Revista Eletrônica de Sistemas de Informação
ISSN 1677-3071

v. 14, n. 1

jan-abr 2015 - Edição Temática sobre Análise de Redes Sociais e Mineração

doi:10.21529/RESI.2015.1401

Sumário

Editorial

BrASNAM

ISSN: 1677-3071

Esta revista é (e sempre foi) eletrônica para ajudar a proteger o meio ambiente, mas, caso deseje imprimir esse artigo, saiba que ele foi editorado com uma fonte mais ecológica, a *Eco Sans*, que gasta menos tinta.

This journal is (and has always been) electronic in order to be more environmentally friendly. Now, it is desktop edited in a single column to be easier to read on the screen. However, if you wish to print this paper, be aware that it uses Eco Sans, a printing font that reduces the amount of required ink.

APRENDIZADO DE MÁQUINA PARA ROTULAÇÃO AUTOMÁTICA DE USUÁRIOS DE UMA REDE SOCIAL ACADÊMICA

MACHINE LEARNING FOR AUTOMATIC LABELLING OF USERS OF AN ACADEMIC SOCIAL NETWORK

(artigo submetido em março de 2013)

Bruno Vicente Alves de Lima
Mestre em Ciência da Computação pela Universidade Federal do Piauí
brunovicente@ufpi.edu.br

Vinicius Ponte Machado
Doutor em Engenharia Elétrica pela Universidade Federal do Rio Grande do Norte e Professor da Universidade Federal do Piauí
vinicius@ufpi.edu.br

Lucas Araújo Lopes
Mestre em Ciência da Computação pela Universidade Federal do Piauí
lucaslopes@ufpi.edu.br

ABSTRACT

Social networks have become relevant in the Internet due to the great variety of Web sites that use the concept. Its users form databases that provide an important way of sharing, organizing, finding content and making contacts. Thus, Scientia.Net is a social networking site that integrates information from various Internet services (forums, article repositories, websites, blogs and other social networks). Besides, the tool provides the user interaction (students, teachers and researchers) for academic purposes, based on their common interests. This paper presents an application developed to automatically group Scientia.Net users, showing the performance of various machine learning algorithms, offering to Scientia.Net a sorting mechanism that presents a list of other researchers to each user of the network, based on their common interests. With this, we intend to contribute to the interaction among users with similar profiles, allowing an improvement in the productivity of their research efforts. Furthermore, this paper proposes a model that uses a combination of supervised and unsupervised learning algorithms to create groups and identify users based on their relevant attributes.

Key-words: machine learning; classification; scientia.net; cluster; profile.

RESUMO

Redes sociais tornaram-se especialmente relevantes na Internet devido à grande variedade de *sites* Web que utilizam o conceito. Seus usuários formam bases de dados que proveem um importante meio de compartilhar, organizar e encontrar conteúdo, estabelecer contatos com base em interesses comuns. Dessa forma, o Scientia.Net é um *site* de rede social que integra informações contidas em diversos serviços da Internet (fóruns, repositórios de artigos, *sites*, *blogs* e demais redes sociais). Além disso, a ferramenta provê a interação de seus usuários (estudantes, professores e pesquisadores) para fins acadêmicos, com base nos seus interesses em comum. Este artigo apresenta uma aplicação desenvolvida para agrupar de forma automática os usuários do Scientia.Net, mostrando o desempenho de vários algoritmos de aprendizagem de máquina, visando a oferecer ao Scientia.Net um mecanismo de classificação que apresente a cada usuário da rede, uma relação de outros pesquisadores com base nos seus interesses em comum. Com isso, pretende-se contribuir para a interação entre usuários de perfis semelhantes e assim possibilitar que estes melhorem a produtividade de suas pesquisas, ao aumentar sua capacidade de troca de conhecimento. Além disso, o presente artigo propõe um modelo que utiliza uma combinação entre algoritmos com aprendizagem supervisionada e não-supervisionada com o objetivo de criar grupos e identificar quais atributos podem defini-los.

Palavras-chave: aprendizado de máquina; classificação; Scientia.Net; cluster; perfil.

1 INTRODUÇÃO

Na sociedade atual, os sistemas de redes sociais se destacam como um meio de interação muito utilizado pelos usuários da Internet. Neste cenário, pode-se começar a refletir sobre uma rede social não só capaz de unir pessoas com interesses diversos, mas também unir pesquisadores, facilitando a comunicação e ajudando a ter acesso a informações acadêmicas relevantes, tais como artigos publicados em sua área de interesse, ocorrência de eventos científicos e iterações com outros pesquisadores. Com o objetivo de contribuir para que ocorra essa facilidade para os pesquisadores, propõe-se uma rede social chamada *Scientia.Net*.

O *Scientia.Net* visa a agregar aos seus usuários itens de relevância acadêmica relacionados ao seu perfil. Por isso, esta rede social possui um mecanismo de classificação automática de usuários. Com o *Scientia.Net* pretende-se ter um mecanismo que classifica os usuários de acordo com seu perfil acadêmico, permitindo-lhes que tenham contato com pesquisadores de sua área de interesse. Em um trabalho anterior (MACHADO *et al.,* 2011) mostrou-se a criação de um mecanismo utilizando redes neurais artificiais para a classificação dos usuários do *Scientia.Net*. Em outro trabalho do grupo, utilizou-se algoritmo de aprendizado não-supervisionado para classificar os usuários (LIMA, MACHADO, IBIAPINA, 2012).

Com a utilização de técnica de aprendizagem de máquina não-supervisionada, os usuários são agrupados, sendo que os grupos gerados no processo não são rotulados. Ou seja, para determinar em que classe cada usuário foi classificado foi necessário analisar todos os grupos de forma manual. Este artigo apresenta uma proposta para resolver o problema de rotulação dos grupos de usuários gerados pelos algoritmos de classificação dos usuários. Esta abordagem utiliza a combinação de um algoritmo de aprendizado não-supervisionado e um algoritmo supervisionado.

Além desta introdução, na seção 2 faremos uma breve descrição do que é o Scientia.Net. Na seção 3 discutiremos as principais técnicas utilizadas neste trabalho e na seção 4 uma descrição da base de dados utilizada. A seguir, trata-se da rotulação das classes de usuários para, por fim, se apresentar uma breve conclusão do trabalho realizado.

2 SCIENTIA.NET

O *Scientia.Net* é uma rede social voltada para o ambiente acadêmico com conteúdos específicos para cientistas que desejam compartilhar suas pesquisas ou avançar em seus trabalhos por meio da interação com outros pesquisadores. É baseada na Internet e foi criada implementando ferramentas que permitem a interação entre seus usuários (estudantes, professores e pesquisadores) com base nos seus interesses em comum, que são identificados automaticamente por meio de algoritmos de aprendizagem de máquina.

Além disso, visa a agregar aos seus usuários, de forma automática, itens de relevância relacionados ao seu perfil. Ou seja, de acordo com o perfil do pesquisador, são sugeridos artigos, eventos e contatos de outros

doi: 10.21529/RESI.2015.1401004

pesquisadores. Dessa forma o *Scientia.Net* cria um grande agregador de informações acadêmicas contidas em diversos serviços da Internet, tais como fóruns, repositórios de artigos, *sites*, *blogs* e demais redes sociais (Figura 1), permitindo aos seus usuários uma melhoria na produtividade de suas pesquisas, além de mecanismos para interatividade e troca de conhecimento entre pesquisadores.

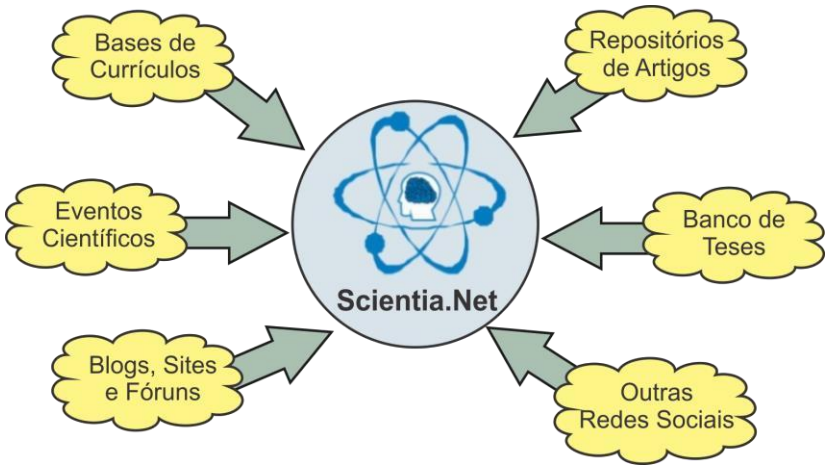

Figura 1. Scientia.Net

Fonte: elaborada pelos autores

O *Scientia.Net*, diferentemente de outras redes sociais, oferece um mecanismo de classificação automático de seus usuários e de conteúdo. Esse diferencial tem como objetivo oferecer a cada usuário do *Scientia.Net* uma relação de outros usuários cujos perfis e interesses são semelhantes. Atualmente o *Scientia.Net* possui um protótipo funcional classificando usuários utilizando a Rede de *Kohonen* e classificando eventos e artigos científicos utilizando Redes Neurais Multicamada (Figura 2).

Figura 2. Tela de perfil de um usuário no Scientia.Net

Fonte: capturada do sistema pelos autores

3 REFERENCIAL TEÓRICO

Para o desenvolvimento deste trabalho utilizaram-se técnicas de aprendizado de máquina supervisionado e não-supervisionado aplicadas ao *Scientie.Net*. Nas seções abaixo discutimos sobre tais técnicas de aprendizado.

3.1 APRENDIZADO DE MÁQUINA

O aprendizado de máquina pode ser descrito como o desenvolvimento de técnicas computacionais sobre o aprendizado bem como a construção de sistemas capazes de adquirir conhecimento de forma automática (MITCHELL, 1997).

A intuição humana não pode, nesse caso, ser totalmente abandonada, desde que o programador do sistema apresente os dados representativos e os mecanismos usados para a caracterização dos dados. Existem três tipos de aprendizado de máquina:

- aprendizado supervisionado – em que existem dados de entrada com suas respectivas saídas, para serem apresentadas ao algoritmo de aprendizado utilizado durante o processo de treinamento (BRAGA; CARVALHO; LUDEMIR, 2007).
- aprendizado não-supervisionado – neste caso supõe-se que o conjunto de exemplos não está rotulado, com isto o sistema tenta classificar estes conjuntos agrupando os semelhantes em determinadas classes (RUSSELL; NORVING, 2004).
- aprendizado por reforço – consiste em mapear situações para ações de modo a maximizar um sinal de recompensa numérico. A ideia principal deste tipo de aprendizado é simplesmente captar os aspectos mais importantes do problema colocando um agente que vai interagir com o ambiente para alcançar uma meta (SUTTON; BARTO, 1998).

Nas próximas subseções são descritos os algoritmos utilizados neste trabalho.

3.2 REDES NEURAIS ARTIFICIAIS

As redes neurais artificiais (RNA) foram introduzidas na década de 40 pelo matemático *Walter Pitts* e neurofisiologista *McCulloch* (LUDWIG JUNIOR; COSTA, 2007). As RNAs são modelos matemáticos que visam a simular o funcionamento do cérebro humano. São compostas por unidades menores chamadas de neurônios artificiais (BRAGA; CARVALHO; LUDEMIR, 2007).

Uma abordagem utilizada das RNAs são as redes neurais multicamada (MLP), que são formadas por camada de neurônios artificiais. Possuem uma camada de entrada, em que são fornecidos os padrões de entrada, uma ou mais camadas ocultas e uma camada de saída, por onde a rede emite a resposta (BRAGA; CARVALHO; LUDEMIR, 2007). Para realizar o treinamento da MLP, utiliza-se o algoritmo *backpropagation*.

O algoritmo *backpropagation* é um algoritmo supervisionado, utilizado para treinar uma rede neural com múltiplas camadas. É executado em duas fases: *forward*, em que é apresentado o padrão de entrada e emitida uma saída pela rede, e *backward*, em que a saída obtida é comparada com a saída desejada e assim calculado o erro. O erro obtido na camada de saída é usado para ajustar diretamente os seus pesos. Posteriormente, esse erro é propagado para as camadas anteriores, até a camada de saída, com o objetivo de ajustar os pesos de todas as conexões da rede neural (LUDWIG JUNIOR; COSTA, 2007).

3.3 MÁQUINA DE VETOR DE SUPORTE

Máquinas de vetor suporte (SVM) constituem uma técnica para classificação e regressão (CORTES; VAPNIX, 1995). O algoritmo de aprendizagem SVM pode ser usado para construir diversos tipos de máquinas de aprendizagem, como, por exemplo, máquinas de aprendizado polinomial, RBFs e Redes Neurais Multicamada. O número de unidades escondidas em cada um desses casos é automaticamente determinado pelo algoritmo de aprendizagem SVM.

Basicamente, o SVM é um algoritmo linear que constrói hiperplanos, com o objetivo de otimizar, ou seja, encontrar hiperplanos que maximizem a margem de separação das classes, para separar os padrões de treinamento (HAYKIN, 2001).

3.4 REDE DE *KOHONEN*

A rede de *Kohonen*, ou mapas auto-organizáveis, tem como princípio a aprendizagem competitiva, simulando processos específicos do cérebro humano na aprendizagem por respostas sensoriais.

O treinamento de uma rede de *Kohonen* é competitivo e não-supervisionado. Este algoritmo organiza os neurônios em vizinhanças locais. Cada vez que um novo padrão é apresentado à rede, os neurônios competem entre si para ver qual deles gera a melhor saída. Escolhido o neurônio vencedor e seus vizinhos, dentro de um raio ou área de vizinhança, atualizam seus pesos. Durante o treinamento, a taxa de aprendizagem e o raio de vizinhança são decrementados à medida que o algoritmo vai sendo executado (BRAGA; CARVALHO; LUDEMIR, 2007).

3.5 ALGORITMO K-MEANS

O k-means é uma técnica de aprendizado de máquina não-supervisionado apresentada por MacQueen (1967) e tem o objetivo de criar partições de uma população n-dimensional em *k* grupos em uma dada base de dados.

O algoritmo *k-means* utiliza um parâmetro de entrada *k*, que determina a quantidade de *clusters* com n elementos cada. Após a execução pretende-se obter uma alta similaridade dos elementos de um grupo e baixar a similaridade entres os *clusters* criados pelo algoritmo (SOUSA; ESMIN, 2011).

No algoritmo *k-means* são escolhidos aleatoriamente *k* objetos da base de dados como centros iniciais de cada grupo criado. Posteriormente, atribui-se cada objeto ao *cluster* ao qual o objeto é mais similar, de acordo com o calor médio dos objetos igualmente agrupados. Então, são atualizadas todas as médias dos *clusters*, ou seja, calcula-se a média dos objetos para cada grupo. E, por fim, é testado o critério de parada, então finalizado o algoritmo, ou é realizada novamente a atribuição de objetos aos *clusters*.

4 BASE DE DADOS UTILIZADA

A base de dados possui cerca de dois mil usuários de vinte áreas distintas do conhecimento e foi estruturada utilizando o gerenciador de banco de dados *MySql*.

Essa estrutura da base é formada por oito atributos que representam a parte acadêmica do perfil dos usuários. Estes atributos são: graduação, área de interesse do usuário, mestrado e doutorado e suas respectivas subáreas.

5 CLASSIFICAÇÃO DE USUÁRIOS

5.1 TRABALHOS RELACIONADOS

Anteriormente a este trabalho, uma série outros foi realizada classificando usuários com base em algoritmos de aprendizado de máquina. Em um primeiro trabalho, apresentou-se uma aplicação utilizando redes neurais artificiais para classificar automaticamente usuários do Scientia.Net, de acordo com o perfil acadêmico (MACHADO *et al.,* 2011).

Com o objetivo de melhorar os resultados, realizou-se outro experimento e ampliou-se a base de dados. Neste caso foi utilizado o método de *Cross-Validation* (validação cruzada) para validar os resultados (LIMA; MACHADO; ARAÚJO, 2011).

Por fim, realizou-se uma análise comparativa de algoritmos de aprendizado não-supervisionado para a classificação automática de usuários. Neste ponto do trabalho, utilizou-se o gerenciador de banco de dados mencionado na seção 4. Os métodos utilizados no trabalho foram: rede de *Kohonen*, algoritmo *Cobweb* e algoritmo *K- means* (LIMA; MACHADO; ARAÚJO, 2011).

Em nossas pesquisas conseguimos detectar alguns trabalhos semelhantes em que são aplicados mecanismos inteligentes envolvendo redes sociais. Em Valiati *et al.* (2012) pode-se visualizar um mecanismo de detecção de conteúdo relevante e usuários fluentes na rede social *Twitter*.

Em um trabalho publicado por Pennacchiotti e Popescu (2011) mostrou-se a classificação de usuários da rede social *Twitter* levando em consideração o perfil dos usuários, mensagens de usuários, comportamentos dos usuários a partir dos *tweets* e características desses usuários na rede social. Abordagem semelhante foi adotada por Valiati *et al.,* 2012).

5.2 METODOLOGIA PARA CLASSIFICAR OS USUÁRIOS

Para a classificação dos usuários foram utilizados quatros algoritmos de aprendizado de máquina, descritos na seção 2. A aplicação criada neste trabalho foi implementada na linguagem Java, com o objetivo de utilizar os algoritmos presentes no *WEKA* (WITTEN; FRANK, 2005), com exceção da rede de *Kohonen*, que foi implementada separadamente, pois não há implementações deste método nas bibliotecas do *WEKA*.

A tabela da base de dados de usuários foi replicada para mais nove tabelas com os mesmos registros, alterando a ordem de inserção de registros para executar os algoritmos, com o objetivo de garantir a legitimidade dos resultados, uma vez que os algoritmos são sensíveis à ordem de apresentação. Os resultados das execuções foram reunidos para se chegar a uma média dos resultados. Os algoritmos foram executados em um computador com Windows 8 Professional como sistema operacional, memória RAM de 6 GB, processador Intel Core i5 2,5 GHz.

Para a classificação utilizando os algoritmos de aprendizado de máquina supervisionado é necessário definir as classes nas quais se deseja classificar os dados (RUSSELL; NORVING, 2004). Para as redes neurais multicamada e a máquina de vetor de suporte foram definidas vinte classes, de acordo com a área de interesse dos usuários, com base nas áreas de conhecimento especificadas pela Capes[1]. Para os algoritmos não-supervisionados foram analisados manualmente os grupos gerados após a execução e determinada a homogeneidade de tais grupos, levando em consideração a similaridade de todos os atributos dos perfis inclusos em um grupo. Assim, foi possível determinar se um usuário estava inserido em um grupo erroneamente.

Para todos os algoritmos a base de dados foi dividida em duas partes: treinamento e teste. Utilizou-se inicialmente um total de 10% dos dados para treinar e executou-se dez vezes com os registros das dez tabelas em ordem aleatória. De forma análoga, realizou-se o experimento com 20%, 30%, 40%, 50% e 60% dos dados no treinamento. A porcentagem restante foi usada para testar os algoritmos.

Após determinar as classes para os algoritmos supervisionados e os grupos para os algoritmos não-supervisionados, analisou-se cada classe e cada grupo separadamente, para determinar a taxa de erro em porcentagem, isoladamente. A taxa de erro considerada foi a média das dez execuções, a partir da qual se determinou a taxa de erro médio geral para o algoritmo. Foi analisado também o tempo de execução médio dos algoritmos, como a média dos tempos obtidos nas dez execuções de cada algoritmo (tempo de treinamento).

---

[1] Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (http://www.capes.gov.br)

5.3 RESULTADOS DA CLASSIFICAÇÃO

Na classificação de usuários, os resultados foram divididos de acordo com as duas abordagens de aprendizado utilizadas neste trabalho (supervisionadas e não-supervisionadas) pois, como os grupos e as classes não são sempre compatíveis, torna-se inviável comparar os algoritmos não-supervisionados com algoritmos supervisionados.

Representando os métodos supervisionados, utilizaram-se as redes neurais multicamada e a máquina de vetor de suporte e para o não-supervisionado utilizaram-se a rede de *Kohonen* e o algoritmo *K-means*. A Figura 3 mostra os tempos de execuções dos algoritmos para a classificação dos usuários, levando em consideração a porcentagem de dados utilizados no treinamento.

O gráfico da Figura 3 mostra o tempo de execução dos algoritmos testados neste trabalho. As redes neurais multicamada e a rede de *Kohonen* obtiveram uma queda significativa com o aumento dos dados do treinamento. O algoritmo *K-means* e a máquina de vetor de suporte também apresentaram oscilações em seus tempos de execução. Porém, mesmo aumentando esses dois métodos obtiveram menor tempo de execução em todos os pontos do experimento.

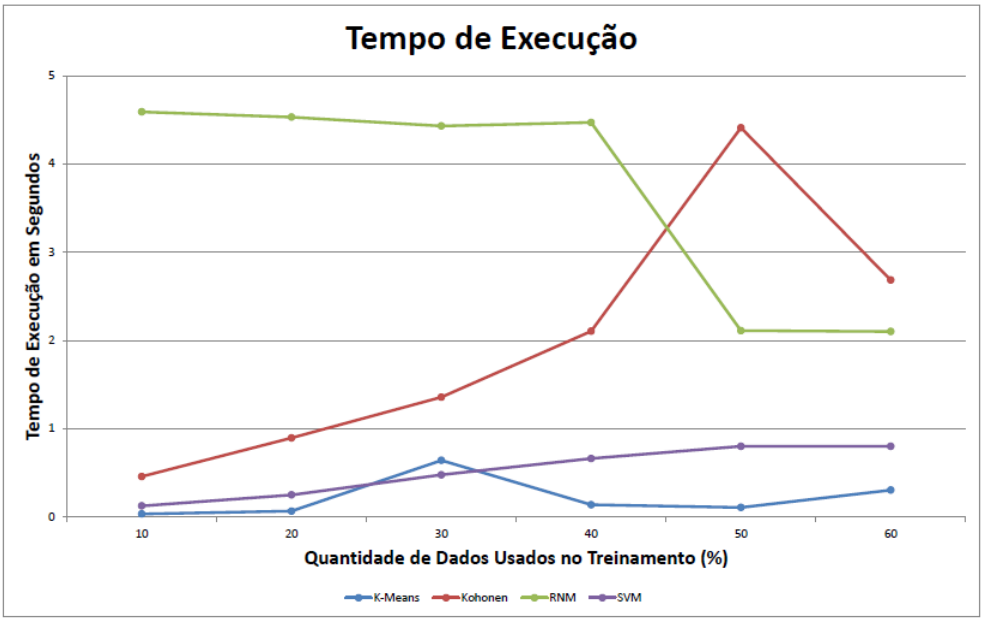

Figura 3. Gráfico dos tempos de treinamento dos algoritmos

Fonte: elaborada pelos autores

A Tabela 1 mostra a taxa de acerto com 10% e 60% de dados utilizados para o treinamento dos algoritmos, sendo este o intervalo utilizado para porcentagem de dados. No caso dos algoritmos não-supervisionados, a taxa de acerto foi determinada, analisando os *clusters* e rotulando-os manualmente. Desta forma, pode-se determinar as taxas de acerto para

as técnicas não-supervisionadas. Na Tabela 1 pode-se visualizar o tamanho do salto da taxa de acerto de cada algoritmo.

Tabela 1. Taxas de acerto (%) dos algortimos com 10% e 60% de dados de treinamento.

Algoritmos	10%		60%	
Redes neurais multicamadas	95,658	± 0,92	97,813	± 0,14
Rede de *Kohonen*	94,722	± 2,7	99,305	± 0,91
Máquina de vetor de suporte	87,511	± 1,92	99,812	± 0,10
Algoritmo *K-means*	81,560	± 3,68	91,978	± 1,15

Fonte: elaborada pelos autores com base em dados da pesquisa de campo

Na Figura 4 pode-se visualizar o resultado da classificação de usuários da base de testes através das RNMs. Essa técnica, utilizando 10% dos dados para treinamento, de acordo com o gráfico da Figura 8, apresentou um desempenho satisfatório, sendo que mesmo com o aumento da porcentagem de dados para treinamento, a taxa de acerto não aumentou de forma significativa, ou seja, a taxa de acerto para este algoritmo foi considerada satisfatória, mantendo-se assim ao longo da execução do experimento.

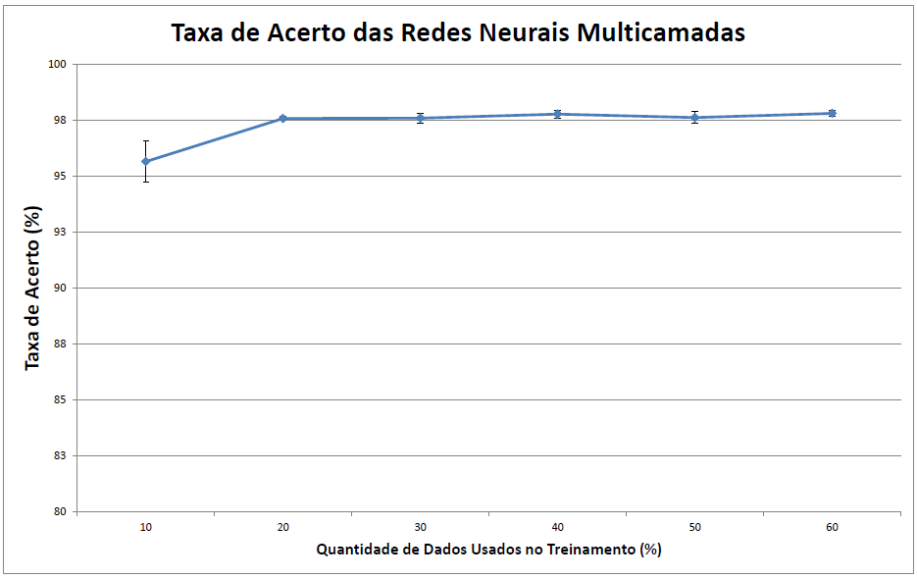

Figura 4. Gráfico da evolução da taxa de acerto das redes neurais multicamadas

Fonte: elaborada pelos autores

A classificação da máquina de vetor de suporte pode ser vista no gráfico da Figura 5. Neste caso, houve um aumento significativo da taxa de acerto à medida que foi aumentando a porcentagem dos dados de treinamento.

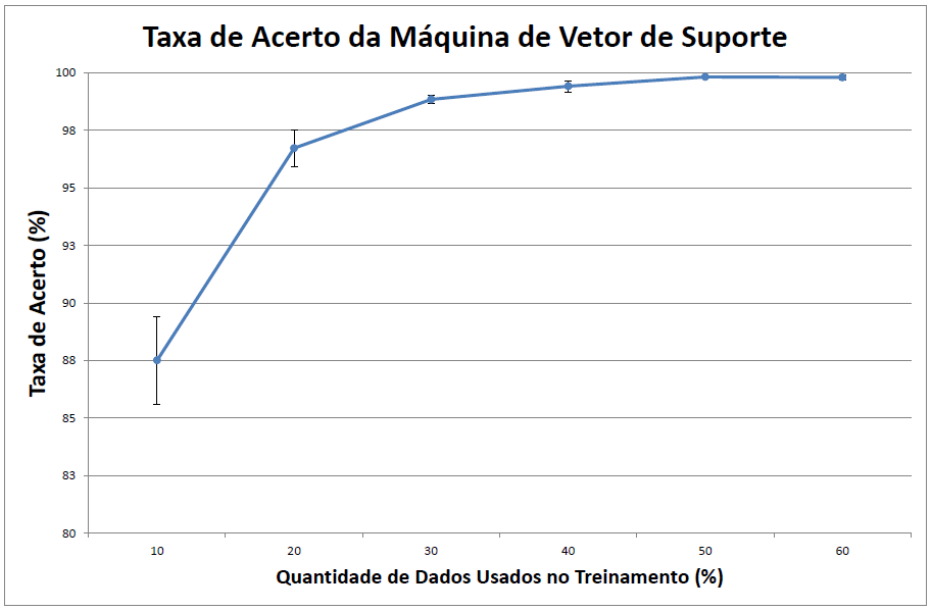

Figura 5. Gráfico da evolução da taxa de acerto da máquina de vetor de suporte

Fonte: elaborada pelos autores

Já a rede de *Kohonen* gerou uma média de 18 à 20 grupos de usuários nas 10 execuções, e cada grupo é uma área de conhecimento distinta, com exceção de dois grupos, onde a rede gerou um grupo com usuários de Direito e Economia e outro grupo com Medicina e Odontologia, devido as sub áreas em comuns desses usuários. O gráfico da Figura 6 mostra a evolução da taxa de acerto da Rede de Kohonen a medida que vai aumentando os dados para treinamento.

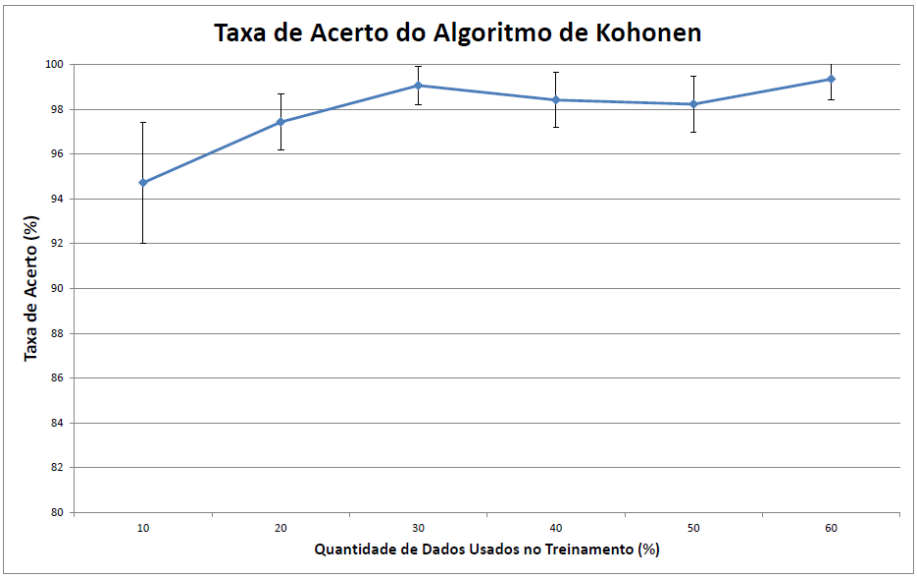

Figura 6. Gráfico da Evolução da Taxa de Acerto da Rede de *Kohonen*.

Fonte: elaborada pelos autores

A Figura 7 mostra o gráfico com a taxa de acerto do algoritmo *K-means*. Assim como a rede de *Kohonen*, o *k-means* também criou de 18 a 20 grupos em dez execuções, sendo dois destes possuindo usuários de duas áreas de conhecimento. Um desses grupos envolveu usuários de História e Direito e o outro usuários de Medicina e Odontologia, como já havia ocorrido para a rede de *Kohonen*. Este algoritmo apresentou uma taxa de acerto não muito satisfatória em relação aos outros algoritmos.

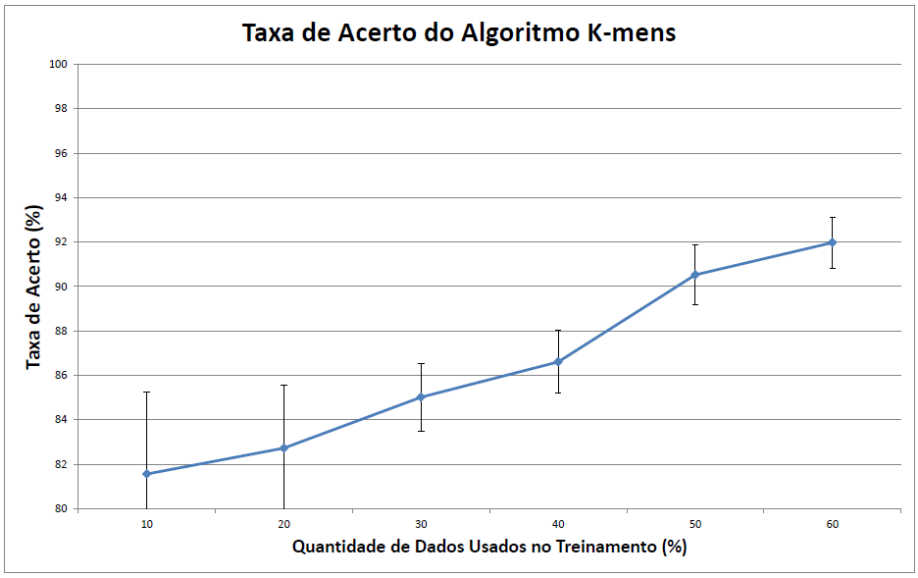

Figura 7. Gráfico da evolução da taxa de acerto do algoritmo *K-means*

Fonte: elaborada pelos autores

6 ROTULAÇÃO DAS CLASSES DE USUÁRIOS

Como dito na seção 5, os usuários são classificados com algoritmos de aprendizado de máquina, que geram grupos de usuários sem rotulação.

Na proposta deste trabalho é necessária a utilização de um algoritmo com aprendizagem não-supervisionada para realizar a tarefa de agrupamento. Dentre os vários algoritmos existentes a técnica escolhida para este experimento foi o *K-means*. Contudo, qualquer outro algoritmo de agrupamento poderia ter sido utilizado.

Em outra etapa da proposta será necessário o uso de um algoritmo com aprendizagem supervisionada. Para tal tarefa foi escolhido o uso de redes neurais artificiais, principalmente por sua característica de detecção de padrões (LUDWIG JÚNIOR; COSTA, 2007).

Frente ao problema de rotulação apresentado anteriormente, nossa proposta consiste em definir um modelo que possibilite a rotulação dos *clusters*.

Inicialmente aplica-se um algoritmo com aprendizagem não-supervisionada com o intuito de formar diversos grupos entre os elementos em questão. Para cada grupo formado é aplicado um segundo algoritmo,

desta vez, com um processo de aprendizagem supervisionada, permitindo a identificação de características relevantes. O esquema da Figura 8 demonstra a proposta.

Para que o segundo algoritmo obtenha um melhor desempenho em relação a valores contínuos, foi realizado um processo de discretização, em que os diferentes possíveis valores se resumem a intervalos ou faixas.

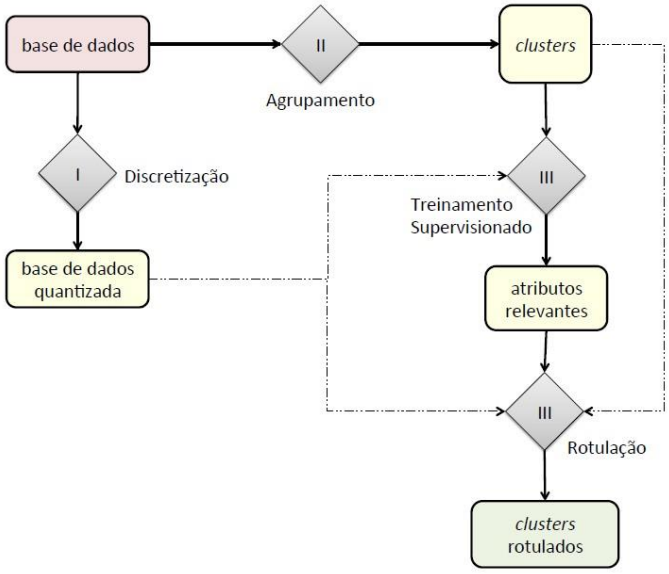

Figura 8. Esquema de rotulação no Scientia.Net

Fonte: elaborada pelos autores

Inicialmente, têm-se como entrada uma base de dados. Esta base conta com diferentes tipos de dados para os quais, dependendo do tipo (discreto/contínuo), pode ser necessária a utilização de um modelo de discretização (I).

A segunda etapa (II) é realizada por um algoritmo não-supervisionado, que realiza a tarefa de agrupamento. Uma vez que os *clusters* tenham sido gerados, aplica-se um algoritmo supervisionado (III) a fim de detectar quais os atributos relevantes para a definição de cada *cluster*. Por fim, a rotulação (IV) é executada em cada *cluster*.

6.1 DISCRETIZAÇÃO

A etapa I consiste em discretizar os dados. Isto é, para os atributos que podem assumir diferentes valores dentre um domínio contínuo são estabelecidos novos valores discretos. Dessa forma, o algoritmo com aprendizagem supervisionada pode identificar com mais facilidade uma possível relação entre os atributos, apresentando melhores resultados em sua classificação.

Na literatura existem vários métodos de discretização. Os dois métodos mais utilizados são discretização em intervalos iguais (*equal width discretization* - EWD) e discretização em frequências iguais (*equal frequency discretization* - EFD).

O modelo de discretização proposto neste trabalho é o EWD e utiliza quatro faixas de valores que são calculadas por três médias. A primeira média (*m*) é a média aritmética simples entre o menor e o maior valor do atributo em questão. A segunda (*mEsq*) e a terceira (*mDir*), ambas aritméticas simples, podem ser calculadas utilizando-se a primeira média (*m*) com o menor e o maior valor, respectivamente. Dessa forma, para cada atributo teremos 4 faixas de valores (Figura 9).

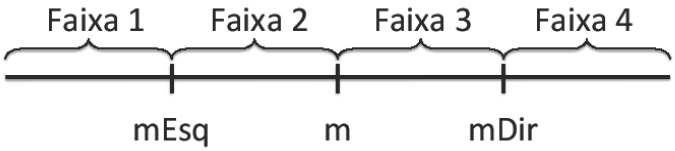

Figura 9. Faixas de valores

Fonte: elaborada pelos autores

- Faixa 1: valor menor ou igual à segunda média (*mEsq*);
- Faixa 2: valor maior que a segunda média (*mEsq*) e menor ou igual à primeira média (*m*);
- Faixa 3: valor maior que a primeira média (m) e menor ou igual à terceira média (*mDir*);
- Faixa 4: valor maior que a terceira média (*mDir*);

Desta forma, permite-se que a RNA trabalhe com faixas de valores tornando possível a detecção de atributos relevantes - trabalhar com os valores reais de atributos contínuos com várias casas decimais iria exigir uma tolerância de erro maior, tornando o processo dispendioso computacionalmente.

O modo como foi utilizado — quatro faixas de valores definidas por três médias — é algo a ser discutido conforme o problema em questão. O modelo de discretização será representado por *MD*. Os valores discretizados são armazenados e utilizados posteriormente durante as etapas III (treinamento), como entrada das RNAs, e IV (rotulação), como os limites dos intervalos -- faixas de valores.

Após a discretização, ocorre a geração de *clusters* (etapa II). Nesta etapa, tem-se como entrada uma base de dados e como saída seus elementos agrupados em *K clusters*.

6.2 TREINAMENTO SUPERVISIONADO

Em cada *cluster* gerado é aplicado um algoritmo supervisionado. A ideia nesta etapa é detectar quais atributos são relevantes para o grupo. Para isso, uma RNA (supervisionada) é criada para cada atributo, que é tratado como um atributo classe (saída). Os demais são tratados como entrada da rede, com o intuito de descobrir quais atributos podem classificar o grupo corretamente.

Para cada atributo dos elementos pertencentes a um dado *cluster* é criada uma RNA que tem como entrada os demais atributos e que apresenta como saída o valor estimado para o atributo em questão. Todas as RNAs de um mesmo *cluster* possuem os mesmos elementos, variando apenas a forma como são utilizados na rede, ora entrada, ora saída. Os valores de entrada não são os valores exatos, mas sim os valores discretizados, já calculados durante a etapa I. O valor de saída da rede corresponde a uma faixa de valores dentre as especificadas, também conforme o modelo de discretização.

Considerando um *cluster* qualquer, as RNAs têm seus elementos divididos em duas partes (distintas para cada rede): treinamento e teste. A parte de treinamento é utilizada pela rede para o aprendizado em si, isto é, para o processo de aprendizagem de um padrão. A parte de testes é utilizada para medir a eficiência da rede em relação ao aprendizado obtido durante o treinamento. Após o aprendizado, durante a fase de testes, caso o valor de saída da rede seja igual ao valor correspondente à faixa do atributo para seu valor, houve acerto. Caso contrário, houve um erro.

Dessa forma, cada RNA é criada para representar e estimar a importância de cada atributo. De uma forma mais ampla, cada *cluster* tem, então, uma taxa de acerto para cada RNA, ou seja, uma taxa de acerto para cada atributo avaliado. Dessa forma, podemos saber qual atributo é relevante em relação aos demais para um determinado *cluster*: aquele que obteve maior taxa de acerto na fase de teste. Para uma maior precisão em relação ao atributo, existe uma média de iterações nesta etapa. Cada iteração corresponde a uma RNA para cada atributo.

6.3 PROCESSO DE ROTULAÇÃO

A última etapa (IV) consiste em nomear os *clusters* em função de seus atributos. Após a etapa de treinamento, cada *cluster* tem a média de acerto de seus atributos. A maior média entre as taxas de acerto indica o(s) atributo(s) relevante(s).

Outro parâmetro, variação *V*, ajuda a selecionar os demais atributos que possuem uma taxa de acerto com variação, de no máximo, *V* (dada em porcentagem) em relação ao principal atributo. Dessa forma, temos um conjunto de atributos considerados relevantes para a definição do *cluster*.

Depois de definido o conjunto de atributos relevantes, verifica-se qual das faixas de valores (definidas na etapa de discretização) domina o grupo. Isto é, detectar qual faixa de valor de cada atributo apresenta maior frequência em um *cluster* qualquer. Dessa forma, temos a precisão da importância de cada atributo (taxa de acerto), bem como seus prováveis valores (faixas). Essas duas informações são suficientes para a rotulação. A Figura 11 a seguir demonstra, em um algoritmo escrito em linguagem natural, o funcionamento da proposta:

```
Entrada: Base de Dados
1:  Carregar a base de dados;
2:  Discretizar cada atributo contínuo;
3:  Realizar agrupamento (algoritmo não-supervisionado);
4:  para cada cluster do
5:  para cada iteração i=1 para i faça
6:  Definir conjuntos de treinamento e teste;
7:  para cada atributo faça
8:  Realizar treinamento (algoritmo supervisionado);
9:  Calcular a taxa de acerto;
10: fim para
11: fim para
12: Calcular as médias das taxas de acerto;
13: fim para
14: Rotular;
```

Figura 10. Algoritmo de rotulação proposto

Fonte: elaborada pelos autores

6.4 RESULTADO DA ROTULAÇÃO DAS CLASSES

A rotulação automática é feita em função dos *clusters* gerados pelo *K-means* e que, como mostra a Tabela 3, se assemelham (em sua maioria) com os resultados obtidos em Lima e Machado (2012). Ainda assim, os rótulos aqui apresentados também são específicos e podem ser diferentes a cada execução, conforme o agrupamento realizado.

A coluna de relevância (*Rel.*) representa a média das taxas de acerto do algoritmo de aprendizado para o atributo em questão. Em outras palavras, representa a relevância de tal atributo para o *cluster*.

Após os resultados apresentados pelo programa foi feita a análise. Conforme visto na Tabela 2, para cada *cluster* foi sugerido um conjunto de atributos bem como seus respectivos valores. Por possuírem valores discretos, e não faixas de valores contínuos, os grupos são mais bem definidos pelo algoritmo não-supervisionado.

Isto se deve à etapa de agrupamento, uma vez que a taxa de acerto foi 100% em ambos os *clusters*. Em alguns casos um agrupamento pode separar elementos semelhantes em grupos distintos, bem como agrupar elementos diferentes em um mesmo *cluster*. Entretanto, de uma forma geral, o agrupamento realizado foi satisfatório, pois a maioria dos *clusters* ficou bem definida.

O *cluster* 2 foi o único que apresentou um dos atributos de uma subárea. Estes atributos contêm uma informação mais específica e, por

isso, não tendem a ser um atributo com alta relevância. Entretanto, ele foi sugerido no *cluster* 2 e, como pudemos observar, foi uma classificação correta uma vez que o grupo contém apenas doze elementos, sendo este um grupo bem específico.

Tabela 2. Resultados da rotulação com a base de dados do Scientia.Net

Cluster	Elemento	Resultados			Análise	
		Nome	Rel. (%)	Valor	Erro	Acerto
1	100	Pós Doutorado	100	27	0	100
		Graduação	100	20	0	100
		Mestrado	100	27	0	100
		Doutorado	100	27	0	100
2	12	Pós Doutorado	100	17	0	100
		Mestrado	100	17	0	100
		Doutorado Sub	100	17	0	100
		Doutorado	100	13	0	100
...	...	...	...	...	...	...
4	160	Pós Doutorado	100	18	60	62.5
		Mestrado	100	18	60	62.5
		Doutorado	100	18	60	62.5
...	...	...	...	...	...	...
7	103	Mestrado	98.0952	14	3	97.0874
		Doutorado	97.619	14	3	97.0874
		Pós Doutorado	97.8571	14	3	97.0874
...	...	...	...	...	...	...
15	172	Pós Doutorado	100	9	80	53.4884
		Graduação	100	5	80	53.4884
		Mestrado	100	9	80	53.4884
		Doutorado	100	9	80	53.4884
...	...	...	...	...	...	...

Fonte: elaborada pelos autores

Conforme a análise dos resultados, têm-se as piores taxas de acerto em 53,4884% e 62,5%. Não coincidentemente, estas taxas se referem aos maiores *clusters* com 172 e 160 elementos, respectivamente. O rótulo não está errado, de fato, pois representa a maioria do grupo. Entretanto, a baixa taxa de acerto se deve à má definição dos grupos que contêm 72% e 60% elementos a mais do que foi apresentado no trabalho anterior.

Observando os demais grupos, teve-se que a maioria (55%) dos *clusters* apresentou uma taxa de acerto de 100%. De uma forma geral, o resultado foi bastante satisfatório, atingindo uma média de 93,357% de acerto dos elementos conforme os rótulos sugeridos.

7 CONCLUSÃO

Considerando as duas formas de aprendizagem discutidas neste artigo, a não-supervisionada torna-se mais adequada para o classificação dos usuários dentro do *Scientia.Net*, pois para a classificação utilizando algoritmos supervisionados é necessário saber de antemão as classes em que os

usuários serão classificados, para o processo de treinamento. Os algoritmos não-supervisionados geram grupos de acordo com a similaridade dos usuários, não necessitando de uma atribuição de classes antecipadamente (SOUSA; ESMIN, 2011).

Neste trabalho, também foi apresentado um modelo para a rotulagem das classes em que os usuários são classificados. Um algoritmo não supervisionado é utilizado para a definição dos *clusters* e, mais tarde, um algoritmo de aprendizagem supervisionada é aplicado para cada atributo de cada *cluster*. A avaliação do algoritmo não-supervisionado pode identificar quais atributos são relevantes para o problema. Vale ressaltar que o passo de discretização realizado em atributos contínuos é fundamental para este modelo.

Os resultados obtidos são satisfatórios, de modo que a maioria dos *clusters* avaliados em ambas as bases de dados foram marcados com taxa de acerto elevada, com uma média de mais de 90%. É notável que o processo de marcação é feito em um *cluster* e, portanto, depende essencialmente dos seus elementos. Assim, um grupo mal definido terá uma rotulagem imprecisa, já que o algoritmo não-supervisionado recebe uma contribuição importante do algoritmo supervisionado para o resultado da rotulação.

Considerando a diversidade de técnicas existentes relativas aos algoritmos supervisionados, os resultados indicam um melhor desempenho com a rede de Kohonen. Algoritmos não supervisionados e modelos de discretização possibilitam ainda uma melhoria significativa.

O processo de rotulação pode ainda ser utilizado para melhorar a classificação de artigos, pois com a rotulação dos grupos dos usuários pode-se definir o nome das classes em que os artigos científicos podem ser classificados.

Os aspectos de rotulação podem ser melhorados. Para isso serão testados futuramente outros algoritmos mencionados neste artigo (rede de *Kohonen* e máquina de vetor de suporte). Além disso, pode-se analisar os parâmentos utilizados nas redes neurais, tais como: taxa de aprendizado, quantidade de neurônios, pesos iniciais, entre outros. Com isso, será possível propor melhorias no algoritmo de rotulação ora proposto.

Pretende-se, além disso, realizar este trabalho utilizando outra base de dados para propor uma melhoria da rotulação, avaliando seu desempenho no uso de cada uma das técnicas e propondo nova abordagem, se conveniente.

REFERÊNCIAS

BRAGA, P. B.; CARVALHO, F. L.; LUDEMIR, T. B. *Redes neurais artificiais:* teoria e aplicações. 2. ed. Rio de Janeiro, 2007.

CORTES, C; VAPNIX, V. Support-vector networks, *Machine Learning*, n. 20, p. 273–297, 1995. http://dx.doi.org/10.1007/BF00994018

HAYKIN, S. *Neural networks:* a comprehensive foundation, 2. ed., Porto Alegre: Bookman, 2001.

LIMA, B. V. L.; MACHADO, V. P. Machine learning algorithms applied in automatic classification of social network users. In: *4th International Conference on Computational Aspects of Social Networks - Cason*, São Carlos-SP, 2012.

LIMA, B. V. L.; MACHADO, V. P.; ARAÚJO, S. W. I. Classificação dos usuários da rede social *Scientia.Net* através de redes neurais artificiais. In: *Escola Reginal de Computação Ceará, Maranhão e Piauí- ERCEMAPI*, Tereina-PI, 2012a.

LIMA, B. V. L.; MACHADO, V. P.; ARAÚJO, S. W. I. Classificação automática de usuários de uma rede social utilizando algoritmos não-supervisionados. In: Brazilian Workshop on Social Network Analysis and Mining - BraSNAM, Curitiba. Anais do Brasnam, 2012b.

LUDWIG JUNIOR, O.; COSTA, E. M. M. *Redes neurais:* fundamentos e aplicações com programas em C. Editora Ciência Moderna, 2007.

MACHADO, V. P.; LIMA, B. V. L.; ARNALDO, H. A.; ARAÚJO, S. W. I. In: Congresso Tecnológico TI e Telecon – INFOBRASIL, 4., Fortaleza. Anais do INFOBRASIL, 2011.

MACQUEEN, J. Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, 1997.

MITCHELL, T. M. *Machine learning*. 1997.

PENNACCHIOTTI, M.; POPESCU, A. M. A machine learning approach to Twitter user classification. In: Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media, 2011.

RUSSELL, S.; NORVING, P. *Inteligência artificial*. Editora Campus, 2004.

SOUSA, G. H. A.; ESMIN, A. A. A. Algoritmo de enxame de partículas híbrido aplicado a clusterização de dados. In: ENIA - VIII Encontro Nacional de Inteligência Artificial, Natal-RN, 2011.

SUTTON, R. S.; BARTO, A. G. *Reinforcement learning:* an introduction. *Cambridge*, 1998.

VALIATI, H.; SILVA, A.; GUIMARÃES, S, JÚNIOR, W. M. Detecção de conteúdo relevante e usuários influentes no Twitter. In: Brazilian Workshop on Social Network Analysis and Mining - BraSNAM, Curitiba. *Anais do Brasnam*, 2012.

WITTEN, I. H.; FRANK, E. *Data mining:* practical machine learning tools and techniques. 2. ed., Editora Elsevier, 2005.