Classificação de atividades no desktop a partir de padrões de rastreio ocular

Luiz Menon Jr.*
[1] *Departamento de Física, PUC-Rio, Rua Marquês de São Vicente 225, 22451-900, Rio de Janeiro, RJ, Brasil*
(Dated: 9/07/2022)

Métodos de inteligência computacional são uma ferramenta poderosa na identificação de padrões no comportamento humano. Neste trabalho são estudados os padrões gerados pela fixação de olhar frente a uma tela. Utilizando a metodologia de redes neurais artificiais multi layer perceptron classificamos diferentes atividades frente ao computador, por meio do rastreamento ocular feito por um equipamento específico para esta tarefa. Foram feitas classificações binárias e de múltiplas classes entre atividades utilizando primeiramente um e em seguida os vinte e quatro participantes do estudo.

Keywords: rastreamento ocular, redes neurais, classificação

I. INTRODUÇÃO

O padrão de olhar do ser humano pode revelar muito sobre o seu comportamento, saúde e emoções [1–3]. Os primeiros trabalhos a utilizarem o rastreamento ocular datam da década de 1970 no campo da cognição e percepção visual [4]. Mas apenas recentemente na última década o rastreamento ocular vem sendo combinado com técnicas de inteligência computacional para o estudo da e o reconhecimento da atividade humana [5]. Essa combinação de tecnologia vem se mostrado proeminente em diversas áreas do conhecimento, Podemos citar pesquisas feitas no rastreamento ocular para o diagnóstico de pessoas no espectro autista[6], Na analise da complexidade e compreensão de diferentes textos [7] e no reconhecimento de tomada de decisões [8]. Neste trabalho focou não no reconhecimento de padrões individuais como os citados anteriormente, mas sim na identificação de padrões coletivos em diversas atividades feitas no desktop. Para esta identificação utilizamos a metodologia de redes neurais multilayer perceptron (MLP) e dados extraídos de estudos anteriores[9]

II. BASE DE DADOS

A base de dados utilizados neste trabalho é provinda de um estudo anterior [9] onde os autores identificam e contabilizam dois diferentes padrões de olhar em diferentes atividades executadas em desktop. Além dos já estabelecidos padrões de baixo nível [10, 11], Srivastava e colaboradores [9] definem os padrões chamados de médio nível. Esta base de dados conta com conjuntos de dados de 24 participantes executando diferentes atividades em frente a um computador (desktop). Cada participante realizou 5 atividades consideradas comuns e mais 3 especificas de profissionais que atuam em áreas de desenvolvimento de software(DS) [9]. Os dados coletados formam séries temporais bidimensionais mostrando a localização do olhar na tela do computador contendo o valor de x e y referente a fixação do olhar na tela e o tempo transcorrido até o próximo ponto. Uma descrição das atividades é comentada a seguir, aqui foi decidido manter os rótulos originais das atividades na língua inglesa.

Read: Nesta atividade é considerada a *leitura em tela*, em que os participantes eram apresentados a três diferentes materiais de leitura, trechos de livros, artigos ou contos. ***Watch:*** Cada participante é convidado *assistir* um video em tela cheia, cada video tinha variação entre um, dois ou três personagens e tinha duração média de cinco minutos.Estes pequenos filmes variavam entre uma animação em preto e branco, outra animação em cores e um filme independente. ***Browse:*** Os participantes eram livres para *navegar* na internet, na grande maioria, navegaram em sites de notícias. Também escolheram navegar em sites escritos em sua primeira língua, o que tornou grande a variedade de padrões para esta atividade. ***Search:*** Os participantes tinham a tarefa de *buscar* a resposta para perguntas elaboradas anteriormente em ferramentas de busca online. Algumas delas eram automaticamente respondidas pela ferramenta, outras demandavam algum esforço maior. ***Play:*** Nesta atividade cada participante precisava *jogar* um simples jogo no computador. Antes dos dados começarem a ser adquiridos os participantes passavam por um pequeno treino de um minuto sendo utilizados três diferentes jogos. ***Interpret:*** Esta é a primeira das atividades específica na área de DS. Aqui os participantes precisavam *interpretar* uma função programada em uma linguagem de programação e responder qual era a saída apos este código ser executado. ***Debug:*** Na segunda atividade de DS, os participantes tinham como objetivo consertarem trechos de códigos que continham vários bugs. ***Write:*** Na última tarefa ligada ao DS os participantes eram instruídos a *escrever* alguns códigos em ordem crescente de complexidade. Os códigos incluíam: imprimir na tela o produto de conjuntos numéricos, imprimir os dez pri-

* Correspondence email address: luizmenonjr@gmail.com

meiros números da sequência de Fibonacci e por último implementar um algorítimo de busca e ordenamento.

III. METODOLOGIA DE CLASSIFICAÇÃO

A identificação e reconhecimento do comportamento e da atividade humana é formulado como um problema típico de classificação [12]. Dentro desse contexto podem ser empregadas várias técnicas de classificação, via algorítimos inteligentes, dos quais é possível citar SVM, k-NN, AdaBoostin, bagging, arvores de decisão e Random Forest [13]. Neste trabalho foi utilizada a metodologia de redes neurais multilayer perceptron (MLP), os detalhes sobre o funcionamento de uma rede MLP pode ser encontrado na ref. [14]. Neste manuscrito apenas arquitetura da rede, entradas e saídas, e hiperparemetros serão descritos em detalhes.

Neste trabalho foram realizadas 4 categorias de análises que foram dividas em classificação binária e classificação multi-classe, utilizando apenas 1 participante e também para os 24. Primeiramente foram diferenciadas duas atividades, em um primeiro momento foram diferenciadas as atividades *Read* e *Write*, em seguida foram avaliadas *Read* e *Watch*. Em seguida a classificação foi feita para todas as atividades. Para melhores resultados, por último, foram consideradas classes que eram melhores identificadas.

A. Preparação dos padrões

Os padrões de cada atividade são extraídos das séries temporais da base de dados. O procedimento foi definir uma janela de pontos, $\mathbf{Z}_i = (x_1, y_1),\ (x_2, y_2),\ (x_3, y_3) \ldots,\ (x_{nj}, y_{nj})$, seguindo o ordenamento da série temporal, n_j é o tamanho da janela. Os vetores $\mathbf{Z_i}$ servirão como entradas da rede neural, com i indo de 1 até o número total de padrões. Os alvos são definidos por T_i, no caso binário T assume dois valores, 0 ou 1, para a caso de múltiplas classes T assume valores de 0 a 7. Vale deixar claro que os pontos presentes em um padrão não se repetem em outros padrões, diz se, que a janela é não deslizante, prática comum em problemas de previsão de séries temporais [15]. Para as classificações binárias com 1 e 24 participantes e multiclasse com 1 participante foi utilizada à normalização MinMax. já no caso de múltiplas classes para 24 participantes a normalização normal foi a utilizada (A1).

B. Arquiteturas de MLP

Para as classificações foram usadas diferentes arquiteturas. no caso binário foram utilizadas redes MLP semelhantes, com entradas iguais ao número de janela de padrões n_j, com uma camada escondida com função de ativação sigmoide de 64 processadores e dois processadores na camada de saída com função de ativação linear. A função de perda escolhida foi a entropia cruzada e o gradiente descendente estocástico como algorítimo de aprendizagem. As épocas de treinamento, taxa de aprendizagem e tamanho de batch são definidos na sessão de resultados.

Para as classificações de múltiplas classes foram utilizadas duas arquiteturas de MLP. Primeiramente a rede neural foi similar as utilizadas nas classificações binárias, na etapa classificação de atividades para apenas 1 participante. Para o caso onde foram considerados todos os 24 uma rede de apenas uma camada não foi suficiente para esta tarefa então foi utilizada uma arquitetura com três camadas escondidas de $28 \times 14 \times 10$ processadores, desta vez utilizando funções de ativação tangente hiperbólica ($tanh$).

IV. CLASSIFICAÇÃO BINÁRIA

As classificações binárias foram feitas escolhendo três atividades, *Read*, *Write* e *Watch*. Primeiramente avaliamos a classificação entre *Read* e *Write* para 1 participante em seguida para os 24. Após isso as atividades comparadas foram *Read* e *Watch*, também primeiramente para 1 em seguida para os 24 participantes.

A. Read e Write 1 participante

Para 1 participantes a rede neural descrita na sessão anterior foi treinada por 200 épocas de treinamento em batchs contendo 64 padrões. A evolução da acurácia do conjunto de validação é mostrada na parte superior da Fig.(1) e a evolução da perda em relação ao conjunto de validação na parte inferior da Fig.(1).

Na Tabela I é mostrado as métricas de avaliação do modelo, mediante ao conjunto de teste, pode-se perceber um bom resultado na classificação que teve uma acurácia superior a 95% e f1-score também acima de 96%. Já na 2 é mostrada a matriz de confusão para esta classificação, nota-se que para classe *Read* apenas dois padrões são identificados como *Write* este é fato é refletido no alto valor de sensibilidade presente na Tabela(I)

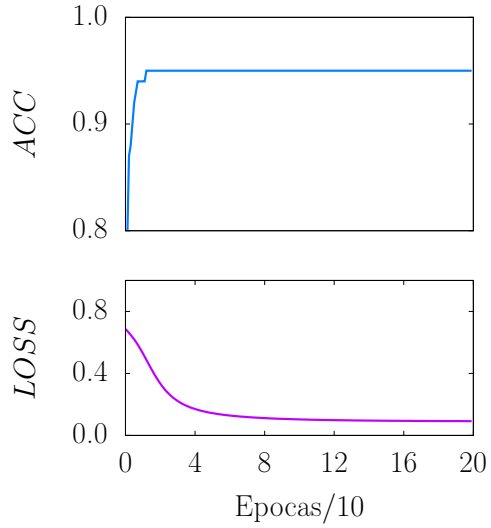

Figura 1. (Painel superior) Evolução da acurácia em relação ao conjunto de validação durante o treinamento em 200 épocas de treino. (Painel inferior) Evolução da perda em relação ao conjunto de validação durante o treinamento em 200 épocas de treino.

	Precisão	Sensibilidade	f1-score	Suporte
Read	0.93	0.99	0.96	192
Write	0.99	0.93	0.96	208
Acurácia			0.96	400

Tabela I. Métricas de avaliação para classificação entre *Read* e *Write* de 1 participante.

B. Read e Write 24 participante

Na análise de classificação entre *Read* e *Watch* considerando os 24 participantes os resultados são similares ao caso de apenas 1 participante. O modelo perde em acurácia, tabela (IV B), em comparado com o treinado e testado com apenas 1 participantes, mas mantém bos métricas, vale notar que a sensibilidade para a categoria *Read* se torna 1.00, ou seja, poucos padrões que originalmente são *Write* são medidos como *Read* em comparação aos acertos da classe. Isto também evidenciado pela matriz de confusão 4

C. Read e Watch 1 participante

Agora foi escolhido para a análise duas atividades consideradas comuns em [9], que não envolvem DS, *Read* e *Watch*. Aqui foram feitas análises para o tamanho de janela, n_j, de modo a verificar se aumento desse número causa uma melhora na classificação e nas respectivas

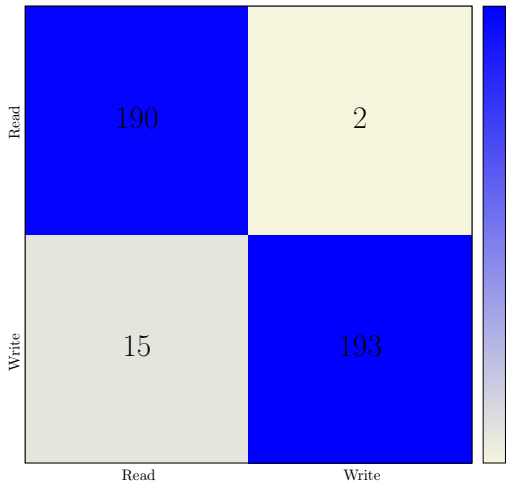

Figura 2. Matriz de confusão para classificação binária entre *Read* e *Write* para 1 participante. O eixo y representa a categoria real e o eixo x representa a categoria prevista.

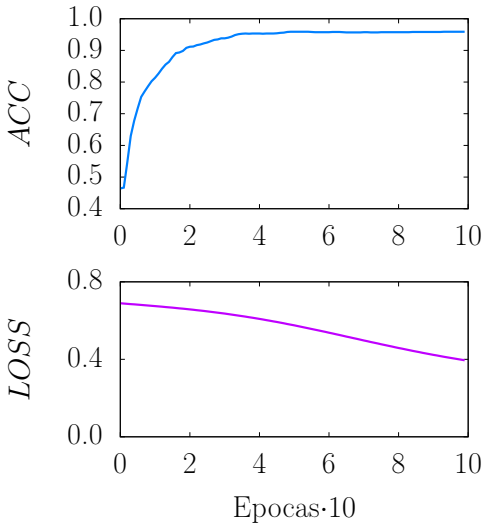

Figura 3. Classificação binária entre *Read* e *Write* para 24 participantes.(Painel superior) Evolução da acurácia em relação ao conjunto de validação durante o treinamento em 100 épocas de treino. (Painel inferior) Evolução da perda em relação ao conjunto de validação durante o treinamento em 100 épocas de treino.

métricas. Conforme a tabela IV C, a acurácia total do modelo permanece em torno de 70% exceto no caso em que $n_j = 5$, em que a acurácia cai para 60%. Em razão dessa homogeneidade da acurácia em relação à variação de tamanho de janela, é factível avaliar o modelo através da métrica f1-score, assim concluí-se que melhor escolha para o tamanho de janela é de $n_j = 7$.

	Precisão	Sensibilidade	f1-score	Suporte
Read	0.92	1.00	0.95	2854
Write	0.99	0.91	0.95	2894
Acurácia			0.95	5748

Tabela II. Métricas de avaliação para classificação entre *Read* e *Write* de 24 participantes.

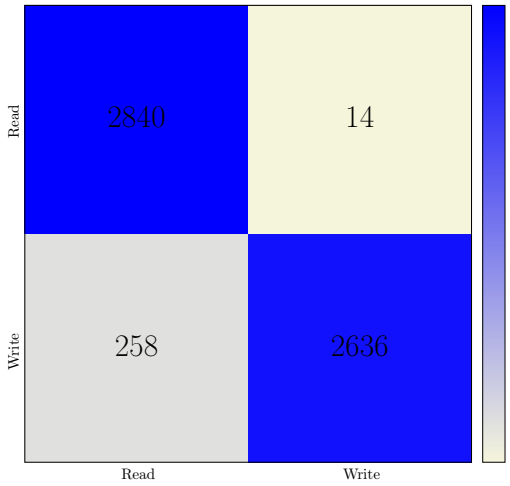

Figura 4. Matriz de confusão para classificação binária entre *Read* e *Write* para 24 participantes. O eixo y representa a categoria real e o eixo x representa a categoria prevista.

Janela		Precisão	Sens.	f1-score	Suporte
$n_j = 4$	*Read*	0.66	0.93	0.77	304
	Write	0.84	0.42	0.56	256
	Acur.			0.70	560
$n_j = 5$	*Read*	0.65	0.86	0.74	234
	Write	0.76	0.49	0.60	214
	Acur.			0.68	448
$n_j = 6$	*Read*	0.66	0.83	0.73	187
	Write	.77	0.57	0.65	214
	Acur.			0.70	373
$n_j = 7$	*Read*	0.64	0.87	0.74	157
	Write	. 0.82	0.55	0.66	166
	Acur.			0.70	323

Tabela III. Métricas de avaliação para classificação entre *Read* e *Watch* para 1 participante com variação no tamanho de janela de padrões.

D. Read e Watch para 24 participantes

Embora, para o caso de 1 participante, a classificação entre *Read* e *Write* uma janela de $n_j = 7$ foi a escolha com melhor desempenho, isto não acontece ao considerarmos os 24 participantes. A melhora escolha para este hiper-parâmetro foi de $n_{=}5$, mediada através de diversos testes. A evolução de perdas e acurácia para esta classificação é mostrada na Figura(5), também é possível notar dois regimes de treino, um até aproximadamente a época 240 onde o crescimento da acurácia é pequeno e perda decai lentamente. Após esta época temos um crescimento repentino na acurácia e um decaimento acentuado nas perdas.

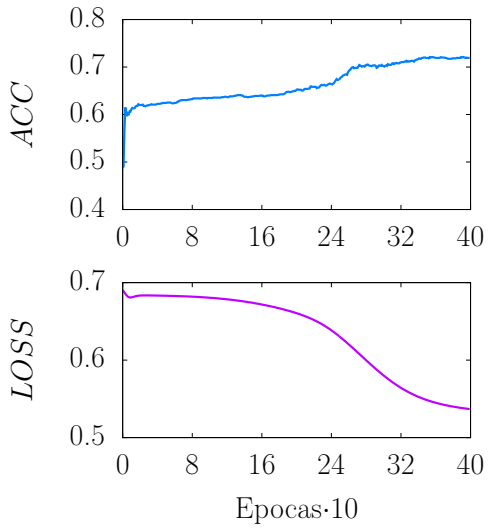

Figura 5. Classificação binária entre *Read* e *Watch* para 24 participantes (Painel superior) Evolução da acurácia em relação ao conjunto de validação durante o treinamento em 400 épocas de treino. (Painel inferior) Evolução da perda em relação ao conjunto de validação durante o treinamento em 400 épocas de treino.

	Precisão	Sensibilidade	f1-score	Suporte
Read	0.66	0.84	0.74	2311
Watch	0.78	0.56	0.65	2289
Acurácia			0.70	4600

Tabela IV. Métricas de avaliação para classificação entre *Read* e *Watch* de 24 participantes.

A partir da matriz confusão Fig. 6 é possível perceber maiores erros na identificação da atividade *Watch* em geral *Read* é bem detectada em ambos os casos. A acurácia desta classificação ficou em 70%.

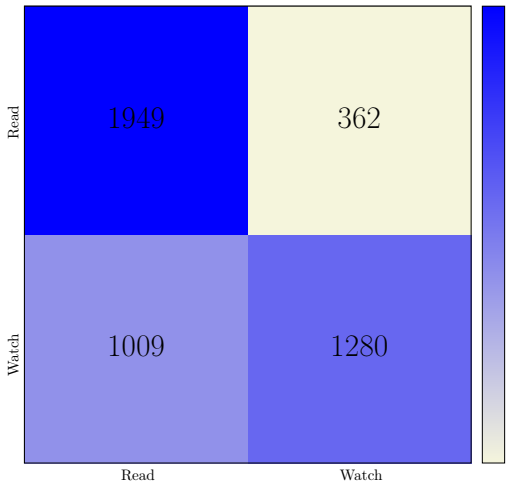

Figura 6. Matriz de confusão para classificação binária entre *Read* e *Watch* para 24 participantes. O eixo y representa a categoria real e o eixo x representa a categoria prevista.

V. CLASSIFICAÇÃO DE MÚLTIPLAS CLASSES

Nesta sessão o objetivo era tentar diferencia as 8 oito atividades, para ambos os casos 1 e 24 participantes algumas atividades causavam ruido nas demais então após uma primeira análise foram deixadas de lado. Ademais a classificação de múltiplos classes para todos os participantes requiriu uma modificação da arquitetura da rede MLP aumentando sua complexidade.

A. Classificação 1 participante

Para todas as atividades as redes neurais necessitaram de mais tempo de treinamento, passando para em média 1000 épocas de treinamento.

A acurácia total calculada para o conjunto de teste chegou a 45% conforme a tabela V. As classes que obtiveram pior desempenho segundo a matriz de confusão Fig.(7), também olhando para o f1-score da tabela V, foram *Watch*, *Search* e *Browse*. O procedimento a partir daqui foi de retirar estas classes patológicas de modo a obter uma melhor classificação. Após a exclusão das três classes citadas no parágrafo anterior o modelo obteve uma melhora geral em todas suas métricas, como observado na tabela VI. A acurácia total sobe para 62% e a classe com maior precisão é a *Read*. Visualmente isto é evidenciado através da inspeção da matriz confusão Fig.(9).

	Precisão	**Sensibilidade**	**f1-score**	**Suporte**
Browse	0.47	0.31	0.38	241
Debug	0.42	0.59	0.49	212
Inter.	0.55	0.60	0.57	230
Play	0.37	0.68	0.48	210
Read	0.46	0.51	0.48	218
Search	0.42	0.15	0.22	212
Watch	0.49	0.23	0.31	220
Write	0.49	0.54	0.52	217
		Acuraccy	0.45	1760

Tabela V. Métricas de avaliação para classificação entre as 8 classes de atividades de 1 participante.

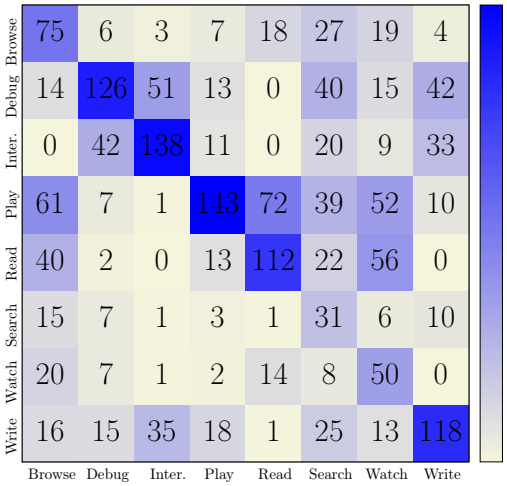

Figura 7. Matriz de confusão para classificação de múltiplas classes entre para 1 participante. O eixo y representa a categoria real e o eixo x representa a categoria prevista.

B. Classificação com 24 participantes

Vale a pena comentar que análise para os dados de todos os 24 participantes em todas as classes a arquitetura de MLP com apenas uma camada escondida mostrava um desempenho pobre de acurácia em torno de 35%. Neste cenário a solução foi utilizar mais camadas escondidas, neste caso foram 3. Após as modificações da rede, os resultados foram semelhantes ao que foram obtidos na classificação utilizando os dados de apenas 1 participante, com 45% de acurácia. Novamente existem classes patológicas, que atrapalham as demais classificações em geral, procedendo como anteriormente no caso de 1 participantes estas classes são deixadas de

	Precisão	Sensibilidade	f1-score	Suporte
Debug	0.65	0.48	0.55	183
Inter.	0.55	0.64	0.59	169
Play	0.61	0.47	0.53	159
Read	0.72	0.81	0.76	192
Write	0.56	0.68	0.61	177
		Acurácia	0.62	880

Tabela VI. Métricas de avaliação para classificação entre as 5 classes de atividades de 1 participante. Exclusão de classes patológicas

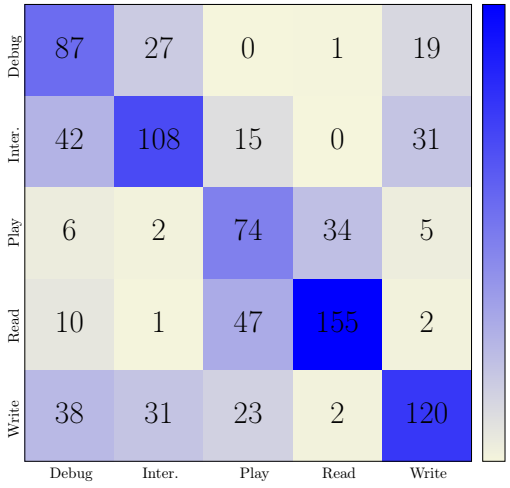

Figura 8. Matriz de confusão para classificação de múltiplas classes entre para 1 participante excluindo classes patológicas. O eixo y representa a categoria real e o eixo x representa a categoria prevista

lado. Entretanto, não são as mesmas do último caso, agora foram excluídas *Debug*, *Browse* e *Search*, escolhidas conforme o f1-score da Tabela VIII.

Com exclusão das classes citadas acima, o desempenho da classificação melhorou com uma acurácia de 64%.

VI. CONCLUSÕES

Foram realizadas classificações binárias e multiclasse tanto para um praticante quanto para os 24, Classificações binárias tiveram bons indicadores em ambos os casos utilizando uma arquitetura de MLP com uma camada escondida. A Classificação multiclasse para um participante melhoram seus indicadores após a remoção

	Precisão	Sensibilidade	f1-score	Suporte
Browse	0.39	0.34	0.36	793
Debug	0.38	0.26	0.31	768
Inter.	0.42	0.63	0.50	779
Play	0.49	0.50	0.50	760
Read	0.51	0.80	0.62	790
Search	0.36	0.18	0.24	833
Watch	0.54	0.47	0.50	801
Write	0.41	0.41	0.41	805
	Acurácia	0.45		6329

Tabela VII. Métricas de avaliação para classificação entre as 8 classes de atividades de 24 participantes.

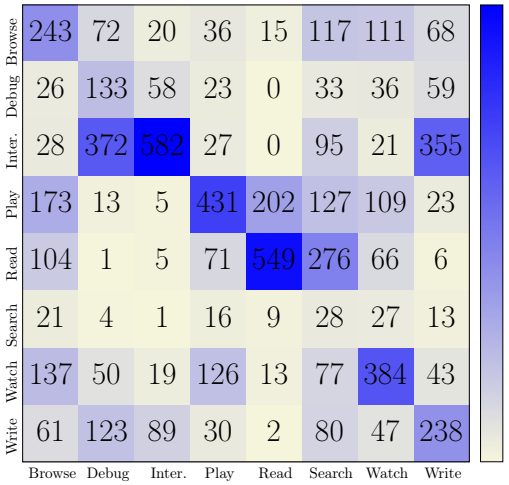

Figura 9. Matriz de confusão para classificação de múltiplas classes entre para os 24 participantes. O eixo y representa a categoria real e o eixo x representa a categoria prevista

	Precisão	Sensibilidade	f1-score	Suporte
Inter.	0.60	0.77	0.67	788
Play	0.58	0.63	0.60	790
Read	0.76	0.77	0.77	797
Watch	0.69	0.55	0.61	793
Write	0.60	0.48	0.54	788
	Acurácia		0.64	3956

Tabela VIII. Métricas de avaliação para classificação entre as 5 classes de atividades de 24 participantes. Excluindo classes patológicas

de classes ruidosas. Por último Para classificação multiclasse para todos os participantes foi preciso mudar a

arquitetura da rede neural adicionando mais duas camadas escondidas e mudando funções de ativação, além disso, também foram extraídas classes patológicas.

VII. REFERÊNCIAS

[1] O. Špakov, H. Istance, K.-J. Räihä, T. Viitanen, and H. Siirtola, in *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications* (2019) pp. 1–9.

[2] J. Z. Lim, J. Mountstephens, and J. Teo, Sensors **20**, 2384 (2020).

[3] Y. Mao, Y. He, L. Liu, and X. Chen, Frontiers in Neuroscience **14**, 798 (2020).

[4] A. Peshkovskaya and M. Myagkov, Frontiers in behavioral neuroscience **14**, 525087 (2020).

[5] M. G. Glaholt and E. M. Reingold, Journal of Neuroscience, Psychology, and Economics **4**, 125 (2011).

[6] Z. Boraston and S.-J. Blakemore, The Journal of physiology **581**, 893 (2007).

[7] D. Torres, W. R. Sena, H. A. Carmona, A. A. Moreira, H. A. Makse, and J. S. Andrade Jr, PloS one **16**, e0260236 (2021).

[8] J.-C. Rojas, J. Marín-Morales, J. M. Ausín Azofra, and M. Contero, Frontiers in Psychology **11**, 570470 (2020).

[9] N. Srivastava, J. Newn, and E. Velloso, Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **2**, 1 (2018).

[10] L. Itti and C. Koch, Nature reviews neuroscience **2**, 194 (2001).

[11] M. Rubo and M. Gamer, Scientific reports **8**, 1 (2018).

[12] O. Majidzadeh Gorjani, R. Byrtus, J. Dohnal, P. Bilik, J. Koziorek, and R. Martinek, Sensors **21**, 6207 (2021).

[13] A. Gupta, K. Gupta, K. Gupta, and K. Gupta, in *2020 international conference on communication and signal processing (ICCSP)* (IEEE, 2020) pp. 0915–0919.

[14] H. Simon, *Neural networks: a comprehensive foundation* (Prentice hall, 1999).

[15] T. Koskela, M. Lehtokangas, J. Saarinen, and K. Kaski, in *Proceedings of the world congress on neural networks* (Citeseer, 1996) pp. 491–496.

Apêndice A: Normalizações

A normalização *MinMax* ,$X : A \to A'$, que leva um elemento $X \in A$, do conjunto A não normalizado para um conjunto normalizado $A' \in [0, 1]$ é definida por

$$X' = \frac{X - \min\{A\}}{\max\{A\} - \min\{A\}}. \tag{A1}$$

. A normalização normal em que leva os conjuntos estar distribuídos em uma distribuição normal com média 0 e desvio padrão igual a 1 é definida por

$$x' = \frac{x - \langle \mathbf{x} \rangle}{\sqrt{\langle \mathbf{x^2} \rangle - \langle \mathbf{x} \rangle^2}} \tag{A2}$$