Ministério da Educação
Universidade Tecnológica Federal do Paraná
Câmpus Santa Helena
Bacharelado em Ciência da Computação

TRABALHO DE SISTEMA INTELIGENTES APLICADOS

NAIVE BAYES

Teste do algoritmo de classificação com software Weka

Hugo José Teixeira de Freitas[1]
Marcos Alexandre dos Anjos[2]

SANTA HELENA/PR
07/2021

[1] hugofreitas@alunos.utfpr.edu.br
[2] marcosanjos@alunos.utfpr.edu.br

SUMÁRIO

1. INTRODUÇÃO

Estar conectado a maior parte do tempo a internet permite gerar um grande volume de dados diariamente pelos usuários. As redes sociais são exemplos de ferramentas que utilizamos no dia a dia que permitem a criação de dados o tempo todo através de postagens e transferências.

Esse grande volume de informações aglomerado na maioria das vezes não representa nenhum sentido. Mas desde que os dados são armazenados em um banco de dados (BD) é possível extrair informações a partir de consultas. Essas consultas são técnicas simples de consulta e são conhecidas como informações triviais. Por outro lado, podemos extrair informações não triviais que são extraídas via técnicas de mineração.

Algoritmos de mineração de dados podem ser aplicados em várias áreas do conhecimento. Por exemplo, numa empresa, podemos utilizar para aumentar o número de vendas de um determinado produto. Neste contexto, esse relatório tem como finalidade apresentar princípios do algoritmo naive bayes aplicado numa base de dados de classificação de filmes do IMDB, Amazon e Yelp, utilizando para isso o software Weka.

2. ALGORITMO NAÏVE BAYES

O Classificador probabilístico mais simples é naïve bayes, onde que são estimadas pela contagem da frequência de cada característica para as instâncias dos dados de treino. Assim uma nova instância dos dados pode-se estimar a probabilidade de essa instância pertencer a uma classe específica, baseada no produto das probabilidades condicionais individuais para os valores característicos da instância (GOMES, 2009)

3. MATERIAIS MÉTODOS

A base de dados é composta por um arquivo que contém dados da IMDB, Amazon e Yelp, sendo encontrado em dataset ics. O número total de registros reunindo as três bases é de 3000 registros. A construção do classificador completou em duas etapas (a) modelagem da base de dados e (b) criação do modelo de classificador.

A modelagem da base de dados consiste primeiramente na preparação da base de dados que será realizada através da limpeza dos dados. Na sequência para criação do modelo de classificação da base de dados para ser realizado o upload no software Weka. Dessa forma, o classificador com uso do algoritmo naïve bayes, se trata de um classificador que probabilístico que emprega o conceito do Teorema de Bayes para gerar uma estimativa de novo objeto pertencer a mesma (GONÇALVES, 2014).

4. RESULTADOS E DISCUSSÃO

A base de dados fornecida estava em arquivos separados em formato TXT, onde foram realizados alguns ajustes para carregar a base de dados no software Weka. Como o software Weka trabalha com modelo de estruturação proprietário os passos para deixar no padrão arff, foram (a) agrupamento das três bases de dados fornecidas em um único arquivo. (b) Como os dados eram no formato de uma *string* seguida vírgula com zero ou um. Onde a string estava com aspas duplas, sendo método prático utilizado a função replace trocando as duas aspas duplas por vazio, sendo utilizando o editor de texto vscode. Segundo passo, removi todas as vírgulas do texto com o mesmo procedimento, mas para manter o padrão csv onde gostaria de ter uma coluna zero ou um. Para atender esse padrão a estratégia foi a mesma utilizada mas trocando 0 por ,0 assim sucessivamente. (c) Terceiro passo consistiu numa limpeza de dados onde foi implementado um algoritmo onde utilizamos bibliotecas para ajudar remover palavras indesejadas e assim deixar a frases somente com as palavras que faz sentido. (d) No terceiro passo temos que padronizar os metadados do nosso arquivo de configuração:

- *% Nome_base_dados*
- *@relation Nome_relação*
- *@attribute Descrição_colunas + tipo_coluna*
- *@data*

Essas são as principais características para preparar o arquivo da base de dados no padrão arff, como mostra figura 1.

```
% Lista de filmes
@relation filmes

@attribute descricao string
@attribute avaliacao { 0, 1 }

@data
"So there is no way for me to plug it in here in the US unless I go by a converter. ", 0
"Good case Excellent value. ", 1
"Great for the jawbone. ", 1
```

Figura 1: Arquivo no padrão arff.

Fonte: Autor , 2021.

A etapa de junção dos arquivos foi utilizada shell script para tal função e algumas remoções.Etapa de limpeza foi utilizado python para a importação do arquivo de texto, limpeza de *stop_words*, caracteres e cadeias não desejadas, como mostra figura 2.

```
separator = ' '
useless = ["n't", ".", "!", "(", ")",
           "$", "[", "]", "'", "*",
           "&", "%", "#"]

stop_words = set(stopwords.words('english'))
data = pd.read_csv("filmes.txt", sep="\t", header=None)

col = []

for sentence in data[0]:
   splited = word_tokenize(sentence)
   sentence = [s for s in sentence if not s in useless]
   sentence = [splt for splt in splited if not splt.lower() in stop_words]
   col.append(separator.join(sentence))

data[0] = list(map(lambda x: x.lower(), col))
data.to_csv("base-final.csv", header=None, index=False)
```

Figura 2: Código para limpeza de dados .

Fonte: Autor , 2021.

Com o arquivo configurado no padrão foi carregado no software Weka, como mostra a figura 2. O carregamento do arquivo na aba *Process -> Open file -> arquivo*. Podemos observar que nossa base de dados contém 3000 registros e duas colunas.

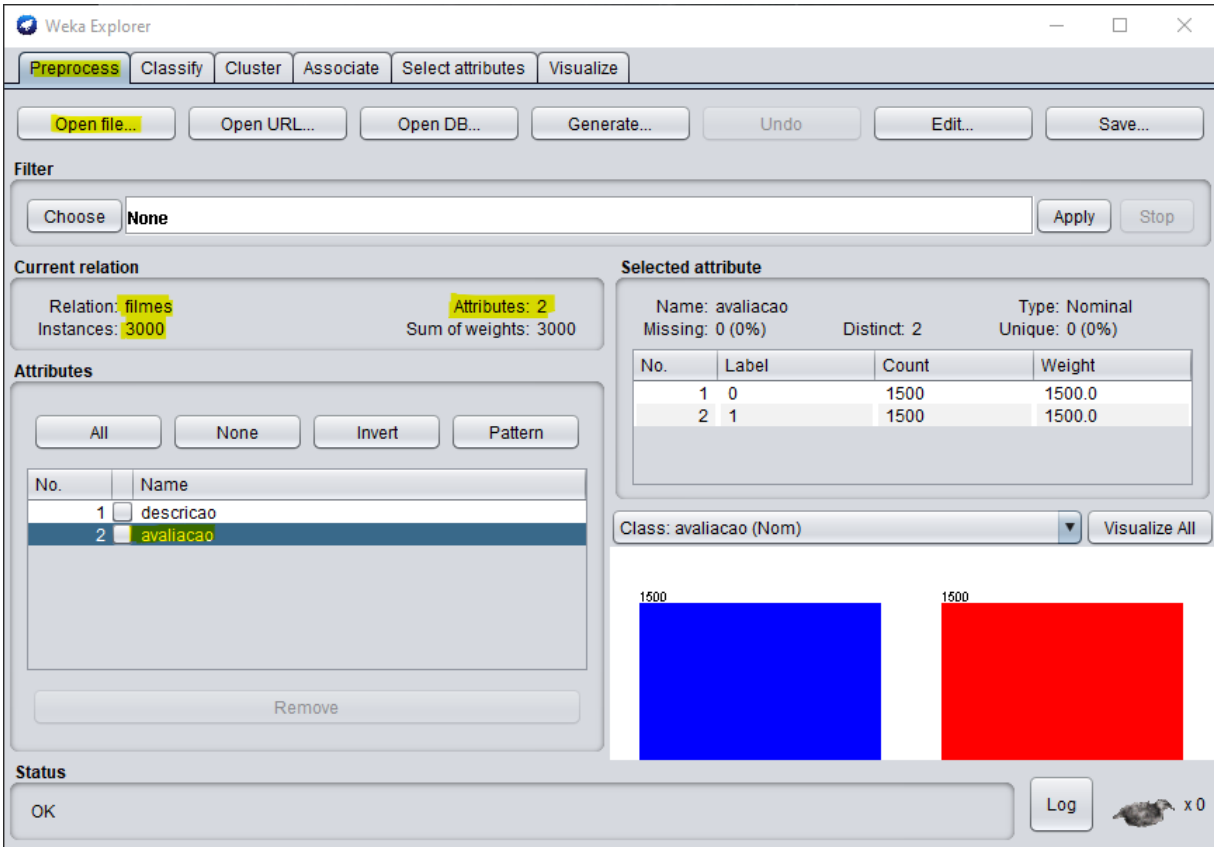

Figura 1: Base de dados carregada no Weka.

Fonte: Autor , 2021.

Com nossa base de dados carregada no Weka, foi aplicado o filtro *String Word Vector* para quebrar todas as strings da base de dados em palavras separadas. Na figura 2 podemos observar que foram quebradas 2091 palavras, números, símbolos. Vale ressaltar que estamos testando duas versões (a) base de dados sem limpeza e (b) aplicando técnicas de limpeza de frases com texto removendo aqueles caracteres que não fazem sentido.

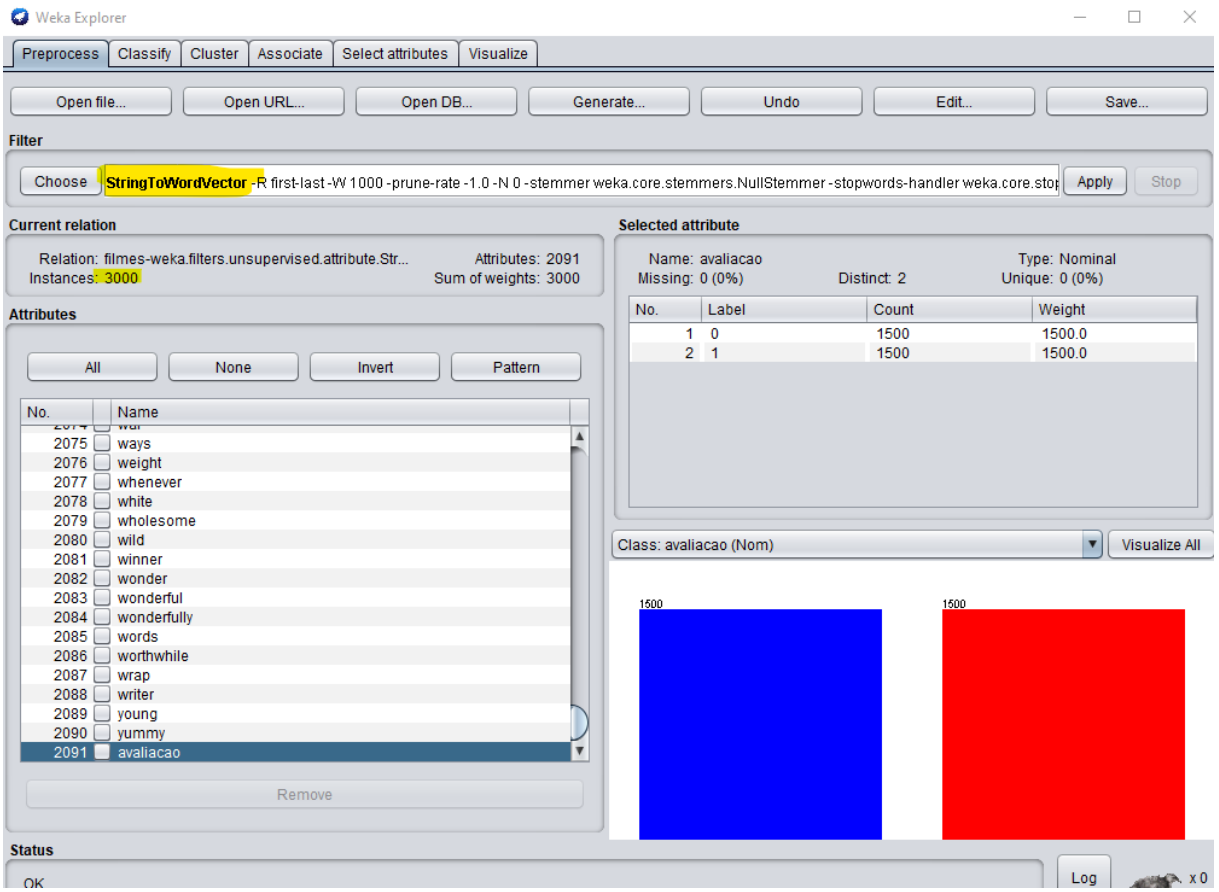

Figura 2: Aplicação do filtro *String Word Vector* na base de dados.

Fonte: Autor , 2021.

Agora próximo passo a nossa base de dados está no padrão, para realização do teste do algoritmo naive bayes. Os testes foram divididos em três partes de acordo com o documento original da atividade.

1) Resultados de performance e da classificação do algoritmo em relação a base de dados, cole a imagem do resultado com foco nos resultados de acurácia, verdadeiro positivo e falso positivo e precisão obtidos. Gere gráficos com os valores das métricas e discorra sobre elas. Pesquisem o significado e uso de cada uma das métricas para utilizar seus valores na descrição dos resultados obtidos.

Teste na base de dados sem filtro de limpeza

Algoritmo Naive Bayes

Split : 66%

```
Correctly Classified Instances         682               66.8627 %
Incorrectly Classified Instances       338               33.1373 %
Kappa statistic                          0.3381
Mean absolute error                      0.3764
Root mean squared error                  0.483
Relative absolute error                 75.2634 %
Root relative squared error             96.5802 %
Total Number of Instances             1020
```

```
=== Detailed Accuracy By Class ===

                 TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 0.637     0.299     0.690      0.637     0.663      0.339    0.723     0.706     0
                 0.701     0.363     0.649      0.701     0.674      0.339    0.723     0.726     1
Weighted Avg.    0.669     0.330     0.670      0.669     0.668      0.339    0.723     0.716
```

```
=== Confusion Matrix ===

   a   b   <-- classified as
 332 189 |   a = 0
 149 350 |   b = 1
```

Tempo (log)

10:37:56: Started weka.classifiers.bayes.NaiveBayes
10:37:56: Command: weka.classifiers.bayes.NaiveBayes
10:38:00: Finished weka.classifiers.bayes.NaiveBayes
10:38:01: Warning : data contains more attributes than can be displayed as attribute bars.

Teste na base de dados sem filtro de limpeza

Algoritmo Naive Bayes

Split : 93%

```
Correctly Classified Instances         147               70      %
Incorrectly Classified Instances        63               30      %
Kappa statistic                          0.4029
Mean absolute error                      0.3587
Root mean squared error                  0.4681
Relative absolute error                 71.7085 %
Root relative squared error             93.5672 %
Total Number of Instances              210
```

Acurácia

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.658	0.250	0.758	0.658	0.704	0.407	0.753	0.748	0
0.750	0.342	0.649	0.750	0.696	0.407	0.753	0.733	1
0.700	0.292	0.708	0.700	0.700	0.407	0.753	0.741	

```
=== Confusion Matrix ===

  a  b   <-- classified as
 75 39 |  a = 0
 24 72 |  b = 1
```

Tempo (log)

10:39:06: Started weka.classifiers.bayes.NaiveBayes
10:39:06: Command: weka.classifiers.bayes.NaiveBayes
10:39:13: Finished weka.classifiers.bayes.NaiveBayes
10:39:14: Warning : data contains more attributes than can be displayed as attribute bars.

Teste na base de dados com filtro de limpeza

Algoritmo Naive Bayes

Split : 66%

O Split significa que é separada 66% da base de dados para o treinamento e o restante será utilizado para o teste do modelo.

```
Correctly Classified Instances         729               71.4706 %
Incorrectly Classified Instances       291               28.5294 %
Kappa statistic                          0.426
Mean absolute error                      0.3829
Root mean squared error                  0.4413
Relative absolute error                 76.5622 %
Root relative squared error             88.232  %
Total Number of Instances             1020
```

Acurácia

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,846	0,423	0,676	0,846	0,752	0,441	0,796	0,770	0
	0,577	0,154	0,783	0,577	0,664	0,441	0,796	0,794	1
Weighted Avg.	0,715	0,291	0,728	0,715	0,709	0,441	0,796	0,782	

Matriz de Confusão:

```
=== Confusion Matrix ===

   a   b   <-- classified as
 441  80 |   a = 0
 211 288 |   b = 1
```

Tempo de execução: ~2 segundos.

```
10:37:46: Started weka.classifiers.bayes.NaiveBayes
10:37:46: Command: weka.classifiers.bayes.NaiveBayes
10:37:48: Finished weka.classifiers.bayes.NaiveBayes
10:37:48: Warning : data contains more attributes than can be displayed as attribute bars.
```

Teste na base de dados com filtro de limpeza

Algoritmo Naive Bayes

Split : 70%

O Split significa que é separada 70% da base de dados para o treinamento e os outros 30% serão utilizados para o teste do modelo.

```
Correctly Classified Instances         640               71.1111 %
Incorrectly Classified Instances       260               28.8889 %
Kappa statistic                          0.4201
Mean absolute error                      0.3843
Root mean squared error                  0.4419
Relative absolute error                 76.8475 %
Root relative squared error             88.3726 %
Total Number of Instances              900
```

Acurácia:

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,846	0,428	0,670	0,846	0,748	0,436	0,799	0,776	0
	0,572	0,154	0,784	0,572	0,661	0,436	0,799	0,797	1
Weighted Avg.	0,711	0,293	0,726	0,711	0,705	0,436	0,799	0,786	

Matriz de Confusão:

```
=== Confusion Matrix ===

   a   b   <-- classified as
 386  70 |   a = 0
 190 254 |   b = 1
```

Tempo de execução: ~2 segundos.

```
10:43:49: Started weka.classifiers.bayes.NaiveBayes
10:43:49: Command: weka.classifiers.bayes.NaiveBayes
10:43:51: Finished weka.classifiers.bayes.NaiveBayes
10:43:51: Warning : data contains more attributes than can be displayed as attribute bars.
```

A segunda atividade consiste em (2) utilizar três diferentes valores na estratégia de testes cross-validation, discorra sobre os resultados obtidos de acordo com: o Instancias classificadas de forma correta e incorreta o Taxa de Verdadeiro Positivo e Falso Positivo o Precisão o Matriz de Confusão.

Teste na base de dados sem filtro de limpeza

Algoritmo Naive Bayes

Cross-validation: 10

```
Correctly Classified Instances        2091               69.7    %
Incorrectly Classified Instances       909               30.3    %
Kappa statistic                          0.394
Mean absolute error                      0.3617
Root mean squared error                  0.4593
Relative absolute error                 72.3456 %
Root relative squared error             91.8691 %
Total Number of Instances             3000
```

Acurácia

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.709	0.315	0.693	0.709	0.700	0.394	0.756	0.722	0
0.685	0.291	0.702	0.685	0.693	0.394	0.756	0.769	1
0.697	0.303	0.697	0.697	0.697	0.394	0.756	0.745	

```
=== Confusion Matrix ===

    a    b   <-- classified as
 1063  437 |    a = 0
  472 1028 |    b = 1
```

Tempo (log)

10:41:24: Started weka.classifiers.bayes.NaiveBayes
10:41:24: Command: weka.classifiers.bayes.NaiveBayes
10:41:46: Finished weka.classifiers.bayes.NaiveBayes
10:41:47: Warning : data contains more attributes than can be displayed as attribute bars.

Teste na base de dados sem filtro de limpeza

Algoritmo Naive Bayes

Cross-validation: 25

```
Correctly Classified Instances        2101               70.0333 %
Incorrectly Classified Instances       899               29.9667 %
Kappa statistic                          0.4007
Mean absolute error                      0.3608
Root mean squared error                  0.4581
Relative absolute error                 72.1691 %
Root relative squared error             91.6229 %
Total Number of Instances             3000
```

Acurácia

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.709	0.308	0.697	0.709	0.703	0.401	0.757	0.724	0
0.692	0.291	0.704	0.692	0.698	0.401	0.757	0.770	1
0.700	0.300	0.700	0.700	0.700	0.401	0.757	0.747	

```
=== Confusion Matrix ===

    a    b   <-- classified as
 1063  437 |    a = 0
  462 1038 |    b = 1
```

Tempo (log)

10:47:41: Started weka.classifiers.bayes.NaiveBayes
10:47:41: Command: weka.classifiers.bayes.NaiveBayes
10:48:14: Finished weka.classifiers.bayes.NaiveBayes
10:48:15: Warning : data contains more attributes than can be displayed as attribute bars.

Teste na base de dados sem filtro de limpeza

Algoritmo Naive Bayes

Cross-validation: 5

```
Correctly Classified Instances        2086               69.5333 %
Incorrectly Classified Instances       914               30.4667 %
Kappa statistic                          0.3907
Mean absolute error                      0.3625
Root mean squared error                  0.461
Relative absolute error                 72.505  %
Root relative squared error             92.2048 %
Total Number of Instances             3000
```

Acurácia

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.711	0.320	0.690	0.711	0.700	0.391	0.754	0.720	0
0.680	0.289	0.702	0.680	0.691	0.391	0.754	0.766	1
0.695	0.305	0.696	0.695	0.695	0.391	0.754	0.743	

```
=== Confusion Matrix ===

    a    b   <-- classified as
 1066  434 |    a = 0
  480 1020 |    b = 1
```

Tempo (log)

12:02:29: Started weka.classifiers.bayes.NaiveBayes
12:02:29: Command: weka.classifiers.bayes.NaiveBayes
12:02:40: Finished weka.classifiers.bayes.NaiveBayes
12:02:41: Warning : data contains more attributes than can be displayed as attribute bars.

Teste na base de dados com filtro de limpeza

Algoritmo Naive Bayes

Cross-validation: 5

Correctly Classified Instances	2071	69.0333 %
Incorrectly Classified Instances	929	30.9667 %
Kappa statistic	0.3807	
Mean absolute error	0.3864	
Root mean squared error	0.4489	
Relative absolute error	77.2772 %	
Root relative squared error	89.7704 %	
Total Number of Instances	3000	

Acurácia:

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,827	0,447	0,649	0,827	0,728	0,396	0,774	0,745	0
	0,553	0,173	0,762	0,553	0,641	0,396	0,774	0,780	1
Weighted Avg.	0,690	0,310	0,706	0,690	0,684	0,396	0,774	0,763	

Matriz de Confusão

a	b	<-- classified as
1241	259	a = 0
670	830	b = 1

Tempo de execução: ~7 segundos.

```
11:07:28: Started weka.classifiers.bayes.NaiveBayes
11:07:28: Command: weka.classifiers.bayes.NaiveBayes
11:07:35: Finished weka.classifiers.bayes.NaiveBayes
11:07:35: Warning : data contains more attributes than can be displayed as attribute bars.
```

Teste na base de dados com filtro de limpeza

Algoritmo Naive Bayes

Cross-validation:10

```
Correctly Classified Instances        2071               69.0333 %
Incorrectly Classified Instances       929               30.9667 %
Kappa statistic                          0.3807
Mean absolute error                      0.3855
Root mean squared error                  0.4467
Relative absolute error                 77.1066 %
Root relative squared error             89.35   %
Total Number of Instances             3000
```

Acurácia:

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,836	0,455	0,647	0,836	0,730	0,398	0,779	0,750	0
	0,545	0,164	0,769	0,545	0,638	0,398	0,779	0,785	1
Weighted Avg.	0,690	0,310	0,708	0,690	0,684	0,398	0,779	0,768	

Matriz de Confusão:

```
=== Confusion Matrix ===

   a    b   <-- classified as
1254  246 |    a = 0
 683  817 |    b = 1
```

Tempo de execução: ~11 segundos.

```
11:01:26: Started weka.classifiers.bayes.NaiveBayes
11:01:26: Command: weka.classifiers.bayes.NaiveBayes
11:01:37: Finished weka.classifiers.bayes.NaiveBayes
11:01:37: Warning : data contains more attributes than can be displayed as attribute bars.
```

Teste na base de dados com filtro de limpeza

Algoritmo Naive Bayes

Cross-validation: 25

```
Correctly Classified Instances        2069               68.9667 %
Incorrectly Classified Instances       931               31.0333 %
Kappa statistic                          0.3793
Mean absolute error                      0.3853
Root mean squared error                  0.446
Relative absolute error                 77.0572 %
Root relative squared error             89.1999 %
Total Number of Instances             3000
```

Acurácia:

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,835	0,456	0,647	0,835	0,729	0,397	0,780	0,752	0
	0,544	0,165	0,768	0,544	0,637	0,397	0,780	0,785	1
Weighted Avg.	0,690	0,310	0,707	0,690	0,683	0,397	0,780	0,769	

Matriz de Confusão:

```
   a    b   <-- classified as
1253  247 |    a = 0
 684  816 |    b = 1
```

Tempo de execução: ~26 segundos

```
11:50:11: Started weka.classifiers.bayes.NaiveBayes
11:50:11: Command: weka.classifiers.bayes.NaiveBayes
11:50:36: Finished weka.classifiers.bayes.NaiveBayes
11:50:37: Warning : data contains more attributes than can be displayed as attribute bars.
```

Conclusão sobre a primeira parte da atividade, na tabela 1 temos os principais termos que utilizamos para realização dessa atividade.

Termo	**Definição**
Split	Quantidade separada para treino e teste do modelo preditivo.
Cross-validation	Consiste no particionamento dos dados em conjuntos(partes), onde um conjunto é realizado para treino e outro conjunto é utilizado para teste e avaliação do desempenho do modelo.
Acurácia	Representa a porcentagem de observações do conjunto de teste que são corretamente classificadas por ele. Caso a acurácia seja alta, o modelo de classificação é considerado eficiente e pode ser utilizado para classificar novos casos.
Matriz de confusão	Mostra as frequências de classificação para cada classe do modelo: - Verdadeiro positivo: ocorre quando o conjunto real, a classe que estamos buscando foi prevista corretamente. - Falso positivo: quando no conjunto real, a classe que estamos buscando prevê que foi incorretamente. - Falso verdadeiro: quando no conjunto real, a classe que não estamos buscando prevê que foi corretamente. - Falso negativo: quando o conjunto real, a classe que não estamos buscando prevê que foi incorretamente.

Tabela 1: Definição de alguns termos.

Os testes foram realizados comparando a mesma base de dados, com a diferença na aplicação do filtro de limpeza. Lembrando que base sem filtro foi realizada na máquina A e com aplicação do filtro na máquina B como mostra tabela 2. Resumo dos testes podemos observar na tabela 3.

Especificações	**Máquina A**	**Máquina B**
Modelo da Máquina	Lenovo IdeaPad 330-IKB 15	Asus Rog GL552VW
Sistema Operacional	Windows 10	Ubuntu 20.04 LTS
Processador	i3-8130 U	i7-6700HQ
Disco rígido	240GB (SSD)	500GB (HD)
Memória Ram	12GB	16GB

Tabela 2: Configuração das máquinas para teste dos algoritmos.

Fonte: Autores, 2021.

Algoritimo Naive Bayes									
Máquina A					**Máquina B**				
Split		**Cross-validation**			**Split**		**Cross-validation**		
66%	**93%**	**5**	**10**	**25**	**66%**	**70%**	**5**	**10**	**25**
Correto 66.86% Incorreto 33,13%	Correto 70% Incorreto 30%	Correto 69.53% Incorreto 30.46%	Correto 69.7% Incorreto 30.3%	Correto 70.03% Incorreto 29.96%	Correto 71.47% Incorreto 28.53%	Correto 71.11% Incorreto 28.89%	Correto 69.03% Incorreto 30.97%	Correto 69.03% Incorreto 30.97%	Correto 68.96% Incorreto 31.03%

Tabela 2: Resultado da comparação dos testes dos algoritmos.

Fonte: Autores, 2021.

Observando os resultados gerados na tabela 2, podemos concluir que o algoritmo Naive Bayes é um bom classificador de sentenças. Para o treinamento do algoritmo para Split utilizamos 66% e 93% para base sem limpeza e 66% e 70% para base com limpeza. Assim o algoritmo gerou algumas características que chamou a atenção foi a diferença dos resultados executados em bases diferentes (com e sem limpeza dos dados). Onde que naive bayes configurado com split 66% sem limpeza apresentou performance melhor de 4.6% em relação com a base que foi aplicado a limpeza. No caso do algoritmo Cross-validation apresentou características semelhantes não apresentando um resultado expressivo.

A Terceira atividade consiste em :

Clicando com o botão esquerdo na caixa de texto ao lado do botão 'Choose' no Weka, abra a janela de opções. Modifique a opção *'UseSupervisedDiscretization'* para 'true'. Nessa opção, ao invés de tratar cada atributo como uma variável real gaussiana, o algoritmo primeiro discretiza os atributos e depois utiliza contagens para estimar probabilidades. Execute o algoritmo e reporte a taxa de acerto obtida com essa opção e responda com sua análise:

Teste na base de dados sem filtro de limpeza

Naive Bayes

Cross-validation: 10

Discretização Supervisionada (true):

```
Correctly Classified Instances        2040               68      %
Incorrectly Classified Instances       960               32      %
Kappa statistic                          0.36
Mean absolute error                      0.3813
Root mean squared error                  0.4371
Relative absolute error                 76.258  %
Root relative squared error             87.4109 %
Total Number of Instances             3000
```

Discretização Supervisionada (false):

```
Correctly Classified Instances        2091               69.7    %
Incorrectly Classified Instances       909               30.3    %
Kappa statistic                          0.394
Mean absolute error                      0.3617
Root mean squared error                  0.4593
Relative absolute error                 72.3456 %
Root relative squared error             91.8691 %
Total Number of Instances             3000
```

Acurácia | Discretização Supervisionada (true):

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.827	0.467	0.639	0.827	0.721	0.377	0.759	0.743	0
0.533	0.173	0.755	0.533	0.625	0.377	0.759	0.778	1
0.680	0.320	0.697	0.680	0.673	0.377	0.759	0.761	

Acurácia | Discretização Supervisionada (false):

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.709	0.315	0.693	0.709	0.700	0.394	0.756	0.722	0
0.685	0.291	0.702	0.685	0.693	0.394	0.756	0.769	1
0.697	0.303	0.697	0.697	0.697	0.394	0.756	0.745	

Matriz de Confusão:

Sem a Opção:

```
=== Confusion Matrix ===

   a    b   <-- classified as
1241  259 |    a = 0
 701  799 |    b = 1
```

Tempo (log)

13:31:57: Started weka.classifiers.bayes.NaiveBayes
13:31:57: Command: weka.classifiers.bayes.NaiveBayes -D
13:32:21: Finished weka.classifiers.bayes.NaiveBayes
13:32:21: Warning : data contains more attributes than can be displayed as attribute bars.

Discretização Supervisionada:

```
=== Confusion Matrix ===

   a    b   <-- classified as
1063  437 |    a = 0
 472 1028 |    b = 1
```

Tempo (log)

10:41:24: Started weka.classifiers.bayes.NaiveBayes
10:41:24: Command: weka.classifiers.bayes.NaiveBayes
10:41:46: Finished weka.classifiers.bayes.NaiveBayes
10:41:47: Warning : data contains more attributes than can be displayed as attribute bars.

Teste na base de dados com filtro de limpeza

Naive Bayes Discretização Supervisionada

Cross-validation: 10

Sem a opção

Correctly Classified Instances	2071	69.0333 %
Incorrectly Classified Instances	929	30.9667 %
Kappa statistic	0.3807	
Mean absolute error	0.3855	
Root mean squared error	0.4467	
Relative absolute error	77.1066 %	
Root relative squared error	89.35 %	
Total Number of Instances	3000	

Discretização Supervisionada:

Correctly Classified Instances	2088	69.6 %
Incorrectly Classified Instances	912	30.4 %
Kappa statistic	0.392	
Mean absolute error	0.379	
Root mean squared error	0.4322	
Relative absolute error	75.8066 %	
Root relative squared error	86.4398 %	
Total Number of Instances	3000	

Acurácia:

Sem a opção:

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,836	0,455	0,647	0,836	0,730	0,398	0,779	0,750	0
	0,545	0,164	0,769	0,545	0,638	0,398	0,779	0,785	1
Weighted Avg.	0,690	0,310	0,708	0,690	0,684	0,398	0,779	0,768	

Discretização Supervisionada:

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,944	0,552	0,631	0,944	0,756	0,451	0,747	0,721	0
	0,448	0,056	0,889	0,448	0,596	0,451	0,747	0,773	1
Weighted Avg.	0,696	0,304	0,760	0,696	0,676	0,451	0,747	0,747	

Matriz de Confusão:

Sem a Opção:

```
   a    b   <-- classified as
1254  246 |    a = 0
 683  817 |    b = 1
```

Discretização Supervisionada:

```
   a    b   <-- classified as
1416   84 |    a = 0
 828  672 |    b = 1
```

Tempo de execução:

Sem a opção: ~11 segundos

```
11:01:26: Started weka.classifiers.bayes.NaiveBayes
11:01:26: Command: weka.classifiers.bayes.NaiveBayes
11:01:37: Finished weka.classifiers.bayes.NaiveBayes
11:01:37: Warning : data contains more attributes than can be displayed as attribute bars.
```

Discretização Supervisionada: ~15 segundos

```
13:29:10: Started weka.classifiers.bayes.NaiveBayes
13:29:10: Command: weka.classifiers.bayes.NaiveBayes -D
13:29:25: Finished weka.classifiers.bayes.NaiveBayes
13:29:25: Warning : data contains more attributes than can be displayed as attribute bars.
```

Na tabela 3, podemos observar uma comparação lado a lado com a configuração ativando o método “Discretização”, os testes foram realizados nas duas versões da base de dados conforme mostra a tabela.

Algoritimo Naive Bayes			
Máquina A		**Máquina B**	
Cross-validation		**Cross-validation**	
Discretização - Base sem limpeza		**Discretização - Base com limpeza**	
10 (False)	**10 (True)**	**10 (False)**	**10 (True)**
Correto 68% Incorreto 32%	Correto 69.7% Incorreto 30.3%	Correto 69.03% Incorreto 30.97%	Correto 69.6% Incorreto 30.4%

Tabela 3: Resultado da comparação dos testes dos algoritmos.

Fonte: Autor, 2021.

A mudança foi significativa?

O algoritmo apresentou resultados interessantes onde avaliamos o desempenho das duas bases de dados (a) sem realização de limpeza, quando ajustamos a discretização supervisionada para *True,* obteve um aumento de 1.7% em relação à mesma base, mas sem estar ativado esta *flag*. Caso semelhante ocorreu na base (b) com limpeza de dados onde apresentou aumento de quase 0.6%.

Pela matriz de confusão podemos observar também que houve um aumento da classe 0 sendo classificado como 0, porém o mesmo acontece com 1, ou seja, quem implica que existem poucas instâncias classificadas de forma errada e consequentemente mais instantâneas classificadas.

Por que isso aconteceu?

A discretização tem como objetivo transformar atributos contínuos em discretos. Se diz supervisionada pois torna discreto também a classe do modelo. A discretização faz com que dados contínuos passem a ser representados pelo grupo pertencente. Assim, grupos passam a representar um *range* de valores e gerando outros relacionamentos, com atributos e classe, dos quais não teriam por não realizar a discretização, podendo assim afetar tanto positiva quanto negativamente o modelo.
Houve pouca diferença entre as análises pré e pós a discretização devido ao fato da base de dados não possuir nenhum atributo ou classe de valores contínuos.

Resumo dos resultados e qual algoritmo possui melhor performance e qual possui melhor resultado de classificação?

Como podemos observar na tabela 3, apresentamos resultados comparando o uso da discretização. Logo podemos concluir que foi viável a utilização do método de discretização onde apresentou resultados superiores. Vale ressaltar que quando olhamos para caso onde a base de dados não foi aplicada limpeza, o uso deste método teve impacto de 1.7% superior, estes resultados são com configuração Naive Bayes com uso Cross-validation.

CONCLUSÃO

Podemos concluir que o trabalho apresentou um estudo inicial sobre mineração e classificação de strings, onde podemos compreender a partir de uma base de dados de comentários de filmes o funcionamento do algoritmo Naive Bayes. Assim verificamos que podemos aplicar filtros e técnicas de mineração de dados para melhorar o desempenho do modelo de classificação.

REFERÊNCIAS

GOMES, J. L. DE S. Paralelização de algoritmo de simulação de Monte Carlo para a adsorção em superfícies heterogêneas bidimensionais. 2009.

GONÇALVES, E. C. **NBBR: A Baseline Method for the Evaluation of Bayesian Multi-label Classification Algorithms**. 2014 14th International Conference on Computational Science and Its Applications. **Anais**... In: 2014 14TH INTERNATIONAL CONFERENCE ON COMPUTATIONAL SCIENCE AND ITS APPLICATIONS. jun. 2014.