IMPLEMENTAÇÃO DE UMA ARQUITETURA BASEADA EM ALGORITMOS INTELIGENTES PARA DESCOBERTA DE MOTIFS EM EXPRESSÕES GENÉTICAS

MICHEL PEDROSO[1]; AUGUSTO SCHMIDT[2]; MARILTON AGUIAR[3]

[1]*Universidade Federal de Pelotas – mspedroso@inf.ufpel.edu.br*
[2]*Universidade Federal de Pelotas – augustgs@inf.ufpel.edu.br*
[3]*Universidade Federal de Pelotas– marilton@inf.ufpel.edu.br*

1. INTRODUÇÃO

Bioinformática é uma área da ciência que vem mostrando um crescimento exponencial nos últimos anos, tornando necessária a análise de dados em biologia molecular (PROSDOSCIMI, 2007). Também é necessária a adaptação e criação de novas ferramentas para essa área, facilitando e diminuindo o tempo de execução, aumentando eficiência e rapidez, pois os dados e informações aumentam cada vez mais (PROSDOSCIMI, 2007).

Motifs são entidades não aleatórias, encontradas em cadeias de DNA, caracterizado por um padrão, um fenômeno não único. Motifs além de possuírem padrões recorrentes na sequência analisada, também possuem uma função biológica. Pode-se definir motifs como um curto segmento compartilhado por múltiplas sequências de DNA que pode conter informações sobre evolução, estrutura ou função (YI-PING PHOEBE CHEN, 2005).

Encontrar possíveis motifs válidos nas sequências de DNA ou proteínas é um dos maiores problemas da bioinformática, pertencendo à classe NP-Completo, devido a quantidade de sequências geradas a partir das ferramentas de procura e alinhamento.

Portanto, é necessária a utilização de algoritmos inteligentes para melhorar o desempenho de ferramentas consolidadas, mas com limitações de desempenho, de uma forma que possa obter melhores resultados e um melhor tempo de execução para um conjunto de ferramentas.

Neste contexto, este trabalho apresenta resultados preliminares da implementação de uma arquitetura baseada em algoritmos genéticos para descoberta de motifs em expressões genéticas, em desenvolvimento no grupo de pesquisa.

A base desta implementação é a utilização de um algoritmo, com o principal objetivo de realizar buscas para encontrar soluções aproximadas em um determinado problema, que neste contexto é encontrar Motifs aleatórios que exerçam uma determinada função dentro de uma expressão genética (BARRET, STEVEN, 2006).

Uma das principais ferramentas utilizadas atualmente para alinhamento de sequências é o BLAST (*Basic Local Alignment Search Too*l), um conjunto de algoritmos de bioinformática para comparação de sequências montadas para exploração de informações contidas em sequências de DNA e proteínas (ALTSCHUL, STEPHEN, et al. 1990).

Para refinar estas informações utilizou-se a ferramenta chamada CD-Hit, que reduz a quantidade de informações, descartando todas que não estejam dentro do padrão estabelecido (LI, WEIZHONG, ADAM GODZIK. 2006).

Para realizar os inúmeros alinhamentos, utilizou-se um pacote de algoritmos chamado MUSCLE (*Multiple Sequence Comparison by Log-Expectation*) que é

baseado em uma abordagem eficiente, realizando a análise direta de sequências, a construção de uma árvore guia e alinhamento (EDGAR, RC. 2004).

Ainda, a ferramenta HMMER é usada para encontrar sequências de Motifs que foram geradas em diversas bases de dados, utilizando técnicas probabilísticas de Modelos Ocultos de Markov para criar um modelo definido com todas informações anteriores (FINN RD, et al. 2011).

Mas essas ferramentas podem ser utilizadas separadamente ou em ordem diferente, sendo isso um critério do pesquisador que está realizando os procedimentos para descobertas de possíveis Motifs válidos entre uma determinada expressão.

Neste cenário, a proposta e implementação desta arquitetura tem o objetivo de sistematizar a tecnologia atualmente disponível para os pesquisadores da bioinformática, tornando cada vez mais satisfatórios os resultados obtidos.

2. METODOLOGIA

Para a realização da implementação foram utilizados todos conceitos discutidos anteriormente. O algoritmo genético foi implementado utilizando a linguagem Java e, também, utilizou-se a linguagem Python junto com a biblioteca BioPython para realização dos procedimentos, mais especificamente, na conversão do formato .XML para .Fasta e na manipulação e preparação dos dados entre as ferramentas.

A dificuldade desta implementação é o grande número de sequências geradas por todas etapas de execução, por isso foi necessário realizar procedimento para refinar as informações obtidas (LI, WEIZHONG, ADAM GODZIK. 2006).

A primeira etapa é a utilização do Algoritmo Genético, que parte de um arquivo contendo as sequências genéticas necessárias para a criação dos possíveis Motifs aleatórios. O objetivo do algoritmo genético é criar um arquivo contendo todos possíveis Motifs promissores, gerando sequências aleatórias baseadas no arquivo utilizado como entrada do mesmo (BARRET, STEVEN, 2006).

Após a utilização do Algoritmo Genético, utiliza-se a ferramenta BLAST, para realizar os procedimentos de alinhamento local utilizando como base de dados o resultado obtido na etapa anterior. O formato escolhido para armazenar as informações foi o .XML, devido a facilidade de manipulação e organização das informações, tornando possível a utilização de *tags* para retirar as informações necessárias para a próxima etapa de execução.

Depois desta etapa é necessário converter o arquivo gerado (.XML) para um formato utilizado pelas ferramentas de bioinformática (.FASTA), nesta etapa também é realizado um filtro entre os possíveis Motifs para realizar uma eliminação de Motifs incorretos.

A próxima etapa é a utilização de uma ferramenta específica de refinamento (CD-Hit), para garantir que somente informações válidas sejam utilizadas nas próximas etapas. O filtro utilizado nesta ferramenta foi de 70% em relação ao valor de acertos encontrados nos possíveis Motifs.

Com informações menos redundantes e errôneas, utiliza-se a ferramenta MUSCLE para realizar o procedimento de alinhamento múltiplo, realizando análises sobre cada sequência encontrada.

A última etapa de execução é a utilização da ferramenta HMMER, que cria um modelo com as informações anteriores. Esta ferramenta é necessária pois torna possível analisar os resultados de forma eficiente e padronizada, onde cada campo de informação do arquivo anterior contém uma coluna com seus respectivos resultados.

3. RESULTADOS E DISCUSSÃO

O tempo médio de execução da arquitetura foi de 2 minutos e 51.956 segundos, um tempo satisfatório para a quantidade de etapas realizadas e para a quantidade de informações em cada procedimento realizado.

A Tabela 1 mostra a quantidade de sequências encontradas em cada etapa da arquitetura.

Tabela 1 – Resultados preliminares da implementação desenvolvida indicando a quantidade de sequências encontradas em cada etapa.

Algoritmo Genético	Blast	Conversão	CD-Hit	Muscle	% Motifs Válidos
1.757	174.914	17.283	28	28	1,59%

Como pode-se analisar, as 3 primeiras ferramentas geram uma quantidade significativa de sequências, deixando claro a necessidade da utilização da ferramenta de refinamento, que reduz drasticamente a quantidade de sequências encontradas.

As ferramentas seguintes não modificam a quantidade de sequências, apenas executam suas funcionalidades nas sequências encontradas. O último campo da tabela informa a porcentagem de Motifs válidos encontrados a partir das sequências da primeira etapa.

A quantidade de sequências obtidas no final da execução foi de: 28 sequências, que são possíveis Motifs válidos dentro do arquivo inicial carregado pelo Algoritmo Genético. A partir destes resultados será possível realizar procedimentos da área de bioinformática para verificar a eficácia dos Motifs encontrados.

4. CONCLUSÕES

Como pode ser observado, foi obtido um ganho de 98,41% sobre a quantidade de sequências encontradas anteriormente, um ótimo ganho em relação ao maior número de sequências encontradas (174.914 sequências) anteriormente.

Para melhorar os resultados obtidos algumas mudanças interessantes poderiam ser realizadas no algoritmo genético, que é a base de criação dos Motifs promissores aleatórios, sendo elas: aumentar o tamanho de possíveis Motifs e realizar alterações que permitam que o mesmo receba uma retroalimentação das informações finais da arquitetura.

Realizando essas alterações o algoritmo genético iria gerar Motifs cada vez mais corretos, pois sempre receberá como entrada um resultado refinado e com uma garantia de 70% de chance de estar correto, podendo talvez encontrar diretamente um Motif promissor.

Com isso é possível concluir que a arquitetura apresentou um bom resultado, um ótimo ganho de informações relevantes contendo possíveis Motifs e um tempo de execução satisfatório, tornando possível a utilização da arquitetura de forma eficiente e rápida.

Após todas informações contidas neste artigo pode-se concluir que a arquitetura não é apenas válida e usual, mas que é promissora para sua devida área de atuação. Também é possível notar, que a criação de um Software utilizando esta arquitetura como base, seria extremamente importante para a área

de bioinformática pois tornaria mais fácil a utilização de todas as ferramentas, como também a utilização de alguma das ferramentas de forma individual.

5. REFERÊNCIAS BIBLIOGRÁFICAS

ALTSCHUL, STEPHEN F., et al. **Basic local alignment search tool.** Journal of molecular biology 215.3 (1990): 403-410.

BARRET, STEVEN J. **Intelligent Bioinformatics: The application of Artificial Intelligence Techniques to Bioinformatics Problems.** Genetic Programming and Evolvable Machines 7.3 (2006): 283-284.

EDGAR, RC. **MUSCLE: multiple sequence alignment with high accuracy and high throughput.** Nucleic Acids Research 32 (5): 1792–97. doi:10.1093/nar/gkh340. PMC 390337. PMID15034147. (2004).

FINN RD, CLEMENTS J, EDDY SR. **HMMER web server: Interactive sequence similarity searching. Nucleic Acids Research.** Web Server Issue 39:W29-W37. (2011).

LI, WEIZHONG, and ADAM GODZIK. **Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences.** Bioinformatics 22.13 (2006): 1658-1659.

PROSDOCIMI, F. **Introdução à bioinformática**. Belo Horizonte: Biotecnologia ciência e desenvolvimento, (2007).

YI-PING PHOEBE CHEN. **Bioinformatics Technologies**. Springer Science & Business Media, (2005).