



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS DE RUSSAS
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

MATEUS SANTOS OLIVEIRA

**SELEÇÃO E AGRUPAMENTO DE CARACTERÍSTICAS PARA CLASSIFICAÇÃO
DE AGENTES AUTOMATIZADOS EM REDES SOCIAIS**

RUSSAS

2019

MATEUS SANTOS OLIVEIRA

SELEÇÃO E AGRUPAMENTO DE CARACTERÍSTICAS PARA CLASSIFICAÇÃO DE
AGENTES AUTOMATIZADOS EM REDES SOCIAIS

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Ciência da Computação
do Campus de Russas da Universidade Federal
do Ceará, como requisito parcial à obtenção do
grau de bacharel em Ciência da Computação.

Orientador: Prof. Ms. Alex Lima Silva

RUSSAS

2019

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

- O48s Oliveira, Mateus Santos.
Seleção e agrupamento de características para classificação de agentes automatizados em redes sociais /
Mateus Santos Oliveira. – 2019.
39 f. : il.
- Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Russas,
Curso de Ciência da Computação, Russas, 2019.
Orientação: Prof. Me. Alex Lima Silva.
1. Aprendizado de Máquina. 2. Rede Sociais. 3. Bots. I. Título.

CDD 005

MATEUS SANTOS OLIVEIRA

SELEÇÃO E AGRUPAMENTO DE CARACTERÍSTICAS PARA CLASSIFICAÇÃO DE
AGENTES AUTOMATIZADOS EM REDES SOCIAIS

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Ciência da Computação
do Campus de Russas da Universidade Federal
do Ceará, como requisito parcial à obtenção do
grau de bacharel em Ciência da Computação.

Aprovada em:

BANCA EXAMINADORA

Prof. Ms. Alex Lima Silva (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Francisco Nauber Bernardo Gois
Universidade Federal do Ceará (UFC)

Profa. Ms. Tatiane Fernandes Figueiredo
Universidade Federal do Ceará (UFC)

LISTA DE TABELAS

Tabela 1 – Matriz de confusão	26
Tabela 2 – Grupos	31
Tabela 3 – Grupos Balanceados	32
Tabela 4 – Resultados dos testes com seleção de características	34
Tabela 5 – Resultados dos testes com todas características	34
Tabela 6 – Tabela de comparação com trabalhos anteriores	35

LISTA DE ABREVIATURAS E SIGLAS

<i>API</i>	<i>Application Programming Interface</i>
AA	Agentes Automatizados
AM	Aprendizagem de Máquina
FN	Falsos Negativos
FP	Falsos Positivos
KNN	<i>K Nearest Neighbor</i>
PM	Perceptron Multicamada
RF	<i>Random Forest</i>
RNA	Rede Neural Artificial
VN	Verdadeiros Negativos
VP	Verdadeiros Positivos

“Stay Hungry, Stay Foolish.”

(Steve Jobs; 2005)

RESUMO

A crescente utilização de redes sociais levou a também a ascensão da utilização de meios automatizados para gerar ou propagar conteúdo, a divulgações de tais conteúdos pode ser danosa a esfera social. Com isto foi descoberta uma nova gama de problemas para a identificação de tais meios automatizados, este trabalho visa analisar novas abordagens quanto ao tratamento das informações e novos modelos de aprendizagem de máquina para os perfis em redes sociais proporcionando a identificação automatizada da utilização de tais práticas. Primeiramente é realizada uma análise da base de dados disponibilizadas nos perfis, com isto é possível analisar quais características são relevantes para a classificação dos perfis. Utilizando a análise de são aplicadas técnicas de agrupamento visando analisar o resultado dos modelos utilizando está abordagem para o problema de classificação de perfis. Logo após são treinados modelos utilizando as abordagens propostas neste trabalho e a utilizando os dados de uma base de dados, onde os dados foram processados para uma melhor utilização nos modelos ao fim do treinamento os resultados serão comparadas com o estado da arte.

Palavras-chave: Aprendizado de Máquina. Rede Sociais. Bots.

ABSTRACT

The increasing use of social networks has also led to the rise of the use of automated means to generate or propagate content, the dissemination of such content can be harmful to the social sphere. With the discovery of a new range of problems for the identification of such automated media, this paper aims to analyze new approaches to information handling and new machine learning models for social networking profiles providing automated identification of the use of such practices. First, an analysis of the database made available in the profiles is performed, with this it is possible to analyze which characteristics are relevant to the classification of the profiles. Using the analysis of grouping techniques are applied to analyze the results of the models using this approach to the problem of profile classification. Soon after, models are trained using the approaches proposed in this work and using the data from a database, where the data were processed for better use in the models at the end of the training, the results will be compared with the state of the art. **Keywords:** Machine Learning. Social Networks. Bots.

LISTA DE FIGURAS

Figura 1 – Aprendizagem de Máquina (AM) supervisionada	17
Figura 2 – Aprendizagem não supervisionada	20
Figura 3 – Gráfico de saída da função binário	22
Figura 4 – Gráfico de saída da função sigmóide	23
Figura 5 – Gráfico de saída da função tangente hiperbólica	23
Figura 6 – Perceptron	23
Figura 7 – Perceptron multicamadas	24
Figura 8 – Ilustração do método empregado.	29

SUMÁRIO

1	INTRODUÇÃO	11
1.1	Objetivos	12
<i>1.1.1</i>	<i>Objetivo Geral</i>	<i>12</i>
<i>1.1.2</i>	<i>Objetivos específicos</i>	<i>12</i>
1.2	Organização do Trabalho	13
2	TRABALHOS RELACIONADOS	14
2.1	Abordagens para a identificação de Agentes Automatizados	14
2.2	Identificação de perfis por meio de análise comportamental	15
2.3	Conclusão	15
3	FUNDAMENTAÇÃO TEÓRICA	16
3.1	Aprendizagem de Máquina	16
<i>3.1.1</i>	<i>Aprendizagem de máquina supervisionada</i>	<i>16</i>
<i>3.1.1.1</i>	<i>K Nearest Neighbor (KNN)</i>	<i>17</i>
<i>3.1.1.2</i>	<i>Algoritmo de votação</i>	<i>18</i>
<i>3.1.1.3</i>	<i>Random Forest</i>	<i>19</i>
<i>3.1.2</i>	<i>Aprendizagem de máquina não supervisionada</i>	<i>19</i>
<i>3.1.2.1</i>	<i>K-means</i>	<i>20</i>
<i>3.1.3</i>	<i>Redes Neurais</i>	<i>21</i>
<i>3.1.3.1</i>	<i>Funções de ativação</i>	<i>22</i>
<i>3.1.3.2</i>	<i>Perceptron</i>	<i>23</i>
<i>3.1.3.3</i>	<i>Perceptron multicamadas</i>	<i>24</i>
3.2	Métricas	25
<i>3.2.1</i>	<i>Accuracy</i>	<i>26</i>
<i>3.2.2</i>	<i>Recall</i>	<i>26</i>
<i>3.2.3</i>	<i>Precision</i>	<i>27</i>
<i>3.2.4</i>	<i>F1-Score</i>	<i>27</i>
3.3	Características de usuários	27
3.4	Agentes Automatizados	28
4	PROCEDIMENTOS METODOLÓGICOS	29
4.1	Base de dados	29

4.2	Pré-processamento dos dados	31
4.3	Treinamento dos modelos	33
4.4	Resultados	34
5	CONCLUSÃO	36
5.1	Trabalhos Futuros	36
	REFERÊNCIAS	37

1 INTRODUÇÃO

O crescimento na utilização de redes sociais¹ nos últimos anos trouxe um novo meio de interação social, no qual as pessoas confiam em seu círculo de amizades, criado nesses espaços, e por este são influenciados. No entanto, este novo meio de interação está sujeito a manipulação por outros usuários em prol de objetivos próprios.

A manipulação de discussões em esferas sociais não é necessariamente uma interação de usuário para usuário. É perceptível a utilização de programas de computador para executar rotinas automatizadas, de modo que as rotinas são utilizadas para auxiliar usuário ou onde o usuário auxilia as mesmas para dificultar a sua detecção, pois as rotinas podem ser utilizadas de forma completamente independente sendo ativadas por um evento na rede social, caracterizando o que este trabalho se refere como Agentes Automatizados (AA).

A capacidade de conseguir popularidade e de desenvolver influência sobre aqueles em um círculo social, utilizando técnicas para a automatização de rotinas em redes sociais, demonstram o potencial de tais práticas com propósito de manipular a interação social, como no trabalho de Aiello *et al.* (2012), que apresenta dados no qual, as interações de um agente, podem influenciar o círculo social criando fluxos de propagação de informações, baseados na interferência criada pela interação.

No trabalho de Gilani *et al.* (2017a), que aborda o impacto da automatização de interações em redes sociais, neste caso o *Twitter*, é demonstrada a aplicação de AA para popularizar *URLs* na rede social, o que pode maximizar a divulgação de notícias falsas por mídias sociais.

Para a identificação de ações de AA, com a intenções de manipular ou de divulgar tópicos sensíveis em esferas sociais, ocorreu uma expansão no desenvolvimento de técnicas para realizar tal tarefa, utilizando diversas abordagens para a sua realização.

Dentre as abordagens utilizadas para a identificação, as que possuem maior destaque são: manuais, como a proposta por Forelle *et al.* (2015), em que, a identificação é desempenhada por humanos seguindo listas de checagem dos atributos do perfil e automatizadas, como demonstrado no trabalho de Davis *et al.* (2016), no qual, um modelo de aprendizagem de máquina foi empregado em uma ferramenta online para consulta². Contudo, não existe uma técnica de identificação automatizada amplamente aceita para a identificação de AA em redes sociais.

¹ <https://www.logicadigital.com.br/crescimento-das-redes-sociais/>

² [https://botometer.iuni.iu.edu/!](https://botometer.iuni.iu.edu/)

Devido a atualizações nas políticas de acesso a *Application Programming Interface* (API)s das redes sociais, existe uma certa escassez de novos dados para a realização de novas análises. Com isto a avaliação da viabilidade da utilização de coleta de dados públicos, direto das páginas da rede sociais para a criação ou expansão de novas bases por meio de *Web Scrappers*, ferramentas para automatizar a coleta de dados em web sites, é bastante relevante para o estado a tua de acesso a informações das redes sociais.

Então este trabalho propõe uma nova abordagem para classificação de AA em redes sociais, visando a avaliação do impacto em utilizar apenas características públicas para a identificação dos perfis, além de propor uma nova forma de seleção de atributos para modelos de classificação, utilizando-se de modelos de agrupamento para a seleção de atributos da base de dados.

Com isto é esperado a criação de novos modelos de classificação baseados em informações públicas e que nos apresentam os resultados de análises sem utilizar as dependências das redes sociais e sem violara a privacidade dos usuários.

1.1 Objetivos

1.1.1 Objetivo Geral

Criar um modelo de classificação AA em redes sociais, baseado em características dos usuários de rede social. Este modelo deve utilizar apenas características públicas dos usuários.

1.1.2 Objetivos específicos

- Realizar uma nova análise nas bases de dados fornecidas por trabalhos relacionados.
- Utilizar um modelo de agrupamento para selecionar as características baseadas em seu nível de relevância.
- Utilizar uma nova abordagem na base de dados utilizando modelos de Redes Neurais Artificiais.
- Realizar comparações dos resultados obtidos com este trabalho e o estado da arte através das métricas de *accuracy*, *recall*, *precision* e *f1-score*.

1.2 Organização do Trabalho

Esta proposta organiza se da seguinte forma: no capítulo 2 é descrito o estado da arte sobre identificação de AA em que este trabalho se fundamenta; o capítulo 3 apresenta conceitos necessários para o melhor entendimento do trabalho; capítulo 4 contém o processo metodológico proposto e utilizados neste trabalho; capítulo 5 traz as conclusões deste trabalho.

2 TRABALHOS RELACIONADOS

Neste capítulo são apresentados trabalhos que abordam a classificação de perfis com base em características extraídas de redes sociais. A seção 2.1 contém as diferentes abordagens de identificação de perfis; a seção 2.2 demonstra como o comportamento dos usuários foi utilizado para a classificação dos perfis; a seção 2.3 conclui o capítulo com uma relação dos trabalhos citados.

2.1 Abordagens para a identificação de Agentes Automatizados

Algumas abordagens são utilizadas para a identificação dos AA. Os principais métodos abordados até então são:

- **Colaboração coletiva**

No qual voluntários realizam o processo de classificação e utilizando uma técnica de checagem realizam a avaliação dos resultados rotulados por outros, uma abordagem como está foi proposta em Forelle *et al.* (2015) onde a identificação de AA no *Twitter*, é realizada utilizando a análise manual dos perfis.

- **Análise de rede**

A partir da análise das conexões, nas sub-redes dentro das redes sociais, é possível encontrar os AA, além de conseguir verificar o impacto de uma sub-rede em relação aos tópicos analisados. Por exemplo, Kramer (2017) que utiliza uma análise de entropia de informação, para analisar a rede e identificar os AA.

- **Aprendizado de Máquina**

Modelos de aprendizagem de máquina propostos para a identificação necessitam de uma entrada previamente rotulada, como no trabalho de Varol *et al.* (2017), que utiliza um modelo composto de algoritmos o *Random Forest* (RF), no qual é utilizado o algoritmo de árvores de decisão para compor o modelo e realizar a classificação de característica do usuário.

Stukal *et al.* (2017), também utiliza um modelo de algoritmo de composição, porém, utilizando um sistema de votação absoluta, onde cada algoritmo é responsável por uma análise completa do perfil.

2.2 Identificação de perfis por meio de análise comportamental

O comportamento dos usuários em redes sociais pode ser mapeado utilizando as informações do acesso e de como o usuário interage com a rede social. A interação de usuários com a rede produz uma série de informações. Com isto existe uma grande diversidade de formas de utilizar estas interações para identificação de perfis.

A proposta de Azab *et al.* (2016), aplica um modelo de avaliação estatística em características de usuários do *Facebook* para a realização da classificação dos perfis de humanos e AA. Já as propostas de Gilani *et al.* (2017b) e Chu *et al.* (2012) aplicam modelos de aprendizagem de máquina de forma a reealizar o processo de classificação de perfis de forma automatizada, neste caso foi utilizado o *Twitter* como fonte dos dados de usuários. Para realizar a tarefa de classificação foi empregado o RF como modelo de AM em ambos os trabalhos.

Uma outra abordagem de utilização dos atributos de perfis é proposto por Stukal *et al.* (2017), onde é empregado um modelo de votação absoluta, modelo este que é composto por outros modelos de AM, como por exemplo árvores de decisão, um processo que utiliza tal abordagem de votação, torna o modelo mais confiável para identificação já que cada modelo possui abordagens diferentes quanto a utilização dos dados para a classificação, pode levar a uma baixa taxa de detecção devido à forma de consenso utilizada.

Nos trabalhos analisados, dentre as informações dos perfis, se encontram conteúdos privados aos usuários comuns da rede social, sendo necessário a utilização da *API* para a coleta dos dados.

2.3 Conclusão

Este trabalho se assemelha aos propostos nos trabalhos de Gilani *et al.* (2017b) e Stukal *et al.* (2017) quanto a forma de automatizar a identificação de perfis de usuários, utilizando de características de usuários para a identificação. No entanto, a abordagem proposta neste trabalho, é realizada de forma diferente, já que a proposta dos estudos anteriores possuem acesso irrestrito as informações dos perfis e este trabalho busca a utilização de informações disponíveis para qualquer usuário na rede social. Além de que este trabalho propõe a utilização de modelos que até então não foram explorados para este problema, como a utilização de modelos não supervisionados para a identificação.

3 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados conceitos que possibilitam um melhor entendimento desta pesquisa; a seção a seção 3.1 apresenta conceitos de e aprendizagem de máquina que devem ser utilizados para a identificação de padrões;a seção 3.1.3 apresenta os elementos base de redes neurais artificiais; a seção 3.2 apresenta as métricas utilizadas para a avaliação dos modelos de AM;a seção 3.3 apresenta como classificar informações obtidas de redes sociais em classes de características de usuários; a seção 3.4 apresenta o que é um agente automatizado.

3.1 Aprendizagem de Máquina

Na área de inteligência artificial, há um crescente desenvolvimento da subárea de AM, para a este trabalho é empregada a definição proposta em Mitchell (1997):

Diz-se que um programa de computador aprende da experiência E com relação a alguma classe de tarefas T e medida de desempenho P, se seu desempenho nas tarefas em T, medido por P, melhorar com a experiência E.

Existem diversas formas de se abordar AM, para este trabalho serão apresentadas duas abordagens sobre o treinamento a aprendizagem supervisionada e a aprendizagem não supervisionada, também será apresentado um conjunto de técnicas que podem ser aplicadas para ambas as Rede Neural Artificial (RNA), que para este trabalho serão abordadas no âmbito de aprendizagem supervisionada.

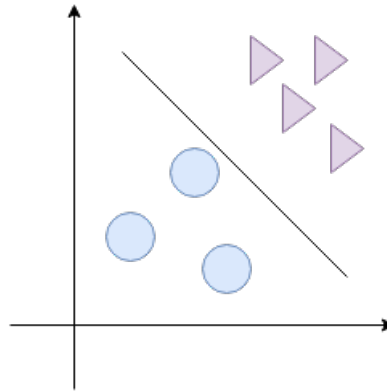
3.1.1 *Aprendizagem de máquina supervisionada*

Os algoritmos de aprendizagem supervisionada consistem de técnicas que necessitam de uma entrada com informações organizadas em pares no qual, o par é construído pelo dado a ser classificado e um rótulo que classifique aquele dado em uma das classes rotuladas (RUSSELL; NORVIG, 2009).

Modelos de aprendizagem supervisionadas são divididos em modelos de classificação e modelos de regressão. Modelos de classificação são utilizados para rotular um elemento, similar aos utilizados para o treinamento do modelo, em um dos rótulos empregados nos dados já modelos de regressão conseguem um valor como resposta aos dados de entrada a um modelo (FACELI *et al.*, 2011).

Uma forma de visualizar a classificação de dados é a linha criada entre os dois tipos de entrada na Figura 1, que pode facilmente classificar uma nova entrada pela posição em relação à linha que divide os dois conjuntos.

Figura 1 – AM supervisionada



Fonte: Elaborado pelo autor (2019)

Na figura 1 é apresentada um conjunto de elementos previamente rotulados em triângulos, círculos e a linha separando os dois grupos é representação da separação entre os grupos, que pode ser obtida utilizando um modelo de separação linear para os dados apresentados.

A classe de algoritmos AM supervisionada possui os algoritmos que são composições de outras técnicas de aprendizagem, classificados como Ensemble Learning, tendo sido utilizados dois algoritmos pertencentes esta classe neste trabalho sendo eles o Algoritmo de votação e o Random Forest juntamente como o *K Nearest Neighbor* (KNN) apresentados nas próximas subseções.

3.1.1.1 *K Nearest Neighbor* (KNN)

Como definido por Mitchell (1997), o algoritmo *K Nearest Neighbor* (KNN) utiliza uma abordagem no qual, todas os elementos de uma base correspondem pontos em um espaço n-dimensional e os vizinhos mais próximos de um elemento são definidos pela distância entre os pontos.

A classificação utilizando este algoritmo é descrita por meio dos passos a seguir.

1. Recebe a base de dados com valores previamente rotulados;
2. Recebe um dado não classificado;
3. Mede a distância do novo elemento em relação aos dados previamente atribuídos ao modelo;

4. Obtém as k, menores distâncias;
5. Verifica a classe de cada um dos elementos que tiveram as menores distâncias e conta a quantidade quantas vezes cada classe aparece;
6. Como resultado a classe que mais apareceu dentre os dados que tiveram as menores distâncias;
7. Classifica o novo dado com a classe tomada como resultado da classificação.

A medida de distância entre dois pontos pode ser realizada de diversas. Formas para exemplificar, e utilizada a Distância euclidiana, que é definida na fórmula 3.1.

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^n (x_{ir} - x_{jr})^2} \quad (3.1)$$

Onde x_i e x_j são elementos da base n atributos dos elementos x_i e x_j , respectivamente. Com o cálculo da distância é realizado a contagem dos k vizinhos mais próximos, assim contabilizando qual a classe mais comum entre os vizinhos.

3.1.1.2 Algoritmo de votação

O algoritmo de votação é a forma mais simples de criar um algoritmo de *ensemble learning*, visto que é necessário apenas a utilização de algoritmos já treinados na composição. Em um algoritmo como este os classificadores podem ser algoritmos diferentes ou podem ser compostos por algoritmos similares empregados em diferentes combinações de informações na base.

Cada algoritmo recebe os novos dados e realiza a predição. Assim compondo a forma de participação dos algoritmos na votação. A contabilização dos votos pode ocorrer de três formas (RASCHKA, 2015) sendo elas:

- **Unanimidade:** todos os participantes têm que concordar na votação.
- **Majoritário:** a maioria dos participantes ganha a votação.
- **Popularidade:** o resultado mais comum é a classe utilizada

A contabilização de votos por popularidade e majoritária funcionam de forma similar, utilizando a moda como métrica entre os resultados de predição, representada pela fórmula 3.2.

$$\hat{y} = \text{moda}\{C_1(\mathbf{x}), C_2(\mathbf{x}), \dots, C_m(\mathbf{x})\} \quad (3.2)$$

Onde C_j representa os classificadores utilizados na votação, x é o elemento que está sendo classificado e é selecionado o \hat{y} rótulo que recebe mais votos.

Além do rótulo retornado pelo classificador a votação pode utilizar a probabilidade de um elemento pertencer a uma classe como resultado para o voto. A fórmula 3.3 apresenta esta abordagem.

$$\hat{y} = \arg \max_i \sum_{j=1}^m w_j p_{ij} \quad (3.3)$$

Onde p_{ij} é a probabilidade do j -ésimo classificador para o rótulo i e w_j é o peso associado com o classificador. Desta forma temos uma maior flexibilidade do algoritmo de votação e uma maior confiança na votação.

3.1.1.3 *Random Forest*

O algoritmo de RF é uma composição de árvores de decisão, onde as árvores de decisão são maximizadas para o ganho de informação e a classificação de cada árvore é utilizada em um arranjo de votação majoritária (RASCHKA, 2015).

Árvores de decisão utilizam uma abordagem em que os dados de um problema de classificação são subdivididos em avaliações dos atributos dos dados. Em seguida, os dados são organizados na estrutura da mesma visando a melhor classificação dos dados (FACELI *et al.*, 2011). Uma árvore de decisão é representada por um grafo acíclico direcionado onde temos dois tipos de nó:

- **Nós folha** que são utilizados para rotular os elementos utilizando a moda dentre os mesmos.
- **Nós de divisão** que realizam testes condicionais onde as características são selecionadas por sua capacidade em subdividir a base dentre suas classes.

Esta abordagem do RF divide o espaço de busca entre as árvores de decisão e utiliza suas soluções para compor a solução do problema como um todo.

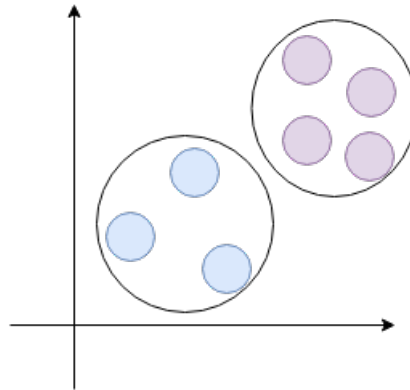
3.1.2 *Aprendizagem de máquina não supervisionada*

Os algoritmos de aprendizagem não supervisionada são modelos que podem aprender padrões de entradas fornecidas mesmo que, nenhuma informação sobre os padrões seja informado (RUSSELL; NORVIG, 2009).

Esta capacidade de retirar informações dos dados é normalmente aplicada para a detecção de grupos onde, padrões similares são colocados no mesmo grupo quando possuem similaridade entre os elementos de um grupo e diferenças dos demais itens.

A Figura 2 ilustra a identificação de dois grupos distintos pela proximidade das características dos elementos, gerando uma saída distinta de uma linha como a Figura 1.

Figura 2 – Aprendizagem não supervisionada



Fonte: Elaborado pelo autor (2019)

A identificação dos dois grupos na Figura 2 é o resultado do modelo, que utiliza a semelhança entre elementos para decidir em que grupo o elemento se encaixa melhor com os outros elementos.

Para realizar os procedimentos de agrupamento o método *K-means* foi utilizado é descrito em seguida.

3.1.2.1 *K-means*

O *K-means* é um algoritmo de clusterização que recebe como entrada uma base de dados e um número de grupos. Então é aplicada uma escolha aleatória onde k pontos são selecionados para serem os centroides dos grupos inicialmente. A partir dos centroides serão realizadas operações para que seja obtido os melhores elementos para serem utilizados como centroides, assim maximizando a separação dos entre grupos.

Os procedimentos realizados pelo *K-means* consistem em:

1. Escolha de k pontos aleatoriamente como centroides iniciais;
2. O posicionamento dos elementos baseado na distância com relação aos centroides.
3. Mede a distância do novo elemento em relação aos dados previamente atribuídos ao modelo;
4. Calcular as distâncias entre o centroides e todos os elementos do grupo;
5. Mover os centroides de forma a maximizar a densidade dos grupos e minimizar

as distâncias dos elementos;

6. Realizar os três passos anteriores até que a movimentação dos centroides não obtenham retorno positivo ou o número máximo de iterações seja alcançado;
7. Apresentar os k grupos encontrados ao fim das iterações.

A partir dos centroides iniciais os elementos restantes são reunidos no grupo do centroide mais próximo, desta forma é realizado o cálculo das distâncias totais e o próximo elemento é escolhido para centroide visando a minimização das distâncias entre os elementos do grupo e o centroide que o representa.

O algoritmo visa a minimização das distâncias entre elementos dos grupos e seu centroide, desta forma quando as distâncias dos elementos para o centroide chega ao seu menor valor, temos a melhor separação dos k grupos (KIRK, 2017).

3.1.3 *Redes Neurais*

Este trabalho utiliza como livro texto base para conceituar os pontos abordados nesta seção Haykin (2008). A definição do que é uma RNA para este trabalho adapta a definição apresentada no livro texto, onde uma rede neural é definida a seguinte forma:

Uma rede neural é um processador distribuído massivamente paralelo, composto de processamento simples unidades com propensão natural para armazenar conhecimento experimental e disponibilizá-lo para uso. Assemelha-se ao cérebro em dois aspectos:

1. O conhecimento é adquirido pela rede a partir de seu ambiente através de um processo de aprendizado.
2. Os pontos fortes da conexão interneurônio, conhecidos como pesos sinápticos, são usados para armazenar as conhecimento adquirido

RNA são modelos desenvolvidos a partir da representação do comportamento de células do sistema nervoso, de forma artificial. Neurônios, são a base da comunicação do sistema nervoso e realizam sua comunicação via sinapses, onde recebem impulsos baseados em sua experiência decidem como repassar o impulso para as demais células conectadas. Esta estrutura simples é descrita em três elementos básicos:

1. Um conjunto de conexões, onde cada uma possui um peso independente.
2. Uma função de que opera todas as entradas do neurônio, utilizando o peso da conexão.
3. Uma função de ativação que utilizando o valor processado a partir das entradas decide se passa ou não o impulso para as próximas conexões.

O primeiro item se refere a uma estrutura para representar as sinapses de sistemas nervosos orgânicos. O segundo item, tem como representação um somatório de todas as ligações multiplicadas pelo viés do neurônio. o terceiro item é a função que possibilita a propagação para as próximas conexões.

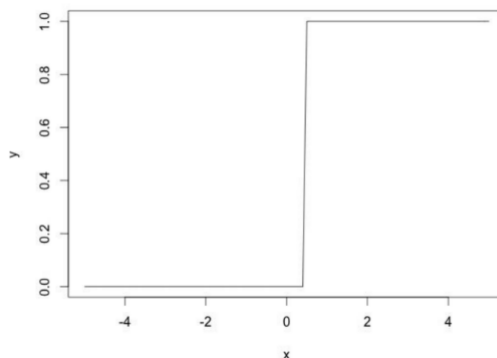
3.1.3.1 Funções de ativação

Existem diversas funções de ativação, serão apresentadas a seguir três destas funções a função degrau, sigmóide e tangente hiperbólica.

$$f(x) = \begin{cases} 1, & \text{se } x_k \geq 0 \\ 0, & \text{se } x_k < 0 \end{cases} \quad (3.4)$$

A função degrau descrita na fórmula 3.4 gera uma saída binária que resulta em um gráfico como o apresentado na Figura 3 abaixo.

Figura 3 – Gráfico de saída da função binário



Fonte: <http://www.decom.ufop.br/imobilis/redes-neurais-funcoes-de-ativacao/>

A função sigmóide descrita na fórmula 3.3, que possui gráficos como o apresentado na Figura 8.

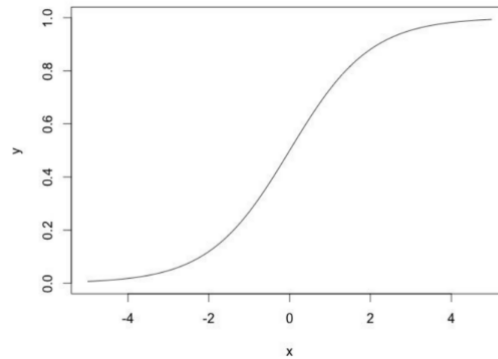
$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (3.5)$$

A função de tangente hiperbólica é uma função sigmoideal não linear, que é definida de modo geral pela fórmula 3.6.

$$f(x) = \tanh(x) \quad (3.6)$$

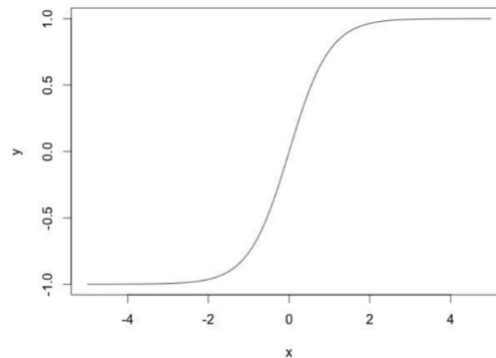
A função descrita na fórmula 3.6 gera valores entre -1 e 1 gerando gráficos como o apresentado na Figura 6

Figura 4 – Gráfico de saída da função sigmóide



Fonte: <http://www.decom.ufop.br/imobilis/redes-neurais-funcoes-de-ativacao/>

Figura 5 – Gráfico de saída da função tangente hiperbólica

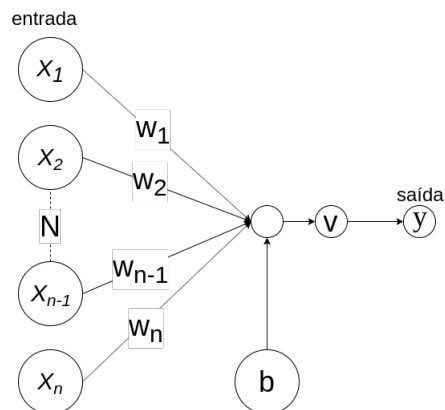


Fonte: <http://www.decom.ufop.br/imobilis/redes-neurais-funcoes-de-ativacao/>

3.1.3.2 Perceptron

O perceptron foi a primeira RNA proposta e por este motivo é uma das representações mais simples e é limitado a problemas linearmente separáveis, a Figura 6 apresenta uma representação do mesmo.

Figura 6 – Perceptron



Fonte: Elaborado pelo autor (2019)

Onde w_n é o vetor dos pesos das conexões, x_n é o valor dos inputs por, n é o numero de entradas disponíveis na RNA e b é o viés do perceptron, v pode ser representado utilizando a fórmula 3.7:

$$v = \sum_{i=1}^m w_i x_i + b \quad (3.7)$$

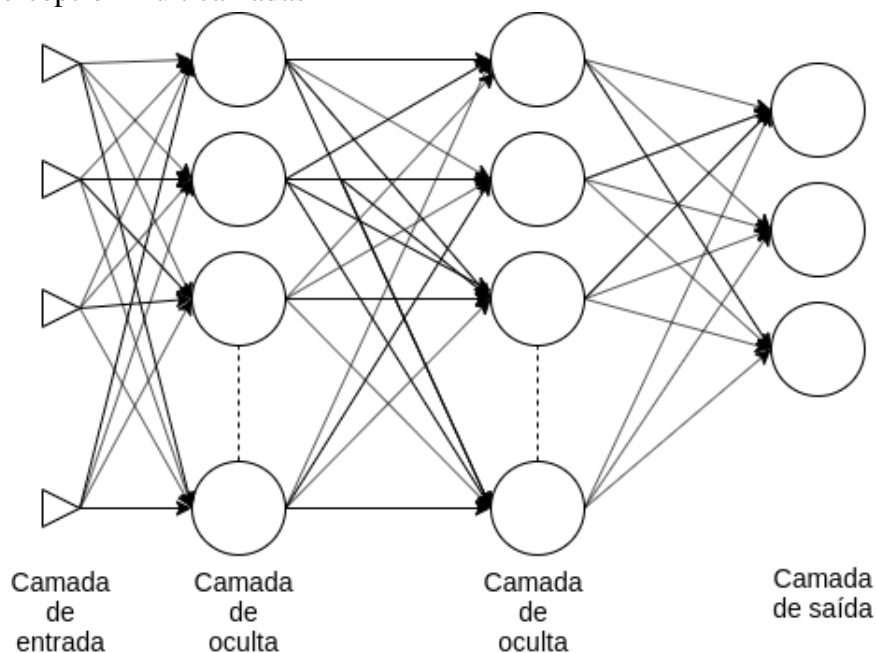
O perceptron apresentado na Figura 6 como por exemplo a classificação de imagens de forma binária ou seja entre uma classe A e uma classe B, utilizando um vetor de inputs x que será operado com o vetor de pesos das conexões w resultantes do processo de treinamento, juntamente com o viés b que é obtido no mesmo processo.

O perceptron em sua representação mais simples é um neurônio com pesos e viés ajustáveis como definido em (HAYKIN, 2008). Porém podemos utilizar diversos perceptrons organizados em uma camada, para lidarmos com problemas de classificação além da classificação binária.

3.1.3.3 Perceptron multicamadas

Perceptron multicamadas são diversas camadas de perceptrons organizados de modo que, exista uma camada de entrada dos dados, uma camadas para o processamento das saídas e uma ou mais camadas ocultas. A organização de tal RNA é apresentada na Figura 7.

Figura 7 – Perceptron multicamadas



Fonte: Elaborado pelo autor (2019)

Como demonstrado na Figura 7, todos os neurônios de uma camada possuem conexões com todos os neurônios na camada seguinte. Esta organização determina que a o processo de treinamento determinara como as características de um problema são representadas nas camadas ocultas (HAYKIN, 2008).

Por conta da organização deste modelo o mesmo necessita utilizar funções de ativação que sejam diferenciáveis tal como a tangente hiperbólica, com isto o treinamento do modelo utiliza-se de procedimentos específicos para o mesmo e o procedimento mais popular é o back propagation, este método segundo Haykin (2008) consiste em duas etapas:

- Os dados são apresentados a camada de entrada e os pesos das conexões são atualizados nas camadas até que uma resposta seja obtida na camada de saída.
- O resultado obtido é comparado com o resultado esperado e uma mensagem de erro é repercutida da camada de resposta até a de entrada atualizando os pesos das conexões.

A realização do primeiro passo é uma operação simples, já que um neurônio já realiza tal repercussão do impulsos recebidos porém o calculo do erro no sentido contrario é uma nova operação adicionada pelo perceptron multicamada.

O processo de treinamento de perceptrons multicamadas consistem em cinco passos, como primeiro passo ao iniciar uma rede neural é necessário os valores iniciais das conexões entre neurônios que de modo simples podem ser selecionados de forma aleatória, em uma distribuição uniforme respeitando os limites da função de ativação utilizadas nos neurônios, então como segundo passo os valores são apresentados a rede de modo a gera uma saída e realizar a valoração dos pesos nas conexões, como quarto passo realizamos o procedimento de correção de erro por meio da propagação de trás para frente e como quinto passo realizamos as iterações do terceiro e do quarto passo a quantidade de vezes especificada.

Ao fim dos procedimento citados obtemos um modelo que necessita ser avaliado, as métricas para realizar tais avaliações serão apresentadas na próxima seção.

3.2 Métricas

Para a avaliação dos modelos, existem métricas que conseguem dar significados aos resultados obtidos, as métricas que serão utilizadas neste trabalho para modelos de classificação são *Accuracy*, *Recall*, *Precision* e *F1-Score*.

Porém, existem métricas para a análise de modelo de aprendizagem não supervisionado nestes trabalho é utilizada a homogeneidade, que é definida como a similaridade de

elementos pertencentes ao mesmo grupo em um processo de clusterização (FACELI *et al.*, 2011).

Para as métricas de classificação, são necessárias as definições de quatro valores, que são ilustrados e descritos em seguida.

Tabela 1 – Matriz de confusão

		Classe predita	
		A	B
Classe Verdadeira	A	VP	FP
	B	FN	VN

- **Verdadeiros Positivos (VP)** A quantidade de itens classificados como A corretamente.
- **Verdadeiros Negativos (VN)** A quantidade de itens classificados como B de forma correta.
- **Falsos Positivos (FP)** A quantidade de itens que foram classificados como A de maneira incorreta.
- **Falsos Negativos (FN)** A quantidade de itens que foram classificados como humanos de forma incorreta.

3.2.1 Accuracy

A *Accuracy* consiste em:

$$Accuracy = \frac{VP + VN}{n}$$

Com isto temos o quanto o modelo acerta ($VP + VN$) das n previsões possíveis, onde n é igual a $VP + VN + FP + FN$. Assim temos a porcentagem de acerto do classificador.

3.2.2 Recall

O *Recall* é a representação de quantos dos elementos positivos foram classificados corretamente e é representada por:

$$Recall = \frac{VP}{VP + FN}$$

3.2.3 Precision

A *precision* demonstra quantos daqueles classificados como uma classe realmente pertencem a ela. A *precision* é definida por:

$$Precision = \frac{VP}{VP + FP}$$

3.2.4 F1-Score

O *f1-Score* é um valor único que demonstra o balanço entre o *recall* e o *precision*, o *f1-Score* é definida por:

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall}$$

3.3 Características de usuários

Devido a grande diversidade nas informações que podem ser obtidas a partir dos perfis de usuários e de metadados nas redes sociais este trabalho adota a definição de Varol *et al.* (2018), para a classificação de características em:

- **Características baseadas nos usuários.**

Englobando metadados dos perfis de usuários como nome, descrição, idade da conta, fuso horário e interações com postagens, incluindo taxas de interação com a rede social.

- **Características de amizades.**

Quanto a interação no meio social existem diferentes formas de relacionamento então esta classe de características se responsabiliza pelas informações sobre a relação de um usuário em relação ao outro e sobre suas interações, já que um usuário pode ser classificado como seguidor ou amigo em redes sociais.

- **Características de rede.**

Características de rede são o modo de classificar a informações provenientes da disseminação de informações pela rede social acompanhando principalmente o fluxo da informação.

- **Características de conteúdo e idioma.**

Analisar o conteúdo que está sendo compartilhado ou criado por um usuário pode fornecer informações importantes para a identificação de agentes automatizados promotores para esta análise se torna necessário um entendimento do idioma utilizado nas postagens analisadas, este tipo de informações representadas por esta classe de características.

- **Características sentimentais.**

Análise de sentimentos podem ser utilizadas para a identificação de emoções em postagens em redes sociais que pode demonstrar as reações das pessoas a um evento ou tópico de debate.

- **Características temporais.** Características temporais são informações sobre o tempo de criação e compartilhamento de postagens, que podem ser utilizados para identificar comportamentos mecânicos como diversos compartilhamentos em escalas temporais perfeitamente padronizadas.

3.4 Agentes Automatizados

Os agentes automatizados para este trabalho são qualquer tipo de automação realizada para um usuário de uma rede social que possua a intenção de realizar ações que possam dificultar a experiência de usuários que não possuem o recurso.

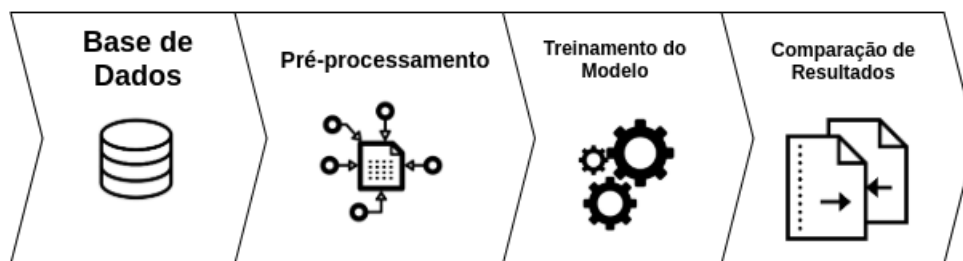
Este trabalho segue uma abordagem diferente do trabalho de Chu *et al.* (2012) que propõem a criação de três formas para classificar os usuários sendo elas humanos, bots e cyborgs, onde cyborgs são humanos com auxílio de rotinas automatizadas ou rotinas automatizadas com auxílio de humanos já os bots são agentes completamente automatizados, funcionando sem interferência humana.

4 PROCEDIMENTOS METODOLÓGICOS

Neste capítulo é descrito o processo utilizado para atingir os objetivos propostos por este trabalho. Para tal as seções posteriores demonstram os procedimentos efetuados. A seção 4.1 apresenta os dados utilizados neste trabalho; a seção 4.2 descreve como os dados foram tratados neste trabalho; a seção 4.3 demonstra a forma como se deu o treinamento dos modelos; a seção 4.4 demonstra a comparação dos resultados obtidos com o estado da arte.

A Figura (abaixo) ilustra o fluxo dos dados durante os procedimentos realizados neste capítulo.

Figura 8 – Ilustração do método empregado.



Fonte: Elaborado pelo autor (2019)

Os procedimentos realizados seguem a ordem descrita na Figura (acima), iniciando com a apresentação dos dados e os procedimentos necessários para tratar os dados utilizados neste trabalho. Utilizando os dados pré-processados os modelos foram treinados. Em seguida é realizado a comparação dos resultados

Para a realização destes experimentos foram utilizados as bibliotecas weka para o treinamento dos modelos e scikitlearn ¹ para a seleção das características, pandas ² e a linguagem de programação *python* 3.6 em um computador i3 com 4GB de *Ram*.

4.1 Base de dados

A base de dados utilizada é disponibilizada por Gilani *et al.* (2017c), contendo 3431 perfis dos quais 1939 são humanos e 1492 são AA. Estes dados foram coletados e processados por meio da ferramenta *Stweeler* (GILANI *et al.*, 2016) desenvolvida com propósito de realizar tratamentos em dados coletados diretamente da *API* do twitter. A coleta dados ocorreu em um

¹ <https://scikit-learn.org/stable/>

² <https://pandas.pydata.org/>

período de 30 dias em abril de 2016. Os dados foram rotulados por classificação manual de voluntários.

Esta base de dados foi selecionada para este trabalho (GILANI *et al.*, 2017c), pois, além de ser uma base rotulada, apresenta características acessíveis para qualquer usuário da rede social assim possibilitando uma expansão da base utilizando um *Web Scrapper*.

A seguir, são apresentadas todas as características disponibilizadas na base, sendo elas:

- **screen-name**: Nome utilizado para identificação dos perfis na rede social;
- **user-status**: Campo de texto destinado à descrição do seu perfil de usuário;
- **user-tweets**: Número de *tweets* publicados ou quantidade de publicações do perfil;
- **user-retweets**: Número de *retweets* ou compartilhamentos publicados;
- **user-favourites**: Número de publicações marcadas como favorita;
- **user-replies-and-mentions**: Número de respostas a *tweets* e número de menções ao usuário em postagens;
- **likes-per-tweet**: Média de *likes* por *tweet* publicado;
- **retweets-per-tweet**: Média de *retweets* por *tweet* publicado;
- **lists-per-user**: Média de listas publicas por usuário;
- **follower-friend-ratio**: Relação entre número de seguidores e quem o perfil segue (Amigos);
- **tweet-frequency**: Frequência de postagem dos *tweets*;
- **favourite-tweet-ratio**: Relação entre número de *tweets* e *tweets*: favoritados;
- **age-of-account-in-days**: Idade da conta em dias até a data da coleta;
- **sources-count**: Número de formas diferentes utilizadas para interagir com a rede social;
- **urls-count**: Número de *urls* nas publicações;
- **cdn-content-in-kb**: Quantidade de dados em Quilobyte gerados pelo perfil;
- **source-identity**: Identificação dos meios de publicação utilizados;

O fato do campo de descrição ser optativo com a facilidade de alteração do nome de usuário em redes sociais, torna campos textuais extremamente voláteis, em relação a perfis de usuários, em geral. Então, a avaliação de campos textuais para a classificação, não se aplica neste trabalho, portanto, os seguintes campos foram removidos.

- **screen-name**: Nome utilizado para identificação dos perfis na rede social;
- **user-status**: Campo de texto destinado à descrição do seu perfil de usuário;

A base de dados possui uma segmentação por número de seguidores. A Tabela 2 apresenta os valores das subdivisões, segmentadas em quatro grupos por do número de seguidores.

Tabela 2 – Grupos

	1 mil	100 mil	1 milhão	10 milhões	Total
Humanos	498	704	265	25	1492
Bots	791	734	389	25	1939
Perfis por seguidores	1289	1438	654	50	3431

Fonte: Elaborado pelo autor (2019)

A primeira coluna da Tabela 2 representa o segmento com 1 mil, a segunda coluna reapresenta o segmento com 100 mil, a terceira coluna representa o segmento 1 milhão e a quarta coluna representa o segmento 10 milhões de seguidores na base.

Este trabalho utiliza-se de uma base com um total de 3431 elementos com doze características de acesso público que será utilizada.

4.2 Pré-processamento dos dados

O dados obtidos são submetidos a tratamentos, com o objetivo de obter uma melhor extração de informações dos dados da base. Para que seja possível a utilização de tais dados nos modelos de AM para fins de comparação.

Os campos que possuem valores numéricos, passam por um processo no qual o objetivo é evitar a prevalência de dados desproporcionais. Para isto, método de normalização min-max foi empregado, cujos valores são tratados para que se tornem valores na escala, entre o valor mínimo, 0 e o valor máximo, 1.

Este processo propõe-se a transformar os valores com o objetivo de evitar que, alguns valores predominem sobre outros, devido a limites inferiores ou superiores muito diferentes, ou valores em escalas diferentes (FACELI *et al.*, 2011).

A base de dados apresenta um desequilíbrio, como demonstrado na Tabela 2. Buscando o melhor balançamento dos grupos, foi utilizado o processo de exclusão aleatória de elementos para que segmentos de seguidores possuam o mesmo número de integrantes, tornando os grupos balanceados.

Como resultado deste procedimento temos uma base mais equilibrada, porém, temos redução do tamanho geral da base, a Tabela 3, demonstra os valores obtidos após este

procedimento.

Tabela 3 – Grupos Balanceados

	1 mil	100 mil	1 milhão	10 milhões	Total
Humanos	498	704	265	25	1488
Bots	498	704	265	25	1488
Perfis por seguidores	996	1408	530	50	2984

Fonte: Elaborado pelo autor (2019)

Desse modo, ode-se assegurar que a base possui o mesmo número de exemplares de cada segmento pertencente a cada grupo. Desta forma o treinamento é mais confiável quanto a classificação de ambas as classes.

Como a base de dados possui uma coleção de características, realiza-se um processo de avaliação e seleção das mesmas, visando a escolha daquelas que possuam maior relevância para a classificação. Assim, teremos uma redução na quantidade de informações necessárias para a classificação.

Para realizar o processo de seleção das características foi utilizado o algoritmo de clusterização, *K-means*, visando a avaliação dos dados pós processados e para tal foi utilizada a homogeneidade dos grupos, ou seja similaridade entre elementos dos mesmo grupos, criando grupos com maior proximidade e por consequência uma melhor separação dos dados.

Com isto será analisado se características que potencializam a separação grupos podem ser utilizadas como parâmetro para melhorar os valores de classificação, desta forma este trabalho visa avaliar a possibilidade de utilizar este método de seleção.

As características foram permutadas de forma que, todas as combinações maiores que três elementos pudessem passar pelo procedimento de clusterização utilizando o algoritmo *K-means*, sendo selecionado o conjunto de características que obteve a maior homogeneidade na clusterização. As características selecionadas utilizando este processo são:

- **user-retweets**: Número de *retweets* ou compartilhamentos publicados.
- **tweet-frequency**: Frequência de postagem dos *tweets*.
- **favourite-tweet-ratio**: Relação entre número de *tweets* e *tweets*: favoritados.

Com a utilização das características selecionadas nesta etapa, obtém-se dados com maior similaridade intragrupo e uma menor similaridade intergrupo, visto que a semelhança entre elementos dos mesmo grupos está maior.

4.3 Treinamento dos modelos

Como proposto neste trabalho serão descritos os procedimentos realizados para treinar e avaliar modelos de AM, visando a aplicação de técnicas pouco exploradas no problemas de identificação de AA. Serão utilizado quatro modelos de classificação são eles RF, KNN, e Perceptron Multicamada (PM).

Para a análise dos modelos devem ser calculadas a *accuracy*, para média de vezes que os modelos acertam, o *precision* que tem o valor de quantos dos elementos foram classificados como uma classe realmente pertenciam a mesma, *recall* que traz a frequência que os elementos são rotulados como uma certa classe está corretamente rotulado e o *f1-score* que demonstra uma avaliação geral do modelo com um valor único.

Os três modelos recebem os dados pré-processados, e seguem os procedimentos de treinamento de forma paralela. Os modelos foram treinados utilizando a validação cruzada com 10 *folds*. Logo após o treinamento será realizada a avaliação dos resultados obtidos, visando selecionar o modelo que melhor se adapta a este problema.

O treinamento dos modelos foi dividido em dois experimentos, o primeiro utiliza a seleção descrita na seção anterior para a classificação e o segundo utilizando todas as características, depois os resultados obtidos nos modelos serão comparados para que a abordagem que possui melhores métricas possa ser selecionada.

Os parâmetros utilizados para o treinamento dos algoritmos para classificação no primeiro experimento estão descritos em seguida:

- KNN: distância de *manhattan* e 47 vizinhos;
- RF: critério utilizado é o *entropy*, profundidade máxima de 8 e 100 estimadores;
- PM: taxa de aprendizado 0.3, *momentum* 0.2, limite de interações 500, camadas ocultas 7.

Os parâmetros utilizado no segundo experimento foram:

- KNN: distância euclidiana e 47 vizinhos;
- RF: critério utilizado é o *gini*, profundidade máxima de 8 e 100 estimadores;
- PM: taxa de aprendizado 0.3, *momentum* 0.2, limite de interações 500, camadas ocultas 7.

Além dos modelos apresentados, também foi empregado um modelo de votação utilizando os modelos propostos, visando obter um modelo mais robusto por meio da técnica de *Ensemble Learning*, assim temos em ambos os experimentos modelos de votação, compostos pelos modelos resultantes de cada experimento.

A Tabela 4 apresenta os resultados obtidos nos experimentos utilizando a abordagem

de seleção.

Tabela 4 – Resultados dos testes com seleção de características

	accuracy (%)	f1-score(%)	recall(%)	precision(%)
Random Forest	62.0	62.1	62.1	62.1
KNN	61.1	61.2	61.2	61.2
PM	67.72	67.4	67.7	68.5
votação	62.3	62.3	62.3	62.3

Fonte: Elaborado pelo autor (2019)

O segundo experimento, utiliza todas às doze características para possa ser avaliado a utilização do procedimento de seleção realizado para a seleção de características. A Tabela 5 apresenta os resultados desta etapa.

Tabela 5 – Resultados dos testes com todas características

	accuracy (%)	f1-score(%)	recall(%)	precision(%)
Random Forest	76.4	76.5	76.5	76.5
KNN	50.0	33.4	50.0	75.0
PM	99.5	99.5	99.5	99.5
votação	89.0	88,9	89.0	91.0

Fonte: Elaborado pelo autor (2019)

A partir dos resultados podemos observar que a técnica de seleção de características, não obtém ganho quanto a utilização das características apenas pré processadas, além de podermos observar que o desempenho dentre os modelos apresentados o PM possui melhores métricas em relação aos demais e inclusive a o modelo de votação que poderia apresentar maior robustez quanto as métricas. Devido aos resultados obtidos podemos perceber que o PM do segundo experimento possui um desempenho muito superior aos demais e portanto este é o modelo que utilizaremos como produto final deste processo.

A próxima etapa compara os resultados obtidos pelo modelo PM, como foi descrito anteriormente e o estado da arte em classificação de perfis em redes sociais.

4.4 Resultados

Esta seção compara os resultados obtidos neste trabalho e os resultados encontrados em trabalhos relacionados.

A Tabela 6 apresenta as métricas para a comparação entre as diferentes propostas.

Como apresentado na tabela 6, o modelo resultante deste trabalho possui um desem-

Tabela 6 – Tabela de comparação com trabalhos anteriores

	accuracy (%)	f1-score(%)	recall(%)	precision(%)
Gilani <i>et al.</i> (2017c)	86.4	83.6	82.2	85.4
Stukal <i>et al.</i> (2017)	Não informado	Não informado	77.0	99.0
Varol <i>et al.</i> (2017)	89.0	Não informado	Não informado	Não informado
PM	99.5	99.5	99.5	99.5

Fonte: Elaborado pelo autor (2019)

penho superior a estado da arte em classificação de perfis, mesmo quando comparado com os resultados de Gilani *et al.* (2017c), quem originalmente produziu a base de dados que este trabalho utiliza. Assim temos um modelo bem mais robusto quanto a classificação e com melhores métricas para a confiabilidade da classificação devido aos resultados superiores quanto as demais métricas apresentadas.

5 CONCLUSÃO

Os objetivos propostos no trabalho foram alcançados, as abordagens de agrupamentos foram testadas assim como a utilização de novos modelos para o problema abordado neste trabalho. Tivemos grandes resultados quanto a utilização do modelo de RNA o que trás grande uma contribuição, já que com os resultados obtidos neste trabalho conseguimos superar os resultados obtidos, no geral, inclusive o trabalho que fornece a base de dados utilizadas neste trabalho.

Podemos analisar a proposta de agrupamento de características visando a melhora da classificação e temos resultados de que está abordagem não trás benefícios para este problema em específico.

Com isto adicionamos mais um modelo promissor para classificação de perfis em redes sociais, melhorando a capacidade de detecção de AA em redes sociais. Temos um novo modelo e uma área de expansão para este problema que sempre está em evolução, assim como as abordagens para detecção, a próxima seção apresenta propostas para melhorias e incrementos do modelo e de novas abordagens em trabalhos futuros.

5.1 Trabalhos Futuros

Este trabalho apresenta uma nova abordagem tanto no uso das informações e de perfis quanto na técnica de seleção das características utilizadas para a classificação de perfis.

A expansão da base de dados é um dos pontos esperados para a melhor análise das características e a criação de modelos com melhores taxas de classificação correta. Com uma base maior será possível a utilização de modelos que necessitam de grandes quantidades de dados como as redes neurais e a sua grande variedade de arquitetura de classificação e de agrupamento.

Em trabalhos posteriores é promissora a adição processo de classificação os procedimentos utilizados em processamento de linguagem natural para viabilização da análise de campos textuais como os *tweets* que pode ser recuperados utilizando um *web scrapper*. Assim como a utilização de RNA mais sofisticadas como por exemplo aplicando técnicas de aprendizagem recorrente garantindo uma grande adaptação para o modelo em relação as mudanças dos AA

Com propostas como esta a utilização de redes sociais caminha para um estado de utilização mais saudável reduzindo a influência de usuários que se utilizam de artifício para criar ou propagar informações danosas em esferas sociais.

REFERÊNCIAS

- AIELLO, L. M.; DEPLANO, M.; SCHIFANELLA, R.; RUFFO, G. People are Strange when you're a Stranger: Impact and Influence of Bots on Social Networks. **Links**, v. 697, n. 483,151, p. 1–566, 2012.
- AZAB, A. E.; IDREES, A. M.; MAHMOUD, M. A.; HEFNY, H. Fake Account Detection in Twitter Based on Minimum Weighted Feature set. **International Journal of Computer, Electrical, Automation, Control and Information Engineering**, v. 10, n. 1, p. 13–18, 2016.
- CHU, Z.; GIANVECCHIO, S.; WANG, H.; JAJODIA, S. Detecting automation of Twitter accounts: Are you a human, bot, or cyborg? **IEEE Transactions on Dependable and Secure Computing**, v. 9, n. 6, p. 811–824, 2012. ISSN 15455971.
- DAVIS, C. A.; VAROL, O.; FERRARA, E.; FLAMMINI, A.; MENCZER, F. Botornot: A system to evaluate social bots. In: INTERNATIONAL WORLD WIDE WEB CONFERENCES STEERING COMMITTEE. **Proceedings of the 25th International Conference Companion on World Wide Web**. [S.l.], 2016. p. 273–274.
- FACELI, K.; LORENA, A. C.; GAMA, J.; CARVALHO, A. C. P. d. L. *et al.* Inteligência artificial: Uma abordagem de aprendizado de máquina. 2011.
- FORELLE, M.; HOWARD, P. N.; MONROY-HERNANDEZ, A.; SAVAGE, S. Political Bots and the Manipulation of Public Opinion in Venezuela. **Ssrn**, p. 1–8, 2015. ISSN 1556-5068.
- GILANI, Z.; FARAHBAKHS, R.; CROWCROFT, J. Do Bots impact Twitter activity? **Proceedings of the 26th International Conference on World Wide Web Companion - WWW '17 Companion**, p. 781–782, 2017. Disponível em: <<https://dl.acm.org/citation.cfm?id=3054255&CFID=1025694351&CFTOKEN=28339062%0Ahttp://dl.acm.org.ez1.periodicos.capes.gov.br/citation.cfm?id=3054255&CFID=962728139&CFTOKEN=58825641%0Ahttp://dl.acm.org/citation.cfm?doid=3041021.3054255>>.
- GILANI, Z.; FARAHBAKHS, R.; TYSON, G.; WANG, L.; CROWCROFT, J. An in-depth characterisation of Bots and Humans on Twitter. p. 1–18, 2017. Disponível em: <<http://arxiv.org/abs/1704.01508>>.
- GILANI, Z.; KOCHMAR, E.; CROWCROFT, J. Classification of Twitter Accounts into Automated Agents and Human Users. **Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017 - ASONAM '17**, p. 489–496, 2017. Disponível em: <<http://dl.acm.org/citation.cfm?doid=3110025.3110091>>.
- GILANI, Z.; WANG, L.; ALMEIDA, M.; CROWCROFT, J.; ALMEIDA, M.; FARAHBAKHS, R. Stweeler: A Framework for Twitter Bot Analysis. **Proceedings of the 25th International Conference Companion on World Wide Web - WWW '16 Companion**, n. 2, p. 37–38, 2016. Disponível em: <<http://dl.acm.org/citation.cfm?doid=2872518.2889360>>.
- HAYKIN, S. **Neural Networks and Learning Machines**, 3/E. [S.l.]: Pearson Education India, 2008.
- KIRK, M. **Thoughtful Machine Learning with Python: A Test-driven Approach**. [S.l.]: "O'Reilly Media, Inc.", 2017.
- KRAMER, S. **Identifying viral bots and cyborgs in social media - O'Reilly Media**. 2017.

MITCHELL, T. M. **Machine Learning**. 1. ed. New York, NY, USA: McGraw-Hill, Inc., 1997. ISBN 0070428077, 9780070428072.

RASCHKA, S. **Python Machine Learning**. Birmingham, UK: Packt Publishing, 2015. ISBN 1783555130.

RUSSELL, S. J.; NORVIG, P. **Artificial Intelligence: A Modern Approach, 3rd edition**. [S.l.: s.n.], 2009. ISSN 0269-8889. ISBN 9780136042594.

STUKAL, D.; SANOVICH, S.; BONNEAU, R.; TUCKER, J. A. Detecting Bots on Russian Political Twitter. **Big Data**, v. 5, n. 4, p. 310–324, 2017. ISSN 2167-6461. Disponível em: <<http://online.liebertpub.com/doi/10.1089/big.2017.0038>>.

VAROL, O.; DAVIS, C. A.; MENCZER, F.; FLAMMINI, A. Feature Engineering for Social Bot Detection. **Feature Engineering for Machine Learning and Data Analytics**, CRC Press, p. 311, 2018.

VAROL, O.; FERRARA, E.; DAVIS, C. A.; MENCZER, F.; FLAMMINI, A. Online human-bot interactions: Detection, estimation, and characterization. **arXiv preprint arXiv:1703.03107**, 2017.