Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

VITOR ROCHA CASTRO PEREIRA

HADOOP E MAPREDUCE NO CONTEXTO DE APLICAÇÕES DE ANÁLISE DE DADOS PARA BIOTECNOLOGIA

Assis/SP
2016

Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

VITOR ROCHA CASTRO PEREIRA

HADOOP E MAPREDUCE NO CONTEXTO DE APLICAÇÕES DE ANÁLISE DE DADOS PARA BIOTECNOLOGIA

Projeto de pesquisa apresentado ao curso de Bacharelado em Ciências da Computação do Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

Orientando: Vitor Rocha Castro Pereira
Orientador: Prof. MSc. Guilherme de Cleva Farto

Assis/SP
2016

FICHA CATALOGRÁFICA

PEREIRA, Vitor Rocha Castro
Hadoop e MapReduce no contexto de aplicações de análise de dados para biotecnologia / Vitor Rocha Castro Pereira. Fundação Educacional do Município de Assis – FEMA – Assis, 2016. 43p.

1.Hadoop **2**. Biotecnologia **3**.Analise de Dados **4**.MapReduce

CDD:

Biblioteca da FEMA

HADOOP E MAPREDUCE NO CONTEXTO DE APLICAÇÕES DE ANÁLISE DE DADOS PARA BIOTECNOLOGIA

VITOR ROCHA CASTRO PEREIRA

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Graduação, avaliado pela seguinte comissão examinadora:

Orientador: Prof. MSc. Guilherme de Cleva Farto
Inserir aqui o nome do orientador

Examinador: Prof. Dr. Luiz Carlos Begosso
Inserir aqui o nome do examinador

Assis/SP

2016

DEDICATÓRIA

Dedico este trabalho em especial a
minha família, pai, mãe, irmão, minha
namorada na
qual foram pessoas que me deram forças e
lutaram junto comigo
não deixando que eu nunca desistisse
para que este sonho
se tornasse realidade.

AGRADECIMENTOS

Primeiramente a Deus, por me dar forças e foco para estudar e me dedicar por esses quatro anos de curso, mesmo nos momentos difíceis não me deixar desistir e sempre seguir em frente superando todas as dificuldades colocadas em meu caminho.

Ao meu pai, mãe, irmão por sempre me darem auxilio nos momentos de dúvidas, a minha namorada por me dar forças e sempre me incentivar a melhorar.

Aos professores (Leonor, Marisa, Diomara, Almir, Guilherme, Fábio, Cleiton, Douglas, Fernando, Alex, Begosso, Talo, Osmar e Domingos) que contribuíram de alguma maneira para o meu crescimento pessoal e profissional.

Em especial ao meu orientador prof. Guilherme, que me inspirou e auxiliou sempre que necessitei, tirando minhas dúvidas e sempre me mostrando o melhor caminho a seguir.

Pelos amigos de classe, que sempre me auxiliaram quando precisei e dando todo o apoio uns aos outros para que todos tivessem o melhor desempenho possível nesses quatro anos tirando dúvidas e ajudado uns aos outros.

Aos meus amigos que tiveram do meu lado em todos os momentos e a todos que colaboraram direta ou indiretamente na execução deste trabalho.

Muito obrigado.

RESUMO

A bioinformática é uma área que vem tendo uma grande evolução nos últimos anos. A quantidade de dados armazenados em um banco de dados genômico disponibilizado na Web, cresce cada vez mais conforme os cientistas avançam nas pesquisas a respeito do mapeamento de códigos genéticos. Ferramentas são propostas e desenvolvidas para realizar buscas nas bases de dados genéticos, com o objetivo de realizar comparações entre sequências para definir se há heranças ou semelhanças quando comparadas a outros tipos de organismos.

O objetivo desta pesquisa é o de investigar a adoção de conceitos de Hadoop e MapReduce junto ao repositório GenBank, fornecendo uma nova abordagem para a busca e análise de padrões em sequências genéticas disponibilizadas na Web. Dessa forma, espera-se experimentar os recursos das tecnologias adotadas, Hadoop e MapReduce, na busca e identificação de padrões nas sequências.

Esta pesquisa foi feita em artigos científicos sites e blogs relacionados a plataforma Hadoop e ao GenBank para poder obter-se informações de como integrar esta plataforma ao ambiente de Biotecnologia e realizar pesquisas nas bases de dados retornando somente as bases que possuem uma determinada cadeia genética.

Palavras-chave: Hadoop, MapReduce, GenBank, sequências genéticas, padrões genéticos.

ABSTRACT

Bioinformatics is an area that has had a great development in recent years. The amount of data stored in a genomic database available on the Web increasingly growing as scientists advance in research on mapping genetic codes. Tools are proposed and developed for performing searches in genetic databases in order to perform sequence comparisons to determine whether there inheritance or similarity when compared to other types of organisms.

The objective of this research is to investigate the adoption of concepts of Hadoop and MapReduce with the GenBank repository, providing a new approach to the search and analysis of patterns in genetic sequences available on the Web. Thus, we expect to experience the features of technologies adopted, Hadoop and MapReduce, in the search and identification of patterns in the sequences.

This research was done in scientific articles sites and blogs related to Hadoop platform and GenBank in order to obtain information on how to integrate this platform to the Biotechnology environment and conduct research in the databases returning only the foundations that have a particular genetic chain.

Keywords: Hadoop, MapReduce, GenBank, genetic sequences, patterns genes.

LISTA DE ILUSTRAÇÕES

SUMÁRIO

1. INTRODUÇÃO

A biotecnologia surgiu, em meados de 1800 a.C, junto com a fabricação de vinhos, pães, queijos, adotando-se a técnica da fermentação dos alimentos. Entretanto o termo (biotecnologia) surgiu na década de 70 com o engenheiro húngaro Karl Ereky que desenvolvera uma pesquisa focada na produção em larga escala de suínos alimentados com beterrabas cultivadas com microrganismos. Atualmente, pesquisadores e empresas investigam e utilizam biotecnologia como estratégia para a criação e/ou refinamento de produtos que contribuem com a melhora da qualidade de vida. Para isso, macro e microrganismos têm sido estudados (COELHO, 2009).

Um tema que auxilia a biotecnologia é a bioinformática. Com origem na década de 90, os sequênciadores automáticos de DNA (ácido desoxirribonucleico), contribuíram com a evolução das pesquisas acerca do mapeamento genético envolvendo distintas áreas como, por exemplo, engenharia de software, matemática, estatística, ciência da computação e biologia molecular. Um dos projetos que relacionam biotecnologia e bioinformática é a geração de soluções para problemas de poluição dos rios, mares e ar. Outro exemplo é a adoção de biotecnologia em de embriões, objetivando melhorar a qualidade do gado a partir do refinamento genético de gerar animais para alcançar uma melhor qualidade de carne. Na agricultura, pesquisas no código genético de plantas auxiliam na modificação da estrutura genética para fomentar um crescimento mais saudável e, consequentemente, uma safra mais segura (JUNIOR, 2011).

Segundo Mount (2001), a estrutura do DNA foi descoberta pelo cientista norte-americano James Watson e pelo britânico Francis Crick em março de 1953. A célula padrão de DNA é composta de quatro partes denominadas nucleotídeos. Assim DNA é uma longa fileira composta por blocos que compõem o "alfabeto" do DNA:

- **Adenina (A)**;
- **Citosina (C)**;
- **Guanina (G)**;
- **Timina (T)**.

O mapeamento do genoma humano pode ser utilizado para a prevenção de doenças e, apesar de ser um processo caro, cerca de 5 mil dólares, 80 mil genes que se estimam

existir no DNA humano podem ser mapeados, combinando com uma sequência de 3 bilhões de bases químicas.

Com o crescimento de dados neste contexto, a computação é utilizada para armazenar os dados de forma simples e eficiente, simplificando a consulta e manipulação de dados por meio de ferramentas. Um exemplo disto é o modelo computacional *MapReduce* utilizado para realizar buscas e operações por meio de padrões.

Em 2001, surgiu o termo BigData que, comumente, relaciona-se a três *Vs*: volume, velocidade e variedade. O BigData refere-se à capacidade de armazenar uma grande quantidade de dados e aos recursos para manipulação desse extenso repositório de dados. Com o crescimento das atividades e operações que resultam em grandes volumes de dados, torna-se necessário o uso de uma ferramenta responsável pela análise dos dados. Dessa forma, o conceito de *analytics* possibilita a manipulação de tais dados com o objetivo de investigar como algo aconteceu e o porquê. Dessa forma, padrões podem ser identificados, auxiliando na tomada de decisões futuras (Portal SAS, 2015).

1.1 OBJETIVOS

O objetivo deste trabalho é o de investigar os conceitos de biotecnologia para a análise de código genético disponibilizado em arquivos da plataforma GenBank, focando-se nas buscas de padrões genéticos por meio de ferramentas de *analytics*. Para isso, pretende-se adotar a ferramenta Hadoop e a abordagem de *MapReduce* para apoiar a consulta por padrões genéticos de maneira eficiente, ou seja, a partir de algoritmos específicos da análise de padrões.

1.1.1 OBJETIVOS ESPECÍFICOS

Pretende-se, de maneira mais específica, propor e experimentar uma abordagem para a adoção de Hadoop e *MapReduce* na busca de padrões em arquivos que armazenam estruturas de bases genéticas. Além disso, espera-se analisar os resultados obtidos por meio dessa estratégia de busca genética, e documentar as lições aprendidas.

1.2 JUSTIFICATIVA

A biotecnologia e bioinformática têm influenciado, cada vez mais, o cotidiano das pessoas. Por meio do aprimoramento genético, pode-se melhorar a qualidade de animais tornando-os mais saudáveis. Na agropecuária, os melhoramentos de sementes contribuem para que os agricultores obtenham uma produção melhor e mais segura, resultando em plantas resistentes a pragas. A adoção de Hadoop e *MapReduce* pode tornar o processo de análise genética mais eficiente e rápida. Atualmente, a plataforma Hadoop e a abordagem de *MapReduce* simplificam as pesquisas em larga escala.

1.3 MOTIVAÇÃO

Realizar um experimento com a plataforma Hadoop para realizar buscas em bases de dados genéticos que crescem cada vez mais, com uma ferramenta que possui capacidade para processar grandes bases de dados, assim integrando esta ferramenta de BigData em uma estrutura de Biotecnologia.

1.4 ESTRUTURA DO TRABALHO

Este trabalho está estruturado da seguinte forma:

- **Capítulo 1 – Introdução:** Neste capítulo, apresenta-se uma breve explicação acerca de biotecnologia, contextualizando os conceitos a serem explorados neste trabalho.
- **Capítulo 2 – Fundamentos da biotecnologia:** Neste capítulo, serão apresentados os conceitos da biotecnologia, explorando seus fundamentos e os desafios relacionados.
- **Capítulo 3 – Análise de dados em biotecnologia:** Neste capítulo, pretende-se descrever como os dados podem ser utilizados por, por ferramentas de manipulação e extração de dados.
- **Capítulo 4 – Plataforma Apache Hadoop:** Neste capítulo, serão descritos os conceitos da plataforma Hadoop e da abordagem de *MapReduce*, bem como a definição estrutural e o funcionamento

- **Capítulo 5 – Proposta e desenvolvimento do trabalho:** Neste capítulo, será apresentada uma definição de um cenário experimental para investigar e avaliar a adoção de Hadoop e *MapReduce* no contexto de biotecnologia junto ao GenBank.
- **Capítulo 6 – Conclusão:** Por fim, serão apresentadas as considerações finais, revisitando os resultados deste trabalho, bem como relatadas as sugestões de pesquisas futuras

2. FUNDAMENTOS DA BIOTECNOLOGIA

A biotecnologia é um estudo da biologia, juntamente com a tecnologia para pesquisas na área da agricultura, ciência dos alimentos, medicina, pecuária, entre outros. De acordo com a convenção da ONU Biotecnologia é qualquer aplicação tecnológica que use sistemas biológicos, organismos vivos, ou seus derivados, para fabricar ou alterar produtos ou processos para utilização específica (OLIVEIRA, 2002).

Processos e métodos na agricultura adotam ciências mecânicas e biológicas para que os agricultores possam escolher os melhores métodos de plantio com o objetivo de alcançar maiores e melhores rendimentos. Outros usos da biotecnologia são definidos quando as fazendas se tornarem cada vez maiores e com dificuldades de se manter, pois cada vez mais surgem pragas, doenças e até mesmo o clima pode dificultar o plantio, e a evolução de determinadas culturas. Conforme a agricultura se desenvolve, novas técnicas e métodos de refinamento genético das plantas também se tornam necessárias, resultando na evolução e aumento do uso de biotecnologia aplicada à agroindústria.

No início do século XX, cientistas começaram a utilizar a biotecnologia no cruzamento e refinamento de animais, pois com a modificação de seus genes, animais podem apresentar as características desejas e relevantes (ARAUJO, 2008)

2.1 DOGMA CENTRAL: DNA, RNA E PROTEÍNA

A biologia molecular se desenvolveu em um amplo campo de pesquisa, tornando-se um componente básico a outras ciências de pesquisas básicas, conduzindo uma rápida expansão do conhecimento nesta área. O dogma central define o paradigma da biologia molecular como sendo que a informação genética está contida em uma sequência de ácidos nucléicos e que os genes funcionam por sua expressão na forma de moléculas de proteínas (OLIVEIRA, 2002).

O gene dos seres vivos se compõe, na grande maioria, por estruturas de DNA e o RNA (ácido ribonucleico) que é o material genético. A junção de várias unidades de ácidos nucleicos define uma estrutura formando unidades. No caso do DNA, essas estruturas possuem elementos distintos denominados nucleotídeos. Os nucleotídeos naturais são adenina (A), timina (T), guanina (G) e citosina (C). O DNA tem uma estrutura de formato

helicoidal de fita dupla vinculadas por nucleotídeos opostos (adenina oposto a timina, guanina oposta a citosina), definindo-se os modelos hereditários dos seres vivos e toda a informação que dá sequência dos aminoácidos codificada pelo sequênciamento de nucleotídeos (OLIVEIRA, 2002).

Na Figura 1 é ilustrada uma representação da estrutura do DNA em formato helicoidal com os nucleotídeos A, T, G e C.

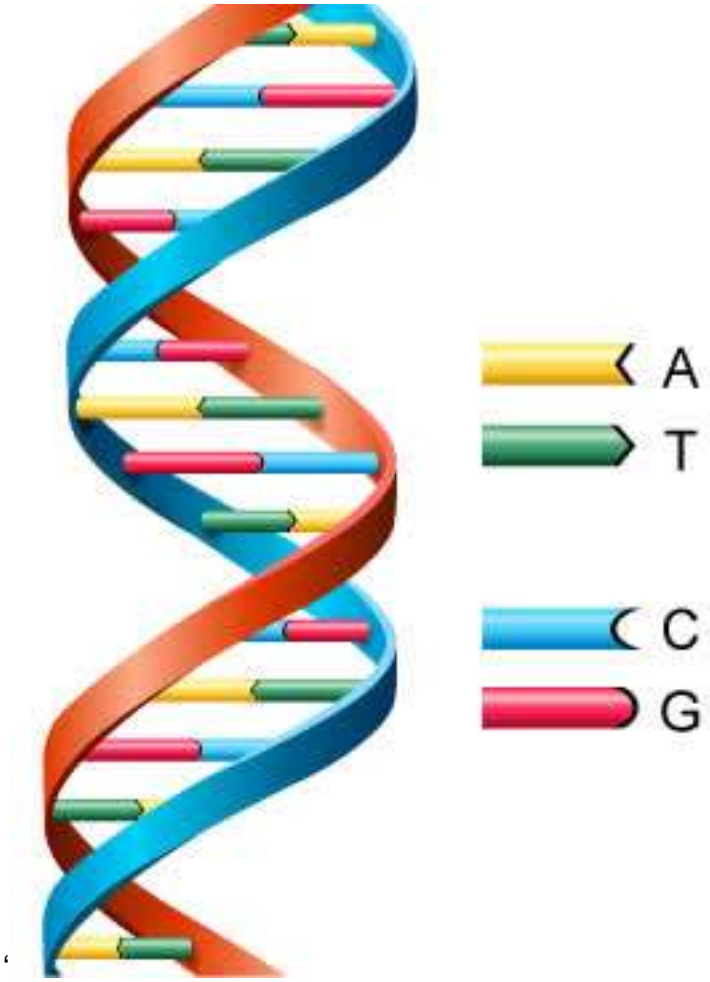

Figura 1 – Estrutura DNA em formato helicoidal

(In: UOL, 2016)

Em todo indivíduo são as proteínas que regulam seu metabolismo, sendo sua estrutura semelhante ao DNA, pois ambos possuem estruturas similares. No caso, as estruturas são chamadas de polipeptídios. Os polipeptídios são definidos por vários aminoácidos sequêncialmente ligados. O que ressalta a importância do DNA na participação das proteínas de um indivíduo são os processos de tradução e transcrição, etapas que

definem a Síntese Proteica. O DNA codifica sua estrutura em uma síntese de aminoácidos, normalmente peptídeos, polipeptídios e proteínas, tento papel fundamental na definição de cada indivíduo (OLIVEIRA, 2002).

O dogma central da biologia molecular foi definido por Francis Crick em 1958. Tal dogma demonstra como é o segmento de informação do código genético. O modelo define que uma sequência de um ácido nucleico pode formar uma proteína, porém o contrário não é possível. Dessa forma, o fluxo da informação genética apresenta o seguinte sentido: DNA → RNA→ PROTEÍNAS. Na Figura 2, pode-se observar todos os processos pelos quais os ácidos nucleotídeos podem passar (OLIVEIRA, 2002).

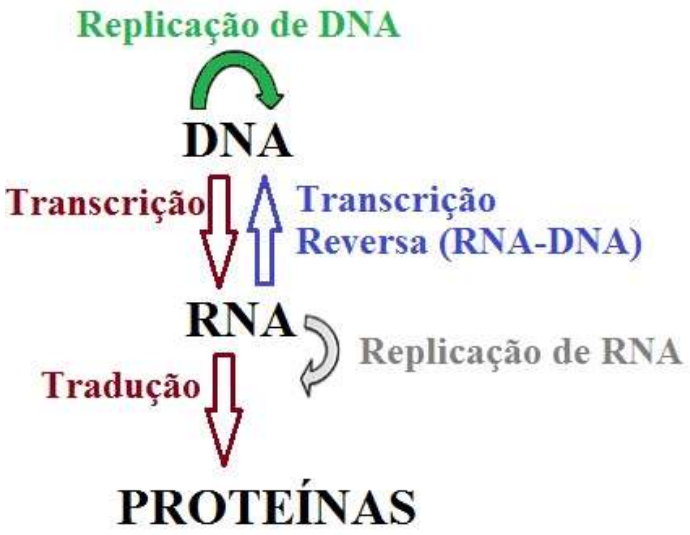

Figura 2 - Processos de passagem dos ácidos nucleotídeos
(In: Mundo Educação, 2016)

Observa-se que o DNA contém a informação genética e, por meio do processo de transcrição, tais informações são transformadas em moléculas de RNA. Dessa forma, a molécula de DNA serve como base para a criação de uma molécula de RNA. Na molécula de RNA, encontram-se os códigos utilizados para organizar a sequência de aminoácidos e formar as proteínas no processo de tradução. Códon corresponde a uma sequência de três bases nitrogenadas consecutivas do RNA mensageiro, que funciona como uma unidade genética de codificação, especificando um aminoácido particular durante a síntese de proteína de uma célula. O processo de tradução é definido na união de aminoácidos, seguindo a ordem de códons de um RNA mensageiro. Pode-se verificar que a replicação do DNA é, um processo em que a molécula de DNA pode formar outra idêntica à original (OLIVEIRA, 2002).

2.2 GENOMA E ORGANIZAÇÃO: EUCARIOTOS E PROCARIOTOS

Os genomas eucarioto ou procarioto tem como utilidade servir como repositório replicativo da informação codificada do DNA que o forma. Contudo, as características dos genomas procarióticos são mais simples que os eucarióticos (ARAUJO, 2008).

Um genoma procarioto normalmente é formado por vírus e bactérias, onde eles são capazes de se autoduplicar, sua estrutura dedica-se a codificação de proteínas quase que exclusivamente, sendo que sua organização genética lhe permite que uma mesma sequência codifique duas proteínas diferentes. Por fora tem uma parede de composição química rígida que protege seu citoplasma em seu interior, esse citoplasma contém ribosomas onde ocorre a síntese proteica, em sua membrana tem o material genético, formado por ácido desoxirribonucleico (DNA), que é muito compactado e tem uma membrana que delimita um núcleo. Células procarióticas possuem uma organização simples. (OLIVEIRA, 2002).

Células eucariontes, são mais complexas que os procariontes, com núcleo organizado em carioteca, nucleoplasma, cromatina e nucléolo, além do citoplasma com organelas organizadas com sistemas de membranas, como complexo de Golgi, retículo endoplasmático, mitocôndria, cloroplasto, entre outras. Não se sabe como essas células na era primitiva, mas sem dúvida, um trecho fundamental de sua evolução foi a aquisição de seu citoplasma as mitocôndrias e cloroplastos. Acredita-se que a célula “primitiva” tivesse sido bem pequena para que se adequasse melhor a relação tamanho e funcionamento fazendo com que fosse necessário que ela crescesse, fazendo com que evoluísse durante milhões de anos. Atualmente, existem células eucariontes individuais que formam seres unicelulares, como algas e protozoários. Estas células também formam seres unicelulares que são os animais e plantas (ARAUJO, 2008).

2.3 ESTRUTURA PRIMÁRIA, SECUNDÁRIA E TERCIÁRIA DA PROTEÍNA

A estrutura primaria é formada por uma sequência de aminoácidos, composta por ligações peptídicas. Uma proteína é formada por um número variado de aminoácidos. Caso elas tenham somente sua estrutura primária, elas seriam compostas de longas moléculas, de forma aleatória (SOUZA, 2016).

A estrutura secundária foi definida por meio de cadeias peptídicas não são esticadas, porém torcidas, dobradas ou enroladas em si mesmas. Isto demonstra que as proteínas possuem uma estrutura espacial e tridimensional. Por meio de pesquisas acerca destas informações, informações Linus Pauling e Robert determinaram as confirmações acrescentadas aos conhecimentos sobre as ligações peptídicas, serem capaz de serem assumidas pelas moléculas proteicas. No arranjo mais simples definido, a cadeia polipeptídica enrolava sobre si mesma, formando uma espiral, esse arranjo ficou conhecido como alfa-hélice (SOUZA, 2016).

A estrutura terciária define-se a estrutura tridimensional total das proteínas e não é totalmente rígida, podendo ser formada por mais de um tipo de estrutura secundária. A diferença entre a estrutura secundária e terciária é que a secundária é formada pelas pontes de hidrogênio estabelecidas entre aminoácidos juntos entre si, enquanto que terciária é formada pelas ligações dos grupos laterais (SOUZA, 2016).

2.4 ALINHAMENTO DE SEQUÊNCIAS

O alinhamento de sequências é um método para organizar sequências primárias de DNA, de RNA ou de proteína para identificar locais similares que podem ser consequência de relações funcionais, estruturais ou evolucionárias entre elas. As sequências de nucleotídeos ou resíduos de aminoácidos quando alinhados são representados em forma de uma linha de uma matriz. Um espaço (*gaps*) pode ser inserido entre os resíduos para que caracteres semelhantes definidos por algum método, sejam alinhados em colunas sucessivas (MEIDANIS, 1997).

Há duas categorias principais para o alinhamento de sequências, sendo estas classificadas em alinhamentos globais e alinhamentos locais. O alinhamento global é definido como uma forma de otimização para forçar o alcance de todo o comprimento de todas as sequências verificadas. Já os alinhamentos locais procuram regiões de similaridade, nas sequências longas que são normalmente afastados. Normalmente alinhamentos locais são mais razoáveis de se utilizar, mas tendo dificuldade de determinar devido ao problema adicional de identificar regiões de similaridade. O alinhamento de sequências é também adotado para definir como é a estrutura de uma doença e dessa forma identificar o método mais eficiente para combatê-la (MEIDANIS, 1997).

2.4.1 Busca de similaridade

O algoritmo para buscas de similaridade é baseado nos conceitos de programação dinâmica. A dificuldade de se utilizar tal algoritmo se deve a sua complexidade, pois pode se tornar inexecutável dependendo de sua utilização. Uma maneira de exemplificar, isto é, por meio de aplicações que realizam buscas em bases de dados moleculares com milhões de sequências genéticas. Outro problema no algoritmo de buscas de similaridade é quando se torna necessário realizar a comparação de uma nova sequência com todas as outras depositadas na base de dados, tendo em vista as milhares de comparações que deve ser constantemente realizadas (MEIDANIS, 1997).

As bases de dados moleculares vêm crescendo constantemente, pois os métodos de sequênciamento de proteínas e nucleotídeos tem melhorado muito, tornando cada vez mais importante a concepção de aplicações que executam consultas nestes repositórios genéticos. Algumas das famílias de algoritmos utilizados nestas buscas são chamadas de FAST e BLAST. Tais famílias fornecem uma melhora significativa nos tempos de respostas nas buscas em bases de sequência genética, pois se baseiam em heurística. A família FAST e BLAST adotam o método de similaridade local.

2.4.2 Técnicas

2.4.2.1 Local

No alinhamento local é irrelevante a posição em que a sequência se inicia nas cadeias e, dessa forma, busca-se somente à similaridade entre elas. Neste tipo de alinhamento a definição de aproximação é relevante e semelhanças resultam em pontos para se obter o valor da aproximação, também conhecida como *score*. Esses alinhamentos devem possuir valores altos quando comparando regiões diferentes da sequência por meio de fragmentos da mesma. Tal técnica de alinhamento descarta sequências com pontuação baixa e as que possuírem um valor alto serão alinhadas para se chegar a uma com valor máximo. O alinhamento local não é o alinhamento final, ou seja, somente procura a sequência com maior tamanho significativo em comum. Na figura 3 é ilustrada a execução da técnica de alinhamento local (CARAZZOLLE, 2016).

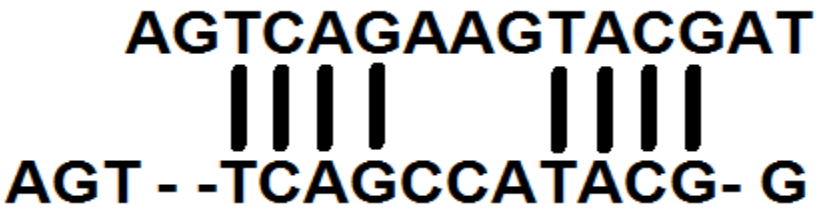

Figura 3 – Alinhamento Local

É realizada uma busca comparando as sequências por meio de uma matriz de alinhamento local. Para isso, as primeiras colunas começam com zero e para qualquer entrada (i, j) há sempre o alinhamento entre os sufixos vazios de s[1...i] e de t [1...j], que tem *score* zero. Portanto, está matriz de alinhamento terá todas as entradas maiores ou iguais a zero. Na Figura 4 é ilustrada uma matriz de um alinhamento local.

	ε	C	A	G	C	A	C	T	C	A	T
ε	0	0	0	0	0	0	0	0	0	0	0
T	0	0	0	0	0	0	0	1	0	0	1
C	0	1	0	0	1	0	1	0	2	0	0
C	0	1	0	0	1	0	1	0	1	1	0
A	0	0	2	0	0	2	0	0	0	2	0
G	0	0	0	3	1	0	1	0	0	0	1
C	0	1	0	1	4	2	1	0	1	0	0
T	0	0	0	0	2	3	1	2	0	0	1
C	0	1	0	0	1	1	4	2	3	1	0
G	0	0	0	1	0	0	2	3	1	2	0

Figura 4 – Matriz de um alinhamento local

O *score* de um alinhamento local considerando como ótimo será a maior entrada em toda a matriz de um alinhamento (CARAZZOLLE, 2016).

2.4.2.2 Global

O alinhamento global realiza buscas por meio de toda a extensão das sequências comparadas, considerando que as sequências devem ser nas mesmas posições das sequências. Na Figura 5 é ilustrada a execução da técnica de alinhamento local.

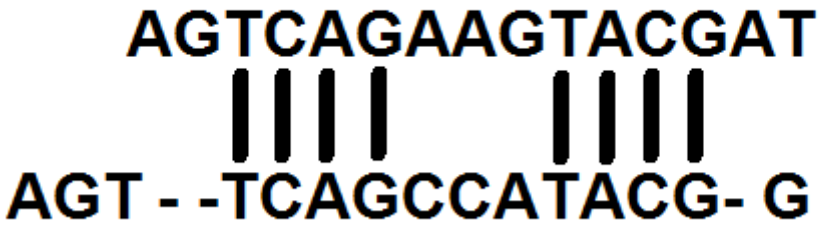

Figura 5 – Alinhamento Global

Caso duas sequências em um alinhamento possuam um ancestral comum, as discordâncias das sequências podem ser vistas como mutações sofridas, espaços (*gaps*) como inserções ou deleções introduzidas em uma ou ambas as sequências (CARAZZOLLE, 2016).

2.4.2.3 Semiglobal

O alinhamento semiglobal é parecido com o alinhamento global, porém em suas buscas por similaridade, a técnica ignora espaços nos extremos e realiza uma busca por sequência de caracteres. Na Figura 6 é ilustrada a execução da técnica de alinhamento semiglobal (CARAZZOLLE, 2016).

Figura 6 – Alinhamento semiglobal

Na Figura 6 uma cadeia com similaridades realizada pelo alinhamento global tendo um *score* bem baixo. Desta forma, pode-se observar como o alinhamento semiglobal realiza uma busca mais profunda nas sequências.

Como demonstrado na figura 7 o alinhamento semiglobal pode realizar buscas, particionando as sequências de modo que em seu meio para realizar uma comparação mais profunda, obtendo-se um maior *score* possível entre as sequências (CARAZZOLLE, 2016).

Figura 7 – Alinhamento semiglobal

2.4.2.4 Múltiplo

Um programa que realiza um alinhamento múltiplo é o ClustalW que se baseia em três passos essenciais: *Pairwise Alignment*, construção de arvores de guia e alinhamento progressivo. Esse programa executa um algoritmo heurístico e uma estratégia de alinhamento progressivo, para se realizar um alinhamento múltiplo (CALHAU, 2008).

Este programa permite a criação de árvores classificativas, retirando das sequências que forem alinhadas. O modo como as sequências são colocadas no alinhamento múltiplo é definida pela estratégia do ClustalW, fazendo com que todos os pares de sequência sejam comparados entre si, sendo agrupados com as sequências similares, ou seja, que tem maior *score* no alinhamento simples. Após a comparação as sequências são adicionadas ao alinhamento múltiplo seguindo a estrutura da árvore guia, das folhas para a raiz. Sequências que possuírem um maior *score* de igualdade são alinhadas primeiro. Seguindo das sequências que tem um *score* menor e no após isso é realizado o alinhamento das sequências resultantes (CALHAU, 2008).

3. ANÁLISE DE DADOS EM BIOTECNOLOGIA

Com os avanços e as descobertas geradas pelas pesquisas para decifrar e mapear o material genético, em uma grande variedade de organismos procarióticos e eucarióticos, uma enorme quantidade de dados.

Atualmente, a contribuição da informática para o desenvolvimento genômico pode ser definido pela sua rapidez, pela sua otimização nos processos e na decodificação de genes. Um exemplo é que uma nova sequência de DNA com 12 mil bases pode ser decifrada em poucos minutos, enquanto há anos atrás essa tarefa poderia levar a média de um ano (PROSDOCIMI, 2004).

Um organismo vivo possui cerca de 50 mil diferentes tipos de proteínas que possuem uma conformação especifica para poder exercer corretamente sua função biológica. A análise de dados em projetos de bioinformática são chave para que eles obtenham sucesso. A análise de dados computacionais de dados torna-se uma parte importante para viabilizar a retirada de informações pertinentes e possibilitar a obtenção do genoma completo ou parcial de um organismo. Dessa forma, a relação da motivação científica e tecnológica ficou muito forte entre a descoberta do DNA, os estudos de transgenia, a materialidade do conhecimento propiciada pela biologia molecular e pela genômica, e o desenvolvimento da bioinformática (PROSDOCIMI, 2004).

3.1. TECNOLOGIAS E FERRAMENTAS DE APOIO

Algumas tecnologias surgiram a partir da necessidade das empresas de que demandaram por ferramentas úteis e com baixo investimento. Dessa forma, a comunidade de desenvolvimento vem aprimorando versões do Linux para distintas utilidades.

Em 2002, o conselho do *Natural Environment Research Council* (NERC), em apoio do seu programa de Genômica Ambiental, estabeleceu a *Environmental Bioinformatics Centre* (NEBC). Como parte de seu mandato, a NEBC desenvolveu e implantou NEBC Bio-Linux. A comunidade inclui desenvolvedores de software, administradores de sistemas e pesquisadores biológicos (PROSDOCIMI, 2004).

Outra plataforma deste tipo é o DNALinux, para distintas atividades nas áreas de bioinformática e biotecnologia. Uma distribuição Linux que fornece ao usuário programas e recursos para distintas atividades nas áreas de bioinformática e biotecnologia.

O GenBank é um banco de dados de nucleotídeos do NLM/NCBI, localizado no *National Institutes of Health* (NIH), que armazena informações sobre sequências nucleotídicas de aproximadamente 260.000 espécies. O GenBank faz parte de um grupo de repositórios genéticos que incluem o *European Molecular Biology Laboratory* (EMBL) e o DNA *DataBank of Japan* (DDBJ). Os três bancos de dados formam a International *Nucleotide Sequence Database Collaboration* (INSDC), compartilhando suas informações para reunir as sequências nucleotídicas e assegurar que possam ser acessadas por pesquisadores do mundo todo. As três bases de dados de nucleotídeos estão em constante atualização, portanto sequências encontradas em um repositório também podem ser encontradas em outros bancos (PROSDOCIMI, 2004).

Proteínas retiradas de um GenBank vem em formato XML, tendo seu conteúdo conforme ilustra a figura 8, tendo não apenas sua estrutura proteica, mas também possuindo todas suas características para a determinar a origem da célula (PROSDOCIMI, 2004).

```
BASE COUNT      518 a    307 c    404 g    470 t
ORIGIN
        1 acttttaatt tgaagtacag tgaagataat caaagatgaa gtttttgatt ggatttggag
       61 tagttctact cctggtagct gttgctgcgc agtacgatcc gcattatctt gacaagagaa
      121 ctgtaatggt tcatttattt gaatggaaat ggaacgacat tgccaaggag tgtcaggatt
      181 atctaggacc aaatgggttt ggtggagttc aggtttcacc ggttaatgag aattgggttt
      241 ctggagaacg tgcctggttc gaacgctacc aaccaatatc atacaaacta accacaagat
      301 ctggtaatga ggacgaattt gccgcaatgg taaaaacctg ccgacaaaat ggcgttcgaa
      361 tcttcgttga cgttgtggtc aaccacatgg cttcaggtgc tcttgaagac acaatttatg
      421 gaacaggtgg atctgaggca catcctggtc cttttgacta tccagctgtt ccgtatgaaa
      481 agaatgactt tcatcctgat tgtagcatct cagactatca ggatgtgtat caagttagaa
      541 attgtcagtt gtctagcttg agggatctca atcaaactat tccctatgtc agagagaaga
      601 tcctagactt cctcaatcat ttagtagact tgggagtagc gggcttcaga atagacgcag
      661 ccaagcatat ggatccgaag gacttgcgat acatctacaa tcatataaag aaattaaata
      721 aggacgctga gtttaaggca ggggacaaag catttatttt ccaggaagtt atagatcttg
      781 gaggagaagc tgtatcatca cgtgaataca tatcgctggg cgttgtaaca gagttcaagg
      841 catccgatga tcttggaaag cttttccggg gacaagttgc tctgtcgacc ttggaaagat
      901 ggggtccaca atatgggctt ctgccttcaa accgagctct tgcctttgtt gagaatcacg
      961 acaacgaacg tgggcacgga gctggcggaa caaatatcct aacgtacaag gatgggaaaa
     1021 tctacacaat ggccgtagta tttaacctag cacactccta tggggttcca cgaatgatga
     1081 gcagctatga attcaacgat ccaagccagg gacctccaca tgatgacaat agcaacattc
     1141 taactcctga attctctgca gatggtaatt cctgcactaa tggttgggtt tgtcagcatc
     1201 gttggcgtcc tatgagaaat atggtgaagt tccggaatat cgtcggtcgg aagtcagtta
     1261 ggaagtggta tgacaacgga agcaatcaga tagccttctc cagaggcaat cagggctttg
     1321 tggcctttaa cttggacatt gttgacttta accagcaagt tccaacggat ttaccagacg
     1381 gggtatattg cgacgttatt tcaggtgaga agaatggcaa tgagtgtact ggaaaggtag
     1441 ttattgtgag caagaggaag gcagctgtta tcctaagggc agatgacaac tacggagtac
     1501 ttgcaattca ttcagagtct aaattgtaag aattatgctg aaatgtttgc tccaacaatt
     1561 caatggcttt ggtgcttgga aggttaagaa gaaacttttt tgtttgagat agtttcagct
     1621 aacctatttc tattattttt ttattttaat aaagaagata acttgctaca gaaaaaaaaa
     1681 aaaaaaaaaa aaaaaaaaa
//

Revised: July 5, 2002.
```

Figura 8: Exemplo de proteína de um GenBank

(In: Universidade Federal de Pernambuco, 2016)

3.2 MANIPULAÇÃO E EXTRAÇÃO DE DADOS

O GenBank é de livre acesso por toda comunidade científica, não havendo qualquer tipo de restrição. Para utilizar os recursos da plataforma GenBank não é necessário cadastro ou solicitação de acesso. Assim como não há um limite de tempo para acesso, uso, cópia e até mesmo distribuição de informações depositadas por pessoas ou grupos que alegam patente. Versões do GenBank são atualizadas a cada dois meses e é permitido aos usuários o acesso gratuito e a possibilidade de se efetuar o download de parte do banco (GENBIO, 2016).

O formato GenBank possui muitos detalhes de armazenando, tendo mais informações do que apenas a sequência nucleotídica. O GenBank fornece informações sobre anotações biológicas e bibliográficas como referência da sequência depositada (GENBIO, 2016).

4. PLATAFORMA APACHE HADOOP

O Apache Hadoop é uma plataforma de software em Java que adota a computação distribuída, voltada para clusters e processamento de grande quantidade de dados. É baseado no *MapReduce* e no GoogleFS (GFS) atualmente mantido pela Apache (BERNARDES, 2014).

A plataforma Hadoop surgiu com o intuito de analisar, manipular e extrair informações de repositórios com grandes volumes de dados gerados todos os dias por equipamentos eletrônicos, redes sociais, aparelhos GPS, servidores, computador e distintos dispositivos computacionais. A análise desta grande quantidade de dados também se relaciona aos conceitos de BigData. O Hadoop é uma ferramenta para análise de dados em larga escala, mesmo com base em dados não estruturados.

O sistema de arquivos distribuídos é formado por um conjunto de máquinas, também chamadas de nós no contexto de *clusters* computacionais, que gerenciam o armazenamento de um ou mais por meio da rede (WHITE, 2012). Isto permite aos programas armazenar e acessar arquivos remotos como se fossem locais (COULORIS, DOLLIMORE e KINDBERG, 2007).

Uma das dificuldades é a existência de problemas na rede, tornando complicado o acesso e o armazenamento de arquivos convencionais (BERNARDES, 2014).

Como mencionado o armazenamento de arquivos no BigData engloba em grande volume de dados e faz com que um mecanismo de manipulação e extração de informações em uma cadeia de máquinas seja indispensável. Para resolver este problema, a camada mais baixa do Hadoop é formada por um sistema de arquivos distribuídos chamado *Hadoop Distributed File System* ou HDFS. O HDFS foi desenvolvido para executar um algoritmo chamado *MapReduce*, que realiza leitura, processamento e escrita de arquivos extensos a uma taxa de transmissão de dados constante, sendo executado em *clusters* com hardware de baixo custo. O grande diferencial entre sistemas distribuídos é a tolerância a falhas, baixo custo de hardware e também sua escalabilidade linear. Suas principais características são:

- Possibilita armazenar e utilizar arquivos com *terabytes* de espaço, sendo utilizado por aplicações que necessitam de uma base de dados volumosa (Shvachko et al, 2010).
- Arquivos podem ser lidos quantas vezes necessário, porém podem ser escritos apenas uma vez. Tal abordagem tem o nome de *write-one, read-many-times* (WHITE, 2012). Este método é definido a partir da ideia de que o modelo é mais eficiente no processamento de dados. O HDFS faz uma leitura com uma taxa constante em qualquer arquivo, pois a prioridade é garantir eficiência na leitura e não diminuir a latência.
- O hardware utilizado para se executar o HDFS não necessita ter uma alta capacidade computacional. O sistema foi desenvolvido com a previsão de que um dos nós ao longo do *cluster* pode falhar, portanto os métodos de tolerância a falha foram feitos para contornar este problema, resultando em uma maior flexibilidade ao utilizar diferentes tipos de hardwares.

O HDFS tem um conceito de armazenagem em blocos, sendo que cada bloco de armazenamento possui 64 MB de tamanho e quando o arquivo é criado no sistema, este é dividido em blocos com estas definições e cada bloco como uma unidade independente, podendo estar em qualquer nó do *cluster*. O arquivo quando armazenado no sistema tem sua divisão feita, espalhando-se pelo *datanodes,* responsáveis por salvar os blocos fisicamente. Na máquina central conta apenas a localização de cada bloco no *cluster*. De acordo com White (2012) este método possibilita que os blocos sejam copiados pelo sistema de arquivos, prevenindo falhas e maior disponibilidade de dados (BERNARDES, 2014).

4.1 API E PRINCIPAIS FUNCIONALIDADES

Devido à grande dificuldade das empresas adotam o Apache Hadoop em seu cotidiano, surgiu a necessidade de desenvolver aplicações que estendam a plataforma. Com o desenvolvimento de novas ferramentas, recursos complementares são adicionados à plataforma Hadoop, tornando-a mais completa e aplicável a diferentes contextos (BERNARDES, 2014).

4.2 MAPREDUCE

A grande quantidade de dados que vem sendo gerado nos últimos anos vem acarretando algumas dificuldades para se armazena-los e utiliza-los. Com a evolução da computação distribuída surgiu o *MapReduce*, uma ferramenta do Hadoop que permite que se faça buscas em paralelo nos *clusters,* que permite as empresas obterem uma resposta muita mais rápida e efetiva, para se obter informações em seus *petabytes* de dados. O *MapReduce* é baseado em duas funções *map* e *reduce*. (BERNARDES, 2014).

4.2.1 Conceitos

O *MapReduce* pode ser definido como um modelo para Hadoop baseado em Java. Tal modelo se destina, principalmente, para o processo em *batch* de uma grande quantidade de dados distribuída em várias máquinas (WHITE, 2012). Para os usuários deste modelo, é necessário utilizar uma função de *map* que define um par {*chave, valor},* para o processamento, onde é gerado um produto dos valores em pares de *{chave, valor},* e também é definida uma função *reduce*, etapa em que os valores são associados a uma mesma classe.

No *MapReduce*, as entradas de arquivos são separadas em partes distintas e utilizadas como entradas das funções *map.* A saída da função é chamada *{chave, valor}* e então transformada em entradas do tipo *{chave, lista (valores)},* para a função *reduce*, e, por fim, resulta em um arquivo de saída (BERNARDES, 2014).

4.2.2 Benefícios

O modelo *MapReduce* permite que as empresas executem um tráfego bem menor de informação pela rede, pois a função *map* armazena, localmente, o resultado obtido na máquina e somente o resultado da função *reduce* transita pela rede, fazendo com que a quantidade de informação gerada por ele seja reduzida (BERNARDES, 2014).

É totalmente escalável, sendo apenas necessário a alocação de mais máquinas para a execução de operações de *map* e de *reduce*. Caso uma máquina pare de funcionar, o sistema continua funcionando devido ao fato de que as demais possuem cópias dos arquivos, sendo apenas recolocado o processamento no modelo *MapReduce*.

4.2.3 Shuffle e Sort

Etapas do modelo *MapReduce* para assegurar que os resultados da função intermediária *{chave, valor}* da função *map* sejam organizados, ligados e utilizados como parâmetros na função de *reduce*. Tais etapas são chamadas de *shuffle* e *sort* e o fragmento do código Hadoop está sempre sendo alterado para refinar o desempenho (WHITE, 2012).

Enquanto a função *map* está em executada, o resultado dos pares *{chave, valor}* são armazenados em um *buffer* de memória conforme a definição acontece. Os dados são divididos em partições e a quantidade é definida pela quantidade de funções *reduce* que serão utilizadas no processamento dos resultados. Após isso, as chaves são classificadas conforme forem sendo escritas no disco. Os resultados são arquivos intermediários e, portanto, não há a necessidade de armazená-los no sistema distribuído, mas sim no disco local da máquina. Esta etapa de particionamento e classificação é chamada de *Shuffle* (BERNARDES, 2014).

Em seguida, um nó principal, também chamado de *máster* informa às máquinas qual a localização das partições a serem utilizadas. No final da cópia das partições é realizada a etapa *sort,* que agrupa os resultados por chave, mantendo a organização por seu valor. Assim, sendo as entradas necessárias para as funções *reduce* estão prontas (BERNARDES, 2014).

4.2.4 Classes Mapper

A classe *mapper* tem tarefas individuais que utilizam chaves de entrada de valores para realizar uma ação nos valores e os transformar em um registro que é utilizado por pouco tempo. Para realizar a transformação dos registros intermediários e os de entrada não necessitam ser do mesmo tipo. Uma chave de entrada pode mapear zero ou vários pares de saída (BERNARDES, 2014).

A classe *mapper* realiza a função *map*, convertendo para uma aplicação real. Esta função deve ser implementada por um método abstrato *map().* A Figura 9 ilustra uma função para a representação de um problema de contagem de palavras.

```
public static class TokenizerMapper extends Mapper<LongWritable , Text,
      Text, IntWritable>{
            private final stativ IntWritable one = new IntWritable(1);
            private Text word = new Text();

            public void map(LongWritable key, Text value, Context context)
throws
                  IOException, InterruptedException {
                        StringTokenizer itr = new
StringTokenizer(value.toString());
                        while(itr.hasMoreTokens()) {
                              word.set(itr.nextToken());
                              context.write(word, one);
                        }
                  }

      }
```

Figura 9: Exemplo de código mapper

Os parâmetros apresentados na classe são equivalentes aos identificadores {*chave, valor}*, para entrada e saída da função *map.* Na figura 9 é possível verificar uma variável de entrada chamda do tipo *LongWritable* e outra do tipo *Text*. As palavras encontradas em *cada quebra (split) do valor na* função *map*, apresentam uma saída do tipo *Text*, que é relacionada a um valor inteiro *IntWritable* que mostra uma incidência nos arquivos de busca (BERNARDES, 2014).

A plataforma Hadoop disponibiliza suas próprias variáveis para maximizar a serialização dos objetos acessados pela rede ou *clusters*. Os tipos *LongWritable*, *IntWritable* e *Text* são, equivalente em Java, aos tipos *Long*, *Integer* e *String*. É permitido, na plataforma Hadoop que o desenvolvedor utilize os tipos primitivos existentes no Java, apenas sendo necessário criar uma especialização que implemente a interface *Writable*. A maioria das vezes isso não é utilizado, pois os tipos primitivos já estão disponibilizados como objetos para *arrays*, *maps* e os tipos primitivos.

4.2.5 Classes Reducer

A classe *Reducer* implementa a função *reduce* onde um método abstrato *reduce*() deve ser sobrescrito. Como a classe *Mapper* a classe *Reducer* apresenta quatro parâmetros que indicam os pares {*chave, valor*} de entrada e saída. As entradas são encontradas nos tipos de dados da saída da função *map* e também dos objetos mantidos na classe *Context* que como na classe *Mapper,* receberá os dados resultantes desta etapa (BERNARDES, 2014).

A primeira variável da função *reduce* é do tipo *Text,* indicando que uma palavra que foi encontrada pela função *map.* A próxima entrada resulta em uma lista de valores do tipo *IntWritable*, resultado das etapas *suffle e sort*. Este método tem como objetivo somar os valores da lista e escrever o resultado dessa contagem no método *Context.* O código Y apresenta uma implementação da classe *Reducer*. Ilustrado na figura 10 um exemplo de código *reduce* (BERNARDES, 2014).

```
public static class IntSumReducer extends Reducer<Text, IntWritable, Text,
            IntWritable>{
            private IntWritable result = new IntWritable();

            public void reduce(Text key, Iterable<IntWritable> values, Context
                  context) throws IOException,
                  InterruptedException {
                  int sum = 0;
                  for(IntWritale val : values){
                        sum += val.get();
                  }
                  result.set(sum);
                  context.write(key, result);
            }

      }
```

Figura 10: Exemplo de código reducer

Como mencionado a classe *Reducer* reduz a quantidade de dados transmitidos pela rede, operações locais. O Hadoop fornece, ao desenvolvedor, a possibilidade de especificar uma função *combine* que pode reduzir a saída das funções *map* (WHITE, 2012). Dessa forma os resultados de uma operação são acoplados em um conjunto como por exemplo, a contagem de palavras:

(ACTAA, ATTGA, GAGAC, ACGAA, ATTGA, ACTAA, GAGAC, GAGAC, ACGAA, TAGCA).

Aplicando a função combine os resultados são simplificados para:

(ACTAA, 2)
(ATTGA, 2)
(GAGAC, 3)
(ACGAA, 2)
(TAGCA, 1)

Os dados transmitidos pela rede são reduzidos significavelmente (BERNARDES, 2014).

5. PROPOSTA E DESENVOLVIMENTO DO TRABALHO

A proposta deste trabalho é a de criar um ambiente experimental que adote a plataforma Hadoop e o modelo computacional *MapReduce* para realizar buscas de padrões em sequências genéticas em um repositório de dados de arquivos XML extraídos do GenBank.

5.1. DEFINIÇÃO DE UM CENÁRIO EXPERIMENTAL

A definição de um cenário experimental para esta proposta de trabalho se constitui, principalmente, pela instalação e configuração da plataforma Apache Hadoop.

Na classe *Mapper* deve ser definida a sequência genética que se deseja comparar com o repositório de dados que armazena os arquivos do GenBank, enquanto que na classe *Reducer* deverão ser identificadas ou obtidas as sequências genéticas que possuem similaridade, tendo como resultado, a identificação das sequências e os locais que possuem valores iguais.

Dentro do ambiente experimental as buscas realizadas devem ser divididas pelos *clusters* do sistema, sendo feitas as reduções localmente para se ter um tráfego de informações reduzido pela rede.

5.2. TECNOLOGIAS E RECURSOS ADOTADOS

Os arquivos obtidos do repositório GenBank são definidos em formato XML, uma linguagem de marcação de dados que pode ser manipulada por APIs na plataforma Java. Dessa forma, os arquivos XML que armazenam as sequências genéticas extraídas do GenBank podem ser manipulados e os valores direcionados aoS recursos da plataforma Hadoop.

De maneira mais detalhada, as implementações das classes *Mapper* e *Reducer* efetuarão as consultas nos arquivos XML.

Foi definido um ambiente experimental configurado para a utilizar o Hadoop para realizar a busca por uma determinada base genética e mostrar cada arquivo XML que possui essa base. A avaliação dos resultados obtidos neste trabalho que mostra a possibilidade de utilizar o Hadoop em um contexto de Biotecnologia, que é uma alternativa para empresas utilizarem para buscas nesse contexto.

5.3. ARQUITETURA IMPLEMENTADA

Para o desenvolvimento do trabalho, um ambiente com o sistema operacional Linux Ubuntu 14.04 foi configurado. A máquina criada para o cenário experimental atua como um nó de *cluster* na plataforma Hadoop. Após a instalação e configuração do sistema operacional, o Kit de Desenvolvimento Java (JDK) também foi instalado, fornecendo todos os recursos de desenvolvimento e execução das funcionalidades *Mapper* e *Reducer* na plataforma Hadoop

Uma vez que o sistema Linux e o JDK foram instalados, um usuário deve ser adicionado ao grupo de super usuários do sistema Linux para fornecer as permissões de execução das operações no Hadoop.

Foi instalado o ssh-server e definido de modo que não necessite de senha com o comando `ssh-keygen -t rsa -P " "`. É necessário executar os seguintes comandos `cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys` e `cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys` para dar permissão de execução do ssh. Desta forma basta iniciar o ssh com o comando `ssh localhost` e confirmar a execução.

Foi realizado o download do Hadoop. Ao final do download o arquivo foi descompactado e movido para o local `/usr/local/hadoop` e permitindo somente que o usuário hduser possa modificar a pasta, esta operação é feita com a seguinte linha de comando `sudo chown -R hduser /usr/local/hadoop`. No arquivo *bashrc* é necessário indica-lo as bibliotecas que serão utilizadas. Para abrir o arquivo deve-se utilizar o comando `sudo nano ~/.bashrc`, e ao final do arquivo indicar os seguintes *exports*.

```
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
export HADOOP_HOME=/usr/local/hadoop
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib"
```

Figura 11 – Configuração ~/.bashrc

O arquivo Java dentro da pasta Hadoop deve ser modificado para poder ser identificado onde está a instalação do programa. Para realizar esta configuração deve-se utilizar o seguinte comando `sudo nano /usr/local/hadoop/etc/hadoop/hadoop-env.sh`, para abrir o arquivo de configuração e alterar o `export JAVA_HOME=${JAVA_HOME}` por `export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64`.

È necessário configurar os arquivos do Hadoop para identificar a execução em *single node*. Os arquivos que devem ser configurados são: core-site.xml, hdfs-site.xml, yarn-site.xml, map-red-site.xml que estão localizados em `/usr/local/hadoop/etc/hadoop/`. Por padrão todos os arquivos citados vêm sem configuração. O primeiro arquivo configurado é o core-site.xml, a figura 12 ilustra como são as configurações colocadas dentro do arquivo indicando o ssh.

```
<configuration>
        <property>
                <name>fs.defaut.name</name>
                <value>hdfs://localhost:9000</value>
        </property>
</configuration>
```

Figura 12 – Configuração core-site.xml

O próximo arquivo configurado é o hdfs-site.xml como ilustra a figura 13, nele é necessário informar que é utilizado somente uma máquina e o local das pastas *namenode* e *datanode*.

```
<configuration>
        <property>
                <name>dfs.replication</name>
                <value>1</value>
        </property>
        <property>
                <name> dfs.namenode.name.dir</name>
                <value>file:/usr/local/hadoop_tmp/hdfs/namenode</value>
        </property>
        <property>
                <name>dfs.datanode.data.dir</name>
                <value>file:/usr/local/hadoop_tmp/hdfs/datanode</value>
        </property>
</configuration>
```

Figura 13 – Configuração hdfs-site.xml

O próximo arquivo configurado é o yarn-site.xml demonstrado na figura 14, onde é configurado a base do *MapReduce* indicando para o Hadoop onde se deve procurar a configurações desta classe.

```
<configuration>
        <property>
                <name>yarn.nodemadnager.aux-services</name>
                <value>mapreduce_suffle</value>
        </property>
        <property>
                <name>yarn.nodemanager.aux-services.mapreduce_suffle.class</name>
                <value>org.apache.hadoop.mapred.SuffleHandler</value>
        </property>
</configuration>
```

Figura 14 – Configuração yarn-site.xml

O próximo arquivo configurado é o mapred-site.xml demonstrado na figura 15, onde é indicado é passado o valor para ele do YARN

```
<configuration>
        <property>
                <name>mapreduce.framework</name>
                <value>YARN</value>
        </property>
</configuration>
```

Figura 15 - Configuração mapred-site.xml

Foi desenvolvido um código em Java utilizando o Eclipse como é ilustrado na figura 16. Para executá-lo é necessário criar um arquivo jar do mesmo, mudar para o usuário hduser para se utilizar o Hadoop. Através do terminal deve ser executado o comando `hadoop jar nomedojar.jar NomeClasse parâmetros` para se executar um arquivo Hadoop.

```
public class SearchFiles {

public static void main(String[] args) throws IOException {

    if (args.length < 2) {

        System.err.println("Usage: [search-dir]");

        return;

    }

    File searchDir = new File(args[0]);

    String searchString = args[1];

    ArrayList<File> matches = checkFiles(searchDir.listFiles(), searchString, new ArrayList<File>());

    System.out.println("Esses arquivos tem '" + searchString + "':");

    for (File file : matches) { System.out.println(file.getPath()); }    }

  private static ArrayList<File> checkFiles(File[] files, String search, ArrayList<File> acc) throws IOException {

    for (File file : files) {

        if (file.isDirectory()) {

            checkFiles(file.listFiles(), search, acc);

        } else { if (fileContainsString(file, search)) { acc.add(file);}          }       }

    return acc;

  }

  private static boolean fileContainsString(File file, String search) throws IOException {

    BufferedReader in = new BufferedReader(new FileReader(file));

    String line;

    while ((line = in.readLine()) != null) {

        if (line.contains(search)) {

            in.close();

            return true;

        }       }

    in.close();

    return false;

  }}
```

Figura 16 – Código Java utilizado

O código somente é necessário a informação do que o Hadoop deve pesquisar pois a programação de Mapper e Reducer é realizada por ele sem a necessidade de informar o que ele deve mapear e reduzir.

5.4. EXPERIMENTOS E AVALIAÇÃO DA ABORDAGEM PROPOSTA

Foi configurado um ambiente experimental onde o Hadoop utiliza o single node para realizar buscas de bases genéticas que possam um determinado trecho genético, mostrando somente os arquivos que possuem este determinado trecho.

A programação realizada neste trabalho demonstra a possibilidade de utilizar esta ferramenta para realizar buscas por padrões genéticos. Foi realizado vários experimentos que obtiveram sucesso para determinar quais bases possuem um determinado trecho genético.

5.5. ANÁLISE DOS RESULTADOS E LIÇÕES APRENDIDAS

Com o Hadoop é possível se realizar a busca por padrões em arquivos, é uma ótima ferramenta para processamento de dados, e por ser uma ferramenta atual possui pouca informação sobre ela, mas é algo que proporciona ao programador uma facilidade pois a parte de map e reduce é realizada pelo próprio programa facilitando o desenvolvimento de algum código.

O tempo de busca é curto mesmo sendo realizado os testes somente em uma máquina, mas com vários arquivos para serem processados, demonstrando assim a possibilita de alto desempenho se realizado testes em um ambiente que possui grandes quantidades de dados, permitindo que quando necessário a adição de hardware para se obter um processamento mais rápido.

6. CONCLUSÃO

A plataforma Hadoop demonstrou ser uma ferramenta poderosa e como uma alternativa para operações de buscas em grandes repositórios de dados, simplificando o processo de consultas a partir da definição de padrões genéticos em arquivos disponibilizados no GenBank.

De maneira sucinta, pôde-se investigar e experimentar a adoção de Hadoop e *MapReduce* no contexto de biotecnologia. A experiência adquirida e os resultados obtidos fornecem subsídios que, mesmo de maneira experimental, tornam possível considerar o uso das tecnologias estudadas como válidas para o contexto proposto.

O tema BigData é um assunto que possui pouca informação atualmente, tendo isso como um problema para se obter informações sobre o assunto. Uma informação para se realizar programação da plataforma Hadoop é algo um pouco complexo tendo pouca informação disponível.

Este trabalho contribuiu, mesmo que de maneira experimental, como base para demonstrar o interesse e a possibilidade de novos aprimoramentos no contexto de biotecnologia e análise de padrões genéticos, bem como na concepção de novas estratégias relacionadas à plataforma Hadoop e ao uso de algoritmos de *MapReduce*. Espera-se que novas pesquisas explorem e investiguem como que os conceitos e recursos de analytics podem apoiar as áreas de biotecnologia e bioinformática.

6.1 TRABALHOS FUTUROS

Como trabalho futuro, sugere-se a modelagem e o desenvolvimento de uma aplicação integrada à plataforma Apache Hadoop e *MapReduce* que permita a configuração e execução de processos por meio de uma interface Web. Dessa forma, ao invés de todo o processo ser manutenido por instruções de comandos, a aplicação Web possibilitaria interagir com os recursos de Hadoop de maneira mais intuitiva. Outra funcionalidade a ser considerada é a adoção de múltiplos nós durante a execução dos processos de *MapReduce*. Assim sendo, a análise de padrões genéticos pode ser descentralizada e executada de forma paralelizada, tornando as operações mais performáticas.

REFERÊNCIAS

ANDRADE, Tiago P. C. **MapReduce - Conceitos e Aplicações.** Campinas. Disponível em <http://www.ic.unicamp.br/~cortes/mo601/trabalho_mo601/tiago_cruz_map_reduce/relatorio.pdf>. Acesso em 13 out.2015.

ARAÚJO, Nilberto D. FARIAS, Rodrigo P. PEREIRA, Patrícia B. FIGUEIRÊDO, Flávia M. MORAIS, Alanna M. B. SALDANHA, Livina C. GABRIEL, Jane E. **A era da bioinformática: seu potencial e suas implicações para as ciências da saúde.** Faculdades Integradas de Patos(FIP). Patos, Paraíba 2008.

BERNARDES, Guilherme L. **Desenvolvimento de Software no Contexto Big Data.** Universidade de Brasília (UnB), Faculdade Unb Gama (FGA) – Brasília 2014.

BRITO, Rogério T. **Alinhamento de sequências biológicas.** Instituto de matemática e estatística da Universidade de São Paulo. São Paulo 2003.

CALHAU, Ana. PISCO, Ângela. Santos, Nuno. **Alinhamento de sequências**. Instituto Superior Técnico. Universidade Técnica de Lisboa. Lisboa. 2008.

CARAZZOLLE, Marcelo F. **Métodos de alinhamento de sequências biológicas.** Disponível em <http://www.lge.ibi.unicamp.br/lgeextensao2008/extsup/alinhamentos.pdf> Acessado em 20 fev. 2016.

COELHO, Keila S. **Curso de Biotecnologia**. Disponível em: <http://www.portaleducacao.com.br/educacao/cursos/cursos_detalhes.asp?id=123>. Acesso em: 14 out. 2015.

GENEBIO, Universidade Federal da Bahia Disponível em <http://www.genebio.ufba.br/?page_id=303>. Acesso em 2 mar. 2016.

GOLDMAN, Alfredo. KON, Fabio. JUNIOR, Francisco P. POLATO, Ivanilton, PEREIRA, Rosangela F. **Apache Hadoop: conceitos teóricos e práticos, evolução e novas possibilidades.**

JUNIOR, Carlos E. S. **Integração de um aplicativo para reconhecimento de padrões na sequência de DNA com banco de dados XML**. 2011. p14. Monografia. Bacharelado

em Ciência da Computação. Fundação Educacional do Município de Assis. São Paulo. Assis. 2011.

KELLENBERGER, Jakob (2002). **"Biotecnology". In:** ***International Herald Tribune****.* 27 de setembro.

MEIDANIS, João, SETUBAL, João C. **Introduction To Computational Molecular Biology.** 1.ed. [s,1]: IE-Thomson, 1997.

MELLO, Gabriel E. C. MERUSSI, Michael. SCHROPEL, Jonathan. CAZAROTTO, Paulo H. **Banco de dados e Bioinformática**. p2. Universidade do Vale do Rio dos Sinos.

MOUNT, D.W., **Bioinformatics: Sequence and Genome Analysis**, Cold Spring Harbor Laboratory Press, 2001.

MUNDO EDUCAÇÃO, Disponível em <http://mundoeducacao.bol.uol.com.br/biologia/dogma-central-biologia-molecular.htm> Acessado 15 fev. 2016.

OLIVEIRA, Deive C. **Alinhamento de sequências.** Universidade Federal de Lavras. Minas Gerais. 2002.

PORTAL SAS, Big Data. Disponível em: <http://www.sas.com/pt_br/insights/big-data/what-is-big-data.html>. Acesso 13 out. 2015.

PROSDOCIMI, Francisco. APOSTILA da oficina prática de genética, genoma e biotecnologia. 4ºMódulo. 2004.

PUCCI, João N. **Comparação de sequências de DNA.** Revista multidisciplinar da UNIESP. 2008.

QUEIROX, Alexandre. **Apostila de introdução a bioinformática**. 2002. p2. Departamento de Biofísica e Farmacologia. Universidade Federal do Rio Grande do Norte. Rio Grande do Norte. Natal. 2002.

REISNER, H. M. Patologia: uma abordagem por estudos de casos. Porto Alegre: AMGH, 2016.

REN, Jiansi. LU, Jiantao. WANG, Lizhe. CHEN, Dan. **Data Visualization in Bioinformatics.** Wihan Medical Union, Wihan China. 2012.

SANTOS, Francisco P. CASTRO, Cibele S. **Síntese de processamento de RNA.** Disponível em <http://www2.bioqmed.ufrj.br/prosdocimi/RNA/processamento.htm>. 2000. Acesso 10 fev. 2016.

SILVA, Marco T. N. **ALINHAMENTO MÚLTIPLO GLOBAL DE SEQÜÊNCIAS PELA REPRESENTAÇÃO DE PROFILE E CLUSTERIZAÇÃO: COMPARAÇÃO COM OS RESULTADOS DO CLUSTALW (EMBL-EBI).** Universidade Federal de Lavras. Minas Gerais. 2006.

SOUZA, Karina Ap. NEVES, Valdir A. **Experimentos de Bioquímica**. Disponível em < http://www.fcfar.unesp.br/alimentos/bioquimica/introducao_proteinas/introducao_proteinas_dois.htm> < http://www.fcfar.unesp.br/alimentos/bioquimica/introducao_proteinas/introducao_proteinas_tres.htm>. Acesso 20 fev. 2016.

UNIVERSIDADE FEDERAL DE PERNANBUCO. Disponível em < https://www.ufpe.br/biolmol/aula9-datamining.htm>. Acesso em 15 mar. 2016;

UOL. Disponível em < http://sitehelpme.xpg.uol.com.br/HelpMe/Site.php/dna.html >. Acesso em 22 jul. 2016.

VICTORINO, Valério I. P. **Mapeando os desafios da biotecnologia:**

Aportes sociológicos na regulação pública. p5. Universidade do Vale do Itajaí (UNIVALI) – Santa Catarina. 2003.

WHITE, T. **Hadoop: The Definitive Guide**. 3º edição. Beijing: O'Reailly, 2012.