# Big Data e Transparência: Utilizando Funções de Mapreduce para incrementar a transparência dos Gastos Públicos

## Alternative Title: Big Data and Transparency: Using MapReduce functions to increase Public Expenditure transparency

Eduardo de Paiva
Programa de Pós-graduação em Informática
Universidade Federal do Estado do Rio de Janeiro
Avenida Pasteur 458, Urca
Rio de Janeiro, RJ, Brasil
eduardo.paiva@uniriotec.br

Kate Revoredo
Programa de Pós-graduação em Informática
Universidade Federal do Estado do Rio de Janeiro
Avenida Pasteur 458, Urca
Rio de Janeiro, RJ, Brasil
katerevoredo@uniriotec.br

## RESUMO

Atualmente todos os entes governamentais devem manter portais de transparência que evidenciem a arrecadação de receitas e as despesas executadas diariamente. No entanto, a mera disponibilização dessas informações em portais governamentais não assegura um aumento efetivo do grau de transparência desses entes, pois o grande volume de dados aliado à falta de padrões torna inviável qualquer tipo de acompanhamento sistemático desses dados. Este artigo sugere a aplicação de técnicas de programação paralela baseadas no paradigma de programação mapreduce para fazer a identificação de um conjunto pré-determinado de produtos comprados pela Administração Pública, além de propor uma forma de consolidação dessas informações de maneira que permita a fácil visualização de disparidades encontradas no grande volume de dados apresentados. A solução proposta foi testada em um estudo de caso executado no Portal da Transparência do Governo Federal. Os resultados obtidos apontaram que tais técnicas se constituem promissoras ferramentas para questões ligadas às áreas de transparência, que normalmente tratam grandes volumes de dados, mas que nem sempre apresentam informações de qualidade.

## Palavras-Chave

Transparência pública, Big data, Mineração de texto.

## ABSTRACT

Nowadays all government entity must maintain transparency portals that shows the all revenue and expenditure carried out daily. However, the mere availability of such information in government portals does not ensure an effective increase in the degree of transparency of these entities, because the large volume of data combined with the lack of standards makes it impossible any systematic monitoring of such data. This paper suggests the application of parallel programming techniques based on mapreduce programming paradigm to the identification of a predetermined set of products purchased by the Public Administration. It also proposes a way to consolidate this information to make easy viewing of disparities found in the large volume of data presented. The proposed solution was tested in a case study performed in the Transparency Portal of the Federal Government. The results suggest that the presented techniques constitute a promising approach to issues related to transparency areas, which normally handles large volumes of data, but it does not always provide quality information.


*SBSI 2016*, May 17–20, 2016, Florianópolis, Santa Catarina, Brazil.


## Categories and Subject Descriptors

I.2.7 [Computing Methodologies]: Natural Language Processing - Text analysis

J.1 [Computer Applications]: Administrative Data Processing - Government

## General Terms

Management, Documentation, Performance.

## Keywords

Public transparency, Big data, text mining.

## 1. INTRODUÇÃO

A popularização da Internet tem ajudado a tornar os portais de transparência governamental importantes ferramentas de consolidação e desenvolvimento da democracia [1]. Um dos fatores que impulsionou a institucionalização desses instrumentos de transparência pública foi a publicação da lei complementar 131 [3], que determina a disponibilização, em tempo real, de informações pormenorizadas sobre a execução orçamentária e financeira da União, dos Estados, e dos Municípios. Ou seja, atualmente todos os entes governamentais devem manter portais de transparência que evidenciem a arrecadação de receitas e as despesas executadas diariamente.

No entanto, a mera disponibilização dessas informações em portais governamentais não assegura um aumento efetivo do grau de transparência desses entes, pois o grande volume de dados aliado à falta de padrões torna inviável qualquer tipo de acompanhamento sistemático desses dados.

A solução para esse tipo de problema se dá através da aplicação de técnicas de tratamentos de dados que permitam a estruturação da informação de forma mais clara e elucidativa para a população. Para isso, é necessário que se desenvolvam ferramentas capazes de processar esse grande volume de dados e que permitam uma visualização consolidada dessas informações, facilitando assim a identificação de informações que se desviem da normalidade.

Nesse sentido, Rommel, Carvalho et al. [18] e Rommel, Carvalho et al. [19] sugerem uma metodologia para aumentar o grau de informatividade do Portal da Transparência do Governo Federal. Os autores propõem a formulação de um banco de preço a partir do tratamento das informações textuais que descrevem as compras. Nessa proposta os produtos são identificados a partir da combinação de palavras chaves. A metodologia desenvolvida se apoia em um processamento sequencial, com a utilização de ferramentas de ETL[1] e posteriormente técnicas de mineração de texto e clusterização, porém, o grande volume de textos a serem analisados torna o processamento muito pesado e inviabiliza a implementação da analise textual de forma concomitante com a carga diária de dados que são feitas na alimentação do portal.

Atualmente, tendências de processamento intensivo de dados e de computação paralela baseadas na infraestrutura do hadoop [21] e no paradigma de programação mapreduce [13] têm se apresentado como soluções para esses e outros tipos de problemas advindos da necessidade de processamento de grandes volumes de dados.

O trabalho em questão propõe uma metodologia de tratamento de informação para a obtenção de visões mais elucidativas sobre os gastos públicos. O objetivo é a aplicação de técnicas de programação paralela baseadas no paradigma de programação mapreduce para fazer a identificação de um conjunto pré-determinado de produtos comprados pela Administração Pública, além de propor uma forma de consolidação dessas informações de maneira que permita a fácil visualização de disparidades encontradas no grande volume de dados apresentados. Esse mecanismo propiciará uma melhor avaliação dos gastos públicos, permitindo com que a população possa entender melhor como o governo está empregando seu dinheiro.

O restante desse artigo está organizado da seguinte forma: a seção 2 faz uma revisão sobre Big Data e o modelo de programação MapReduce. Já a Seção 3 descreve a metodologia proposta, enquanto que as seções 4 e 5 apresentam o estudo de caso desenvolvido e os resultados obtidos. Finalmente, a seção 6 faz a conclusão do trabalho.

## 2. BIG DATA E O MODELO DE PROGRAMAÇÃO MAPREDUCE

Atualmente, existem várias definições para o termo Big Data. Manyika et al. [14] consideram Big Data como sendo o conjunto de dados cujo tamanho é maior do que a capacidade que as ferramentas de software de banco de dados tradicionais têm para capturar, armazenar, gerenciar e analisar. Já Taurion [20] define Big Data como sendo grandes volumes de dados que chegam de uma variedade de fontes, em alta velocidade e com veracidade e que agregam algum tipo de valor a um negócio. Gupta et al. [12] resumem Big Data como sendo um grande volume de dados que excede a capacidade de processamento dos bancos de dados convencionais.

No entanto, independentemente de qualquer definição formal, o fato é que a escalada do volume e variedade de dados vem trazendo dificuldades para o processamento dessas informações. Lin e Dyer [13] apontam que a única abordagem viável para esse tipo de problema é a dividir e conquistar. A estratégia básica dessa abordagem é particionar um problema grande em subproblemas menores. Dessa forma, os subproblemas independentes podem ser tratados de forma paralela por diferentes threads, núcleos de processadores ou máquinas em clusters [13].

No entanto, em ambientes tradicionais de programação paralela ou distribuída, o programador precisa tratar explicitamente questões complexas como por exemplo: forma de paralelização, tolerância a falhas, distribuição de dados entre os nós de processamento, balanceamento de carga e acesso à memória compartilhada.

O MapReduce, introduzido por Dean and Ghemawat [8], é um modelo de programação paralela para grandes volumes de dados. Ele é inspirado na estratégia dividir e conquistar, mas abstrai do programador a complexidade dos problemas típicos do gerenciamento de aplicações distribuídas, permitindo que o desenvolvedor possa se dedicar apenas à solução do problema a ser tratado, deixando que a aplicação execute a distribuição e o paralelismo.

Neste modelo, que roda em clusters de computadores, as tarefas de processamento são distribuídas entre os nós (máquinas do cluster), implementando uma arquitetura mestre-escravo, na qual, um nó mestre é responsável pelo gerenciamento, e os nós escravos são responsáveis pelo processamento propriamente dito.

No modelo mapreduce, o programador só precisa desenvolver duas funções principais: Map e Reduce [9]. Essas funções trabalham basicamente com pares de chaves e valores, sendo que a função Map gera um conjunto de pares (chave-valor) intermediários, que são passados para a função Reduce, que é responsável por processar e juntar todos esses pares (chave-valor) intermediários, associando os pares com a mesma chave em uma espécie de sumarização dos resultados. Entre as operações de Map e reduce, esses pares de chave e valor passam pela operação de "Shuffle and Sort". Nessa fase, que ocorre de forma automatizada, os pares chave-valor, gerados pelos nós de map, são distribuídos para os nós responsáveis pelo reduce, no entanto, antes que a operação reduce comece, esses pares são ordenados pela chave, de forma que a função reducer já receba os pares de maneira ordenada pela chave.

Existem várias ferramentas que implementam o modelo de programação MapReduce, dentre as quais, o Hadoop é a mais conhecida. A figura 1 representa o sistema de execução de programas MapReduce no Hadoop [21].

Dean e Ghemawat [8] descrevem a execução de um programa MapReduce, apresentada na figura 1, pelos seguintes passos: (i) a base de dados de entrada, contendo os arquivos a serem analisados, é armazenada em um sistema de arquivo distribuído composto por diversos registros divididos em blocos. Antes do início da execução propriamente dita, cópias do programa desenvolvido pelo usuário (funções de map e reduce) são distribuídas por todos os nós (máquinas) do cluster (fluxo 1 da figura 1); (ii) o modelo implementa uma arquitetura mestre-escravo, e as tarefas de map e reduce (definidas pelo usuário) são atribuídas aos nós escravos pelo nó mestre (fluxos 2 da figura 1);

[1] ETL: Extract, Transform and Load – Extração, Transformação e Carga. As ferramentas de ETL são utilizadas para auxiliar o tratamento de dados.

(iii) um nó escravo recebe a tarefa map a ser executada tendo como entrada um bloco de dados designado pelo nó mestre (fluxo 3 da figura 1), sendo que, após o processamento da função map, o nó escravo armazena os pares (chave-valor) intermediários em uma memória temporária, para posterior envio à fase seguinte; (iv) periodicamente, os pares produzidos pelos nós responsáveis pela execução da função map são escritos em discos locais (fluxo 4 da figura 1), sendo que a localização desses discos locais são passadas para o nó mestre, que é responsável por passar a localização desse armazenamento temporário aos escravos responsáveis pela execução da tarefa de reducer; (v) quando um nó escravo, responsável pela tarefa de reduce, é notificado pelo mestre sobre essa localização, usa chamadas remotas para ler esses dados no disco local do escravo que executou a tarefa de map (fluxo 5 da figura 1); (vi) o nó escravo responsável pela função de reduce itera esses dados intermediários ordenados e para cada chave ele agrega os valores correspondentes, de acordo com a função reduce definida pelo usuário; (vii) após a conclusão de todas as tarefas de map e reduce, o resultado da execução é disponibilizado em arquivos de saída (fluxo 6 da figura 1).

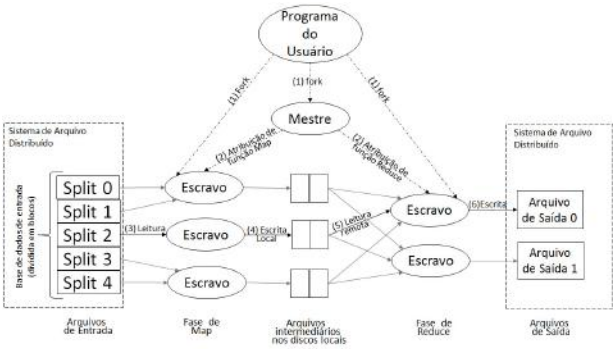


**Figura 1. Modelo de execução do MapReduce, adaptado de [8]**

## 3. METODOLOGIA PROPOSTA

O objetivo da metodologia ora proposta é a utilização dos conceitos de programação MapReduce a fim de tornar viável o processamento dos grandes volumes de dados que são apresentados nos portais de transparência pública. A metodologia apresenta uma forma de analisar as descrições textuais dos itens de empenho[2], que são apresentados nos portais de transparência, possibilitando assim a consolidação de uma série de informações a respeito dessas compras. Como resultado, espera-se obter, a partir de informações textuais (campo de descrição dos itens de empenho), um conjunto de dados estruturados que qualificam o produto comprado.

[2] O empenho é a primeira fase da execução da despesa pública. Essa é fase em que despesa é especificada de forma mais detalhada. Um empenho pode possuir vários itens de empenho, sendo que cada um desses itens especifica uma determinada despesa. De acordo com a legislação brasileira em vigor sobre finanças publica [2] e [4], todas as despesas realizadas pelo poder público devem ser precedidas do empenho, dessa forma, qualquer produto adquirido pela Administração Pública tem um documento de item de empenho correspondente, com as informações relativas a essa compra.

A arquitetura da solução está dividida em duas partes. A primeira, composta pelas funções de "mapper", é responsável pela qualificação dos produtos comprados através da aplicação de regras de identificação. Já a segunda, compostas pelas funções de "reducers", faz a agregação dos produtos identificados na fase anterior, a fim de se obter informações que tracem o perfil das compras feitas pelo governo. No entanto, antes dessas duas fases, o dado precisa passar por um pré-processamento. A figura 2 apresenta a arquitetura da solução proposta, sendo que, as subseções seguintes irão detalhar as atividades representadas na figura.

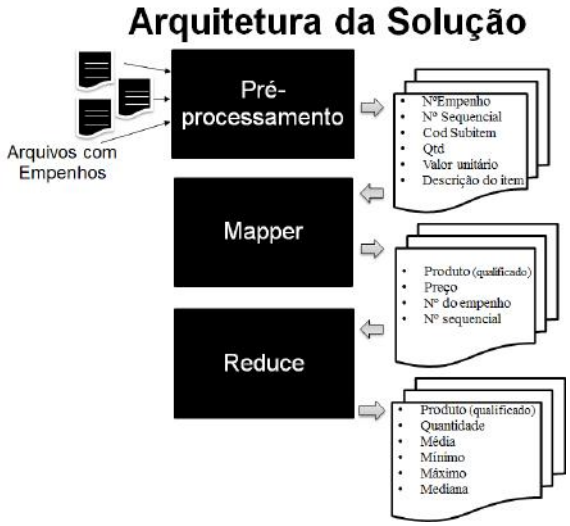


**Figura 2. Arquitetura da solução proposta**

### 3.1 Pré-processamento dos dados

O pré-processamento consiste da extração de algumas informações úteis ao processo de classificação subsequente e da identificação de sentenças dentro dos textos descritivos, visto que, estas sentenças fornecem características dos produtos. Essa atividade também é responsável pela uniformização do texto a ser analisado.

Na etapa de pré-processamento são extraídos alguns atributos que serão úteis às demais fases do processo: o quantitativo do que está sendo comprado, a unidade de medida, a marca do produto em questão, o número do processo e o código do item de material. A identificação dessas informações nos textos foi possível devido ao fato dessas descrições seguirem um padrão. Essas extrações foram realizadas com o emprego da técnica enunciada por Etzioni et al. [10], que sugere a utilização de templates para auxiliar a recuperação de informações.

A figura 3 ilustra as informações recuperadas durante a primeira fase do pré-processamento.

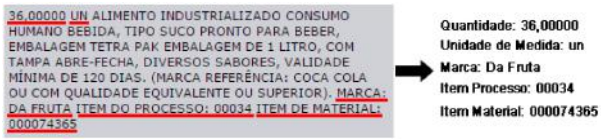


**Figura 3. Recuperação de informações**

A descrição do código do item de material (extraído) é obtida pela tabela de cadastro de materiais, da base de dados do SIASG[3], disponível para download no Portal Brasileiro de Dados Abertos [16].

A fase do pré-processamento também é responsável pela identificação e separação das sentenças textuais dentro das descrições. Essas sentenças servem para fornecer características adicionais dos produtos que serão identificados na mineração de texto, conforme empregado por Munková et al. [17].

Por fim, o pré-processamento também realiza a uniformização dos textos descritivos, passando todos os caracteres para o formato minúsculo e eliminando o excesso de espaços entre as palavras.

## 3.2 Funções Mappers

As funções de mappers são responsáveis pela categorização dos produtos. Elas recebem como entrada um conjunto de arquivos textos contendo os dados dos empenhos. Cada linha desses arquivos representa uma compra distinta e é composta pelos seguintes atributos:

- **Nº do Empenho** - identifica o empenho em questão
- **Nº sequencial** – identifica um determinado item de compra no empenho (cada empenho pode ter vários itens de compra).
- **Código do Subitem** – parte integrante de um código utilizado pela Contabilidade Pública. O subitem identifica a natureza do gasto que está sendo feito.
- **Quantidade** – indica a quantidade, do item em questão, que está sendo comprada.
- **Valor unitário** – identifica o valor unitário pago pelo item em questão.
- **Descrição do item** – campo de texto livre que especifica de forma detalhada o item que está sendo comprado, bem como a unidade de medida que está sendo utilizada para quantificá-lo.

A figura 4 apresenta um fragmento de um arquivo de entrada utilizado pelas funções de Mapprers, sendo que os campos são separados pelo símbolo de "pipe" ( | ).

```
…
114605113012014NE800073|1|1|374,96|3,00|374,96000 litro GASOLINA COMUM GAS …
110404000012014NE801146|5|16|29|24,00|29,00000 RESMA PAPEL A3, LARGURA 297 …
120003000012014NE800170|14|7|140|1,00|140,00|QUILOGRAMA FRUTA IN NATURA, …
…
```

**Figura 4. Fragmento de um arquivo de entrada**

As funções de mapper serão responsáveis por separar essas informações recebidas e tratá-las de forma individualizadas.

Nessa etapa são analisadas as descrições dos itens de empenho. O objetivo dessa fase é fazer a identificação dos produtos que estão sendo descritos nas especificações das compras. Essa identificação se dá através da aplicação de regras de identificação, estabelecidas por especialistas. Tais regras combinam dados presentes nos campos estruturados e informações contidas no campo textual (descrição do item).

Ao final do processo, as funções de mapper geram arquivos em que cada linha possui as seguintes informações:

- **Produto** – nome do produto e unidade de medida utilizada para quantificá-lo. A principal tarefa dos mappers é fazer essa identificação. Esse campo será utilizado como chave na atividade de reducer.
- **Valor** - valor unitário pago pelo produto. Esse campo será utilizado para o cálculo dos valores estatísticos dos produtos, durante a fase de "reducer".
- **Nº do Empenho** – Essa informação não é utilizada pelo reducer, porém ela é arquivada para ser utilizada em auditorias para a identificação de possíveis compras superfaturadas.
- **Nº Sequencial** – Essa informação não é utilizada pelo reducer, porém ela é arquivada para ser utilizada em auditorias para a identificação de possíveis compras superfaturadas.

Cabe ressaltar que o arquivo gerado só conterá as informações dos empenhos que foram categorizados em um dos produtos pré-estabelecidos (os demais empenhos serão descartados). Esses produtos pré-estabelecidos são aqueles que se tem alguma regra de identificação definida por especialistas e implementadas nas funções de mappers.

Os arquivos gerados pelos mappers, além de serem passados para as funções de reducer, também são armazenados para análises futuras, a fim de se identificar as compras que possivelmente apresentem resultados discrepantes, quando comparadas com outras compras similares.

Como apresentado na seção 2, a função de map deve gerar pares de chave-valor, para serem passados para a função de reduce. No caso da solução proposta, a chave é o nome do produto, e o campo valor é formado pelas demais informações que compõem a saída, cabendo à função reducer separar, interpretar e processar esses dados.

## 3.3 Funções Reducers

As funções de reducers recebem como entrada os arquivos gerados pelas funções de mappers e são responsáveis por fazer alguns cálculos estatísticos, que irão indicar o perfil das compras feitas pelos entes governamentais.

Nessa fase do processamento, todas as compras de um determinado produto são grupadas, permitindo assim o cálculo de algumas métricas sobre os produtos identificados. A saída gerada por essa fase é composta por um conjunto de linhas cuja configuração é apresentada abaixo:

- **Produto** - nome do produto e unidade de medida utilizada para quantificá-lo.
- **Quantidade** – indica o número total de compras distintas do produto em questão.
- **Média** – indica o preço médio pago pelo produto em questão.
- **Mínimo** – indica o menor valor pago nas compras do produto em questão.

[3] Sistema Integrado de Administração de Serviços Gerais (SIASG): sistema utilizado para processar as compras e aquisições de materiais e serviços do Governo Federal [11].

- **Máximo** – indica o maior valor pago nas compras do produto em questão.
- **Mediana** - indica a mediana do preço pago pelo produto em questão.

Essas informações, além de oferecerem o perfil das compras feitas pelo governo, fornecem um parâmetro de referência sobre a economicidade das compras realizadas, permitido que possam ser feitas comparações com os preços praticados pelo mercado.

## 3.4 O Processo de execução

O Processo de execução é exemplificado pela Figura 5. Apesar da solução proposta calcular outras métricas para os produtos identificados, a figura 5 apresenta apenas o cálculo do preço médio pago por produto, a fim de tornar o entendimento mais claro.

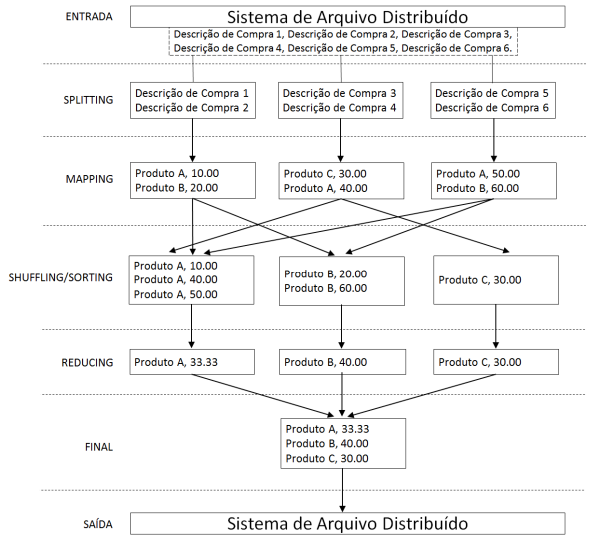


**Figura 5. Exemplo simplificado do processo de execução**

O primeiro passo para a execução do processo é a carga dos arquivos contendo as informações dos itens de compra no sistema de arquivo distribuído. A partir do momento que o usuário dá o comando para o início do processamento dos dados, o nó mestre divide os arquivos de entrada pelos nós escravos responsáveis por executar as funções de Map (essa atividade caracteriza o splitting). Então, esses nós analisam o texto descritivo de cada um dos itens de compra, aplicando as regras de identificação para cada um dos produtos, a fim de identificar o produto que está sendo designado em cada um desses itens de compra. Após essa identificação, as funções de map apresentam como saída pares de chave-valor, sendo que, o produto identificado será a chave e o preço pago na compra em questão o valor. No próximo passo, os pares chave-valor são agrupados e ordenados de acordo com suas chaves (nomes dos produtos). A partir daí cada um dos nós responsáveis pela execução das funções de reduce recebem um conjunto ordenado de pares chave-valor, a fim de agregá-los de acordo com os critérios definidos na função reduce (no caso da figura 5, os nós reduces estão apenas calculando as médias dos preços pagos por produtos). Finalmente, após o término do processamento, os resultados são carregados novamente no sistema de arquivo distribuído.

# 4. ESTUDO DE CASO

Para testar a metodologia proposta, foi aplicado um estudo de caso nos dados do Portal da Transparência do Governo Federal [7].

Esta seção demonstra o estudo de caso desenvolvido e está dividida em 2 subseções, a primeira apresenta a infraestrutura utilizada no estudo de caso, enquanto que a segunda descreve o conjunto de dados utilizado.

## 4.1 Infraestrutura utilizada

O estudo de caso foi desenvolvido no ambiente de computação nas nuvens disponibilizado pela empresa Amazon Web Services [6]. O processamento dos dados foi feito com a utilização de um cluster com o Elastic Mapreduce (EMR), solução baseada no hadoop, disponibilizada pela Amazon WebServices.

Optou-se por essa estrutura pelo fato de tal solução já apresentar todos os recursos de software e hardware necessários para a execução de aplicações que se utilizam hadoop [21].

O cluster utilizado era composto por 4 CPUs com Processadores Intel Xeon E5-2670 v2 (Ivy Bridge) de alta frequência, sendo que, essas máquinas possuíam 15 GiB de memória, 80 GB de armazenamento de instância local baseado em SSD (solid-state drive) e plataforma de 64 bit.

## 4.2 Conjunto de dados utilizados

O trabalho em questão irá utilizar os dados do Portal da Transparência do Governo Federal, mas seria possível a utilização dos dados do Portal de Transparência de qualquer estado ou município brasileiro, uma vez que as regras de contabilidade aplicadas ao setor público no Brasil se aplicam para todos os entes da Administração Pública. Dessa forma, o único requisito para a utilização de uma base de dados de algum estado ou município brasileiro é que esse ente dê publicidade aos dados em questão.

O Portal da transparência do governo federal apresenta uma série de informações sobre diversos assuntos, porém, a parte objeto desse artigo é apenas a referente às despesas públicas, mais especificamente, os documentos de itens de empenhos, que possuem uma atualização diária de cerca de 25.000 registros. Optou-se por esses documentos por eles serem os que melhor especificam o objeto a ser comprado e por estarem na primeira fase da execução da despesa, o que possibilita o cancelamento ou ajuste de alguma irregularidade constatada na especificação da compra.

A figura 6 apresenta um recorte da tela de especificação de produtos de uma nota de empenho apresentada no Portal da Transparência.

Como observado na figura 6, o empenho pode apresentar vários itens (no caso da figura são apresentados 2 itens). Esses itens possuem informações estruturadas, porém, o campo Descrição (apresentado na figura 6), que indica o que realmente está sendo comprado, é um campo texto que carece de interpretação para avaliação do que está sendo adquirido.

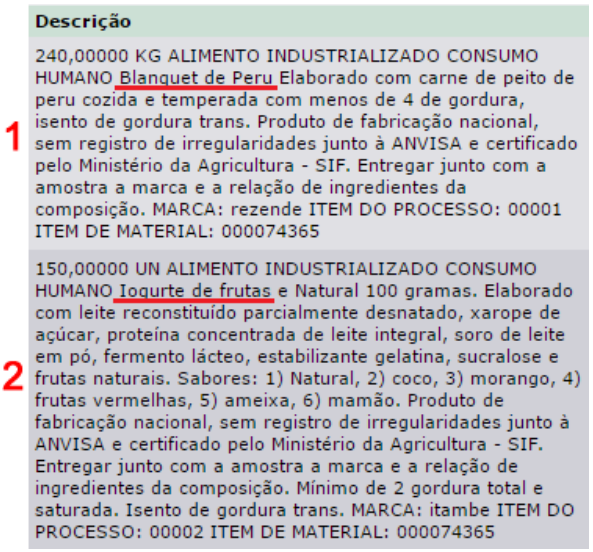


**Figura 6. Descrição de itens de empenho**

O fato dos produtos serem descritos por campos de texto livre faz com que o mesmo tipo de produto seja designado de forma diferente e muitas vezes com unidades de medidas distintas. Em algumas situações, apesar de se utilizar a mesma unidade de medida, essas são escritas de formas diferentes, o que as tornam distintas para mecanismos de comparação automatizados.

Além do campo Descrição do item de empenho, a base de dados também possui outros campos estruturados: Número do empenho (código alfanumérico contendo 23 posições), Número sequencial (campo inteiro que indica a posição de um determinado item dentro do empenho), Código do subitem (campo numérico que identifica a natureza do gasto), quantidade (campo decimal que indica a quantidade do bem que está sendo adquirida) e Valor unitário (campo decimal que traz o preço pago na compra em questão).

Essa base de dados possui dados a partir de maio de 2010, é atualizada diariamente e cresce a uma taxa de aproximadamente 6 milhões de novos registros a cada ano.

O conjunto de dados utilizados no experimento é uma amostra dessa base de dados. Nesse estudo de caso, foram analisados os empenhos relativos ao período de 1 ano (ano de 2014). Os dados em questão correspondiam a um total de 6.251.843 registros e ocupavam um espaço de 7,5 GB.

O conjunto de dados de entrada foi dividido em 12 arquivos (1 referente a cada mês do ano de 2014), sendo que cada linha desses arquivos se referia a um item de compra apresentada no Portal da Transparência. Utilizou-se os dados referentes ao ano de 2014 pelo fato desses serem os dados mais recentes de um ano completo que se tinha disponível na ocasião em que o estudo de caso foi desenvolvido.

A solução proposta foi aplicada para um conjunto limitado de 15 produtos, sendo que novos produtos podem ser incluídos, bastando para isso que sejam definidas suas regras de identificação. Os produtos utilizados no estudo de caso foram: Água Mineral - 20l, Água Mineral - 500 ml, Batata (Kg), Caneta Esferográfica (caixa com 50 unidades), Cebola (Kg), Cenoura (Kg), Diesel (litro), Gasolina (litro), Grampo para Grampeador (caixa com 1000 unidades), Grampo para Grampeador (caixa com 5000 unidades), Laranja (Kg), Notebook (unidade), Papel A4 Branco (resma), Pilha (tamanho D), Presunto (Kg). Optou-se por esse conjunto reduzido de apenas 15 produtos a fim de tornar mais concisa a apresentação dos resultados desse estudo.

## 5. RESULTADOS

O processo analisou o total de 6.251.843 itens de empenho, sendo que 27.087 desses itens foram identificados como sendo um dos produtos pré-estabelecidos. A grande quantidade de empenhos não identificados se dá devido ao fato de apenas 15 produtos terem sido testados e também pelo fato de que muitos desses empenhos não serem referentes à compra de produtos, tratando-se de outros tipos de despesas, como por exemplo, contratação de serviços e pagamento de pessoal. Porém, é importante ressaltar que todos os 6.251.843 itens de empenhos foram processados pela solução proposta.

Ao final do processo, além da identificação do produto especificado em cada uma das descrições das compras, graças às informações extraídas na atividade de pré-processamento, com o emprego das técnicas de recuperação de informação definidas em [10] e [17], também foi possível a identificação de uma série de outras informações relativas ao produto em questão, possibilitando a obtenção de um conjunto de dados estruturados a partir da descrição textual das compras, conforme exemplificado pela figura 7.

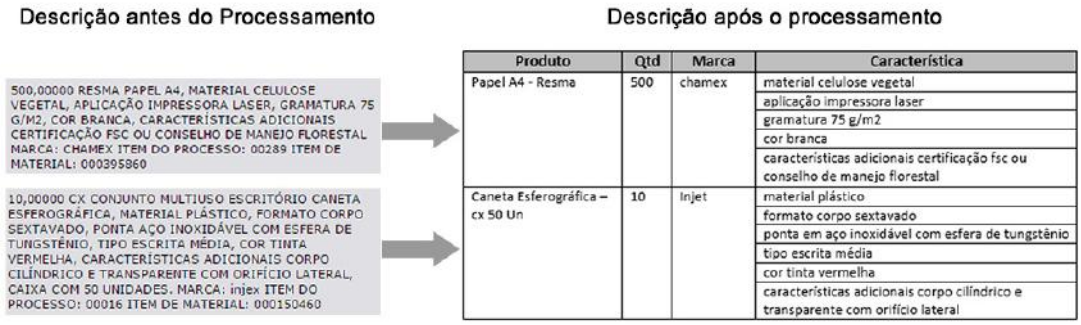


**Figura 7. Estruturação da informação**

Além disso, a aplicação da metodologia também permitiu a obtenção de uma série de métricas que possibilitaram a análise do perfil das compras feitas. A tabela 1 apresenta um quadro resumo dos resultados obtidos ao final do processamento.

**Tabela 1. Quadro resumo dos resultados obtidos**

| Produto | Qtd de Compras | Preço Médio (R$) | Preço Máximo (R$) | Preço Mínimo (R$) | Mediana (R$) |
|---|---|---|---|---|---|
| **Água Mineral – 20l** | 662 | 13,06 | 1203,84 | 0,01 | 6,14 |
| **Água Mineral – 500ml** | 201 | 0,99 | 26,50 | 0,52 | 0,66 |
| **Batata - kg** | 1747 | 2,37 | 6,50 | 0,26 | 2,29 |
| **Caneta esferográfica – cx 50un** | 90 | 17,32 | 39,99 | 1,81 | 15,14 |
| **Cebola – kg** | 2032 | 2,28 | 19,10 | 0,25 | 1,97 |
| **Cenoura – Kg** | 2557 | 2,18 | 18,54 | 025 | 1,97 |
| **Diesel (litro)** | 5881 | 2,71 | 48,68 | 0,52 | 2,57 |
| **Gasolina (litro)** | 5148 | 3,10 | 31,64 | 0,51 | 2,98 |
| **Grampo para Grampeador – cx 1000 un** | 64 | 3,64 | 16,90 | 0,54 | 2,62 |
| **Grampo para Grampeador – cx 5000 un** | 218 | 5,40 | 29,85 | 1,43 | 3,20 |
| **Laranja – Kg** | 2580 | 1,58 | 9,91 | 0,37 | 1,25 |
| **Notebook** | 2119 | 3052,77 | 22980,00 | 553,33 | 2787,00 |
| **Papel A4 Branco – resma** | 753 | 20,68 | 99,99 | 3,15 | 15,80 |
| **Pilha tamanho D - un** | 103 | 5,87 | 16,51 | 0,94 | 5,50 |
| **Presunto - Kg** | 2932 | 13,97 | 38,00 | 0,38 | 13,71 |

A métrica que melhor representa o preço praticado pelo governo é a mediana (última coluna da tabela), pois, diferentemente da média, ela não sofre a influência de possíveis outliers. Esse trabalho não se propõe a analisar a economicidade das compras, visto que, isso implicaria em comparações da mediana dos preços desses produtos com os preços praticados pelo mercado, e não existem fontes oficiais que possam subsidiar tais comparações com o grau de confiança necessário. No entanto, os casos extremos (preços mínimos e máximos) podem caracterizar algum tipo de anomalia e devem ser analisados de forma individualizada, a fim de se comprovar se esses valores realmente estão no portal ou se são frutos de alguma inconsistência nas regras de identificação aplicadas ou na metodologia proposta.

Como exemplo disso, pode-se verificar o caso do preço máximo pago pelo galão de 20 litros de água mineral, apresentado na primeira linha da tabela 1. Conforme identificado pelo processamento dos dados, consta que foi pago um valor de R$ 1.203,84 no galão de água.

Para se verificar se esse valor realmente se refere a uma compra superfaturada ou a algum erro de preenchimento de sistema, são necessárias auditorias específicas a fim de se apurar os fatos ocorridos. Porém, conforme pode ser verificado na figura 8, essa informação está presente no Portal da Transparência e é praticamente impossível se identificar essa disparidade no universo de 30 milhões de documentos publicados no portal sem o auxílio de uma ferramenta de análise e processamento intensivo de dados. Ou seja, a solução proposta é capaz de identificar casos extremos, como o apresentado na figura 8, tornando assim, mais efetiva a transparência que atualmente é exigida pela legislação brasileira.


**Figure 8. Fragmento de uma tela do Portal da Transparência com valor unitário alto na compra de galão de 20 litros de água mineral**

Da mesma forma, também foi verificado um preço mínimo do galão de 20 litros de água mineral bem abaixo do valor esperado. Conforme pode ser visto na primeira linha da tabela 1, foi pago um valor mínimo de R$ 0,01 no galão de água de 20 litros. A figura 9 mostra uma parte da tela do Portal da Transparência que apresenta essa informação.

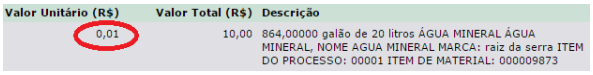


**Figure 9. Fragmento de uma tela do Portal da Transparência com valor unitário baixo na compra de galão de 20 litros de água mineral**

Essa informação de valor unitário extremamente baixo deve ser oriunda de preenchimentos errôneo por parte da pessoa responsável por inserir esses dados no sistema SIAFI[4]. Porém, independentemente de qualquer erro que possa ter ocorrido, essa informação também está no Portal da Transparência e só foi possível identificar essa disparidade pela aplicação da técnica automatizada de análise de dados desenvolvida.

Cabe ressaltar que essa metodologia levou cerca de 10 minutos para analisar textualmente mais de 6 milhões de especificações de compra, o que demonstra a viabilidade incorporação de solução proposta ao processo de carga diário do portal da transparência.

Não foi possível repetir esse mesmo experimento empregando-se técnicas de ETL tradicionais utilizando uma máquina isolada (com 8GB de memória RAM e processador intel Core i7), afim de se comparar os tempos de execução, uma vez que, a referida máquina travava durante o processamento. No entanto, a intensão da metodologia apresentada não é simplesmente propor uma forma de resolver problemas que não eram passíveis de serem tratados com a utilização de técnicas convencionais, mas sim propor uma metodologia que fosse escalável.

Dessa forma, a principal contribuição desse artigo, em relação a outros trabalhos, que de alguma forma também tinham que identificar objetos de compras governamentais, como nos casos dos estudos desenvolvidos por Rommel, Carvalho et al. [13], Rommel, Carvalho et al. [14], Marzagão [15], e Carvalho et al. [5], é a apresentação de uma solução escalável. Ou seja, o aumento expressivo do volume de dados a ser analisado não necessariamente implicará na queda de performance da solução,

[4] Sistema Integrado de Administração Financeira (SIAFI): Sistema Informatizado que processa e controla a execução orçamentária, financeira, patrimonial e contábil da União [11]. Esse sistema é utilizado como fonte de informação para essa parte do portal da Transparência do Governo Federal.

bastando para isso, aumentar-se o número de máquinas do cluster utilizado.

## 6. CONCLUSÃO

Estre trabalho apresentou uma metodologia escalável para aumentar a qualidade das informações dos portais de transparência pública.

A solução proposta tinha como objetivo a identificação dos produtos comprados pela Administração Pública, que são apresentados nos diversos sites de transparência pública. Para isso, utilizou-se os textos que descrevem os itens de empenho que são apresentados nesses portais. O outro objetivo do trabalho foi o cálculo de estatísticas dos preços pagos por tais produtos.

A solução implementada também fez a extração das principais características dos produtos, pela utilização das técnicas de recuperação de informação [10] e [17].

A metodologia sugerida faz uso de técnicas de processamento intensivo de dados com a utilização do paradigma de programação mapreduce [8] e da infraestrutura do sistema de processamento paralelo Hadoop [21].

A proposta apresentada foi testada para um conjunto de 15 produtos com a base de empenhos extraídos do Portal da transparência do Governo Federal. Esse conjunto de dados foi composto por todos os empenhos emitidos no ano de 2014.

Como resultado, obteve-se a relação de compras desses 15 produtos, bem como as principais métricas associadas com os preços unitários de tais produtos. A metodologia proposta também permitiu a identificação de compras superfaturadas que seriam impossíveis de serem identificadas sem a utilização de técnicas de processamento intensivo de dados.

Pode-se concluir que o a utilização de técnicas de processamento intensivo de dados e do paradigma de programação mapreduce são ferramentas promissoras para questões ligadas à área de transparência, que normalmente tratam grandes volumes de dados, mas que nem sempre apresentam informações de qualidade.

Como trabalhos futuros, pretende-se desenvolver técnicas de mineração de texto mais apuradas que permitam a identificação dos produtos sem a necessidade de estabelecimentos de regras prévias de identificação.

## 7. AGRADECIMENTOS

O projeto foi parcialmente financiado pela CAPES-DAAD (processo 4210-15-8).

## 8. REFERÊNCIAS

[1] Agner L. Governo eletrônico e transparência do Estado. Revista Webinsider, Brasília. 2008;

[2] BRASIL. CONSTITUIÇÃO DA REPÚBLICA FEDERATIVA DO BRASIL DE 1988 [Internet]. Constituição Federal, de 5 de outubro de 1988. Recuperado de: http://www.planalto.gov.br/ccivil_03/constituicao/constituicaocompilado.htm

[3] BRASIL. Disponibilização em tempo real de Informações [Internet]. Lei Complementa no 131 de 27 de maio de 2009. Recuperado de: https://www.planalto.gov.br/ccivil_03/leis/lcp/lcp131.htm

[4] BRASIL. Lei 4320 [Internet]. Lei no 4320, de 18 março de 1964. Recuperado de: http://www.planalto.gov.br/ccivil_03/leis/l4320.htm

[5] Carvalho RN, Sales L, Da Rocha HA, Mendes GL. Using Bayesian Networks to Identify and Prevent Split Purchases in Brazil. In: BMA@ UAI. 2014. p. 70–8.

[6] Cloud AEC. Amazon web services. Retrieved November. 2011;9:2011.

[7] Controladoria Geral da União. Portal da Transparência nos Recursos Públicos Federais [Internet]. 2004 [citado 18 de abril de 2015]. Recuperado de: http://transparencia.gov.br/

[8] Dean J, Ghemawat S. Mapreduce: Simplified data processing on large clusters, osdi'04: Sixth symposium on operating system design and implementation, san francisco, ca, december, 2004. S Dill, R Kumar, K McCurley, S Rajagopalan, D Sivakumar, ad A Tomkins, Self-similarity in the Web, Proc VLDB. 2001;

[9] Dean J, Ghemawat S. MapReduce: simplified data processing on large clusters. Communications of the ACM. 2008;51(1):107–13.

[10] Etzioni O, Cafarella M, Downey D, Popescu A-M, Shaked T, Soderland S, et al. Unsupervised named-entity extraction from the web: An experimental study. Artificial intelligence. 2005;165(1):91–134.

[11] Feijó. Curso de SIAFI: uma abordagem prática da execução orçamentária e financeira. 2006.

[12] Gupta R, Gupta H, Mohania M. Cloud computing and big data analytics: what is new from databases perspective? In: Big Data Analytics. Springer; 2012. p. 42–61.

[13] Lin J, Dyer C. Data-intensive text processing with MapReduce. Synthesis Lectures on Human Language Technologies. 2010;3(1):1–177.

[14] Manyika J, Chui M, Bughin J, Dobbs R, Bisson P, Marrs A. McKinsey Global Institute D. 2013;

[15] Marzagão T. Using SVM to pre-classify government purchases. arXiv preprint arXiv:160102680. 2015;

[16] Ministério do Planejamento. Portal de Dados Abertos do Governo Brasileiro [Internet]. 2012 [citado 15 de janeiro de 2016]. Recuperado de: http://dados.gov.br/

[17] Munková D, Munk M, Vozár M. Data pre-processing evaluation for text mining: Transaction/sequence model. Procedia Computer Science. 2013;18:1198–207.

[18] Rommel, Carvalho, de Paiva E, da Rocha H, Mendes G. Methodology for Creating the Brazilian Government Reference Price Database. 2013; Recuperado de: http://www.lbd.dcc.ufmg.br/colecoes/eniac/2013/0033.pdf

[19] Rommel Carvalho, Eduardo de Paiva, Henrique da Rocha, Gilson Mendes. Using Clustering and Text Mining to Create a Reference Price Database. Learning and NonLinear Models. 2014;12:38–52.

[20] Taurion C. Big data. Brasport; 2013.

[21] White T. Hadoop: The definitive guide. O'Reilly Media, Inc.; 2012.