Guilherme Augusto Bauer

GERAÇÃO DE CONHECIMENTO ATRAVÉS DE DADOS DA PLATAFORMA LATTES COM O USO DE TÉCNICAS DE MINERAÇÃO DE DADOS

Trabalho de conclusão apresentado ao curso de Ciência da Computação da Universidade de Santa Cruz do Sul para obtenção do título de Bacharel em Ciência da Computação.

Orientador: Prof. Me. Eduardo Kroth

Santa Cruz do Sul
2016

RESUMO

Existe atualmente uma grande necessidade de as organizações terem informação correta e de forma rápida e nas universidades existe uma dificuldade em se conseguir informações referentes a produção bibliográfica, científica e tecnológica de seus professores. Esses dados podem ser encontrados na Plataforma Lattes que possui uma grande base de dados de currículos de pesquisadores e professores. Para realizar a extração de conhecimento de uma grande base de dados podem ser utilizadas técnicas de KDD (Descoberta de Conhecimento em Bases de Dados) e principalmente de Mineração de Dados. Este trabalho apresenta uma fundamentação bibliográfica relacionada a KDD, Mineração de Dados, mais especificamente Clusterização e Regras de Associação, e a sugestão de uma solução para coletar dados dos currículos Lattes dos professores da UNISC (Universidade de Santa Cruz do Sul) e aplicar algoritmos de mineração de dados sobre os dados coletados a fim de gerar conhecimento que poderá ser utilizado em tomadas de decisão por gestores da universidade.

Palavras Chave: Associação, Clusterização, KDD, Lattes, Mineração

ABSTRACT

There is currently a big necessity for organizations to have correct and quickly information and universities have a difficulty in getting information about the bibliographic, scientific and, technology production of their teachers. This data can be found in the Lattes Platform which has a large base curriculum data of researchers and teachers. To perform the extraction of knowledge from a large database can be used KDD techniques (Knowledge Discovery in Databases) and especially of Data Mining. This paper presents a bibliographic basis related to KDD, Data Mining, specifically Clustering and Association Rules, and the suggestion of a solution to collect data from Lattes curriculum of UNISC (University of Santa Cruz do Sul) teachers and apply data mining algorithms on data collected in order to generate knowledge that can be used in decision-making by university managers.

Keywords: Association, Clustering, KDD, Lattes, Mining,

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Com a grande competitividade existente atualmente no mercado, possuir informações corretas e de forma rápida para realizar tomadas de decisões é de suma importância para as organizações. Entre essas organizações estão as universidades que entre outros dados desejam ter conhecimento da produção científica, bibliográfica e tecnológica de seu corpo docente. E com o grande crescimento no volume de dados que circulam pelas organizações e o consequente aumento no tamanho de suas bases de dados é necessário o uso de tecnologias de informação para conseguir coletar, gerir, analisar, armazenar e utilizar da melhor forma possível essas informações; é nesse contexto que surgem ferramentas para auxílio nesses diversos processos.

Uma dessas ferramentas que podem ser utilizadas é o KDD (Knowledge Discovery in Databases), processo utilizado para descobrir conhecimento em grandes bases de dados. Entre as etapas do processo de KDD a principal delas é o Data Mining ou Mineração de Dados que consiste no uso de algoritmos específicos para localizar padrões de comportamento nos dados disponíveis. Dentre as técnicas da Mineração de Dados foram selecionadas para uso nesse trabalho a Descoberta de Associações e a Clusterização. Na descoberta de associações são descobertos conjuntos de dados que possuem uma correlação com outros conjuntos de dados distintos nos registros de uma base de dados. E na clusterização pode ser realizado um agrupamento dos dados de acordo com similaridades entre eles e das diferenças para os dados dos demais agrupamentos.

Muitos dos dados referentes a produção científica, bibliográfica e tecnológica do corpo docente de uma universidade podem ser encontrados nos currículos dos professores cadastrados na Plataforma Lattes, que é uma ferramenta desenvolvida pelo CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) e que possui um módulo direcionado ao cadastro de currículos de pesquisadores brasileiros com dados referentes a sua produção científica, bibliográfica e tecnológica.

Nesse trabalho é realizado um estudo das técnicas de KDD e Mineração de Dados e essas técnicas foram posteriormente usadas no desenvolvimento de uma ferramenta para extração de conhecimento dos dados dos currículos disponíveis na Plataforma Lattes.

1.1 Estrutura do Trabalho

No capítulo 2, é apresentada uma revisão bibliográfica dos principais temas e conceitos teóricos necessários para a elaboração deste trabalho.

No capítulo 3, é apresentada a Plataforma Lattes, com enfoque na estruturação dos dados nos currículos Lattes.

O capítulo 4 trata sobre o sistema de classificação de produção científica Qualis.

O capítulo 5 possui a metodologia utilizada no desenvolvimento do trabalho.

O capítulo 6 contém dados e informações a respeito de trabalhos relacionados ao assunto abordado nesse trabalho e que, em alguns casos, foram utilizados de fonte para o mesmo.

No capítulo 7 encontra-se a explicação referente a solução desenvolvida para a extração dos dados dos currículos Lattes e posterior aplicação de algoritmos de Mineração de Dados.

No capítulo 8 estão as considerações finais sobre o problema, os tópicos estudados, a solução desenvolvida e sugestões de trabalhos futuros.

1.2 Objetivos

1.2.1 Objetivo principal

Aplicar algoritmos de Data Mining a fim de identificar conhecimento útil a partir de dados de currículos Lattes de professores da Universidade de Santa Cruz do Sul (UNISC).

1.2.2 Objetivos específicos

- Desenvolver uma ferramenta para selecionar e extrair dados específicos dos currículos Lattes;
- Verificar e selecionar técnicas de Data Mining a serem aplicadas sobre esses dados;
- Aplicar técnicas e algoritmos de Data Mining sobre uma base de dados;
- Analisar e estruturar o conhecimento adquirido para que ele possa ser interpretado e utilizado por usuários finais do sistema.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 KDD (Descoberta de Conhecimento em Bases de Dados)

Com a grande competitividade existente atualmente no mercado, possuir informações corretas e de forma rápida para realizar tomadas de decisões é de suma importância para as organizações. E com o grande crescimento no volume de dados que circulam pelas organizações e o consequente aumento no tamanho de suas bases de dados é necessário o uso de tecnologias de informação para conseguir coletar, gerir, analisar, armazenar e utilizar da melhor forma possível essas informações; é nesse contexto que surgem ferramentas para auxílio nesses diversos processos.

Segundo Goldschmidt, Passos (2005, p. 1), a análise de grandes quantidades de dados pelo homem é inviável sem o uso de ferramentas computacionais apropriadas.

Descoberta de Conhecimento em Bases de Dados, da sigla em inglês KDD(Knowledge Discovery in Databases) é o termo que contempla todo o processo de uso dessas ferramentas para descobrir conhecimento em grandes bases de dados.

De acordo com Fayyad (1996, apud Macedo e Matos, 2010) “KDD é um processo, de várias etapas, não trivial, interativo e iterativo, para identificação de padrões compreensíveis, válidos, novos e potencialmente úteis a partir de grandes conjuntos de dados”. Para entender melhor o conceito de KDD é necessário entender as diferenças entre dado, informação e conhecimento conforme ilustrado na figura 1.

Figura 1 - Hierarquia entre dado, informação e conhecimento

Fonte: Goldschmidt, Passos (2005), p. 2

Os dados, na base da pirâmide, podem ser interpretados como itens elementares, captados e armazenados por recursos da Tecnologia da informação (GOLDSCHMIDT, PASSOS, 2005, p. 2). Segundo Carvalho (2000), o dado é tido como somente um ponto no espaço ou no tempo, que não guarda referência a qualquer outro espaço ou tempo. O seu significado depende da sua associação com outras coisas e a existência de um contexto.

As informações representam os dados processados, com significados e contextos bem definidos (GOLDSCHMIDT, PASSOS, 2005, p. 2). De acordo com Setzer(2015) "Informação é uma abstração informal (isto é, não pode ser formalizada através de uma teoria lógica ou matemática), que está na mente de alguém, representando algo significativo para essa pessoa". Se a representação da informação for feita por meio de dados pode ser armazenada em um computador. Mas, atenção, o que é armazenado na máquina não é a informação, mas a sua representação em forma de dados (SETZER, 2015).

No topo da Pirâmide encontra-se o conhecimento que é o padrão ou conjunto de padrões, cuja formulação pode envolver e relacionar dados e informações (GOLDSCHMIDT, PASSOS, 2005, p. 3). Segundo Setzer(2015) "o conhecimento não pode ser descrito; o que se descreve é a informação ou o dado. Também não depende apenas de uma interpretação pessoal, como a informação, pois requer uma vivência do objeto do conhecimento. Assim, o conhecimento está no âmbito puramente subjetivo do homem ou do animal. Parte da diferença entre estes reside no fato de um ser humano poder estar consciente de seu próprio conhecimento, sendo capaz de descrevê-lo parcial e conceitualmente em termos de informação, por exemplo".

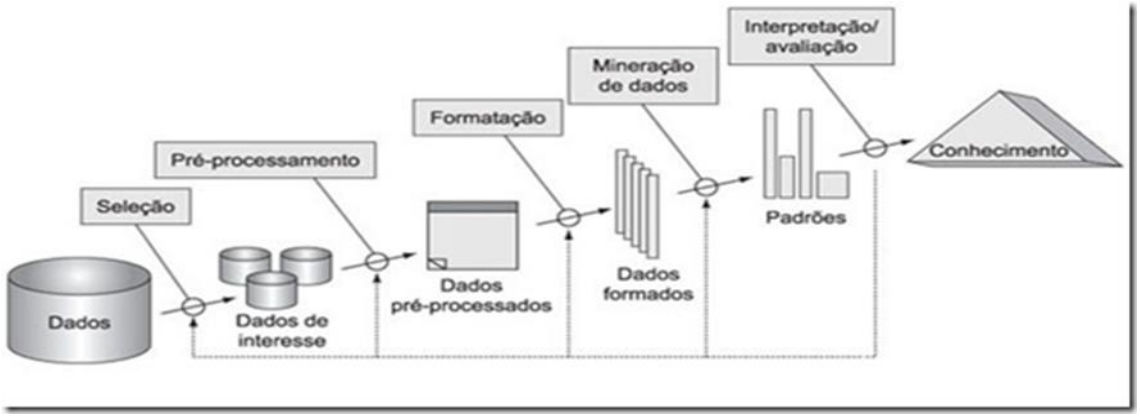
Para Barreto (1996, apud Carvalho, 2000), o conhecimento é "toda alteração provocada no estado cognitivo do indivíduo, isto e, no seu estoque mental de saber acumulado, proveniente de uma interação positiva com uma estrutura de informação".

De acordo com Elmasri, Navathe(2005, p. 625), o processo de KDD é composto de seis fases: seleção de dados, limpeza, enriquecimento, transformação ou codificação, Data Mining e a construção de relatórios de apresentação usando os padrões identificados.

Para Fayyad(1996, apud Lobo, Ramos, 2003), o processo de KDD é dividido em nove passos básicos: compreender o domínio da aplicação, criar um conjunto de dados para análise, realizar a limpeza e pré processamento dos dados, transformar e reduzir os dados, escolher o objetivo da mineração de dados, escolher o(s) algoritmo(s) de mineração, realizar a mineração

dos dados, interpretar os padrões resultantes e se necessário refazer os passos anteriores, consolidar o conhecimento adquirido. A figura 2 ilustra esses passos do processo de KDD.

Figura 2 - Passos do processo de KDD



Fonte: Lobo, Ramos

2.1.1 Compreensão do domínio da aplicação

Nesse passo deve-se identificar e compreender o domínio da aplicação definindo os objetivos a serem alcançados com o processo de KDD. O ideal nessa fase é que exista uma ou mais pessoas que sejam especialistas no domínio da aplicação.

2.1.2 Seleção dos dados

Devem ser analisados os dados e atributos de objetos existentes para definir quais são realmente importantes e serão utilizados no processo de KDD. Nesse passo é muito importante a qualidade dos dados salvos e também uma correta avaliação por parte dos analistas humanos pois o processo é muito dependente do conhecimento deles sobre a base de dados e de suas escolhas. Se um atributo importante for descartado nessa fase o resultado do processo pode não ser satisfatório.

2.1.3 Limpeza e Pré-Processamento dos dados

Na fase de limpeza e pré processamento são corrigidas inconsistências e ruídos nos dados pois segundo Navega(2002, apud Costa et al.) “as bases de dados são dinâmicas, incompletas, redundantes, ruidosas e esparsas, necessitando de um pré-processamento para limpá-las”.

Segundo Goldschmidt, Passos(2005, p.12), no processo de limpeza dos dados devem ser executadas as seguintes funções:

2.1.3.1 Correção de dados ausentes

Em alguns casos podem existir atributos cujos valores não estão presentes na base de dados. Para esses casos existem as seguintes opções de solução:

- Exclusão dos registros: consiste em desconsiderar os registros com valores ausentes. Não é uma opção muito recomendada pois se existirem muitos registros com valores ausentes na base de dados restarão poucos atributos com informações completas para a execução da mineração.
- Preenchimento manual dos valores: Essa opção não é recomendada devido ao grande custo de tempo e recursos se em um grande volume de dados existirem muitos registros com dados ausentes e pela dificuldade de localização dos valores corretos a serem utilizados.
- Preenchimento com valor padrão: Consiste no preenchimento dos dados ausentes com um valor padrão podendo ser um valor de outro registro do atributo ou uma constante como “valor desconhecido” ou “valor nulo”. Não é uma técnica muito recomendada pois em alguns casos os algoritmos de mineração podem considerar essas constantes como valores importantes atrapalhando o processo de aquisição de conhecimento.
- Preenchimento com dados estatísticos: Nessa técnica são utilizados dados estatísticos, levantados a partir dos demais dados existentes nos atributos, para o preenchimento dos dados ausentes como por exemplo média no caso de atributos numéricos.
- Preenchimentos com métodos de mineração de dados: Mesmo ainda na fase de pré-processamento podem ser utilizados métodos de mineração de dados para sugerir valores prováveis para os dados ausentes.

2.1.3.2 Correção de dados inconsistentes

Podem ocorrer casos de dados que foram inseridos de forma errada pelos usuários e também casos de dados redundantes, que registram a mesma informação mas escrita de formas diferentes. No caso dos dados informados incorretamente e que ficam fora do padrão dos demais registros de um determinado atributo eles podem ser localizados e desconsiderados através de uma verificação feita pelas pessoas envolvidas no processo com auxílio de computadores. E para os dados redundantes pode ser feita uma correlação desses dados com outros atributos para verificar qual será o dado mantido para a execução da mineração.

2.1.4 Transformação e Redução de dados

Muitas vezes é necessário realizar transformações e reduções nos dados para possibilitar a execução dos algoritmos de mineração.

Transformação dos dados: Determinados algoritmos de mineração utilizam apenas valores numéricos ou categóricos e nesses casos é necessário realizar transformações de valores categóricos em numéricos ou vice-versa.

Segundo Camilo, Silva(2009), algumas das técnicas que podem ser usadas nesse passo são: suavização(remove valores errados dos dados), agrupamento(agrupa valores em faixas sumarizadas), generalização(converte valores muito específicos para valores genéricos), normalização(coloca as variáveis em uma mesma escala), e criação de novos atributos a partir de atributos já existentes.

Redução de dados: Em alguns casos um volume muito grande de dados acaba dificultando os processos de análise e mineração dos dados. Nesses casos podem ser utilizadas técnicas de redução de dados para diminuir o volume de dados a ser processado fazendo com que os algoritmos de mineração tenham uma execução mais eficiente melhorando os resultados conseguidos. De acordo com Camilo, Silva(2009), “as estratégias adotadas nesta etapa são: criação de estruturas otimizadas para os dados (cubos de dados), seleção de um subconjunto dos atributos, redução da dimensionalidade e discretização”.

2.1.5 Mineração de dados

É a principal etapa do processo de KDD. Nessa etapa ocorre a real busca por conhecimento útil com a aplicação de algoritmos sobre os dados para a extração de padrões de comportamento. É uma área multidisciplinar possuindo três principais áreas de destaque: Aprendizado de máquina, Estatística e Banco de Dados.

Segundo Lobo, Ramos(2003), os dois objetivos principais do Data Mining são:

Previsão: utiliza variáveis e campos da base de dados para prever valores desconhecidos e futuros de outras variáveis relevantes.

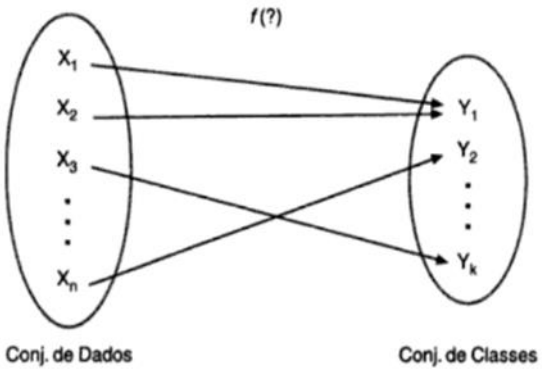
Descrição: utiliza os dados para identificar padrões que possam ser interpretados pelos usuários.

2.1.5.1 Tarefas de Data Mining

Existem diversas tarefas de Data Mining que podem ser executadas de acordo com o resultado que se deseja obter do processamento dos dados. Estas tarefas são detalhadas abaixo.

Classificação: essa tarefa tem como objetivo encontrar uma função que mapeie os registros de uma base de dados em um conjunto de rótulos pré-definidos chamados de classe. Depois de descoberta a função pode ser aplicada em novos registros para definir em que classe esses registros se enquadram. Segundo Goldschmidt, Passos(2005, p. 13), "redes neurais, algoritmos genéticos, lógica indutiva são exemplos de tecnologias que podem ser aplicadas na tarefa de classificação".

Figura 3 - Classificação de dados em suas respectivas classes



Fonte: Goldschmidt, Passos (2005), p. 13

Descoberta de Associação: tem como objetivo encontrar itens que ocorrem de forma simultânea e frequente em transações de uma mesma base de dados determinando padrões entre esses itens. Essa tarefa é muito utilizada em verificações de padrões de compra de determinados tipos de clientes.

> Tipicamente, regras de associação representam padrões existentes em transações armazenadas. Por exemplo, a partir de uma base de dados, na qual registram-se os itens adquiridos por clientes, uma estratégia de mineração, com o uso de regras de associação, poderia gerar a seguinte regra: {cinto, bolsa} → {sapato}, a qual indica que o cliente que compra cinto e bolsa, com um determinado grau de certeza, compra também sapato. Este grau de certeza de uma regra é definido por dois índices: o fator de suporte e o fator de confiança. (Carvalho, Vasconcelos, 2004).

Como a tarefa de descoberta de associação faz parte das tarefas de Data Mining que serão utilizadas neste trabalho ela é descrita de forma mais detalhada no item 3.3.

Regressão: também conhecida como Estimação, tem como objetivo estimar o valor de uma determinada variável com base nos valores de outras variáveis da mesma base de dados. De acordo com Goldschmidt, Passos(2005, p. 13), a tarefa de regressão é similar a tarefa de classificação mas restrita a atributos numéricos e compreende a busca de uma função que mapeie os registros de uma base de dados em valores reais.

Alguns exemplos de regressão: estimativa de um paciente sobreviver dado o resultado de um conjunto de exames, prever o risco de determinado investimento, prever o PIB de um país.

Sumarização: a sumarização, também denominada descrição de conceitos, consiste em definir um conjunto de características que seja capaz de identificar um conjunto de objetos. Segundo Goldschmidt, Passos(2005, p. 73) “consiste em identificar e apresentar, de forma concisa e compreensível, as principais características dos dados contidos em um conjunto de dados”. Um exemplo da tarefa de sumarização é identificar as características dos assinantes de uma determinada revista que residam na região sudeste do Brasil.

Clusterização: a tarefa de clusterização, também chamada de agrupamento ou análise de cluster tem o objetivo identificar e agrupar registros similares da base de dados formando clusters(grupos) de registros similares entre si e diferentes dos registros de outros clusters.

> Essa tarefa se assemelha com a tarefa de classificação. A diferença é que na classificação as classes são definidas de forma prévia, enquanto que no agrupamento as classes são definidas durante a tarefa de acordo com o estabelecimento do conjunto de atributos que devem direcionar essa categorização. Os grupos são formados de acordo com a similaridade desses atributos direcionadores. (Bueno, Viana, 2012).

Como a tarefa de clusterização faz parte das tarefas de Data Mining que serão utilizadas neste trabalho ela é descrita de forma mais detalhada no item 3.2.

Detecção de desvios: tem o objetivo de detectar registros que possuam características fora do padrão dos demais registros do mesmo tipo. Um exemplo é a detecção de fraudes em dados de movimentações com cartões de crédito.

Modelo de dependência: descreve dependências significativas entre os atributos. Esses modelos existem em dois níveis: estruturado, que especifica quais variáveis são localmente dependentes; e quantitativo, que especifica o grau de dependência usando alguma escala numérica.

Descoberta de sequencias: é utilizada para localizar padrões de dados em uma sequencia de transações temporais. Segundo Macedo, Matos(2010), “é uma extensão da tarefa de descoberta de associações em que se busca itens frequentes considerando-se várias transações ocorridas ao longo de um período”. Um exemplo de descoberta de sequencias é a descoberta, em uma base de dados de compras, que 30% dos compradores de notebooks voltam em até um mês para comprar um mouse.

2.1.6 Interpretação e avaliação dos padrões

Nessa fase os padrões resultantes da mineração de dados devem ser interpretados e avaliados por analistas que tem grande conhecimento a cerca do domínio da aplicação para que se definam quais padrões são importantes e úteis e se interprete e demonstre esses padrões de forma que eles sejam compreensíveis aos usuários finais do sistema. Segundo Nicolaio, Pelinski(2006, apud Macedo, Matos, 2010), nessa etapa são utilizados os seguintes procedimentos:

Avaliação: etapa onde se realiza a avaliação do conhecimento extraído com as técnicas de mineração. Esse conhecimento é avaliado pela sua precisão, compreensibilidade e utilidade.

Interpretação e Explanação: nessa etapa deve-se interpretar o conhecimento adquirido e torna-lo claro para o usuário modificando-o, comparando-o com conhecimentos adquiridos anteriormente e documentando-o.

Filtragem: deve-se filtrar o conhecimento extraído por meio de alguma técnica a ser definida para que se mantenha apenas o conhecimento útil para a fase de tomada de decisões.

Interpretação: fase onde é realizada a interpretação final dos padrões de conhecimento coletados e a verificação da necessidade de repetir os passos de processo de mineração para tentar adquirir conhecimento mais útil e de forma mais clara para o usuário ou a finalização do processo para que o conhecimento possa ser utilizado como apoio na tomada de decisão.

2.1.7 Utilização do conhecimento

Nessa fase, o conhecimento adquirido na mineração de dados é consolidado podendo ser incorporado a sistemas utilizados pelos usuários finais ou apenas documentado e comunicado a pessoas interessadas pelos resultados. Nesse passo também devem ser

resolvidos possíveis diferenças e conflitos entre o conhecimento já existente e o adquirido no processo de mineração.

2.2 Clusterização

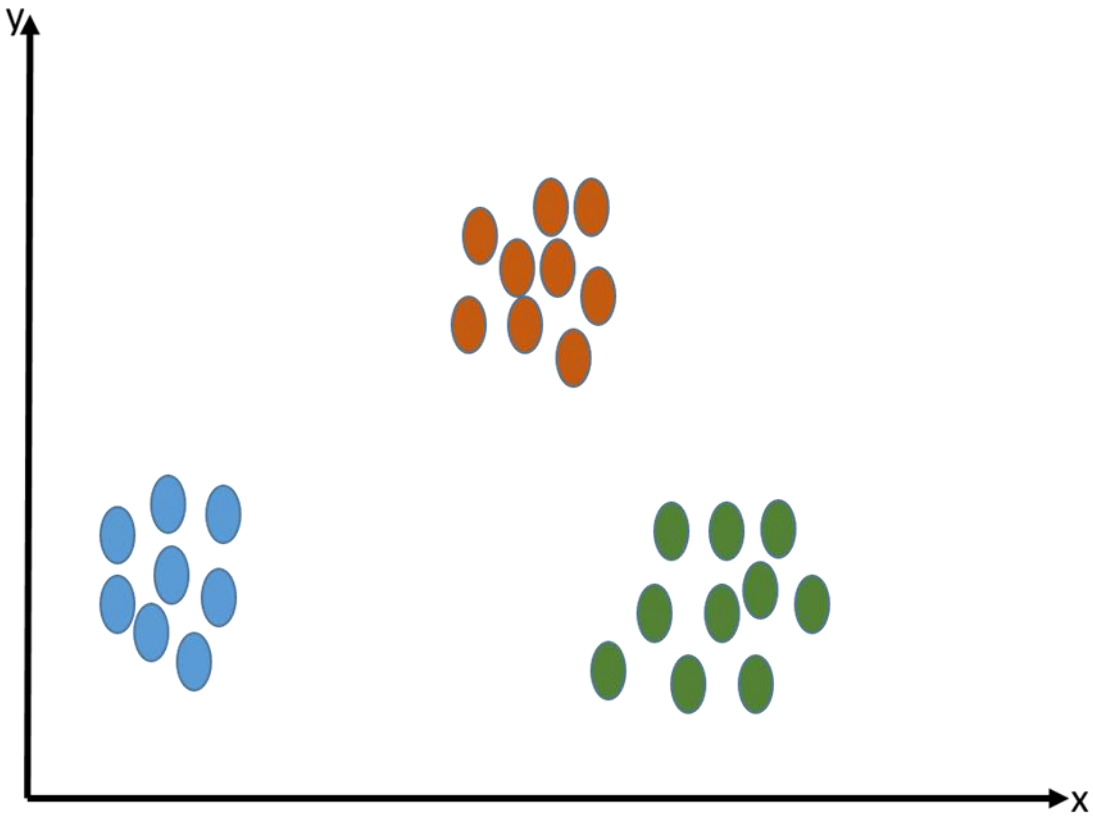
A tarefa de clusterização, também chamada de agrupamento ou análise de cluster tem o objetivo identificar e agrupar registros similares da base de dados formando clusters(grupos) de registros similares entre si e diferentes dos registros de outros clusters.

Segundo Oliveira(2008), “a clusterização de dados em grupos pode oferecer uma maneira de entender e extrair informações relevantes de grandes conjuntos de dados”.

Para Moscato, Von Zuben, “a análise de clusters envolve, portanto, a organização de um conjunto de padrões (usualmente representados na forma de vetores de atributos ou pontos em um espaço multidimensional – espaço de atributos) em clusters, de acordo com alguma medida de similaridade”.

A clusterização é um tipo de classificação não supervisionada pois não existem classes previamente definidas e os dados são agrupados a partir das suas similaridades para que sejam criados clusters que tenham algum sentido e utilidade. Já na classificação supervisionada são conhecidos um conjunto de dados e suas respectivas classes e existe a necessidade de definir a qual classe pertencem novos dados que surgem e ainda não tem a sua classe conhecida. A figura 4 ilustra um modo de se demonstrar a construção e distribuição de clusters.

Figura 4 - Demonstração do agrupamento de dados em clusters



Fonte: Do autor

> De uma forma mais formal, podemos definir Problemas de Clusterização da seguinte forma: dado um conjunto com n elementos X = {X1, X2, ..., Xn}, o problema de clusterização consiste na obtenção de um conjunto de k clusters, C = {C1, C2, ..., Ck}, tal que os elementos contidos em um cluster Ci possuam uma maior similaridade entre si do que com os elementos de qualquer um dos demais clusters do conjunto C. (Dias, Ochi, Soares).

O conjunto C é considerado uma clusterização com k clusters caso as seguintes condições sejam satisfeitas:

$$\bigcup_{i=1}^{k} C_i = X \tag{1}$$

$$C_i \neq \emptyset, para\ 1 \leq i \leq k \tag{2}$$

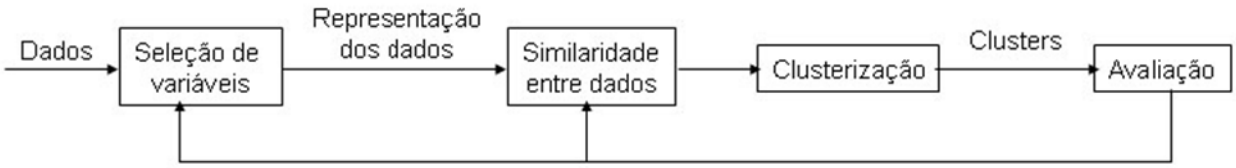
$$C_i \cap C_j = \emptyset, para\ 1 \leq i.j \leq k\ e\ i \neq j \tag{3}$$

De acordo com Dias, Ochi, Soares, se o valor de k for fornecido como parâmetro o problema é referenciado como problema de k-clusterização, e caso o valor de k seja desconhecido é referenciado como problema de clusterização automática e o valor de k é obtido como parte da solução do problema.

2.2.1 Principais fases do processo de clusterização

O processo de clusterização é dividido em diversas fases dentre as quais as principais são: pré-processamento e seleção de variáveis, medidas de similaridade, algoritmos de clusterização, validação e análise dos resultados. O ciclo a ser seguido entre essas tarefas está ilustrado na figura 5.

Figura 5 - Fases do processo de Clusterização



Fonte: Oliveira (2008)

Pré-processamento e seleção de variáveis: etapa na qual são verificadas as variáveis e atributos que possuem relevância no conjunto de dados inicial. Nessa etapa algumas variáveis são eliminadas por não serem úteis para a mineração e os dados são formatados para serem processados pelo algoritmo de clusterização. Segundo Oliveira (2008), “em um conjunto formado por n dados, o resultado é uma matriz n x d, onde d é o número de atributos. Assim um dado corresponde a um ponto no espaço d-dimensional, e a tarefa de clusterização consiste em identificar conjuntos de dados próximos nesse espaço d-dimensional”.

Medidas de Similaridade: a medida de similaridade é utilizada para definir a similaridade entre dois dados para que se possa definir a qual cluster os dados devem pertencer. Existem diversas formas de medir essa similaridade entre pares de dados e a escolha da medida de similaridade correta é muito importante para a clusterização dos dados. Segundo Oliveira (2008), a medida mais comum utilizada para medir a similaridade é a distância Euclidiana. No item 3.2.2 é feito um detalhamento do assunto e apresentadas as medidas de similaridade mais utilizadas.

Algoritmos de clusterização: Existem diversos algoritmos de clusterização disponíveis que realizam o agrupamento dos dados de diferentes maneiras. Por esse motivo é de

fundamental importância analisar o problema para poder selecionar ou desenvolver o algoritmo mais adequado para a solução do problema. De acordo com Oliveira (2008), dentre os algoritmos de clusterização destacam-se os algoritmos de clusterização hierárquica e os algoritmos por particionamento, esses e demais tipos de algoritmos serão aboradados no item 3.2.3.

Validação e análise dos resultados: diferentes algoritmos de clusterização agrupam os dados de diferentes formas e por isso é necessário haver uma validação dos resultados obtidos. Com a análise dos resultados podem ocorrer alterações na escolha das variáveis e da medida de similaridade que foram definidos nas etapas anteriores do processo. Existem três critérios para a validação dos resultados: índices externos: nesse critério os resultados obtidos são comparados com estruturas pré-estabelecidas; índices internos: são verificados os resultados obtidos pelo algoritmo para verificar se eles são apropriados com relação aos dados de entrada e ao problema enfrentado; índices relativos: os resultados obtidos por diferentes algoritmos são comparados para definir qual melhor representa os dados e auxilia na solução do problema.

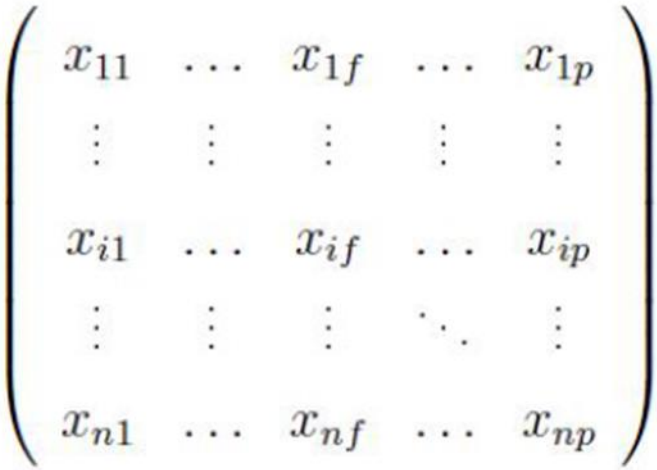
2.2.2 Medidas de Similaridade

Para os algoritmos de clusterização conseguirem realizar o agrupamento entre os dados é necessário que eles utilizem estruturas de dados para armazenar os objetos a serem processados e as relações entre eles, também é necessário definir uma medida de similaridade.

> Muitos métodos de clusterização utilizam como ponto de partida uma matriz que reflete de maneira quantitativa a proximidade entre os elementos de um conjunto de dados. Essa proximidade pode representar a dissimilaridade, distância ou similaridade entre dois elementos. Quanto maior a similaridade, ou menor a dissimilaridade ou distância, entre dois elementos, mais próximos esses elementos encontram-se. Essa matriz recebe o nome de matriz de similaridade ou matriz de proximidade. (Oliveira, 2008).

Matriz de dados: as linhas representam cada um dos objetos a serem clusterizados e as colunas os atributos ou características de cada objeto. Considerando n objetos cada qual com p atributos, tem-se uma matriz n x p como ilustrado na figura 6.

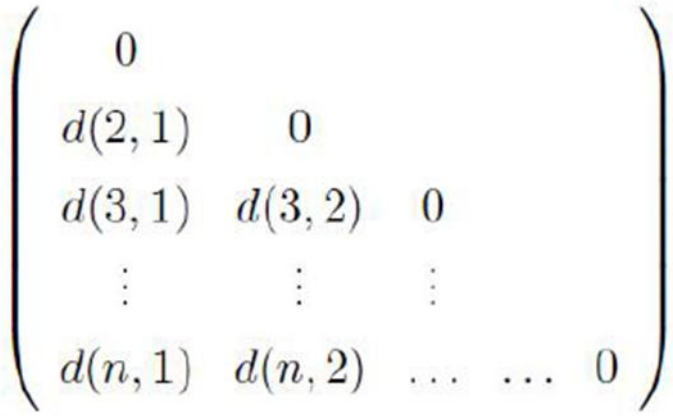
Figura 6 - Matriz de dados



Fonte: Morais (2010)

Matriz de dissimilaridade: cada elemento da matriz representa a distância entre pares de objetos. Considerando n objetos a serem clusterizados tem-se uma matriz quadrada Dn x n como a da figura 7.

Figura 7 - Matriz de dissimilaridade



Fonte: Morais (2010)

Com d(i, j) representando a distância ou dissimilaridade entre o objeto i e o j.

De acordo com Oliveira (2008), as medidas de similaridade variam conforme a representação e a escala dos atributos e um atributo pode possuir representação binária, discreta ou contínua. A escala indica o grau de importância do atributo em relação aos demais. Um atributo pode indicar o nível de aceitação de um produto, com valores variando de 0 a

100, ou pode indicar a distância entre duas cidades, caso em que o valor absoluto do atributo é relevante.

Entre os algoritmos existentes alguns utilizam apenas similaridade entre dados ou entre clusters e alguns algoritmos utilizam as duas medidas. Por esse motivo existem medidas para medir a similaridade entre dois dados e entre dois clusters, e essas medidas são comentadas nos itens abaixo.

2.2.2.1 Similaridade entre dados

Para clusterizar os dados de acordo com sua similaridade deve-se definir uma medida do quão próximos dois atributos estão ou quão bem seus valores se comparam. Uma distância pequena entre dois atributos deve indicar uma alta similaridade entre eles. Uma função de distância deve ser definida de tal forma que obedeça às seguintes propriedades:

$$d(i,j) \geq 0 \tag{4}$$

$$d(i,j) = d(j,i) \tag{5}$$

$$d(i,j) \leq d(i,h) + d(h,j) \tag{6}$$

$$d(i,j) = 0, se\ e\ somente\ se\ i = j \tag{7}$$

A unidade de medida dos dados pode afetar na clusterização, para que isso não ocorra é recomendado normalizar os dados. Uma das formas de realizar essa normalização é converter as medidas em atributos sem unidade. Essa normalização pode ser feita da seguinte maneira:

Calcular a média do desvio absoluto, sf:

$$sf = \frac{1}{n}(|x_{1f} - m_f| + |x_{2f} - m_f| + \cdots + |x_{nf} - m_f|) \tag{8}$$

Os valores x1f a xnf são os valores do atributo f para os n objetos a serem clusterizados e mf, é o valor médio do atributo f, isto é:

$$m_f = \frac{1}{n}(x_{1f} + x_{2f} + \cdots + x_{nf}) \tag{9}$$

O valor do i-ésimo objeto do atributo f normalizado será dado por:

$$z_{if} = \frac{x_{if} - m_f}{S_f} \tag{10}$$

Segundo Oliveira (2008), “a distância mais utilizada no cálculo da similaridade entre dois dados é a distância de Minkowski”. Na equação 11 está a fórmula para o cálculo dessa distância, com d sendo o número de atributos do dado.

$$d(i,j) = \sqrt[p]{\sum_{k=1}^{d}(|x_{ik} - x_{jk}|)^p}, p \geq 1 \tag{11}$$

De acordo com Oliveira (2008), a variação do parâmetro p define distâncias diferentes. As três variações mais comuns da distância de Minkowski são distância de Manhattan, distância euclidiana e a distância sup, demonstradas nas equações abaixo:

Distância de Manhattan, p = 1:

$$d(i,j) = \sum_{k=1}^{d}(|x_{ik} - x_{jk}) \tag{12}$$

Distância Euclidiana, p = 2:

$$d(i,j) = \sqrt{\sum_{k=1}^{d}(|x_{ik} - x_{jk}|)^2} \tag{13}$$

Distância sup, p → ∞:

$$\max_{1 \leq k \leq d} |x_{ik} - x_{jk}| \tag{14}$$

Em algumas execuções de clusterização há a necessidade ou o interesse de aumentar a importância de um atributo ou de um conjunto de atributos, nesse caso atribui-se pesos para cada um dos atributos. Essa alteração pode ser realizada para todas as medidas de distância demonstradas anteriormente. No caso da distância Minkowski temos:

$$d(i,j) = (W_1|x_{i1} - x_{j1}|^q + W_2|x_{i2} - x_{j2}|^q + \cdots + W_p|x_{ip} - x_{jp}|^q)^{\frac{1}{q}} \quad (15)$$

2.2.2.2 Similaridade entre clusters

Segundo Oliveira (2008), em alguns algoritmos de clusterização é necessário unir dois clusters similares.

> Uma maneira de medir essa similaridade é calcular a distância entre todos os pares de pontos dos dois clusters, onde cada ponto pertence a um cluster diferente. Podem ser escolhidas a distância mínima (distância do vizinho mais próximo), a distância máxima (distância do vizinho mais distante) ou a média da distância entre os pares de dados (distância média). (Oliveira, 2008).

De acordo com Oliveira (2008), uma outra forma de medir a similaridade entre clusters é a interconectividade entre eles. Nesse caso dois dados, um de cada cluster, possuem uma conexão caso a similaridade entre eles exceda algum limite. A similaridade entre os clusters é dada pela soma das conexões entre os pares de dados. A ideia é que sub-clusters pertencentes a um mesmo cluster tenham alta interconectividade.

2.2.3 Técnicas ou Métodos de Clusterização

Segundo Dias, Ochi, Soares, "no processo de clusterização, a busca pela melhor solução no espaço de soluções viáveis é um problema NP-Difícil". Dessa forma diversos métodos heurísticos foram propostos, entretanto pela grande diversidade de problemas de clusterização essas heurísticas são desenvolvidas para determinadas classes de problemas não existindo uma heurística que seja genérica o suficiente para que possa obter resultados satisfatórios em todas aplicações de clusterização.

De acordo com di Carlantonio (2001), o método ideal de clusterização deveria atender aos seguintes requisitos:

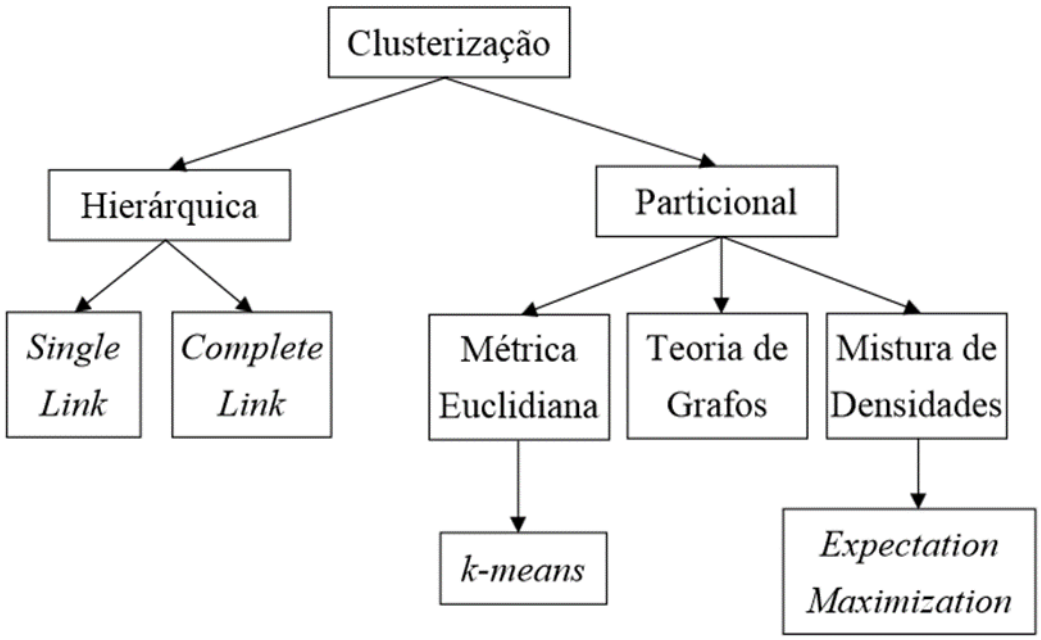
- Descobrir clusters com forma arbitrária;
- Identificar clusters de tamanhos variados;
- Aceitar os diversos tipos de variáveis possíveis;
- Ser insensível a ordem de apresentação dos objetos;
- Trabalhar com objetos com qualquer número de atributos;
- Ser escalável para lidar com qualquer quantidade de objetos;

- Fornecer resultados interpretáveis e utilizáveis;
- Ser robusto na presença de ruídos;
- Exigir o mínimo de conhecimento para determinar os parâmetros de entrada;
- Aceitar restrições;
- Encontrar o número adequado de clusters.

Conforme Agrawal(1998, apud di Carlantonio, 2001), “nenhuma técnica de Clustering corrente atende a todos estes pontos adequadamente, embora um trabalho considerável tem sido feito para atender a cada ponto separadamente”.

Segundo Oliveira (2008), “os algoritmos usados na clusterização podem ser classificados, por exemplo, de acordo com a abordagem utilizada na definição dos clusters: particionamento, redes auto organizáveis, baseado em densidade, hierárquico e baseado em grafos”. Dentre esses tipos de algoritmos os mais conhecidos e utilizados são os hierárquicos e os particionais. Na figura 8 estão demonstradas as subdivisões desses dois tipos de algoritmos.

Figura 8 - Subdivisões dos algoritmos de clusterização



Fonte: Moscato e Von Zuben

Nos itens abaixo estão detalhadas as diversas classificações dos algoritmos de clusterização.

2.2.3.1 Algoritmos por particionamento

Conforme di Carlantonio (2001), nos algoritmos por particionamento os dados são divididos em um número determinado de clusters, esse número de clusters é comumente chamado de k e é definido pelo usuário.

No início da execução, o algoritmo seleciona k objetos que serão os centros dos clusters. Depois os demais objetos são divididos entre os clusters de acordo com a medida de similaridade utilizada e cada objeto deve ficar no cluster que forneça a menor distância entre o objeto e o centro do cluster para se conseguir uma alta similaridade entre os objetos do mesmo cluster e diminuir a similaridade com objetos de outros clusters. Após isso, de acordo com di Carlantonio (2001), "o algoritmo utiliza uma estratégia iterativa de controle para determinar que objetos devem mudar de cluster de forma que otimizemos a função objetivo usada".

Existem duas possibilidades que podem ser utilizadas para definir os elementos que serão os centros dos clusters e utilizados para calcular a medida de similaridade:

- Utilizar a média dos objetos que compõem o cluster. Essa técnica é conhecida como k-means.
- Escolher como objeto central, o que se encontra mais perto do centro de gravidade do cluster. Essa técnica é conhecida como k-medoids.

2.2.3.1.1 Algoritmo k-means

O algoritmo k-means é o mais popular e mais simples algoritmo particional devido a sua facilidade de implementação, simplicidade e eficiência.

O funcionamento do algoritmo segue os passos definidos abaixo:

- Escolher o número de clusters e os objetos centrais dos clusters: nesse passo deve ser definido o número de clusters que serão utilizados e os objetos que serão o centro dos k clusters, esses objetos centrais normalmente são definidos de forma aleatório entre os objetos do conjunto de dados.
- Atribuir objetos aos clusters: nesse passo os demais objetos são distribuídos entre os clusters de acordo com a distância que possuem do centro do cluster, atribuindo cada objeto ao cluster que está a menor distância do objeto.

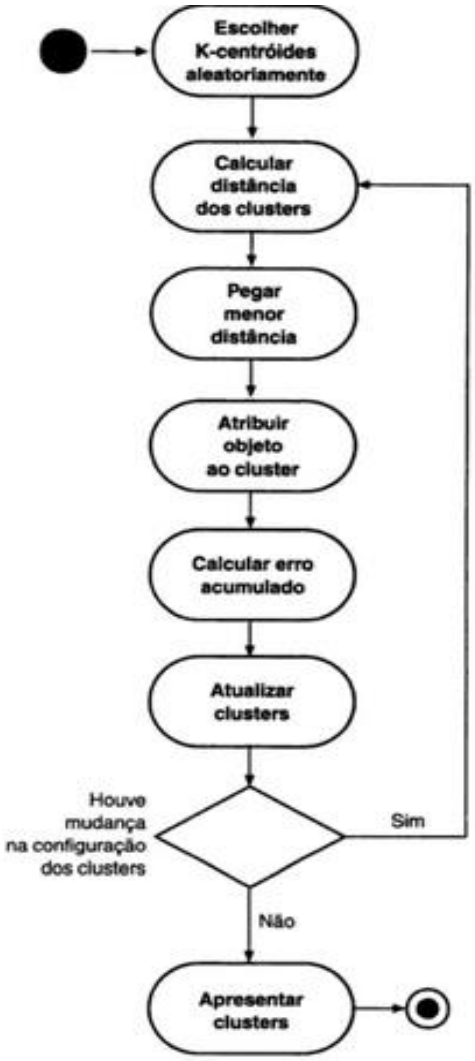
- Atualizar o centro dos clusters: nesse passo os centros dos clusters são redefinidos a partir do cálculo da média dos objetos pertencentes ao cluster.

Esse processo é repetido até a função objetivo ser minimizada e otimizada o máximo possível. A função objetivo mais utilizada é a função de erro quadrático definida na equação 16. Como não há garantias de que a função foi realmente otimizada ao máximo o critério de parada normalmente utilizado no algoritmo k-means é quando os centros dos clusters não sofrerem mais alterações.

$$E^2 = \sum_{k=1}^{K} \sum (x_i - c_k)^T (x_i - c_k), x_i \in C_k \tag{16}$$

A figura 9 demonstra o passo-a-passo da execução do algoritmo k-means.

Figura 9 - Passos do algoritmo k-means



Fonte: Goldschmidt, Passos (2005), p. 104

2.2.3.1.2 Algoritmo k-medoids

Segundo di Carlantonio (2001), “O algoritmo k-means é sensível a ruídos visto que um objeto com um valor extremamente grande pode, substancialmente, distorcer a distribuição de dados”.

> Para diminuir essa sensibilidade no algoritmo k-medoids, ao invés de utilizar o valor médio dos objetos em um cluster como um ponto referência, o medoid pode ser usado, que é o objeto mais centralmente localizado em um cluster. Assim, o método de particionamento pode ainda ser desempenhado no princípio de minimizar a soma das dissimilaridades entre cada objeto e seu ponto referência correspondente. Isto forma a base do método k-medoids. (di Carlantonio, 2001).

O algoritmo k-medoids inicialmente localiza o objeto localizado mais no centro do cluster (medoid). Os objetos restantes são clusterizados com o medoid com o qual possuem uma maior similaridade. Depois disso são realizadas trocas iterativas entre objetos medoids e não medoids para melhoria da clusterização. A qualidade é estimada por uma função custo que mede a similaridade média entre os objetos e o medoid de cada cluster (GOLDSCHMIDT, PASSOS, 2005, p. 103).

2.2.3.2 Algoritmos hierárquicos

De acordo com Oliveira (2008), nos algoritmos hierárquicos são produzidas diversas partições do conjunto inicial de dados pela junção ou divisão dos clusters de acordo com a medida de similaridade.

Segundo ESTER et al. (1998, apud di Carlantonio, 2001), “algoritmos hierárquicos criam uma decomposição hierárquica da base de dados. A decomposição hierárquica é representada por um dendrograma, uma árvore que iterativamente divide a base de dados em subconjuntos menores até que cada subconjunto consista de somente um objeto”.

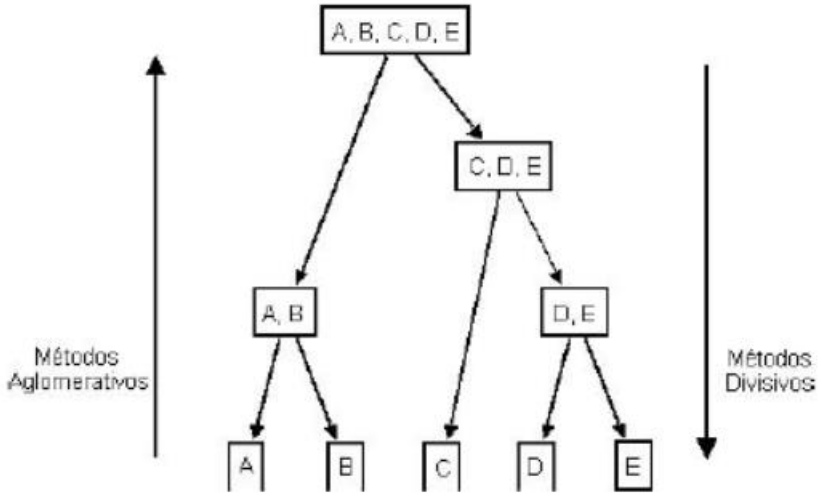
Esse dendograma pode ser criado de uma das duas seguintes formas:

- Abordagem aglomerativa (bottom-up): Nessa abordagem inicialmente cada um dos objetos é um cluster e em cada etapa do processo são calculadas as distâncias entre cada par de clusters, escolhem-se os dois clusters com a menor distância e é realizada a junção deles. Esse processo é repetido até reunir todos os objetos em um único cluster ou até que ocorra uma condição de término pré-definida.
- Abordagem divisiva (top-down): Nessa abordagem o processo é inverso ao bottom-up. Inicia-se com todos os objetos em um mesmo cluster e em cada etapa

um dos clusters é dividido em duas partes menores formando novos clusters. O processo é repetido até que existam n clusters de apenas um objeto ou até que uma condição de parada seja alcançada. Um exemplo de condição de parada é ter sido alcançada a quantidade desejada e definida de clusters.

A figura 10 mostra o esquema de um dendograma com a divisão dos diversos clusters de forma hierárquica, mostrando também as direções possíveis de construção do dendograma de acordo com a abordagem escolhida.

Figura 10 - Esquema de dendograma de algoritmo hierárquico de clusterização



Fonte: Oliveira (2008)

> Berkhin aponta como vantagens dos algoritmos de clusterização hierárquica a facilidade em lidar com qualquer medida de similaridade utilizada e a sua consequente aplicabilidade a qualquer tipo de atributo (numérico ou categórico). As desvantagens relacionam-se à imprecisão do critério de parada e ao fato de que a maioria dos algoritmos desta classe não revisitam os clusters formados ao longo de suas execuções. Este último aspecto está relacionado ao fato dos algoritmos para clusterização hierárquica serem apenas algoritmos construtivos, não permitindo o refinamento de soluções obtidas durante a sua execução. (Dias, Ochi, Soares).

Os algoritmos que utilizam a abordagem aglomerativa são mais populares e mais utilizados que os algoritmos com abordagem divisiva. E segundo Oliveira (2008), as medidas de similaridade utilizadas nos métodos hierárquicos podem ser baseadas na matriz de similaridade ou na matriz de dados e podem ser divididas nos seguintes tipos:

- Single Linkage: também conhecido como nearest neighbor, une os clusters com a distância mínima entre pares de dados de clusters diferentes.
- Complete Linkage: também conhecido como furthest neighbor, utiliza uma medida de distância oposta ao Single Linkage, utilizando a distância entre pares de dados mais distantes.
- Group Average Linkage: a distância entre dois clusters é a média da distância entre todos os pares formados por dados de clusters diferentes.
- Centroid Clustering: para cada cluster é calculado um centro de cluster, e são unidos os dois clusters com a menor distância entre os centros entre todos os pares de clusters.

Esses tipos de medidas são definidos mais formalmente nas equações 17, 18, 19 e 20 respectivamente.

$$d_{min}(C_i, C_j) = \min_{p \in C_i, p' \in C_j} |p - p'| \quad (17)$$

$$d_{max}(C_i, C_j) = \max_{p \in C_i, p' \in C_j} |p - p'| \quad (18)$$

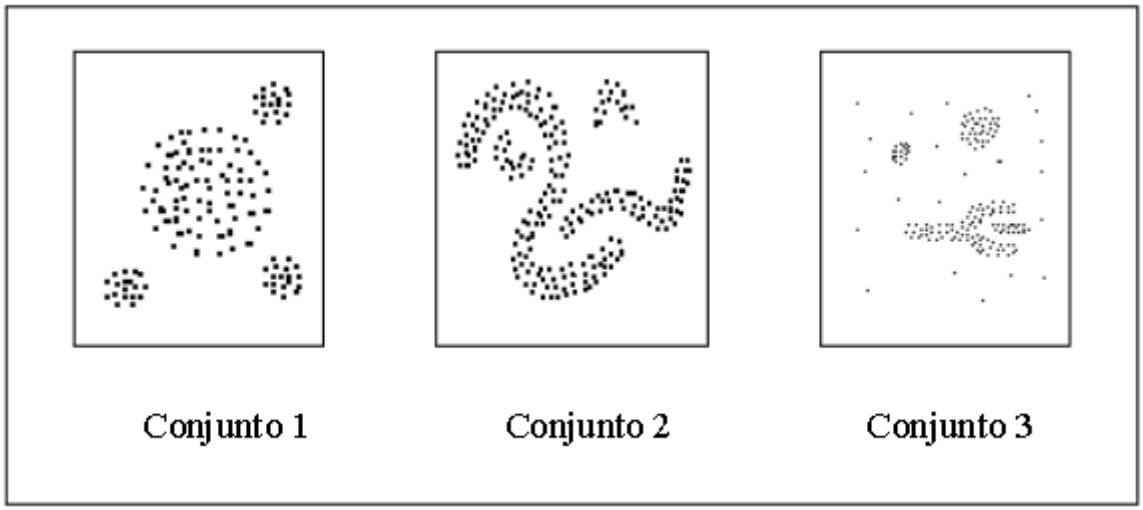
$$d_{mean}(C_i, C_j) = |m_i - m_j| \quad (19)$$

$$d_{ave}(C_i, C_j) = \frac{1}{n_i \, n_j} \sum_{p \in C_i} \sum_{p' \in C_j} |p - p'| \quad (20)$$

2.2.3.3 Algoritmos baseados em densidade

Segundo di Carlantonio (2001), algoritmos por particionamento e hierárquicos podem encontrar dificuldades para descobrir clusters com formas arbitrárias como elíptica ou cilíndrica. Para esses casos podem ser utilizados algoritmos baseados em densidade. A figura (Nº da Figura) mostra exemplos de formas de clusters que causam dificuldades para outros algoritmos e que são mais facilmente detectados por algoritmos baseados em densidade.

Figura 11 - Modelos de clusters identificados por algoritmos baseados em densidade



Fonte: di Carlantonio (2001)

De acordo com Oliveira (2008), "os métodos de clusterização por densidade utilizam critério de clusterização local, por considerarem a densidade de ligações entre os dados".

Conforme Agrawal et al. (1998, apud di Carlantonio, 2001), algoritmos baseados em densidade consideram como clusters regiões com uma densidade de objetos maior do que sua região vizinha ou os clusters são regiões densas de objetos no espaço de dados que são separadas por ruídos que são regiões de baixa densidade.

Um exemplo de algoritmo de clusterização baseado em densidade é o algoritmo DBSCAN que é explicado de forma mais detalhada no item abaixo.

2.2.3.3.1 DBSCAN

Segundo Oliveira (2008), o algoritmo DBSCAN (Density Based Spatial Clustering of Applications with Noise) utiliza uma abordagem baseada em centro e a ideia principal do algoritmo é que para pertencer a um cluster um dado deve ter uma vizinhança com um determinado raio (Eps) e que contenha um número mínimo de pontos (MinPts).

Nesse caso a vizinhança de um dado xi é definida pela equação.

$$N_{Eps}(x_i) = \{x_j \in X | d(x_i, x_j) \leq Eps\} \quad (21)$$

Depois de localizar regiões densas, os pontos podem ser classificados de acordo com a sua posição em relação a região mais densa:

$$x_i \text{ é } um\ ponto\ central\ se\ \left|N_{Eps}(x_i)\right| \geq MinPts \quad (22)$$

$$x_i \text{ é } um\ ponto\ periférico\ se\ x_i \in N_{Eps}(x_j) \quad (23)$$

Se xi não se enquadrar nos dois casos anteriores, ele é considerado um ponto ruidoso.

De acordo com Oliveira (2008), as entradas do algoritmo são o tamanho da vizinhança (Eps), o número mínimo de pontos da vizinhança (MinPts) e um conjunto de dados.

> O método DBSCAN encontra clusters verificando a vizinhança Eps de cada ponto na base de dados, começando por um objeto arbitrário. Se a vizinhança Eps de um ponto p contém mais do que MinPts, um novo cluster com p como um centro é criado. O método DBSCAN, então, iterativamente coleta objetos alcançáveis por densidade diretamente destes centros, que pode envolver a união de alguns clusters alcançáveis por densidade. O processo termina quando nenhum novo ponto pode ser adicionado a qualquer cluster. (di Carlantonio, 2001).

2.2.3.4 Algoritmos baseados em grafos

Segundo Oliveira (2008), algoritmos de clusterização baseados em grafos representam os dados e suas proximidades através de um grafo onde cada vértice representa um objeto do conjunto de dados e há uma aresta ligando dois objetos de acordo com a proximidade entre eles.

Conforme Oliveira (2008), "a maneira mais simples de estabelecer as ligações entre os vértices é conectar cada vértice aos vértices restantes, onde o peso indica a similaridade entre os dois dados". Nesse tipo de algoritmo um cluster é um subgrafo do grafo original.

Os algoritmos baseados em grafos são muito relacionados com os algoritmos hierárquicos e por particionamento e por esse motivo o resultado obtido nesse tipo de algoritmo pode ser uma partição ou uma hierarquia de partições.

2.2.3.5 Algoritmos baseados em redes auto organizáveis

De acordo com Oliveira (2008), as redes neurais são mais aplicadas na tarefa de classificação de dados, onde os dados são separados em grupos já conhecidos anteriormente. Mas existe um tipo de rede neural, as redes auto organizáveis ou SOM (Self Organizing Maps), que pode separar os dados em grupos que não são conhecidos inicialmente.

> Em linhas gerais, as redes auto organizáveis são formadas por um conjunto de neurônios, onde cada dado tem seus atributos conectados a todos os neurônios da rede. A essa ligação entre neurônio e atributo é dado um peso inicialmente aleatório. O aprendizado ocorre à medida em que os dados são apresentados à rede, e o neurônio, ou grupo de neurônios, com conjunto de pesos mais próximo do dado é escolhido para representá-lo. O neurônio vencedor tem seus pesos alterados a fim de representar melhor o dado atribuído a ele. Assim cada neurônio, ou grupo de neurônios, torna-se especialista na identificação dos atributos. (Oliveira, 2008).

Para Oliveira (2008), as vantagens do uso de redes auto organizáveis na clusterização são ser possível representar um conjunto de dados com diversos atributos em um mapa de baixa dimensão e o fato dos clusters similares estarem bem próximos no mapa. E entre as desvantagens, a principal é poder representar apenas clusters com forma esférica.

2.3 Descoberta e regras de associação e sequências temporais

A tarefa de descoberta de associação ou de regras de associação tem como objetivo demonstrar o quanto a ocorrência de um determinado conjunto de dados em uma base de dados influencia na ocorrência de um outro conjunto de dados e com isso encontrar dados que ocorrem simultaneamente e frequentemente juntos em transações de grandes bases de dados.

Um dos principais usos dessa técnica é analisar as transações de bases de dados de compra para localizar produtos que são frequentemente comprados juntos por um determinado tipo de cliente.

Segundo Silva (2004), existe um modelo matemático formalizado para problemas de descoberta de regras de associação conforme os pontos abaixo:

- I é um conjunto de itens de venda descritos por $I = \{i_1, i_2, i_3, ..., i_m\}$
- T representa uma transação de venda tal que $T \subseteq I$
- TID representa a chave que identifica a transação
- D representa um conjunto de transações de venda

- X e Y são conjuntos de itens de venda contidos em uma transação $X \subseteq T$ e $Y \subseteq T$
- Uma regra de associação é uma implicação de $X \Longrightarrow Y$ onde $X \subset I$,$Y \subset I$ e $X \cap Y = \emptyset$
- s é o Suporte de uma determinada regra $X \Longrightarrow Y$ e é dado através do total de transações contido no subconjunto de transações que contem $X \cup Y$ sobre o conjunto de transações D. O suporte é descrito pela é descrito pela seguinte fórmula:

Suporte = Nº de registros da tabela que contém o conjunto / Nº total de registros da tabela

- c é a Confiança de uma determinada regra $X \Longrightarrow Y$ e é dada através do total de registros do subconjunto $X \cup Y$ sobre o total de registros do subconjunto X. A confiança é descrita pela seguinte fórmula:

Confiança = Nº de registros da tabela que contém todos os itens da regra / Nº de registros da tabela que contém o antecendente da regra

De acordo com Silva (2004), "o Suporte e a Confiança das regras são de suma importância para o processo, sendo que somente as regras com um alto grau de suporte e confiança serão usadas".

> Normalmente estas regras são associadas a valores denominados Suporte (Sup) e Confiança (Conf) dos elementos das regras. O valor de Sup tem o objetivo de indicar a porcentagem de ocorrências da associação em relação ao montante total de registros, isto é, a probabilidade de que uma transação satisfaça a condição X. Já o valor de Conf, tem a função de indicar todas as ocorrências em percentual do antecedente onde o item consequente está associado, ou seja, é a probabilidade de que uma transação satisfaça a condição Y, se ela satisfaz a condição X. (Farias Junior, 2008).

Segundo Carvalho, Vasconcelos (2004), o problema da descoberta de regras de associação pode ser dividido nas duas partes abaixo:

- Encontrar todos os conjuntos de dados que possuem suporte acima de um limite mínimo determinado. O suporte de um conjunto é o número de transações em que existe esse conjunto.
- Gerar regras de associação a partir dos conjuntos de dados frequentes encontrados e selecionar apenas as regras que possuam grau de confiança que seja igual ou superior a confiança mínima definida.

2.3.1 Algoritmos de regras de associação

Existem vários algoritmos que podem ser utilizados para a descoberta de regras de associação. Nesse item serão apresentados os principais entre esses algoritmos.

Segundo Farias Junior (2008), “O funcionamento dos algoritmos para extração das regras se dá a partir da análise das combinações dos dados do conjunto a ser pesquisado. Com isto, a sua operação tem um crescimento exponencial em função do número de itens a ser comparado”.

De acordo com Silva (2004), existem dois tipos de algoritmos de regras de associação, os sequenciais e os paralelos. Nos algoritmos sequenciais considera-se que os conjuntos de itens estão em ordem lexicográfica (baseada no nome do item) criando uma maneira lógica de contar e gerar os grupos de itens. Já nos algoritmos paralelos existe a preocupação em paralelizar a tarefa de encontrar os grupos de itens.

2.3.1.1 Algoritmo AIS

Segundo Silva (2004), foi o primeiro algoritmo desenvolvido para gerar grandes grupos de itens em uma base de dados. Tem seu foco em descobrir regras quantitativas e é limitado a apenas um item subsequente.

> O AIS executa vários passos dentro da base de dados inteira, durante cada passo ele verifica todas as transações e no primeiro passo verifica o suporte para cada item, determinando o quanto estes são frequentes na base de dados. Para cada passo, o grande conjunto de itens é estendido para gerar os conjuntos de itens candidatos. Depois de mapear as transações são determinados os conjuntos de itens comuns entre o grande conjunto de passos anterior e os itens da transação atual, estes conjuntos de itens comuns são estendidos com outros itens da transação para gerar o novo conjunto de itens candidatos. (Silva, 2004).

Para executar esta tarefa são utilizadas ferramentas de estimativa e técnicas de poda para determinar conjuntos candidatos. Os candidatos que possuem suporte maior ou igual ao suporte mínimo pré-determinado são selecionados nos grandes conjuntos de itens. O processo é encerrado quando não há mais conjuntos de itens para serem determinados.

2.3.1.2 Algoritmo SETM

Conforme Silva (2004), no algoritmo SETM cada membro do conjunto de grandes conjuntos de itens é formalizado por <TID, conjunto_de_itens>, onde TID é um

identificador único da transação. E cada membro dos conjuntos candidatos também é formalizado por <TID, conjunto_de_itens>.

De acordo com Silva (2004), o algoritmo SETM é similar ao algoritmo AIS e também realiza vários passos sobre a base de dados. No primeiro passo o algoritmo calcula o suporte de cada item determinando o quanto eles são frequentes e grandes na base de dados. Após são gerados conjuntos de itens candidatos através da extensão do conjunto de itens em vários passos. Ao gerar os conjuntos candidatos, o algoritmo grava, de maneira sequencial, uma cópia do conjunto de itens candidatos juntamente com o TID da transação. Depois os conjuntos de itens candidatos são organizados e os pequenos conjuntos de itens são apagados através de uma função de agregação. O algoritmo encerra quando não encontrar mais grandes conjuntos de itens.

2.3.1.3 Algoritmo Apriori

O algoritmo Apriori é um dos algoritmos de regras de associação mais conhecidos e mais utilizados.

Segundo Farias Junior (2008), "a premissa básica do algoritmo Apriori é garantir que qualquer subconjunto de um conjunto de itens seja frequente". Com isso um conjunto de candidatos possuindo k itens pode ser gerado com uma combinação dos conjuntos de itens de tamanho k – 1 e como consequência eliminar os itens que contenham algum subconjunto não frequente, ou seja que não possui suporte igual ou superior ao suporte mínimo definido.

> O algoritmo principal (Apriori) faz uso de duas sub-rotinas: apriori-gen, para gerar o conjunto de itens candidatos (conjunto composto pelos valores correspondentes ao suporte de cada item). Neste conjunto são considerados todos os itens, independente deles atenderem o suporte_mínimo especificado) e eliminar aqueles que não são frequentes, e a subrotina subconjunto, utilizada para extrair as regras de associação. De forma geral, a sua meta é procurar por relações entre os dados enquanto eles são separados. Simultaneamente, o algoritmo calcula o valor correspondente à confiança e ao suporte. (Carvalho, Vasconcelos, 2004).

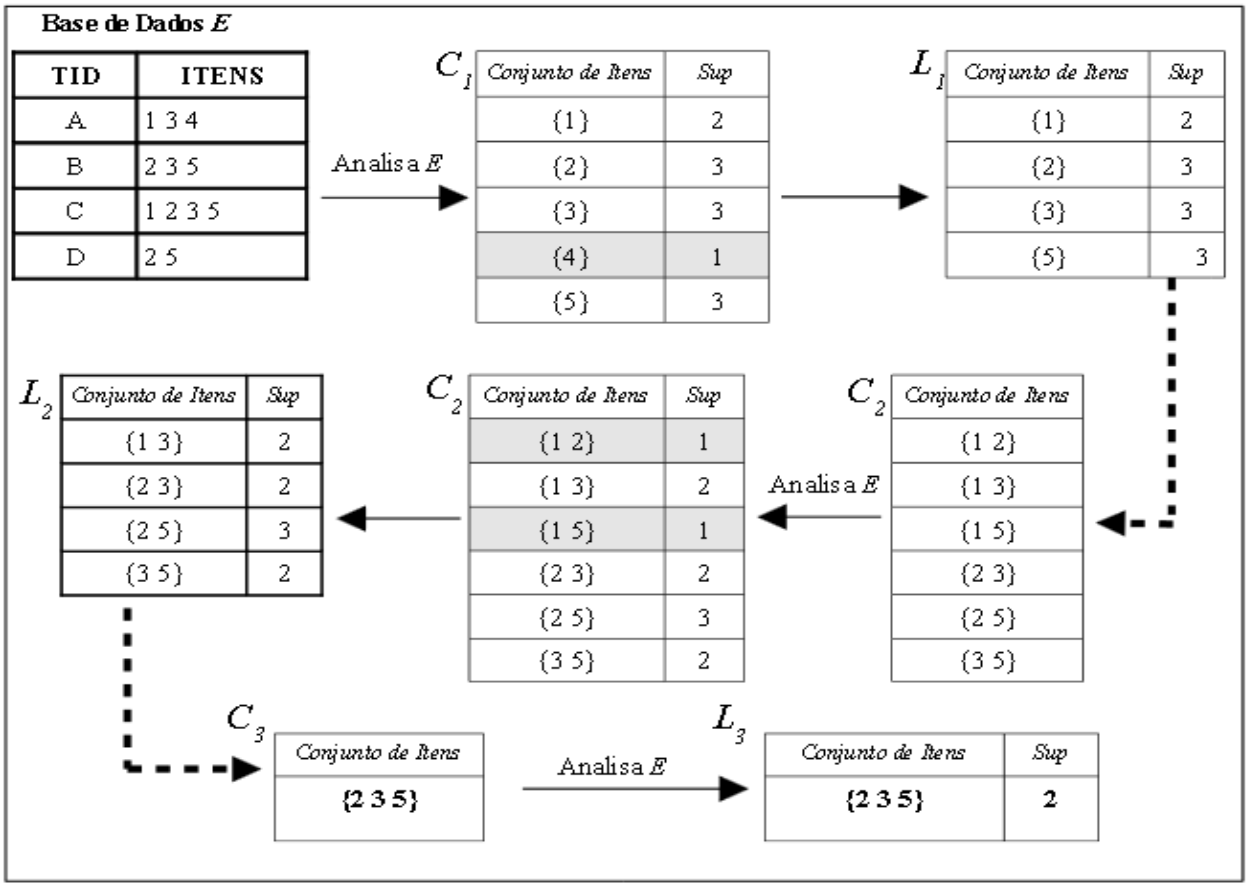
Como entrada do algoritmo Apriori é necessário fornecer um valor de suporte mínimo, um valor de confiança mínima e um arquivo de itens e transações da base de dados. E segundo Carvalho, Vasconcelos (2008) o funcionamento do algoritmo é o seguinte:

- Na primeira passagem do algoritmo, é calculado o suporte para cada item individual e todos os que satisfazem o suporte mínimo são selecionados e formam os conjuntos-de-1-item frequentes (F_1).

- Na segunda iteração são gerados conjuntos candidatos de 2 itens a partir da junção dos conjuntos-de-1-item, essa junção é feita através da rotina apriori-gen. Os suportes desses conjuntos de 2 itens são determinados e é realizada novamente a verificação dos conjuntos que satisfazem o suporte mínimo, com isso encontrando os conjuntos-de-2-itens frequentes.
- O algoritmo segue executando iterativamente, até que o conjunto-k-itens encontrado seja vazio.

A figura 12 mostra um exemplo de funcionamento do algoritmo apriori com a geração dos conjuntos de 1, 2 e 3 itens, as verificações na base de dados para remoção dos conjuntos que não possuem suporte igual ou superior ao suporte mínimo e a geração de novos conjuntos até a finalização da execução do algoritmo quando não é mais possível gerar novos conjuntos.

Figura 12 - Exemplo de execução do algoritmo Apriori



Fonte: Farias Junior (2008)

2.3.1.4 Algoritmo AprioriAll

O funcionamento básico do algoritmo AprioriAll é o mesmo do algoritmo Apriori mas além de considerar itens frequentes também considera sequências frequentes nas transações da base de dados e por essa razão para o algoritmo AprioriAll as transações devem ser datadas.

Segundo Vanzin (2004), o algoritmo AprioriAll é composto pelas seguintes cinco fases:

- Fase de ordenação: os dados de entrada do algoritmo são ordenados por atributo agrupador e pelo momento de ocorrência da transação.
- Fase de itens frequentes: todos os itens e conjuntos de itens que possuem suporte acima do suporte mínimo definido são identificados, ou seja, todos os itens cujo percentual de transações em que ocorrem é maior que o suporte mínimo.
- Fase de transformação: para otimizar o tempo de resposta do algoritmo os itens frequentes são mapeados para números inteiros.
- Fase da sequência: são geradas as sequências candidatas através da combinação dos itens frequentes levando em conta o critério temporal e o suporte mínimo.
- Fase maximal: nesta fase são localizadas as sequências maximais contidas no conjunto de sequências geradas, para diminuir o número de padrões sequenciais. Uma sequência é maximal quando ela não é uma subsequência de nenhuma outra sequência. Muitas vezes o interesse do usuário pode ser pelo suporte das subsequências das sequências maximais e por essa razão grande parte das implementações deste algoritmo não implementam esta fase.

As figuras 13, 14, 15 e 16 mostram exemplos de cada uma das fases de execução do algoritmo

Figura 13 - Fase de ordenação

ID comprador	Data	Itens
1	25/03/93	30
1	30/03/93	90
2	10/06/93	10, 20
5	12/06/93	90
2	15/06/93	30
2	20/06/93	40, 60, 70
3	25/06/93	30, 50, 70
4	25/06/93	30
4	30/06/93	40, 70
4	25/07/93	90

ID comprador	Data	Itens
1	25/03/93	30
1	30/03/93	90
2	10/06/93	10, 20
2	15/06/93	30
2	20/06/93	40, 60, 70
3	25/06/93	30, 50, 70
4	25/06/93	30
4	30/06/93	40, 70
4	25/07/93	90
5	12/06/93	90

Fonte: Santos, Wilkens (2009)

Figura 14 - Fase dos itens frequentes

ID comprador	Data	Itens
1	25/03/93	30
1	30/03/93	90
2	10/06/93	10, 20
2	15/06/93	30
2	20/06/93	40, 60, 70
3	25/06/93	30, 50, 70
4	25/06/93	30
4	30/06/93	40, 70
4	25/07/93	90
5	12/06/93	90

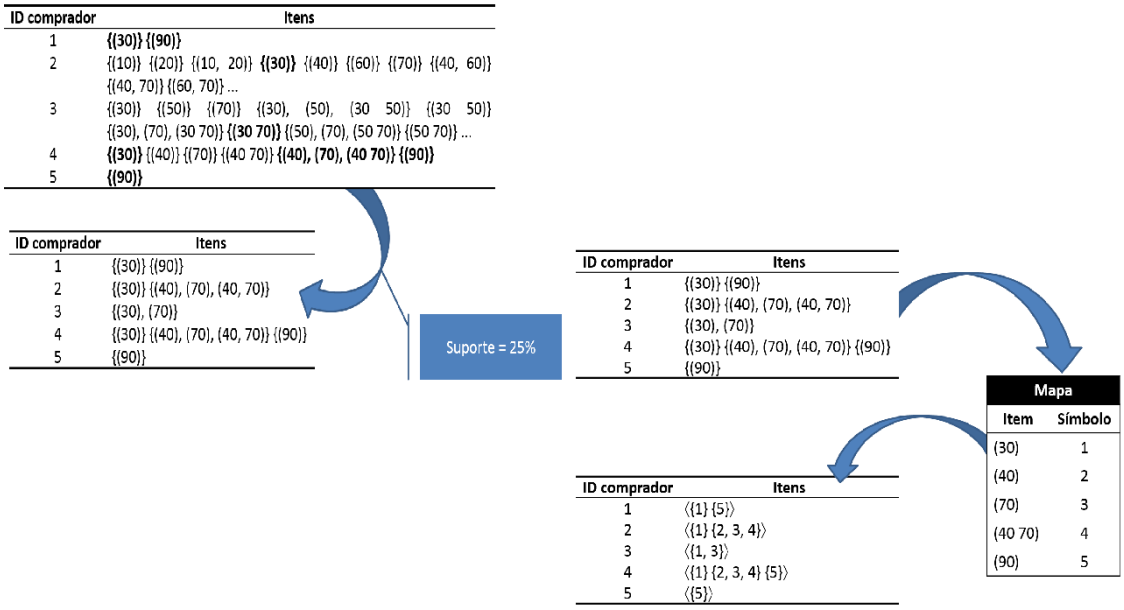
ID comprador	Itens
1	(30) (90)
2	(10, 20) (30) (40, 60, 70)
3	(30, 50, 70)
4	(30) (40, 70) (90)
5	(90)

ID comprador	Itens
1	(30) (90)
2	(10, 20) (30) (40, 60, 70)
3	(30, 50, 70)
4	(30) (40, 70) (90)
5	(90)

ID comprador	Itens
1	{(30)} {(90)}
2	{(10)} {(20)} {(10, 20)} {(30)} {(40)} {(60)} {(70)} {(40, 60)} {(40, 70)} {(60, 70)} ...
3	{(30)} {(50)} {(70)} {(30), (50), (30 50)} {(30 50)} {(30), (70), (30 70)} {(30 70)} {(50), (70), (50 70)} {(50 70)} ...
4	{(30)} {(40)} {(70)} {(40 70)} {(40), (70), (40 70)} {(90)}
5	{(90)}

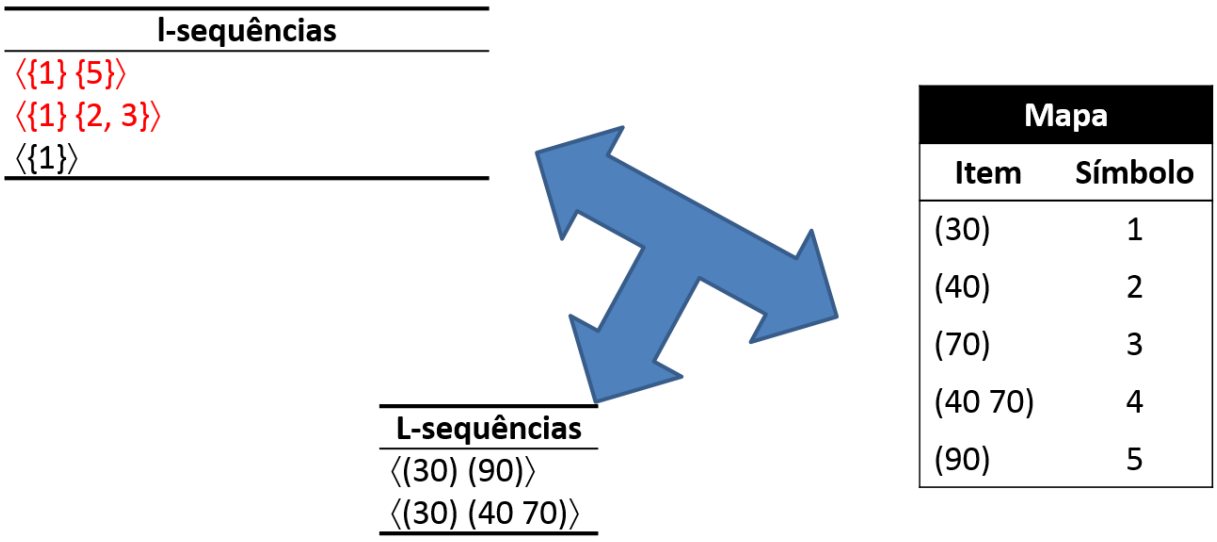
Fonte: Santos, Wilkens (2009)

Figura 15 - Fase de transformação



Fonte: Santos, Wilkens (2009)

Figura 16 - Fase maximal



Fonte: Santos, Wilkens (2009)

2.3.1.5 GSP

De acordo com Barbosa (2006), o algoritmo Generalized Sequential Patterns (GSP) foi uma das primeiras estratégias utilizadas para descoberta e extração de padrões sequenciais. O algoritmo extrai os padrões sequenciais iterativamente, de tal forma que, a

cada iteração k, a base de dados é inteiramente lida e são encontradas as sequências frequentes de tamanho k.

Segundo Barbosa (2006), no algoritmo GSP existem duas funções principais que ocorrem em todas iterações:

- Geração de sequências candidatas: no início de cada iteração, o conjunto de sequências candidatas de tamanho k (Ck) é gerado. A geração é feita em duas etapas: a etapa de Junção, quando ocorre a geração das sequências candidatas através da combinação das sequências do conjunto de sequências frequentes de tamanho k − 1 (Fk−1); e a etapa de Poda, quando as sequências candidatas que possuem alguma subsequência não frequente são eliminadas.
- Contagem do Suporte das Sequências Candidatas: esta função é executada após a geração do conjunto Ck. É feita uma leitura completa da base de dados para a contagem do suporte das sequências de Ck. O suporte de cada candidata é incrementado todas as vezes em que ela estiver contida em uma sequência de consumidor.

2.3.1.5.1 Geração de sequências candidatas

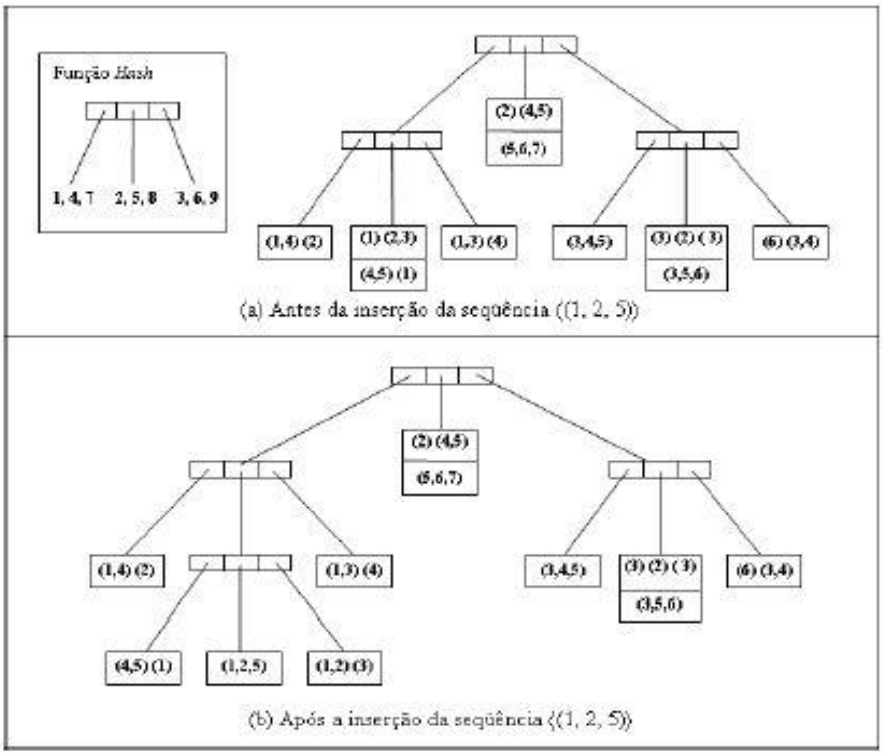
Neste algoritmo a geração de sequências candidatas é realizada em duas fases, a fase de junção e a fase de poda.

> A fase de junção, gera sequências juntando Lk-1 com Lk-1, em que uma sequência s1 junta-se com uma sequência s2 se a subsequência obtida da remoção do primeiro item de s1 for a mesma que a subsequência obtida pela remoção do último item de s2. A sequência candidata gerada da junção de s1 com s2 é a sequência s1 estendida com o último item de s2. O item adicionado torna-se num elemento separado se ele era um elemento separado em s2 e parte do último elemento de s1, caso contrário, quando juntamos L1 com L1, é necessário adicionar o item em s2 como parte do conjunto de itens e como um elemento separado. (Calapez, 2008).

De acordo com Calapez (2008), na fase de poda são removidas sequências contíguas candidatas que possuem subsequências que possuem suporte menor que o suporte mínimo. E se não houver restrição quanto à diferença máxima também são removidas sequências candidatas que possuam qualquer subsequência abaixo do suporte mínimo.

Para armazenar as sequências candidatas em memória principal é usada uma árvore hash para reduzir o tempo de acesso às sequências no momento da contagem de suporte.

Figura 17 - Exemplo de árvore hash manho 3.



Fonte: Barbosa (2006)

> Cada nó da árvore hash pode conter uma lista de sequências (nó folha) ou uma tabela hash (nó interno), em que cada entrada aponta para outro nó. O número de entradas na tabela hash determina o grau da árvore, que na Figura 2.1 (a) é igual a 3. Um nó folha armazena, de forma ordenada, uma lista de sequências candidatas, sendo que cada uma possui um contador para armazenar sua frequência na base de dados. A raiz da árvore hash é definida como tendo profundidade 1. Um nó interno de profundidade n aponta para outro nó de profundidade n + 1. A partir do nó raiz, as seqüências candidatas geradas são adicionadas à árvore hash. Percorre-se a árvore hash até alcançar um nó folha. O caminho percorrido através dos nós internos até alcançar um nó folha é determinado pelo resultado de uma função hash aplicada sobre os itens da sequência. Quando estiver percorrendo a árvore hash através de um nó interno de profundidade n, deve-se aplicar a função hash sobre o item de ordem n da seqüência. O retorno desta função informa qual ramo do nó será alcançado. (Barbosa, 2006).

2.3.1.5.2 Contagem do suporte de sequências candidatas

De acordo com Calapez (2008), a contagem do suporte no algoritmo GSP é dificultada pela definição de distância mínima e máxima, e por essa razão existe um teste de contenção que possui duas fases e verifica se uma sequência de dados contém uma sequência candidata. O algoritmo começa em fase forward a partir do primeiro elemento da sequência. Nessa fase o algoritmo encontra elementos sucessivos da sequência sempre que a diferença entre o tempo final do último elemento encontrado e o tempo inicial do elemento anterior for

menor que a distância máxima. Caso essa diferença seja superior a distância máxima o algoritmo altera para a fase de backward.

Na fase backward o algoritmo começa de trás e retira os elementos anteriores.

> Se si for o elemento corrente e o tempo final de si for t, o algoritmo entra o primeiro conjunto de transações contendo si-1 cujos tempos de transação são maiores que a diferença entre t e a distância máxima entre ocorrências. O tempo inicial de si-1 pode ser posterior ao tempo final de si. O algoritmo move para trás até que a restrição de distância entre o último elemento retirado e o elemento que o antecede seja satisfeita, ou até que o primeiro elemento seja retirado. O algoritmo então irá alternar para a fase de forward, encontrando elementos da sequência nos dados a partir do elemento seguindo ao último a ter sido retirado. Se um qualquer elemento não puder ser retirado, ou seja, se não existir um conjunto de transações subsequente que contenha o elemento, a sequência de dados não contem a sequência que se pretende encontrar. (Calapez, 2008).

Nesta fase de contagem do suporte, o algoritmo tem como objetivo reduzir o número de candidatos que necessitam ser verificados, adaptando com esse propósito a estrutura de dados em árvore (CALAPEZ, 2008).

3 PLATAFORMA LATTES

A Plataforma Lattes integra bases de dados de currículos de pesquisadores, de grupos de pesquisa e de instituições em um único sistema de informação desenvolvido pelo CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico). Essa plataforma foi lançada em agosto de 1999 com a primeira versão do currículo Lattes.

A plataforma é composta pela integração de quatro sistemas distintos: currículo Lattes, que é um sistema de informação responsável por registrar a vida curricular pregressa e atual dos pesquisadores; diretório de grupos de pesquisa, que é um sistema responsável por manter informações sobre os grupos de pesquisa existentes no país; diretório de instituições, cujo objetivo é armazenar informações sobre os institutos de pesquisa, universidades e outros, que demandam fomento ao CNPq; e sistema gerencial de fomento, cujo objetivo é aumentar a qualidade das atividades de fomento do CNPq.

3.1 Estrutura do currículo Lattes

O currículo Lattes é estruturado como um arquivo XML com estruturação hierárquica e suas principais áreas são:

Dados Gerais: possui um resumo do currículo, dados de identificação, dados de endereço, formação acadêmica, titulação, atuação profissional, áreas de atuação, idiomas, prêmios e títulos. Os dados de formação acadêmica e titulação incluem graduação, especialização, mestrado, doutorado com informações da instituição em que a formação foi realizada, do período em que foi realizada, da área ou das áreas em que se enquadra a formação, do título do trabalho apresentado. Os dados de atuação profissional englobam os dados das empresas e instituições em que o pesquisador trabalhou, com o período em que trabalhou, o cargo ocupado, funções desempenhadas.

Produção bibliográfica: reúne informações referentes a toda produção bibliográfica do pesquisador como trabalhos apresentados em eventos, artigos e livros publicados entre outros. Entre os dados dos trabalhos apresentados em eventos estão o nome, a data e o local do evento, o nome dos autores, o assunto do trabalho apresentado. Nas informações de artigos e livros publicados também existem dados dos autores, o assunto tratado, onde e quando o trabalho foi publicado.

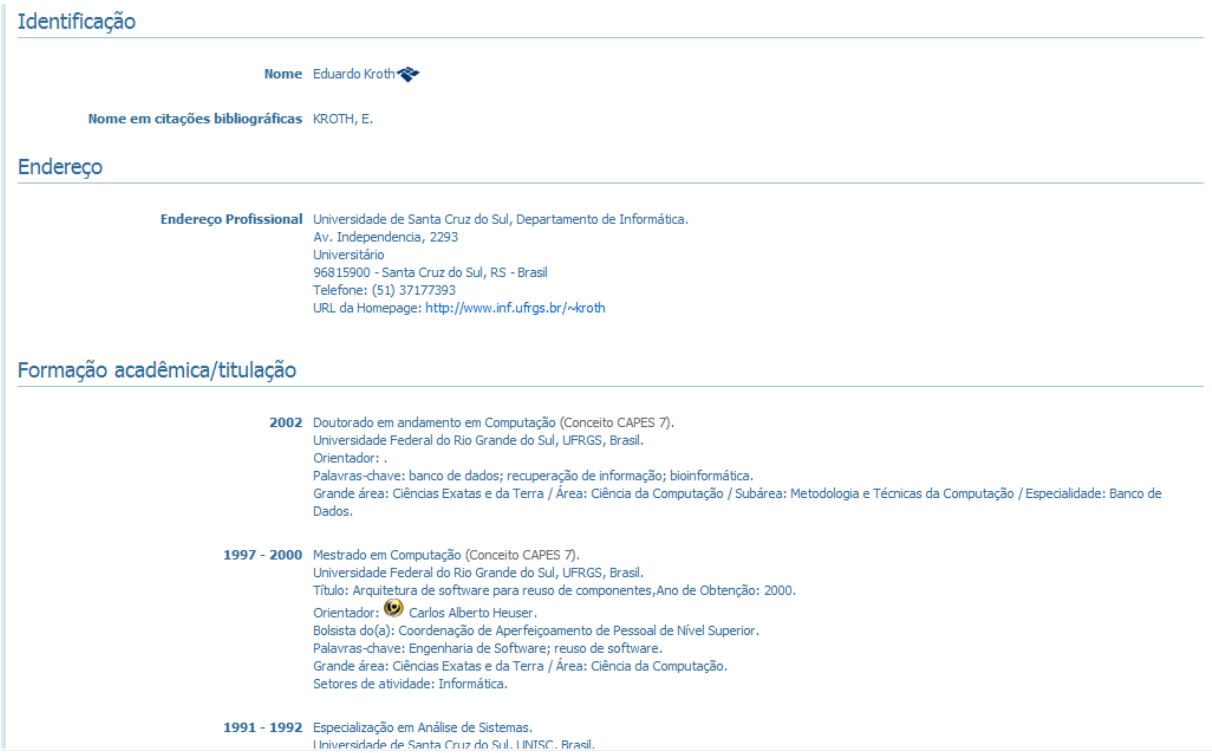
Produção técnica: reúne informações da produção técnica do pesquisador como trabalhos técnicos, softwares e produtos desenvolvidos, trabalhos apresentados, cursos ministrados. Nesses trabalhos técnicos existem informações dos autores, de onde e quando o trabalho foi ou está sendo executado.

Outra produção: reúne informações sobre orientações de trabalhos de conclusão de curso, mestrado, doutorado, realizadas pelo pesquisador. Nas informações das orientações realizadas pelo professor estão informações como o nome do aluno orientado, o título do trabalho, o tipo de trabalho (trabalho de conclusão de graduação, dissertação de mestrado, tese de doutorado), o ano em que a orientação foi realizada e a instituição na qual o trabalho foi apresentado.

Dados Complementares: reúne informações sobre formações complementares nas quais o pesquisador participou. Também reúne os dados de congressos, de bancas avaliadoras e julgadoras das quais ele já participou e de orientações em andamento. As informações disponíveis sobre bancas avaliadoras nas quais o professor participou estão divididas em bancas de doutorados, mestrados, especializações e trabalhos de conclusão de curso. Em todas elas existem informações como nome dos avaliadores, nome de quem apresentou o trabalho, título do trabalho apresentado, ano da apresentação e instituição na qual ocorreu a apresentação. Nas orientações de trabalhos em andamento existem informações como o nome do aluno que está sendo orientado, o título do trabalho, o tipo de trabalho (trabalho de conclusão de graduação, dissertação de mestrado, tese de doutorado), o curso e a instituição onde o trabalho será apresentado.

A figura 18 demonstra um exemplo de exibição dos dados do currículo Lattes, a figura 19 demonstra a visualização dos mesmos dados em formato XML e a figura 20 apresenta um esquema dos dados existentes no currículo Lattes, suas relações e hierarquia.

Figura 18 - Exibição padrão dos dados de currículo Lattes



Identificação

Nome Eduardo Kroth

Nome em citações bibliográficas KROTH, E.

Endereço

Endereço Profissional Universidade de Santa Cruz do Sul, Departamento de Informática.
Av. Independencia, 2293
Universitário
96815900 - Santa Cruz do Sul, RS - Brasil
Telefone: (51) 37177393
URL da Homepage: http://www.inf.ufrgs.br/~kroth

Formação acadêmica/titulação

2002 Doutorado em andamento em Computação (Conceito CAPES 7).
Universidade Federal do Rio Grande do Sul, UFRGS, Brasil.
Orientador: .
Palavras-chave: banco de dados; recuperação de informação; bioinformática.
Grande área: Ciências Exatas e da Terra / Área: Ciência da Computação / Subárea: Metodologia e Técnicas da Computação / Especialidade: Banco de Dados.

1997 - 2000 Mestrado em Computação (Conceito CAPES 7).
Universidade Federal do Rio Grande do Sul, UFRGS, Brasil.
Título: Arquitetura de software para reuso de componentes,Ano de Obtenção: 2000.
Orientador: Carlos Alberto Heuser.
Bolsista do(a): Coordenação de Aperfeiçoamento de Pessoal de Nível Superior.
Palavras-chave: Engenharia de Software; reuso de software.
Grande área: Ciências Exatas e da Terra / Área: Ciência da Computação.
Setores de atividade: Informática.

1991 - 1992 Especialização em Análise de Sistemas.
Universidade de Santa Cruz do Sul, UNISC, Brasil.

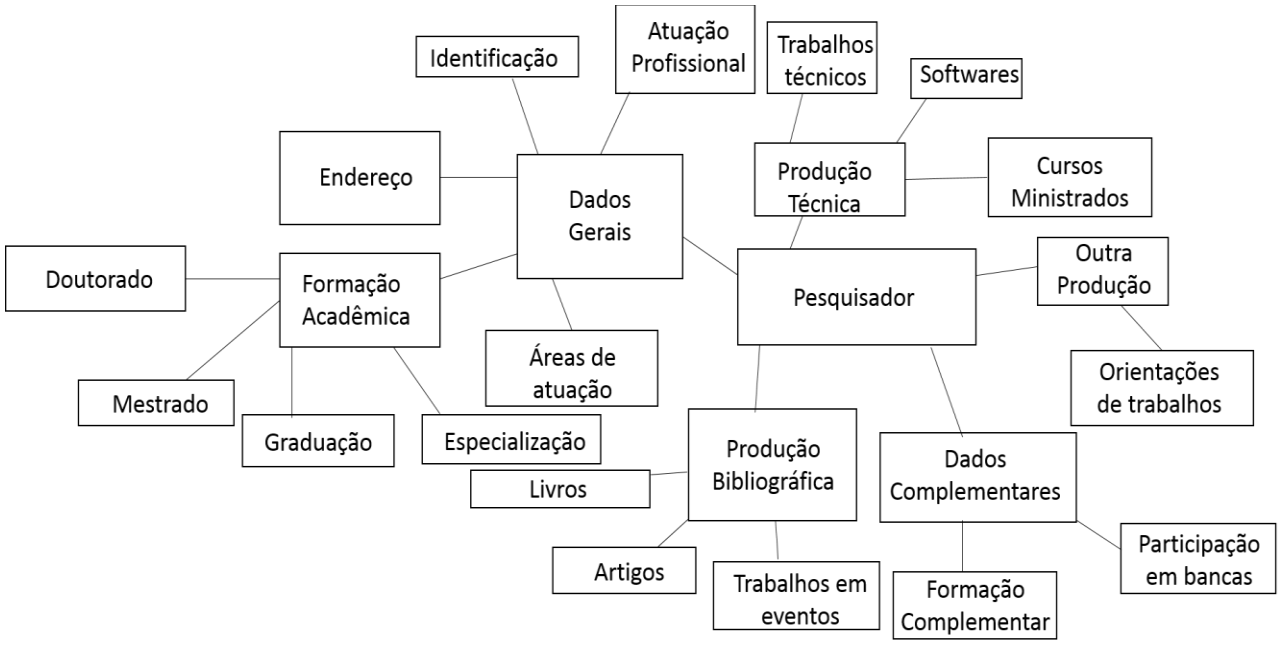
Fonte: Plataforma Lattes

Figura 19 - Exemplo de currículo Lattes em formato XML


```
<?xml version="1.0" encoding="ISO-8859-1"?>
- <CURRICULO-VITAE NUMERO-IDENTIFICADOR="" HORA-ATUALIZACAO="131702" DATA-ATUALIZACAO="14102013" SISTEMA-ORIGEM-XML="LATTES_OFFLINE">
  - <DADOS-GERAIS DATA-FALECIMENTO="" PERMISSAO-DE-DIVULGACAO="NAO" CIDADE-NASCIMENTO="São Paulo" UF-NASCIMENTO="SP" PAIS-DE-NASCIMENTO="Brasil"
    NACIONALIDADE="B" NOME-EM-CITACOES-BIBLIOGRAFICAS="KROTH, E." NOME-COMPLETO="Eduardo Kroth">
      <RESUMO-CV TEXTO-RESUMO-CV-RH-EN="" TEXTO-RESUMO-CV-RH=""/>
      <OUTRAS-INFORMACOES-RELEVANTES OUTRAS-INFORMACOES-RELEVANTES=""/>
    - <ENDERECO FLAG-DE-PREFERENCIA="ENDERECO_INSTITUCIONAL">
        <ENDERECO-PROFISSIONAL HOME-PAGE="http://www.inf.ufrgs.br/~kroth" FAX="" RAMAL="" TELEFONE="37177393" DDD="51" CEP="96815900" CAIXA-POSTAL=""
          CIDADE="Santa Cruz do Sul" BAIRRO="Universitário" LOGRADOURO-COMPLEMENTO="Av. Independencia, 2293" UF="RS" PAIS="Brasil" NOME-
          ORGAO="Departamento de Informática" CODIGO-ORGAO="531001000995" NOME-UNIDADE="" CODIGO-UNIDADE="" NOME-INSTITUICAO-EMPRESA="Universidade
          de Santa Cruz do Sul" CODIGO-INSTITUICAO-EMPRESA="531000000002"/>
      </ENDERECO>
    - <FORMACAO-ACADEMICA-TITULACAO>
        <GRADUACAO NOME-CURSO-INGLES="" TITULO-DO-TRABALHO-DE-CONCLUSAO-DE-CURSO-INGLES="" CODIGO-CURSO-CAPES="" NUMERO-ID-ORIENTADOR="" NOME-
          AGENCIA="" CODIGO-AGENCIA-FINANCIADORA="" NOME-ORIENTADOR-GRAD="" NOME-INSTITUICAO-OUTRA-GRAD="" CODIGO-INSTITUICAO-OUTRA-GRAD="" NOME-
          INSTITUICAO-GRAD="" CODIGO-INSTITUICAO-GRAD="" TIPO-GRADUACAO="" FLAG-BOLSA="NAO" ANO-DE-CONCLUSAO="1986" ANO-DE-INICIO="1984" STATUS-DO-
          CURSO="CONCLUIDO" CODIGO-AREA-CURSO="" NOME-CURSO="Tecnólogo Em Processamento de Dados" CODIGO-CURSO="" NOME-INSTITUICAO="Universidade do
          Vale do Rio dos Sinos" CODIGO-INSTITUICAO="000900000007" NOME-DO-ORIENTADOR="" TITULO-DO-TRABALHO-DE-CONCLUSAO-DE-CURSO="" NIVEL="1"
          SEQUENCIA-FORMACAO="2"/>
        <ESPECIALIZACAO NOME-CURSO-INGLES="" NOME-AGENCIA="" CODIGO-AGENCIA-FINANCIADORA="" FLAG-BOLSA="NAO" ANO-DE-CONCLUSAO="1992" ANO-DE-
          INICIO="1991" STATUS-DO-CURSO="CONCLUIDO" NOME-CURSO="Análise de Sistemas" CODIGO-CURSO="" NOME-INSTITUICAO="Universidade de Santa Cruz do
          Sul" CODIGO-INSTITUICAO="531000000002" NOME-DO-ORIENTADOR="" NIVEL="2" SEQUENCIA-FORMACAO="3" TITULO-DA-MONOGRAFIA-INGLES="" CARGA-
          HORARIA="" TITULO-DA-MONOGRAFIA=""/>
      - <MESTRADO NOME-CURSO-INGLES="Computer Science" CODIGO-CURSO-CAPES="42001013004P4" NUMERO-ID-ORIENTADOR="0455487141833418" NOME-
          AGENCIA="Coordenação de Aperfeiçoamento de Pessoal de Nível Superior" CODIGO-AGENCIA-FINANCIADORA="045000000000" FLAG-BOLSA="SIM" ANO-DE-
          CONCLUSAO="2000" ANO-DE-INICIO="1997" STATUS-DO-CURSO="CONCLUIDO" CODIGO-AREA-CURSO="10300007" NOME-CURSO="Computação" CODIGO-
          CURSO="42000041" NOME-INSTITUICAO="Universidade Federal do Rio Grande do Sul" CODIGO-INSTITUICAO="019200000005" NIVEL="3" SEQUENCIA-FORMACAO="4"
          NOME-DO-CO-ORIENTADOR="" TITULO-DA-DISSERTACAO-TESE-INGLES="" NOME-COMPLETO-DO-ORIENTADOR="Carlos Alberto Heuser" TITULO-DA-DISSERTACAO-
          TESE="Arquitetura de software para reuso de componentes" ANO-DE-OBTENCAO-DO-TITULO="2000" NOME-ORIENTADOR-DOUT="" NOME-INSTITUICAO-OUTRA-DOUT=""
          CODIGO-INSTITUICAO-OUTRA-DOUT="" NOME-INSTITUICAO-DOUT="" CODIGO-INSTITUICAO-DOUT="" TIPO-MESTRADO="">
          <PALAVRAS-CHAVE PALAVRA-CHAVE-6="" PALAVRA-CHAVE-5="" PALAVRA-CHAVE-4="" PALAVRA-CHAVE-3="" PALAVRA-CHAVE-2="reuso de software" PALAVRA-
            CHAVE-1="Engenharia de Software"/>
        - <AREAS-DO-CONHECIMENTO>
            <AREA-DO-CONHECIMENTO-1 NOME-DA-ESPECIALIDADE="" NOME-DA-SUB-AREA-DO-CONHECIMENTO="" NOME-DA-AREA-DO-CONHECIMENTO="Ciência da
              Computação" NOME-GRANDE-AREA-DO-CONHECIMENTO="CIENCIAS_EXATAS_E_DA_TERRA"/>
          </AREAS-DO-CONHECIMENTO>
          <SETORES-DE-ATIVIDADE SETOR-DE-ATIVIDADE-3="" SETOR-DE-ATIVIDADE-2="" SETOR-DE-ATIVIDADE-1="Informática"/>
        </MESTRADO>
      - <DOUTORADO NOME-CURSO-INGLES="Computer Science" CODIGO-CURSO-CAPES="42001013004P4" NUMERO-ID-ORIENTADOR="" NOME-AGENCIA="" CODIGO-AGENCIA-
          FINANCIADORA="" FLAG-BOLSA="NAO" ANO-DE-CONCLUSAO="" ANO-DE-INICIO="2002" STATUS-DO-CURSO="EM_ANDAMENTO" CODIGO-AREA-CURSO="10300007"
```


Fonte: Plataforma Lattes

Figura 20 - Esquema das relações dos dados no currículo Lattes



Fonte: Do autor

4 QUALIS

Qualis é um sistema criado pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e que é utilizado para classificar a produção científica de programas de pós-graduação com base nos artigos publicados em periódicos científicos.

A classificação é realizada por comitês de consultores de cada área de avaliação de acordo com critérios definidos pela área e aprovados pelo Conselho Técnico-Científico da Educação Superior (CTC-ES) que procuram refletir a importância relativa dos diferentes periódicos para uma determinada área.

A estratificação da qualidade dessa produção é realizada de forma indireta. Dessa forma, o Qualis afere a qualidade dos artigos e de outros tipos de produção, a partir da análise da qualidade dos veículos de divulgação, ou seja, periódicos científicos.

A classificação de periódicos é realizada pelas áreas de avaliação e passa por processo anual de atualização. Esses veículos são enquadrados em estratos indicativos da qualidade - A1, o mais elevado; A2; B1; B2; B3; B4; B5; C - com peso zero. E o mesmo periódico pode ser classificado em duas ou mais áreas diferentes e receber diferentes avaliações em cada uma dessas áreas.

A função do QUALIS é exclusivamente para avaliar a produção científica dos programas de pós-graduação.

5 METODOLOGIA

Foi realizada uma pesquisa quantitativa e exploratória com pesquisa bibliográfica referente aos assuntos tratados neste trabalho e em um momento posterior com relação a geração de conhecimento útil através dos dados de uma base de dados utilizando técnicas de KDD (Descoberta de Conhecimento em Bases de Dados) e Data Mining.

Segundo Dalfovo, Lana, Silveira (2008), na pesquisa quantitativa o pesquisador inicia com quadros conceituais de referência bem estruturados e a partir deles formula hipóteses sobre os fenômenos e situações estudados. A partir dessas hipóteses é levantada uma lista de consequências. E com a coleta de dados surgirão números e informações conversíveis em números que permitirão verificar a ocorrência ou não das consequências levantadas e a aceitação ou não das hipóteses iniciais.

De acordo com Gerhardt, Silveira (2009), a pesquisa quantitativa focaliza uma quantidade pequena de conceitos, inicia com ideias preconcebidas do modo que os conceitos estão interligados, utiliza procedimentos estruturados para a coleta de dados, enfatiza a objetividade na coleta e análise dos dados e analisa dados numéricos através de procedimentos estatísticos.

Para Gil (2007, apud Gerhardt, Silveira, 2009), a pesquisa exploratória tem como objetivo proporcionar maior familiaridade com o problema, tornando-o mais explícito e facilitando a construção de hipóteses. A maioria dessas pesquisas envolve levantamento bibliográfico e/ou entrevista com pessoas que tiveram experiências práticas com o problema e/ou análise de exemplos que auxiliem na compreensão do problema.

Inicialmente foi realizado um levantamento bibliográfico em livros, artigos e outras publicações referentes aos seguintes assuntos que são abordados neste trabalho, KDD, Mineração de Dados com foco em Clusterização e Regras de Associação e a Plataforma Lattes focando nos currículos de professores e pesquisadores. Após esse levantamento foram analisadas as referências coletados selecionando as que realmente tiveram utilidade para o desenvolvimento do trabalho e foi realizada a fase de escrita do mesmo.

Posteriormente foram definidas a forma como será feita a coleta dos dados necessários para a geração do conhecimento útil, as técnicas que foram utilizadas nesse processo e ocorreu o desenvolvimento de uma ferramenta para extração dos dados dos currículos Lattes de professores e pesquisadores, e para a aplicação, sobre esses dados, de

algoritmos de clusterização e de regras de associação para possibilitar a geração de conhecimento útil a partir dos dados que é o objetivo principal deste trabalho.

6 TRABALHOS RELACIONADOS

Existem diversos trabalhos e artigos publicados sobre a aplicação de técnicas de mineração de dados sobre variados tipos de dados. Entre eles destacam-se os seguintes pela relação com esse trabalho:

- Extração de Conhecimento da Plataforma Lattes Utilizando Técnicas de Mineração de Dados: Estudo de Caso POLI/UPE: nesse trabalho existe uma revisão bibliográfica sobre tópicos referentes a mineração de dados e a Plataforma Lattes e é descrita a construção de uma ferramenta para extração de dados de currículos Lattes, o uso de técnicas de mineração de dados sobre os dados coletados com o uso de uma ferramenta gratuita e os testes realizados para validação do trabalho. (MORAIS, 2010).
- Aplicação de Técnicas de Mineração de Dados para Caracterização de Grupos de Cidades Produtoras de Cana-De-Açúcar do Estado de São Paulo e Definição de Políticas Especificas: nesse trabalho existe uma breve revisão de tópicos de mineração de dados e a descrição do uso dessas técnicas sobre dados de produção de cana de açúcar para separar as cidades produtoras do estado de São Paulo em grupos específicos para possibilitar a criação de políticas específicas para cada um dos grupos. (MEDEIROS et al., 2010).
- Uso de Técnicas de Data mining no Monitoramento de Alunos On-line: esse trabalho possui definições sobre os assuntos de ambientes de ensino online e mineração de dados e sobre o uso de técnicas de mineração de dados em ambientes de ensino online. O trabalho também cita exemplos de ambientes de ensino online que fazem uso de técnicas de mineração de dados. (ARAÚJO, CUNHA, 2003).
- Extração e Tratamento de Dados na Base Lattes Para Identificação de Core Competencies em Dengue: esse trabalho relata a busca por dados acerca da doença Dengue nas produções científicas, bibliográficas e tecnológicas de pesquisadores, em grupos de estudo e pesquisa e em dados de instituições disponíveis na Plataforma Lattes. (CHALCO et al., 2014).
- Utilização de Técnicas de “Data Mining” para o Reconhecimento de Caracteres Manuscritos: este trabalho possui uma revisão sobre técnicas de KDD e mineração de dados, principalmente sobre regras de associação. O trabalho também relata o uso da técnica de regras de associação para o reconhecimento de caracteres manuscritos,

testes realizados sobre esse ponto e resultados obtidos. (CARVALHO, MONGIOVI, SAMPAIO, 2000).

7 SOLUÇÃO DESENVOLVIDA

Neste capítulo serão apresentados os pontos referentes a solução desenvolvida para o problema de geração de conhecimento útil a partir dos dados existentes em currículos Lattes.

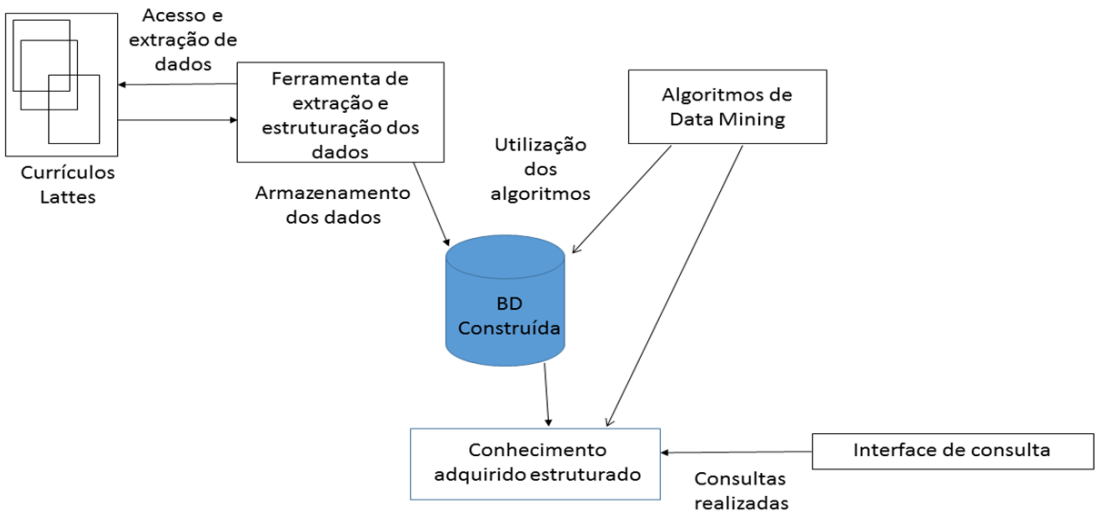
7.1 Visão Geral

A ferramenta desenvolvida neste trabalho realiza a extração de dados referentes à produção científica, bibliográfica e tecnológica dos currículos Lattes dos professores. Os dados são lidos de arquivos XMLs dos currículos Lattes, estruturados e salvos em uma base de dados modelada e construída anteriormente.

A ferramenta também permite que sobre os dados extraídos sejam aplicados algoritmos de Data Mining, mais especificamente de regras de associação, que realizam o vínculo entre a ocorrência de um determinado grupo de dados com a ocorrência de outro grupo de dados diferente nas transações da base de dados possibilitando a verificação da ocorrência simultânea desses grupos de dados; e de clusterização, que agrupa dados similares em clusters para facilitar a localização e verificação desses dados.

A figura 21 mostra um esquema básico de sequência das operações que são realizadas pelo sistema desenvolvido.

Figura 21- Esquema dos passos da solução proposta

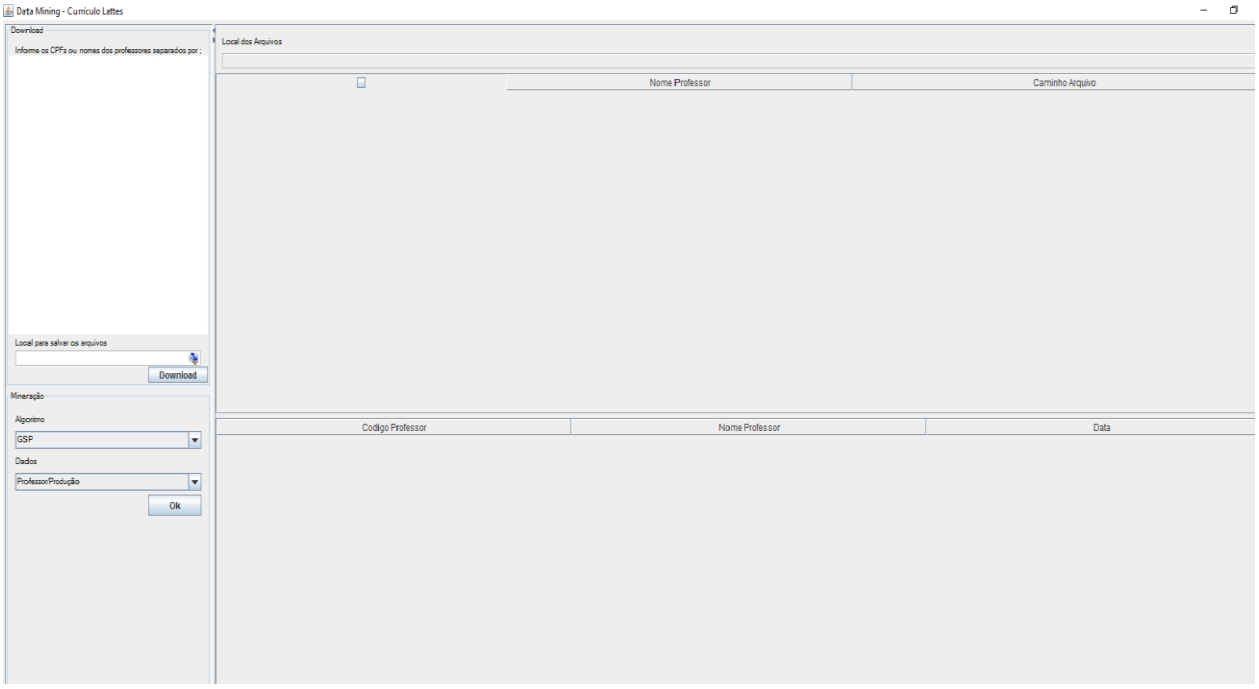


Fonte: Do autor

Os algoritmos a serem executados e os dados sobre os quais eles serão aplicados poderão ser definidos pelo usuário a partir de opções predefinidas na criação da ferramenta. O conhecimento gerado com a aplicação dos algoritmos será estruturado e exibido ao usuário de forma textual e/ou gráfica.

A figura 22 mostra a tela principal do sistema desenvolvido com a separação das operações que podem ser executadas no sistema.

Figura 22 - Tela Principal do Sistema

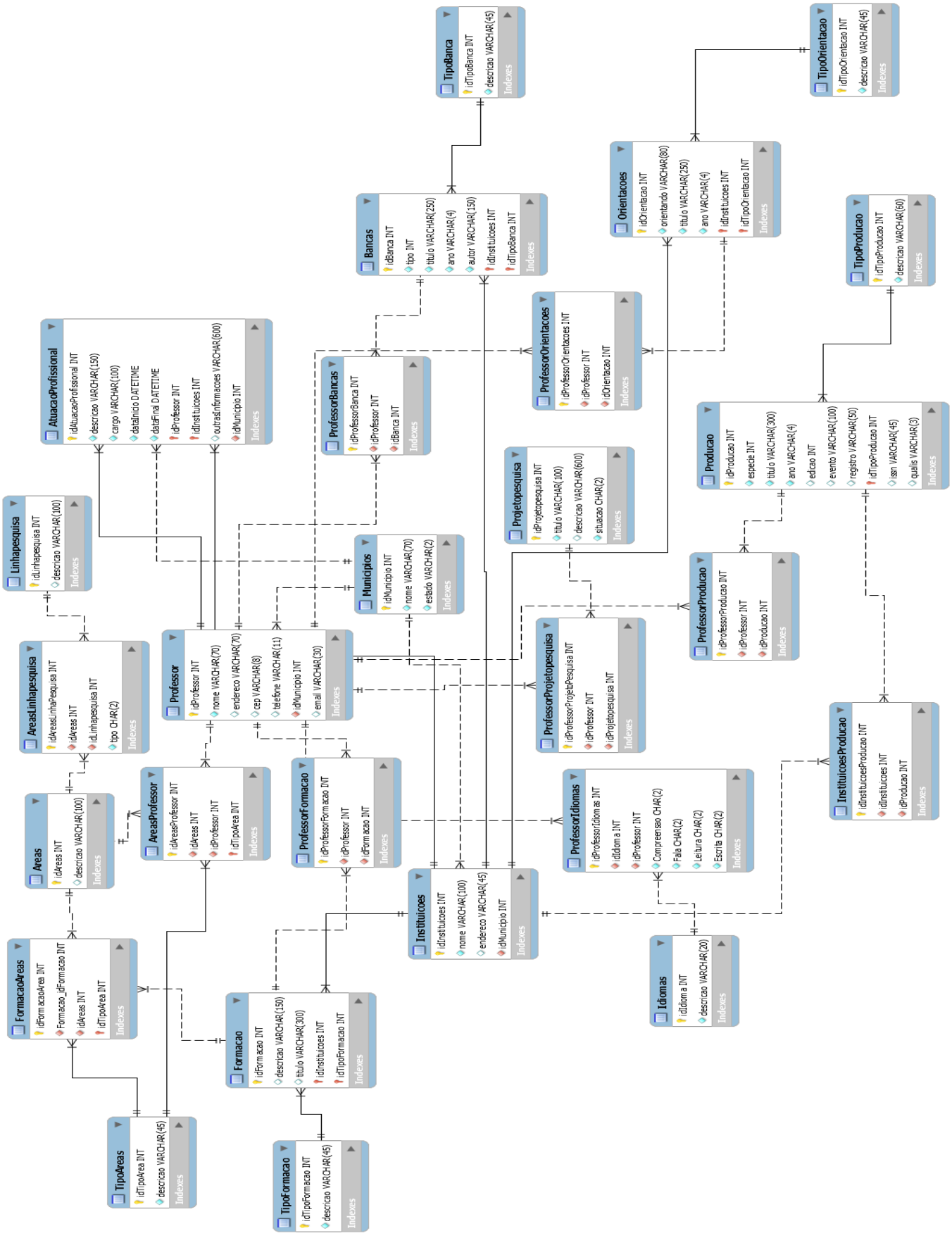


Fonte: Do Autor

7.2 Modelo de Banco de Dados

Na figura 23 está demonstrado o modelo do banco de dados construído para armazenar os dados extraídos dos currículos Lattes.

Figura 23 - Modelo de Banco de Dados



Fonte: Do Autor

7.3 Funcionalidades

7.3.1 Download dos Currículos

O sistema desenvolvido permite o download dos currículos Lattes dos professores através de seus CPFs ou de seus nomes. Na interface destinada a este fim podem ser informados os dados de diversos professores para a realização do download de todos os currículos em um mesmo processo e também pode ser definido o local onde os arquivos dos currículos deverão ser salvos.

7.3.2 Processo de extração e carga dos dados na Base de Dados

No sistema desenvolvido existe a opção de carregar arquivos XMLs dos currículos para serem lidos, estruturados e salvos na base de dados previamente construída. Na interface desenvolvida para este processo existe a opção de selecionar o local onde estão salvos os arquivos, será realizada uma verificação no formato e estrutura dos arquivos para verificar quais são os arquivos dos currículos e os dados destes arquivos são carregados em uma tabela onde existe a possibilidade de selecionar os arquivos que serão lidos e estruturados para serem salvos na base de dados. Para facilitar a identificação dos arquivos uma das informações carregadas nesta tabela de seleção é o nome dos professores. Nesta mesma interface existe uma tabela de consulta na qual estarão disponíveis os últimos currículos importados ou atualizados com o nome do professor e a data da última atualização.

A verificação da estrutura e a leitura dos arquivos são realizadas usando como base um arquivo XML Schema Definition (XSD) que é um arquivo que define o formato padrão que um arquivo XML deve seguir. Nesse arquivo são indicados os nodos principais do arquivo XML na tag element, os subnodos e atributos destes nodos principais indicando, quando necessário, o número mínimo e máximo de ocorrências do elemento ou atributo. Também podem existir neste arquivo de esquema o tipo de dado dos atributos e se a existência desses atributos é obrigatória ou opcional no arquivo XML que deve seguir o esquema.

O arquivo XSD também foi utilizado para geração automática das classes Java utilizadas inicialmente para a leitura dos dados e para que eles possam ser manipulados nas demais classes do sistema. Um exemplo de arquivo XSD está demonstrado na figura 24.

Figura 24 - Exemplo de arquivo XSD

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
           elementFormDefault="qualified"
           attributeFormDefault="unqualified">
  <xs:element name="CURRICULO-VITAE">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="DADOS-GERAIS"/>
        <xs:element minOccurs="0" ref="PRODUCAO-BIBLIOGRAFICA"/>
        <xs:element minOccurs="0" ref="PRODUCAO-TECNICA"/>
        <xs:element minOccurs="0" ref="OUTRA-PRODUCAO"/>
        <xs:element minOccurs="0" ref="DADOS-COMPLEMENTARES"/>
      </xs:sequence>
      <xs:attribute name="SISTEMA-ORIGEM-XML" use="required"/>
      <xs:attribute name="NUMERO-IDENTIFICADOR"/>
      <xs:attribute name="FORMATO-DATA-ATUALIZACAO" default="DDMMAAAA">
        <xs:simpleType>
          <xs:restriction base="xs:NMTOKEN">
            <xs:enumeration value="DDMMAAAA"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name="DATA-ATUALIZACAO"/>
      <xs:attribute name="FORMATO-HORA-ATUALIZACAO" default="HHMMSS">
        <xs:simpleType>
          <xs:restriction base="xs:NMTOKEN">
            <xs:enumeration value="HHMMSS"/>
          </xs:restriction>
        </xs:simpleType>
```

Fonte: Plataforma Lattes

7.3.3 Aplicação de algoritmos de Data Mining

Existe no sistema a possibilidade de utilização de algoritmos de Data Mining sobre os dados salvos. O algoritmo disponibilizado, para aplicação sobre os dados, é o algoritmo de descoberta de sequências temporais Generalized Sequential Patterns (GSP), que pode ser aplicado sobre os dados das produções científicas, tecnológicas e bibliográficas dos professores para que através da classificação Qualis de suas produções atuais seja possível prever as prováveis classificações de suas publicações futuras. Neste trabalho estão sendo utilizadas as classificações Qualis definidas pela CAPES (A1, A2, B1, B2, B3, B4, B5, C) e a classificação Não Classificado (NC) para as produções que não possuem classificação Qualis, transformadas para números inteiros de 1 a 9 que são aceitos para a execução do algoritmo para prever as classificações das produções futuras.

Os dados da classificação Qualis são pesquisados a partir do código International Standard Serial Number (ISSN) dos periódicos onde foram publicados as produções dos professores, usando como base a última versão da classificação existente, Qualis 2014.

O conhecimento gerado pela utilização do algoritmo de Data Mining é apresentado de forma clara e de fácil entendimento ao usuário do sistema para que ele possa tomar decisões com base nas informações encontradas.

O sistema também possui pronta a base para a implementação e uso de algoritmos de clusterização e descoberta de associações como por exemplo o algoritmo Apriori.

Para a aplicação dos algoritmos está sendo utilizada a ferramenta WEKA, desenvolvida pela Universidade de Waikato, na Nova Zelândia e que possui a implementação de diversos algoritmos de técnicas de Data Mining, principalmente para as técnicas de Classificação, Clusterização e Descoberta de Associações. Para o correto funcionamento do algoritmo foi necessário instalar na ferramenta WEKA as extensões timeSeriesFilters e timeSeriesForecasting no menu Tools > Package Manager.

A utilização da ferramenta WEKA está sendo realizada de forma integrada ao sistema desenvolvido por meio da API Java disponibilizada pela ferramenta.

7.4 Casos de Uso

Neste item estão descritos casos de uso das operações que podem ser realizadas no sistema desenvolvido.

7.4.1 Downloads dos arquivos XML dos currículos

Tabela 1 - Caso de Uso - Download dos XMLs dos currículos

Descrição	Downloads dos XMLs dos currículos
Precondição	Possuir os CPFs e/ou nomes dos professores
Fluxo Básico	1 - O usuário informa os CPFs e/ou nomes dos professores no campo específico separados por ponto e vírgula (;). 2 - O usuário informa o local onde devem ser salvos os arquivos XML. 3 - O usuário confirma a operação de download dos arquivos. 4 - O sistema executa o processo de download e salvamento dos arquivos. 5 - O sistema finaliza o processo e exibe mensagem de execução com sucesso para o usuário.
Fluxo Alternativo	A1 - Alternativa ao passo 4 - Ocorre erro no download ou no salvamento dos arquivos. 1.a - Ocorre erro no download ou no salvamento dos arquivos. 2.a - O sistema para o processo de download ou salvamento dos arquivos. 3.a - O sistema exibe mensagem indicando o erro ocorrido ao usuário.

7.4.2 Extração e estruturação dos dados dos currículos

Tabela 2 - Caso de Uso - Extração e estruturação dos dados

Descrição	Extração e estruturação dos dados dos currículos
Precondição	Possuir arquivos XML dos currículos
Fluxo Básico	1 - O usuário informa o local onde estão salvos os arquivos dos currículos. 2 - O sistema verifica os arquivos e carrega os dados na tabela. 3 - O usuário seleciona os arquivos dos quais deseja extrair os dados. 4 - O usuário confirma a operação de extração dos dados. 5 - O sistema executa a extração dos dados e popula as tabelas da base de dados. 6 - O sistema finaliza o processo e exibe mensagem de execução com sucesso para o usuário.
Fluxo Alternativo	A1 - Alternativa ao passo 5 - Ocorre erro na extração dos dados dos arquivos. 1.a - Ocorre erro no processo de extração dos dados dos arquivos. 2.a - O sistema interrompe o processo de extração dos dados. 3.a - O sistema exibe mensagem indicando o erro ocorrido ao usuário.

7.4.3 Aplicação dos algoritmos de Data Mining / Consulta

Tabela 3 - Caso de Uso - Aplicação dos algoritmos de Data Mining / Consulta

Descrição	Aplicação dos algoritmos de Data Mining / Consulta
Precondição	Ter executado a extração de dados e possuir dados na base de dados
Fluxo Básico	1 - O usuário seleciona qual o algoritmo de Data Mining que será aplicado. 2 - O usuário seleciona os dados sobre os quais será feita a aplicação do algoritmo. 3 - O usuário confirma a aplicação do algoritmo selecionado sobre os dados. 4 - O sistema estrutura os dados no formato necessário para a aplicação do algoritmo por meio da ferramenta WEKA. 5 - O sistema realiza a aplicação do algoritmo sobre os dados através da ferramenta WEKA. 6 - O sistema captura os resultados obtidos da aplicação do algoritmo sobre os dados. 7 - O sistema estrutura os dados dos resultados obtidos e os exibe para o usuário de forma textual e/ou gráfica.
Fluxo Alternativo	A1 - Alternativa ao passo 4 - Ocorre erro na estruturação dos dados 1.a - Ocorre erro na estruturação dos dados para a aplicação do algoritmo 2.a - O sistema interrompe o processo de aplicação do algoritmo. 3.a - O sistema exibe mensagem indicando o erro ocorrido ao usuário. A2 - Alternativa ao passo 5 - Ocorre erro na aplicação do algoritmo sobre os dados 1.a - Ocorre erro na aplicação do algoritmo sobre os dados. 2.a - O sistema interrompe o processo de aplicação do algoritmo.

	3.a - O sistema exibe mensagem indicando o erro ocorrido ao usuário.

7.5 Ferramentas e Softwares utilizados

O sistema foi totalmente desenvolvido na linguagem de programação Java e foi utilizado o framework Hibernate para realizar o mapeamento entre as classes Java e as tabelas da base de dados com o uso de arquivos XML do tipo Hibernate Mapping (HBM) e as consultas aos dados estão sendo realizadas sobre as classes Java com o uso da linguagem Hibernate Query Language (HQL) permitindo portabilidade de banco de dados.

Para a construção da base de dados e armazenamento dos dados extraídos dos currículos Lattes foi utilizado um banco de dados relacional Sistema de Gerenciamento de Banco de Dados (SGBD) MySql.

7.5.1 Softwares utilizados

Neste item estão informações sobre os softwares que foram utilizados no desenvolvimento deste trabalho.

- Eclipse: Eclipse é um Integrated Development Environment (IDE) para desenvolvimento Java, porém suporta várias outras linguagens a partir de plugins como C/C++,PHP,ColdFusion, Python, Scala e plataforma Android. O software Eclipse tem a licença Eclipse Public License (EPL) e é atualmente o IDE Java mais utilizado no mundo. No desenvolvimento deste trabalho foram utilizadas as versões Juno e Neon deste IDE.
- SQLYog Community: é um programa desenvolvido pela WEByog Enterprise que possibilita a edição de bancos de dados Mysql, baseados na linguagem SQL. Utilizado na criação, edição, sincronização de banco de dados internos e em servidores. É disponibilizado em duas versões, Enterprise, paga, e Community, gratuita e de código aberto. Neste trabalho foi utilizada a versão SQLYog Community v 10.0 para realização de inserção e consulta dos dados da base de dados.
- MySql Workbench 6.3 CE: é uma ferramenta visual de design de banco de dados que integra desenvolvimento de SQL, com criação, administração, manutenção e design de banco de dados em um único ambiente de desenvolvimento integrado para o sistema de banco de dados MySQL. O software foi utilizado neste trabalho para realizar a

modelagem da base de dados para o armazenamento dos dados extraídos dos currículos Lattes.

- Waikato Environment for Knowledge Analysis (WEKA): ferramenta desenvolvida pela Universidade de Waikato, na Nova Zelândia e que possui a implementação de diversos algoritmos de técnicas de Data Mining, principalmente para as técnicas de Classificação, Clusterização e Descoberta de Associações.

7.6 Testes e validações

Inicialmente foram realizados testes com pequenos grupos de professores para verificar o correto funcionamento do sistema e do algoritmo de mineração. Para as validações do sistema e do algoritmo foram realizadas duas fases de testes na extração dos dados dos currículos e aplicação do algoritmo GSP.

Na primeira dessas fases foram utilizados os dados dos currículos dos onze professores do Programa de Pós Graduação em Sistemas e Processos Industriais (PPGSPI) da UNISC, os dados dos professores podem ser encontrados em: http://www.unisc.br/pt/cursos/todos-os-cursos/mestrado-doutorado/mestrado/mestrado-em-sistemas-e-processos-industriais/corpo-docente-ppgspi.

Na segunda fase foram realizados testes com uma quantidade maior de dados, para esses testes foram utilizados os dados de quarenta e dois professores do Programa de Pós Graduação em Computação (PPGC) da Universidade Federal do Rio Grande do Sul (UFRGS).

Os dados da classificação Qualis são coletados usando como base a última versão da classificação existente, Qualis 2014 através do código ISSN dos periódicos. Esses dados são adicionados manualmente na base de dados.

Para realizar a previsão o algoritmo trabalha com dados numéricos e por esse motivo as classificações Qualis definidas pela CAPES (A1, A2, B1, B2, B3, B4, B5, C) e a classificação Não Classificado (NC) usada para as produções que não possuem classificação Qualis, são transformadas para números inteiros de 1 a 9 que são aceitos para a execução do algoritmo.

Para a correta execução do algoritmo é necessário que os registros possuam data e hora completa, como as produções no currículo Lattes possuem apenas o ano de publicação inicialmente é realizado um processamento sobre os dados para que a variável de data possua um formato aceito pelo algoritmo.

Nesses testes o nível de confiança dos resultados foi definido como 90% e os resultados destes testes com a previsão do nível da classificação Qualis das produções futuras dos professores da UNISC estão na tabela 4 e dos professores da UFRGS na tabela 5.

Tabela 4 - Resultados dos testes UNISC

Professor	Qualis Últimas Produções	Qualis Produções Futuras
Professor 1	NC, A2, A2, A2, B3, B5	B3, B2, B2
Professor 2	B4, A1, B1, B3	B2, B2, B2
Professor 3	NC, NC, NC, NC, NC, NC	NC, B5, B4
Professor 4	B5, B5, B2, C, C, C	B4, B4, B4
Professor 5	C, B2, B4, C, NC, NC	B5, B5, B5
Professor 6	A1, B3, B4, C, NC, NC	B4, B4, B4
Professor 7	B4, B5, B5, C, NC, NC	B5, B4, B4
Professor 8	NC, B4, B4, NC, B5, C	B5, B5, C
Professor 9	B5, B5, C, NC, B5, NC	B5, B5, B5
Professor 10	C, NC, B1, B5, NC, NC	B5, B5, B5
Professor 11	B5, C, A2, B5, B5, A1	B3, B4, B5

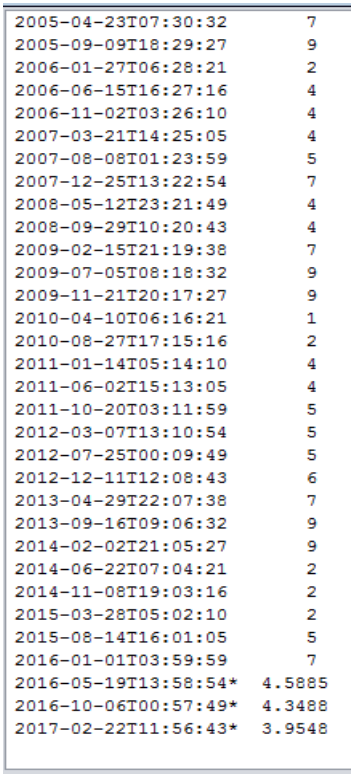
Tabela 5 – Resultados dos testes UFRGS

Professor	Qualis Últimas Produções	Qualis Produções Futuras
Professor 1	NC, A1, A2, A1, B2, A2	B1, B1, B1
Professor 2	C, A1, NC, NC, C, C	B5, C, B5
Professor 3	NC, NC, A1, A2, NC, B1	B5, C, A1
Professor 4	B2, NC, B2, A2, NC, A1	B4, C, A2
Professor 5	B4, A2, NC, B4, A2, A2	B4, B4, A2
Professor 6	B2, B1, B1, NC, NC, B1	B1, B1, B1
Professor 7	NC, B1, A2, B1, A2, A1	B3, A2, B3
Professor 8	NC, B4, B2, B5, NC, A2	B3, B3, B4
Professor 9	A2, C, A1, A1, NC, NC	B3, NC, B4
Professor 10	B5, B4, B1, C, B1, NC	B1, NC, C
Professor 11	C, A2, B4, C, B1, B2	C, B1, C
Professor 12	B1, A2, B1, A2, A1, NC	B3, B2, B3
Professor 13	C, B1, A2, B1, NC, A2	B1, B2, B4
Professor 14	B1, A2, B1, B1, B1, B1	A2, B1, B1
Professor 15	NC, B5, B1, A2, B2, NC	B4, B2, B4
Professor 16	B3, A2, B1, A2, B2, B2	B1, B1, B1
Professor 17	A2, A2, NC, B1, B1, NC	B1, B1, A2
Professor 18	NC, A2, B1, A1, A1, NC	B3, B2, A2
Professor 19	NC, NC, C, B5, NC, B1	B5, NC, B3
Professor 20	A2, A2, B1, B1, A2, A1	B3, A2, B1
Professor 21	NC, A2, B1, A1, A2, A1	A2, A1, A2
Professor 22	A2, B2, B2, NC, B2, B1	B3, B2, B3
Professor 23	NC, C, A2, A2, NC, NC	B4, B4, B4
Professor 24	B1, B1, A2, NC, C, A1	B4, B4, B5
Professor 25	A2, A2, B2, A1, B3, A2	B1, B1, A2
Professor 26	B1, B1, C, B1, A1, B2	B1, B1, A2
Professor 27	NC, B1, NC, NC, B2, A2	B4, NC, B4
Professor 28	A2, A2, B1, A2, A1, A2	A2, A2, B1
Professor 29	A2, A2, B1, A2, A2, A2	B4, A2, A1
Professor 30	B5, A2, B5, A2, B1, B5	B4, B4, B5

Professor 31	A2, A2, A2, B3, B3, B1	B2, B1, B1
Professor 32	B1, NC, A2, B3, B1, A1	B2, B2, B2
Professor 33	B1, B1, B2, A1, B1, A1	A1, A1, B1
Professor 34	B1, A2, B2, NC, NC, NC	A2, B3, B4
Professor 35	B4, NC, B1, C, C, A1	C, B1, B4
Professor 36	B1, NC, B1, A2, NC, A1	B3, B3, B1
Professor 37	A2, NC, B2, A2, A2, A2	B1, B1, B2
Professor 38	NC, NC, B1, C, A2, A2	B4, B4, B2
Professor 39	B2, NC, A2, B1, A2, A2	B2, B2, B2
Professor 40	A2, B1, B1, NC, NC, B1	A1, B4, NC
Professor 41	B2, B1, B1, B1, B5, B3	B2, B1, B2
Professor 42	NC, A2, B1, B2, B2, C	B2, B3, B2

Os resultados exibidos pela execução do algoritmo não são exatos e foram arredondados para definir a classificação Qualis das produções futuras. O resultado do algoritmo é gerado da forma como está demonstrado na imagem 25. Nesta imagem estão as datas, já transformadas para o formato aceito, das produções de um professor e a classificação Qualis, também já transformada para o formato aceito, destas produções. E nas últimas três linhas estão os resultados de previsão das classificações das próximas produções do professor.

Figura 25 - Resultados da execução do algoritmo



Fonte: Do autor

8 CONSIDERAÇÕES FINAIS

Atualmente existe uma dificuldade das universidades de possuir um controle referente à produção bibliográfica, científica e tecnológica de seus professores e de ter a possibilidade de utilizar esses dados para auxiliar em alguma tomada de decisão, devido a grande quantidade de departamentos e cursos existentes e aos mais diversos eventos e publicações em que os professores podem apresentar e publicar seus trabalhos e artigos.

Muitos desses dados podem ser encontrados nos currículos Lattes dos professores existentes na Plataforma Lattes, base de dados de currículos e grupos de pesquisa mantida pelo CNPq e que possui diversos dados referentes à produção bibliográfica, científica e tecnológica de professores e pesquisadores.

Para a extração, estruturação e processamento desses dados podem ser utilizadas técnicas de Descoberta de Conhecimento em Bases de Dados, principalmente Mineração de Dados que é o principal tópico estudado neste trabalho com foco em algoritmos de Clusterização, para agrupamento de dados similares em bases de dados, e de Regras de Associação para localizar relações entre ocorrências simultâneas de conjuntos de dados em transações de bancos de dados.

Neste trabalho foi desenvolvida uma solução para essa dificuldade das universidades controlarem a produção dos professores por meio de uma ferramenta que realizará a extração dos dados dos currículos Lattes dos professores, realizando o armazenamento dos dados coletados em uma base de dados e posteriormente aplicar sobre os dados algoritmos de Clusterização e Regras de Associação para extrair conhecimento útil para tomadas de decisão que poderão ser estruturados e visualizados através de consultas realizadas pelos usuários em uma interface própria para esse fim.

8.1 Sugestões de trabalhos futuros

A partir deste trabalho, diversos outros podem ser desenvolvidos como, por exemplo:

- Adição de novos algoritmos: podem ser incluídos novos algoritmos de mineração de dados no sistema para serem aplicados sobre outros dados dos currículos Lattes e gerarem mais informações e conhecimentos úteis para os gestores das universidades.

- Cruzamento dos resultados com informações de outros sistemas: pode ser realizado o cruzamento dos dados com dados de outros sistemas para obter novas informações relevantes.
- Permitir a modelagem da base de dados e das consultas: permitir que o usuário do sistema construa a sua própria base de dados para salvar apenas os dados que necessitar dos currículos Lattes, e também permitir a montagem de consultas indicando de forma mais livre os dados que serão utilizados e as informações que se deseja encontrar.
- Automatizar a inclusão dos dados Qualis: criar opção no sistema para carregar automaticamente os dados de classificação Qualis das produções a partir do código ISSN.

REFERÊNCIAS

ALMEIDA, Maurício B.; BARACHO, Renata Maria A.; BRANQUINHO, Lucélia P. Descoberta de conhecimento com uso de ontologias na mineração de dados. Disponível em: http://periodicos.pucminas.br/index.php/abakos/article/view/P.2316-9451.2015v4n1p20/8779. Acesso em: 11 maio 2016

ARAÚJO, Rafael L. P. de; CUNHA, Fabrício Rangel. Uso de Técnicas de Data mining no Monitoramento de Alunos On-line. Disponível em: www.e-publicacoes.uerj.br/index.php/cadinf/article/download/6388/4551. Acesso em: 10 maio 2016

ARBEX, Márcio A.; COSTA, Claudio N.; COUTINHO, Jonatas V.; MAGALHÃES, Lúcia H. Descberta de Conhecimento em Bases de Dados. Disponível em: http://fsd.edu.br/revistaeletronica/arquivos/2Edicao/artigo9.pdf. Acesso em: 02 maio 2016

AZEVEDO, Carla S.; SANTOS, Manuel Filipe. Data mining: descoberta de conhecimento em bases de dados. 1ed. Lisboa. FCA, 2005.

AZEVEDO, Ryan R. de; DANTAS, Eric Rommel G.; LIMA, Daniel S. de; PATRÍCIO JUNIOR, José Carlos A. O Uso da Descoberta de Conhecimento em Base de Dados para Apoiar a Tomada de Decisões. In: V SIMPÓSIO DE EXCELÊNCIA EM GESTÃO E TECNOLOGIA, 2008, Rezende. Anais... Rezende, 2008. Disponível em: http://www.aedb.br/seget/arquivos/artigos08/331_331_Artigo_SEGET_EJDR_Versao_Final_010808.pdf. Acesso em: 14 maio 2016

BARBOSA, Ciro Batos. Explorando Técnicas de Redução de Base de Dados na Mineração de Padrões Sequenciais. 2006. 80 f. Dissertação (Programa de Pós-Graduação em Computação da Universidade Federal Fluminense-Mestrado)-Universidade Federal Fluminense, Niterói, 2006.

BUENO, Michel F.; VIANA, Maury R. Mineração de Dados: Aplicações, Eficiência e Usabilidade. In: CONGRESSO DE INICIAÇÃO CIENTIFÍCA DO INATEL-INCITEL, 2012, Santa Rita do Sapucaí. Anais... Santa Rita do Sapucaí, 2012. Disponível em: http://www.inatel.br/ic/component/docman/doc_download/65-mineracao-de-dados-aplicacoes--eficiencia-e-usabilidade. Acesso em: 11 maio 2016

CALAPEZ, Marco M. Guerreiro. SISMAC: Sistema de Monitoria de Alterações Climáticas. 2008. 78 f. Dissertação (Engenharia Informática e de Computadores-Mestrado)-Instituto Superior Técnico-Universidade Técnica de Lisboa, Lisboa, 2008.

CAMILO, Cássio O.; SILVA, João Carlos da. Mineração de Dados: Conceitos, Tarefas, Métodos e Ferramentas. 2009. Disponível em: http://www.inf.ufg.br/sites/default/files/uploads/relatorios-tecnicos/RT-INF_001-09.pdf. Acesso em: 29 abr. 2016

CARLANTONIO, Lando Mendonça di. Novas Metodologias para Clusterização de Dados. 2001. 157 f. Tese (Programas de Pós Gradução em Engenharias)-Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2001.

CARVALHO, Cedric Luiz de; VASCONCELOS, Lívia Maria R. de. Aplicação de Regras de Associação para Mineração de Dados na Web. Disponível em: http://www.inf.ufg.br/sites/default/files/uploads/relatorios-tecnicos/RT-INF_004-04.pdf. Acesso em: 29 abr. 2016

CARVALHO, Hélio Gomes de. Inteligência Competitiva Tecnológica para PMEs através da Cooperação Escola-Empresa: Proposta de um Modelo. 2000. 333 f. Tese (Programa de Pós Graduação em Engenharia de Produção - Doutorado)-Universidade Federal de Santa Catarina, Florianópolis, 2000.

CARVALHO, Juliano V. de; MONGIOVI, Giuseppe; SAMPAIO, Marcus C. Utilização de Técnicas de "Data Mining" para o Reconhecimento de Caracteres Manuscritos. Disponível em: http://www.dsc.ufcg.edu.br/~sampaio/Artigos/reconhecimentocaracteresmanuscritos.pdf. Acesso em: 03 maio 2016

CHALCO, Jesús M.; MAGALHÃES, Jorge; QUONIAM, Luc; SANTOS, André. Extração e Tratamento de Dados na Base Lattes para Identificação de Core Competencies em Dengue. Disponível em: http://www.uel.br/revistas/uel/index.php/informacao/article/view/17679/pdf_32. Acesso em: 03 maio 2016

DALFOVO, Michael Samir; LANA, Rogério Adilson; SILVEIRA, Amélia. Métodos quantitativos e qualitativos: um resgate teórico. Revista Interdisciplinar Científica Aplicada, Blumenau, v. 2, n. 4, p. 01-13, Sem. II, 2008. Disponível em: http://www.unisc.br/portal/upload/com_arquivo/metodos_quantitativos_e_qualitativos_um_resgate_teorico.pdf. Acesso em: 28 maio 2016

DANIEL, Luiz Antonio; MEDEIROS, Gerson A. de; MORAES, Luciana de M.; TOMAZELA, Maria das Graças J. M. Aplicação de Técnicas de Mineração de Dados para Caracterização de Grupos de Cidades Produtoras de Cana-De- Açúcar do Estado de São Paulo e Definição de Políticas Especificas. Disponível em: http://fatecid.com.br/reverte/index.php/revista/article/view/48/51. Acesso em: 09 maio 2016

DIAS, Carlos Rodrigo; OCHI, Luiz S.; SOARES, Stênio S. F. Clusterização em Mineração de Dados. Disponível em: http://www2.ic.uff.br/~satoru/conteudo/artigos/ERI-Minicurso-SATORU.pdf. Acesso em: 13 maio 2016

ELMASRI, Ramez; NAVATHE, Shamkant B. Sistemas de Banco de Dados. 4 ed. São Paulo. Pearson Addison Wesley, 2005.

FARIAS JUNIOR, Euclides P. Estudo Comparativo entre Algoritmos de Regras de Associação de Forma Normal e Incremental de Dados. 2008. 131 f. Dissertação (Programa de

Pós Graduação em Informática da Pontifícia Universidade Católica do Paraná-Mestrado)- Pontifícia Universidade Católica do Paraná, Curitiba, 2008.

GERHARDT, Tatiana E.; SILVEIRA, Denise T. Métodos de Pesquisa. 1 ed. Porto Alegre. Editora da UFRGS, 2009.

GOLDSCHMIDT, Ronaldo; PASSOS, Emmanuel. Data Mining: Um Guia Prático. 1 ed. Rio de Janeiro. Elsevier, 2005.

LOBO, Fernando; RAMOS, Célia. Descoberta de Conhecimento em Bases de Dados. P. 53 – 59. Disponível em: https://www.academia.edu/1501519/Descoberta_de_Conhecimentos_em_Base_de_Dados?auto=download. Acesso em: 20 abr. 2016

MACEDO, Dayana Carla; MATOS, Simone Naser. Extração de Conhecimento através da Mineração de Dados. Revista de Engenharia e Tecnologia. v. 2 n. 2, p. 22 – 30, ago. 2010. Disponível em: http://pg.utfpr.edu.br/dirppg/ppgep/ebook/2010/PERIODICOS/Revista_de_Engenharia_e_Tecnologia/2.pdf . Acesso em: 04 maio 2016

MORAIS, Bruno Carlos S. de. Extração de Conhecimento da Plataforma Lattes Utilizando Técnicas de Mineração de Dados: Estudo de Caso POLI/UPE. 2010. 54 f. Trabalho de Conclusão de Curso (Engenharia da Computação)- Universidade de Pernambuco, Recife, 2010.

MOSCATO, Pablo A.; VON ZUBEN, Fernando J. Uma Visão Geral de Clusterização de Dados. Disponível em: ftp://ftp.dca.fee.unicamp.br/pub/docs/vonzuben/ia368_02/topico5_02.pdf. Acesso em: 13 maio 2016

NAVEGA, Sergio. Princípios Essenciais do Data Mining. In: INFOIMAGEM, 2002, São Paulo. Anais... São Paulo: CENADEM, 2002. Disponível em: http://docplayer.com.br/1201647-Principios-essenciais-do-data-mining.html. Acesso em: 05 maio 2016

OLIVEIRA, Tatyana B. S. de. Clusterização de dados utilizando técnicas de redes complexas e computação bioinspirada. 2008. 112 f. Dissertação (Mestrado em Ciências de Computação e Matemática Computacional)-Instituto de Ciências Matemáticas e de Computação/Universidade de São Paulo, São Carlos, 2008.

PUC-Rio. Clusterização de Dados. Disponível em: http://www.maxwell.vrac.puc-rio.br/24787/24787_5.PDF. Acesso em: 11 maio 2016

SANTOS, Anderson; WILKENS, Rodrigo. Mineração de Padrões Sequenciais. Disponível em: www.inf.ufrgs.br/~alvares/CMP259DCBD/MineracaoPadraoSequencial.ppt. Acesso em: 07 nov. 2016

SETZER, Valdemar W. Dado, Informação, Conhecimento e Competência. 2015. Disponível em: https://www.ime.usp.br/~vwsetzer/dado-info.html. Acesso em: 28 abr. 2016

SILVA, Glauco Carlos. Mineração de Regras de Associação Aplicada a Dados da Secretaria Municipal de Saúde de Londrina – PR. 2004. 94 f. Dissertação (Programa de Pós Graduação em Ciência da Computação-Mestrado)-Universidade Federal do Rio Grande do Sul, Porto Alegre, 2004.

VANZIN, Mariângela. Mecanismos de Apoio a Interpretação e Recuperação de Padrões do Uso da Web Baseados em Ontologia de Domínio. 2004. 150 f. Dissertação (Programa de Pós Graduação em Ciência da Computação - Mestrado)-Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, 2004.