

Cadeira BMT 2024

Ruan Felipe¹ and Natanael Luciano¹

¹Programa de Engenharia de Sistemas e Computação
COPPE - Universidade Federal do Rio de Janeiro

23 de junho de 2024

Resumo

Dentre as diversas atividades realizadas pelos funcionários da secretaria de um curso de graduação e pós-graduação, podemos citar a orientação aos alunos quanto aos documentos necessários para dar entrada em algum processo específico, seja emissão de diploma, aproveitamento de grau e conceito, emissão de certificados, entre outros. No entanto, o grande volume de e-mails inviabiliza a agilidade do atendimento. Para enfrentar esse desafio, propomos um sistema de resposta automatizada que utiliza de técnicas de processamento de linguagem natural como *bag of words*, *Word2vec* para identificar e responder as solicitações de abertura de processos com mensagens padronizadas, informando os documentos necessários para a abertura de processos.

Palavras-chave: *bag of words*, *Word2vec*, Modelos de Linguagem, e-mail, automatização.

Abstract

Among the various activities carried out by the staff of the secretariat of an undergraduate and postgraduate course, we can mention the guidance to students regarding the documents necessary to initiate a specific process, whether it is diploma issuance, credit and grade recognition, certificate issuance, among others. However, the large volume of e-mails hinders the agility of the service. To address this challenge, we propose an automated response system that uses natural language processing techniques such as *bag of words*, *Word2vec* to identify and respond to requests for process initiation with standardized messages, informing the necessary documents for the process.

Keywords: *bag of words*, *Word2vec*, Language Models, e-mail, automation.

Sumário

1	Introdução	4
2	Base de Dados	4

3	Modelos e Experimentos	8
4	Resultados	8
4.1	Primeira mensagem automatizada:	8
5	Conclusão e Trabalhos Futuros	9
A	Matriz de confusão para treinamento em T1	10
B	Matriz de Covariância para treinamento em T2	12

Lista de Figuras

1	Distribuição das Categorias de Requerimento dos e-mails.	5
2	Exemplo de mensagem de e-mail pertencente a duas categorias	5
3	Distribuição das Categorias dos e-mails utilizadas para treinamento. A categoria Outros correspondem aos demais e-mails e não são utilizadas no treinamento	6
4	Exemplo de mensagem de e-mail fora da base de dados e sua classificação	9
5	Matriz de confusão para SVM treinada em <i>T1</i>	10
6	Matriz de confusão para RFC treinada em <i>T1</i>	10
7	Matriz de confusão para Gradient Boost treinada em <i>T1</i>	11
8	Matriz de confusão para MLP treinada em <i>T1</i>	11
9	Matriz de confusão para SVM treinada em <i>T2</i>	12
10	Matriz de confusão para RFC treinada em <i>T2</i>	12
11	Matriz de confusão para Gradient Boost treinada em <i>T2</i>	13
12	Matriz de confusão para MLP treinada em <i>T2</i>	13

1 Introdução

Dentre as diversas atividades realizadas pelos funcionários da secretaria de um curso de graduação e pós-graduação, podemos citar a orientação aos alunos quanto aos documentos necessários para dar entrada em algum processo específico, seja emissão de diploma, aproveitamento de grau e conceito, emissão de certificados, entre outros. No entanto, o grande volume de e-mails inviabiliza a agilidade do atendimento, em particular, durante períodos de alta demanda de emissão de documentos por parte do corpo discente podendo gerar atrasos de tarefas essenciais para a atividade acadêmica como alocação de salas de aulas e professores. Para enfrentar esse desafio, propomos um sistema de resposta automatizada que utiliza de técnicas de processamento de linguagem natural como *textit*bag of words (Qader et al., 2019), *Word2Vec* (Mikolov et al., 2013) para identificar e responder as solicitações de abertura de processos com mensagens padronizadas, informando os documentos necessários para a abertura de processos.

A princípio, este projeto empregará a técnica de *bag of words*, para identificar o contexto e o propósito de cada e-mail recebido. Por ser uma técnica simples na área de processamento de linguagem natural, sua utilização permite compreender melhor os desafios de desenvolvimento e implementação presentes no sistema proposto. Em seguida, utiliza-se as técnicas de *Word2Vec* e redes neurais para a identificação do pedido realizado pelo discente.

O sistema é implementado na linguagem Python, é capaz de processar e-mails, extrair informações relevantes e responder de acordo com as necessidades específicas do remetente.

A eficácia dessa solução reside na capacidade de personalização e adaptação às necessidades individuais ou organizacionais. Ao pré-escrever uma variedade de respostas para diferentes cenários comuns, o sistema pode fornecer uma resposta rápida e precisa, economizando tempo e esforço dos funcionários, além de agilizar a abertura de processos solicitados pelos membros do corpo discente.

2 Base de Dados

Para elaboração deste trabalho foi necessário construir a nossa própria base de dados. Para isso foi construído um script em python que utiliza o protocolo IMAP para realizar um *scrapping* de todos os e-mails recebidos entre 01/01/2023 até 01/05/2024 destinados a secretaria do Departamento de matemática Aplicada (DMA) da Universidade Federal do Rio de Janeiro (UFRJ). As informações retiradas dos e-mails foram

- Assunto do e-mail
- Corpo do e-mail
- Data de recebimento do e-mail
- Se o e-mail foi respondido pela secretaria

Essas informações foram salvas em uma planilha para serem classificadas posteriormente com a ajuda dos funcionários da secretaria,. O conjunto de dados contém um

total de 253 e-mails separados em 30 categorias. As categorias podem ser consultadas na Tabela 1 e a distribuição dos e-mails por categoria em 12.

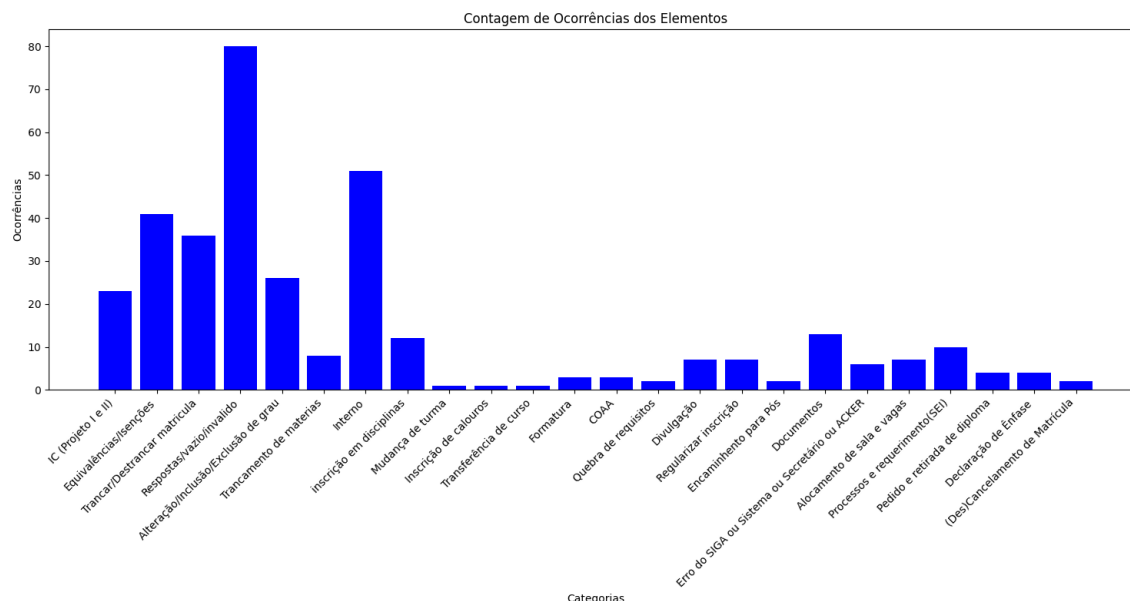


Figura 1: Distribuição das Categorias de Requerimento dos e-mails.

Por conta da natureza do conjunto de dados existem e-mails que pertencem a mais de uma categoria, por exemplo, o e-mail na Figura 2 é classificado como “Trancar/Des-trancar matrícula” e “Respostas/vazio/invalido”, o que dificulta ainda mais o problema de classificação.

Bom dia,

Envie o formulário de destrancamento de matrícula no dia 16/03 e novamente ontem 20/03 mas ainda não obtive resposta.

Poderia por favor verificar a minha situação de matrícula para que eu possa realizar a inscrição em disciplinas? Envio novamente o formulário abaixo.

Obrigada!

Figura 2: Exemplo de mensagem de e-mail pertencente a duas categorias

Além disso, o conjunto de dados é muito desbalanceado, tornando inviável o treinamento de um classificador para as 30 categorias de forma simultânea. Sendo assim, inicialmente, escolhemos as categorias **IC**, **Equivalências/Isenções**, **Trancar/Des-trancar Matrícula** e **Alteração/Inclusão de Grau**, para trabalharmos. Tendo em vista que esses são os processos mais comuns abertos pelo corpo discente. A Figura 3 contem a distribuição dos dados das categorias supracitadas.

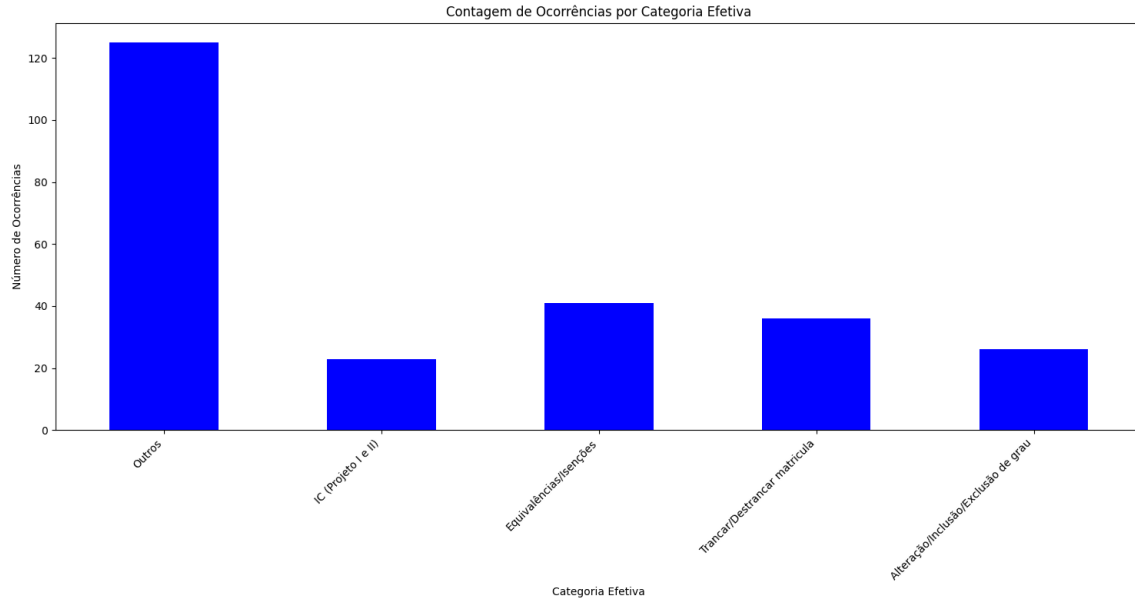


Figura 3: Distribuição das Categorias dos e-mails utilizadas para treinamento. A categoria Outros correspondem aos demais e-mails e não são utilizadas no treinamento

Para preparar o conjunto de dados para treinamento foram realizados os seguintes preprocessamentos:

- Remoção de pontuação e caracteres especiais
- Remoção de *Stopping words*
- Tokenização dos e-mails
- Construção de um dicionário de Tokens
- Transformação dos tokens em um vetor

Dos preprocessamentos realizados, a transformação dos tokens de cada e-mail é a responsável por representar os dados de uma forma que possam ser utilizados nos modelos abordados na Seção 3. Foram implementadas dois métodos de transformação dos tokens de cada e-mail em um vetor. Esses métodos consistem em:

- Construir um vetor com mesmo tamanho do dicionário de tokens onde cada entrada representa uma palavra do dicionário de tokens e o valor da entrada o número de ocorrências da palavra no corpo do e-mail. Posteriormente, o vetor é normalizado. Esse método será referenciado como **T1**.
- Construir um vetor com mesmo tamanho do dicionário de tokens onde cada entrada representa uma palavra do dicionário de tokens e o valor da entrada o inverso do número de ocorrências da palavra no corpo do e-mail. Posteriormente, o vetor é normalizado. Esse método será referenciado como **T2**.

Código	Categoria
0	Outros
1	IC (Projeto I e II)
2	Formatura
3	inscrição em disciplinas
4	Equivalências/Isenções
5	Trancar/Destrancar matrícula
6	Respostas/vazio/invalido
7	Alteração/Inclusão/Exclusão de grau
8	Trancamento de materias
9	Interno
10	Mudança de turma
11	Transferência de curso
12	Regularizar inscrição
13	Documentos
14	Pedido e retirada de diploma
15	Pedido de e-mail @matematica
16	Alocamento de sala e vagas
17	Inscrição de calouros
18	Assinatura de estágio
19	Erro do SIGA ou Sistema ou Secretário
20	Quebra de requisitos
21	Divulgação
22	Requerimento ligado as PR
23	Processos e requerimento(SEI)
24	Encaminhamento para Pós
25	Tradução Juramento
26	Declaração de Ênfase
27	(Des)Cancelamento de Matrícula
28	Jubilamento
29	COAA
30	Reingresso

Tabela 1: Categorias de Requerimentos

3 Modelos e Experimentos

Neste trabalho analisamos os seguintes modelos de classificação *Support Vector Machines* (Boser et al., 1992) *Random Forest Classifier* (Breiman, 2001), *Gradient Boost* (Friedman, 2001) e *Multy Layer Perceptron* (Bourlard et al., 1994). Por conta do grande número de categorias e a alta disparidade entre amostras, inicialmente, escolhemos as quatro categorias mais importantes, de acordo com os secretários do DMA. As categorias escolhidas foram:

- IC
- Equivalências/Isenções
- Trancar/Destrancar matrícula
- Alteração/Inclusão/Exclusão de grau

Para garantir uma comparação justa entre cada modelo utilizamos o mesmo conjunto de treino e teste para cada modelo. Isso foi feito utilizando a biblioteca **Sciki-Learn** juntamente com sua função **Train Test Split** para separar o conjunto de dados em validação, com 30% dos dados, e treino.

Ademais, cada modelo foi treinado uma vez nos dados obtidos após **T1** e **T2**. Mas por conta da dimensionalidade elevada do conjunto de dados, R^{789} , decidimos aplicar PCA (Kurita, 2019) nos dados para reduzir a dimensionalidade para R^{50} .

4 Resultados

Um dos resultados obtidos nesse trabalho foi a elaboração de um script para realização da leitura e armazenamento dos e-mails automatizando a coleta de dados e permitindo a construção da nossa base de dados de e-mails, um script de pré-processamento dos dados e um script para o envio automatizado de e-mails. Os códigos utilizados neste trabalho pode ser acessados em [GitHub](#)

Dentre os modelos treinados os que performaram melhor foram os modelos treinados em **T1**. Suas performances foram similares entre si, e, além disso, apresentaram erros de classificação razoáveis. Isto é, os erros cometidos pelos modelos são prováveis de serem cometidos por pessoas. Quanto aos modelos treinados em **T2** vemos o mesmo comportamento dos modelos em **T1**, mas com uma performance geral inferior. As matrizes de confusão de cada modelo podem ser vistas no Apêndices [A](#) e [B](#).

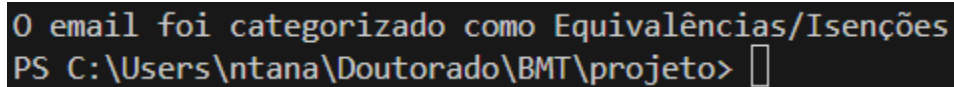
4.1 Primeira mensagem automatizada:

Com os classificadores treinados, é possível fazer a implementação das respostas automatizadas. Para isso é acessado os e-mails não lidos, sua categoria é verificada pelo nosso classificador e a resposta pre-definida é enviada. A Figura 4 é um exemplo de um e-mail, fora da base de dados e a classificação dada pelo modelo SVM treinado em **T1**.

Fala Bob. Tudo bem?

Tava querendo dar entrada com meu processo de equivalência de disciplina
Fiz a matéria XXXXXXX e queria pegar equivalencia em YYYYYY.
Meu DRE é *****

Att: ciclando deca



```
O email foi categorizado como Equivalências/Isenções
PS C:\Users\ntana\Doutorado\BMT\projeto> 
```

Figura 4: Exemplo de mensagem de e-mail fora da base de dados e sua classificação pelo modelo SVM treinado em **T1**

5 Conclusão e Trabalhos Futuros

A partir da análise do problema de classificação de e-mails da secretaria do IM, do processo de construir o conjunto de dados, e da natureza multi-categórica dos e-mails (30 classes e e-mails pertencentes a mais de uma classe) podemos concluir que este é um desafio complexo.

O problema de classificação de e-mails para a secretaria do IM é desafiador pois existe muita interseção entre as palavras presente em e-mails distintos e muitas categorias distintas de e-mails. Sendo necessário um cuidado no preprocessamento da base de dados e na escolha da representação dos e-mails em vetores.

Neste trabalho exploramos dois tipos de representações do corpo do e-mail em vetores, **T1** e **T2**, e treinamos, ao todo, 6 modelos. Sendo 3 modelos treinados em **T1** e 3 em **T2**. Dentre os modelos treinados observa-se uma melhor performance nos treinados em **T1**

Em resumo, este estudo contribui para o entendimento do problema de classificação de e-mails, das aplicações das técnicas de *bag of words* e , *Word2vec* no contexto de classificação de e-mail, e compreensão do processo de construção de um conjunto de dados.

Posteriormente, os autores tem interesse em melhorar o sistema de classificação para poder ser utilizado na secretaria do IM. Mas para isso será necessário expandir a base de dados coletando e-mails de solicitação de abertura de processos de outros departamentos do IM, e estudar a implementação de modelos de *Embeddings* e LLM.

A Matriz de confusão para treinamento em T1

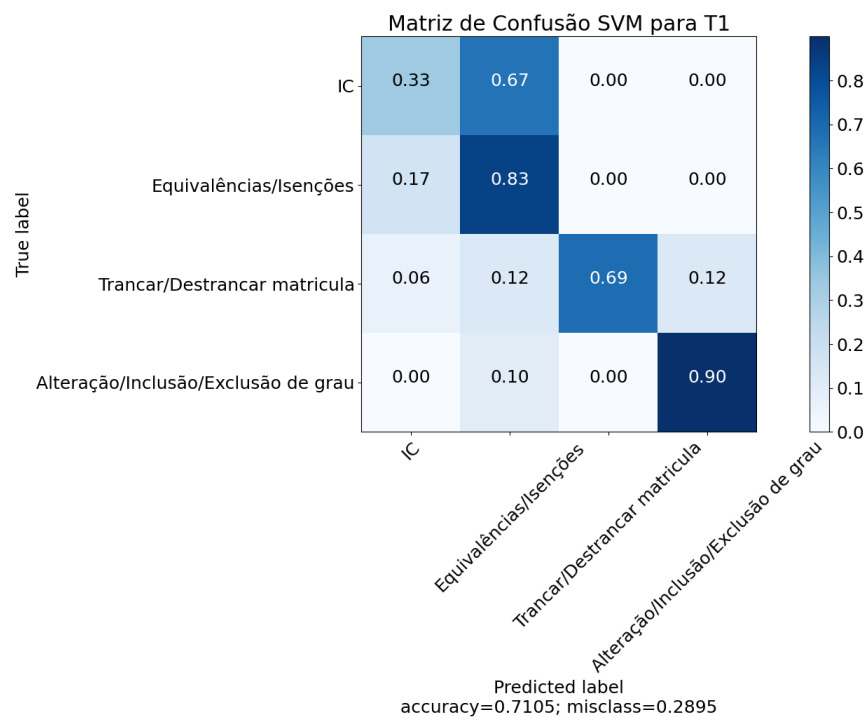


Figura 5: Matriz de confusão para SVM treinada em $T1$.

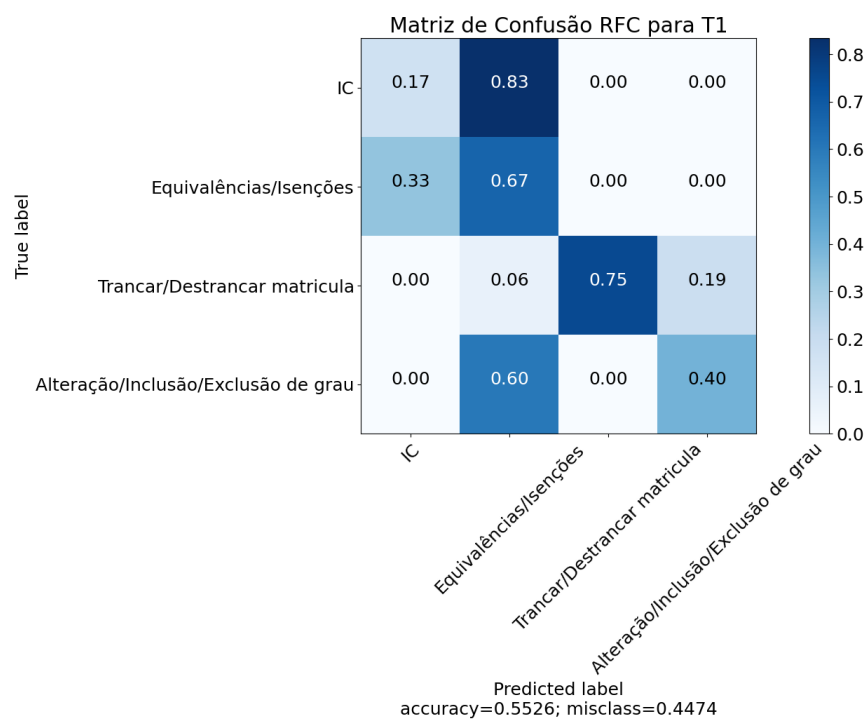


Figura 6: Matriz de confusão para RFC treinada em $T1$.

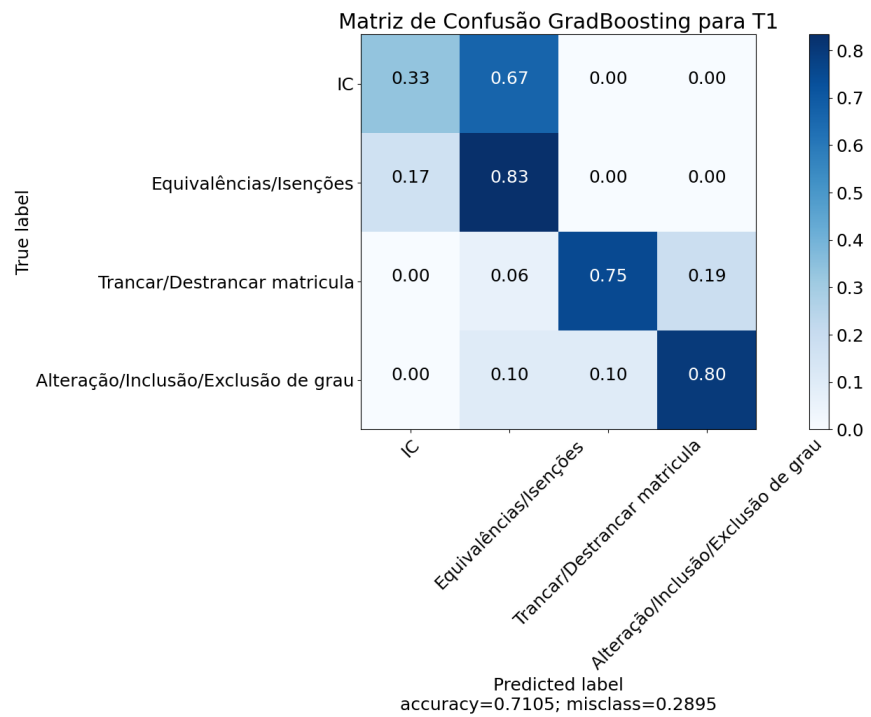


Figura 7: Matriz de confusão para Gradient Boost treinada em $T1$.

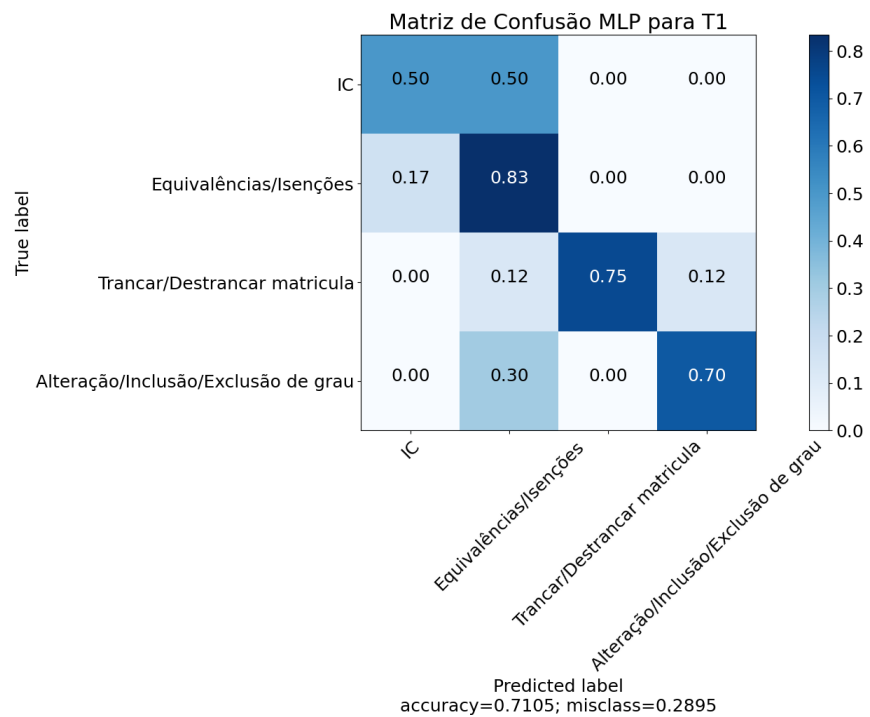


Figura 8: Matriz de confusão para MLP treinada em $T1$.

B Matriz de Covariância para treinamento em T2

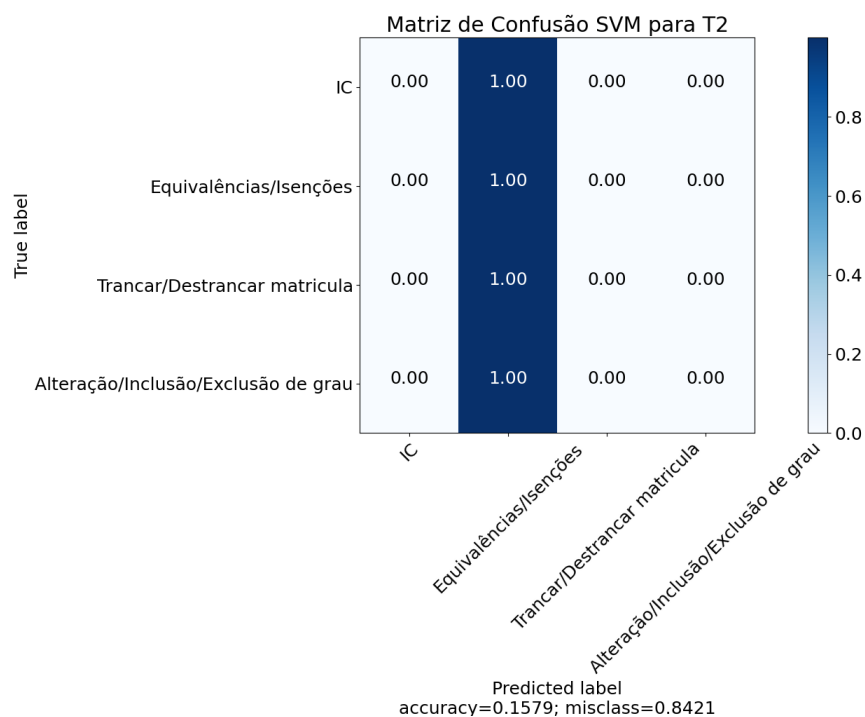


Figura 9: Matriz de confusão para SVM treinada em T_2 .

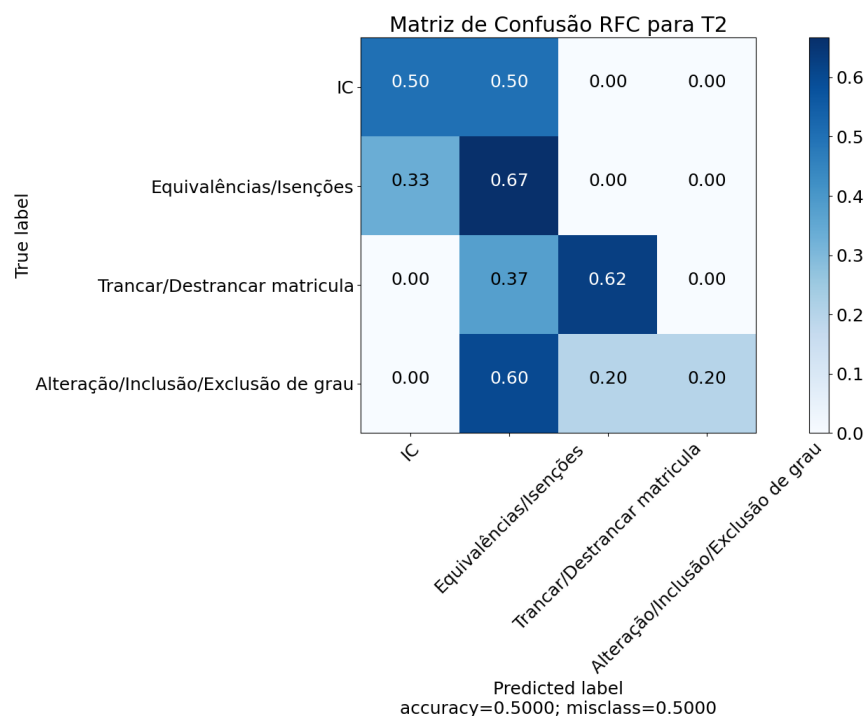


Figura 10: Matriz de confusão para RFC treinada em T_2 .

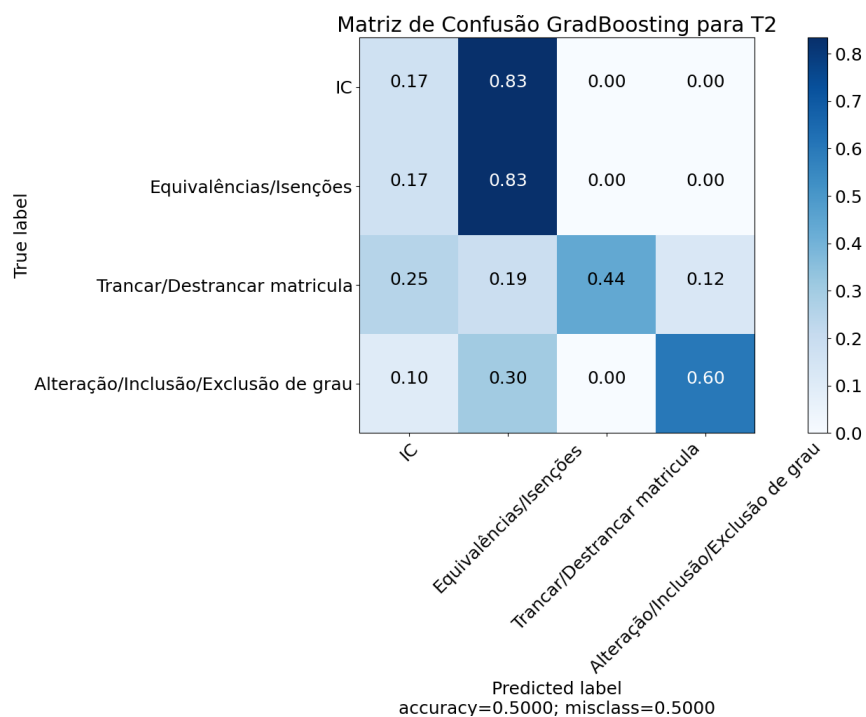


Figura 11: Matriz de confusão para Gradient Boost treinada em $T2$.

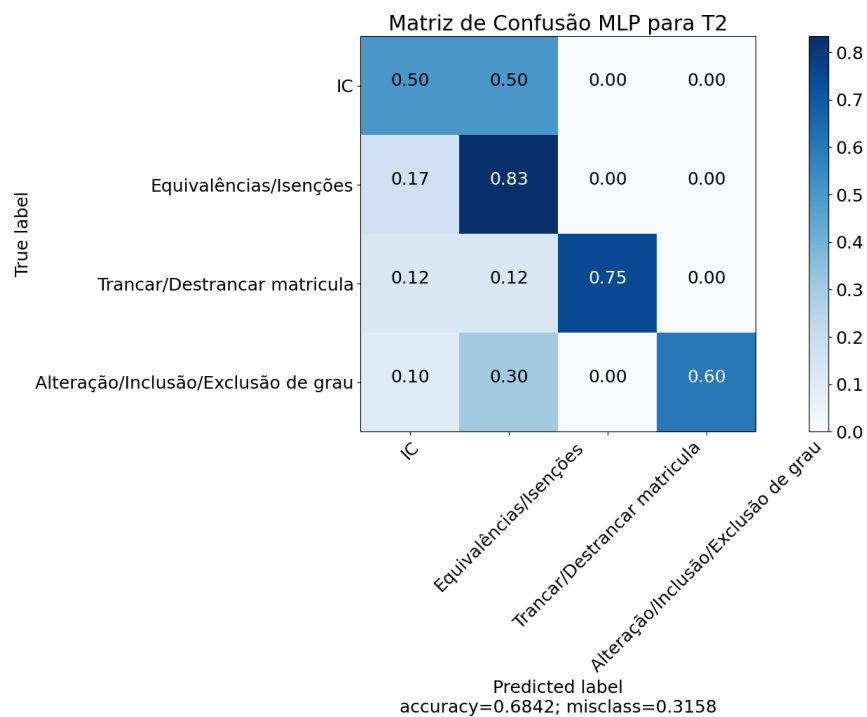


Figura 12: Matriz de confusão para MLP treinada em $T2$.

Referências

- Boser, Bernhard E, Isabelle M Guyon e Vladimir N Vapnik (1992). “A training algorithm for optimal margin classifiers”. Em: *Proceedings of the fifth annual workshop on Computational learning theory*, pp. 144–152 (ver p. 8).
- Bourlard, Hervé A, Nelson Morgan, Hervé A Bourlard e Nelson Morgan (1994). “Multilayer Perceptrons”. Em: *Connectionist Speech Recognition: A Hybrid Approach*, pp. 59–80 (ver p. 8).
- Breiman, Leo (2001). “Random forests”. Em: *Machine learning* 45, pp. 5–32 (ver p. 8).
- Friedman, Jerome H (2001). “Greedy function approximation: a gradient boosting machine”. Em: *Annals of statistics*, pp. 1189–1232 (ver p. 8).
- Kurita, Takio (2019). “Principal component analysis (PCA)”. Em: *Computer Vision: A Reference Guide*, pp. 1–4 (ver p. 8).
- Mikolov, Tomas, Kai Chen, Greg Corrado e Jeffrey Dean (2013). *Efficient Estimation of Word Representations in Vector Space*. arXiv: [1301.3781](#) [`id='cs.CL'`, `full_name='ComputationandLanguage'`, `is_active=True`, `alt_name='cmp-lg'`, `in_archive='cs'`, `is_general=False`, `description='Covers natural language processing. Roughly includes material in language issues broadly construed (natural-language processing, computational linguistics, speech, etc.)`] (ver p. 4).
- Qader, Wisam A., Musa M. Ameen e Bilal I. Ahmed (2019). “An Overview of Bag of Words; Importance, Implementation, Applications, and Challenges”. Em: *2019 International Engineering Conference (IEC)*, pp. 200–204. DOI: [10.1109/IEC47844.2019.8950616](#) (ver p. 4).