from the sEMG. The average F1 score is 0.6500. In terms of the class-wise performance, the f1-score of the protective class is 0.3693, an increase of 0.1870 compared to when no feature selection is applied. The non-protective class has a score of 0.9302.

2.5.4 Joint Energies and sEMG

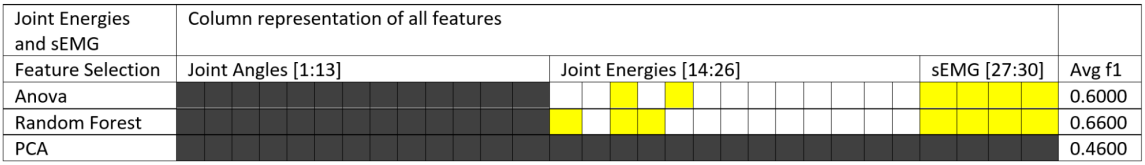

Figure 2.4: The figure illustrates the selected features (yellow) in the experiment using the Joint Energies and sEMG modalities. The average F1 score for the different feature selection methods is shown on the right.

This input set contains 17 features from Joint Energies and sEMG. The average F1 score before and after preprocessing are both 0.4850. Both Anova and Random Forest classifier work well with this subset of features. Random Forest Classifier selects seven features, from both joint Energies and sEMG. The average F1 score increases to 0.6600; the non-protective class has an f1-score of 0.9320, and the protective class scores 0.3886. This is the highest average f1 score, and protective-class F1 score we obtain from all the conducted experiments. Anova selects six features, 2 from the Joint Energies and 4 from the sEMG; the average F1 score increases to 0.6000.

2.6 Conclusion

In this report, we have first presented a small dataset for determining the lost point for a chess player with the player's heart rate and facial expression respectively. Secondly, we attempted the part III of the EMO pain challenge. Specifically, we performed a detailed data inspection on the data, and therefore performed various feature selection algorithms to reduce the redundant data. Our experiments of feature selection on our baseline show that PCA do not perform well, while Anova improves performance and random forest classifier indicates that the following combination of features could yield better results:

1. From the joint angles and sEMG modalities: the left and right full body flexion, left and right knees, left and right shoulder, and left and right lateral bends and the sEMG probes one to four.

2. From the joint energies and sEMG modalities: left full body flexion, left and right inner flexion, and the sEMG probes one to four.

Bibliography

[1] Interview on the dick cavett show with gary kasparov. https://www.youtube.com/watch?v=MPlXC3M8hbg&feature=youtu.be&t=203. Accessed 1 May 2020.

[2] Alireza firouzja vs magnus carlsen — world blitz 2019. https://www.youtube.com/watch?v=lmUgUetQBk8. Accessed 1 May 2020.

[3] The title game — magnus carlsen vs hikaru nakamura — world blitz 2019 playoff game 2. https://www.youtube.com/watch?v=d-7ZpPDrOOo&t=306s. Accessed 1 May 2020.

[4] Nepomniachtchi's funny expressions after offering a draw to carlsen — world blitz 2019. https://www.youtube.com/watch?v=4tWjmTrdVhU. Accessed 1 May 2020.

[5] Super fast opening play from dubov — magnus carlsen vs daniil dubov — world blitz 2019 —. https://www.youtube.com/watch?v=l2XK89YcPDA. Accessed 1 May 2020.

[6] Aishwarya Kumar. The grandmaster diet: How to lose weight while barely moving. www.espn.com/espn/story/_/id/27593253/why-grandmasters-magnus-carlsen-fabiano-caruana-lose-weight-playing-chess. Accessed 27 April 2020.

[7] Amir Zadeh, Yao Chong Lim, Tadas Baltrusaitis, and Louis-Philippe Morency. Convolutional experts constrained local model for 3d facial landmark detection. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 2519–2528, 2017.

[8] Tadas Baltrusaitis, Peter Robinson, and Louis-Philippe Morency. Constrained local neural fields for robust facial landmark detection in the wild. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 354–361, 2013.

[9] Tadas Baltrušaitis, Marwa Mahmoud, and Peter Robinson. Cross-dataset learning and person-specific normalisation for automatic action unit detection. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 6, pages 1–6. IEEE, 2015.

[10] Tadas Baltrusaitis, Amir Zadeh, Yao Chong Lim, and Louis-Philippe Morency. Openface 2.0: Facial behavior analysis toolkit. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 59–66. IEEE, 2018.

[11] Wallace V Friesen, Paul Ekman, et al. Emfacs-7: Emotional facial action coding system. *Unpublished manuscript, University of California at San Francisco*, 2(36):1, 1983.

[12] Karon F Cook, Toni S Roddey, Alyssa M Bamer, Dagmar Amtmann, and Francis J Keefe. Validity of an observation method for assessing pain behavior in individuals with multiple sclerosis. *Journal of pain and symptom management*, 46(3):413–421, 2013.

[13] Nadia Berthouze, Michel Valstar, Amanda Williams, Joy Egede, Temitayo Olugbade, Chongyang Wang, Hongyin Meng, Min Aung, Nicholas Lane, and Siyang Song. Emopain challenge 2020: Multimodal pain evaluation from facial and bodily expressions. *arXiv preprint arXiv:2001.07739*, 2020.

[14] Min SH Aung, Sebastian Kaltwang, Bernardino Romera-Paredes, Brais Martinez, Aneesha Singh, Matteo Cella, Michel Valstar, Hongying Meng, Andrew Kemp, Moshen Shafizadeh, et al. The automatic detection of chronic pain-related expression: requirements, challenges and the multimodal emopain dataset. *IEEE transactions on affective computing*, 7(4):435–451, 2015.

[15] Chongyang Wang, Min Peng, Temitayo A Olugbade, Nicholas D Lane, Amanda C De C Williams, and Nadia Bianchi-Berthouze. Learning bodily and temporal attention in protective movement behavior detection. *arXiv preprint arXiv:1904.10824*, 2019.

[16] Zachary C Lipton, John Berkowitz, and Charles Elkan. A critical review of recurrent neural networks for sequence learning. *arXiv preprint arXiv:1506.00019*, 2015.

Appendix A

Modality	Feature Selection Method	Class	F1 Score
Joint Angles, Energies, and sEMG	No Feature Selection	Non-protective	0.9038
		Protective	0.1373
		Average	0.5200
	PCA	Non-protective	0.7697
		Protective	0.1453
		Average	0.4600
	Anova	Non-protective	0.8821
		Protective	0.3611
		Average	0.6200
	Random Forest	Non-protective	0.9118
		Protective	0.1096
		Average	0.5100
Joint Angles, and Energies	No Feature Selection	Non-protective	0.9632
		Protective	0.0404
		Average	0.5000
	PCA	Non-protective	0.7940
		Protective	0.1026
		Average	0.4450
	Anova	Non-protective	0.9027
		Protective	0.1922
		Average	0.5450
	Random Forest	Non-protective	0.9664
		Protective	-
		Average	0.4850
Joint Angles, and sEMG	No Feature Selection	Non-protective	0.8868
		Protective	0.1823
		Average	0.5350
	PCA	Non-protective	0.7480
		Protective	0.1731
		Average	0.4600
	Anova	Non-protective	0.9302
		Protective	0.3693
		Average	0.6500
	Random Forest	Non-protective	0.9964
		Protective	-
		Average	0.4850
Joint Energies, and sEMG	No Feature Selection	Non-protective	0.9964
		Protective	-
		Average	0.4850
	PCA	Non-protective	0.7404
		Protective	0.1751
		Average	0.4600
	Anova	Non-protective	0.8788
		Protective	0.3218
		Average	0.6000
	Random Forest	Non-protective	0.9320
		Protective	0.3886
		Average	0.6600

Table A.1: Macro and class-wise f1-score when different feature selection method are applied to the different combinations of modality