Utilização de Big Data em Portais de Conhecimento Aberto

Trabalho de Conclusão do Curso de
Tecnologia em Sistemas para Internet

Marcos Vinícius Saturno Ribeiro
Orientador: Andre Peres

[1]Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul (IFRS)
Campus Porto Alegre
Av Cel Vicente, 281, Porto Alegre – RS – Brasil

andre.peres@poa.ifrs.edu.br, mv_saturno@yahoo.com.br

Resumo. *O presente artigo visa demonstrar o potencial de utilização de ferramentas de big data na análise de dados, através da criação de uma interface integrada a um portal de dados abertos. O objetivo é facilitar a utilização de sistemas para tratamento de dados em big bata por usuários que não possuem conhecimento técnico suficiente para executar tarefas nesta estrutura.*

1. Introdução

Na última década, pesquisas e artigos acadêmicos demonstram um vertiginoso crescimento da produção e requisição de dados, nos mais variados âmbitos. Segundo [Sagirolou and Sinanc 2013], todo o volume de dados produzido pela humanidade até o ano de 2003 (5 exabytes, ou 10^8 bytes), é reproduzido atualmente a cada dois dias. As necessidades de armazenamento e transmissão dessa enormes quantidades de dados de forma rápida, segura e confiável, somadas às necessidades de se extrair valor desses dados, produzem as condições necessárias para o surgimento de novas tecnologias de gerenciamento e manipulação de dados.

A necessidade de se trabalhar com um volume cada vez maior de dados acarreta na criação de novas ferramentas como o Hadoop [Apache 2013] e sistemas derivados, ou em outras soluções *big data*. Pesquisas acadêmicas, empresas (sejam estas de pequeno, grande ou médio porte) e até instituições governamentais baseiam suas decisões na análise de grandes conjuntos de dados relevantes às suas respectivas atividades[Weiss and Zgorski 2012].

Um exemplo prático da extensa gama de aplicações que aumentam o volume de dados existentes na internet é a utilização de sensores/atuadores, compreendida pelo termo internet das coisas (do inglês *Internet of Things - IoT*). Estimativas apresentadas em [Mohamed Ali Feki and Trappenies 2013] inferem que por volta de 2020, existirão entre 50 e 100 bilhões de ”coisas”conectadas à internet, gerando e enviando dados na Web de forma constante.

À utilização de sensores, se soma o fato de já existir atualmente diversas fontes de dados disponíveis na *web* contendo dados sensoriais, demográficos, entre outros, em vários portais de conhecimento aberto. A qualidade da água dos rios que abastecem grandes cidades, nível de poluição do ar, tendências políticas e econômicas em determinadas populações, são exemplos de dados disponíveis na internet atualmente. Desse modo, a

aplicação de soluções *big data* torna-se fundamental para possiblitar a análise e extração de informações relevantes dessa massa de dados.

O desenvolvimento e a utilização cada vez mais comum de softwares especializados para realizar operações com *big data* não necessariamente significa pôr termo à utilização de bancos de dados relacionais e Sistemas Gerenciadores de Bancos de Dados (SGBDs). Os SGBDs e ferramentas *big data* possuem objetivos diferentes e complementares.

O SGBD permanece como sendo a solução de armazenamento de dados de forma estruturada e as ferramentas de *big data* operam como uma camada superior para coletar um grande volume destes dados e realizar operações complexas de consolidação de resultados (somatórios, médias, maior valor, menor valor, etc).

O Hadoop, por exemplo, como ferramenta para tratamento de *big data*, pode utilizar dados de diversas fontes (inclusive de SGBDs), coletando estes dados e armazenando-os em arquivos sem realizar nenhum tipo de indexação por registro no ato do armazenamento. Para consultar os dados previamente armazenados, é preciso executar o procedimento de *Map Reduce* que fará uma varredura nos dados desejados e as operações necessárias para exibí-los posteriormente[Apache 2014a].

Partindo do pressuposto que tratam-se de técnicas recentes com constantes estudos e modificações de manipulação de dados, torna-se interessante a criação de uma interface que facilite o uso destes sistemas. Esta interface se encarrega da comunicação entre usuário e infraestrutura e facilita consultas e operações com conjuntos de grandes volumes de dados.

O objetivo deste trabalho é a disponibilização de uma interface de interação com o sistema Hadoop. Esta interface serve não apenas para profissionais de computação interessados em possuir um primeiro contato com sistemas *big data*, mas abre uma nova possibilidade para a utilização desta ferramenta para profissionais de outras áreas.

2. Conceito e Aplicações de *Big Data* e Portais de Dados Abertos

Neste item serão apresentadas a definição e principais características de *big data* e aplicações que utilizam esta estrutura atualmente, além de uma descrição introdutória dos conceitos de portais de dados abertos.

2.1. Conceito de *Big Data*

Devido ao constante crescimento da quantidade de dados produzidos e coletados, as estruturas atuais podem não atender mais às requisições com velocidade satisfatória, ou simplesmente os dados requisitados podem não possuir uma estrutura que corresponda às restrições de um Sistema Gerencial de Banco de Dados (SGBD), portanto faz-se necessário encontrar outras soluções para o seu armazenamento, gerenciamento e, principalmente, análise.

A evolução da tecnologia relativa ao tema em questão é constante e rápida. Esta volatilidade, intrínseca às ciências tecnológicas, também resulta na dificuldade de se estabelecer com precisão definições conceituais e terminológicas. Afinal, um conceito delimitado hoje pode ter suas fronteiras expandidas e não corresponder mais à realidade num

futuro próximo. Portanto, nas fontes consultadas a definição do termo *big data* se dá de forma relativa.

Em síntese, *big data* refere-se a conjuntos de dados cuja captura, armazenamento, gestão e análise necessitam de novas tecnologias e softwares, além dos empregados atualmente [Franks 2012]. Deste modo, a terminologia continuará válida nos próximos anos. Mesmo que as ferramentas e a quantidade ou variedade dos dados mude radicalmente, *big data* continuará sendo uma medida de grandeza além dos padrões contemporâneos.

Atualmente utiliza-se os sistemas e técnicas de *big data* para realização de consultas em bases de dados de grande porte, ou na consolidação de dados distribuídos em diversas fontes e diversos formatos. O objetivo pode ser o da realização de operações e alteração dos dados (operação mais complexa e menos usual), ou a obtenção de informações relevantes a partir deles (uso mais comum).

Big data possibilita lidar com três conceitos principais relacionados aos dados, 3 Vs: volume, velocidade e variedade [Hurwitz et al. 2013]. Volume é a escala de dados que podem ser gerenciados, alcançando atualmente a casa dos zettabytes (10^{21} bytes) ou até unidades de medida superiores. Essa enorme quantidade de dados (para os padrões contemporâneos) precisa ser armazenada de forma segura, pesquisada, acessada e analisada com velocidades satisatórias às suas respectivas aplicações.

A velocidade de processamento é obtida pela utilização de *clusters* de servidores processando tais dados de forma concomitante. Toda essa arquitetura deve ainda possiblitar a inserção de dados variados, categorizados em três tipos: estruturados (compatíveis com *data warehouses*), semi-estruturados (não compatíveis com estruturas de campos em tabelas de dados, porém possuindo alguma padronização que facilita a filtragem e análise) e dados que não possuem estrutura adequada para armazenamento e consulta em bancos de dados tradicionais, nem possuem *tags* ou outros elementos de indexação (como arquivos em formato pdf, vídeos, postagens de redes sociais, etc).

Além desses três pilares citados anteriormente, nas fontes de pesquisa consultadas encontram-se muitas referências à aplicação dessa estrutura de armazenamento e consulta dos dados. A veracidade ou precisão dos dados que possibilitam a execução de *data-mining* ou correlacionamento de dados em busca de padrões de comportamento ou repetições, permite utilizar algoritmos de inteligência artificial para prever movimentos futuros, ou sugerir produtos personalizados a clientes. Por meio dessa interpretação, obtêm-se portanto em 5Vs ao invés de 3: volume, variedade, velocidade, veracidade e valor [Hurwitz et al. 2013].

2.2. Aplicações *Big Data*

Recursos de *big data* já são utilizados e desenvolvidos atualmente por várias empresas do setor tecnológico. O Google, que possui mais de um milhão de servidores ao redor do mundo, analisa e classifica diariamente algo em torno de 7,2 bilhões de páginas, o que significa um montante de 20 petabytes de dados processados diariamente. O Youtube, um de seus serviços mais conhecidos, recebe uploads de aproximadamente 48 horas de vídeo por minuto e 4 bilhões de visitas são realizadas por dia [Hurwitz et al. 2013].

Outros exemplos de utilização de soluções *big data* não faltam. Pode-se citar *web services* como o Flickr, que em 2012 recebeu diariamente uma média de 1,42 milhão de

imagens por dia, e em 2013 tem registrado números ainda maiores [Michel 2013]. Outro exemplo é o Twitter que em 2010 anunciou a doação de sua base de dados [Twitter 2013] para a biblioteca do congresso dos Estados Unidos [Stone 2013] [Watters 2011], abrindo a possibilidade de consulta e análise não-comercial dos *tweets* de todo o mundo desde 2006. Esta possibilidade trouxe problemas para a própria biblioteca, afinal eram aproximadamente 170 trilhões de *tweets* que ocupavam algo em torno de 130 *terabytes*, além do montante diário a ser armazenado, o qual cresce numa taxa de 0,5 bilhões ao dia tornando a tarefa impraticável do ponto de vista financeiro, para a instituição [Huppke 2013].

2.3. Estrutura Base de *Big Data*

Uma das tecnologias mais conhecidas para lidar com *big data* corresponde ao algoritmo de *MapReduce* desenvolvido pelo Google[Dean and Ghemwat 2004], atualmente oferecido como uma solução de código aberto através do software Hadoop [Apache 2013]. Posteriormente, outras camadas foram adicionadas ao Hadoop, como HBase, Pig e Hive (alternativas de sintaxe de interação), visando facilitar a execução de *jobs* no Hadoop e reduzir a complexidade do desenvolvimento de algoritmos *Map Reduce*.

O Pig, por exemplo, constitui-se numa camada (*layer)* que roda sobre o Hadoop, possibilitando a execução de tarefas com fluxo de dados através de um *script* simples que roda em lotes (ou *batches)*. A interação entre o Pig e o Hadoop é realizada através de uma linguagem específica para o Pig, denominada Pig Latin, que será implementada na interface a ser desenvolvida[1].

O algoritmo em si é consideravelmente complexo e abstrato. Como o próprio nome sugere, baseia-se em duas etapas principais: *Map* e *Reduce*. No primeiro passo, (o *Map*), a entrada de dados é lida e subdividida em grupos menores, os quais também são lidos e subdivididos em grupos ainda menores formando uma árvore de dados multi-nível, onde cada grupo de dados recebe etiquetas (*tags*) contendo chave e valor para indexação (esta indexação é feita a nível de *tag* e não de registro como em um SGBD). Essa etapa é processada de forma paralela e concomitante, de preferência em um conjunto de servidores, para que seja completada o mais rápido possível. Então o segundo passo (*Reduce*) é executado e todas as etiquetas chave e valor são organizadas e agrupadas.

O Hadoop gerencia todo esse processo e garante a redundância dos dados através do sistema de arquivos HDFS (*Hadoop Distributed File System*). A ferramenta pode ser implementada em apenas um ou em um conjunto enorme de servidores, dependendo da necessidade da organização. Numa instalação em mais de um servidor, o sistema de arquivos replica os dados que estão sendo executados entre os servidores disponíveis, tornando o programa tolerante a falhas, além de executar as tarefas de forma concomitante, aumentando a velocidade da execução.

A figura 1 fornece uma representação visual de como funciona o algoritmo no Hadoop. A implementação de cada uma das etapas (*Map* ou *Reduce*) cabe ao programador, que definirá como serão tratados os dados inseridos e qual a lógica da aplicação, determinando o que será exibido no resultado final.

[1] http://pig.apache.org/

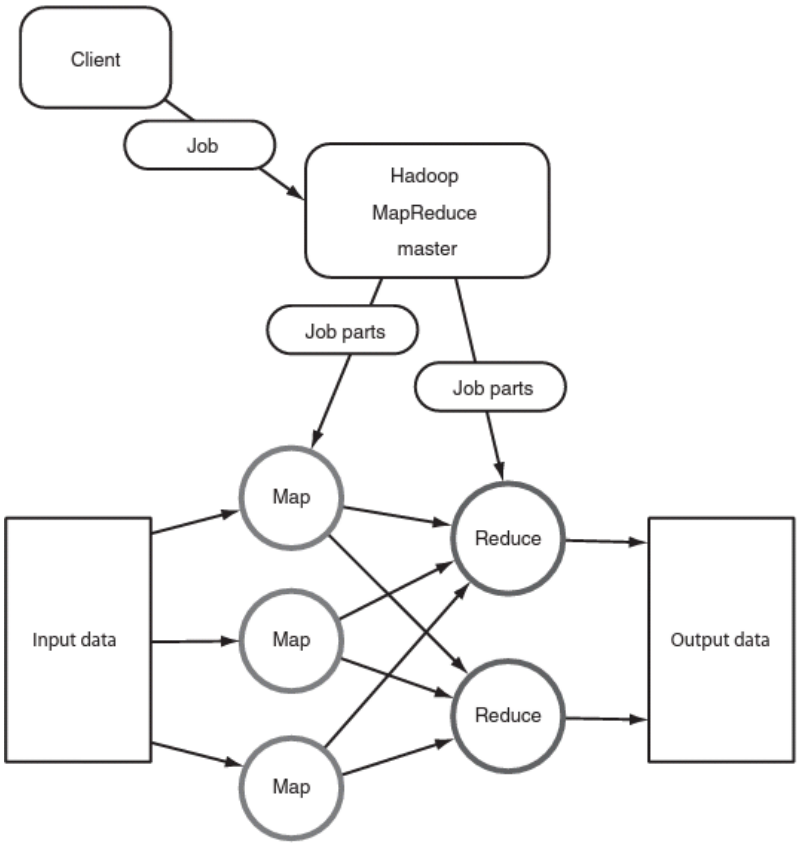

Figura 1. Estrutura do algoritmo *Map Reduce* Fonte: [Holmes 2012]

2.4. Plataformas de *big data*

Big data já é um conceito aplicado e utilizado mundialmente, nos mais variados setores econômicos. Há disponível atualmente várias plataformas e a implementação e armazenamento dos dados atualmente é vendida como uma *commodity*, implementando o conceito de *Software as a service* - SaaS, ou software como um serviço [Rouse 2013].

Neste modelo, a implementação do software e a infraestrutura é oferecida como um serviço disponível, geralmente, na web. Ou seja, o cliente apenas precisa fornecer a fonte dos dados a serem utilizados e as respectivas plataformas providenciam algumas opções de operações a serem realizadas. Geralmente essas plataformas possibilitam a inserção de um *script* ou um arquivo JAR compilado onde o cliente pode manualmente definir as operações a serem realizadas com os dados.

Entre as empresas que disponibilzam serviços de *big data* no formato mencionado, destacam-se a Amazon, a Microsoft e Cloudera. Todas possuem servidores disponíveis para rodar *jobs* com *big data* e geralmente permitem um período de testes para que depois o cliente contrate o serviço e tenha acesso ilimitado às demais funcionalidades. No site Cloudera.com, está disponível - para fins não-comerciais - uma máquina virtual (*Virtual Machine*, ou VM) [2] contendo o sistema operacional CentOS, com uma instalação do Hadoop e *HDFS*, configurado como *single cluster* [3], compatível com Virtual Box, VMWare

[2]Link para download: http://www.cloudera.com/content/support/en/downloads/download-components/download-products.html?productID=F6mO278Rvo

[3]Informações sobre instalações do Hadoop single cluster e multi cluster podem ser obtidas respecti-

e KVM.

A disponibilização desta máquina virtual constitui-se, portanto, de um ótimo recurso que providencia a instalação e configuração do Sistema Operacional e do Hadoop em si e vários outros softwares auxiliares, facilitando a utilização de *big data* no ambiente acadêmico. Portanto, para fins de implementação da interface proposta no presente artigo, foi utilizada esta máquina virtual disponível pela empresa Cloudera.

2.5. Portais de dados abertos

É comum, atualmente, empresas e organizações armazenarem seus dados em servidores distintos, em um ou mais serviços contratados para tal. Esta terceirização do armazenamento e do gerenciamento dos dados, resulta na necessidade de se utilizar portais, cuja função é organizar os dados existentes e fornecer um ponto de acesso único para servidores distintos espalhados através da *web*. Esses portais são conhecidos como sistemas de gerenciamento de dados (*Data Management Systems*, DMS) e possibilitam a publicação, compartilhamento e utilização dos dados.

Existem atualmente alguns desses portais de conhecimento distribuídos gratuitamente na internet. Uma das alternativas, provavelmente a mais conhecida e utilizada, é o CKAN [Ckan 2013], uma ferramenta de código aberto flexível e configurável, utilizado por organizações governamentais e não-governamentais. Entre os usuários da plataforma, podemos encontrar dados coletados pelos governos do Reino Unido[4] e do Brasil[5].

O presente projeto tem como objetivo a utilização de um portal como fonte de dados a serem utilizados em conjunto com sistemas *big data*. O usuário poderá utilizar a interface construída para realizar operações nos dados referenciados pelo portal implementando as técnicas de *Map* e *Reduce*.

O IFRS campus Porto Alegre conta atualmente com um portal de dados abertos CKAN (*http://ckan.inf.poa.ifrs.edu.br*) o qual possui links para dados variados de pesquisas em andamento no campus, entre eles os da pesquisa desenvolvida por [Peres et al. 2013], sobre a qualidade da água do Arroio Dilúvio, na cidade de Porto Alegre - RS.

3. Metodologia e Cenário de Implementação

Para a execução do projeto, foi utilizado o ambiente de virtualização disponível no IFRS. Este ambiente conta com a possibilidade da criação da estrutura necessária para hospedar os servidores Hadoop, além de já contar com um servidor CKAN, que será utilizado para os testes e desenvolvimento da plataforma.

Contando com essa possibilidade, o sistema Hadoop disponibilizado como máquina virtual pela empresa Cloudera foi instalado em um servidor no IFRS.

O cenário criado é composto então por um servidor virtual contendo a infraestrutura do Hadoop (*HDFS, Pig, Hive*) e um segundo servidor contendo o sistema de dados

---

vamente em: http://www.michael-noll.com/tutorials/running-hadoop-on-ubuntu-linux-single-node-cluster e http://www.michael-noll.com/tutorials/running-hadoop-on-ubuntu-linux-multi-node-cluster

[4]http://data.gov.uk/

[5]http://dados.gov.br/

abertos CKAN, onde foi instalada a plataforma desenvolvida que recebe os comandos dos usuários.

Foi preciso configurar a rede para que o servidor Hadoop tenha um IP fixo e aceite conexões *ssh* de um outro *host* conhecido (CKAN) de forma direta (sem necessidade de senha). Com estes servidores instalados, seguiu-se o estudo e realização de testes na estrutura *big data*.

Com o domínio das linguagens de consulta e consolidação dos dados no Hadoop, partiu-se para a criação de uma interface para seu uso. Nesta etapa foram levantados os requisitos necessários para tornar esta interface acessível aos diferentes tipos de usuários do sistema. Assim, quando o usuário executa um comando na interface, esta acionará, via *Secure Shell* - SSH, a *VM* contendo o Hadoop, que executará a tarefa desejada e retornará a URL para os dados obtidos através do *script* executado.

Os dados a serem utilizados pelo Hadoop precisam estar carregados no sistema de arquivos HDFS do servidor Hadoop, portanto o usuário precisa informar na interface a URL de um dataset válido, no formato CSV, para que este arquivo esteja disponível às operações seguintes. Posteriormente, o usuário definirá o delimitador, o número de colunas e os tipos de dados de cda coluna no arquivo.

O CKAN servirá como o portal de entrada no qual o usuário terá acesso à plataforma e aos resultados das operações *big data*.

4. Implementação e Resultados Obtidos

A plataforma desenvolvida no presente trabalho de conclusão foi resultado da integração dos dois sistemas descritos acima. Foi desenvolvida uma interface entre a plataforma de dados CKAN e uma máquina virtual rodando o Hadoop.

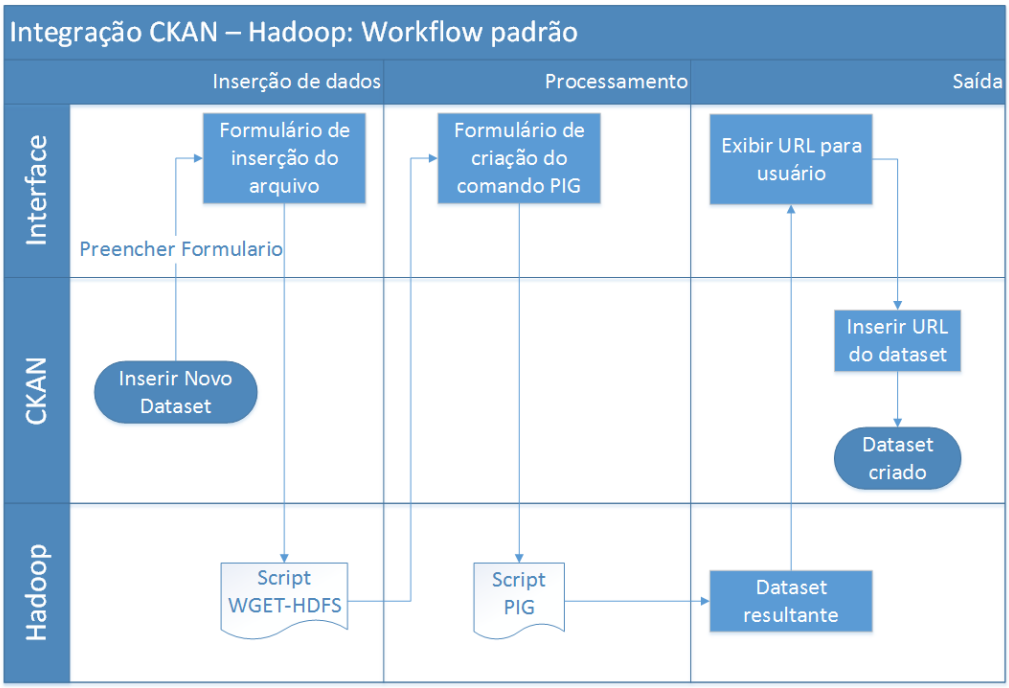

Figura 2. Estrutura do ambiente a ser implementado

Na interface da plataforma, o usuário apenas precisará definir a origem dos dados (link na web) e escolher a(s) tarefa(s) a ser(em) executada(s) pelo Hadoop. Após estas entradas, receberá como retorno o local onde está armazenado arquivo contendo o resultado da operação. Na figura 2, pode-se ver uma ilustração gráfica do sistema em questão.

A interface é voltada a usuários leigos na produção e utilização de algoritmos *Map Reduce* [6], através do carregamento de arquivos com extensão ".CSV"[7]. Uma vez carregado o arquivo, o usuário deve definir o delimitador das colunas utilizado no arquivo, a quantidade de colunas existentes e o tipo de dados das mesmas, e posteriormente escolher uma operação, além de selecionar a respectiva coluna na qual a operação será efetuada.

Partindo do pressuposto que a plataforma proposta pelo presente artigo será utilizada para operações simples em conjuntos de dados complexos e extensos, a utilização do Pig, como ferramenta de interação, tornou-se uma alternativa viável pela facilidade de implementação e interação. Outra característica positiva que facilita a escolha da linguagem é a possibilidade de aplicação de três operações básicas em *big data*: extrair, transfomar e carregar, do inglês ETL - *Extract, Transform & Load*.

A utilização do Pig Latin como linguagem também facilita a implementação do sistema proposto por suas semelhanças com linguagens SQL tradicionais. Na figura 3 é apresentado um exemplo de *script* desenvolvido para rodar em Pig (lado esquerdo) e uma representação em diagrama do fluxo de execução do mesmo. As operações que o Pig Latin suporta são praticamente as mesmas que as linguagens SQL: *Group by, Sort, AVG, Count*, entre outras.

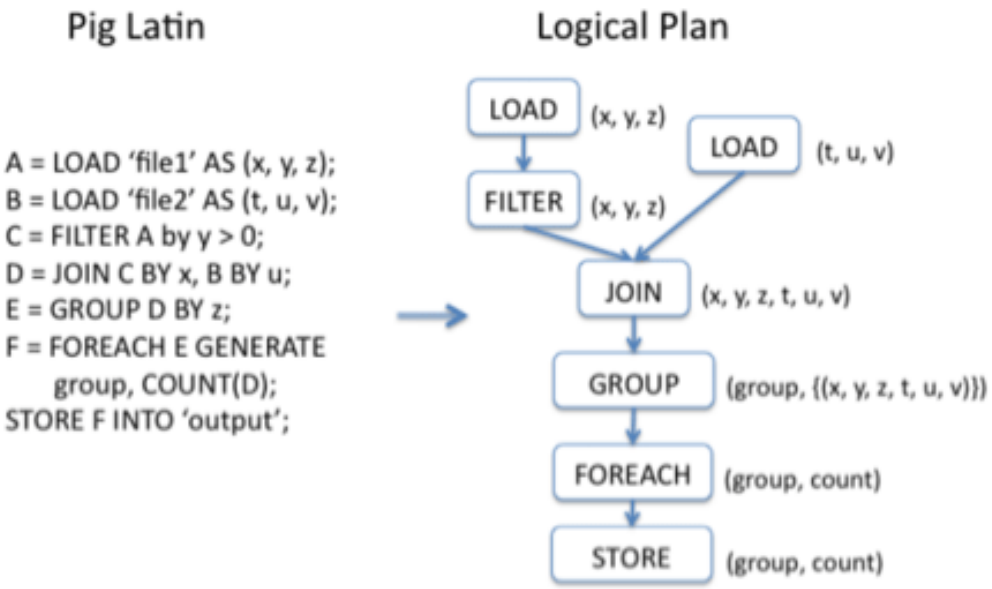

Figura 3. Exemplo de *script* em Pig Latin

[6]Considerando que a criação e execução de algoritmos *Map Reduce* constitui-se de uma tarefa complexa mesmo para desenvolvedores experientes na ferramenta.

[7]A existência de *datasets* com esta extensão é bem comum em sites que disponibilizam *datasets* dos mais variados tipos de dados: clima, cotações financeiras, demografia, entre outros.

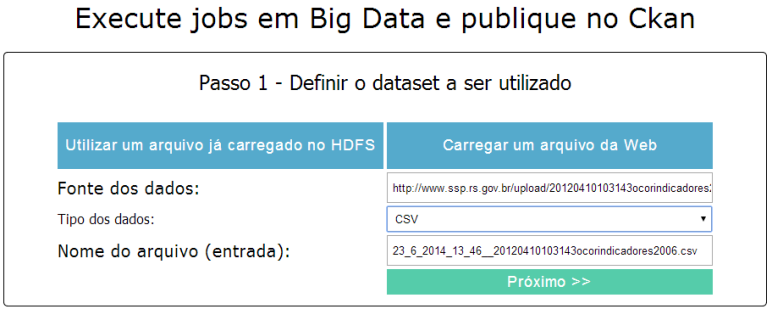

Figura 4. Inserindo um novo arquivo

Figura 5. Arquivos disponíveis no HDFS

No tocante à interação com o usuário, a interface proposta procura ser simples e intuitiva, facilitando a execução das tarefas desejadas. Como se pode perceber pelo exemplo exposto acima, o *script* é constituído por três partes: inserção dos dados (*LOAD*), filtragem e operações, ou funções. A primeira etapa será a da inserção dos dados, onde o usuário deverá providenciar a fonte dos dados a serem consultados.

O usuário pode optar por utilizar um arquivo já carregado no HDFS do Hadoop ou carregar um novo arquivo ".CSV"da web. Caso opte pela primera opção é realizada uma consulta via *Asynchronous Javascript and XML*, AJAX ao servidor do Hadoop e a interface exibe então os arquivos disponíveis no servidor, conforme a figura 5.

Caso opte por adicionar um novo arquivo, é exibido um pequeno formulário para inserção da URL de origem dos dados. O nome do arquivo será gerado automaticamente para evitar conflitos de nome no hdfs, será então preenchido o campo de nome do arquivo contendo um prefixo baseado na data e hora da inserção. Após o nome gerado, aparece então a opção de enviar o arquivo selecionado que será armazenado com o nome exibido na tela, conforme a figura 4 [8].

Independente da opção selecionada na etapa anterior, uma vez selecionado o arquivo, na segunda etapa, o usuário poderá então definir as regras de filtragem e a função a ser executada. Aqui então será montado um algoritmo de execução baseado na entrada do usuário. Esse algoritmo será implementado em Pig Latin e executará no Pig, que ativará a plataforma Hadoop e fará toda a execução do *Map* e do *Reduce* no lado do servidor.

[8]Importante notar que o sistema implementado não aceita arquivos que necessitem de protocolo https pra serem acessados, apenas http.

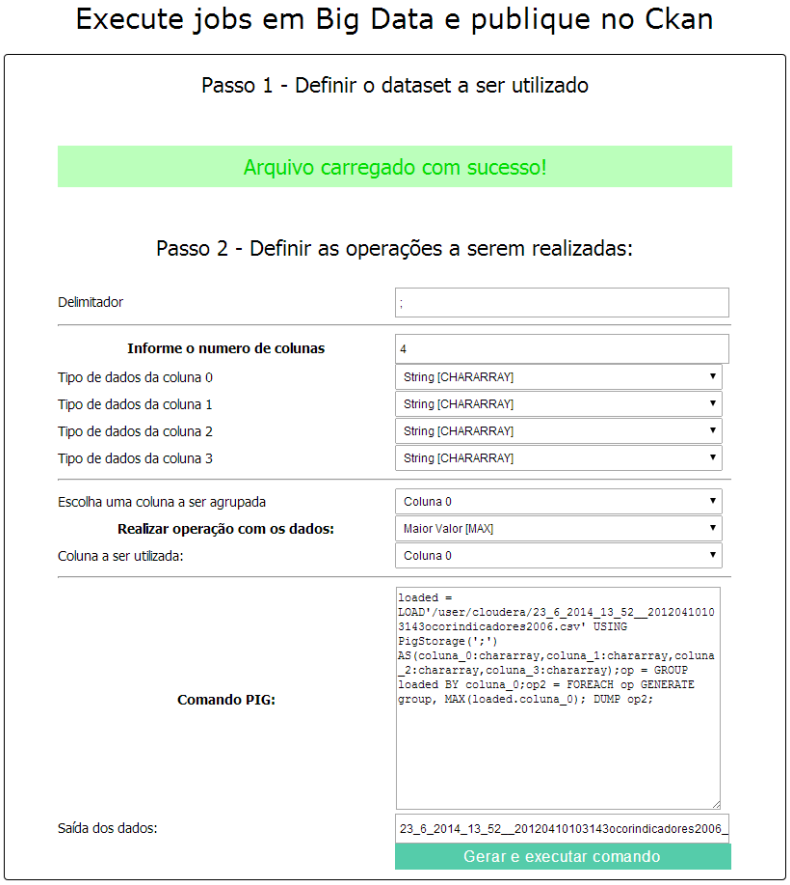

Figura 6. Definindo Regras de filtragem e exibição do comando Pig Latin resultante

O usuário deve então providenciar o delimitador do arquivo (geralmente ”;”ou ”,”) e a quantidade de colunas que o arquivo selecionado possui. Dependendo do número de colunas, serão criados novos campos para que o usuário defina o tipo de dados correspondente a cada uma das colunas que podem ser utilizadas posteriormente.

Então é escolhido uma das opções de operações e uma coluna correspondente, conforme a imagem 6. À medida em que o usuário preenche o formulário da interface que gerará o comando Pig, o mesmo é exibido na tela possibilitando a aprendizagem da estrutura e da sintaxe do Pig Latin [Apache 2014b].

Ao submeter esse formulário o comando Pig Latin é enviado via POST a um script em PHP que realizará uma conexão SSH com o servidor do Hadoop e executará via Shell script o comando gerado no servidor. A interface entra em modo de espera e aguarda pelo fim da execução das operações definidas pelo usuário.

A terceira e última etapa será a exibição do endereço de armazenamento da saída do comando executado. Aqui o usuário poderá fazer um download do arquivo resultante das operações executadas. Cada uma dessas etapas ocorre na mesma tela, através dos recursos de AJAX, contendo elementos de controle próprios.

5. Conclusões

A disponibilidade de se executar *jobs* com ferramentas *big data* no ambiente acadêmico, através de uma interface amigável aos usuários, cria a possibilidade de que pesquisadores das diversas áreas do conhecimento obtenham um ganho susbstancial de análise ao poder examinar e classificar dados sensoriais, climáticos, demográficos, políticos ou econômicos acessíveis na internet, coletados em qualquer lugar do mundo.

Ao longo do desenvolvimento do presente trabalho surgiram algumas dificuldades, especialmente ao se integrar, enviar comandos e monitorar a execução dos mesmos entre a interface e o servidor rodando o Hadoop. Tais operações foram implementadas através do envio do comando gerado na interface web por AJAX a um script PHP, que por sua vez abre uma conexão SSH com o servidor do Hadoop e executa um Shell script com os parâmetros recebidos e inicia a execução do*job*.

Existem diversas melhorias que ainda podem ser implementadas no sistema em questão, entre elas a possibilidade de incluir outros tipos de arquivos além dos ".CSV", gerar comandos mais complexos através da interface (sem exigir do usuário um conhecimento em PIG) e incrementar a escalabilidade da infraestrutura implementada, adicionando novos servidores para a formação de um cluster de alto desempenho nas ações de *map* e *reduce* e expandindo a capacidade máxima atual de 40Gb no HDFS.

A utilização de *big data* em pesquisas cria a possibilidade de que conclusões e inferências sejam feitas utilizando grandes quantidades de dados. A partir da inserção desta tecnologia, análises e constatações podem ser feitas com base na análise do todo, ao invés de partes. A evolução tecnológica passa a possiblitar a abertura de novos horizontes, e sua utilização contribui para facilitar o aprofundamento da análise de dados das pesquisas.

Neste trabalho, foi desenvolvida uma ferramenta capaz de criar uma camada de abstração entre um usuário leigo e a complexidade da análise de *big data*. Espera-se que com esta ferramenta, o acesso a esta inovadora tendência de análise de dados seja facilitado. Não se pretende restringir o usuário aos limites da interface desenvolvida, porém fornecer um forma de introduzir os pesquisadores da instuição às possibilidades do *big data*, algo que antes não era possível a partir da interface disponibilizada pelo Ckan.

Referências

Apache (2013). Apache hadoop webpage. Disponível em: http://hadoop.apache.org/.

Apache (2014a). Apache hadoop wiki. Disponível por em: http://wiki.apache.org/hadoop/HadoopIsNot.

Apache (2014b). Pig latin reference manual. A documentação da sintaxe do PIG pode ser encontrada em: http://pig.apache.org/docs/r0.7.0/piglatin$_ref2.html$.

Ckan (2013). Ckan the open source data portal software. Disponível por www em: http://ckan.org/.

Dean, J. and Ghemwat, S. (2004). Mapreduce: Simplified data processing on large clusters. *Google Research Publication, Inc.* disponível em: http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/pt-BR//archive/mapreduce-osdi04.pdf.

Franks, B. (2012). *Taming the big data tidal wave: Finding opportunities in huge data streams with advanced analytics*, volume 56. Wiley. com.

Holmes, A. (2012). *Hadoop in Practice*. Manning.

Huppke, R. W. (2013). A tower of babble, built out of old tweets: Twitter archive is too massive for library of congress to offer searches. *Chicago Tribune*. disponivel em: http://articles.chicagotribune.com/2013-01-08/news/ct-met-huppke-twitter-0108-20130108_1_twitter-archive-tweets-library.

Hurwitz, J., Nugent, A., Halper, F., and Kaufman, M. (2013). *Big Data for Dummies*. Wiley. com.

Michel, F. (2013). How many photos are uploaded to flickr every day, month, year?

Mohamed Ali Feki, Fahim Kawsar, M. B. and Trappenies, L. (2013). The internet of things: The next technical revolution. *Guest Editors' Introduction. Published by the IEEE Computer Society*.

Peres, A., Miletto, E., Kapusta, S., Ojeda, T., Lacasse, A., and Gagnon, J. (2013). Waits - an it structure for environmental informatio via open knowledge, synamic dashboards and social web of things. In *Proceedings of the IADIS International Conference WWW/Internet 2013*, volume 1.

Rouse, M. (2013). What is software as a service? Disponível por www em: http://searchcloudcomputing.techtarget.com/definition/Software-as-a-Service.

Sagirolou, S. and Sinanc, D. (2013). Big data: A review. *Gaza University*.

Stone, B. (2013). Tweet preservation. Disponível por www em: https://blog.twitter.com/2010/tweet-preservation.

Twitter (2013). Twitter homepage. Disponível por www em: http://www.twitter.com/.

Watters, A. (2011). How the library of congress is building the twitter archive. In Media, O., editor, *Big Data Now*, page 137. O'Reilly Media Inc.

Weiss, R. and Zgorski, L. (2012). Obama administration envails big data. *Officeof Science and technology Policy Executive Office of the President*.