UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGIA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

JONATAS BRITO DE SOUSA

BIG DATA: Análise de Sentimento em Dados de Pesquisa de Opinião utilizando o Framework GridGain e Processamento em Memória

São Luís
2014

JONATAS BRITO DE SOUSA

BIG DATA: Análise de Sentimento em Dados de Pesquisa de Opinião utilizando o Framework GridGain e Processamento em Memória

Monografia apresentada ao curso de Ciência da Computação da Universidade Federal do Maranhão, **como parte dos requisitos necessários** para obtenção do grau de Bacharel em Ciência da Computação.

Orientadora: Prof.ª. Dr. Simara Viera da Rocha

São Luís
2014

Sousa, Jonatas Brito de
Big Data: Análise de Sentimento em Dados de Pesquisa de Opinião utilizando o Framework GridGain e Processamento em Memória / Jonatas Brito de Sousa. – São Luís, 2014.

76 f.

Impresso por computador (Fotocópia).

Orientador: Simara Viera da Rocha.

Monografia (Graduação) – Universidade Federal do Maranhão, Curso de
Ciência da Computação, 2013.

1. Big Data. 2. Análise de Sentimento em Dados. 3. Framework GridGain. 4. Processamento em Memória. 5. Algoritmo MapReduce. I. Titulo.

CDU 004.63

JONATAS BRITO DE SOUSA

BIG DATA: Análise de Sentimento em Dados de Pesquisa de Opinião utilizando o Framework GridGain e Processamento em Memória

Monografia apresentada ao curso de Ciência da Computação da Universidade Federal do Maranhão, **como parte dos requisitos necessários** para obtenção do grau de Bacharel em Ciência da Computação.

Aprovada em: 16 / 12 /2014

BANCA EXAMINADORA

Simara V. da Rocha

Prof.ª Simara Vieira da Rocha, Dra.
(Orientadora)

Maria Auxiliadora Freire

Prof.ª Maria Auxiliadora Freire, M.Sc.
(Membro da Banca Examinadora)

Portela

Prof. Carlos Eduardo Portela Serra de Castro, M.Sc
(Membro da Banca Examinadora)

AGRADECIMENTOS

À Deus, pela graça de vida e amor infinito.

À minha mãe, por sempre me motivar a buscar meus sonhos e lutar por eles.

Ao meu pai, pelo incentivo, paciência e amor.

À minha orientadora, professora Simara, pelo apoio, paciência, competência e dedicação.

Aos meus amigos, pela força nas horas que precisei.

A todos aqueles que contribuiriam direta ou indiretamente para realização deste trabalho.

“In God we trust. All others must bring data.”

W. Edwards Deming

RESUMO

Big Data apresenta-se como uma das principais tecnologias da atualidade. Métodos simples e eficientes para realizar big data análise são responsáveis por diversos avanços em análise comercial, dados médicos e comportamental. Este trabalho apresenta um método para realizar análise de sentimento em dados provenientes de pesquisa de opinião, utilizando o *framework* GridGain. O método é projetado para lidar com dados que possuem características big data, ou seja, permite a análise de grandes volumes de informações, por meio do uso de processamento paralelo e MapReduce. O algoritmo de MapReduce implementado pelo *framework* GridGain, é um algoritmo para processamento paralelo consolidado, pois permiti processamento de grandes volumes de informações de fontes variadas sobre uma estrutura de processamento existente em rede ou por meio de múltiplos nós de processamento em uma máquina. A análise dos dados por meio da metodologia, aponta uma nova maneira de extrair informação útil dos dados de pesquisa de opinião, sob uma ótica diferente das tradicionais análises matemáticas empregadas, permitindo abordar os dados sobre outra perspectiva de análise, a dos sentimentos expressos pelas respostas dos entrevistados.

Palavras-Chave: Big Data, GridGain, Data Análise, MapReduce, Análise de Sentimento e Pesquisa de Opinião.

ABSTRACT

Big Data is presented as one of today's leading technologies. Simple and efficient methods to make big data analysis are responsible for many advances in business analysis, medical and behavioral data. This paper presents a method to perform sentiment analysis on data from survey, using GridGain framework. The method is designed to handle data that have big data characteristics, or allows analysis of large volumes of information, through the use of parallel processing and MapReduce. The MapReduce algorithm implemented by GridGain framework, an algorithm is consolidated to parallel processing, since processing large volumes enable a variety of sources of information on an existing processing structure in a network or across multiple processing nodes on a machine. Data analysis by methodology indicates a new way to extract useful information from the survey data, in a different light the traditional mathematical analysis employed, allowing addressing data on another perspective analysis, the feelings expressed by the answers of respondents.

Keywords: Big Data, GridGain, Data Analysis, MapReduce, Sentiment Analysis and Opinion Research.

LISTA DE FIGURAS

SUMÁRIO

1 Introdução

Para (LIU, 2010) a expressão de opiniões é algo inerente a natureza humana, é maneira com a qual imprimimos nossas crenças, escolhas tomadas, juízo de caráter e a forma como vemos e avaliamos o mundo.

Opiniões e conceitos relativos como sentimentos, avaliações e emoções são temas de estudo da análise de sentimento e mineração de opinião (LIU, 2012). Uma área cada vez mais importante, com a vasta quantidade de informação gerada pelas mídias digitais; tais como *reviews*, fóruns de discussões, blogs, microblogs, redes sociais. Sendo assim, pela primeira vez na história, temos à disposição uma imensa quantidade de dados de opinião em formato digital (CHEN et al., 2014). É necessário o uso de métodos e técnicas desenvolvidos especialmente para esse conjunto cada vez maior e variado de informações opinativas.

Análise de sentimento tem se destacado como uma das mais ativas áreas de pesquisa no processamento da linguagem natural. É também muito estudada em mineração de dados, mineração *web* e mineração em texto (LIU, 2012). É notável um crescente interesse de outras áreas, como por exemplo, ciências sociais, na análise de sentimento e mineração de opinião, tendo em vista que estão imersas em dados de opinião e sentimento. Como apontado em (LIU, 2012), os sistemas de análise de sentimento encontraram suas aplicações em quase todos os negócios e no domínio social.

Em (OLIVEIRA, 2013) é observado que todo dia, cerca de 15 Petabytes[1] de dados estruturados e não-estruturados são gerados no mundo, segundo projeções da IBM. Essa montanha de informação inclui mensagens trocadas a partir de dispositivos móveis, e-mails, fotos, vídeos, compras pela web, planilhas, textos, vídeos e outros (IDC, 2011).

Companhias estão diante de uma mina de ouro e começam agora a entender como forjar inteligência de negócios a partir dessa matéria-prima (OLIVEIRA, 2013). É a era do *big data* que, segundo especialistas, está sobre nós e deve ganhar impulso nos próximos anos, movida pelo crescimento do poder computacional e do surgimento de novas fontes geradoras de informação, como dados de dispositivos móveis e das redes sociais (CHEN et al., 2014).

Segundo (OLIVEIRA, 2013; CHEN et al., 2014), *big data* não se refere ao mero armazenamento de grandes volumes de dados. Posto que volume, variedade e velocidade de

[1] Petabyte é uma unidade de medida de informação, equivalente a 10^{15} bytes.

geração de dados são os elementos que compõem *big data*. O valor do conceito está na possibilidade de localizar, peneirar e analisar informações úteis a partir de diferentes fontes e em algumas situações em tempo real. Diferentemente do *Bussines Intelligence* (BI), que analisa o passado, a tecnologia *big data* permite previsão de tendências futuras, balizando e agilizando as tomadas de decisões imediatas (OLIVEIRA, 2013).

A principal proposta de *big data* é ajudar empresas a decidir com base em evidências e análises contínuas. Um exemplo abordado em (CHEN, M. et al., 2014) é a área de seguros, na qual a análise de fraudes poderia ser imensamente melhorada, minimizando-se os riscos com a utilização, por exemplo, de análise de dados que estão fora das bases estruturadas das seguradoras, como os dados que estão circulando diariamente nas mídias sociais.

Portanto, *big data* é uma realidade. Segundo estudo do *McKinsey Global Institutem*, publicado em 2011, 15 dos principais 17 setores econômicos dos Estados Unidos possuíam mais dados armazenados por companhia do que a Biblioteca do Congresso Americano naquele ano que, em abril de 2011, possuía 235 terabyte[2] de dados (McKinsey Global Institute, 2011). No Brasil, *big data* tem crescido ao ponto de o Governo Federal brasileiro, por meio do Instituto Getúlio Vargas, no ano de 2013, desenvolveu uma ferramenta de análise *big data*, que permitiu ao governo analisar o mercado e determinar o menor preço de produtos, gerando uma economia de 860 milhões de reais em gastos com compras governamentais, apenas no Ministério da Educação (CORRÊA, 2013).

Por outro lado, a análise de sentimento é um campo de análise em franca expansão e amadurecimento muito por conta do grande volume de dados de opinião disponíveis em formato digital. Contudo, durante o levantamento e estudo da bibliografia recomendada, não encontrou-se uma metodologia de análise de sentimento que utiliza-se do conceito do processamento paralelo em memória que permite desempenho superior, mesmo em *hardware* de baixo custo.

Finalmente, o uso de um *framework* tem por finalidade, além de permitir que a metodologia proposta seja migrada de plataforma de forma simples, como também simplificar a tarefa de análise de dados (no caso deste trabalho, palavras de sentimento), tornando o método eficiente e intuitivo.

[2] Terabyte é uma unidade de medida de informação, equivalente a 10^{12} bytes.

1.2 Objetivos

1.2.1 Objetivo Geral

Apresentar um método para análise de sentimento em *big data* utilizando o *framework* GridGain, a partir de pesquisas de opinião pública.

Com utilização desse método propõem-se realizar uma análise diferente da tradicional mineração de dados estatísticos, comum em pesquisas de opinião, e demonstrar como as tecnologias *big data* podem otimizar o processamento de grandes volumes de informações mesmo em hardware de baixo custo.

1.2.2 Objetivos Específicos

- Analisar conceitos de *big data* e suas tecnologias.
- Elencar as principais plataformas *open source* aplicadas para *data analysis*.
- Analisar conceitos de análise de sentimento e mineração de opinião.
- Perscrutar o *framework* GridGain.
- Desenvolver uma metodologia para realizar a análise de sentimento em uma base de dados proveniente de pesquisas de opinião.

1.3 Trabalhos Relacionados

Uma metodologia para análise de sentimento em nível de documento visa classificar a opinião total acerca do documento (sobre uma única entidade) baseado em classes, normalmente positiva e negativa (PANG et al., 2002; TURNEY, 2002). Esse tipo de problema é, provavelmente, o mais abordado na literatura disponível, a qual traz trabalhos que tratam do mesmo objetivo abordado pelo método proposto.

Em (PANG et al., 2002; TURNEY, 2002), foram propostos algoritmos para classificação de sentimento em nível de documento, que visa classificar a opinião total sobre o documento como um sentimento positivo ou negativo. O algoritmo proposto em (TURNEY, 2002) realiza a classificação de sentimento baseado em alguns padrões de sintática fixos, que são mais comumente usados para expressar opinião. Em ambos os trabalhos foram abordados

os domínios dos *reviews online* e não consideram uma opinião neutra. Em geral, a maioria dos trabalhos ignora as opiniões neutras, pois a classificação deste tipo de opinião é difícil.

Um algoritmo baseado em léxico foi proposto por (HU e LIU, 2004) para realizar classificação de sentimento em nível de aspecto, mas o método também pode determinar a orientação do sentimento em nível de sentença. Foi com base na geração de sentimento léxica, usando estratégia de *bootstrapping,* com uma palavra de sentimento positiva ou negativa, que servia como semente e a partir dela seus sinônimos e antônimos buscado no *WordNet*. A orientação do sentimento foi determinada somando a pontuação da orientação das palavras de sentimento numa sentença. Uma palavra positiva dava ao sentimento uma pontuação de +1 e uma palavra negativa dava ao sentimento uma pontuação -1. Palavras de negação ou contrariedade (mas, contudo, etc.) também foram consideradas.

Finalmente, uma abordagem similar foi adotada por (KIM e HOVY, 2004, 2007). O método proposto para compilar o sentimento do documento também utilizava uma pontuação dada a orientação das palavras de sentimento. Contudo, neste trabalho, foram consideradas múltiplas pontuações para as palavras na sentença. Em (KIM e HOVY, 2006) foi aplicado aprendizado supervisionado para identificar tipos específicos de opinião.

1.4 Organização do Trabalho

Este trabalho está organizado em mais 4 capítulos.

No capítulo 2 será apresentada a fundamentação teórica necessária para compressão deste trabalho. São abordados: *big data*, tecnologias relacionadas, análise *big data*, aplicações de *big data*, plataformas *open source* utilizadas para análise de dados, com destaque para o *middleware GridGain*, análise de sentimento, aplicações para análise de sentimento, definições de opinião e níveis de análise de sentimento e técnicas para análise de sentimento.

No capítulo 3, será descrita a metodologia proposta como objetivo deste trabalho. Essa metodologia é demonstrada por meio de um estudo de caso envolvendo dados de pesquisa de opinião.

Por fim no capítulo 4, serão feitas serão feitas conclusões a respeito desta monografia, bem como apresentado sugestões de trabalhos futuros.

2 Fundamentos Teóricos

Este capítulo apresenta a fundamentação teórica utilizada no desenvolvimento deste trabalho e necessária para compreensão dos métodos utilizados para alcançar os objetivos esperados pelo mesmo. Aborda-se big data, tecnologias relacionadas, análise big data, aplicações de big data, plataformas *open source* utilizadas para *data analysis* e o *middleware GridGain*, análise de sentimento, aplicações para análise de sentimento, definições de opinião, níveis de análise de sentimento e técnicas para análise de sentimento.

2.1 Big Data

Big data é um conceito abstrato. Além de claramente descrever um volume massivo de dados também possui outras características que determinam a diferença entre ela mesma e o "*massive data*" ou "*very big data*" (CHEN et al., 2014).

Em (BEYER e LANEY, 2012), *big data* é definida como grande volume, grande velocidade e/ou alta variedade de informações importantes que requerem novas formas de processamento que permitam capacidade de decisão apurada, descobertas de *insight* e otimização de processamento. A essa definição – conhecida como três Vs – ainda foram acrescentados um quarto v (veracidade) e um quinto v (valor), que tiveram como objetivo destacar um novo ponto ligado à confiabilidade e incerteza acerca dos dados (e da análise dos mesmos) e o último para ressaltar a importância dos Vs anteriores e de big data no cenário mundial, dada a necessidade de seu estudo e aplicação (WARD e BARKER, 2013).

Em 2010, segundo apontado por (CHEN et al., 2014), Apache Hadoop definiu *big data* como conjunto de dados que não podem ser capturados, gerenciados e processados por computadores comuns sem um escopo aceitável. Outra definição, feita por (McKinsey Global Institute, 2011), refere-se à *big data* como conjuntos de dados cujo tamanho está além da capacidade de ferramentas típicas de *software* de banco de dados para capturar, armazenar, gerenciar e analisar.

A definição dada por (McKinsey Global Institute, 2011), implica de duas maneiras sobre o conceito de big data: primeiro que o volume dos conjuntos de dados considerado padrão para big data estão mudando e podem superar a atual capacidade tecnológica de lidar com eles; segundo que esse volume de conjuntos de dados padrão para big data são diferentes entre si, conforme a aplicação. A partir das definições anteriormente apresentadas, pode-se presumir

que big data não se refere apenas a volume de dados, mas principalmente a capacidade de manipulá-la.

A verdadeira questão não diz respeito à coleta e armazenamento de grandes quantidades de dados, mas sim ao que fazer com eles. Sob essa perspectiva, a análise de *big data* tornou-se uma área de pesquisa em evidência. Muito disso se deve à necessidade de modelos e metodologias distintos para uma grande variedade de tipos de dados coletados, pré-processados, armazenados e enviando para análise por meio da aplicação, um processo para o qual existem muitas maneiras e ferramentas disponíveis (CHEN et al., 2014). Algumas delas serão abordadas neste trabalho.

2.1.1 Tecnologias Relacionadas a Big data

Para um entendimento mais profundo sobre big data é preciso entender-se as tecnologias que são relacionadas a ela, tais como: Computação em Nuvem, Internet of Things (IoT), Data Centers e Hadoop.

- **Computação em Nuvem**: a computação em nuvem possui uma relação intrínseca com big data. Big data é o objeto da operação de computação intensiva e responsável pela massa de informações que ocupam cada vez mais espaço nos sistemas de computação em nuvem. O principal objetivo da computação em nuvem é usar computação de alto desempenho e recursos de armazenamento sob gerenciamento concentrado e, também, prover aplicações big data com capacidade de análise refinada (CHEN et al., 2014).

  Apesar da computação em nuvem e big data possuírem muitos aspectos em comum, elas diferem em dois pontos, segundo apontado por (CHEN et al., 2014). Primeiramente, ambos são conceitos diferentes: enquanto a computação em nuvem transforma a arquitetura de TI (Tecnologia da Informação), big data é o principal responsável por influenciar na tomada de decisões. No entanto, big data depende da computação em nuvem como a estrutura responsável por prover um bom funcionamento. Segundo que big data e computação em nuvem possuem diferentes consumidores alvo. Computação em nuvem é a tecnologia visa a sua utilização como uma solução de TI avançada, enquanto big data foca as operações do negócio auxiliando na tomada de decisões.

- **Internet of Things (IoT):** no paradigma do IoT, uma grandessíssima quantidade de sensores conectados à rede estão embutidos em vários dispositivos e máquinas no mundo real. Esses sensores implantados permitem a coleta de vários tipos de informações, como condições climáticas, dados geográficos, dados astronômicos e dados logísticos. São apenas alguns dos exemplos de dados que podem ser coletados (CHEN et al., 2014). Dispositivos móveis, repartições públicas, meios de transporte e sensores domésticos são exemplos de equipamentos que podem ser usados para aquisição de dados em IoT.

Como discutido em (CHEN et al., 2014) a big data gerada por IoT tem características diferentes se comparado com big data em geral por conta dos tipos de dado coletados, dos quais os tipos mais clássicos incluem heterogeneidade, variedade, características desestruturadas, ruído e alta redundância. Atualmente, IoT não representa o tipo predominante de big data gerado, mas segundas estimativas da HP (Hewlett Packard), por volta do ano 2030, a quantidade de sensores irá atingir um trilhão, e então os dados do IoT será a parte mais importante de big data. Um relatório da Intel aponta que big data gerado por IoT possui três características concernentes com o paradigma big data: vários terminais gerando massiva quantidade de dados, dados gerados são semiestruturados ou desestruturados, dados do IoT são úteis apenas quando analisados.

- **Data Centers:** no modelo de big data, os data centers não somente são a plataforma para armazenamento de dados, mas também adquirem mais responsabilidades, tais como: aquisição de dados, gerenciamento de dados e alavancamento dos valores dos dados e funções (CHEN et al., 2014). Diferente do indicado pelo nome, data center diz mais respeito sobre "*data*" do que "*center*", visto que essa estrutura possui quantidade massiva de dados e os organiza e gerencia de acordo com objetivo e meta de desenvolvimento propostos. O crescimento repentino e acelerado de aplicações big data foi responsável por uma revolução e constante aperfeiçoamento dos data centers. A passagem de mais responsabilidades de big data para os *data centers* permite que as instituições personalizem a análise dos dados existentes, descobrindo problemas na operação do negócio e desenvolvendo soluções big data voltada a cada problema especifico.

- **Hadoop:** o funcionamento da tecnologia Hadoop será abordada em detalhes no tópico 2.1.4.2. Por enquanto destacam-se apenas informações acerca do mercado em torno da tecnologia. Atualmente, Hadoop é amplamente usado por aplicações big data

no mercado, principalmente por aplicações que realizam processamento complexo *offline* de big data em lote, tais como filtros spam, indexação de pesquisas, otimização de conteúdo, processamento de conteúdo de *feeds* e previsão e diagnóstico falhas. Por conta disso, um número considerável de pesquisadores acadêmicos tem se dedicado a pesquisas voltadas para Hadoop. Segundo declarado (CHEN et al., 2014), em Junho de 2012, Yahoo mantinha Hadoop em 42 mil servidores em quatro data centers para suportar produtos e serviços. No mesmo mês o Facebook anunciou que seu *cluster* Hadoop permite o processamento 100 Petabytes de dados, que mantêm um crescimento de 0.5 PB por dia, como divulgado em novembro de 2012. Muitas empresas provem execução e suporte comercial de Hadoop, incluindo Cloudera, IBM, MapR, EMC e Oracle.

2.1.2 Análise Big Data

Uma cadeia de valor big data consiste dos passos necessários para transformar o dado em informação útil no processo de tomada de decisão (BEYER e LANEY, 2012). De forma geral, essa cadeia de valor é apresentada na Figura 2.1. Como pode-se observar, é formada por 4 fases: Fontes de Dados (redes sociais, informações de rede, informações de faturamento, etc.); Coleta de Dados (inclui diferentes tecnologias para captura de dado); Armazenamento e Gerenciamento de Dados (tecnologias de armazenamento que permitam acesso rápido e consistente ao dado, por exemplo Hadoop) e Análise de Dados (aplicação de técnicas de análise de dados em busca de informação de valor, que serão abordadas posteriormente nesse tópico), propondo previsões e resultados em formato simples e usável, como por exemplo gráficos.

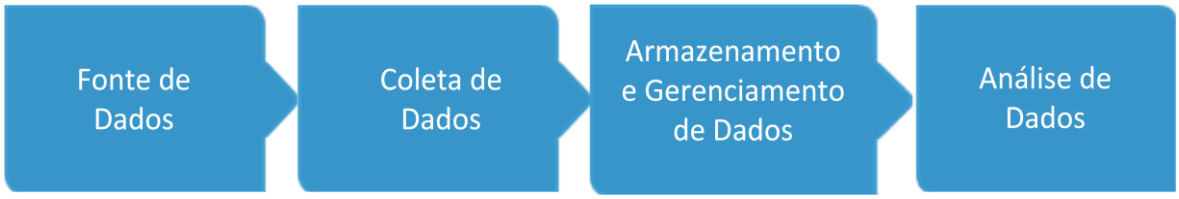


Figura 2.1: Cadeia de valor big data. Fonte: (BEYER e LANEY, 2012).

Neste trabalho, aborda-se apenas a etapa de análise de dados, pois esta compreende de uma variedade muito maior de arranjos, ferramentas e metodologias relacionados com os temas de interesse desenvolvidos posteriormente neste trabalho, o qual aborda alguns dos principais conceitos ligados a análise de dados.

A análise de big data envolve, principalmente, métodos analíticos para dados tradicionais e big data, arquitetura analítica para big data e *software* usado para mineração e análise de big data. Análise de dados é fase final e mais importante em valor na cadeia de big data, com propósito de extrair valores úteis, provendo sugestões e/ou decisões. Dessa forma, pode-se visualizar a análise big data sob duas perspectivas: as orientadas à decisão (mais semelhante ao tradicional *business intelligence*) e orientadas à ação (utilizadas para se obter resposta rápida). Diferentes níveis de valores potenciais podem ser gerados através da análise de conjuntos de dados em diferentes níveis (McKinsey Global Institute, 2011). Contudo, análise de dados é uma ampla área que muda com frequência e é extremamente complexa.

Análise de dados tradicional consiste em utilizar métodos apropriados para analisar volume maciço de dados, para concentrar, extrair e refinar dados discretos úteis no lote conjunto de dados caóticos, e identificar uma lei inerente ao assunto de interesse, assim como maximizar o valor do dado (CHEN et al., 2014). Análise de dados desempenha um grande papel de orientação no desenvolvimento de planos para um país, entender demandas de comércio através dos hábitos do consumidor e na previsão de tendências de mercado para companhias. Big data análise pode ser considerada como uma técnica de análise para um tipo especial de dado. Portanto, muitos métodos tradicionais de análise ainda podem ser utilizados para realizar big data análise. A seguir, são apresentados os principais métodos tradicionais utilizados para realizar big data análise, muitos são de áreas estatísticas e outros da ciência da computação, destacados em (CHEN et al., 2014).

- **Cluster Análise**: é um método conveniente para identificar grupos homogêneos de objetos chamados *clusters*. Objetos em um *cluster* específico compartilham muitas características, mas são muito diferentes de objetos que não pertencem aquele *cluster*. Para permitir o entendimento de como a *cluster* análise é útil, tome-se, por exemplo, que seja necessário a uma empresa organizar a sua base de clientes buscando alcançar grupos alvo, seguindo uma estratégia de preços diferenciada. A *cluster* análise permite identificar o grupo alvo de clientes por meio de agrupamento de clientes que comumente adquirem produtos em determinadas faixas de preço, permitindo a empresa maximizar o seu lucro oferecendo o produto adequado a cada grupo (*cluster*) alvo de clientes. *Cluster* análise é um método de estudo não-supervisionado sem dados de treino.

- **Análise de Fator:** é uma classe de processos utilizados na redução e sumarização de dados. As principais aplicações dessa técnica são: identificar fatores que justifiquem as correlações observadas entre variáveis e/ou substituir o conjunto original

de variáveis, em geral grandes, e correlacionadas por um conjunto menor de variáveis sem correlação ou baixa correlação e como consequência reduzir a complexidade do problema abordado.

- **Análise de correlação**: é uma técnica muito popular no campo da estatística. No caso da correlação simples (entre duas variáveis) é buscado verificar o grau de relacionamento linear entre essas variáveis aleatórias. A correlação é considerada como uma medida de associação mútua ou conjunta entre duas variáveis, ou seja, encontrar a magnitude do relacionamento entre as duas variáveis aleatórias. Além da correlação simples, existem a correlação múltipla, correlação parcial, correlação canônica, entre outras, sendo que a distinção entre elas consiste da quantidade de variáveis aleatórias e da metodologia aplicada para encontrar a magnitude do relacionamento. Alguns exemplos onde a análise de correlação pode ser aplicada seriam a relação entre a prática de esportes e ritmo cardíaco; resultado da produção e tempo de processo.

- **Análise de Regressão**: técnica estatística usada para modelar e investigar a relação linear quantitativa entre uma variável e uma ou mais variáveis, explicitando a forma dessa relação. Ela é baseada em grupos de experimentos ou dados observados, análise de regressão que identifica relacionamentos de dependência entre variáveis ocultas pela aleatoriedade, sendo esse modelo de análise frequentemente utilizado para previsões. Alguns exemplos onde a análise de regressão pode ser aplicada seriam: número de clientes e venda, tempo de estudo e nota na prova.

- ***Bucket Test***: também conhecido como teste A/B. É uma tecnologia para determinar como melhorar variáveis alvo pela comparação do grupo testado. Para entender melhor, suponhamos que um portal *web* queira testar a aceitação de um novo *layout* para a sua página inicial e, para tanto use o *bucket test*, ele irá separar um grupo de usuários em dois subgrupos que irão avaliar as duas páginas (nova e antiga), destacando as qualidades. O melhor será aquele que obtiver maior qualificação. Um ponto positivo dessa técnica é sua velocidade de avalição, mas, no caso de big data, ela irá requerer um imenso número de testes a ser executado e analisado.

- **Análise Estatística**: é baseada na teoria estatística, um ramo da matemática aplicada. Análise estatística provém uma descrição e uma inferência em big data. Estatística descritiva pode sumarizar e descrever conjunto de dados, enquanto análise estatística inferencial permite desenhar conclusões do assunto para variações aleatórias.

Análise estatística é amplamente aplicada nos campos econômicos (CHINNICI, D'AMICO e PECORINO, 2002) e médico (BLAND, 2000).

- **Algoritmos de Mineração de Dados**: Mineração de Dados é um processo para extração de informação escondida, desconhecida, mas potencialmente útil e conhecimento de dados maciços, incompletos, ruidosos, *fuzzy* e aleatórios. Em 2006, A IEEE Conferência Internacional sobre Mineração de Dados em Serie (ICDM) identificou os dez mais influentes algoritmos de mineração de dados através de procedimento de seleção rigoroso (PHILIP et al., 2008), incluindo C4.5, k-means, SVM, Apriori, EM, Naive Bayes e Cart, etc. Esses dez algoritmos compreendem classificação, clusterização, regressão, aprendizado estatístico, análise associativa e mineração de links, todos que são mais importantes problemas na pesquisa de mineração de dados.

Anteriormente foram apresentadas as metodologias tradicionais utilizadas para realizar big data análise. Contudo, dado o aumento do volume, da variedade de dados e das fontes de informações que surgiram graças ao crescimento da *web* na última década, há uma gigantesca massa de informação disponível não só na *internet* mas também em diversos bancos de dados de organizações e empresas. Sendo assim, como obter rapidamente e com precisão, informação chave dessa massa de dados? A solução encontrada foi o desenvolvimento de modelos de análise mais adequado a esse volume e variedades de dados e de fontes de informações. A seguir, são apresentados os principais métodos de processamento de big data utilizados atualmente (CHEN et al., 2014).

- **Filtro Bloom**: é uma estrutura de dados que tem por objetivo compactar a representação de um conjunto de dados, ou seja, ele utiliza um vetor de *bits* para representar um conjunto de dados. Para que isso aconteça ele faz de uma série de funções *hash*[3]. O princípio do filtro Bloom é armazenar valores *hash* dos dados ao invés do dado em si utilizando a *array* de *bit*, que essencialmente, é um índice *bitmap*, que usa funções *hash* para conduzir armazenamento comprimido com perda de dados. Tem como vantagens eficiência no uso do espaço e alta velocidade de consulta dos dados e como desvantagens reconhecimentos falhos e eliminação.

- ***Hashing***: é um método que essencialmente transforma dados em valores numéricos de tamanho fixo curto ou valores de índice. *Hashing* tem como vantagem

[3] Função que transforma os elementos de um universo arbitrário em inteiros relativamente pequenos que indexem uma tabela (chamada tabela *hash*), permitindo que os registros na tabela *hash* sejam endereçados diretamente a partir de uma chave de pesquisa.

rápidas leituras, escritas e velocidades de consultas, mas é difícil encontrar uma função *hash* de confiança, que não gere perda de dados na transformação e não possa ser quebrada.

- **Índice**: é um método efetivo para reduzir o custo de leitura e escrita, e melhorar a inserção, deleção, modificação e velocidade de consulta em bancos de dados tradicionais que lidam com dados estruturados ou outras tecnologias que lidam com dados semiestruturados ou desestruturados. Apesar das vantagens, índice possuí uma desvantagem que implica em custo adicional para armazenamento do índice de dados que deve ser atualizado dinamicamente junto com os dados.

- **Computação Paralela**: comparada a computação serial tradicional, computação paralela consiste no uso simultâneo de recursos computacionais para completar uma tarefa de computação. A ideia básica é decompor o problema inicial e atribui-lo a diversos processos separados para ser independentemente completado, de modo a alcançar coprocessamento. Alguns dos tradicionais métodos de processamento paralelo incluem MPI (*Message Passing Interface*), MapReduce (DEAN e GHEMAWAT, 2004) e Dryad (ISARD et al., 2007).

A escolha de uma das técnicas anteriormente apresentadas é determinante para o sucesso do resultado esperado de uma análise big data, pois deve não só atender todos os requisitos do trabalho proposto, mas também levar em consideração os recursos disponíveis (*hardware*, tempo, dinheiro, etc.), visto que algumas delas demandam recursos abundantes, enquanto outras permitem a flexibilização de recursos disponíveis. Neste trabalho, busca-se otimizar uma técnica de análise de permita a utilização de *hardware* de baixo e/ou médio poder de processamento, como será demonstrado no capítulo 3.

2.1.3 Aplicações de Big Data

Na seção anterior foi abordada a análise big data, que é parte final e mais importante fase da cadeia de valor de big data (CHEN et al., 2014). Análise big data permite prover valores úteis por meio de sugestões, decisões, julgamentos ou auxílio.

Em (CHEN et al., 2014), é proposta uma divisão da análise de dados em seis importantes campos de análise – tanto de um ponto de vista de mercado, quanto de pesquisa – incluindo análise de dados estruturada, análise de texto, *web data* análise, análise multimídia, análise de dados de rede e análise de dados móveis.

- **Análise de Dados Estruturados**: é fomentada principalmente graças a massiva quantidade de informações estruturadas geradas por aplicações comerciais e pesquisas cientificas. Nesse âmbito, a análise de dados é baseada em mineração de dados e análise estatística. Por exemplo, aprendizado de máquina estatístico baseado em modelos matemáticos exatos e poderosos algoritmos tem sido aplicado em detecção de anomalias (BAAH, GRAY e HARROLD, 2006) e controle de energia (MOENG e MELHEM, 2010).

- **Análise de Texto**: como a maioria da informação armazenada se encontra em formato textual (por exemplo: e-mails, páginas *web*, mídias sociais), é simples entender a importância dos sistemas voltados para busca de informação relevante nessa massiva quantidade de informação. De maneira geral, a análise de texto consiste em um processo de extração de informação útil a partir de texto estruturado. A grande maioria dos sistemas para mineração em texto são baseados em NLP (*Natural Language Processing*), pois NLP permite ao sistema analisar, interpretar e até mesmo gerar texto estruturado (CHEN et al., 2014).

- ***Web Data* Análise**: tem crescido significativamente como um ativo campo de pesquisa. Esse tipo de análise busca realizar de maneira automática as etapas de recuperação, extração e avaliação de informação advindas de documentos e serviços *web* assim como revelar informação de valor nesses dados processados (CHEN et al., 2014). A *web data* análise é uma área relacionada com diversas outras, incluindo banco de dados, recuperação de informação, NLP e mineração de texto. *Web data* análise pode ser classificada em três campos distintos (PAL, TALWAR e MITRA, 2002): Mineração de conteúdo *web*[4]; mineração de estrutura *web*[5]e a mineração de uso na *web* (análise que é utilizada para auxiliar a geração de conteúdo através de atividades ou diálogos na *web*, exemplos dela incluem logs de acesso em *web* e proxy servers). *Web data* análise e suas ramificações são responsáveis pelo desenvolvimento e crescimento da *web* e consigo melhorando a maneira com que empresas e usuários lidam com produtos e serviços ligadas a *web*.

[4] Consiste do processo de descoberta de informação útil em páginas da web, que em geral compreende conteúdo multimídia.
[5] Envolve a criação de modelos para descoberta de estruturas de links web, um exemplo é *Page Rank* (índice de páginas da web).

- **Análise multimídia**: o contexto da *web* 2.0 e o fenômeno big data estão intimamente ligadas, pois os dados multimídias (na sua maioria imagens, vídeos e áudio) têm crescido em velocidade surpreendente, principalmente por conta das redes sociais, o que torna a extração de informação útil e detecção da correlação de dados algo efervescente. Muitas pesquisas ligadas a esse tipo de análise e aplicações voltadas para essa tarefa tem surgido e fomentado uma necessidade da indústria. Pesquisas voltadas em análise multimídia compreendem muitas áreas: sumarização de multimídia, anotação de multimídia, indexação e recuperação de multimídia, sugestão de multimídia, detecção de eventos multimídia, entre outros (PAL, TALWAR e MITRA, 2002).

- **Análise de Dados de Rede**: essa modalidade de análise evolui de dois outros campos, análise quantitativa e análise sociológica de rede, para a análise de rede social no começo do século 21, por conta do interesse crescente sobre os vários serviços de redes sociais, incluindo Facebook, Twitter, LinkedIn, Google+, entre outras (CHEN et al., 2014). Essas redes sociais comumente incluem massivas conexões de dados e conteúdo. Essa conexão de dados consiste principalmente de estruturas gráficas, descrevendo as comunicações entre duas entidades e o conteúdo é constituído principalmente por textos, imagens e outros dados no formato multimídia.

Em concordância com a perspectiva centralizada nos dados, as pesquisas existentes na área de análise de dados de rede sociais podem ser classificadas em duas categorias: análise estrutural, focada em conexões e análise baseada em conteúdo (CHEN et al., 2014). Análise estrutural, baseada em conexões tem focado em predição de conexões, descoberta de comunidade, evolução da rede social, análise de influência social, etc. Essas análises, em geral, utilizam métodos probabilísticos, caracterização baseada em características e álgebra linear. Análise baseada em conteúdo em redes sociais também conhecida como análise de mídia social, inclui texto, multimídia, posicionamento (opinião) e comentários. Sendo que essa segunda é considerada ainda uma área muito recente e em expansão (CHEN et al., 2014).

- **Análise de Dados Móveis**: é outra tendência em escalada por conta do crescimento acelerado do número de aplicações para plataformas móveis (IOS, Android, Windows Phone), de forma que de essas aplicações cobrem praticamente todas as categorias. No fim de 2012, a quantidade de dados móveis gerada mensalmente tem alcançado 885 Petabytes (Cisco Visual Networking Index, 2013). Com o crescimento

dos usuários móveis e melhora na performance dos dispositivos, esses dispositivos se tornaram úteis para criação e manutenção de comunidades, como comunidades baseadas em compartilhamento de localização geográfica (como o Foursquare[6]). Esse cenário só ressalta a importância de técnicas e ferramentas capazes de realizar análise de dados móveis. Uma das pesquisas baseadas em análise de informação móveis é a pesquisa desenvolvida na Gjovik University College em Norway e Derawi Biometrics colaboraram para o desenvolvimento de aplicações para smartphones, que analisa os passos de uma pessoa quando esta caminha e usas essas informações para destravar o sistema de segurança (MAYER-SCHÖNBERGER e CUKIER, 2013).

2.2 Plataformas e Frameworks Open Source para Análise Big Data

Uma plataforma *open source* compreende um programa ou ferramenta que realiza uma tarefa especifica e cujo código fonte é aberto e divulgado para adições e/ou modificações sobre seu *design* original, sem que haja alguma forma de ônus ao programador (CHEN et al., 2014). O crescente interesse em big data fez com surgissem várias ferramentas voltadas para a etapa de análise big data. Algumas características comuns a todas ferramentas e plataformas apresentadas nesta seção deste trabalho são: suporte a variedade de dados (em alguns casos de múltiplas fontes em paralelo), processamento em lote *offline* e/ou *streaming* de dados em tempo real, algoritmos otimizados a cada tipo de análise a ser realizada. A seguir, abordaremos as seguintes plataformas e *frameworks*: MapReduce, Hadoop, Storm, Hortonworks, Apache Drill e o *framework* GridGain (que será adotado no estudo de caso apresentado nesse trabalho).

2.2.1 MapReduce

MapReduce (DEAN e GHEMAWAT, 2004) é modelo de programação criado pelo Google em 2004. Foi desenvolvido para simplificar o processamento de dados em paralelo sobre uma estrutura de grandes *clusters* distribuídos. O modelo de programação é baseado nas primitivas *map* e *reduce*, encontrada em linguagens funcionais como *Lisp*, entre outras.

MapReduce é muito eficiente e altamente escalável, tolerante a falhas, permitindo uma distribuição de dados e balanceamento de carga entre os *clusters*. Um *Framework* MapReduce

[6] Rede Social que permite aos seus usuários o compartilhamento da sua localização com os outros membros da rede por meio do recurso de *check-in*.

consiste na utilização das primitivas *map* (função que processa um par chave/valor para gerar um conjunto de pares chaves/valor intermediário) e *reduce* (função que agrupa todos os valores associados com a mesma chave intermediária) que automaticamente paraleliza e executa sobre a estrutura de *clusters* o programa escrito.

A ideia do MapReduce é ocultar a dificuldade da paralelização de dados, tolerância a falhas, distribuição e balanceamento de carga de dados por meio de uma biblioteca simples. Além do problema computacional, o programador precisa apenas definir parâmetros para controle da distribuição de dados e paralelismo (LAMMEL, 2008). MapReduce foi projetado pelo Google para operar sobre uma grande estrutura de *clusters* conectados em rede. Outras implementações foram introduzidas baseadas no *design* original, por exemplo o Hadoop (Apache Hadoop, 2008) – abordado no tópico 2.2.2 – é uma implementação *open source* do MapReduce, escrita em Java. Assim como MapReduce do Google, o Hadoop utiliza um grande número de máquinas em um *cluster* para processamento distribuído de dados.

A execução de uma tarefa MapReduce é apresentada de forma global pela Figura 2.2. Quando rodando o programa do usuário, a biblioteca MapReduce primeiro divide o dado de entrada em M pedaços. Cada um tem tipicamente entre 16-64MB por parte. Em seguida a biblioteca executa as várias cópias do programa sobre o conjunto de máquinas do *cluster*. Uma das cópias é eleita como nó mestre, que designa as tarefas *map* e *reduce* para os nós trabalhadores. São M *map* tarefas para serem executadas, uma para cada dado de entrada divido. Quando um nó trabalhador é designado para uma tarefa *map*, este lê a entrada de dado correspondente e passa o par chave/valor para a função *map*, definida pelo usuário. O conjunto de pares intermediários chave/valor são armazenados na memória e periodicamente escritos para o disco local, particionado em R partes. Uma função de particionamento definida pelo usuário (por exemplo, *hash* (chave) mod R) é usado para gerar R partições. Localização dos conjuntos de pares chaves/valor intermediários são passados de volta para o nó mestre, que encaminha a informação para os nós *reduce* quando necessário.

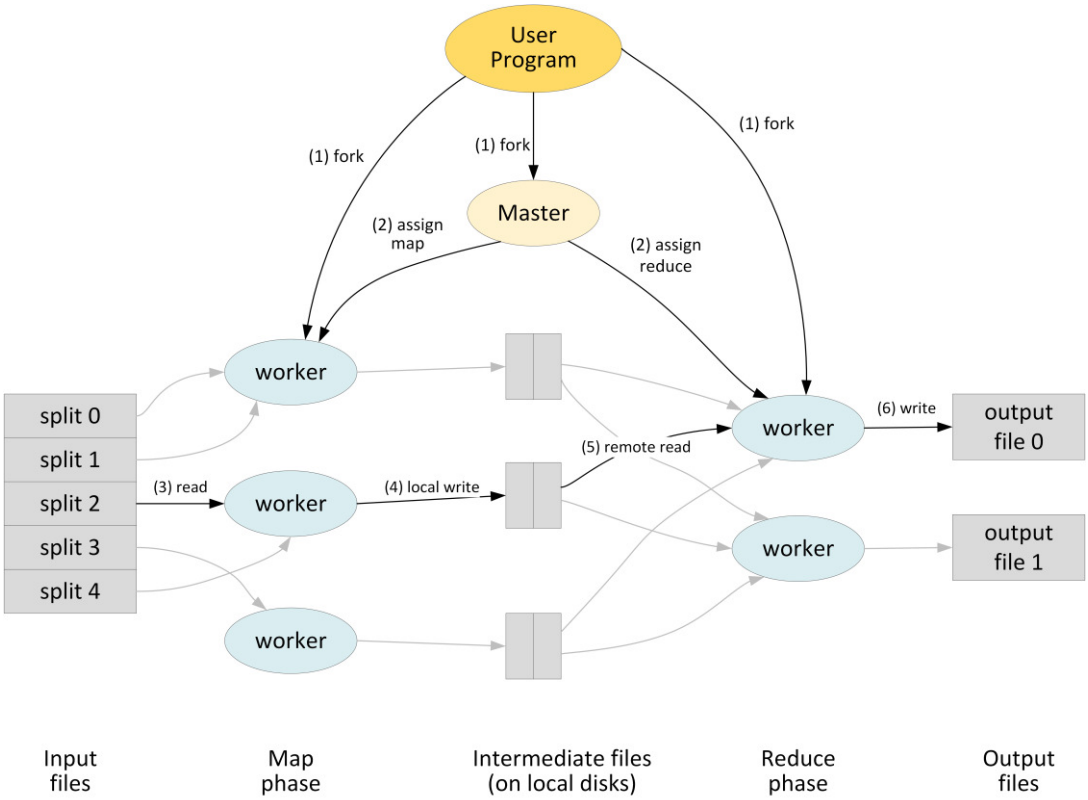


Figura 2.2: Visão global de uma execução MapReduce. Fonte: (DEAN e GHEMAWAT, 2004).

O exemplo mais popular do uso do MapReduce é o problema da contagem do número de palavras distintas em grande conjunto de documentos. Mas muitas outras aplicações foram sendo implementadas internamente pelo Google, como discutido em (DEAN e GHEMAWAT, 2004), tais como clusterização de problemas para o Google *News*, problemas de aprendizado de máquina, extração de dados de uso para produção de relatórios de consultas populares (por exemplo, Google *Zeitgeist*), computações gráficas de alta escala, entre outras.

2.2.2 Hadoop

Hadoop (Apache Hadoop, 2008) é uma implementação *open source* do MapReduce desenvolvida pela Apache. A arquitetura do Hadoop é essencialmente a mesma da implementação do Google, e a principal distinção é que Hadoop é distribuído sobre licença Apache, assim como muitos outros produtos *open source*.

Dados encontram-se distribuídos sobre o conjunto de máquinas em rede usando o Hadoop *Distributed File System* (HDFS). HDFS é um sistema de arquivos distribuídos que redistribui os dados em computadores próximos ao *cluster* e cria réplicas dos blocos de dados

para aumentar a confiabilidade. Discos locais de máquinas conectadas na rede são usados para persistir dados, que permite a acessibilidade dos dados para outras máquinas na rede.

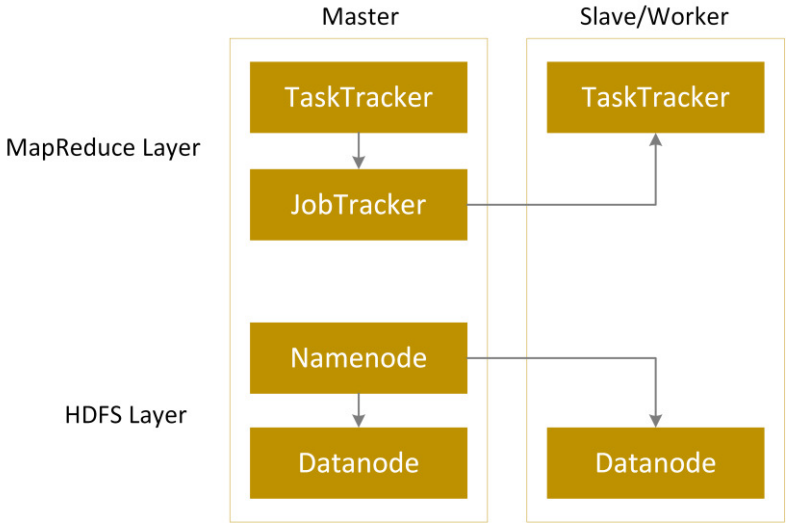


Figura 2.3: Visão global das camadas da arquitetura interna do Apache Hadoop. Fonte: (Apache Hadoop, 2008).

HDFS consiste de dois processos principais, o *Namenode* e número de *Datanodes*, como é possível verificar na Figura 2.3 (Apache Hadoop, 2008). O eletivo *Namenode* secundário pode também ser usado como um processo backup para o para o *Namenode*. O *Namenode* executa em uma única máquina mestre. Esta Contém informação sobre todas as máquinas no *cluster* e os detalhes sobre os blocos de dados persistidos nas máquinas componentes do *cluster*. O *Datanode* processa a execução em todos as outras máquinas do *cluster*, elas se comunicam com o *Namenode* para saber quando buscar dados em seu disco rígido local. O *framework* MapReduce do Hadoop consiste de um único *JobTracker* e um número de processos *TaskTracker* (Apache Hadoop, 2008). O *JobTracker* normalmente executa na mesma máquina que o *Namenode*. Usuários designam seus *jobs* MapReduce para o *JobTracker*, que divide a tarefa entre as máquinas do *cluster*. Cada máquina no *cluster* executa um processo *TaskTracker*, este se comunica com o *JobTracker*, que designa uma tarefa *map* ou *reduce* quando possível. Hadoop pode ser organizado para executar múltiplas simultâneas tarefas *map* em nós simples. Em sistemas de núcleos múltiplos essa é um grande benefício, já que permite o uso total dos núcleos.

As principais aplicabilidades de Hadoop estão nas aplicações que necessitam realizar o processamento *offline* de imensos lotes de históricos e/ou *payloads* analíticos, nos quais latência e transações não são realmente importantes. Exemplos desse tipo de processamento são

otimização de conteúdo, índice de busca, processamento de conteúdo de *feeds*, otimização de propagandas e filtros spam. Muitas empresas como Yahoo, Facebook, Twitter, Linkedin, The New York Times, American Air Lines (WIKI Apache 2014) utilizam *clusters* Hadoop em suas principais tarefas de processamento e análise de dados.

2.2.3 Storm

Storm é um sistema de computação em tempo real *open source*, distribuído, tolerante a falhas, desenvolvido sob licença pública pela BackType. Foi adquirido pelo Twitter para realizar processamento de grandes volumes de *streaming* de informações produzidas no microblog. Storm não trabalha com dados estatísticos, mas sim com dados contínuos (Storm, 2011). Com os usuários do Twitter gerando 140 milhões de *tweets* por dia, é fácil entender como essa tecnologia é útil (MARZ, 2011).

Storm é um exemplo de sistema *Complex Event-Processing* (CEP). Sistemas CEP são geralmente categorizados como sendo orientados a cálculos e detecção, cada um dos quais podem ser implementados em Storm usando algoritmos definidos pelo usuário. CEPs podem, por exemplo, ser usados para identificar eventos significativos em uma avalanche de eventos e realizar ações relacionadas a eles em tempo real (Storm, 2011).

O *cluster* Storm é composto de um nó mestre e nós trabalhadores (MARZ, 2011). Os nós mestres executam um programa chamado “Nimbus” que é responsável pela distribuição de código, designando tarefas e buscando por falhas. Cada nó trabalhador executa um programa chamado “Supervisor” que espera pelas chamadas de tarefas e iniciam e param processos trabalhadores. Os programas Nimbus e Supervisor são *fail-fast* e *stateless*, que permite robustez e coordenação entre eles e gerenciados pelo Apache ZooKeeper (APACHE ZOOKEEPER, 2008).

O que diferencia Storm de outras soluções de big data é o seu paradigma. Hadoop é basicamente um sistema de processamento em lote. Os dados são introduzidos no Hadoop *Distributed File System* (HDFS) e distribuídos entre nós para processamento. Quando o processamento é concluído, os dados resultantes são devolvidos ao HDFS para uso da aplicação. Storm permite a criação de topologias que transformam fluxos não terminados de dados. Essas transformações, ao contrário das tarefas de Hadoop, nunca param, continuando a processar os dados à medida que eles chegam.

Storm implementa um conjunto de características que o define em termos de desempenho e confiabilidade. Storm usa ZeroMQ[7] para passagem de mensagens, o que remove o enfileiramento intermediário e permite que as mensagens passem diretamente entre as próprias tarefas.

Storm utiliza um modelo de fluxo de dado no qual os dados fluem continuamente através de uma rede de entidades de transformação, demostrado na Figura 2.4. A tupla é como estrutura que pode representar tipos de padrão (como números inteiros, flutuações e *array* de *bytes*) ou tipos definidos pelo usuário com algum código de serialização adicional. Cada fluxo é definido por um ID exclusivo que pode ser usado para criar topologias de origens e dissipadores de dados. Os fluxos originam-se de *spouts*, que passam de origens externas para topologia de Storm.

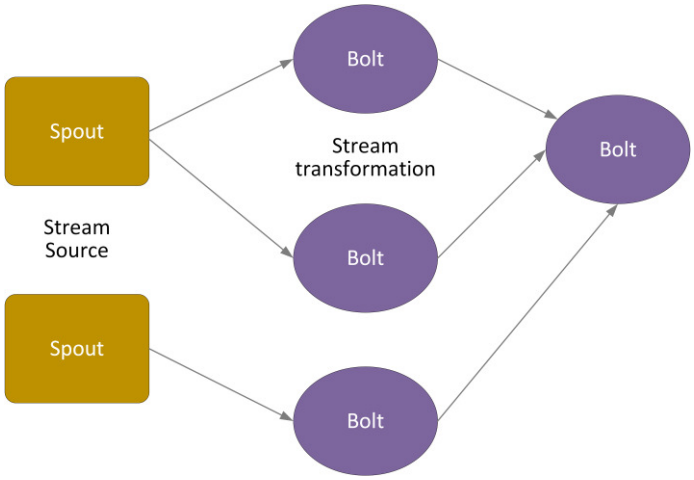


Figura 2.4: Arquitetura conceitual do modelo do fluxo de dados contínuo Storm. Fonte: (Storm, 2011).

Storm suporta virtualmente qualquer linguagem de programação. Por padrão, ele suporta as linguagens Clojure, Java, Ruby e Python. Porém, outras linguagens são suportadas por meio de *plugins* (Storm, 2011).

Como descrito em (Storm, 2011), além das aplicações de análise desenvolvidas pelo Twitter para processamento e busca de *insights* em *tweets*, outros tipos de análise são possíveis mediante a utilização da *Application Programming Interface*[8] (API) disponibilizada para

[7] ZeroMQ é um *middleware* orientado a mensagem, que permite as aplicações (ou módulos da aplicação) comunicarem entre si por meio de passagem de mensagens instantâneas e tem seus recursos disponibilizados por meio de uma biblioteca de desenvolvimento.
[8] Consiste de um *software* programa ou biblioteca que facilita a interação com outros *softwares*, por meio de uma interface que defini as regras de comunicação com o *software* ou componente de um *software*.

desenvolvedores, tendo como foco aplicações voltadas para processamento paralelo em tempo real para análise de dados em fluxo continuo, a saber: aprendizado de máquina *online*, RPC (*Remote Procedure Call*) distribuído, *update* de banco de dados em tempo real, computação (*queries* continuas em *streaming* de dados e geração de *streaming* de resultados) .

2.2.4 Hortonworks Data Plataforma

A Hortonworks Data Plataforma (HDP) é uma plataforma para gerenciamento de dados *open source* para Apache Hadoop voltada para empresas. Fundada em 2011 por 24 engenheiros das esquipes de desenvolvimento e operações originais Hadoop para o Yahoo!, a Hortonworks realiza todo seu desenvolvimento de acordo com os processos da Apache Software Foundation[9], o que significa que o código da plataforma é aberto, sem nenhuma extensão proprietária (Hortonworks, 2011).

Os componentes principais da HDP são o YARN (*Yet Another Resource Negotiator*) codinome para plataforma Hadoop 2.0 e o HDFS (sistema de gerenciamento de arquivos distribuído da plataforma Hadoop). Essa evolução da arquitetura do Hadoop é apresentada na Figura 2.5. A seguir, é abordado o sistema operacional para gerenciamento de aplicações YARN, que é o principal componente da nova arquitetura Hadoop 2.0. O sistema de gerenciamento de arquivos distribuídos HDFS foi discutido no tópico 2.1.4.2, portanto, não será abordado novamente.

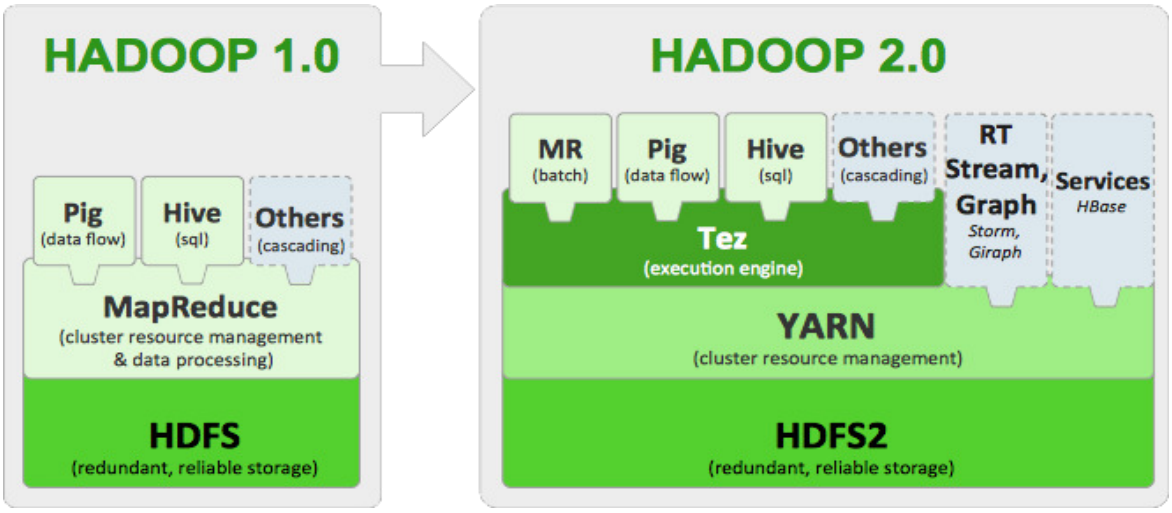


Figura 2.5: Evolução da arquitetura do Hadoop introduzida pela Hortonworks. Fonte: (Hortonworks, 2011).

[9] http://www.apache.org/foundation/

Como pode ser observado na Figura 2.5, o YARN é o *framework* para gerenciamento de recursos que permite o processamento de dados simultâneos de várias maneiras. Além de permitir outras tarefas que não MapReduce sobre um *cluster* Hadoop, YARN também introduz um novo gerenciador de recursos que atua como um escalonador de recursos, responsável por atribuir de maneira arbitraria os recursos disponíveis no *cluster*.

A ideia fundamental do YARN é a dividir as duas maiores responsabilidades do *JobTracker* (gerenciamento de recursos e escalonamento/monitoração de *job*) em dois processos separados: um *ResourceManager* global e um *ApplicationMaster* (AM) por aplicação. O *ResourceManager* e o *NodeManager* (NM) formam o novo e genérico sistema operacional para gerenciamento de aplicações em uma maneira distribuída.

Como é possível verificar na Figura 2.6, o *ResourceManager* é a autoridade que arbitraria a alocação de recursos entre todas as aplicações no sistema. O *ApplicationMaster*, por aplicação, é uma entidade *framework* especifica e fica encarregada pela negociação de recursos do *ResourceManager* e cooperação com o *NodeManager*(*s*) para executar e monitorar as tarefas do componente. O *ResourceManager* possui um escalonador de componentes plugável, que é responsável pela alocação de recursos para as várias aplicações executando, sujeitas, a restrições comuns de capacidade, consultas, etc. O escalonador é um escalonador puro no sentido que não realiza monitoramento ou rastreamento do *status* da aplicação, oferecendo nenhuma garantia de reinicio em caso de falha, seja devido à aplicação ou falha de *hardware*.

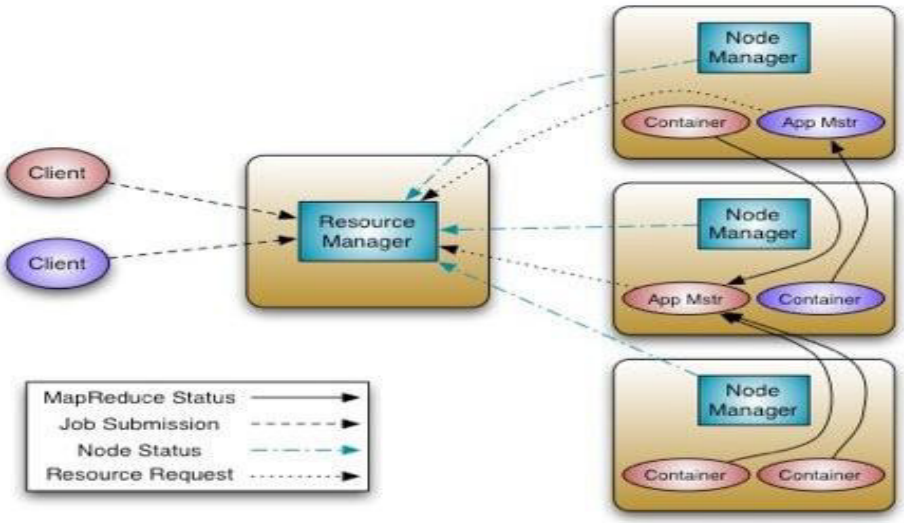


Figura 2.6: Novos elementos de controle YARN. Fonte: (Hortonworks, 2011).

O *NodeManager* é o escravo por máquina, que é responsável por iniciar o container de aplicação do escalonador, monitorar sua utilização de recursos (CPU, memória, disco, rede) e reportá-los para o *ResourceManager*.

Essa arquitetura plugável disponibilizada por meio do *framework* YARN é responsável por permitir a aplicação de *hardware* de baixo custo na solução de virtualmente qualquer problema big data em lote, interativo, *online*, *streaming*, gráfico, em memória, etc. Desde que a aplicação esteja integrada com o YARN.

2.2.5 Apache Drill

Apache Drill é um sistema distribuído que suporta aplicações intensiva de dados para análise ad-hoc interativo de grandes conjuntos de dados. Apache Drill é desenvolvido para lidar com Petabytes de dados distribuídos por milhares de servidores, o objetivo de Drill é responder a consultas ad-hoc com baixa latência (HAUSENBLAS e NADEAU, 2012).

Apache Drill é versão *open source* do Dremel (SERGEY et al., 2010) proposto pelo Google que está disponível como um *Interface as a Service* (IaaS) chamado Google BigQuery, o qual introduziu duas principais inovações: manipulação genérica de dados aninhados com representações *column-striped* (incluindo registro de montagem) e árvore de consulta de execução em vários níveis, permitindo processamento paralelo de dados espalhados sobre milhares de nós (HAUSENBLAS e NADEAU, 2012).

Em alto nível, a arquitetura Apache Drill (Figura 2.7) compreende as seguintes camadas:

- **Usuário**: responsável por prover a interface, como a interface por linha de comando (CLI), a interface REST, JDBC/ODBC, etc. para interação humana ou dirigida por aplicação.
- **Processamento**: responsável por permitir consultas por várias linguagens de forma plugável, assim como o preparo da consulta, sua execução e motores de armazenamento.
- ***Data Sources***: fontes de dados plugáveis localmente ou em uma instalação *cluster*, provendo processamento de dado *in-situ*.

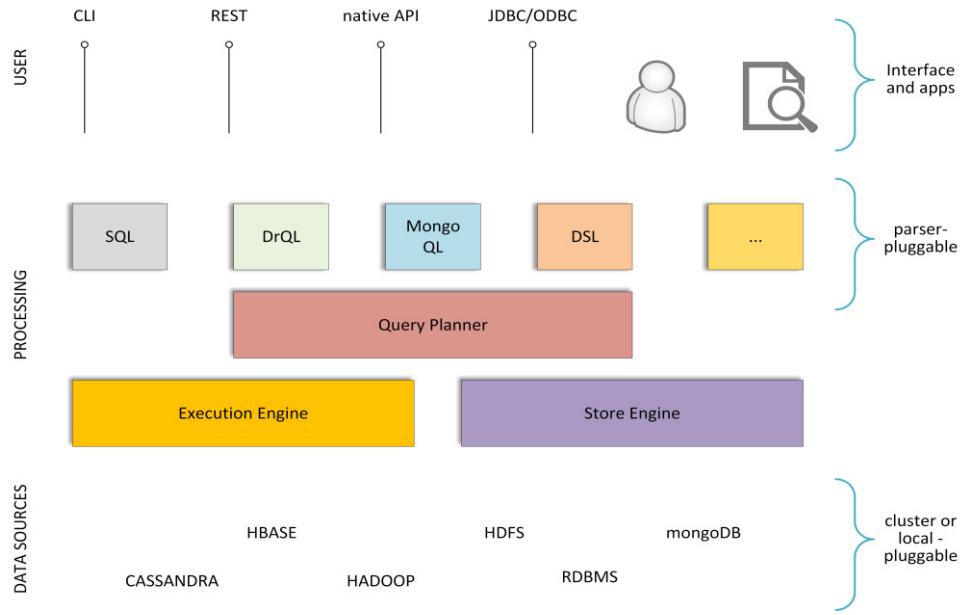


Figura 2.7: Arquitetura de alto nível Apache Drill. Fonte: (HAUSENBLAS e NADEAU, 2012).

Note que Apache Drill não é um banco de dados, mas sim, uma camada de consulta que trabalha com vários bancos de dados subjacentes. É primariamente desenhado para realizar um escaneamento completo da tabela em busca de dados relevantes. Diferente do MapReduce, que compõem o Hadoop, que provê um *framework* para processamento paralelo, Apache Drill provê um *framework* flexível para execução de consulta, permitindo a utilização de vários casos de uso, por meio de uma rápida agregação de estatísticas para exploração da análise de dados.

Apache Drill inclui um ambiente de execução distribuído, proposto para processamento de dados em larga escala. No núcleo do Apache se encontra o serviço *Drillbit*, que é responsável por aceitar requisições do cliente, processando as consultas e retornando os resultados para o cliente. O serviço *Drillbit* executa em cada nó processador buscando maximizar a localização dos dados, evitando a movimentação redundante de dados entres os nós do *cluster*.

Junto com o ambiente de execução de consultas, que serve como base para as consultas e processamento de dados em escala, existem muitos elementos do núcleo arquitetural do Apache Drill que o tornam um motor altamente flexível e eficiente de consultas em comparação com as tecnologias SQL-Hadoop existentes (HAUSENBLAS e NADEAU, 2012). Dentre esses elementos, os mais destacados pela comunidade desenvolvedora seguem:

- **Regras de extensibilidade**: Apache Drill é desenhado para extensibilidade, por meio de uma API bem definida, que permite a utilização várias linguagens para realizar consultas na base de dados.

- **Dados Aninhados**: Como os tipos de dados semiestruturados (como JSON/BSON em documentos armazenados, XML, *clickstream*, logs e dados de sensores típicos de IoT) estão se tornando cada vez mais comuns, Apache Drill lida com dados complexos e semiestruturados, como um conceito nativo.

- **Tipificação Dinâmica ou Ligação Tardia de *Schemas***: Apache Drill não requer *schema* ou especificação do tipo do dado para começar o processo de execução da consulta. Apache Drill processa o dado em unidades chamadas lotes de registro e descobre os *schemas* durante o processamento.

Essa arquitetura plugável por meio de uma API flexível torna o Apache Drill uma solução *open source* ideal para quem busca realizar análise de dados com baixa latência. Ele representa uma alternativa cada vez mais proeminente por conta da comunidade que, por meio da incubadora Apache que hospeda o projeto, tem evoluído o projeto, o qual conta inclusive com uma versão *single-node* disponível[10]. Dentre os trabalhos de melhoramento do projeto, inclui-se um interpretador SQL para motores de armazenagem comuns no mercado tais como: Cassandra[11], HBase[12], Hadoop, etc. (HAUSENBLAS e NADEAU, 2012).

2.2.6 GridGain

GridGain é um *middleware open source* baseada em *Java Virtual Machine* (JVM) que foi desenvolvida para permitir aplicações escaláveis com capacidades de tempo real por meio da integração de *In-Memory Data Grid* (IMDG) com *In-Memory Compute Grid* (IMCG), que se encontra atualmente na versão 6.2.0, lançada em 24 de agosto de 2014. Essa integração é uma das ideias principais em sistemas distribuídos de alta performance para lidar com imensos conjuntos de dados (GridGain, 2011). Por essas características GridGain é uma alternativa ideal para realizar processamento de alta performance mesmo em *hardware* de baixo custo, por meio da API que permite migrar o processamento para os grids em memória que, oportunamente, podem estar armazenando junto aos dados a serem processados.

In-Memory Compute Grid e suas implementações são menos frequentemente utilizadas que o paradigma de armazenamento tradicional. Isso acontece historicamente pela seguinte razão: a indústria concentra seus esforços nas tecnologias de armazenamento, tipicamente das

[10] https://cwiki.apache.org/confluence/display/DRILL/Apache+Drill+Wiki
[11] http://cassandra.apache.org/
[12] http://hbase.apache.org/

variedades NoSQL[13], NewSQL[14] e IMDG. Uma vez que a tecnologia de armazenamento está implantada, a adição de um novo tipo de uma não-tradicional capacidade de processamento em memória sobre a estrutura inicialmente implantada se torna difícil, ou mesmo impossível. Isto por causa das características de IMCG, que são, geralmente, mais fundamentais para o produto como um todo, precisa ser implementado primeiro ou fortemente integrado para ser usado como núcleo da plataforma de armazenamento.

GridGain e Hadoop são duas distribuições no mercado que combinaram com sucesso ambas armazenamento e processamento em um único sistema, embora estes tenham alcançado isso de formas bem diferentes. Enquanto Hadoop HDFS realiza somente as ações de gerenciamento de dados sobre a estrutura de *clusters* distribuídos, GridGain IMDG além de gerenciar os dados, permite a adição de transações, novas partições de dados e consultas SQL em cada nó cujo dados são necessários para alguma operação e que estejam em *cache* em memória, evitando movimentação redundante dos dados tanto em *cache* quanto em disco, diminuindo tempo de processamento por conta da espera pela movimentação dos dados.

GridGain *In-Memory Compute Grid* possui sua própria implementação de MapReduce, que é desenhada especificamente para casos de uso de processamento em tempo real em memória e é essencialmente diferente da implementação do Apache Hadoop. O MapReduce do GridGain é composto de um nó mestre e múltiplos nós trabalhadores. GridGain provê mecanismos úteis para usuários adicionarem propriedades, que são visíveis para o nó mestre. Nós mestres podem identificar nós trabalhadores a partir das propriedades adicionadas, essas propriedades podem, por exemplo, especificar diferentes propósitos para cada nó de acordo com a necessidade da aplicação (nome lógico do nó, nome do papel do nó, etc.).

O objetivo principal do GridGain MapReduce é dividir uma tarefa em múltiplas subtarefas, realizar o balanceamento de carga entre essas subtarefas e dividí-las entre os nós *cluster* disponíveis, os quais são detectados automaticamente, executá-las em paralelo, e por fim agregar o resultado das subtarefas e retorná-las ao usuário por meio da interface de aplicação, como demonstrado na Figura 2.8.

[13] Sistema de armazenamento de dados que permite flexibilidade na manipulação de grandes volumes dados por meio de *schemas* de armazenamento como chave-valor, colunar, entre outros.
[14] Sistema de armazenamento de dados relacional moderno, que permite escalabilidade e performance semelhante aos sistemas NoSQL para transações *online*.

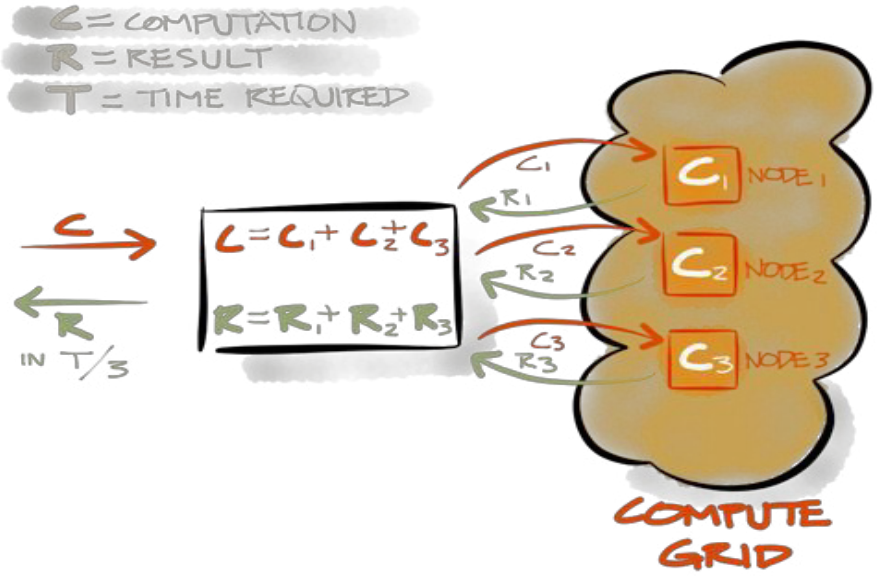


Figura 2.8: Modelo de execução no GridGain *In-Memory Compute Grid*. Fonte: (GridGain, 2011).

A divisão de tarefas em múltiplas subtarefas a atribuí-las aos nós *clusters* disponíveis é a tarefa *mapping* (etapa *map* do MapReduce) e agregação dos resultados é tarefa *reducing* (etapa *reduce* do MapReduce). Entretanto, não existe conceito de dado mandatório nessa implementação de MapReduce, isso permite que ele possa trabalhar mesmo com ausência de dados o que torna-o um bom ajuste para ambas, computação *stateless* e *state-full*, como os tradicionais sistemas de computação de alta performance. Nos casos em que o dado está presente, GridGain IMCG também aloca automaticamente a computação com o nó onde o dado requerido se encontra, evitando dessa forma a movimentação redundantes de dados sobre a estrutura de *clusters* distribuído, melhorando a performance.

Apesar de GridGain IMCG parecer um uma opção viável para muitos cenários, segundo (GridGain, 2011), o GridGain IMCG é um arranjo melhor alocado para processamento de computações que possuem um pequeno ciclo de vida, como computações que levem menos de 100 milissegundos e talvez não requeira nenhuma tarefa de *mapping* ou *reducing*. Isso otimiza o tempo de processamento e uso de recursos de memória.

GridGain *In-Memory Data Grid* (IMDG) é um esquema de armazenamento chave-valor em memória (*In-Memory Key-Value*). O GridGain IMDG é baseado no *caching* distribuído, contudo, este incorpora transações, particionamento de dados e consultas SQL para dados em cache. Em relação ao HDFS (utilizado no Apache Hadoop), a principal diferença é a habilidade de transacionar e atualizar qualquer dado em tempo real.

Em (GridGain, 2011) é apontado que GridGain IMDG é uma solução para trabalhar com conjuntos de dados operacionais, pois esse conjunto de dados comumente são atualizados e consultados ao mesmo tempo, enquanto o HDFS é mais adequado para trabalho em dados de históricos que nunca mudam.

Diferente de um arquivo de sistema, GridGain IMDG trabalha com um modelo de domínio de usuário, realizando diretamente *caching* dos dados da aplicação do usuário. Objetos são acessados e atualizados por uma chave que permite IMDG trabalhar com dados voláteis que requerem acesso baseado em chave.

GridGain IMDG permite indexação em chaves e valores e suporta SQL nativo para consultas e processamentos de dados. Em (GridGain, 2011), é apontado que GridGain IMDG provê suporte para *joins* distribuídos que permitem a execução de consultas SQL complexas no dado em memória sem limitações.

Desde a versão 3.0 GridGain incluiu a capacidade de *zero-deployment* em seu *data grid*. *Zero deployment* é a capacidade onde todas as classes e recursos necessários são carregados por carregamento por classe (*P2P class loading*), GridGain provê três diferentes modos de suporte *peer-to-peer deployment* suportando a maioria dos ambientes como arquivos WAR/EAR, etc. Dessa forma o desenvolvedor não tem que realizar o *deploy* manual do código Java ou Scala em cada nó do *grid* e *re-deploy* a cada vez uma alteração é realizada.

As características apresentadas anteriormente tornam GridGain um *middleware* independente de arquiteturas de armazenamento, que apresenta uma latência muita baixa em ambas operações transacionais e não-transacionais em tempo real. Além de uma API que permite a integração com arquitetura Hadoop existente.

Algumas das principais aplicabilidades descritas em (GridGain, 2011) focam o processamento de dados em paralelo e em larga escala (em tempo real, se necessário). Entre as histórias de sucesso apresentadas, estão casos de gerenciamento de risco, processamento de dados médicos, *games multiplayer online*, educação à distância, plataforma de e-commerce, tecnologias de genoma. Mas muitas outras aplicabilidades são possíveis, como: detecção de fraude em tempo real, análise de modelos de investimento, análise de sentimento (em lote e tempo real), processamento geoespacial/geolocalizacional, processamento de linguagem natural (NPL) e computação cognitiva.

A escolha deste *framework* para desenvolvimento do estudo de caso proposto é justificada por conta deste ser um *framework* baseado em Java, linguagem que permite

portabilidade entre sistemas operacionais, ampla documentação disponível em sítio *web*[15], código fonte aberto[16] e uma comunidade desenvolvedora sólida em torno de si. Todos esses aspectos contribuem para um desenvolvimento consistente e em tempo hábil do trabalho proposto.

2.3 Análise de Sentimento

Análise de sentimento, também chamada de mineração de opinião, é campo de estudo que analisa a opinião das pessoas, seus sentimentos, avaliações, atitudes e emoções, seja essa análise ligada a produtos, serviços, organizações, indivíduos, problemas, eventos, tópicos em particular e seus atributos (LIU, 2012).

Essa definição expande o problema a um conjunto muito grande de análise, o qual compreende muitos subconjuntos de diferentes tarefas, tais como: análise de sentimento, mineração de opinião, extração de opinião, mineração de sentimento, análise de subjetividade, análise de afeição, análise de emoção, mineração de *reviews*, etc. Contudo, todas essas análises se enquadram sob a classificação de análise de sentimento ou mineração de opinião. Enquanto comercialmente o termo análise de sentimento é mais comum, academicamente ambos: análise de sentimento e mineração de opinião são frequentemente utilizados, sendo que ambos representam basicamente o mesmo campo de estudo.

O termo análise de sentimento apareceu primeiro em (NASUKAWA e YI, 2003), e o termo mineração de opinião apareceu primeiro em (DAVE et al., 2003). Contudo, as pesquisas sobre sentimentos e opiniões apareceram antes (DAS e CHEN, 2001; MORINAGA et al., 2002; PANG et al., 2002; TONG, 2001; TURNEY, 2002; WIEBE, 2000). Como propõem (LIU, 2012) dado que uso desses termos são intercambiáveis, adota-se o termo "opinião" para denotar opinião, sentimento, avaliação e emoção. O conceito de opinião é amplo cabe seu uso nos termos propostos, contudo, por uma questão didática eles serão distinguidos quando necessário. Análise de sentimento e mineração de opinião focam principalmente em opiniões que expressam ou sugerem sentimentos positivos ou negativos.

A análise de sentimento tem se tornado cada vez mais utilizada e pesquisada, há muitas razões para isso. Para (LIU, 2012), isso se deve a três fatores em particular: o primeiro seria a grande quantidade de aplicações, presentes em quase todos os domínios. Toda a indústria em

[15] http://atlassian.gridgain.com/wiki/
[16] https://github.com/gridgain/gridgain

torno da análise de sentimento floresceu devido a proliferação de aplicações comerciais, que, sistematicamente, promoveu em grande escala o interesse em pesquisas voltadas para essa área. O segundo que essa é uma área que ainda oferece muito problemas a serem pesquisados, que dificilmente foram estudados anteriormente. Terceiro que, pela primeira vez na história da humanidade, há disponibilidade de um imenso volume de dados de opinião nas mídias sociais na *web*. Sem esses dados, muitas pesquisas não seriam possíveis. Acrescentar-se-ia ainda baseado em (CHEN et al., 2014) o grande volume de dados de opinião presentes em comentários de *blog* e em páginas de comércio *online*.

Esse aumento de interesse, não por coincidência, ocorre com a explosão de crescimento das mídias sociais. Na verdade, a análise de sentimento hoje possui como principal objeto de pesquisa as mídias sociais.

2.3.1 Aplicações para Análise de Sentimento

As opiniões ocupam uma posição de destaque em quase todas as atividades humanas, pois estas influenciam nosso comportamento e escolhas. Sempre que precisamos tomar uma decisão consultamos a opinião de alguém, seja este alguém especialista com domínio na área, ou não (LIU, 2012). Hoje em dia, empresas das mais diversas áreas de atuação e organizações buscam saber os que seus clientes pensam e como consomem. Nesse cenário, consumidores individuais buscam saber opiniões a respeito de um produto ou serviço antes de adquiri-lo, opinião a respeito de um candidato a um cargo político antes de votar em uma eleição. É comum empresas conduzirem pesquisas de opinião em grupos-alvos em busca de opinião a respeito da qualidade de um produto ou serviço.

Com o massivo crescimento de dados gerados pelos meios de comunicação na *web* (*reviews*, *blog*, fóruns de discussão, postagens no Twitter e nas redes sociais), tanto indivíduos quanto organizações estão cada vez mais usando esse conteúdo na tomada de decisões (LIU, 2012). Após essa explosão das mídias sociais, uma mudança é notável. Em geral, quando uma pessoa deseja comprar um produto ou serviço, antes de consultar um amigo ou familiar, ela busca a página desse produto na internet ou nas redes sociais e busca por *reviews* ou comentários a respeito do produto ou serviço e, após adquirir ou não, a pessoa deixa o próprio comentário, *review* ou nota para aquele produto ou serviço. Empresas e organizações não mais conduzem pesquisas de opinião buscando opinião pública, pois há abundância dessas informações na *web.* Contudo, isto implica numa tarefa exaustiva para um analista ler cada

postagem e filtrar informação relevante nessa torrente de informações espalhadas em diversos sites e mídias sociais na *web*. Sistemas para análise de sentimento são, então, necessários.

Nos últimos anos, é notável o aumento de *posts* expressando opinião nas mídias sociais influenciando uma mudança nos negócios, impactando diretamente nossos sistemas sociais e políticos (LIU, 2012). Esses *posts* são responsáveis, por exemplo, por mobilizar massas por mudanças políticas, como nos protestos de junho de 2013, a princípio, em São Paulo, e que em pouco tempo se espalharam pelo Brasil. Por essas razões, surgiu uma demanda pela coleta e estudo de opiniões na *web*. A *web* não é a única fonte de informações de dados ou documentos de opinião, muitas organizações públicas e privadas mantêm dados internos, como pesquisas de opinião, *feedback* coletado de *emails* e *call centers*.

Essas diversas aplicações são responsáveis por aumentar o interesse pela análise de sentimento e mineração de opinião, que se espalharam por quase todos os domínios possíveis, como serviços, saúde, produtos de consumo, finanças e eleições políticas. Só nos Estados Unidos, segundo estimativas (IDC, 2011), cerca de 60 companhias *start-up* estão envolvidas com projetos voltados para a área. As grandes companhias, como Google, IBM, Microsoft, HP, SAP e SAS também possuem produtos e serviços direcionados à análise de sentimento e mineração de opinião. Essas aplicações práticas e interesses comerciais e industriais tem provido grandes motivações para pesquisas em análise de sentimento. O presente trabalho demonstra esse interesse por pesquisas na área.

2.3.2 Definições de Opinião

Uma opinião é uma quadrupla (g, s, h, t), onde g é a opinião (ou sentimento) alvo, s é o sentimento a respeito do alvo, h é detentor da opinião, e t é o tempo em que a opinião é expressada (LIU, 2012). Essa definição, apesar de bem concisa, não é exatamente prática, especialmente no domínio de *reviews online* de produtos, serviços e marcas porque a descrição completa do alvo pode ser complexa, podendo não ocorrer na mesma sentença.

Uma entidade e é um produto, serviço, tópico, problema, pessoa, organização ou evento. É descrita com um par, e: (T, W), onde T é uma hierarquia de partes, subpartes, e assim por diante e W é um conjunto de atributos de e. Cada parte da subparte também possui seu próprio conjunto de atributos. Essa definição essencialmente descreve a decomposição hierárquica de entidade baseada na relação parte-de.

Após a decomposição de opinião apresentada anteriormente, pode-se redefinir uma opinião (HU e LIU, 2004; LIU, 2010).

Uma opinião é uma quíntupla, (e_i, a_{ij}, s_{ijkl}, h_k, t_l), onde e_i é o nome de uma entidade, a_{ij} é um aspecto de e_i, s_{ijkl} é o sentimento no aspecto a_{ij} da entidade e_i, h_k é o detentor da opinião e t_l é o tempo em que a opinião é expressada por h_k (LIU, 2012). O sentimento s_{ijkl} é positivo, negativo, ou neutro, ou expressado com diferentes níveis de força/intensidade, por exemplo, em *reviews*, costuma-se utilizar uma escala de 1 a 5 estrelas na maioria dos sites de *reviews* sobre filmes. Quando uma opinião é uma entidade em si como um todo, o aspecto especial GENERAL é usado para denota-lo. Onde, e_i e a_{ij} juntos representam a opinião-alvo.

2.3.3 Tipos de Opinião

Uma opinião regular é frequentemente referida a uma simples opinião na literatura e possui dois subtipos (LIU, 2006, 2011). Uma opinião direta refere-se a uma opinião expressada diretamente a uma entidade ou um aspecto da entidade; por exemplo, “A cor desse carro é bonita”. Uma opinião indireta é aquela expressada indiretamente sobre uma entidade ou uma de suas características baseado nos efeitos de alguma outra entidade. Esse subtipo é comum no domino médico; por exemplo, “Depois da injeção do remédio, minhas juntas estão péssimas” descreve um efeito indesejável da droga nas “juntas” do paciente, o que indiretamente causa uma opinião ou sentimento negativo sobre o medicamento.

A maioria dos estudos sobre opiniões regulares focam em opiniões diretas. Elas são mais simples de lidar, em contrapartida das opiniões indiretas que são mais difíceis de trabalhar por conta do tratamento da sentença que deve levar em consideração mais de uma entidade ao avaliar um domínio.

Uma opinião comparativa expressa uma relação de similaridades ou diferenças entre duas ou mais entidades e/ou preferência do detentor da opinião baseado em aspectos compartilhados das entidades (JINDAL e LIU, 2006a, 2006b). Por exemplo, as sentenças, “Cupuaçu é mais saboroso que Buriti” e “Cupuaçu é mais gostoso” expressam duas opiniões comparativas.

Uma opinião explicita é a uma afirmação subjetiva que dá uma opinião regular ou comparativa; por exemplo, “Cupuaçu é mais gostoso”. Uma opinião implícita é uma afirmação objetiva que implica uma opinião regular ou comparativa. Em geral, expressa a fato desejável,

ou não; por exemplo, “Comprei esse celular, semana passada e ele desliga sozinho o tempo todo.”

Opiniões explicitas são mais facilmente detectadas e classificadas que opiniões implícitas, o que leva a grande maioria dos pesquisadores a trabalhar com opiniões explicitas. Na pesquisa de opiniões implícitas destaca-se (ZHANG e LIU, 2011b) por conta de seu trabalho abordar o alfabeto tradicional chinês para essa análise, sendo o pioneiro trabalho proposto a essa análise em um idioma diferente do inglês americano (LIU, 2012). Em uma direção diferente, (GREENE e RESNIK, 2009) estudaram a influência de escolhas sintáticas nas percepções de sentimentos implícitos (por exemplo, para a mesma estória, diferentes manchetes podem implicar em diferentes sentimentos).

2.3.4 Níveis de Análise de Sentimento

A importância da área de pesquisa envolvida e dos resultados esperados da análise desejada determinarão o nível em que será aplicada a análise de sentimento.

- **Análise a nível de documento:** consiste em classificar se um documento inteiro expressa um sentimento positivo ou negativo como proposto em (PANG et al., 2002) e (TURNEY, 2002). Este tipo de análise assume que cada documento expressa opinião sobre uma única entidade. Por exemplo, dado um conjunto de *posts* constituindo opinião sobre um produto, cabe ao sistema determinar se esse conjunto de *posts* projetam uma opinião positiva ou negativa a respeito do produto.

- **Análise a nível de sentença:** classifica se determinada expressão expressa uma opinião positiva, negativa ou neutra como, proposta por (WIEBE et al., 1999). Este nível de análise é muito próximo da classificação de subjetividade, que distingue sentenças (chamadas de sentenças objetivas) expressam informação fatual das sentenças (chamadas sentenças subjetivas) que expressa um ponto de vista subjetivo ou opinião.

- **Análise a nível de aspecto e entidade:** As análises em nível de documento e em nível de sentença não revelam o que exatamente as pessoas gostaram ou não (tomando como exemplo o conjunto de *posts* sobre um produto). O sistema em nível de entidade realiza uma refinada análise que, em vez de construções de linguagem (documentos, parágrafos, sentenças ou cláusulas), busca diretamente a opinião em si (HU e LIU, 2004). É baseada na ideia que uma opinião consiste de um sentimento (positivo ou

negativo) e um alvo (de opinião). Perceber a importância da opinião-alvo também ajuda a entender o problema da análise de sentimento melhor.

Em (JINDAL e LIU 2006b), é proposta a existência de dois tipos de opinião: as opiniões regulares e opiniões comparativas. Uma opinião regular expressa um sentimento apenas sobre uma entidade particular ou um aspecto de entidade; por exemplo, “A VIVO possui bom sinal” que expressa um sentimento positivo sob o aspecto qualidade do sinal da operadora VIVO. Uma opinião comparativa compara múltiplas entidades baseada em algum dos seus aspectos compartilhados; por exemplo, “A VIVO possui sinal melhor do que a TIM” que compara as operadoras VIVO e TIM baseado na sua qualidade do sinal (um aspecto) e expressa uma preferência pela operadora VIVO.

2.3.5 Técnicas Para Análise de Sentimento

Classificação de sentimento estabeleceu-se como uma tarefa geral de classificação de uma sequência de texto de entrada em certos tipos ou classificando o texto de entrada com uma certa pontuação (BESPALOV et al., 2011).

Classificação de sentimento pode ser realizada em nível de documento, nível de sentença ou em nível de aspecto ou característica (PANG e LEE, 2008). Na classificação em nível de documento, o documento inteiro é classificado como contendo um sentimento positivo ou negativo. A classificação em nível de sentença classifica cada sentença como positiva, negativa ou neutra. A classificação a nível de aspecto ou característica se preocupa com a identificação e extração de características da fonte dos dados.

Existem duas abordagens principais para análise de sentimento; baseada em aprendizado de máquina e baseado em léxico (PANG e LEE, 2008). Aprendizado de máquina usa técnicas de classificadores (por exemplo, SVM) para classificar o texto. Métodos baseados em léxico, usam dicionários com palavras de opinião e casam a palavra de entrada com o dicionário para determinar a polaridade. A seguir, abordam-se as essas duas técnicas em detalhes para melhor compreensão do processamento realizado pelas mesmas.

- **Técnicas baseadas em Aprendizado de Máquina**: As técnicas baseadas nessa abordagem pertencem às técnicas de classificação supervisionada. Nesse tipo de técnica, dois conjuntos de documentos são necessários: conjunto de treino e conjunto de teste (PANG e LEE, 2008). Um conjunto de treino é usado por um classificador automático

para aprender a diferenciação de características de documento e o conjunto de teste é usado para checar quão bem o classificador executa.

Algumas das técnicas de aprendizado mais comum segundo (CHEN et al., 2014) são: *Naive Bayes* (NB), *Maximum Entropy* (ME) e *Suport Vector Machine* (SVM). Aprendizado de máquina inicia com a coleta do conjunto de dados de treinos, após essa coleta, é realizado o treino do classificador com os dados de treino. Uma vez que é escolhida a técnica de classificação, o próximo passo é escolher qual característica será abordada, pois essa escolha definirá como o documento será representado.

As características mais comuns abordadas na classificação de sentimento segundo (PANG e LEE, 2008) são: presença do termo e sua presença (essas características incluem uni-gramas ou n-gramas e sua presença ou frequência); informação parte do discurso (é usada para guiar seleção de característica); negação é também uma característica importante para detecção de sentimento reverso; e frases e palavras de opinião (usadas principalmente para identificar a orientação semântica).

- **Técnicas baseadas léxico**: são técnicas de aprendizado não supervisionado, nas quais a classificação é feita por meio da comparação de características entre um texto dado e um dicionário de sentimento (PANG e LEE, 2008). O dicionário de sentimento contém uma lista de palavras e expressões usadas para expressar sentimentos e opiniões. Os passos básicos das técnicas baseadas são demostrados abaixo (ANNETT e KONDRAK, 2008):

1. Pré-processamento de cada texto (por exemplo, remoção de *tags* HTML)
2. Inicializar a pontuação de sentimento total no texto: s ← 0.
3. Tokenizar o texto[17]. Para cada *token*, checar se está presente no dicionário de sentimentos.
    a. Se o *token* está presente no dicionário,
        i. Se o *token* é positivo, então s ← 0 + w.
        ii. Se o *token* é negativo, então s ← 0 - w.
4. Olhar a pontuação de sentimento total no texto s,
    a. Se s > entrada, então classificar o texto com positivo.
    b. Se s < entrada, então classificar o texto com negativo.

[17] Dividir um texto ou frase em palavras individuais para simplificar a análise.

Existem três métodos para construir um sentimento léxico (PANG e LEE, 2008): construção manual, métodos baseado em *corpus* e métodos baseado em dicionário. A construção manual de sentimento léxico é uma tarefa difícil e que consome muito tempo (ANNETT e KONDRAK, 2008). Técnicas baseadas em *corpus* depende de padrões sintáticos em um *corpus* maior. Esse tipo de abordagem permite a produção de palavras de opinião com relativamente alta acurácia e ajuda a encontrar um domínio especifico para as palavras de opinião e sua orientação. Na abordagem baseada em dicionário, a ideia é, primeiramente, colecionar um pequeno conjunto de palavras manualmente que expressa orientação, e então este cresce por meio de buscas no dicionário *WordNet*[18] por seus sinônimos e antônimos. Essas novas palavras são adicionadas à lista, e a próxima iteração começa, a para somente quando não há mais palavras encontradas (PANG e LEE, 2008).

Métodos baseadas em aprendizado de máquina mostram melhor performance que os métodos baseados em léxico (ANNETT e KONDRAK, 2008). No entanto, os métodos não supervisionados são importantes também porque métodos supervisionados demandam imensos conjuntos de dados de treino rotulado que, em geral, são difíceis de encontram e, na sua maioria, são pagos (e caros), enquanto a aquisição de dados não rotulados é mais fácil. A maioria dos domínios, exceto *reviews* de filmes possuem poucos dados de treino rotulados. Nesse caso, métodos não supervisionados são mais úteis para o desenvolvimento de aplicações (ANNETT e KONDRAK, 2008).

A metodologia proposta por este trabalho utiliza a técnica baseada em léxico com auxílio de dicionário. Pois essa técnica se adéqua melhor ao modelo GridGain MapReduce, utilizado neste trabalho, para permitir o uso de uma estrutura de *hardware* existente e permitir processamento paralelo de dados, buscando alto desempenho. Se adéqua no sentido em a lógica da computação pode ser distribuída usando aspectos nativos das diretivas *map* e *reduce* presentes no modelo GridGain MapReduce, simplificando a codificação da aplicação e análise de sentimento.

Neste capítulo, foi apresentada a fundamentação teórica utilizada no desenvolvimento da metodologia proposta para análise de sentimento. Como a metodologia visa realizar análise de um conjunto de dados com características big data, este capítulo, abordou o conceito de big data, tecnologias relacionadas (computação em nuvem, internet of things, data centers e

[18] http://wordnet.princeton.edu/

hadoop), análise big data (métodos tradicionais de análise e principais métodos de processamento big data utilizados atualmente) e ferramentas utilizadas para análise de dados, com destaque para o *framework* GridGain, ferramenta empregada na análise proposta neste trabalho.

A análise de sentimento foi exposta discutindo sua definição, finalidade e principais aplicações nas atividades humanas. Foram apresentadas também, os níveis aplicáveis para análise de sentimento e as técnicas utilizadas para realizar esta análise. Dentre as duas técnicas apresentadas, a técnica baseada em léxico foi escolhida para desenvolvimento da metodologia expendida no capítulo a seguir, que apresenta o estudo de caso.

3 Estudo de Caso

Este capítulo apresenta a metodologia proposta como objetivo deste trabalho, a qual é demonstrada por meio de um estudo de caso que desenvolve e utiliza os conceitos abordados anteriormente neste trabalho.

A metodologia é apresentada de forma global no tópico 3.2. Posteriormente são demonstradas em detalhes cada uma das etapas que compõem a metodologia. A etapa da coleta da base de dados é discutida primeiro. Em seguida, a etapa de adaptação da base de valores numéricos para a sua correspondente opção no questionário da pesquisa de opinião. A etapa de armazenamento dos dados em banco de dados *open source*. A etapa da utilização do GridGain MapReduce para realizar o paralelismo de processamento sobre a estrutura de *grid* do GridGain localmente (não impedindo, como será demonstrado posteriormente, a utilização de uma estrutura de máquinas em rede) e da aplicação do algoritmo desenvolvido para realizar a classificação binária baseado em léxico e, finalmente, a etapa da validação dos resultados.

3.1 Software e Hardware utilizados

Para implementação da aplicação, foi utilizada a linguagem Java, através do ambiente de desenvolvimento Eclipse Luna, disponível no site da Eclipse *Foundation*[19]. A utilização da linguagem Java no ambiente Eclipse torna simples a migração entre plataformas Microsoft Windows, Linux ou Mac OS.

Para acesso as informações da base foi empregado o *software* IBM SPSS Statistics 22, pois todas as informações da base são disponibilizadas apenas em formato compatível com o *software* da IBM. Como toda metodologia utiliza *softwares* gratuitos e/ou de código aberto, os dados foram exportados para uma planilha de trabalho em formato compatível com Excel e Calc. Como será observado posteriormente, o Calc foi utilizado para adaptação da base, pois este possui versões compatíveis com ambiente Microsoft Windows e Linux.

Para armazenamento dos dados de pesquisa de opinião adaptados, foi utilizado o banco de dados *open source* MySQL Community Server 5.6.2 0.

O *framework* GridGain – plataforma utilizada na metodologia para se alcançar o processamento paralelo em memória sobre uma estrutura de máquinas existente – foi obtido

[19] http://www.eclipse.org/download/

por meio de sítio web[20], sob uma licença Apache versão 2.0 de janeiro de 2004. O GridGain é disponibilizado em 4 edições otimizadas para tarefas especificas; são elas: In-Memory Computing Plataform (edição completa com todas as dependências), In-Memory HPC (otimizada para alta performance em clusterização, computações, mensagens e processamento de eventos), In-Memory Data Grid (voltada para *caching* distribuído de dados em memória e alta performance de processamento) e, a última, In-Memory Streaming (otimizada para processamento em tempo real e *Complex Event-Processing*). Todas essas edições estão disponíveis na versão 6.2.0 lançada em 25 de agosto de 2014, para os ambientes Microsoft Windows, Linux e Mac OS. A edição utilizada no desenvolvimento deste trabalho foi a In-Memory Computing Plataform, por conter todas as dependências.

As máquinas utilizadas para testes foram: um Core I5, 6GB de memória e 1TB de HD[21], Windows e core I7 8GB de memória, 500GB de HD, Linux.

3.2 Metodologia Proposta

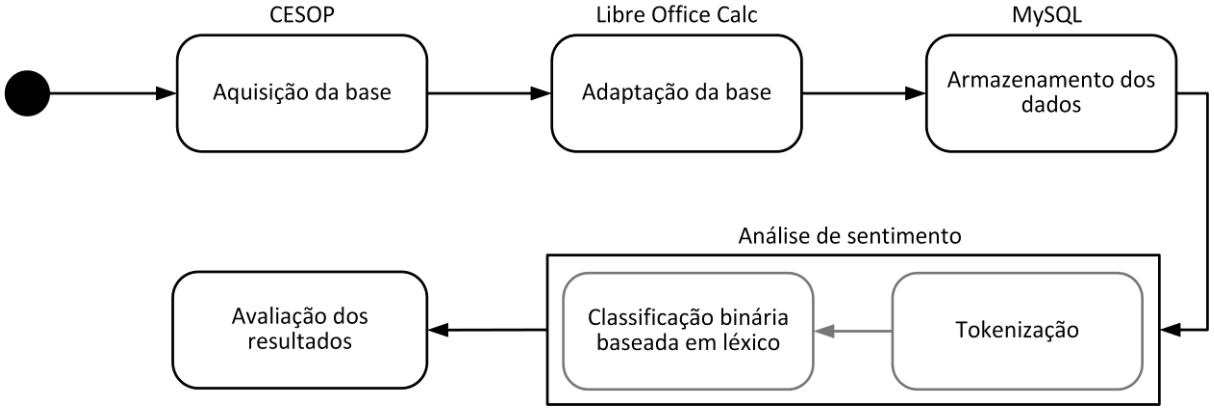


Figura 3.1: Etapas da metodologia proposta pelo trabalho.

Os passos da metodologia proposta por este trabalho são apresentados na Figura 3.1. São eles: aquisição da base, adaptação da base, armazenamento dos dados, análise de sentimento (composta de duas subtarefas: tokenização e classificação binária baseada em léxico) e a avaliação de resultados.

[20] http://www.gridgain.org/download
[21] Hard Drive (disco rígido)

3.2.1 Aquisição da base

A base de dados de pesquisas de opinião pública foi adquirida junto ao Centro de Estudos de Opinião Pública (CESOP) e compreende tópicos de pesquisa de opinião em particular, tais como: opinião pública sobre fatores econômicos e sociais além de opinião sobre assuntos políticos administrativos brasileiros. Esses tópicos em particular foram escolhidos pois o conjunto de respostas obtido com esse tipo de pesquisa é composto de palavras de sentimento, permitindo a análise de sentimento ou mineração de opinião. Outro fator importante para o trabalho proposto era disponibilidade de acesso a base de forma gratuita para propósitos acadêmicos.

O CESOP é uma iniciativa interdisciplinar combinada entre os âmbitos acadêmico e empresarial na área de opinião pública, comportamento político e social e metodologia de pesquisa (CESOP, 2014). O CESOP é estruturado sobre uma composição tripartite interuniversitária e empresarial (Universidade Estadual de Campinas (UNICAMP), Universidades e Centros Científicos e Empresas de pesquisa) fundado em 1992, que tem como objetivo a recuperação, organização e disponibilização para o pesquisador dos *surveys* realizados por empresas e instituições científicas nas áreas predominantes de opinião pública, sociedade, temas culturais, comportamento em geral, além de dados socioeconômicos em educação.

A Figura 3.2 mostra a página inicial do banco de dados do CESOP, a qual apresenta algumas informações acerca deste, do seu conteúdo, do formato de disponibilização dos dados, sobre como realizar pesquisas no índice do banco de dados e sobre as empresas e instituições que cedem os dados ao CESOP.

Figura 3.2: Página inicial do banco de dados do CESOP.

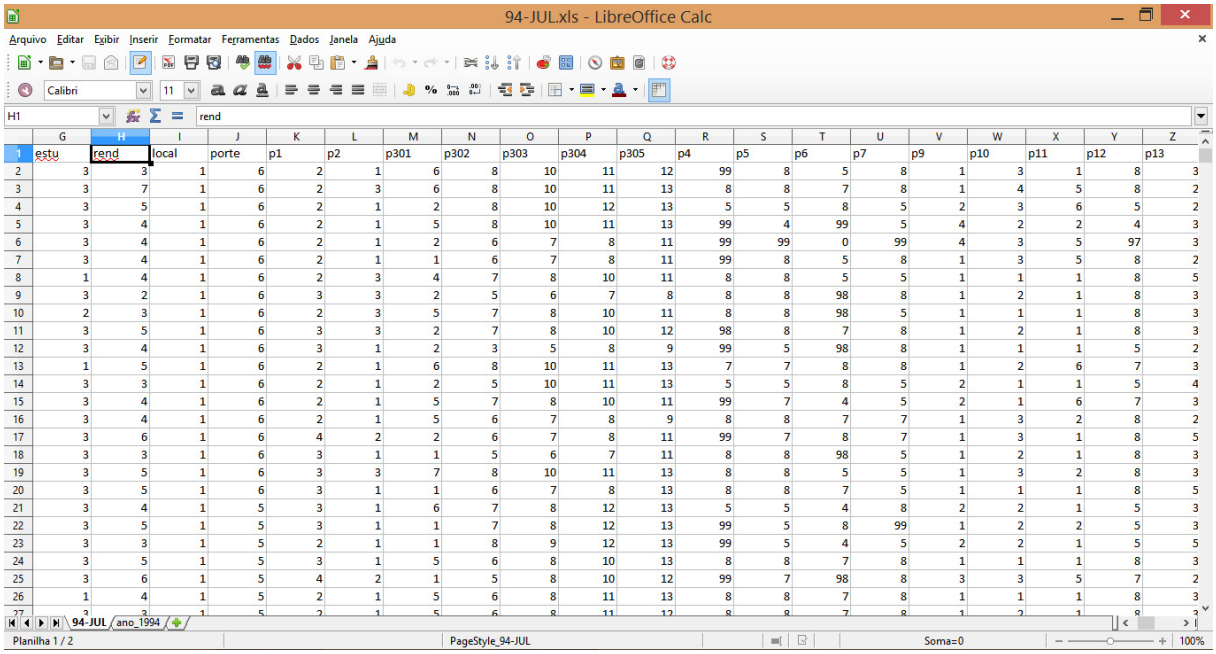
A base de dados adquirida junto ao CESOP possui dados de pesquisa de opinião relativos aos anos de 1994, 1996, 1999, 2001, 2003, 2004 e 2013. A base é composta de valores numéricos correspondentes às alternativas disponíveis no questionário da pesquisa de opinião (Figura 3.3).

3.2.2 Adaptação da base

As técnicas baseadas em léxico necessitam que os dados analisados estejam no formato textual, para que o conjunto de dados de entrada possa ser comparado com o dicionário de palavras que expressam sentimento e opiniões. Isso posto, se fez necessário realizar uma adaptação da base de dados.

A Figura 3.3 apresenta os dados da pesquisa de opinião, coletados pelo instituto IBOPE em todo brasil no mês de julho de 1994. Pode-se perceber que dado a análise realizada sobre esse tipo de informação ser de natureza estatística, ela é disposta com as respostas em formato

numérico correspondente à alternativa de resposta dada pelo entrevistado durante a realização da pesquisa, visto que isso facilita a interpretação e tratamento dessas informações para fins estatísticos.



	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	estu	rend	local	porte	p1	p2	p301	p302	p303	p304	p305	p4	p5	p6	p7	p9	p10	p11	p12	p13
2	3	3	1	6	2	1	6	8	10	11	12	99	8	5	8	1	3	1	8	3
3	3	7	1	6	2	3	6	8	10	11	13	8	8	7	8	1	4	5	8	2
4	3	5	1	6	2	1	2	8	10	12	13	5	5	8	5	2	3	6	5	2
5	3	4	1	6	2	1	5	8	10	11	13	99	4	99	5	4	2	2	4	3
6	3	4	1	6	2	1	2	6	7	8	11	99	99	0	99	4	3	5	97	3
7	3	4	1	6	2	1	1	6	7	8	11	99	8	5	8	1	3	5	8	2
8	1	4	1	6	2	3	4	7	8	10	11	8	8	5	5	1	1	1	8	5
9	3	2	1	6	3	3	2	5	6	7	8	8	8	98	8	1	2	1	8	3
10	2	3	1	6	2	3	5	7	8	10	11	8	8	98	5	1	1	1	8	3
11	3	5	1	6	3	3	2	7	8	10	12	98	8	7	8	1	2	1	8	3
12	3	4	1	6	3	1	2	3	5	8	9	99	5	98	8	1	1	1	5	2
13	1	5	1	6	2	1	6	8	10	11	13	7	7	8	8	1	2	6	7	3
14	3	3	1	6	2	1	2	5	10	11	13	5	5	8	5	2	1	1	5	4
15	3	4	1	6	2	1	5	7	8	10	11	99	7	4	5	2	1	6	7	3
16	3	4	1	6	2	1	5	6	7	8	9	8	8	7	7	1	3	2	8	2
17	3	6	1	6	4	2	2	6	7	8	11	99	7	8	7	1	3	1	8	5
18	3	3	1	6	3	1	1	5	6	7	11	8	8	98	5	1	2	1	8	3
19	3	5	1	6	3	3	7	8	10	11	13	8	8	5	5	1	3	2	8	3
20	3	5	1	6	3	1	1	6	7	8	13	8	8	7	5	1	1	1	8	5
21	3	4	1	5	3	1	6	7	8	12	13	5	5	4	8	2	2	1	5	3
22	3	5	1	5	3	1	1	7	8	12	13	99	5	8	99	1	2	2	5	3
23	3	3	1	5	2	1	1	8	9	12	13	99	5	4	5	2	2	1	5	5
24	3	5	1	5	3	1	5	6	8	10	13	8	8	7	8	1	1	1	8	3
25	3	6	1	5	4	2	1	5	8	10	12	99	7	98	8	3	3	5	7	2
26	1	4	1	5	2	1	5	6	8	11	13	8	8	7	8	1	1	1	8	3

Figura 3.3: Dados da pesquisa de opinião coletados pelo IBOPE em 1994.

O questionário de uma pesquisa de opinião inclui muitas questões cujas respostas não enquadram-se nos requisitos necessários para realização de uma análise de sentimento (que a resposta correspondente fosse uma palavra ou expressão de sentimento ou opinião) como uma questão sobre o sexo do entrevistado por exemplo. Após escolha em conjunto com a orientadora de um grupo de questões que atendesse completamente os requisitos buscados, foi realiza a correspondência entre o número indicado como resposta e sua respectiva alternativa textual no questionário disponibilizado junto aos dados da pesquisa.

A Figura 3.6 apresenta as perguntas (p1 e p2) do questionário da pesquisa de opinião feita pelo instituto IBOPE em todo brasil no mês de julho de 1994 e suas respectivas alternativas de resposta apresentadas pelo entrevistador ao entrevistado.

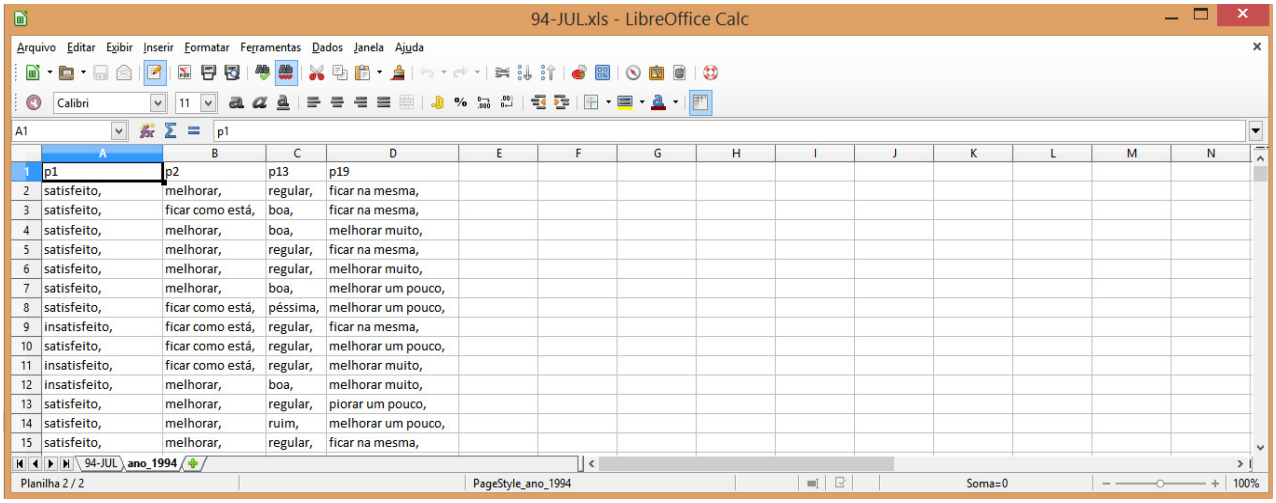
```
var label p1 "Para começar, como o(a) sr(a) diria que se sente com relação à vida que vem levando hoje?
O(a) sr(a) está:".
val label p1
1 "Muito satisfeito"
2 "Satisfeito"
3 "Insatisfeito"
4 "Muito insatisfeito"
5 "Não sabe/não opinou".

var label p2 "E qual a sua expectativa em relação ao futuro do país, o(a) sr(a) diria que a situação vai
melhorar, piorar, ou ficar como está?".
val label p2
1 "Melhorar"
2 "Piorar"
3 "Ficar como está"
4 "Não sabe/não opinou".
```

Figura 3.4: Questões (p1, p2) da pesquisa de opinião feita pelo IBOPE em 1994.

A adaptação compreende a tarefa de localizar e substituir os valores numéricos (Figura 3.3) por sua respectiva resposta textual corresponde no questionário. Por exemplo, tomando-se a pergunta p1 da pesquisa de opinião feita pelo IBOPE em 1994 (Figura 3.4), que busca saber dos entrevistados como este se sente sobre a vida que este está levando no ano de 2004, os entrevistados dispõem de 5 alternativas (Muito satisfeito, Satisfeito, Insatisfeito, Muito insatisfeito e Não sabe/não opinou) uma vez que o entrevistador coleta a resposta, a mesma é registrada sob um valor numérico (1, 2, 3, 4, 5) corresponde às 5 alternativas disponíveis da pergunta p1. Para realizar a substituição do valor numérico por sua respectiva resposta textual, foi utilizado o recurso de localizar e substituir do software Calc[22].

O resultado da tarefa de adaptação realizada sobre o grupo de questões selecionadas pode ser melhor observado através da Figura 3.5 que exibe uma pasta de trabalho no Calc com as respectivas respostas textuais ao invés do número correspondente como visto anteriormente na Figura 3.3.

[22] Software da suíte *office open source* LibreOffice para criação e edição de planilhas.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	p1	p2	p13	p19										
2	satisfeito,	melhorar,	regular,	ficar na mesma,										
3	satisfeito,	ficar como está,	boa,	ficar na mesma,										
4	satisfeito,	melhorar,	boa,	melhorar muito,										
5	satisfeito,	melhorar,	regular,	ficar na mesma,										
6	satisfeito,	melhorar,	regular,	melhorar muito,										
7	satisfeito,	melhorar,	boa,	melhorar um pouco,										
8	satisfeito,	ficar como está,	péssima,	melhorar um pouco,										
9	insatisfeito,	ficar como está,	regular,	ficar na mesma,										
10	satisfeito,	ficar como está,	regular,	melhorar um pouco,										
11	insatisfeito,	ficar como está,	regular,	melhorar muito,										
12	insatisfeito,	melhorar,	boa,	melhorar muito,										
13	satisfeito,	melhorar,	regular,	piorar um pouco,										
14	satisfeito,	melhorar,	ruim,	melhorar um pouco,										
15	satisfeito,	melhorar,	regular,	ficar na mesma,										

Figura 3.5: Pasta de trabalho com os dados adaptados da pesquisa de opinião feita pelo IBOPE em 1994.

3.2.3 Armazenamento de Dados

O armazenamento de dados é outro aspecto importante da metodologia aqui proposta, busca-se uma metodologia que possa ser migrada entre plataformas (Windows, Linux). A melhor forma de se obter essa liberdade é o com uso de ferramentas *open source*. O banco de dados escolhido para o armazenamento dos dados foi o banco de dados MySQL Community Server 5.6.20.0, sendo essa a versão mais atual no momento da elaboração desta monografia.

A escolha do MySQL é justificada por esta ser uma ferramenta consolidada e conter vasta documentação disponível, além de manter uma interface de gerenciamento bastante simples e intuitiva.

Os dados, após a adaptação (que fora demonstrada no tópico 3.2.2) estão em uma pasta de trabalho do Calc. Para realizar a exportação desses dados no banco de dados MySQL Community Server de maneira simples e segura, foi utilizada o *software open source* MySQL-Front 5.3, que permite gerenciamento de dados entre o banco de dados MySQL e fontes externas de dados em formatos variados (como arquivos SQL, planilhas de trabalho do Excel e Access, arquivos XML, HTML e PDF) por meio de interface visual como pode ser observado na Figura 3.6.

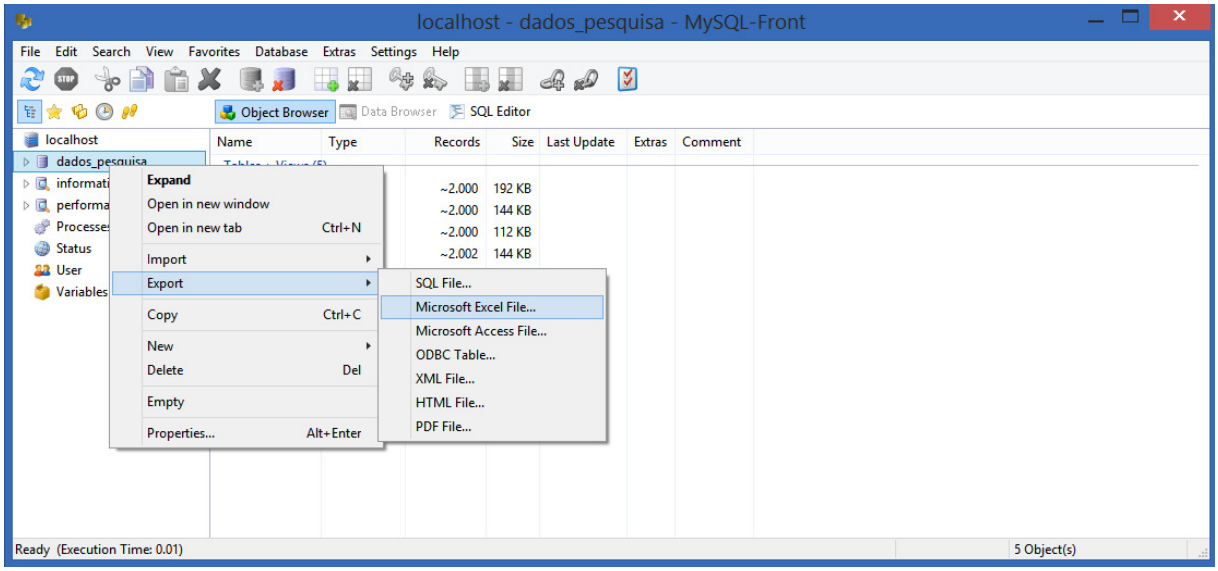


Figura 3.6: Exportação pasta de trabalho para o banco de dados.

As tabelas criadas para manter os dados seguem um padrão: com a palavra ano seguida do ano correspondente aos dados de origem da pesquisa em formato numérico (ano_1994, por exemplo). A Figura 3.7 apresenta a situação do banco após a exportação dos dados e exibe também uma consulta aos dados da tabela ano_1994.

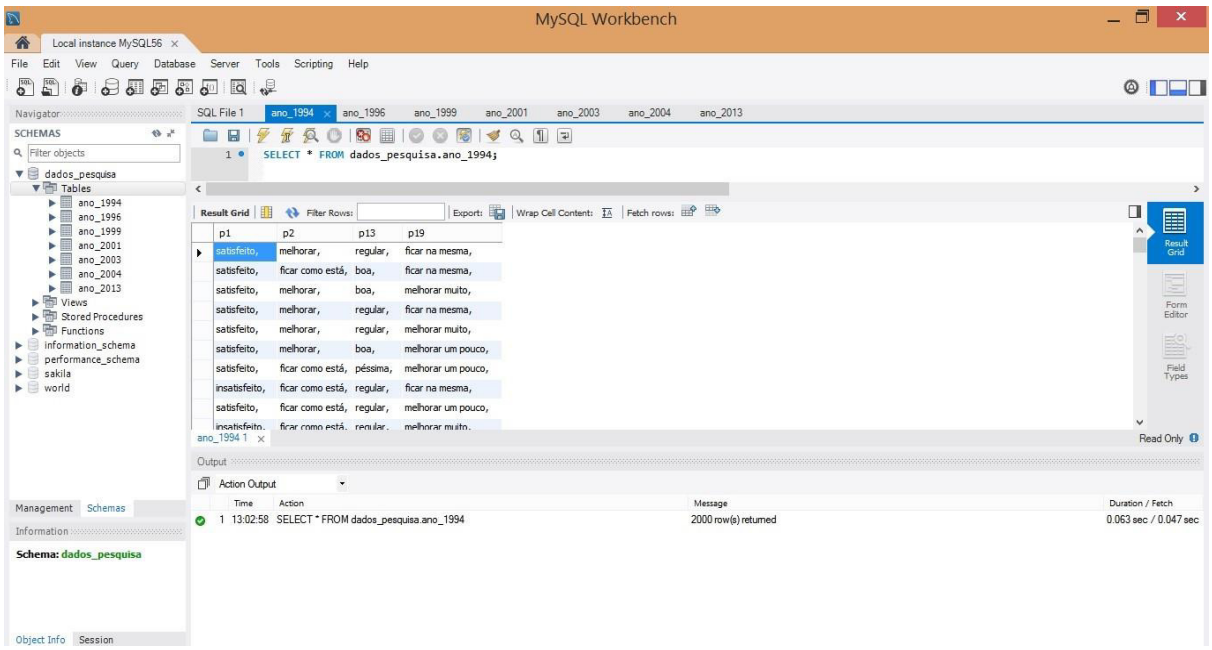


Figura 3.7: Dados de pesquisa do ano de 1994 armazenados no banco de dados.

A etapa de armazenamento dos dados da pesquisa permite que esses dados possam ser migrados entre ambientes de trabalho (Windows e Linux), além de permitir que a aplicação possa ter acesso aos dados de forma padronizada garantindo a integridade, segurança e simplicidade durante esse acesso.

3.2.4 Análise de Sentimento

A etapa de análise de sentimento proposta pela metodologia apresentada utiliza uma técnica baseada em léxico (discutida no tópico 2.3.5) e propõem-se a classificar o sentimento total da pesquisa de opinião ano a ano de forma bipolarizada, ou seja, como positiva ou negativa.

Considerando que a metodologia deseja processar big data, a etapa de análise de sentimento está subdividida em duas suas tarefas (tokenização e polarização), buscando simplificação e otimização de processamento em cada nó GridGain que esteja executando. A análise de sentimento compreende a penúltima etapa da cadeia de valor big data (análise big data) proposta na metodologia apresentada neste trabalho. Esta etapa, segundo (BEYER e LANEY, 2012) pode ser considerada o epicentro da transformação do dado em informação útil. Essa transformação é realizada utilizando o modelo de programação distribuída MapReduce, apresentado anteriormente no tópico 2.2.1 e demonstrado na Figura 2.2.

A adoção do modelo MapReduce para realizar essa análise, não só permite utilizar o poder de processamento de uma estrutura existente (pois MapReduce realiza a paralelização automática da computação), mas também alocar a lógica da análise de sentimento utilizando as tarefas *map* e *reduce*, o que simplifica o processo da análise e a codificação da aplicação.

Uma visão global do processo MapReduce utilizado na metodologia proposta pode ser observada na Figura 3.8. Esse processo será explicado em detalhes a partir de agora.

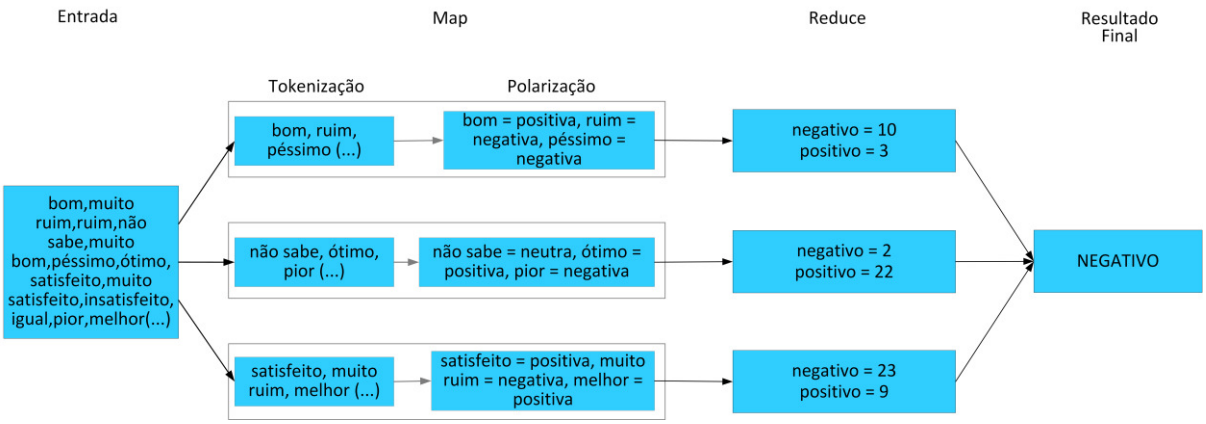


Figura 3.8: Visão global do processo MapReduce para classificação de sentimento binário.

A entrada dos dados é feita por meio de uma *string* correspondente a todas as respostas dadas às perguntas feitas na pesquisa de opinião, armazenadas no banco de dados MySQL e organizadas em tabelas por ano em que a pesquisa foi realizada. Uma classe da aplicação

implementa um método que realiza uma consulta SQL responsável por buscar esses dados e retorná-los em formato de *string* (o que facilita a análise). Essa etapa de processamento da entrada ocorre apenas na máquina responsável por manter o banco de dados e responsável por iniciar o nó GridGain que exercerá o papel de nó mestre, nó encarregado de designar as tarefas *map* e *reduce* para os demais nós, chamados trabalhadores. Estes nós, correspondem a instâncias GridGain, demonstrado na Figura A.2 do Apêndice A.

Uma vez que os nós GridGain tenham sido configurados e iniciados (como pode ser visto no Apêndice A), a computação realizada pela aplicação no ambiente de desenvolvimento Eclipse será automaticamente distribuída entre os nós GridGain.

A aplicação inicia uma instância do GridGain passando um arquivo XML de configuração especificando detalhes como: o endereço do host, método de carregamento das classes, método de serialização de objetos, método de descoberta de host na topologia, entre outras. Em seguida passa a *string* de entrada para computação pelo método que implementa a tarefa MapReduce para análise de sentimento.

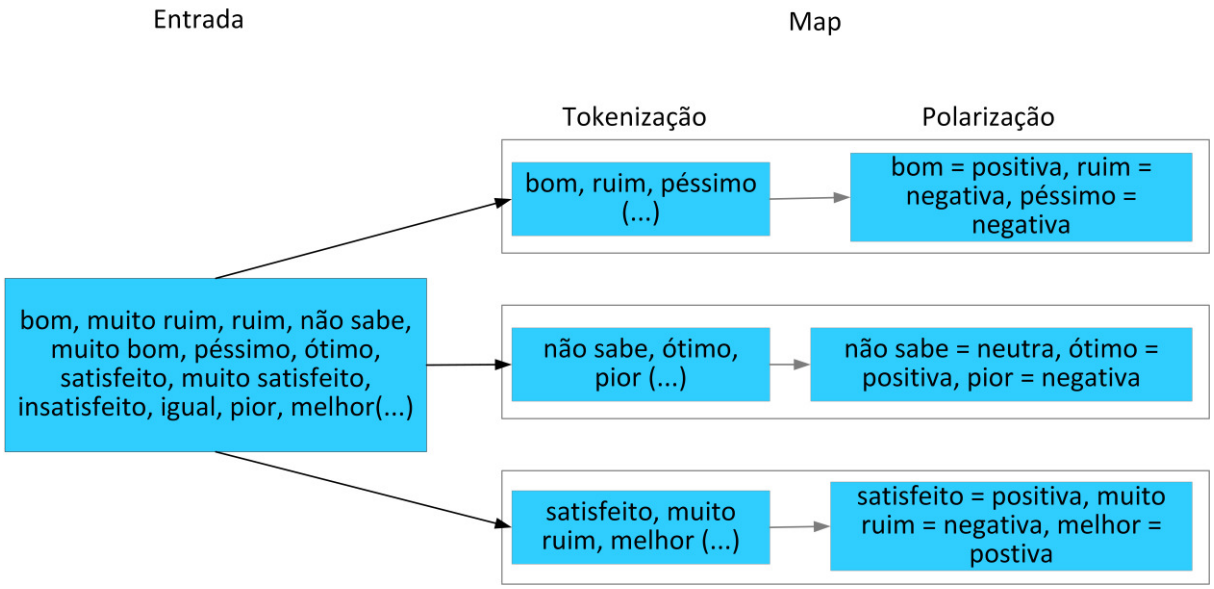


Figura 3.9: Processamento da entrada na tarefa *map*.

Durante a tarefa *map* ocorre a tokenização e a polarização, como pode ser acompanhado na figura 3.9. O processamento da entrada é realizado pela tarefa de tokenização. Essa tarefa tokeniza a *string* recebida (que contém todas as palavras de sentimento da pesquisa de opinião) em palavras individuais (ou *tokens*), cria um *job* filho para cada uma dessa palavras e envia esses *jobs* para os outros nós GridGain para processamento.

A tarefa de polarização consiste da aplicação do método baseado em léxico. Esse método (como visto no tópico 2.3.5) compara a palavra recebida da tarefa de tokenização com dois dicionários de sentimento, que mantém todas as palavras expressam os sentimentos positivos e negativos e classifica o sentimento expresso pela palavra como uma das duas polaridades. A metodologia deste trabalho, como outros trabalhos que utilizam o método baseado em léxico, descarta palavras que contenham sentimento neutro (como: igual, regular, etc.), de maneira que essa palavra não seja polarizada.

O dicionário de palavras positivas para os dados de pesquisa de opinião realizada em 1994 pelo instituto IBOPE inclui as seguintes as palavras ou expressões: satisfeito, muito satisfeito, melhorar, melhorar muito, melhorar um pouco, boa; as palavras que constituem o dicionário de palavras negativas são: insatisfeito, muito insatisfeito, piorar, péssima. As palavras consideradas neutras e que a na metodologia aqui proposta são ignoradas, constituem o seguinte dicionário: não sabe, ficar como está, ficar na mesma, regular.

Cada *job* ao realizar a tarefa de polarização, exibe uma saída na instância de nó GridGain, no qual a palavra foi recebida para processamento. A Figura 3.10 mostra a saída de processamento para as palavras processadas da pesquisa de opinião realizada em 1994.

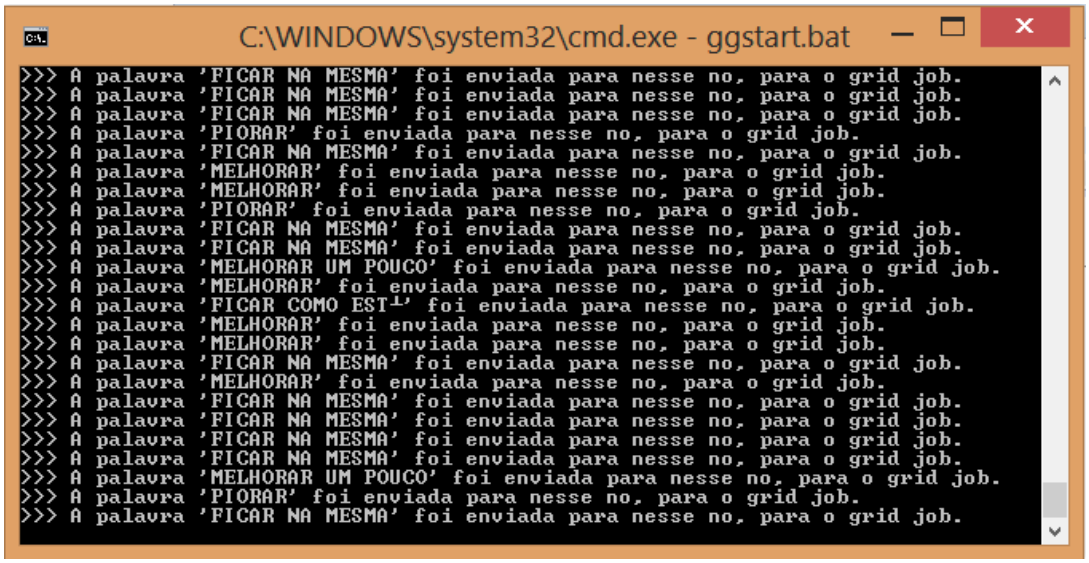


Figura 3.10: Saída de processamento em instância GridGain em ambiente Windows.

Cada nó (ou instância) GridGain retorna um *Map* contendo o resultado do processamento do *job*. Esse *Map* é encaminhado para outro nó para etapa de *reduce*. Na etapa *reduce* (exibida na Figura 3.11), os resultados são agrupados em conjuntos de saída correspondente ao processamento naquele nó GridGain, conforme a polaridade da palavra determinada pela análise de sentimento.

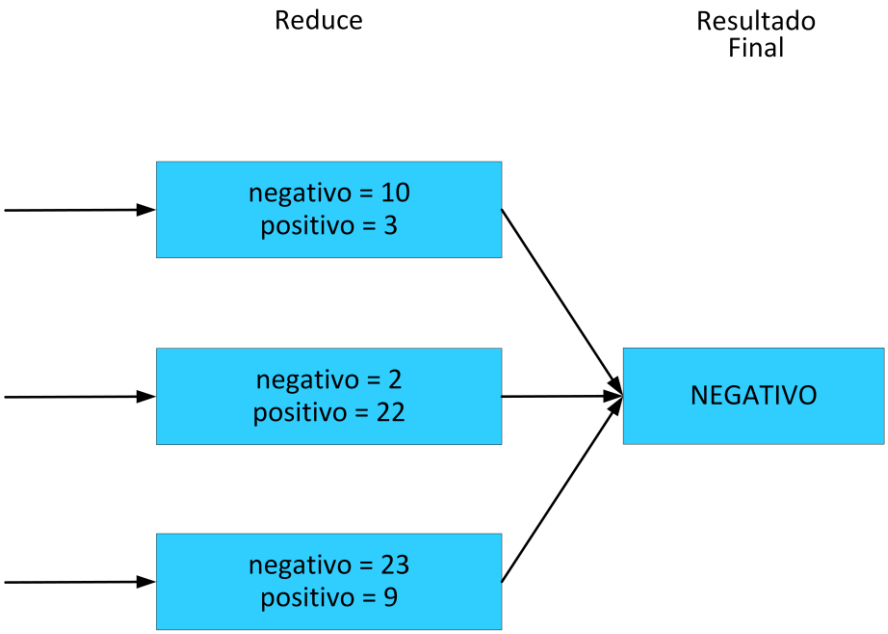


Figura 3.11: Tarefa *reduce* agrupando os conjuntos de saída com os totais de polaridades.

O sentimento total da pesquisa é, então definido, somando a quantidade de palavras positivas e negativas determinada pela análise de sentimento, conforme a método baseado em léxico propõe. O resultado final contendo o sentimento total da pesquisa é retornado ao nó mestre, no caso, o nó GridGain iniciado no ambiente de desenvolvimento Eclipse, que exibe o resultado da análise determinando o sentimento predominante da pesquisa de opinião na saída do console da máquina virtual Java do ambiente, como mostrado na Figura 3.12.

Para os dados da pesquisa de opinião realizada no ano de 1994 analisados, o resultado final atribuído como sentimento total da pesquisa de opinião foi negativo, como pode ser observado em destaque na Figura 3.12 a seguinte saída no console do Eclipse:

```
>>> O Sentimento Total da pesquisa é: NEGATIVO
```

Uma avaliação das razões que conduziram a esse resultado para o ano de 1994 e para cada um dos anos analisados por meio da metodologia são discutidas no tópico seguinte.

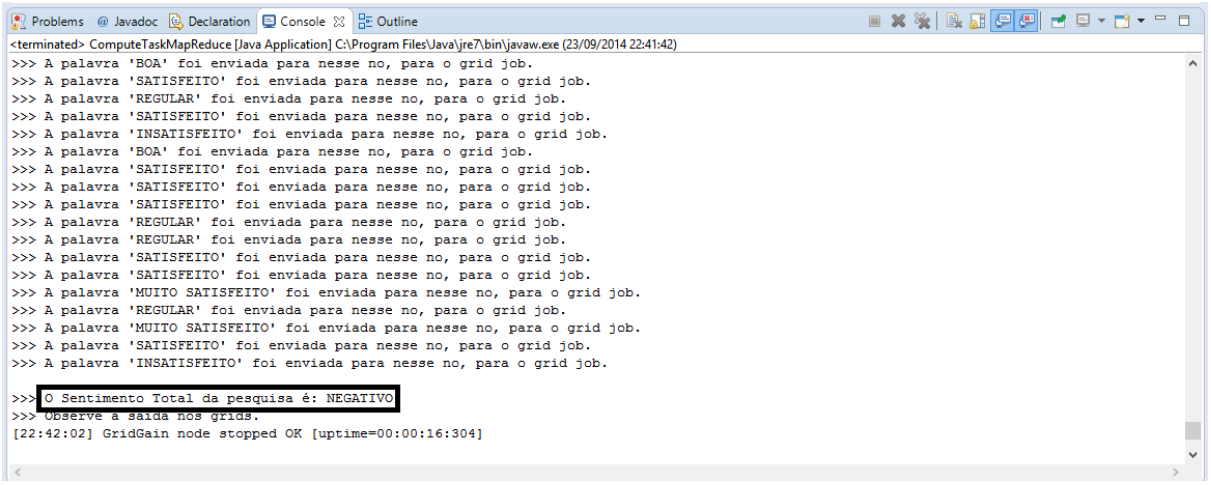


Figura 3.12: Resultado da análise de sentimento no console do Eclipse.

No momento em que o processamento é concluído, a máquina virtual Java do ambiente Eclipse encerra sua execução, ou seja, é encerrado também a instância do GridGain executando naquela máquina, causando a retirada desse nó na topologia de nós GridGain.

3.2.5 Avaliação de Resultados

Para avaliar os resultados produzidos, foram realizadas análises de sentimento sobre o conjunto de respostas de perguntas das pesquisas de opinião. As pesquisas de opinião disponíveis por meio da base, como observado anteriormente, tratam de opinião pública sobre fatores econômicos e sociais, além de opinião sobre assuntos políticos administrativos brasileiros. Portanto, a melhor maneira de avaliar os resultados obtidos é levar em consideração o contexto histórico no momento em que a pesquisa de opinião foi realizada.

A metodologia proposta foi demonstrada expendida nos tópicos anteriores deste capítulo e resultou, como visto anteriormente em sentimento negativo para a pesquisa realizada no ano de 1994. A mesma metodologia foi aplicada nos demais dados de pesquisa de opinião, disponíveis por meio da base para outros anos produzindo o gráfico da Figura 3.13.

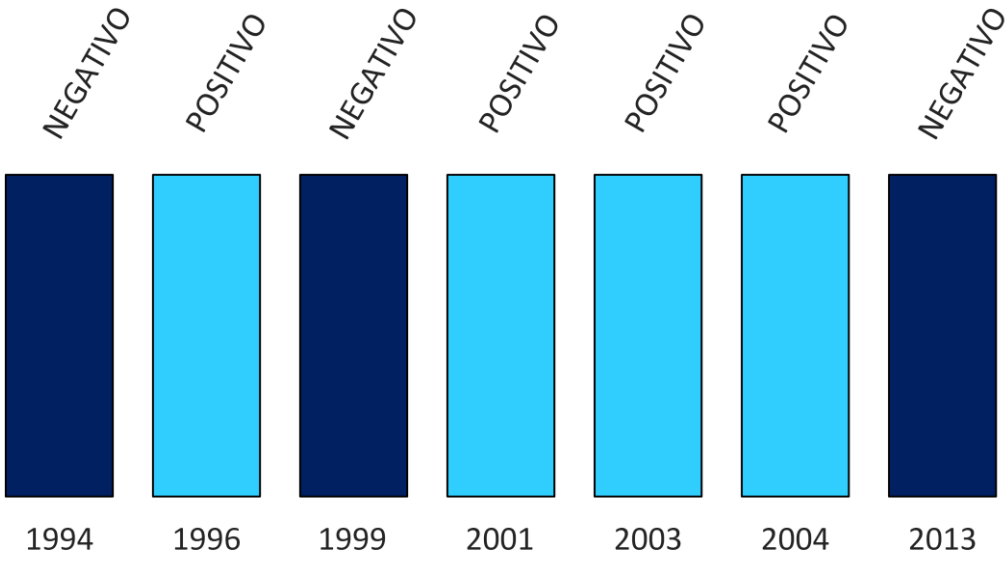


Figura 3.13: Resultados da análise de sentimento ano a ano.

A coleta de informações feita pela pesquisa de opinião realizada em 1994 aconteceu durante o mês de julho. Naquele ano, este foi um mês de grandes mudanças econômicas com a criação de uma nova moeda corrente no Brasil (MORAES, 2010). Como pode ser visto na Figura 3.13, o sentimento negativo foi predominante nas respostas da pesquisa de opinião, o que deixa claro a preocupação do brasileiro com a nova conjuntura do cenário econômico e sua insatisfação com a vida levada até aquele momento – a primeira pergunta de todos os questionários busca saber a satisfação do entrevistado, como pode ser visto no tópico 3.2.2 por meio da Figura 3.6.

Para o ano de 1996, pode-se perceber pelo resultado da análise de sentimento na Figura 3.13 que o sentimento total expressado na pesquisa de opinião para aquele ano foi positivo. Aqui novamente, a economia foi um quesito determinante para o sentimento apurado na pesquisa de opinião no ano de 1996. O cenário econômico brasileiro se encontrava favorável, conforme (MORAES, 2010), o que tornou a expectativa para aquele ano positiva. Isso refletiu diretamente na pesquisa de opinião, conduzindo a um resultado com sentimento positivo.

O sentimento negativo apurado (gráfico da Figura 3.13) para o ano de 1999 é reflexo da preocupação da população com áreas como saúde, educação e segurança que apresentavam índices muito baixo em comparação a períodos anteriores. A situação econômica se estabilizava, mas nesse ano, segundo (MORAES, 2010), a desvalorização do real chegou a 8,9% devido a um calote sofrido pelo governo. Essa situação fez com que o governo adotasse o câmbio flutuante, levando a população a se preocupar com a volta da superinflação.

Com a chegada do século 21, as pessoas se sentiam mais confiantes em relação as suas vidas (MORAES, 2010). Essa alta na confiança da população, aliada a taxa de juros em torno de 16,75%[23], fez com a análise de sentimento realizada sobre os dados da pesquisa de opinião, realizada em março de 2001 apresentasse um sentimento positivo (como pode ser visto no gráfico da Figura 3.13). Outros fatores, como a satisfação com a administração federal, com um prognostico de mudança nas políticas públicas e satisfação pessoal dos entrevistados, influenciaram o resultado da pesquisa.

O resultado de sentimento total positivo no ano de 2003 é resultado da confiança da população ante a mudança de cenário econômico e social promovida pela nova administração federal (MORAES, 2010). Essa confiança é confirmada novamente pela pesquisa realizada no ano de 2004, durante o mês de novembro, em que se manteve um sentimento total positivo em relação à pesquisa realizada no ano de 2001.

A pesquisa feita durante o mês de junho de 2013 obteve sentimento total negativo como, pode ser visto no gráfico da Figura 3.13. Esse resultado é efeito das manifestações ocorridas durante esse período. A população, revoltava com um conjunto de fatores sociais (saúde, educação, segurança), qualidade de serviços públicos prestados e corrupção levou à rua multidões de pessoas em todo país para protestar por uma melhora do quadro geral brasileiro naquele ano (GRIPP, 2013).

Como foi possível verificar, a metodologia proposta, proporciona uma nova perspectiva de análise para dados de pesquisa de opinião. Em geral, dados de pesquisa de opinião são analisados de um ponto de vista estatístico, fornecendo base para análises de cunho estatístico descritivo (média, mediana, máximo, desvio padrão, amplitude), análise de regressão, análise de fator (CHEN et al., 2014). A análise de sentimento sobre esses dados entrega uma nova visão, pois permite avaliar os dados e relacioná-los com eventos (normalmente sentimentos são influenciados por eventos, e isso é refletido nas opiniões expressas), possibilitando que as instituições utilizem esse método para alcançar um ponto de vista acerca do sentimento implícito nas respostas, útil por exemplo, para avaliação de produtos, serviços oferecidos e satisfação. No processo de tomada de decisão, a autoavaliação importante para sucesso do negócio (OLIVEIRA, 2013).

Big data entre outros objetivos, destaca-se por auxiliar no processo de tomada de decisão, pois as tecnologias big data permitem a descoberta de *insights* e até mesmo predição

[23] http://www.bcb.gov.br/?COPOMJUROS

de cenários por meio da análise proposta (CHEN et al., 2014). A metodologia proposta por este trabalho pode ser aplicada através de um sistema – tanto por instituições públicas como privadas – para buscar opinião ou sentimento em um volume maciço de informações com grande capacidade de processamento e baixo custo de investimento e manutenção, permitindo a estas instituições a capacidade de evolução, melhoria e avaliação de produtos ou serviços.

3.2.6 Comparação com Trabalhos Relacionados

Dos trabalhos relacionados apresentados no Capítulo 1, os métodos apresentados em (PANG et al., 2002; TURNEY, 2002) se baseiam no mesmo tipo de análise proposta pela metodologia deste trabalho (análise de sentimento em nível de documento baseado em léxico). Contudo, a análise de sentimento proposta neste trabalho, é importante se comparada à (PANG et al., 2002; TURNEY, 2002) porque contempla o processamento de grandes volumes de informações, que é uma característica primária de big data, justamente o tipo de conjuntos de dados alvo desta metodologia.

As metodologias propostas por (PANG et al., 2002; TURNEY, 2002) têm que passar por uma adaptação no método utilizado em ambas, pois não estão preparadas para comportar o processamento de grandes volumes de informações. Enquanto a metodologia proposta lida com esse situação de forma nativa pois utiliza um *framework* (GridGain) projetado especificamente para trabalhar com o tipo de processamento requerido por big data (em alta escala e de forma distribuída), permitindo ainda a utilização de uma estrutura existente para realizar o paralelismo e o processamento distribuído dos dados, o que garante que análise big data proposta seja simples, intuitiva e viável economicamente para instituições como por exemplo universidades e órgãos públicos.

Os métodos propostos em (HU e LIU, 2004; KIM e HOVY, 2004, 2007; KIM e HOVY, 2006) alcançam os mesmos resultados que a metodologia apresentada neste trabalho, contudo, se diferenciam entre si e em relação a este trabalho no nível de análise proposto (LIU, 2012), de forma que a análise proposta por eles realiza a classificação de sentimento (neste trabalho chamada de polarização) em nível de sentença (HU e LIU, 2004; KIM e HOVY, 2004, 2007), enquanto este trabalho propõe uma classificação de sentimento em nível de documento. A classificação de sentimento em nível de sentença, utilizada nesses métodos lhe confere uma capacidade de análise mais refinada que a método proposto por este trabalho, contudo, nenhum destes métodos realiza o processamento em paralelo dos dados analisados, demandando ainda mais os recursos utilizados para realizar análise, visto que, por ser mais refinada, necessita de

mais capacidade de processamento. E como cresce a tendência de que cada vez mais volume de informação necessita ser processado, um método que análise esse grande volume de informações, garante a entrega de informação útil, ainda que lide com grande demanda de dados.

No caso do método proposto em (KIM e HOVY, 2006), no qual uma técnica de aprendizado supervisionado se propõe a identificar um tipo de opinião específico, se distancia da proposta deste trabalho, pois o método aqui proposto não busca identificar tipos de opinião expressas, mas sim a classificar em positivo e negativo os dados de pesquisa de opinião analisados.

Por fim, esse capítulo demostrou detalhadamente o método proposto para análise de sentimento de dados opinião com características big data. Foram apresentados os *softwares* e *hardware* utilizados durante o desenvolvimento da metodologia. Em seguida, foi explicado como preparar o ambiente, seja este, Windows ou Linux, para execução do GridGain e permitir o processamento em paralelo dos dados da pesquisa de opinião. Cada uma das quatro primeiras etapas do método (aquisição da base de dados, adaptação da base, armazenamento dos dados e análise de sentimento) foi demonstrada utilizando os dados de pesquisa de opinião apurados durante o ano de 1994 e a etapa final, de avaliação de resultados, discute os resultados da análise de sentimento para os anos analisados, disponíveis por meio da base. E, por fim, é feito uma comparação do método desenvolvido neste trabalho com trabalhos relacionados. Dessa forma, é possível compreender como a método proposto se diferencia de outros trabalhos relacionados.

No próximo capítulo serão apresentadas algumas conclusões acerca da pesquisa desenvolvida nesta monografia, bem como apresentados sugestões de trabalhos futuros.

4 Conclusão

Na presente monografia foi proposto um método para análise de sentimento em nível de documento. O método apresentado é baseado em léxico e permite determinar a polaridade da pesquisa de opinião (por meio da análise das respostas dos questionários das pesquisas) em positivo ou negativo como um todo. E foi desenvolvido para realizar esta análise em conjuntos de dados com características de big data de forma nativa, por meio do processamento em paralelo e distribuído dos dados analisados.

O objetivo principal deste trabalho consistia em realizar uma análise diferente da tradicional mineração de dados estatísticos, comum em pesquisas de opinião, e demonstrar como as tecnologias big data podem otimizar o processamento de grandes volumes de dados. A metodologia alcança esse objetivo, pois realiza uma análise que determina sentimento a partir de dados de pesquisa de opinião, permitindo abordar os dados sobre outra perspectiva de análise, a dos sentimentos expressos pelas respostas dos entrevistados. De forma nativa, a metodologia garante a capacidade de processamento de um grande volume de informações, isso é possível pela utilização do *framework* GridGain e do modelo de processamento MapReduce para processamento em paralelo distribuído.

O desenvolvimento de uma metodologia, para análise big data, que permita a utilização de uma estrutura existente (um dos objetivos desse trabalho) implica em lidar com certos problemas, tais como: a escolha de uma ferramenta que permita a otimização de processamento de grande volume de informações e se adapte a diferentes ambientes ou estrutura de hardware disponível. Nesse quesito, o GridGain foi uma escolha vantajosa, se comparado às outras ferramentas disponíveis, discutidas no capítulo 2, pois permite que o processamento seja realizado de forma paralela em mais de uma máquina utilizando o hardware disponível, sem que haja prejuízo à realização da análise e manteve a simplicidade no desenvolvimento da lógica utilizada na análise de sentimento.

A metodologia proposta possui limitações. As principais estão relacionadas com o nível da análise de sentimento realizada. A análise em nível de documento não propicia a detecção de aspectos mais refinados acerca da opinião expressada (por exemplo, múltiplas opiniões em uma mesma sentença ou sarcasmo). Outra limitação é a impossibilidade de analisar um tópico em particular utilizando mais de uma fonte de dados de forma que a análise realizada possa, por exemplo, comparar opiniões em busca de tendências e perceber como o sentimento em relação ao tópico varia conforme a fonte de dados.

Em trabalhos futuros, pretende-se expandir a metodologia para que esta permita o processamento de informações de múltiplas fontes (*web*, banco de dados, etc.), refinar a análise para que esta ocorra em nível de sentença ao invés de documento, podendo, assim, permitir uma análise de sentimento compartimentalizada, o que é interessante para detecção de tendências de sentimento durante o processo de análise.

Referências Bibliográficas

Apache Hadoop (2008). Documentação. Disponível em: <http://hadoop.apache.org/core>. Acessado em 12/06/2014.

ANNETT, M.; KONDRAK G. **A Comparison of Sentiment Analysis Techniques: Polarizing Movie Blogs**. In *Canadian Conference on AI*. pp. 25–35. 2008.

ARCHAK, Nikolay et al. **Show me the Money!: Deriving the Pricing Power of Product Features by Mining Consumer Reviews**. *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD-2007)*. 2007.

BAAH, G.K.; GRAY, A.; HARROLD, M.J. **On-Line Anomaly Detection of Deployed Software: A Statistical Machine Learning Approach**. *Proceedings of the 3rd International Workshop on Software Quality Assurance*. ACM, pp. 70–77. 2006.

BEINEKE, Philip. et al. **An Exploration of Sentiment Summarization**. *Proceedings of AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications*. 2003.

BESPALOV, Dmitriy. et al. **Sentiment Classification Based on Supervised Latent n-gram Analysis**. *The 20th ACM Conference on Information and Knowledge Management*. *ACM*, *pp 277–286*. 2011.

BEYER, Mark; LANEY, Douglas. **The Importance of 'Big Data': A Definition**. Gartner. 2012.

BLAIR-GOLDENSOHN, S. et al. **Building a Sentiment Summarizer for Local Service Reviews**. *Proceedings of WWW-2008 Workshop on NLP in the Information Explosion Era*. 2008.

BLAND, Martin. et al. **An Introduction to Medical Statistics**. Oxford University Press, 2000.

CESOP. **Centro de Estudos de Opinião Pública**. UNICAMP. Campinas. Disponível em: <http://www.cesop.unicamp.br/site/htm/apre.php>. Acessado em 10 jun. 2014.

CHEN, M. et al. **Big Data: Related Technologies, Challenges and Future Prospects**. Briefs in Computer Science. Springer International Publishing. 2014.

CHEN, Y.; XIE, J. **Online Consumer Review: Word-Of-Mouth As A New Element Of Marketing Communication Mix**. Management Science. 2008. pp. 477–491. Disponível

em: <http://pubsonline.informs.org/doi/pdf/10.1287/mnsc.1070.0810>. Acessado em 20 jun. 2014.

CHINNICI, G.; D'AMICO, M.; PECORINO, B. **A Multivariate Statistical Analysis on the Consumers Of Organic Products**. British Food Journal, v. 104, n. 3/4/5, p. 187-199. 2002.

Cisco Visual Networking Index. **Global mobile data traffic forecast update, 2012–2017**. Disponível em: <http://www.cisco.com/en.US/ solutions/collateral/ns341/ns525/ns537/ns705/ns827/white paper c11-520862.html>. Acessado em 22 jun. 2014.

CORRÊA, Marcello. Uso de Big Data ajuda governo brasileiro a gastar de forma mais eficiente. **O GLOBO**. Rio de Janeiro, 03 mar. 2013. Disponível em: <http://oglobo.globo.com/sociedade/tecnologia/uso-de-big-data-ajuda-governo-brasileiro-gastar-de-forma-mais-eficiente-8582240>. Acessado em: 02 jun. 2014.

DAS, Sanjiv; CHEN, Mike. **Yahoo! for Amazon: Extracting Market Sentiment From Stock Message Boards**. *Proceedings of APFA-2001*. 2001.

DAS, Sanjiv; CHEN, Mike. **Yahoo! For Amazon: Sentiment Extraction From Small Talk on the Web**. Management Science, pp. 1375–1388. 2007. Disponível em: <http://pubsonline.informs.org/doi/pdf/10.1287/mnsc.1070.0704>. Acessado em 18 jun. 2014.

DAVE, Kushal. et al. **Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews**. *Proceedings of international conference on World Wide Web (WWW-2003)*. 2003.

DEAN, J.; GHEMAWAT, S. **MapReduce: Simplified Data Processing on Large Clusters**. *OSDI*. 2004. Disponível em: <http://research.google.com/archive/mapreduce-osdi04.pdf>. Acessado em 16 jun. 2014.

DELLAROCAS, C. et al. **Exploring the Value of Online Product Reviews in Forecasting Sales: The Case of Motion Pictures**. Journal of Interactive Marketing. pp. 23–45. 2007. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1094996807700361>. Acessado em 20 jun. 2014.

GREENE, Stephan; RESNIK, Philip. **More Than Words: Syntactic Packaging and Implicit Sentiment**. *Proceedings of Human Language Technologies*: The 2009 *Annual Conference of the North American Chapter of the ACL (NAACL-2009)*. 2009.

GridGain (2011). Publicado na Web. Disponível em: <http://www.gridgain.org>. Acessado em 17 jun. 2014.

GRIPP, Alan. Retrospectiva: Manifestações não foram pelos 20 centavos. **FOLHA DE S. PAULO**. São Paulo, 27 dez. 2013. Disponível em: <http://www1.folha.uol.com.br/poder/2013/12/1390207-manifestacoes-nao-foram-pelos-20-centavos.shtml>. Acessado em: 03 jun. 2014.

HAUSENBLAS, Michael; NADEAU, Jacques. **Apache Drill: Interactive Ad-Hoc Analysis at Scale**. 2012. Disponível em: <https://www.mapr.com/sites/default/files/apache_drill_interactive_ad-hoc_query_at_scale-hausenblas_nadeau1.pdf>. Acessado em 09/06/2014.

Hortonworks (2011). Disponível em: <http://docs.hortonworks.com/>. Acessado em 12 jun. 2014.

HU, Minqing; LIU, Bing. **Mining and Summarizing Customer Reviews**. *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2004)*. 2004. Disponível em: <http://www.cs.uic.edu/~liub/publications/kdd04-revSummary.pdf>. Acessado em 20 jun. 2014.

IDC. GANTZ, J.; REINSEL, D. **Extracting Value From Chaos**. IDC iView, pp 1–12. 2011. Disponível em: <http://www.emc.com/collateral/analyst-reports/idc-extracting-value-from-chaos-ar.pdf>. Acessado em 12 jun. 2014.

JINDAL, Nitin; LIU, Bing (2006a). **Identifying Comparative Sentences in Text Documents**. *Proceedings of ACM Sigir Conf. on Research and Development in Information Retrieval (SIGIR-2006)*. Disponível em: <http://www.cs.uic.edu/~liub/publications/sigir06-comp.pdf>. Acessado em 20 jun. 2014.

JINDAL, Nitin; LIU, Bing (2006b). **Mining Comparative Sentences and Relations**. *Proceedings of National Conf. on Artificial Intelligence (AAAI-2006)*.

KIM, Soo-Min; HOVY, Eduard. **Determining the sentiment of opinions**. *Proceedings of interntional conference on computational Linguistics (COLING-2004)*. 2004.

KIM, Soo-Min; HOVY, Eduard. **Extracting opinions, opinion holders, and topics expressed in online news media text**. *Proceedings of the conference on empirical methods in natural Language Processing (EMNLP-2006)*. 2006.

KIM, Soo-Min; HOVY, Eduard. **Crystal: analyzing predictive opinions on the web**. *Proceedings of the Joint conference on empirical methods in natural Language Processing and computational natural Language Learning (EMNLP/CoNLL-2007)*. 2007.

KU, Lun-Wei. et al. **Opinion Extraction, Summarization and Tracking in News and Blog Corpora**. *Proceedings of AAAI-CAAW'06*. 2006.

LAMMEL, R. **Google's Mapreduce Programming Model – Revisited**. Science of Computer Programming, 70(1):1 – 30. 2008.

LIU, Bing. **Sentiment Analysis and Subjectivity, Handbook Of Natural Language Processing.** Second Edition. N. Indurkhya e F.J. Damerau, Eds. 2010.

LIU, Bing. **Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data**. Second Edition. Springer International Publishing. 2011.

LIU, Bing. **Sentiment Analysis and Opinion Mining, in Synthesis Lectures on Human Language Technologies**. Morgan e Claypool Publishers. 2012.

MARZ, Nathan. **Preview of Storm: The Hadoop of Realtime Processing - BackType Technology**. 2011. Disponível em: <http://www.memonic.com/user/pneff/folder/queue/id/1qSgf>. Acessado em 15 jun. 2014.

MAYER-SCHÖNBERGER, V.; CUKIER, K. **Big Data: A Revolution that Will Transform How We Live, Work, and Think**. Eamon Dolan/Houghton Mifflin Harcourt. 2013.

McKinsey Global Institute (2011). **Big Data: The Next Frontier for Innovation, Competition, and Productivity**. McKinsey Global Institute. Disponível em: <http://www.mckinsey.com/insights/business_technology/big_data_the_next_frontier_for_innovation>. Acessado em 10 jun. 2014.

MOENG, M.; MELHEM, R. **Applying Statistical Machine Learning to Multicore Voltage & Frequency Scaling**. *Proceedings of the 7th ACM international conference on computing frontiers. ACM, pp 277–286*. 2010.

MORAES, J. G. V. **História. Geral e Brasil**. Volume 3. 1. Ed. São Paulo: Saraiva, 2010. v. 3. 304 p.

MORINAGA, S. et al. **Mining Product Reputations on the Web**. *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery And Data Mining (KDD-2002).* 2002. Disponível em: <http://goo.gl/m7ZS6t>. Acessado em 10 jun. 2014

NASUKAWA, T.; YI, J. **Sentiment Analysis: Capturing Favorability Using Natural Language Processing**. *Proceedings of the K-CAP-03*, *2nd International Conference on Knowledge Capture*. 2003.

OLIVEIRA, D. Big Data: O desafio de garimpar informações. **COMPUTERWORLD**, São Paulo, Ano XIX, n. 554, fev./mar. p. 12-17. 2013.

OLSTON et al. **Pig Latin: A not-so-foreign Language for Data Processing**. *SIGMOD*. 2008. Disponível em: <http://infolab.stanford.edu/~usriv/papers/pig-latin.pdf>. Acessado em 15 jun. 2014.

PAL, S.K.; TALWAR, V.; MITRA, P. **Web Mining in Soft Computing Framework, Relevance, State of the Art and Future Directions**. *Proceedings of IEEE Transac Neural Netw* 13(5):1163–1177. 2002.

PANG, Bo; LEE, Lillian.**Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval**. Vol. 2. pp. 1–135. 2008.

PANG et al. **Thumbs up?: Sentiment Classification Using Machine Learning Techniques**. *Proceedings Of Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)*. 2002.

PHILIP et al. **Top 10 Algorithms in Data Mining**. Knowl INF SYST 14(1):1–37. 2008.

SERGEY et al. **Dremel: Interactive Analysis of Web-Scale Datasets**. *OSDI*. 2010. Disponível em: <http://research.google.com/pubs/pub36632.html>. Acessado em 12 jun. 2014.

SEKI et al. **Opinion-focused Summarization and its Analysis at DUC 2006**. *Proceedings of the Document Understanding Conference (DUC)*. 2006.

STORM (2011). Disponível em: <http://storm.incubator.apache.org/>. Acessado em 12 jun. 2014.

TONG, Richard M. **An Operational System for Detecting and Tracking Opinions in on-line Discussion**. *Proceedings of SIGIR Workshop on Operational Text Classification*. 2001.

TURNEY, Peter D. **Thumbs up or Thumbs Down?: Semantic Orientation Applied to Unsupervised Classification of Reviews**. *Proceedings of Annual Meeting of the Association For Computational Linguistics (ACL-2002).* 2002.

WARD, Jonathan Stuart; BARKER, Adam. **Undefined By Data: A Survey of Big Data Definitions**. 2013. 2 f. University of St Andrews, Reino Unido. 2013.

WIEBE, Janyce. **Learning Subjective Adjectives From Corpora**. *Proceedings of National Conf. on Artificial Intelligence (AAAI-2000).* 2000. Disponível em: <http://www.cs.columbia.edu/~vh/courses/LexicalSemantics/Orientation/wiebe-aaai2000.pdf>. Acessado em 12 jun. 2014.

WIEBE, Janyce. et al. **Development and use of a Gold-Standard Data set for Subjectivity Classifications**. *Proceedings of the Association for Computational Linguistics (ACL-1999).* 1999.

WIKI Apache (2014). **Hadoop User's**. Disponível em: <http://wiki.apache.org/hadoop/PoweredBy>. Acessado em 11 jun. 2014.

ZHANG, Lei; LIU, Bing (2011b). **Identifying Noun Product Features That Imply Opinions**. *Proceedings of the Association for Computational Linguistics (short paper) (ACL-2011).*

Apache ZooKeeper (2008). Disponível em: <http://zookeeper.apache.org/>. Acessado em 15 jun. 2014.

Apêndice A

Configuração de ambiente para o GridGain

O GridGain é um *middleware open source* baseada em *Java Virtual Machine*. Dentre os principais requisitos da versão atual (6.2.0), no momento da elaboração deste trabalho, o principal é a instalação do Java SE *Development Kit* (JDK) 1.7 ou posterior.

A instalação e configuração do GridGain é a mesma independente do ambiente (Windows, Linux ou Mac OS). Após a instalação do Java JDK, disponível no site da SUN[24], o passo seguinte é extrair o conteúdo do arquivo zip "gridgain-platform-os-x.x.x-aaa" –obtido no site do GridGain –onde "x.x.x" indica a versão do mesmo e "aaa" indica o ambiente de execução ("win" para Windows e "nix" para Linux ou Mac OS).

Para que o GridGain funcione adequadamente, é preciso configurar a sua variável de ambiente. No Windows, essa configuração é feita por meio das "configurações avançadas do sistema", como pode ser visto na Figura 3.1. Por meio dessa configuração, cria-se uma variável de ambiente chamada GRIDGAIN_HOME, que aponta para o diretório escolhido para instalação do GridGain na máquina.

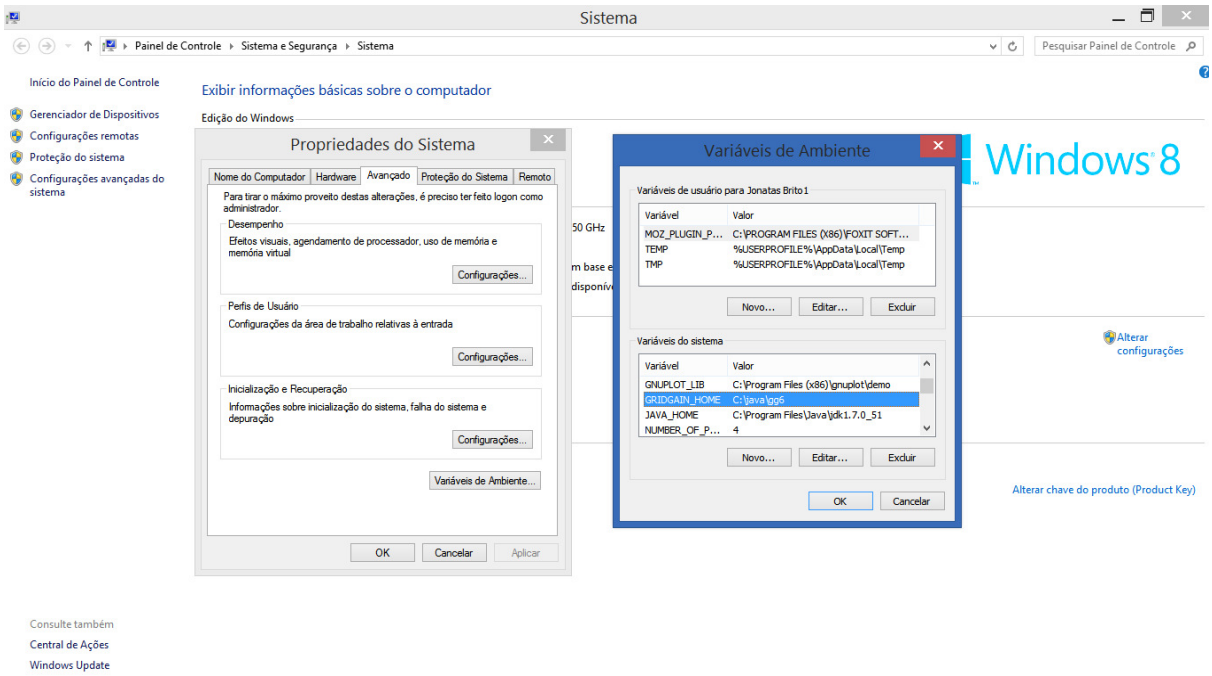


Figura A.1: Configuração da variável de ambiente no Windows.

[24] http://www.oracle.com/technetwork/pt/java/javase/downloads/

No Linux a configuração da variável de ambiente é feita por linha de comando no terminal. Para isso é utilizado o comando:

export GRIDGAIN_HOME=/../gridgain

onde "/../gridgain" é o diretório que foi criado para manter a instalação. Para finalizar a configuração, é preciso editar os arquivos "/etc/bashcr" e "/etc/profile" e acrescentar uma nova linha no fim do arquivo com o mesmo conteúdo do comando utilizado terminal.

Após as configurações anteriormente apresentadas, o ambiente se encontra configurado para executar um ou mais nós[25]. Para execução de um nó no ambiente (Windows, Linux) é preciso por acessar o diretório "../bin" por meio do *prompt* de comando no Windows e do terminal no Linux e executar o seguinte comando:

ggstart.bat, para Windows

ggstart.sh, para Linux

A Figura 3.2 exibe o *script* de um nó do GridGain iniciando em ambiente Windows. Ao iniciar um nó GridGain disponibiliza uma série de informações, como a versão do GridGain instalada, a data de lançamento da versão, as informações de log, as informações acerca daquela instância (identificador e memória alocada) e uma visão da topologia (quantidades de nós na topologia, ordem do nó na topologia).

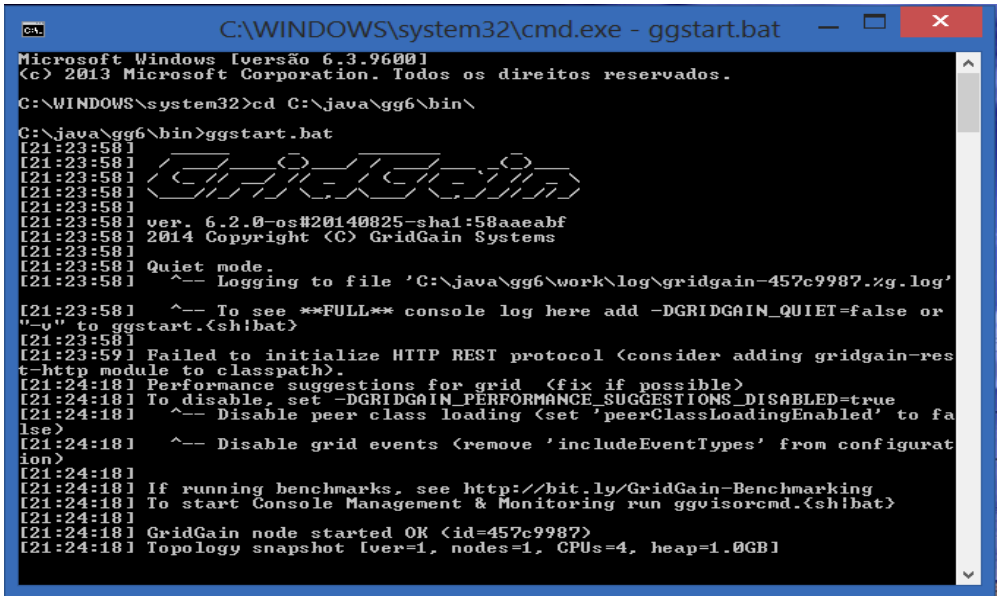


Figura A.2: *Script* de um nó do GridGain iniciando em ambiente Windows.

[25] Cada nó, corresponde a uma instância independente do GridGain.

Uma vez que o ambiente do sistema foi configurado seguindo os passos apresentados anteriormente, torna-se possível, que a computação definida na aplicação desenvolvida no Eclipse seja automaticamente distribuída entre os nós GridGain toda vez que aplicação é executada (processo também chamado de paralelização de processamento) por meio do modelo GridGain MapReduce.

Como discutido no tópico 2.2.6, o GridGain possui o recurso de descoberta automática de nós da topologia, quer seja em nós numa máquina local ou em máquinas conectadas em rede. Isso garante que o processamento seja dividido entre a estrutura de nós disponível. Outro recurso, o de balanceamento automático de carga, impede que um nó acabe tendo mais demanda de processamento que os demais nós da estrutura.

Em (GridGain, 2011) é observado que a configuração do ambiente de sistema deve ser realizada de maneira consistente para evitar a sobrecarga de memória e permitir a execução simples e segura do GridGain.