CURSO DE CIÊNCIA DA COMPUTAÇÃO

Vanessa da Silva Soares

MINERAÇÃO DE TEXTOS PARA IDENTIFICAR PERFIS DE SATISFAÇÃO DE CLIENTES

Santa Cruz do Sul, RS – Brasil
2016

Vanessa da Silva Soares

MINERAÇÃO DE TEXTOS PARA IDENTIFICAR PERFIS DE SATISFAÇÃO DE CLIENTES

Trabalho de Conclusão apresentado ao Curso de Ciência da Computação da Universidade de Santa Cruz do Sul.

Orientador: Eduardo Kroth

Santa Cruz do Sul, RS – Brasil
2016

RESUMO

Muitas empresas estão cada vez mais interessadas em descobrir meios adequados e criativos de garantir um atendimento de qualidade a clientela em geral. Mais do que produzir produtos e serviços de qualidade superior, uma organização deve manter satisfações superiores em todos os seus relacionamentos. Satisfazer o cliente é o caminho certo para se atingir um bom índice de desenvolvimento, portanto, criar estratégias para atrair mais clientes é uma forma viável de alcançar o sucesso no âmbito comercial. Para isso, muitos gestores têm utilizado recursos para mensurar a qualidade de seu serviço, porém apenas pesquisas de satisfação não tem sido suficiente, pois muitos clientes não emitem a insatisfação. Este trabalho visa identificar através dos atendimentos prestados aos clientes, o grau de satisfação atingido pelo cliente. Desta forma os gestores das empresas saberão onde direcionar suas energias afim de melhorar seus atendimentos. Para fundamentação teórica deste trabalho foram verificados conceitos de satisfação de clientes para compreender melhor suas expectativas. Para análise de classificação do atendimento e nível de satisfação do atendimento, o conteúdo de mineração de textos foi extremamente essencial.

Palavras chave: Satisfação de clientes. Mineração de Textos.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

KDD – Descoberta de conhecimento de dados

KDD – Descoberta de conhecimento de textos

UNISC – Universidade de Santa Cruz do Sul

CINTED – Centro Interdisciplinar de Novas Tecnologias em Educação

UFRGS – Universidade Federal Rio Grande do Sul

Sumário

1. INTRODUÇÃO

O mercado de software está cada vez mais competitivo, as empresas precisam dia a dia se destacar para conquistar seus clientes, e somente as ferramentas de desenvolvimento de software não são mais suficientes para manter a empresa com um padrão de excelência, é preciso outros sistemas para auxiliar a equipe a manter a qualidade e atender as expectativas dos clientes.

Diariamente as empresas armazenam uma massa muito grande de informações produzidas por seus colaboradores e clientes através do meio eletrônico, são inúmeras páginas de textos, históricos da empresa, relatórios, índices e estatísticas com informações precisas e outras nem tanto. Estes dados, se classificados, processados e analisados, podem auxiliar na tomada de decisão, apresentando informações úteis para as empresas que almejam aumentar a qualidade do serviço prestado ao cliente. Claramente, tal manancial de informações não somente influi direta e indiretamente nos mercados, como nos meios de vigilância já em uso pelos estados nacionais e suas instituições, produzindo registros e meta informação sobre usuários e indivíduos (LOPES,2004; PIMENTA, 2013).

A procura pela excelência da satisfação de clientes com relação aos produtos e serviços, a partir dos anos oitenta tornou-se uma das principais preocupações dos gestores empresariais. Entretanto, a medição da qualidade de produtos é mais fácil do que de serviços, isto porque há um diferencial entre os dois, onde os serviços não são mensuráveis. Esta qualidade do serviço depende fortemente da percepção de seu consumidor, posto que este compara o que lhe foi entregue com suas expectativas acerca do que foi contratado.

Dados citados por Reichheld & Sasser (REICHHELD e SASSER,1990) mostraram que na média das organizações ocorre uma perda de 15 a 20% dos consumidores a cada ano, e na maior parte das vezes devido aos serviços prestados. E que as companhias podem aumentar em 100% seus lucros, em um ano, retendo apenas 5% a mais de seus clientes. Além disso clientes de longo prazo compram mais, tomam menos tempo da empresa, são menos sensíveis ao preço e trazem novos clientes, além de não possuírem custo de aquisição. Estes dados foram reafirmados por autores em um estudo sobre relacionamento com clientes, onde demonstram quão relevante é a gestão estratégica do relacionamento entre as

organizações e seus clientes (DEMO, 2014). Um estudo citado por Richard Whiteley revelou que a baixa qualidade dos serviços foi apontada como principal razão para a mudança para um competidor. Enquanto apenas 15% dos clientes mudaram por terem encontrado um produto melhor e 15% por terem encontrado um produto mais barato, verificou-se que 20% mudaram pela falta de contato e atenção pessoal e 49% por terem recebido um atendimento de baixa qualidade (RH, 2015).

Portanto, é prudente recomendar aos profissionais que atuam em áreas dessa natureza, que invistam considerável esforço no sentido de detectar em suas atividades quais as lacunas existentes na sua forma de atuação ou na forma de atuação de sua empresa, e quais as expectativas dos seus contratantes. Após isso ter sido determinado, os gestores devem agir de imediato, ou, na pior das hipóteses, reduzi-las a ponto de tornar-se inócuos seus reflexos negativos, reflexos estes que poderão implicar na perda de seus clientes.

Em virtude desse crescimento contínuo do volume de dados eletrônicos disponíveis, técnicas de extração de conhecimento automáticas tornam-se cada vez mais necessárias para valorizar a gigantesca quantidade de dados armazenada nos sistemas de informação. Além disso, como as técnicas desenvolvidas para mineração de dados foram desenvolvidas para dados estruturados, técnicas específicas para mineração de textos têm sido aprimoradas para processar uma parte importante da informação disponível que pode ser encontrada na forma de dados não- estruturados.

Mineração de Dados é um ramo da computação que teve início nos anos 80, quando os profissionais das empresas e organizações começaram a se preocupar com os grandes volumes de dados informáticos estocados e inutilizados dentro da organização (MORAIS, 2007). Atualmente a Mineração de dados que consiste de técnicas de análise e extração de dados em uma grande base de informação é utilizada por exemplo, para levantar informações e necessidades reais e hipotéticas de cada cliente para realizar ações de marketing.

O processo de descoberta do conhecimento envolve a aplicação de algoritmos computacionais que processam e identificam informações úteis e implícitas, que normalmente não poderiam ser recuperadas utilizando métodos tradicionais de consulta, pois a informação contida não pode ser obtida de forma direta, uma vez que, em geral, estão armazenadas em formato não estruturado (AMO, 2016) .

Segundo Aranha (2006), pesquisas realizadas mostram que cerca de 80% do conteúdo *online* está em formato textual e dentro das empresas o mesmo percentual reflete a organização não estruturada dos dados.

Considerando as afirmações de Aranha (2006), é preciso além de analisar os dados considerar também os textos, para isso as aplicações de mineração de textos podem fornecer uma nova dimensão das informações disponíveis nas empresas, sendo utilizadas no acompanhamento da gerência de equipes e de clientes, com a modelagem de perfis de clientes baseado em históricos podendo alertar a empresa sobre situações pré-determinadas.

Em todos os segmentos e empresas, os clientes estão a cada dia mais exigentes, buscando a empresa com melhor qualidade dos serviços prestados, e em contrapartida as empresas estão trabalhando fortemente para disponibilizar um serviço diferenciado. Mas como atender as expectativas dos clientes, e quais ferramentas estão disponíveis para auxiliar os gestores?

A possibilidade de identificar o perfil dos clientes e possíveis problemas que este virá a enfrentar para resolvê-los com agilidade e precisão é extremamente atraente. O ganho de tempo na análise de informações, selecionando somente o que realmente interessa sobre o cliente para à empresa, suas opiniões, dúvidas e expectativas é o que a empresa precisa para conquistar ainda mais o cliente e ganhar o mercado.

Conforme índices de uma empresa que presta atendimento, por ano são mais de cem mil atendimentos, que podem ser de assuntos variados falando sobre situações e problemas que ocorrem no sistema, vezes é má utilização do software, falta de conhecimento, ou em outros casos alguma falha do programa. Além dos atendimentos, por ano são abertos mais de quinze mil tickets e ministrados mais de mil treinamentos *online* com seus clientes. Atualmente estes dados não podem ser classificados, nem por área ou situação do sistema, por isso a possibilidade de contabilizar e classificar estes atendimentos por recursos da plataforma, e uma ferramenta capaz de agrupar e classificar os atendimentos e clientes, de forma que possibilite uma posterior análise para tomada de decisão pode se tornar uma forma de aumentar a qualidade do serviço prestado. (SOLUTIONS, 2016)

1.1. Objetivos

O objetivo principal deste trabalho é desenvolver um sistema capaz de analisar e identificar o perfil de satisfação de um cliente baseado em um atendimento em formato textual.

Estas análises serão capazes de identificar as seguintes situações:

- Clientes que tiveram atendimentos com problemas recorrentes e que precisam de maior atenção;
- Problemas que ocorrem em mais clientes para possível solução pela equipe de desenvolvimento;
- Áreas do sistema que mais demandam suporte;
- Classificação do atendimento em relação a satisfação do cliente ao término;

1.2. Objetivos específicos

- Desenvolver um sistema de classificação de atendimentos;
- Realizar a análise a partir dos perfis visando à qualidade na prestação dos serviços.

1.3. Metodologia

Este trabalho se originou de uma pesquisa exploratória e de uma pesquisa explicativa, já além de envolver levantamento bibliográfico, conversas com especialistas e análise dos casos de insatisfação, foram necessários. A pesquisa tem uma natureza qualitativa, já que trata de métodos para melhorar a qualidade de atendimentos. Como tópicos detalhados da metodologia pode-se destacar:

- Pesquisar trabalhos relacionados sobre mineração de dados e textos;
- Analisar a forma manual de análise de históricos de atendimentos, para conhecer o funcionamento;
- Identificar perfis dos clientes;
- Efetuar testes para mensurar a qualidade da ferramenta desenvolvida;

2. REFERENCIAL TEÓRICO

Neste capítulo são apresentados os principais conceitos envolvidos no processo de descoberta de conhecimento em texto, bem como ferramentas que podem ser utilizadas nesse processo. Também são apresentados conceitos de satisfação de clientes.

2.1. Satisfação de Clientes

O nível de “satisfação” de clientes constitui uma das prioridades de gestão nas organizações comprometidas com a qualidade de seus serviços e com os resultados alcançados junto a seus clientes. Ligada aos processos de qualidade, a pesquisa sobre a “satisfação” de clientes insere-se como pré-requisito que sustenta ações eficazes de marketing (ROSSI e SLONGO , 1998; DEMO, 2014).

O atendimento ao cliente é uma das bases de sucesso da empresa, por isso é necessário que o setor empresarial se preocupe em investir na preparação dos funcionários, para garantir um atendimento de qualidade ao cliente, que além de buscar um produto que venha satisfazer o seu gosto, o mesmo também espera ser bem atendido (COBRA,1997). Reforçando a importância da qualidade do atendimento já citada por Cobra, o autor Macarini (2014) fez um estudo de como o diferencial de atendimento pode garantir a continuação e sucesso de uma empresa por intermédio da qualidade e satisfação dos clientes, neste estudo, objetivou identificar maneiras de como o Conselho de Administração da Sicredi Sul Santa Catarina pode oferecer excelência no atendimento a seus associados. Para tanto, Macarini realizou um diagnóstico da situação atual do atendimento da cooperativa, identificou junto aos cooperados, por meio de questionário, as principais reclamações e elogios do atendimento prestado.

Muitas organizações estão cometendo um erro grave, elas oferecem um produto de qualidade e esquecem de prestar um atendimento adequado, ignoram os elogios e reclamações dos clientes, e para estes nada adianta uma empresa ter um bom produto se as pessoas que interagem com eles não refletem esta mesma qualidade.

Quando o atendimento é bom, e proporciona à satisfação do cliente, a organização está assim trilhando seu sucesso e garantindo um futuro promissor, para

sobrevivência de tal empreendimento em meio à alta competitividade existente no mercado atual. Os clientes podem perdoar erros, falhas no sistema e até mesmo produtos defeituosos, o que eles acham difícil perdoar são atitudes negativas constantes, onde o pessoal parece desinteressado e não prestam um bom atendimento, daí parte a importância de as empresas focarem no atendimento, e na gestão de relacionamento com o cliente (CHIAVENATO,2005; MACARINI, 2014).

É indiscutível que a satisfação do cliente é o alicerce para o sucesso da empresa, para se ter esta satisfação as pesquisas são de extrema importância para poder focalizar nos gostos e necessidades dos clientes para saberem o que eles almejam dos produtos e serviços de sua organização. Segundo Moutella (2003), a satisfação se mede através da relação entre o que o cliente recebeu ou percebeu e o que esperava ter ou ver (percepção x expectativa). Se a percepção é maior do que a expectativa, o cliente fica muito mais satisfeito do que esperava. Mas se for menor, frustra-se e não registra positivamente a experiência.

A satisfação do cliente depende do que ele percebeu em relação ao desempenho do serviço em comparação com suas expectativas, se não corresponder às expectativas do cliente, o mesmo ficará insatisfeito, se corresponder ele ficará satisfeito, se exceder ele ficará altamente satisfeito e maravilhado. Os clientes insatisfeitos podem ou não revelar sua insatisfação, e quando os insatisfeitos vão embora, tiram a oportunidade de as empresas repararem os problemas percebidos por eles que causam sua insatisfação, já quando eles reclamam dão a oportunidade de reverter à situação.

Uma empresa centralizada no cliente está preocupada em facilitar o processo de recebimento de sugestões e reclamações, estudos comprovam que enquanto os consumidores ficam insatisfeitos com uma em quatro compras, menos de 5% deles reclamaram. Muitas organizações têm em vista à alta satisfação, pois clientes que estiverem apenas satisfeitos estão mais dispostos a mudar quando surgirem uma melhor oferta (KOTLER, 2000; SILVA, 2012).

2.2. Descoberta de conhecimento

A descoberta de conhecimento tem como seu objetivo principal manipular dados para que se obtenha um conhecimento que não é visível, isto porque a

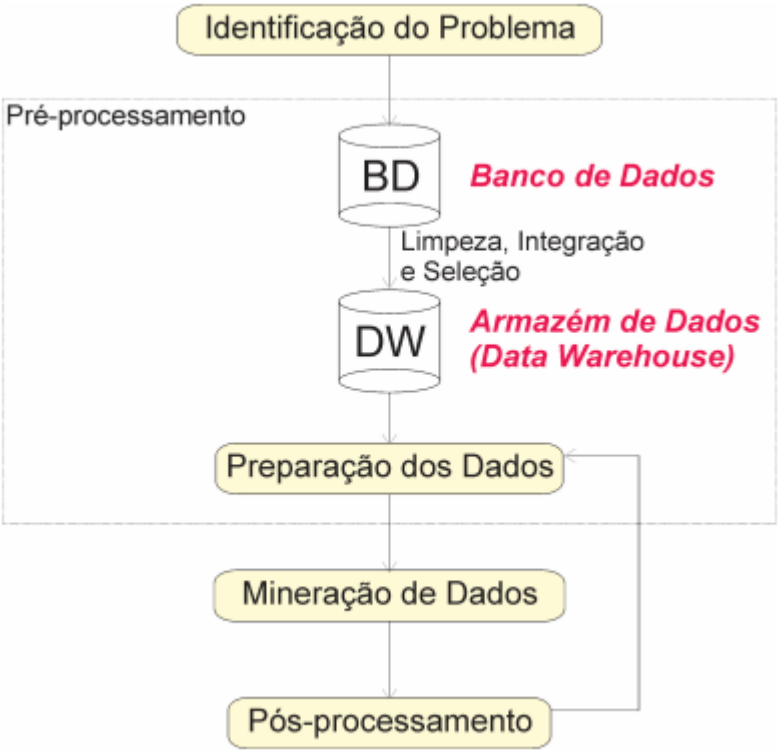
quantidade de dados a ser analisados é imensa. Diariamente, muitos dados são gerados, são inúmeras as fontes de informação: sistemas de gestão, sistemas de atendimento, internet, enfim, geramos informação a todo tempo. E essas informações se tiverem um processamento adequado podem se tornar informações uteis para empresas, pode-se formatar conhecimento inteligente baseado nos casos já ocorridos, e auxiliar na tomada de decisão. Levando em consideração que as fontes de informação podem ser as mais diversas, separa-se em dois grandes grupos de analises de dados: dados estruturados e dados não estruturados (WIVES, 2002; AMO,2016).

Quando se deseja analisar informações que estão em Banco de dados, onde existe correlações bem definidas entre as tabelas, em planilhas ou em dados que estejam formatados e estruturados, é possível utilizar técnicas de Descoberta de Conhecimento em Dados (Knowledge Discovery in Databases – KDD).

Segundo Amo (2016), 80% dos dados são compostos de informações não estruturadas, como textos, postagens da internet e informações desordenadas. Desta forma as técnicas orientadas a KDD tornam-se ineficazes, sendo necessária a utilização da Descoberta de Conhecimento em Textos (Knowlegde Discovery from Text –KDT). Quando se trata de análise de textos existe uma complexidade maior na análise, isto porque precisa-se considerar que a linguagem natural e padrões linguísticos podem estar presentes, e estes não podem interferir nos resultados finais.

Segundo Silberchatz (2006), o processo de busca por conhecimento tem quatro fases principais que precisam ser respeitadas, as etapas são divididas em: Identificação do Problema, Preparação dos dados, Mineração da Informação e Pós Processamento, na figura 1 é possível analisar cada uma destas fases.

Figura 1 - Fases do Processo de Descoberta de Conhecimento



Fonte: (Silberchatz, 2006)

A primeira fase, **a identificação do problema**, é relativamente simples, porém é uma das fases mais importantes e que precisa estar muito bem organizada, é necessário saber qual o problema que precisa ser solucionado, nesta etapa os gestores da empresa, os especialistas no negócio que está sendo tratado poderão auxiliar de forma muito eficiente. A partir desta fase é que todas as demais serão formatas e direcionadas para a resolução do principal problema.

No **pré-processamento** dos dados analisa-se as fontes de dados fornecidas e quando diversas fontes são utilizadas pode-se utilizar um *Data Warehouse*, onde é feita a integração de todas as bases de conhecimentos. Este processo pode variar de acordo com o tipo de informação e estrutura de cada uma das bases que serão utilizadas.

Ainda neste processo inicial é necessário analisar a necessidade de limpeza destes dados, a falta de informações em campos que podem ser triviais, e em casos onde envolve-se texto a **preparação dos dados**. Para estes processos podem ser utilizadas ferramentas que auxiliam na organização dos dados e estruturação das

informações. Porém é necessário já ter definida qual será a forma de mineração dos dados, qual será o algoritmo, qual a técnica que precisara ser aplicada para que o Problema inicial possa ser identificado e tratado através da formatação dos dados realizada. Ao longo deste trabalho, apresentaremos algumas técnicas e ferramentas que podem auxiliar nesta fase.

Na fase onde o conhecimento é gerado, a **mineração dos dados**, algoritmos específicos podem ser utilizados para descobrir padrões, situações que são comuns e quais associações que diversos dados ou situações podem ter, além da classificação dos dados. Para que a mineração seja eficiente para solucionar o problema inicial é necessário que as técnicas utilizadas sejam devidamente analisadas, e se necessário o processo de mineração repetido diversas vezes para que os resultados mais adequados sejam alcançados. É importante que o conhecimento gerado seja de compreensão humana, pois os especialistas que descreveram o problema precisam analisar o que foi gerado.

Com todas as etapas anteriores concluídas passa-se então para a etapa final, que é a análise e **avaliação dos resultados gerados**. Então especialistas avaliam a informação que foi gerada e validam com a realidade do negócio e casos de uso já ocorridos, onde não tendo o resultado esperado é possível voltar a qualquer uma das fases anteriores.

2.3. Mineração de dados

A Mineração de Dados é um ramo da computação que teve início nos anos 80, quando os profissionais das empresas e organizações começaram a se preocupar com os grandes volumes de dados informáticos estocados e inutilizados dentro da empresa. Nesta época, mineração de dados consistia essencialmente em extrair informação de gigantes bases de dados da maneira mais automatizada possível. Atualmente, mineração de dados consiste sobretudo na análise dos dados após a extração, buscando-se por exemplo levantar as necessidades reais e hipotéticas de cada cliente (AMO, 2016).

A mineração faz parte do processo de KDD e consiste em extrair somente os dados que tem considerável relevância. É uma área que está cada vez mais sendo explorada pelos gestores das empresas, pois utilizando as técnicas da forma correta ela pode apresentar propostas reais e hipotéticas tornando mais simples e prática a análise dos gestores. É uma área multidisciplinar, que pode envolver Banco de dados,

inteligência artificial, redes neurais, dados estatísticos e matemáticos, com isso a definição pode variar de acordo com o tema a ser associado a Mineração de dados. Em Zhou (2003), são destacados três autores que definem conforme a sua área de atuação a Mineração de dados:

- Definição de uma perspectiva de banco de dados: “Mineração de dados é um campo interdisciplinar que junta técnicas de máquinas de conhecimentos, reconhecimento de padrões, estatísticas, banco de dados e visualização, para conseguir extrair informações de grandes bases de dados” (CABENA,1998).
- Analisando de uma forma estatística: “Mineração de dados é a análise de grandes conjuntos de dados a fim de encontrar relacionamentos inesperados e de resumir os dados de uma forma que eles sejam tanto uteis quanto compreensíveis ao dono dos dados” (HAND, 2001).
- Na área de aprendizado de máquina: “Mineração de dados é um passo no processo de Descoberta de Conhecimento que consiste na realização da análise dos dados na aplicação de algoritmos de descoberta que, sob certas limitações computacionais, produzem um conjunto de padrões de certos dados” (FAYYAD, 1996).

2.3.1. Tarefas da Mineração de dados

Apesar das diferentes definições dos autores citados, a Mineração de dados pode ser utilizada para qualquer situação problema, onde se tem grande massa de dados que precisa ser analisada. Para tal, é preciso definir qual a tarefa a ser utilizada para solucionar o problema, esta escolha é baseada no tipo do problema que ser resolver, são denominadas Tarefas de Mineração de dados: Regras de associação, Padrões sequenciais, de Classificação ou Análise de agrupamentos. Cada tarefa tem suas respectivas técnicas e algoritmos de aplicação, abaixo segue a explicação e definição de cada Tarefa, bem como exemplos do cotidiano afim de facilitar a associação da tarefa com o problema a ser resolvido:

Regras de Associação

Nas regras de associação, fatos são analisados e relacionados, verificasse que um fator tem total relação com outro. Como exemplo pode-se utilizar uma atividade cotidiana, como fazer um churrasco, neste caso, todas as pessoas que compram carne para churrasco compram também carvão e cerveja. Em uma aplicação de

negócio pode-se dispor estes itens próximos para facilitar ao consumidor que vai adquiri-los.

Padrões Sequenciais

Os algoritmos aplicados nesta tarefa são capazes de identificar padrões sequenciais que estão implícitos na base de dados analisada. Pode-se considerar como exemplo, clientes de uma loja de computadores que compram Computadores com HD de 500 GB. Todos estes clientes dois meses após a primeira compra adquirem um HD externo. Desta forma é possível facilmente prever que clientes que compram Computadores com baixo armazenamento tem tendência a comprarem um HD pouco tempo depois.

Classificação

A tarefa de classificação pode ser utilizada para predizer alguma informação, ou seja, baseado em um conjunto de dados é feita uma análise e classificado em grupos. Suponha uma cooperativa de crédito, está para liberar um determinado valor para um associado, precisa fazer uma análise do seu histórico de compras, onde podem ser considerados ainda diversos fatores, como a idade, renda média entre outros; baseado nessas informações é possível classificar e predizer se o associado será um bom pagador ou não.

Análise de Argumentos

Os agrupamentos podem ser facilmente confundidos com a classificação, porque este consiste em agrupar as classes, porém estas já estão etiquetadas, já tem uma definição e não é necessária a predição. Exemplo, pode-se considerar a faixa etária das pessoas que assistem determinado filme, por mais que haja uma classificação, esta já está explícita e resta apenas que ser agrupada.

Após a associação do problema a ser solucionado ou conhecimento a ser adquirido e da tarefa de mineração de dados passa-se então ao passo mais técnico da mineração de dados: a análise e comparação dos algoritmos aplicados ao problema inicial.

2.4. Mineração de textos

Assim como a mineração de dados, a mineração de textos também tem várias definições. Segundo Lopes, o termo se refere ao processo de extração de padrões interessantes e não triviais, ou conhecimento a partir de documentos em textos não-

estruturados. Moura descreve a mineração de textos, como sendo uma área de pesquisa tecnologia cujo objetivo é a busca por padrões, tendências e regularidades em textos escritos em linguagem naturas (LOPES, 2004).

A mineração de textos consiste na descoberta da informação através da extração de dados a partir de coleções de textos dos mais variados tipos. É mais complexa que a mineração de dados, isto porque trabalha-se com dados não-estruturados, desta forma é necessário passar por todo um processo para estruturar estes dados para que possam ser avaliados. Esta área tem despertado grande interesse em decorrência da grande massa de dados gerada atualmente, principalmente em decorrência da popularidade da internet, da geração e fácil acesso a documentos textuais, são dados muito variados, que podem ser e-mails, artigos, documentos em diferentes formatos, textos de conversação, posts da internet, enfim são várias as origens destes dados, e por este motivo é necessária a aplicação de várias técnicas.

Fatores como a gramática, linguística e regionalismo são fundamentais para tratamento destes textos, análises qualitativas e quantitativas são necessárias para avaliar e mensurar os termos existentes. No processamento dos textos um usuário não inicia exatamente uma busca, ele inicia um processo de análise do documento.

2.5.1. Processos da Mineração de textos

Entre as diversas maneiras de explicar os processos de Mineração de textos vamos destacar duas formas: a primeira representação que expõem os processos de uma forma mais simples e a segunda onde os processos ficam mais divididos, porém com passos mais detalhadas.

Segundo Ebecken (2003), pode-se classificar os processos em três etapas: Pré-processamento dos documentos, Extração de padrões com agrupamentos de textos e Avaliação dos resultados, na figura 2 segue a representação gráfica do processo segundo o autor.

Figura 2- Processos de Mineração de Textos Simplificado

Fonte: (Ebecken, 2003)

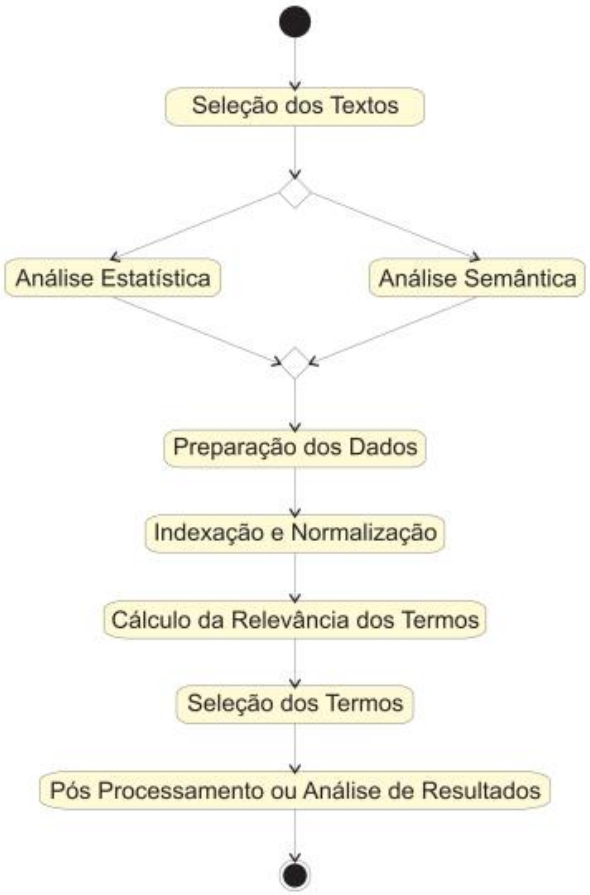
Ebecken (2003), apresenta ainda outro modelo para representação do processo de Mineração de textos, este de uma forma mais detalhada, considerando desde a classificação dos documentos até a avaliação dos resultados. Neste segundo formato os processos são classificados em: Seleção dos documentos, análise dos dados textuais, a preparação dos dados, Indexação e Normalização, Cálculo da Relevância dos termos, Seleção dos termos e o Pós-processamento e Análise de Resultados, na figura 3 segue a representação do processo.

Vamos utilizar o segundo modelo de representação dos processos de Ebecken para detalhar cada uma das fases da mineração de textos.

Seleção dos textos

A seleção dos textos é o primeiro passo, onde consiste apenas na identificação e classificação de quais informações serão avaliadas pelo processo.

Figura 3 - Processos de Mineração de Textos Detalhado



Fonte: (Ebecken, 2003)

Análise dos dados textuais

A análise de dados pode ser abordada de duas formas, a análise semântica e a análise estatística.

A análise semântica consiste em considerar a importância das palavras, e a sequência dos termos no texto é avaliada. É fundamentada pelas técnicas de Processamento de Linguagem Natural, e envolve conhecimento morfológico, da forma e das inflexões das palavras. Envolve ainda conhecimento sintático e semântico para identificar o significado das palavras independente do contexto, neste considera-se a combinação de palavras em um sentido mais complexo. Análise do contexto e de como a interpretação das palavras pode levar a resultados diferentes. Enfim a analise semântica considera muito mais o conhecimento e a interpretação do texto e dos fatores por ele representados (CORDEIRO,2005).

Já na análise estatística considera-se a frequência dos termos, a importância que o termo tem no contexto não é dada através de conhecimento, mas no número de vezes que o mesmo aparece. Os especialistas indicam uma codificação dos termos para que estes possam ser relevantes para alcance do objetivo principal. A representação do texto é vista em formato de blocos de informação, então a pontuação e disposição das palavras nos textos são irrelevantes, considera-se apenas a raridade e a constância dos termos, podendo assim fazer associações entre os termos.

Preparação dos dados

Este é o primeiro passo para a descoberta de conhecimento em textos, onde ele tenta identificar a similaridade das palavras, considerando morfologia e significado. Como as consultas são baseadas em termos, é necessário que estes estejam bem estruturados e definidos pelos especialistas, pois do contrário a busca não terá a eficiência esperada, pois a indexação dos mesmos não encontrará os termos mais adequados. Com um mecanismo de Análise de Relevância e a função de similaridade é possível identificar uma relação entre os termos de consulta e os existentes no texto. É necessário observar que esta comparação é direta e pode não considerar sinônimos, polissemia e ambiguidade, desta forma é necessário observar que nem sempre os resultados serão satisfatórios.

Para o Cálculo de similaridade de termos são disponibilizadas diversas funções para a Recuperação da Informação, Wives (2002) apresenta diversos métodos: Método Booleano, Modelo de Espaço Vetorial, Modelo Probabilístico, Modelo Difuso, Busca Direta, Aglomerados, Modelo Lógico, Modelo Contextual ou Textual. O modelo mais apropriado vai de acordo com o que se deseja obter de resultados, na tabela 1 segue uma comparação dos Modelos para facilitar a compreensão.

Tabela 1 - Comparação de Modelos de Similaridade

Modelo	Forma de Representação	Forma de relevância	Considera incerteza	Considera Contexto
Booleano	Grupos ligados por expressões booleanas	Grau de Intervenção	Não	Não
Espaço Vetorial	Vetores	Peso de termos	Não	Não
Probabilístico	Índices	Estatísticas	Não	Não
Difuso	Vetores	Grau de relevância	Sim	Não
Busca Direta	Texto	Encontro o termo ou não encontrou o termo	Não	Não
Clusters	Grupos	Similaridade das palavras	Não	Não
Lógico	Booleano	Lógica preditiva	Não	Não
Conceitual	Ontologias	Peso dos termos	Sim	Sim

Fonte: do autor

A tabela comparativa, serve para visualmente identificar pontos estratégicos dos modelos de busca por similaridade, cada modelo tem seu ponto forte, é necessário saber exatamente que tipo de retorno na busca do Conhecimento está se querendo obter, para assim poder escolher o melhor modelo. Por exemplo, o modelo Difuso, este utiliza lógica Fuzzy e por este motivo considera um grau de incerteza, atributo que em outros modelos não é considerado.

O modelo Conceitual ou Textual considera a presença dos termos no documento, ele é capaz de associar termos para que se avalie o contexto da expressão no texto e da busca realizada. Esta tarefa de associar termos e fazer a busca por similaridade nos documentos pode ao mesmo tempo que ser eficiente fugir do contexto e responder um incorretamente. A seguir seguem os métodos de similaridade explicados detalhadamente segundo Wives (2002).

No modelo Booleano consideram-se os documentos como sendo conjuntos de palavras. Possui esse nome justamente por manipular e descrever esses conjuntos através de conectivos booleanos (*and*, *or* e *not*). As expressões booleanas são capazes de unir conjuntos, descrever intersecções e retirar partes de um conjunto. Em uma busca, por exemplo, o usuário indica quais são as palavras que o documento resultante deve ter para que seja retornado. Assim, os documentos que possuem interseção com a consulta são retornados. Os documentos podem ser ordenados pelo grau de interseção, onde o mais alto é aquele que contém todas as palavras especificadas na consulta do usuário, e o mais baixo o que contém somente uma.

Considerando o modelo de espaço-vetorial, cada documento é representado por um vetor de termos e cada termo possui um valor associado que indica o grau de importância desse no documento. Portanto, cada documento possui um vetor associado que é constituído por pares de elementos. Nesse vetor são representadas todas as palavras da coleção e não somente aqueles presentes no documento. Os termos que o documento não contém recebem grau de importância zero e os outros são calculados através de uma fórmula de identificação de importância. Isso faz com que os pesos próximos de um indiquem termos extremamente importantes e pesos próximos de zero caracterizem termos completamente irrelevantes. O peso de um termo em um documento pode ser calculado de diversas formas. Esses métodos de cálculo de peso geralmente se baseiam na contagem do número de ocorrências dos seus termos.

Para trabalhar com o modelo probabilístico, são utilizados conceitos provenientes da área de probabilidade e estatística. Nesse, busca-se saber a probabilidade de um documento ser relevante a uma consulta, caso os termos especificados por esta apareçam naquele. Existem diversas formas de se obter estatisticamente essa informação, porém, a base matemática adotada para esse modelo é o Método Bayesiano. Devido a isso, esse modelo também é chamado de Modelo Bayesiano.

No modelo Difuso, os documentos também são representados por vetores de palavras com seus respectivos graus de relevância. A diferença está no conceito relacionado à relevância. Na teoria de conjuntos difusos, todas as características de determinado universo estão presentes em todos os conjuntos. A diferença é que a presença pode ser medida e pode não ser exata, ou seja, pode haver incerteza. Logo, não há conjunto vazio, mas sim um conjunto cujos elementos possuem uma relevância muito baixa. A teoria difusa permite trabalhar com esses valores intermediários que indicam o quanto determinado objeto pertence ou não ao conjunto, pois esta foi construída com a finalidade de tratar incertezas e imprecisões.

O modelo de busca direta também é denominado de Modelo de Busca de Padrões, e utiliza métodos de busca de *strings* para localizar documentos relevantes. Na prática, esse modelo é utilizado na localização das *strings* no documento. As buscas são realizadas diretamente nos textos originais, em tempo de execução. O resultado da busca é a localização de todas as ocorrências do padrão de consulta em

um documento ou conjunto de documentos. Sua utilização é aconselhada em casos onde a quantidade de documentos é pequena, sendo muito utilizada em softwares de edição de documentos para que o usuário possa localizar palavras ou expressões no texto que está editando.

Já no modelo de aglomerados que também é conhecido como *Clustering Model*, utilizam-se técnicas de Agrupamento de documentos. Seu funcionamento consiste em identificar documentos de conteúdo similar e armazená-los ou indexá-los em um mesmo grupo ou aglomerado. A identificação de documentos similares em conteúdo dá-se pela quantidade de palavras similares e frequentes que eles contêm. Quando o usuário especifica sua consulta, o sistema identifica um documento relevante e retorna para o usuário todos os documentos pertencentes ao mesmo grupo.

O Modelo lógico, que se baseia em métodos e teorias provenientes da lógica matemática para modelar o processo de recuperação de documentos, as aplicações existentes neste modelo são aparentemente de âmbito acadêmico e teórico. Para que o modelo lógico funcione torna-se necessário modelar os documentos através de lógica predicativa, o que exige um grande esforço no trabalho de modelagem, incorporando semântica ao processo de recuperação. Com isso o sistema passa a conhecer o conteúdo dos documentos, podendo julgar melhor a relevância desses para seu usuário.

Todos os modelos já apresentados consideram a presença de termos em documentos. Eles realizam o “casamento” entre um documento e uma consulta somente se as palavras contidas no documento tiverem a mesma morfologia que as palavras especificadas na consulta. Logo, os documentos que não possuem as palavras identificadas são considerados irrelevantes por possuírem uma morfologia diferente. Devido à ambiguidade e incerteza inerentes à linguagem natural, esta abordagem torna-se muito restritiva, causando problemas de sinonímia e polissemia. Neste sentido, o modelo contextual é desenvolvido a partir do princípio de que todo documento possui um contexto, pois a pessoa que escreve um texto o faz desenvolvendo um assunto específico, e utiliza frases interconectadas ou encadeadas que fazem sentido dentro do assunto.

Indexação e Normalização

O processo de Indexação e Normalização consiste em indexar os termos, selecionar quais serão os termos relevantes de busca, ou seja, quais são as palavras chave daquele documento, por quais termos ela poderá ser indexada. Esta escolha, é necessária de acordo com cada área que está sendo trabalhada, isto porque o índice utilizado em determinados documentos pode ser irrelevante para outros, desta forma os mesmos precisam ser identificados cuidadosamente. Esta seleção de termos pode ser feita manualmente ou automaticamente, nas duas formas o resultado deve ser um índice de termos relevantes. Considerando uma construção automática, leva-se em conta os seguintes passos: Identificação dos termos, Remoção de *Stopwords* e Normalização Morfológica.

Na identificação dos termos é feito um tratamento inicial do documento, onde todos os caracteres podem ser convertidos para maiúsculo ou minúsculo, a tabulação e estruturação do documento pode ser modificada. É o momento de padronização, onde pode ser utilizado até mesmo um dicionário de termos para verificação de erros ortográficos. É o processo onde o tratamento das palavras acontece, sendo palavras simples ou compostas. Nesta fase, também são tratadas as expressões, onde o termo é composto de mais palavras, que se separadas não tem sentido algum, porém se colocados lado a lado podem dar forma a uma expressão que precisa ser considerada. Desta forma pode-se haver um dicionário de expressões, para facilitar na busca ou ainda utilizar a busca identificando termos ocorrem com frequência nos documentos, desta forma ele poderá ser automaticamente inserido como uma expressão de termos.

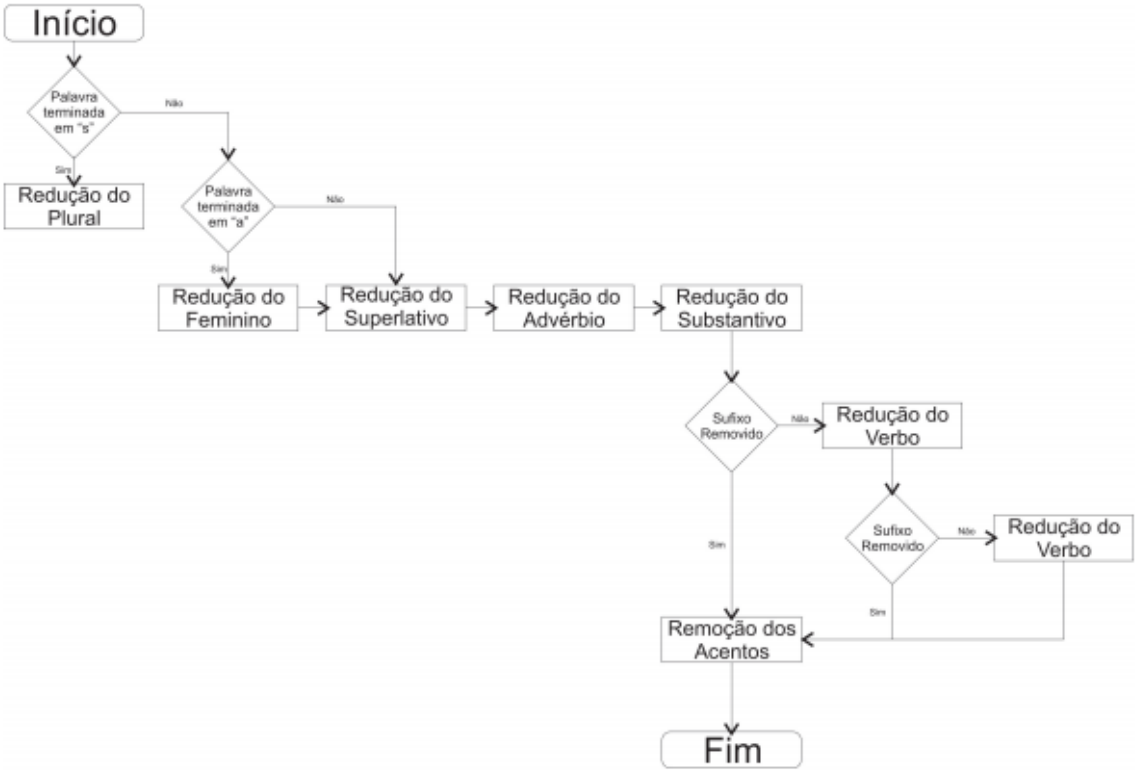
Ainda no processo de indexação e normalização é feito uma limpeza do documento, onde algumas palavras são eliminadas: são pronomes, preposições, artigos, advérbios e outras classes que contem palavras auxiliares, estas são denominadas *Stopwords*, palavras que não tem relevância no texto, e que se mantidas só farão com que a lista de índices aumente.

Além da remoção dos *Stopwords* o texto passa por um processo de normalização morfológica, neste são agrupadas palavras no plural, no sentido feminino ou masculino do termo, é o chamado passo de *Steamming*, este tem como objetivo principal reduzir a dimensionalidade dos termos. Os métodos de *Steamming* são escolhidos de acordo com a linguagem que se está trabalhando, abaixo segue a

descrição de um método de *Stemming* para a Língua Portuguesa segundo Orengo (2001) (Figura 4).

Note, que são todas etapas de tratamento de linguística e que tem um conjunto de regras, que são examinadas em sequência para normalizar e padronizar os termos, desta forma consegue-se reduzir significativamente o número de índices, tornando a busca mais rápida. Em contrapartida, é importante entender que quanto maior a remoção de termos menos eficiente pode ser a busca, pois menos índices ela terá.

Figura 4 - Stemming da Língua Portuguesa



Fonte: (Orengo, 2001)

Cálculo de Relevância dos Termos

A relevância de um termo no texto pode ser dada de diferentes formas, pode-se considerar a disposição do termo no texto. Por exemplo, se a palavra está no título pode ter maior relevância; pode-se considerar também a frequência que ela ocorre no texto analisado ou em uma coleção de textos. A relevância por frequência é um dos cálculos mais simples para utilização, pois considera apenas a frequência, disposição e estrutura dos termos.

Em outros cálculos a Linguagem Natural pode ser considera o que eleva consideravelmente a complexidade de identificação do peso de relevância de cada Termo. Segundo Wives (2002), muitas formas de calcular o peso podem ser apresentadas, porém três formas são as mais utilizadas: a frequência absoluta, a frequência por relevância e a frequência inversa.

Na frequência absoluta é considerado apenas quantas vezes o mesmo termo aparece no documento que ele está localizado, este modelo é o mais simplório porque ele não faz nenhum tipo de relação com os demais documentos. Desta forma ela não é capaz de definir se o termo é relevante ou não completamente, pois um termo que pode aparecer com muita frequência em um documento em outro pode estar completamente ausente.

No cálculo de relevância de frequência relativa, consideram-se os demais termos que estão neste mesmo documento, sendo capaz de calcular o peso do termo de acordo com o número de termos presentes no arquivo, ou seja, ele utiliza o valor de frequência absoluto do termo no texto e divide pelo número total de termos do documento. Da mesma forma que a frequência absoluta esta não considera demais documentos.

Já o cálculo de frequência inversa considera o termo em seu documento original e também nos demais arquivos da coleção, desta forma ele utiliza o número de vezes que o termo aparece no documento em questão e divide pelo número de documentos que contém o termo desejado.

Seleção dos Termos

Corresponde a etapa de seleção de palavras retiradas do texto, os índices, após o pré-processamento e o cálculo de relevância. Wives (2002), apresenta várias técnicas para seleção de termos: filtragem baseada no peso do termo, análise de co-ocorrência, análise de linguagem natural entre outras. Porém o método de Luhn é uma das técnicas mais tradicionais para seleção de termos utilizando a medida frequência dos termos.

Esse método foi baseado na Lei de Zipf, também conhecida como Princípio do Menor Esforço. Em textos, ao contabilizar a frequência dos termos e ordenar o histograma resultante em ordem decrescente, forma-se a chamada Curva de Zipf, na qual o k-ésimo termo mais comum ocorre com frequência inversamente proporcional

a k. Os termos de alta frequência são julgados não relevantes por geralmente aparecerem na grande maioria dos textos, não trazendo em geral, informações úteis. Já os termos de baixa frequência são considerados muito raros e não possuem caráter discriminatório. Assim, são traçados pontos de corte superior e inferior da Curva de Zipf, de maneira que termos com alta e baixa frequência são descartados, considerando os termos mais significativos os de frequência intermediária.

Pós Processamento

A etapa de análise de resultados é uma das mais importantes, pois é esta que dita se as técnicas utilizadas foram ou não eficientes, é nesta etapa que se determina se é necessário voltar e refazer algum processo para alcançar o resultado esperado no sistema de Recuperação de Informações. Pode ser utilizado métodos matemáticos e estatísticos para definir as métricas dos resultados, outra forma é definir em segmentos para classificar a eficiência, uma das opções é: Documentos relevantes para a consulta que foram recuperados, Documentos relevantes que não foram recuperados, Documentos irrelevantes que foram recuperados e Documentos irrelevantes que não foram recuperados. Baseados nesta classificação é possível mensurar a eficácia do Sistema de recuperação, pois o cenário ideal é um Sistema que busque o máximo de documentos relevantes a consulta e o mínimo ou ne documento irrelevante seja apontado.

2.4.2. Ferramentas para Mineração de Textos

Muitos estudos e pesquisas foram realizadas na área da Mineração de textos, e com isso muitas ferramentas foram desenvolvidas para facilitar esse processo, nas próximas sessões a autora apresenta algumas destas ferramentas.

2.4.2.1. Sobek (Reategui, 2011)

O Software Sobek, foi desenvolvido pelo professor Eliseo Reategui, da Faculdade de Educação da Universidade Federal do Rio Grande do Sul (UFRGS), num projeto iniciado em 2010, onde criou com a ajuda de alunos uma ferramenta digital capaz de extrair automaticamente os conceitos principais de um texto e mostrar graficamente seu grau de importância e suas inter-relações.

A ferramenta, disponível em português e inglês, é chamada Sobek (nome de uma divindade egípcia que simboliza força, devastação e reconstrução) e pode ser acessada gratuitamente no *site* da universidade. Qualquer texto submetido a ela é

decomposto em seus conceitos principais, representados por meio de um nodo. O método é estatístico, portanto a importância dos conceitos é medida pelo número de vezes que uma mesma palavra é repetida no texto. Há filtros que descartam as palavras frequentes que, no entanto, não geram sentido isoladamente, como artigos e preposições.

O modelo utilizado para a extração de conceitos foi o do algoritmo de Schenker, criado em 2003. No entanto, a Sobek apresenta uma representação simplificada que torna a leitura mais concisa e acessível.

A ferramenta vem sendo experimentada em várias áreas, e alunos participantes ou próximos do Centro Interdisciplinar de Novas Tecnologias em Educação (CINTED) da UFRGS estruturaram projetos-pilotos em escolas do Estado.

Por estar disponível sem restrições na internet, a utilização da Sobek é igualmente irrestrita. Atualizações e aperfeiçoamentos também ocorrem com grande frequência e conforme a demanda identificada. Um grupo de alunos-programadores trabalha permanentemente nisso. Recentemente toda a interface do ambiente foi reformulada para ficar mais dinâmica e em pouco tempo será lançado um aplicativo para utilização da ferramenta em equipamentos móveis, mesmo não conectados à internet.

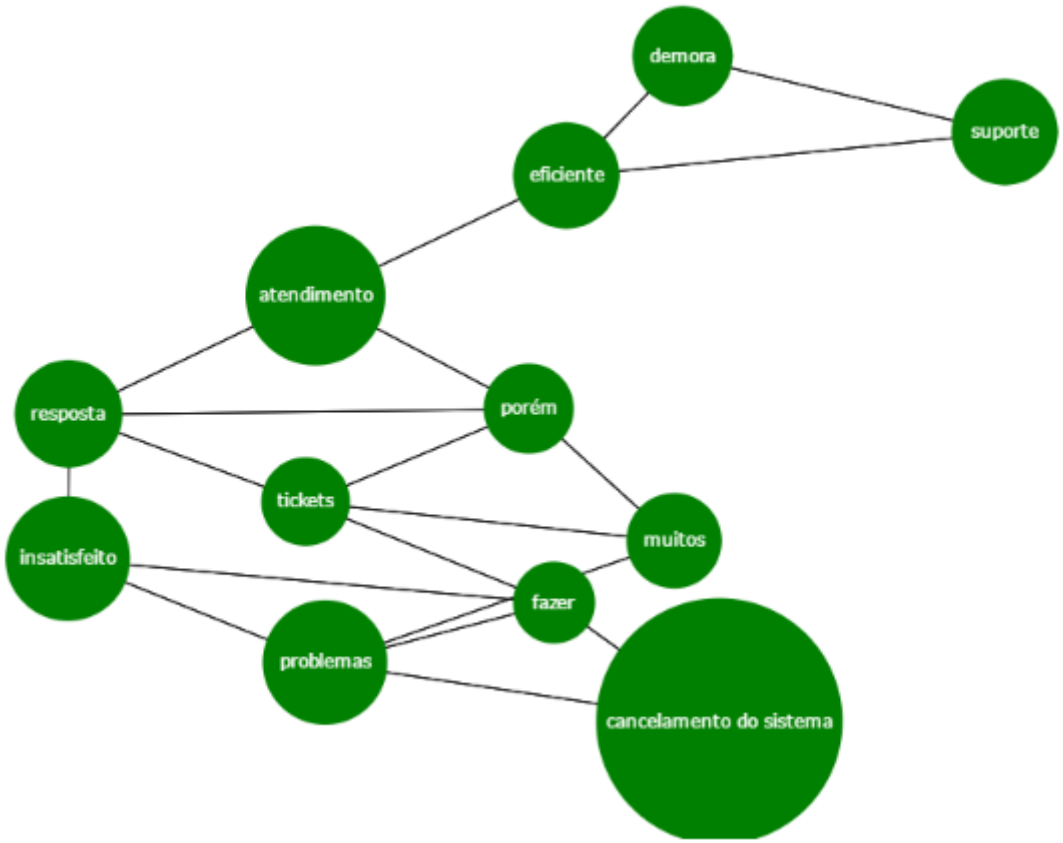
Para minerar o texto dentro dessa ferramenta, primeiro copia-se o texto na área de entrada que pode ser em formato txt, doc ou pdf. Depois o software cria uma base de conceitos automaticamente, definindo quais palavras que incidem com mais frequência dentro do texto. No último passo, o programa gera um grafo, que mostra uma visualização gráfica das interligações existentes no documento, por meio das palavras que incidem com maior frequência (SOBEK, 2016). Abaixo segue um texto de exemplo fornecido pela empresa MK Solutions para exemplificarmos o modelo de processamento da ferramenta Sobek:

> *“Eu tenho muitos atendimentos antigos e muitos tickets sem resposta, por isso eu estou totalmente insatisfeito. Vou fazer o cancelamento do Sistema, não estou gostando dos problemas que estou tendo. Estou muito insatisfeito.*
> *O que vocês podem fazer para evitar que eu efetue o cancelamento do sistema. Gosto do suporte de vocês, o Suporte é eficiente e resolve quando a pessoa correta nos atende, porém estou com muitos problemas recorrentes.*

> *Vai fazer mais de um ano que eu abri diversos tickets com sugestões para o seu desenvolvimento, porém alguns nem obtive resposta, estou ficando irritado com isso.*
> *O atendimento do suporte de vocês demora muito para entender o que nós queremos, aí sempre entra em um processo burocrático e demorado, o que ocasiona em não solução do problema.*
> *O suporte com a Vanessa é eficiente, só que com a Mariele demora muito."*

Após o Software Sobek processar o texto, é possível visualizar que ele conseguiu identificar diversos termos relevantes e suas ligações, desta forma é possível avaliar cada um dos termos e validar o atendimento quanto a que se refere e também quais os termos mais frequentes no texto, na figura 5 é possível analisar a árvore gerada.

Figura 5 - Exemplo de análise do Software Sobek



Fonte: (Sobek, 2016)

A figura 5 ilustra como o Software Sobek retorna graficamente os termos localizados no texto, bem como a relação entre os termos e expressões nele contidas.

2.4.2.2. Apache Lucene

O Apache Lucene é um Framework que foi criado por Doug Cutting em 2000, e é uma das mais famosas e mais usadas bibliotecas para indexação e consulta de textos disponível em código aberto. Sob o domínio da Apache Foundation, a biblioteca escrita em java pode ser utilizada em qualquer aplicativo de código aberto ou não.

A biblioteca é composta por duas etapas principais: a indexação e a pesquisa. A indexação processa os dados originais gerando uma estrutura inter-relacionada eficiente para a pesquisa baseada em palavras-chave. A pesquisa por consulta o índice pelas palavras digitadas em uma consulta e organiza os resultados pela similaridade do texto com a consulta.

O Lucene implementa uma linguagem para consulta que proporciona pesquisas restritivas por campos, por expressões regulares, consultas e em ranges. Além disso permite ao desenvolvedor criar dois tipos de consulta avançadas: a fuzzy, que utiliza a distância de Levenshtein ao avaliar a proximidade entre as palavras.

Os índices podem ser criados em ambientes distribuídos, aumentando a performance e a escalabilidade da ferramenta. Calcula-se que o Lucene consiga indexar cerca de 20MB de texto por minuto em um computador com um único core de 1.5 Ghz. Os arquivos de índices comprimidos ocupam cerca de 25% do tamanho sem compressão (LUCENE, 2016).

A criação de um aplicativo de procura usando o Lucene envolve, a indexação de dados, a procura de dados e a exibição de resultados de procura (IBM, 2016).

O Processo de Indexação

A Indexação é um processo de converter os dados de texto em um formato que facilita a procura rápida. Uma analogia simples é um índice que seria localizado no final de um manual: Esse índice aponta para o local dos tópicos que aparecem no manual. O Lucene armazena os dados de entrada em uma estrutura de dados chamada de índice invertido, que é armazenado no sistema de arquivos ou na memória como um conjunto de arquivos de índice. A maioria dos mecanismos de procura da Web usa um índice invertido. Ele permite que os usuários executem procuras rápidas por palavras-chave e localizem os documentos que correspondem a uma determinada consulta. Antes que os dados de texto sejam incluídos no índice, eles são processados por um analisador.

Análise

Análise é a conversão dos dados de texto em uma unidade de procura fundamental, chamada de termo. Durante a análise, os dados de texto passam por várias operações: extração das palavras, remoção de palavras comuns, ignorar pontuação, redução de palavras para o formato de raiz, alteração das palavras para minúsculas, etc. A análise acontece imediatamente antes de analisar a indexação e a consulta. A análise converte os dados de texto em *tokens*, e esses são incluídos como termos no índice do Lucene.

O Lucene é fornecido com vários analisadores integrados, como o *SimpleAnalyzer*, o *StandardAnalyzer*, *StopAnalyzer*, *SnowballAnalyzer*, e outros. Eles diferem na maneira pela qual tokenizam o texto e aplicam os filtros. Conforme a análise remove as palavras antes de indexar, ela diminui o tamanho do índice, mas isso poderá ter um efeito negativo na precisão do processamento da consulta. É possível ter maior controle sobre o processo de análise ao criar analisadores customizados usando os blocos de construção básicos fornecidos pelo Lucene.

Procurando Dados Indexados

A procura é o processo de buscar palavras no índice e de localizar os documentos que contêm essas palavras. A criação de recursos de procura usando a API de procura do Lucene é um processo direto e fácil

Procurador é uma classe base abstrata que possui vários métodos de procura sobrecarregadas. *IndexSearcher* é uma subclasse normalmente utilizada que permite procurar índices armazenados em um determinado diretório. O método procurar retorna uma coleta ordenada de documentos classificados pelas pontuações computadas. O Lucene calcula uma pontuação de cada documento que corresponde a uma determinada consulta. *IndexSearcher* é um thread-safe, ou seja, uma instância única que pode ser usada por vários encadeamentos simultaneamente.

Exibindo Resultados de Procura

IndexSearcher retorna uma matriz de referências nos resultados de procura classificados, como documentos que correspondem a uma determinada consulta. Você pode decidir o número dos principais resultados de procura que precisam ser

recuperados ao especificá-lo no método de procura do *IndexSearcher*. A paginação customizada pode ser criada sobre isso. As classes primárias envolvidas na recuperação dos resultados de procura são *ScoreDoc* e *TopDocs*.

3. TRABALHOS RELACIONADOS

Todo o processo apontado para preparação dos dados, indexação, seleção de termos e cálculo de relevância já conta atualmente com diversos softwares e estudos realizados. Cada uma das ferramentas e autores que serão apresentados utilizam uma técnica diferente para minerar os textos, desta forma os resultados obtidos se comparados podem não ser os mesmos. Além das técnicas utilizadas, as ferramentas utilizam plataformas e formas de apresentar os resultados diferentes

Junior (2015) apresentou um estudo para descobrir o motivo da evasão dos alunos dos cursos da computação da Universidade de Santa Cruz do Sul (UNISC), utilizando as tarefas de associação e classificação presentes no software Weka. Este autor utilizou a tarefa de associação para traçar o perfil dos alunos que cursaram determinadas disciplinas em sequência, e terminaram por abandonar o curso, para tal avaliou as disciplinas cursadas bem como o status de aprovado, não aprovado e desistente. Com a tarefa de classificador do Weka, pode fazer uma classificação por tempo de permanência no curso para identificar com quanto tempo o aluno tende a evadir dos cursos de computação. O autor ainda fez outras classificações para identificar o perfil de alunos que evadem o curso, fez diversos experimentos utilizando os algoritmos Apriori, o J48, e o FpGrowth para a realização de tarefas. Junior concluiu que o total de disciplinas cursadas e o status final das disciplinas do primeiro semestre são os fatores que mais colaboram para a evasão do curso, com os quais pode-se traçar um perfil para os alunos que evadem, como o perfil dos alunos que sempre fazem menos de três disciplinas, o perfil dos alunos que fazem mais que cinco disciplinas no primeiro semestre, os alunos que reprovam nas primeiras disciplinas do curso ou mesmo o perfil dos alunos que não seguem um padrão na quantidade de disciplinas cursadas a cada semestre.

Sausen (2015) propôs o uso de técnicas de mineração de opiniões, a partir dos dados extraídos em sites de lojas eletrônicas, permitindo a definição de aspectos e polaridade dos dados, possibilitando a extração dos dados para arquivos de extensão arff, suportado pela ferramenta Weka. Utilizou a execução de processo não-supervisionado de filtragem em conjuntos de dados para treinamento, permitindo a seleção, o pré-processamento e aplicação de algoritmos para definição do sentimento, gerando estatísticas para posterior análise. A análise foi com dados em Validação Cruzada e o algoritmo *NaiveBayes*, onde se confirmou a compatibilidade nas

avaliações. O autor utilizou a definição dos aspectos nos textos através do cadastro de ontologia e o sentimento através do cadastro de polaridade. Uma das limitações que o autor cita diz respeito ao tempo de processamento, que se mostrou superior a outros softwares, principalmente ao Weka, no qual foi comparado. Estes tempos podem ser reduzidos pela aplicação de técnicas de programação paralela ou a partir de melhorias na arquitetura da ferramenta desenvolvida.

Suptitz (2013) desenvolveu uma ferramenta que recupera informações do banco de dados dos serviços prestados por uma empresa de TI com o apoio de ontologias, servidor de apoio a consultores, técnicos e desenvolvedores que trabalham com suporte a usuários.

A ferramenta possibilita a manutenção das conexões às fontes dos dados que devem ser extraídos e processados, conexões estas utilizadas no processo de importação dos dados que serão usados na mineração de textos. Conta com o processo de pré-processamento, onde ocorre o tratamento do texto, removendo comandos HTML, filtro de caracteres permitidos, remoção de *stopwords*, correções ortografias e radicalização.

Suplitz desenvolveu o processo de indexação de termos, que gera uma estrutura de índices que serão utilizadas para consulta e geração de padrões. A manutenção manual de ontologias, conceitos vinculados, de acordo com sua hierarquia dentro da estrutura definida com base nos relacionamentos estabelecidos entre os conceitos.

O diferencial nos processos da ferramenta é apoiar o especialista na tarefa de criar e manter a ontologia, com a utilização da ferramenta Weka e foi realizada a construção de um software de apoio com interação com a ferramenta desenvolvida, nesta tela só é mostrada a interface da interação. Neste processo foram utilizados algoritmos de agrupamento como o "*SimpleKMeans*", "*xMeans*" e "*HierarchicalClusterer*".

A ferramenta conta com uma interface de consulta onde o usuário informa os termos da consulta que pretende submeter, seguido pelos resultados que serão achados, ordenados por relevância conforme cálculo baseado em referência de trabalhos pesquisados. Essa interface de consulta é aplicada todas as funcionalidades importantes do ponto de vista prático do objetivo do trabalho. Para fins, realizou dois

estudos de casos distintos: no primeiro, utilizou uma coleção de documentos previamente catalogados com o objetivo de gerar estatísticas com as métricas *precision* e *recall*, verificando o comportamento da aplicação em um ambiente distinto do problema proposto; no segundo tratou-se com dados de chamados técnicos e ordens de serviços de uma empresa de TI, configurando o ambiente do problema proposto.

Bulsing (2013) desenvolveu uma ferramenta que realiza a extração de dados de diversas fontes da web, armazenando-as em um banco de dados Big Data, com computação distribuída e um framework *Hadoop* em Java que permite trabalhar com milhares de nodos e volume de dados na escala de petabytes. As informações extraídas pelo motor de busca são armazenadas em um banco de dados NoSQL (Apache Cassandra).

Com a ideia de facilitar as buscas, Bulsing utilizou estruturas de dados que visam auxiliar nas consultas, as Ontologias e as técnicas de Recuperação de Informação, podendo modelar, catalogar e indexar os termos específicos na área do conhecimento, tornando mais precisa a consulta. Nas opções de execução do software é possível fazer uma consulta por termos nos dados já coletados pelo *crawlers*, deixando desmarcado o checkbox “Executar Crawler”. Marcando o motor de busca iniciará uma nova coleta de informações baseadas nos sites sementes, não havendo duplicidade de dados. O autor utilizou a linguagem de programação Java, e para a interface gráfica foi utilizado a API JavaFX 2, para a indexação das buscas foi utilizado o Apache Lucene.

Oliveira e França (2013), utilizaram a análise de sentimento acerca dos protestos que ocorreram no Brasil entre os meses de Junho e Agosto de 2013. Analisaram sobre uma grande massa de dados formada por mensagens disponibilizadas pelas pessoas na Web. Para tanto, foi criada uma base de *tweets* escritos em português brasileiro. Essa base foi pré-processada para criação do corpus de mensagens com menos ruídos. Esse corpus foi analisado para extração do sentimento presente nas mensagens. Observou-se a polaridade (apoio ou repúdio aos protestos) expressa nos *tweets*.

Os dados foram analisados e o resultado final demonstrou que a maioria das mensagens apoiaram os protestos. Para realizar as análises, optou-se pelo uso de

modelos estatísticos de aprendizagem *Naive Bayes* que é um sistema de classificação que independe de linguagem e que tem apresentado bons resultados na literatura segundo Oliveira e França.

Comentários do Autor

Os trabalhos relacionados serviram para auxiliar na definição da melhor técnica a ser utilizada, visto que há trabalhos que utilizam técnicas de mineração de dados, outros utilizam mineração de textos e trabalhos que associam a mineração com ontologias. Neste trabalho desenvolvido, o autor pode identificar que a melhor opção seria a utilização de mineração de textos, já que a fonte de informação não é estruturada. Junior utilizou mineração de dados para extrair informações de um Banco de dados estruturado e neste cenário a técnica mostrou-se eficiente.

Alguns trabalhos como o de Suplitz utilizaram Ontologias, porém é necessário ter uma estrutura bem definida e hierárquica para utilizar esta definição, e no caso deste trabalho não se tem essa hierarquia.

4. DESENVOLVIMENTO

No desenvolvimento da ferramenta de análise de satisfação que foi projetada neste trabalho, foram utilizadas diferentes tecnologias de acordo com a necessidade da solução. Na aplicação dos algoritmos de busca e indexação dos termos, foi utilizada a API Java do Lucene, devido à facilidade no acesso e suporte ao conteúdo.

Para identificação dos termos frequentes nos documentos foram acessados os webservices da ferramenta Sobek, pois este retorna uma lista com os termos e frequências.

O software foi desenvolvido através do Net Beans IDE 8.0.2, aplicação escolhida pela facilidade que o autor já possui com suas ferramentas, e a linguagem de programação Java.

4.1. Principais Funcionalidades

Dentre as funcionalidades do Software destaca-se a análise do nível de satisfação dos clientes, para tal foi utilizada técnicas e ferramentas de mineração de textos, com o Software Sobek e Apache Lucene, nas próximas sessões serão explicadas e exemplificas as formas de utilização.

Ainda utilizando recursos de mineração de textos, foi feita a análise do atendimento quanto a sua classificação, onde o software compara um dicionário de termos do sistema com termos expressados no texto e informa a que setor o atendimento se refere.

Analisadores de Texto

Para analisar o texto foi utilizado o analisador ***BrazilianAnalyzer***, pois este trata palavras da Língua Portuguesa. Para indexação dos núcleos foi utilizada a ***RAMDirectory*** que é uma classe que armazena todos os índices na memória. Para localizar os termos dentro do texto foi utilizada a classe ***IndexSearcher*** que é uma classe base abstrata que possui vários métodos de procura sobrecarregadas, a qual retorna uma lista ordenada de documentos classificados pelas pontuações computadas, onde a pontuação é quantas vezes o termo foi localizado no texto; a API retorna ainda em qual frase encontrou a expressão e o título do documento. Para navegar entre os resultados da classe utilizou-se um ponteiro simples contido nos

resultados de procura, o ***score docs***, que engloba a posição de um documento no índice e a pontuação calculada pelo Lucene.

Para analisar o texto quanto a frequência de termos, foi utilizada a ferramenta Sobek, esta utiliza um método estatístico, onde ela decompõe os termos que do texto e retorna os termos que tiveram maior frequência. A ferramenta também tem métodos e filtros internos que descartam as palavras frequentes, como artigos e preposições que isoladas não geram sentido. Para utilizar a ferramenta no sistema, foi utilizado um método post, onde o sistema informa aos *webservices* da Sobek uma variável que contém todo o texto a ser analisado e o *webservice* retorna os termos frequentes e a quantidade de vezes que o mesmo incidiu no documento. Vale ressaltar que a opção de número de vezes que o termo apareceu foi implementada pela equipe de desenvolvedores do Sobek durante a implementação deste trabalho e por sugestão da autora. Pois ao utilizar o método texto do Sobek sem integração gráfica é necessário que seja retornado também o número de incidências, do contrário não é possível fazer a diferenciação de relevância dos termos retornados.

A diferença básica entre o software Sobek e a API do Lucene, é que para o Lucene é necessário informar os parâmetros, ou seja, informar quais termos estão sendo buscados no texto, já para o Sobek é informado apenas o texto, e a própria ferramenta retorna os termos mais frequentes.

Banco de dados

O Banco de dados utilizado para armazenar as informações e fazer o processamento dos dados foi o My sql, pois a base que contém os dados de texto que foram avaliados já estava neste formato, sendo necessário apenas incrementar as tabelas de processamento na base sem a necessidade de converter todos os dados para outra linguagem. A base de dados para a análise foi fornecida pela empresa MK Solutions.

As tabelas incrementadas no Banco de dados foram três: tc_termos; tc_setor e tc_classif, A tabela tc_termos contém todos os termos que o usuário adicionou a ontologia de satisfação, bem como os pesos referentes a cada um dos termos; na tabela tc_setor, constam todos os termos que referem a localização do setor, ou seja, todos os termos que se localizados no texto podem ser associados a determinado setor. Essas duas tabelas são as tabelas que o sistema consulta durante a avaliação

do texto quanto a classificação do atendimento (tc_setor), e quanto a satisfação (tc_termos). A terceira tabela armazena todos os atendimentos que já foram avaliados pelo software, esta contém mais atributos: cod_atend que contém o código do atendimento, o campo ava, que armazena a avaliação que cliente emitiu durante o atendimento, a variável tempo que armazena o tempo total do atendimento em minutos. Os demais campos que são variáveis que o sistema calcula e fornece posteriormente, como quantidade de termos relevantes encontrados no texto (campo hits), somatório dos pesos dos termos relevantes (campo peso), setor que o atendimento foi classificado (campo setor) e nível de satisfação calculado pelo sistema para o cliente (sati). Estas são as tabelas que precisam ser inseridas na base de dados para que a ferramenta funcione corretamente.

Parametrização

O sistema utiliza basicamente três variáveis de avaliação do atendimento quanto a satisfação do texto: Tempo mínimo de atendimento, Tempo máximo de atendimento, quantidade de hits (termos) localizados no texto, e peso que os termos somam.

Baseado nos testes iniciais da ferramenta e nas estatísticas fornecidas pela Empresa que forneceu a base de dados, um atendimento que tem tempo muito curto ou que se estende demasiadamente pode deixar o cliente insatisfeito. Desta forma o autor deixou a cargo do usuário parametrizar a variável de **tempo do atendimento**, com um mínimo e máximo, e caso o tempo de atendimento estiver fora da faixa informada será categorizado como cliente insatisfeito.

O **número de hits** representa quantos dos termos relevantes foram localizados no texto, caso o sistema localize mais termos do que o número de hits fornecido pelo usuário o atendimento poderá ser classificado como insatisfatório também. O cálculo do número de hits se da pela fórmula:

$$\sum_{=} (vezes\ que\ o\ termo\ \text{é}\ localizado)$$

Cada termo pode ter um **peso**, e este pode ser considerável ou não, pois em um texto o termo “Cancelamento” pode ter mais relevância e maior peso que o termo “irritado”, por exemplo. Desta forma o sistema faz o somatório de todos os pesos dos

hits localizados no texto, dando assim um peso total do atendimento. Caso o texto ultrapasse o peso estipulado, poderá ser classificado como insatisfatório. O cálculo do peso se dá pela fórmula:

$$peso = \sum_{=} (número\ dos\ hits * peso\ do\ hit)$$

Análise de Satisfação

Com os parâmetros já definidos pelo usuário, nesta fase o sistema começa a fazer as comparações para identificar quão satisfeito o cliente ficou no atendimento. Após alguns testes para identificar a melhor parametrização do sistema, foi possível concluir que para que o atendimento fosse classificado com insatisfatório eram necessários dois parâmetros serem negativos.

1) (tempo de atend.<Tempo mínimo de atend.)
2) (tempo de atend. > Tempo máximo de atend.)
3) (número de hits do atend. > número de hits máximo)
4) (peso dos hits > peso máximo dos hits)

Para que o atendimento seja classificado como insatisfatório, duas das variáveis acima precisa ser falsa.

Por exemplo: Caso o atendimento tenha mais que 45 minutos onde o parâmetro máximo é 40; e tenham sido localizados 25 hits onde o parâmetro definido são 20, o sistema identifica como atendimento insatisfatório. Mesmo que o peso dos hits não tenha atingido o limite estipulado, ele considera apenas duas negações para concluir a insatisfação.

Um atendimento tendo apenas um dos seus requisitos não atendidos não pode ser classificado como insatisfatório, isso porque com base nos atendimentos avaliados em testes não é possível assumir que um texto, que tenha tido um tempo elevado de atendimento seja insatisfatório. Da mesma forma um atendimento que tenha muitos hits não pode ser classificado como insatisfatório, é necessário que os hits encontrados tenham pesos relevantes também.

Análise de Classificação

A análise de classificação visa identificar a área que o atendimento pertence. Então é feita uma busca no atendimento onde os termos elencados são com relação a classificação. Essa busca é feita também através da API do Lucene.

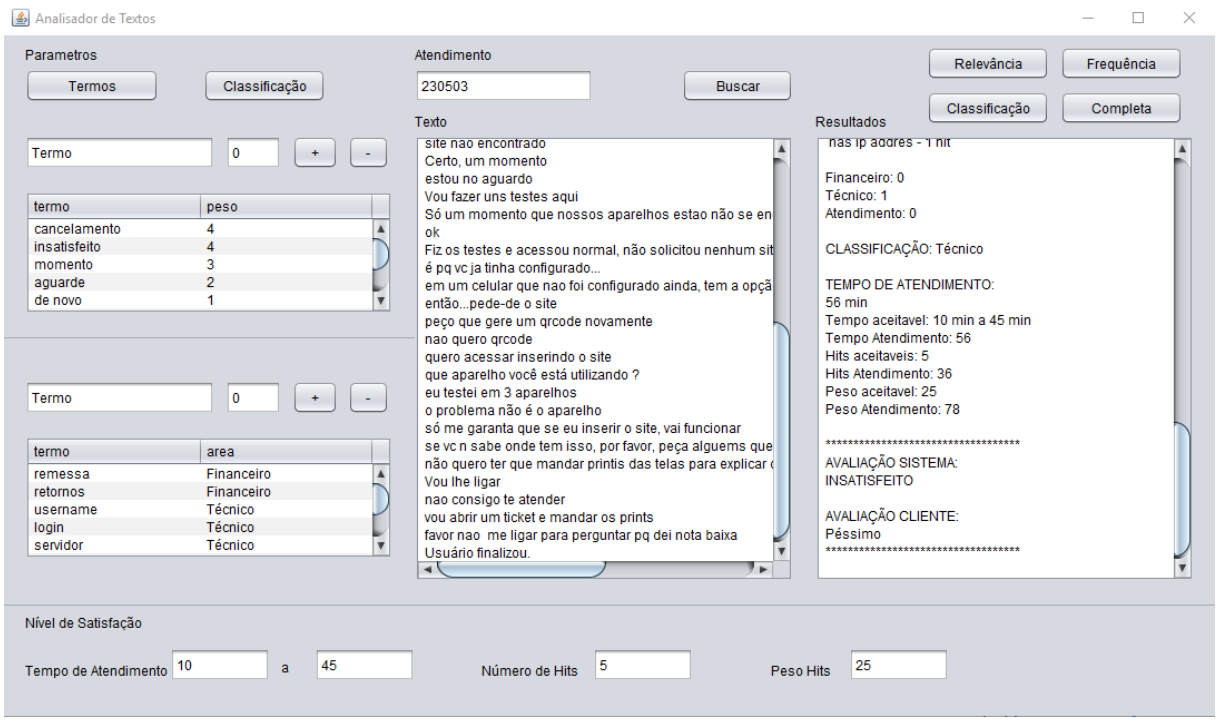
Ao localizar os termos no texto a ferramenta soma quantos hits de cada área foram encontrados. Ao fim o somatório o setor que mais tiver hits é definido como a área do atendimento, segue abaixo pseudocódigo que define as áreas do texto.

if (∑ hits do financeiro > ∑ hits do técnico) *then* área = "Financeiro";
if (∑ hits do técnico > ∑ hits do financeiro) *then* área = "Técnico";
if (∑ hits do financeiro > ∑ hits do atendimento) *then* área = "Financeiro";
if (∑ hits do atendimento > ∑ hits do financeiro) *then* área = "Atendimento";
if (∑ hits do atendimento > ∑ hits do técnico) *then* área = "Atendimento";
if (∑ hits do técnico > ∑ hits do atendimento) *then* área = "Técnico";
if (∑ hits do técnico = ∑ hits do atendimento) *then* área = "Técnico e Atendimento";
if (∑ hits do técnico = ∑ hits do financeiro) *then* area = "Técnico e Financeiro";
if (∑ hits do financeiro = ∑ hits do atendimento) *then* area = "Financeiro e Atendimento";

Por exemplo, se o sistema localizar 35 hits no texto, e destes 15 forem da área financeira, 9 da área técnica e 11 da área de atendimentos, o sistema é classificado como atendimento do setor financeiro. Caso haja empate nos hits encontrados em alguma classificação o sistema informa que o atendimento pertence as duas áreas simultaneamente, isto porque é possível que no mesmo atendimento sejam tratados dois assuntos.

Na figura 6 é possível acompanhar imagem da interface do sistema durante avaliação de um atendimento em especifico, o autor definiu que as análises do atendimento poderiam ser feitas em partes para facilitar na interpretação das rotinas que estão sendo executadas pelo usuário, desta forma pode-se analisar a classificação do atendimento sem depender da análise de satisfação.

Figura 6 - Interface do Sistema desenvolvido



Fonte: do autor

Os parâmetros ficam todos acessíveis em tela, bem como um registro de logs do atendimento, o qual segue exemplo a seguir:

ANÁLISE DE ATENDIMENTO:
****
TERMOS RELEVANTES:
momento - 2 hit
aguarde - 1 hit
novamente - 1 hit
problema - 1 hit
não acessa - 7 hit
não abre - 6 hit
não loga - 6 hit
não autentica - 6 hit
receita - 1 hit
nota - 1 hit
deu certo - 2 hit
funcionou - 1 hit
***HITS**:*
36
PESO DOS TERMOS:
78
TERMOS FREQUENTES:
nao,7 hits
site,5 hits
ja,4 hits
momento,3 hits
vc,3 hits
voce,3 hits
Vou,3 hits
aparelhos,2 hits
aplicativo,2 hits
certo,2 hits
configurado,2 hits
favor,2 hits
ligar,2 hits
mandar,2 hits
nao quero,2 hits
pq,2 hits
qrcode,2 hits
site nao encontrado,2 hits
testei,2 hits
***TERMOS CLASSIFICAÇÃO**:*
nas ip addres - 1 hit
Financeiro: 0
Técnico: 1
Atendimento: 0
CLASSIFICAÇÃO: Técnico
TEMPO DE ATENDIMENTO:
56 min
Tempo aceitável: 10 min a 45 min
Tempo Atendimento: 56
Hits aceitáveis: 5
Hits Atendimento: 36
Peso aceitável: 25
Peso Atendimento: 78
****
AVALIAÇÃO SISTEMA:
INSATISFEITO
AVALIAÇÃO CLIENTE:
Péssimo
****

Todo acompanhamento da análise feita no atendimento pode ser feito através do registro do log, nele consta todos os termos relevantes e termos frequentes que foram encontrados, bem como o número de vezes que cada um é localizado no texto.

A análise quanto a classificação também é demonstrada no extrato de log. Durante a avaliação de satisfação o sistema deixa claro quanto as negativas de parametrização, no exemplo citado, são informadas todos os parâmetros que fizeram com que o atendimento fosse classificado como insatisfatório.

5. RESULTADOS E VALIDAÇÃO

Para fazer a validação do Software foi utilizada uma base de dados de uma empresa que presta atendimentos para clientes via chat online e tickets, esta base de dados contém aproximadamente quinhentos mil atendimentos e tickets e informações de mais de cinco anos de atendimentos.

A mesma forneceu também o dicionário de termos do sistema, estruturando os termos dos principais recursos e quais as palavras chave que precisam ser analisadas.

Considerando a experiência que a empresa tem para avaliar os atendimentos, a mesma validou o dicionário de termos de satisfação de clientes, e informou quais são as diretivas que indicam que um atendimento não está sendo satisfatório para o cliente.

Termos utilizados para Satisfação

O dicionário de termos de satisfação disponibilizada pela empresa conta com os termos e respectivos pesos (Tabela 2), a mesma contempla termos negativos e positivos, ou seja, termos que remetem a satisfação (negativos) e a insatisfação de clientes (positivos):

Tabela 2 - Termos utilizados para primeira massa de testes

Termo	Peso
Insatisfeito	4
Momento	3
Aguarde	2
de novo	1
Bug	3
Erro	3
Retorno	2
Bloqueio	3
indevido	2
novamente	2
problema	4
pagamento	3

bloqueado	3
falha	2
errado	3
remessa	3
boletos	2
procon	4
não acessa	4
não abre	4
transtorno	2
caixas	2
não loga	3
não autentica	3
HD	3
decepcionado	4
irritado	4
péssimo	4
receita	3
nota	2
excluir	2
excluído	2
sumiu	1
deu certo	-10
funcionou	-10
obrigado	-4
cancelamento	4
insatisfeito	4

Fonte: do autor

Todos os termos contidos na tabela foram usados na avaliação dos atendimentos, durante a execução dos testes foi identificado que poderiam ser considerando também termos que remetessem a satisfação dos clientes, como por exemplo: obrigado, funcionou, entre outros. E que atendimentos que apresentem estas características tendem a ser atendimentos satisfatórios. Desta forma, o autor liberou a inserção de pesos negativos, onde mesmo que o atendimento tenha termos

de insatisfação, pode ter termos satisfatórios, e com isso se faz um balanceamento dos termos e níveis de satisfação.

Análise quanto a satisfação

Para avaliação de desempenho e assertividade do software foram separados 3 grupos de textos, um com 50 atendimentos, outro com 100 e o último com 150 atendimentos. Foram utilizados apenas atendimentos que tiveram avaliação emitida pelo cliente, isto para que fosse possível realizar a comparação entre o nível de satisfação que o cliente emitiu e que o Software avaliou.

No primeiro teste foram utilizados os seguintes parâmetros:

- Tempo de atendimento mínimo: 10 min;
- Tempo de atendimento máximo: 45 min;
- Número de hits aceitáveis: 5
- Número de peso dos hits aceitáveis: 25

Neste primeiro teste quando avaliados 50 atendimentos, onde todos os 50 satisfatórios para os clientes. O software acertou 32 atendimentos satisfatórios, gerando uma assertividade de 64%.

No segundo teste, onde os parâmetros se mantiveram os mesmos, porém o número de atendimentos avaliados foi de 100 atendimentos, onde 75 eram satisfatórios e 25 insatisfatórios. O software identificou 36 atendimentos satisfatórios e 18 insatisfatórios, totalizando 54 acertos, ou seja, 54% de assertividade.

No terceiro teste foram analisados 150 atendimentos, destes 91 eram satisfatórios e 60 insatisfatórios. Como resultado foram colhidos 45 atendimentos satisfatórios e 44 insatisfatórios, levando o índice para 59,33% de assertividade.

Para os próximos testes alguns parâmetros do software foram alterados, porém os atendimentos avaliados se mantiveram os mesmos, segue abaixo a nova parametrização:

- Tempo de atendimento mínimo: 10 min;
- Tempo de atendimento máximo: 45 min;
- Número de hits aceitáveis: 15
- Número de peso dos hits aceitáveis: 35

Quarto teste, avaliando 50 atendimentos, constando 50 atendimentos, o software identificou 32 satisfatórios, gerando uma assertividade de 64%.

No quinto teste, onde 100 atendimentos foram processados pela ferramenta, sendo destes 75 satisfatórios, e 25 insatisfatórios, foram identificados 53 como satisfatórios para os clientes, e 6 como insatisfatórios, em um índice de assertividade de 59%.

No sexto teste, com o número de 150 atendimentos, onde 90 eram satisfatórios e 60 insatisfatórios, o software identificou 52 satisfatórios e 40 insatisfatórios, num nível de 61,33% de acertos.

Na tabela 3 segue o resumo dos testes realizados.

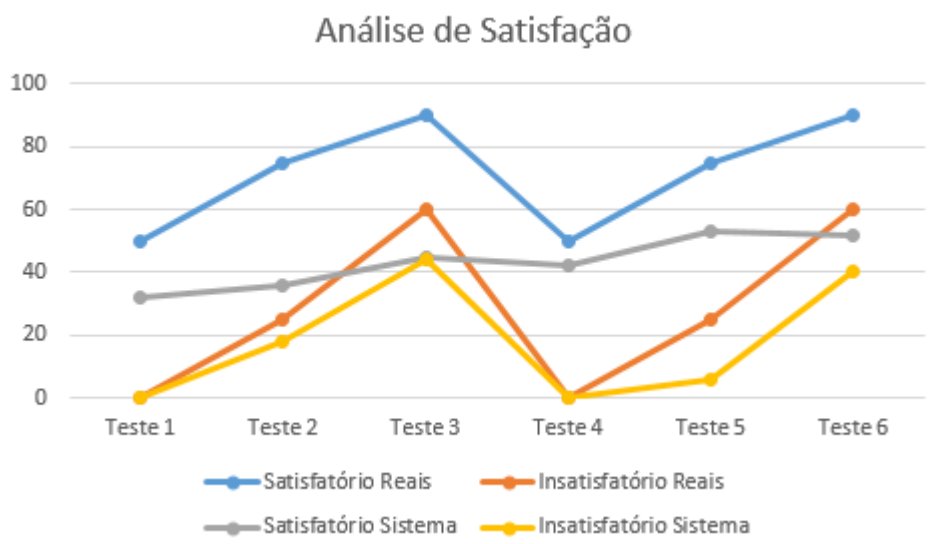
Tabela 3: Análise de resultados de satisfação

	Análise real feita por um técnico			Resultado gerado pelo sistema				
	Avaliados	Satisfatório	Insatisfatório	Satisfatório		Insatisfatório		Percentual de acertos
				Acertos	Erros	Acertos	Erros	
Teste 1	50	50	0	32	18	0	0	64%
Teste 2	100	75	25	36	39	18	7	54%
Teste 3	150	90	60	45	45	44	16	59,33%
Teste 4	50	50	0	42	8	0	0	64%
Teste 5	100	75	25	53	22	6	19	59%
Teste 6	150	90	60	52	38	40	20	61,33%

Fonte: do autor

Na figura 7 é possível identificar graficamente o nível de assertividade do sistema se comparando o resultado dos testes.

Figura 7 - Análise de satisfação



Fonte: do autor

O sistema durante os seis testes efetuados mostrou uma assertividade média de 60,27% dos atendimentos, com a utilização da ontologia. Notou-se que quanto mais refinada a ontologia melhor será a média de assertividade do software.

Análise de classificação

Para a classificação do atendimento foram utilizadas basicamente três classificações: Técnico, Financeiro e Atendimento. Foram realizados 3 testes, onde o sistema classificou no total 300 atendimentos diferentes. Na tabela 4 é possível acompanhar os resultados.

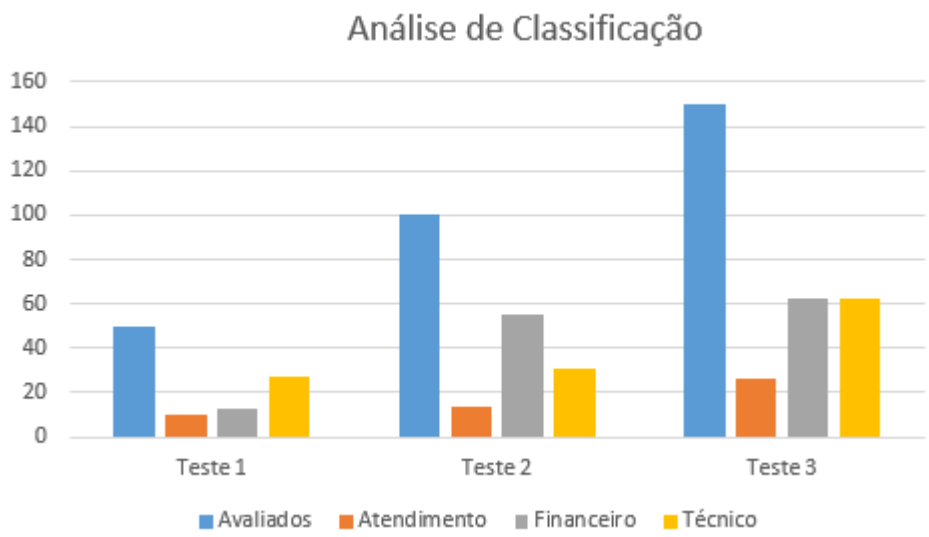
Tabela 4: Análise de resultados de classificação

	Avaliados	Atendimento	Financeiro	Técnico
Teste 1	50	10	13	27
Teste 2	100	14	55	31
Teste 3	150	26	62	62

Fonte: do autor

Na figura 8 pode-se acompanhar o número de atendimentos por área de negócios.

Figura 2 - Análise de classificação



Fonte: do autor

A quantidade de atendimentos por classificação não é constante, visto que no primeiro teste os atendimentos com classificação técnica formavam o maior grupo, já no segundo teste essa categoria ficou com menos atendimentos do que os que compõem a categoria de atendimentos financeiros. No terceiro teste o grupo de atendimentos técnicos e financeiros se igualou. Isso ocorre porque os atendimentos avaliados nos três testes são todos atendimentos diferentes. Porém é notável que os setores mais procurados para atendimento são os técnicos e financeiros, restando um percentual de atendimentos de outras áreas muito baixo.

6. CONCLUSÃO

Neste trabalho de conclusão de curso, foram analisadas diversas bases bibliográficas, aprofundando os conhecimentos da autora no processo de descoberta de conhecimento, abrangendo a mineração de dados e de textos, tópicos estes que contribuem para diversas áreas.

Documentando os passos da mineração de dados e de textos foi possível compreender todo o processo que envolve a descoberta do conhecimento, desde a fase de análise do texto até a avaliação e validação dos resultados. Foi possível também compreender pontos de vista de diversos autores quanto ao tema, estudar e comparar as ferramentas que estes autores desenvolveram e utilizaram para a descoberta de conhecimento, cada um deles utilizando técnicas diferentes.

A base teórica foi estudada para projetar uma ferramenta capaz de identificar o perfil de satisfação de clientes. O autor optou por montar um algoritmo de identificação de grau de satisfação baseado em parâmetros que definem a satisfação em um atendimento em formato de texto, e em e testes realizados, porém este desenvolvimento só foi possível após estudos das técnicas e trabalhos já desenvolvidos na área de mineração de dados e textos.

O software mostrou-se eficiente quanto a análise de satisfação quando atingiu na média uma assertividade de 60,27% nos atendimentos avaliados. Notou-se que o sistema acompanhou ligeiramente a linha de satisfação que os clientes emitem nos atendimentos, sendo possível utilizar a ferramenta para avaliar atendimentos os quais os clientes não emitiram nenhuma avaliação. Durante o desenvolvimento e testes a autora identificou que para um nível de assertividade maior os parâmetros como ontologia de termos e somatório total de pesos podem ser aprimorados, desta forma deixou os campos em forma de parametrização para que o próprio usuário possa informar os níveis que julgar corretos para os atendimentos em avaliação.

Com relação a classificação de atendimento a ferramenta conseguiu classificar os atendimentos conforme os termos informados, porém também é necessário um grande dicionário de termos para que os mesmos possam ser classificados de forma fiel ao atendimento.

Vale ressaltar que a ferramenta já está em uso como projeto piloto na própria empresa que disponibilizou a base de dados, para tal é necessário informar a range de atendimentos que deseja avaliar.

Como trabalho futuro e aprimoramento da ferramenta, poderia ser informado o código do cliente, para que o Software gerasse um histórico e perfil de todos os atendimentos já realizados, podendo assim traçar o perfil de atendimentos prestados a este cliente, quanto a classificação e nível de satisfação que o mesmo emite para a Empresa. Esta era uma das propostas para o software, porém devido à complexidade das demais rotinas desenvolvidas na aplicação fica apenas como sugestão de melhoria no trabalho.

Ainda como sugestão de trabalho na área de satisfação de clientes, destaca-se a analise considerando os estados afetivos, onde além de avaliar a satisfação do cliente poderia considerar o seu estado afetivo, por exemplo se o cliente estava com raiva, se estava chateado, se estava feliz com o atendimento, entre outros estados. Desta forma poderia diagnosticar o atendimento considerando dicionários de estados afetivos positivos e negativos.

REFERÊNCIAS

Amo, S. Técnicas de Mineração de dados. Universidade de Uberlândia. http://www.deamo.prof.ufu.br/arquivos/JAI-cap5.pdf. Acessado em outubro/2016

ARANHA, C.; PASSOS, E. A. Tecnologia de Mineração de Textos, Revista Eletrônica de Sistemas de Informação, VOL 5 N°2, 2006.

BULSING, G. M. Ferramenta para extração de dados semi-estruturados para carga de um Big Data. UNISC. Santa Cruz do Sul, 2013.

CABENA, P; HADJINIAN, P; STADLER, R; JAAPVERHEES; ZANASI, A. Discovering Data Mining: From Concept to Implementation. Prentice Hall, 1998.

CHIAVENATO, I. Gestão de pessoas.Rio de Janeiro: Editora Atlas,2005.

COBRA, M. Marketing básico: uma perspectiva brasileira. São Paulo: Atlas, 1997.

CORDEIRO, A. D. Gerador Inteligente de Sistemas com Auto-aprendizagem para Gestão de Informações de Conhecimento. PhD thesis, Universidade Federal de Santa Catarina, Departamento de Engenharia de Produção, 2005.

DEMO, G.; Fogaça, N.; Ponte V.; Fernandes T.; Cardoso H. Marketing de relacionamento (CRM): estado da arte, revisão bibliométrica da produção nacional de primeira linha, Institucionalização da pesquisa no brasil e agenda de pesquisa, 2014

EBECKEN, N.F.F.; LOPES M. C. S. ; COSTA M. C. “Mineração de textos,” in Sistemas Inteligentes: Fundamentos e Aplicações, 1st ed., S. O. Rezende, Ed. Manole, 2003, ch. 13, pp. 337–370.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P.. From data mining to knowlegde Discovery in database. AI Magazine, v.12, n.3, p.37, 1996.

HAND, D; MANNILA, H; SMYTH, P. Principles of Data Mining. MIT Press, 2001.

IBM. https://www.ibm.com/developerworks/br/java/library/os-apache-lucenesearch/. Acessado em setembro de 2016

JUNIOR, R. C. Uso da Mineração de Dados na identificação de alunos com perfil de evasão do Ensino Superior. UNISC, 2015.

KASAMA, D. Y., ZAVAGLIA, C.; AlLMEIDA, G. M. d. B. Do termo à estruturação semântica: Representação ontológica do domínio da Nanociência e Nanotecnologia utilizando a Estrutura Quali. Linguamática (2010).

KLEMANN, M.; REATEGUI, E.; LORENZATTI, A. O Emprego da Ferramenta de Mineração de Textos SOBEK como Apoio à Produção Textual . XX Simpósio Brasil De Informática Na Educação: Florianópolis ,2009.

KOTLER, P.;KELLER,L. Administração de marketing. 12. ed. São Paulo: Pearson Prentice Hall, 2007.

LOPES, M. C. S. Mineração de dados Textuais utilizando técnicas de Clustering para o Idioma Português. Rio de Janeiro, 2004 (Tese de Doutorado).

MACARINI, Z. C. Qualidade no atendimento ao cliente como diferencial das cooperativas de crédito: um estudo na Sicredi Sul Santa Catarina. Criciúma, 2014.

LUCENE. http://lucene.apache.org/core/. Acessado em setembro de 2016

MORAIS, E. A. M.; AMBRÓSIO, A. P. L. Mineração de Textos. Relatório Técnico. Goiás, 2007.

MOURA, M.F. Proposta de utilização de mineração de texto para seleção, classificação e qualificação de documentos. Embrapa Informática e Agropecuária, 2004, ISSN 1677-9274, 2004.

OLIVEIRA J.; FRANÇA C. T. Análise de Sentimento de Tweets Relacionados aos Protestos que ocorreram no Brasil entre Junho e Agosto de 2013. Rio de Janeiro, 2013.

ORENGO, V. M; HUYCK, C. A stemming algorithm for the portuguese language.p. 186–193, 2001.

PIMENTA, M. R. Big data e controle da informação na era digital: tecnogênese de uma memória a serviço do mercado e do Estado. Tendências da Pesquisa Brasileira em Ciência da Informação, v. 6, n. 2, 2013.

REATEGUI, E.; EPSTEIN, D.; LORENZATTI, A. ; KLEMANN, M. Sobek: a Text Mining Tool for Educational Applications. In: International Conference on Data Mining, 2011, Las Vegas, Estados Unidos. Anais do DMIN '11, 2011. p. 59-64.

REICHHELD, F. F; SASSER Jr, W. E. Zero defections - quality comes to services. Harvard Business Review(September-October), p. 107-111, 1990.

RH, http://www.rhportal.com.br/artigos/rh.php?idc_cad=ejwf5ms66. Acessado em: fevereiro de 2016.

RIBEIRO, L.S. Uma abordagem semântica para seleção de atributos no processo de kdd (2010). Disponível em: http://www.ppgi.di.ufpb.br/?p=1691. Acessado em outubro 2013.

ROSSI, CAV; SLONGO LA. Pesquisa de satisfação de clientes: O Estado-da-arte e proposição de um método brasileiro. Revista de Administração Contemporânea, v.2, n.1, p.101-25, 1998.

SAUSEN, F. J. Projeto e desenvolvimento de um sistema para definição de aspectos e análise de sentimentos em textos, 2015.

SILBERCHATZ, A; KORTH, H; SUDARSHAN, S. Sistema de Banco de Dados. ELSEVIER, 5 edition, 2006.

SILVA, R.A.P. Qualidade no atendimento e serviços para satisfação do usuário/cliente: um olhar para a administração pública. João Pessoa, 2012.

SOBEK, http://sobek.ufrgs.br/about.html. Acessado em: maio de 2016.

SOLUTIONS, MK. http://mksolutions.com.br. Acessado em: janeiro de 2016.

SOWA, J. F. Knowledge representation; logical, philosophical, and computational foundations. Pacific Grove: Brooks-Cole, 2000. 594p.

SUPTITZ, L. I. Aplicação de mineração de texto com o apoio de ontologias para extração de conhecimento em bases de dados textuais, 2013.

SWARTOUT, W. ; TATE, A. Guest editors' introduction: ontologies. *IEEE Intelligent Systems*, v. 14, n.1, p. 18-19, Jan. 1999.

WEKA. http://www.cs.waikato.ac.nz/~ml/weka/. Acessado em: fevereiro de 2016.

WIVES, L. Tecnologias de descoberta de conhecimento em textos aplicadas à inteligência competitiva. Exame de Qualificação EQ-069, PPGC-UFRGS, 2002.

ZHOU, Z.-H. Three perspectives of data mining. Artificial Intelligence Journal, p. 139–146, 2003.