



UNIVERSIDADE ESTADUAL DE SANTA CRUZ  
PRO-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO

PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM COMPUTACIONAL  
EM CIÊNCIA E TECNOLOGIA

**ARTURO ERNESTO MACHADO ORTEGA**

**LINGUAGENS FORMAIS PARA GERAÇÃO, RECONHECIMENTO E DIFERENCIAMENTO DE  
6 FAMÍLIAS DE SEQUÊNCIAS ALU**

**ILHÉUS-BA  
2016**

**ARTURO ERNESTO MACHADO ORTEGA**

**LINGUAGENS FORMAIS PARA GERAÇÃO,  
RECONHECIMENTO E DIFERENCIADA DE 6 FAMÍLIAS DE  
SEQUÊNCIAS ALU**

Dissertação apresentada ao Programa de Pós-Graduação  
em Modelagem Computacional em Ciência e Tecnologia  
da Universidade Estadual de Santa Cruz, como parte  
das exigências para obtenção do título de Mestre em  
Modelagem Computacional em Ciência e Tecnologia.

Orientador: Prof. Dr. Luciano Ângelo de Souza  
Bernardes

Coorientador: Prof. Dr. César Alberto Bravo Pariente

ILHÉUS-BA  
2016

077

Ortega, Arturo Ernesto Machado.

Linguagens formais para geração, reconhecimento e diferenciação de 6 famílias de sequências  
ALU / Arturo Ernesto Machado Ortega. – Ilhéus, BA:  
UESC, 2016.

61f.: il.

Orientador: Luciano Ângelo de S. Bernardes.

Dissertação (Mestrado) – Universidade Estadual de Santa Cruz. Programa de Pós-Graduação em Modelagem Computacional em Ciência e Tecnologia.

Inclui referências e apêndices.

1. Linguagens formais. 2. Algoritmos computacionais. 3. Genômica. I. Título.

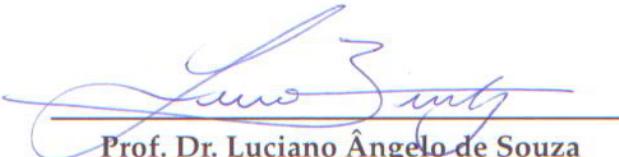
CDD 005.133

**ARTURO ERNESTO MACHADO ORTEGA**

**LINGUAGENS FORMAIS PARA GERAÇÃO,  
RECONHECIMENTO E DIFERENCIADA DE 6 FAMÍLIAS DE  
SEQUÊNCIAS ALU**

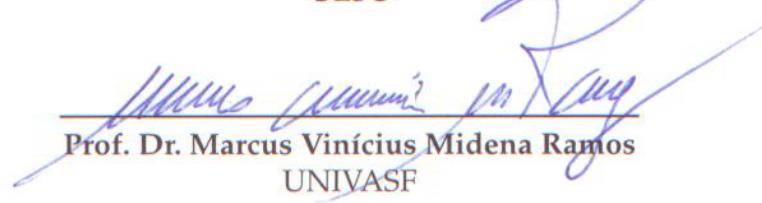
Ilhéus-BA, 12/09/2016

Comissão Examinadora

  
**Prof. Dr. Luciano Ângelo de Souza  
Bernardes**  
UESC  
(Orientador)

  
**Prof. Dr. César Alberto Bravo Pariente**  
UESC  
(Coorientador)

  
**Prof. Dr. Paulo Eduardo Ambrósio**  
UESC

  
**Prof. Dr. Marcus Vinícius Midena Ramos**  
UNIVASF

Dedico este trabalho a minha família

## **Agradecimentos**

- A Deus, pela sabedoria e iluminação para realizar este trabalho.
- Ao Programa de Pós-Graduação em Modelagem Computacional em Ciência e Tecnologia e a CAPES pela oportunidade e financiamento.
- Aos Professores Luciano de Souza e César Bravo, pelo apoio e orientação no caminho da formação no Mestrado.
- Aos Professores Dany Sanchez, Susana Marrero, Paulo Ambrósio e Francisco Bruno, pelos ensinamentos e conselhos recebidos.
- A minha família e a minha namorada, pelo amor e apoio constantes apesar da distância.
- A minha avó Elvia, por ser luz eterna na minha vida.
- Aos amigos Edgardo, Luis, Itzel, Andrea, Victoria, Ricardo, Hugo e Maria, por ser um apoio incondicional.
- Aos amigos Rocío, Mateo, Yuliana, Andrés, Alberto, Nayelli e Jhon, pelo incentivo e por ser a minha família estrangeira.
- Aos amigos Leo, Tarcila, Jorge, Rogério e colegas do Mestrado, pelo apoio e convívio com carinho brasileiro.

*"Tudo parece ser impossível, até que seja feito."*

Nelson R. Mandela

# Linguagens formais para geração, reconhecimento e diferenciação de 6 famílias de sequências ALU

## Resumo

Para os estudos de distância genômica entre varias espécies ou tipos de sequências biológicas, existem muitos dados que podem ser utilizados. Geralmente, no mundo que observamos a olho nu nas características morfológicas, não conseguimos identificar as diferenças entre organismos que embora pertençam à mesma espécie, em termos genéticos apresentam características diferentes. Segundo David Searls (1995), estruturas e formações biológicas nos genomas de qualquer espécie podem ser modeladas através de algum tipo de linguagem formal, a qual ao mesmo tempo contêm várias estruturas sintáticas que apresentam a possibilidade de serem interpretadas para definir uma métrica para distância genética. Através da análise das diferenças nas estruturas, o objetivo é estabelecer restrições sobre as linguagens associadas correspondentes que refletem essa distância. Dentro da caracterização dos genomas dos primatas, existe um tipo de sequência com natureza repetitiva conhecida como ALU, o qual é um elemento móvel dentro do genoma, e representa até 11% do mesmo, segundo Häslar Strub (2006, p. 5491). Biologicamente, uma sequência ALU qualquer, na sua forma de mRNA (RNA amadurecido para produção de proteínas), apresenta diferentes construções, geralmente em forma de estruturas secundárias, que podem ser modeladas com alguma gramática formal. A técnica proposta no presente projeto é baseada na combinação da informação obtida de tais sequências e programar um modelo que mostre a distância expressa especificamente em estruturas sintáticas. Para mostrar as diferenças de modo específico, usaremos uma técnica gráfica de filogenia, a qual é definida como a relação evolutiva entre grupos de organismos (por exemplo, espécies ou populações), a qual é descoberta por meio do sequenciamento de dados moleculares e matrizes de dados morfológicos. Utilizando a informação das linguagens formais associadas sobre o alfabeto dos estados adotados pelos autômatos, definidos especificamente para reconhecer sequências para as diferentes famílias de sequências ALU, foi estabelecida a proposta para construção de árvores filogenéticas baseadas apenas em estruturas sintáticas. A importância da escolha das sequências ALU radica no fato, além do tamanho relativamente prático, de existir antecedentes sobre as muitas funções das mesmas, como por exemplo, atuando como modificadores de RNA ao longo do genoma e validando fatores evolutivos em espécies ancestrais de primatas, segundo Currat & Excoffier (2004). No entanto, existem poucas referências à diferenciação sequências expressas em estruturas de linguagens formais, especificamente com espécies de primatas e seres humanos, considerando-se o processo evolutivo através dos anos. As bases gerais do projeto são: linguagens formais para expressão de diferentes sequências genômicas, algoritmos de programação para

criação do modelo computacional para comparação, e os fundamentos e conceitos de biologia molecular envolvidos nas sequências ALU.

**Palavras-chave:** (Inserção ALU, gramáticas formais, informação filogenética).

# Formal languages for expression, recognition and differentiation of 6 ALU sequences families

## Abstract

For genetic distance studies between several species or kinds of biological sequences, there are many types of data that can be used. Generally, in the world we can observe with the naked eye, in morphological characteristics we cannot identify the differences between organism that, even belonging to the same species, in genetic terms have many different features. According to David Searls (1995), biological structures and compositions within the genome of any species can be modelled through some type of Formal Languages, which simultaneously presents several syntactical structures that are possible to be interpreted to define a metric to measure genetic distance. Through several analysis in those structural differences, the primary objective is to establish restriction levels applied over the associated languages, that ultimately reflect said genetic distance. Within the primates genome characterization, there exists a type of sequence with repetitive nature, known as ALU, which is a mobile element present in the human and primates genome, and represents up to 11% of its composition, according to Häslar Strub (2006, p. 5491). Biologically, any ALU sequence in its mRNA expression (Mature RNA for protein production), presents different types of constructions, generally in Secondary Structures, which can be modelled through some Formal Grammar. The proposed technique for this project is to combine the information obtained from said sequences, and create a computer model which shows the genetic distance expressed in syntactical structures. To explain the differences in a more specific way, we will use a graphic tool of Phylogeny, which basically is defined as the evolutionary reason between several groups of organisms (for example, species or populations groups), which are constructed through molecular data processing and different morphological data arrays. Using information from the associated formal languages over the alphabet of states adopted by the Automates, defined specifically to recognize sequences for the different ALU families, we establish our proposal for phylogenetic tree construction based solely on syntactical structures. The importance for the selection of ALU sequences for our model is based, besides their practical size, on several past studies where many of their functions were discovered like, for example, their capacity to modify RNA throughout the genome and validating evolutionary factors in ancestral species of primates, according to Currat & Excoffier (2004). However, there are very few references to sequence differentiation expressed in formal grammars structures, specifically with human and primate species, considering the evolutionary process throughout the years. The general bases of this project are: formal grammar to express different genomic

sequences, programming algorithms to create a computer model for comparison, and the concepts and theoretical concepts of molecular biology concerning ALU sequences.

**Keywords:** ALU insertion, formal grammars, phylogenetic information.

# Lista de figuras

Figura 1 – Bases técnicas do projeto . . . . .	3
Figura 2 – Distribuição dos diferentes tipos de sequências de nucleotídeos . . . . .	4
Figura 3 – Tipos de Sequências repetitivas . . . . .	4
Figura 4 – Possíveis eventos de splicing alternativo de uma sequência ALU . . . . .	5
Figura 5 – Árvore filogenética para a provável evolução da linhagem humana . . . . .	5
Figura 6 – Ferramentas computacionais . . . . .	6
Figura 7 – Elementos de uma linguagem formal . . . . .	7
Figura 8 – Exemplo de um autômato finito determinístico programado na linguagem RUBY aplicado en sequências ALU para busca de palavra 'aca' e o seu respectivo diagrama de transições . . . . .	8
Figura 9 – Classificações principais dos Transposons nos mamíferos . . . . .	9
Figura 10 – Exemplo de uma sequência ALU . . . . .	10
Figura 11 – Arquitetura dos elementos ALU . . . . .	10
Figura 12 – Sequências de DNA de clones derivados da região de controle mitocondrial de um fóssil de Neandertal . . . . .	13
Figura 13 – Sítios variáveis nas sequências de DNA dos fragmentos obtidos através de sequenciamento direto . . . . .	14
Figura 14 – Alinhamento de sequências de posições 880-1000 de nucleotídeos do gene FOXP2. As substituições de nucleotídeos na linhagem humana são indicadas pelas setas cinzentas. Posições iguais no alinhamento são indicadas por pontos. Os três pares de primers utilizados para recuperar as duas substituições dos neandertais Sidron são indicados por setas . . . . .	14
Figura 15 – Contexto de duas sequências ALU, encontradas no gene precursor de Beta-amiloide (produtos da Proteína Precursora Amiloide), os quais são peptídeos observados no cérebro de pacientes com doença de Alzheimer . . . . .	15
Figura 16 – Exemplo de uma sequência ALU de uma longitude de 290 pares de bases com uma terminação de 7 bases de Adenina e suas principais seções, incluindo as caixas internas . . . . .	15
Figura 17 – União de SRP9/14 com sequência ALU no seu monómero esquerdo. Mudanças de nucleotídeos que acompanharam a evolução de elementos ALU é mostrada com círculos para AluY e com quadros para AluYa5. Os sítios de união com SRP9/14 está indicados por IIB, IB e IV. A região conservada está presente em AluSx, AluY, e AluY5 . . . . .	16
Figura 18 – Exemplo de busca de palavras segundo o algoritmo Knuth-Morris Pratt	18

Figura 19 – Exemplo de busca de palavras segundo o algoritmo Boyer-Moore . . . . .	19
Figura 20 – Diagrama de estados no funcionamento geral do algoritmo Aho-Corasick . . . . .	20
Figura 21 – Diagrama de estados correspondente às palavras chave . . . . .	20
Figura 22 – Exemplo de tabelas de busca de palavras para o algoritmo Aho-Corasick	21
Figura 23 – Regras de produção da Gramática e correspondente árvore de derivação para sequência de RNA <b>caucaggagaagaucucuug</b> . . . . .	23
Figura 24 – Gramática para produção de cortes em uma fita de DNA, onde $\Sigma = a, c, g, t$ , o simbolo $\delta$ representa o símbolo de corte e o símbolo $\epsilon$ representa a palavra vazia . . . . .	24
Figura 25 – Gramática otimizada para produção de cortes em uma fita de DNA, onde $\Sigma = a, c, g, t$ , o simbolo $\delta$ representa o símbolo de corte e o símbolo $\epsilon$ representa a palavra vazia . . . . .	24
Figura 26 – Classes de Linguagens formais utilizadas para expressão de sequências genômicas . . . . .	25
Figura 27 – Caixas internas para família consenso AluSx . . . . .	26
Figura 28 – Diagrama de transição de estados para autômato reconhecedor da família AluSx . . . . .	26
Figura 29 – Construção do <i>Pipeline</i> do projeto . . . . .	27
Figura 30 – Exemplo de escritura de código RUBY para autômato finito determinístico . . . . .	28
Figura 31 – Exemplo de execução de código RUBY para autômato finito determinístico reconhecendo cada símbolo da sequência <b>tctcaaaaaa</b> , sendo aceita com valor <i>true</i> . . . . .	29
Figura 32 – Sítio de pesquisa de sequências através do site do NCBI . . . . .	30
Figura 33 – Alinhamento múltiplo dos consensos das famílias ALU . . . . .	31
Figura 34 – Estrutura Secundária da sequência RNA AluY, com suas caixas internas correspondentes. De cor verde, é marcada a região inicial. A cor azul clara indica a caixa promotora A-Box. A cor azul escura indica a caixa promotora B-Box. A cor amarela indica a região Poly-A média. A cor vermelha indica a região Poly-A final . . . . .	32
Figura 35 – Estruturas Secundárias inferidas para os consensos das famílias ALU encontradas . . . . .	32
Figura 36 – Exemplo da escrita de código para Gramática Livre de Contexto utilizada para gerar sequências ALU . . . . .	34
Figura 37 – Plataforma Web AURELIA para programa <i>Strings Generator</i> . . . . .	35
Figura 38 – Exemplo de <i>script</i> para Autômatos de Pilha para reconhecimento de sequências pertencentes a alguma família ALU . . . . .	36
Figura 39 – Passos para execução real do <i>Pipeline</i> . . . . .	36

Figura 40 – Procedimento para geração de palavras através da substituição de símbolos não-terminais . . . . .	39
Figura 41 – Parcela da lista de palavras finais geradas pelo programa <i>Strings Generator</i> . . . . .	40
Figura 42 – Exemplo da saída com valor <i>True</i> da simulação de um autômato . . . . .	41
Figura 43 – Site de pesquisa de sequências através do BLAST . . . . .	42
Figura 44 – Árvore filogenética dos Consensos das 6 famílias ALU construída no Clustal Omega . . . . .	45
Figura 45 – Exemplo de repetição de estados como estruturas sintáticas nos resultados dos Autômatos para sequências reconhecidas para 6 famílias ALU . . . . .	48
Figura 46 – Árvores filogenéticas, tanto do Clustal Omega como aquela inferida através das estruturas sintáticas da linguagem associada, para 5 sequências reconhecidas pelo autômato correspondente à família AluSq . . . . .	49
Figura 47 – Outras técnicas de alinhamento múltiplo de sequências e construção de árvores filogenéticas aplicadas nas 5 sequências reconhecidas pelo autômato para família AluSq . . . . .	49
Figura 48 – Árvore filogenética inferida através das estruturas sintáticas da linguagem associada, para 5 sequências reconhecidas pelo autômato correspondente à família AluSq com $n = 2$ . . . . .	51
Figura 49 – Árvore filogenética inferida através das estruturas sintáticas da linguagem associada, para 5 sequências reconhecidas pelo autômato correspondente à família AluSq com $n = 6$ . . . . .	51
Figura 50 – Árvore filogenética composta das sequências da família AluSq . . . . .	52

## Lista de tabelas

Tabela 1 – Tamanhos de amostras geradas . . . . .	41
Tabela 2 – Resultados das simulações nos Autômatos de Pilha . . . . .	42
Tabela 3 – Validação de sequências reconhecidas no Banco de Dados GenBank do NCBI, através do algoritmo BLAST. . . . .	43
Tabela 4 – 5 sequências da família AluSq, identificadas pelo número de sequência no arquivo de saída do <i>Strings Generator</i> , agrupadas através das estruturas sintáticas expressas nas palavras associadas sobre o alfabeto dos estados do autômato, com potência de repetição $n = 4$ . . . . .	48
Tabela 5 – 5 sequências da família AluSq, identificadas pelo número de sequência no arquivo de saída do <i>Strings Generator</i> , diferenciadas através das estruturas sintáticas expressas nas palavras associadas sobre o alfabeto dos estados do autômato, com potência de repetição $n = 2$ . . . . .	50
Tabela 6 – 5 sequências da família AluSq, identificadas pelo número de sequência no arquivo de saída do <i>Strings Generator</i> , diferenciadas através das estruturas sintáticas expressas nas palavras associadas sobre o alfabeto dos estados do autômato, com potência de repetição $n = 6$ . . . . .	51
Tabela 7 – Cálculo do <i>Score</i> para construção da árvore filogenética da família AluSp . . . . .	52

## **Lista de abreviaturas e siglas**

UESC	Universidade Estadual de Santa Cruz
DCET	Departamento de Ciências Exatas e Tecnológicas
PPGMC	Programa de Pós-Graduação em Modelagem Computacional em Ciência e Tecnologia
NCBI	National Center of Biotechnological Information
BLAST	Basic Local Alignment Search Tool
DDBJ	DNA DataBank of Japan
EMBL	European Molecular Biology Laboratory
DNA	Ácido desoxirribonucléico
RNA	Ácido ribonucléico
SINE	Short Interspersed Nuclear Elements
LINE	Long Interspersed Nuclear Elements
ALU	Arthrobacter luteus

## **Lista de símbolos**

$\Gamma$	Letra grega Gama maiúscula
$\lambda$	Letra grega Lambda minúscula
$\in$	Símbolo matemática de "Pertence a"
$\Sigma$	Letra grega Sigma maiúscula
$\delta$	Letra grega Delta minúscula
$\alpha$	Letra grega Alpha minúscula
$\cup$	Símbolo matemático de união

# Sumário

<b>1 – Introdução</b> . . . . .	<b>1</b>
1.1 Motivação . . . . .	2
1.2 Objetivos . . . . .	2
1.2.1 Objetivo Geral . . . . .	2
1.2.2 Objetivos específicos . . . . .	2
1.3 Bases técnicas do projeto . . . . .	3
1.3.1 Base da Genética . . . . .	3
1.3.2 Base da Ciência da Computação . . . . .	6
<b>2 – Linguagens Formais e Sequências Genômicas</b> . . . . .	<b>9</b>
2.1 Revisão de literatura . . . . .	9
2.1.1 Elementos ALU como reguladores de expressão gênica . . . . .	9
2.1.2 Sequências alvo em genomas dos Neanderthais . . . . .	12
2.1.3 Estresse diferencial em locos Alu e seus efeitos na expressão gênica	15
2.1.4 Algoritmos de busca de palavras . . . . .	17
2.1.5 Linguagens Formais: Teoria, Modelagem e Implementação . . . . .	21
2.1.6 Gramáticas formais para expressão de estruturas intermoleculares	22
2.1.7 Autômatos para busca de segmentos ALU em sequências genômicas	24
<b>3 – Pipeline para Expressão, reconhecimento e diferenciação de Sequências ALU</b> . . . . .	<b>27</b>
3.1 Construção do <i>Pipeline</i> para Expressão e Reconhecimento de Sequências ALU através de Linguagens Formais . . . . .	27
3.2 Sequências ALU em bancos de dados . . . . .	29
3.3 Linguagens livres de contexto para expressão e reconhecimento de sequências ALU . . . . .	31
3.4 Escrita de <i>scripts</i> para gramáticas, autômatos e gerador de sequências . .	34
3.5 Execução de programas: amostragem e validação de dados . . . . .	36
3.6 Formulação de proposta de métrica para distância genômica . . . . .	37
<b>4 – Resultados obtidos: execução do <i>Pipeline</i> e proposta para métrica para distância genômica</b> . . . . .	<b>38</b>
4.1 Geração de formas sentenciais . . . . .	38
4.2 Geração de palavras . . . . .	39
4.3 Amostragem de dados . . . . .	39
4.4 Reconhecimento de sequências . . . . .	40
4.5 Validação de dados . . . . .	42

4.6	Formalização de proposta de métrica para distância genômica . . . . .	45
<b>5 – Considerações finais</b>	. . . . .	<b>53</b>
5.1	Conclusões . . . . .	53
5.2	Discussão dos resultados . . . . .	54
5.3	Trabalhos futuros . . . . .	55
<b>Referências</b>	. . . . .	<b>56</b>
<b>Apêndices</b>	. . . . .	<b>59</b>
<b>APÊNDICE A – Glossário de termos de Biologia e Genética</b>	. . . . .	<b>60</b>

# 1 Introdução

Existem muitas teorias sobre a origem das espécies. Uma das mais importantes é aquela proposta pelo naturalista Charles Darwin no seu livro "A origem das espécies"(1859), no qual o autor apresenta evidências abundantes da evolução das espécies, mostrando que a diversidade biológica é o resultado de um processo de descendência com modificação, no qual os organismos vivos se adaptam gradualmente através da seleção natural. A partir desta teoria, é possível ter acesso a milhões de dados genômicos que podem ser utilizados para fazer o trabalho de caracterização de espécies, e ainda mais específica, de tipos de sequências.

Especificamente, dentro da família dos primatas e suas diferentes espécies, existe um certo número de sequências biológicas, tanto de RNA como de DNA, com fenômeno repetitivo chamadas de 'Sequências ALU' dentro da família de Elementos Nucleares de Inserção Curta (SINE) as quais, de uma forma ou outra, têm sido agentes importantes na evolução e desenvolvimento das mesmas espécies ([WATSON, 2008](#)). No presente projeto, foi desenvolvido um modelo computacional baseado na análise destas sequências em diferentes etapas, tentando estabelecer diferenças entre algumas famílias encontradas no universo ALU.

A ideia inicial de Watson, na década dos 70, era que o genoma humano continha cerca de 75.000 genes diferentes ao longo da sua estrutura, mas através de estudos vários, foi descoberto que o número real estava por volta de 50.000 genes, dos quais 65% são sequências que tem um comportamento repetitivo dentro do genoma ([WATSON, 2008](#)). Uma das sequências repetitivas mais importantes são aquelas conhecidas como 'Sequências ALU', as quais são segmentos de DNA ou RNA de aproximadamente 300 nucleotídeos (Adenina, Citosina, Guanina e Timina ou Uracilo) que, com ligeiras variações, podem ser encontradas em um grande número de lugares no genoma dos primatas e seres humanos, e que até finais do século XX, foram consideradas como DNA 'lixo' ([HÄSLER; STRUB, 2006](#)).

Para definir as diferenças entre as famílias ALU, são necessárias técnicas para interpretar os dados, e no presente projeto foram usadas três básicas: construção de linguagens formais para expressão de sequências genômicas (sequências ALU), programas computacionais para execução de um modelo que utiliza essas linguagens formais, e a definição de uma métrica para medir a distância genômica entre as famílias estudadas, através da análise das estruturas sintáticas expressas nas linguagens formais construídas.

## 1.1 Motivação

Atualmente, o mundo da genética tem muitas formas para fazer caracterização de sequências genômicas, através das diferentes análises baseadas nas suas composições e utilizando ferramentas como alinhamento de sequências, construção de árvores filogenéticas, análises proteômicas, entre várias outras. No entanto, considerando o ponto de vista computacional, temos uma grande possibilidade de desenhar modelos que possam, através da caracterização de estruturas sintáticas, explicar a diferenciação de várias famílias de sequências, baseada nas múltiplas composições de bases nitrogenadas. Tais estruturas sintáticas podem ser expressas por intermédio de linguagens formais, utilizando os símbolos já conhecidos da genética para representar as bases nitrogenadas (A, C, G, T e U).

As análises feitas para o presente projeto são baseadas no comportamento das sequências ALU presentes, na sua maioria, em diferentes espécies de primatas e que têm um comportamento repetitivo dentro do genoma além de um tamanho prático para serem geradas através de gramáticas formais, o qual simplificam pesquisas iniciais. No entanto, o objetivo foi construir um modelo que possa ser aplicado a quaisquer tipos de sequências genômicas presentes em qualquer espécie, e que através de alguma linguagem formal, forneça dados confiáveis suficientes nas métricas definidas em suas estruturas para a construção de árvores filogenéticas precisas, segundo as diferentes características presentes em tais espécies.

## 1.2 Objetivos

Na presente seção, são definidos os objetivos do presente projeto, em termos gerais quanto específicos, em cada uma das bases técnicas envolvidas na pesquisa.

### 1.2.1 Objetivo Geral

Construir um modelo computacional no qual possam ser introduzidos os dados da informação das sequências ALU de várias espécies, e ter como resultado as diferenças a nível genômico, expressas em estruturas sintáticas de alguma gramática formal.

### 1.2.2 Objetivos específicos

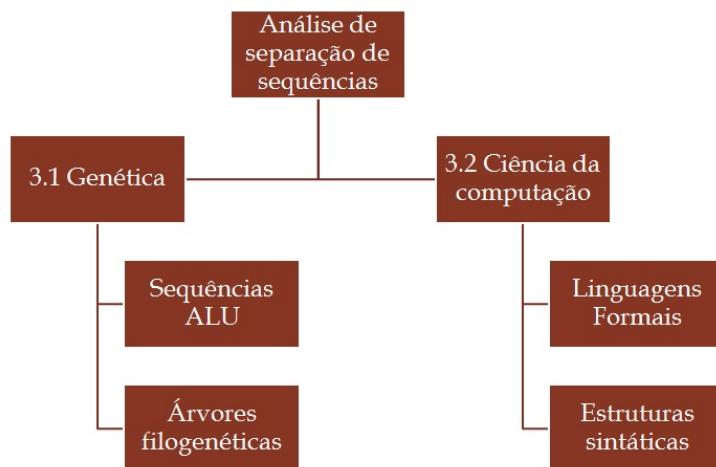
1. Construir e definir linguagens formais, as quais possam ser utilizadas para enumerar, gerar e reconhecer os diferentes símbolos e termos que fazem parte das expressões genômicas correspondentes às famílias de sequências ALU.
2. Definir uma métrica para medição da distância genômica entre as famílias ALU, expressa em estruturas sintáticas das linguagens formais utilizadas.

3. Desenhar um modelo computacional segundo o qual seja possível ingressar os dados genômicos das sequências a serem estudadas, para obter como resultado as diferenças entre elas, expressas em alguma linguagem formal.

### 1.3 Bases técnicas do projeto

O conjunto de ferramentas principais para o desenvolvimento do projeto está voltado para construção de um modelo estruturado através do qual seja possível definir uma diferenciação de sequências ALU através da sua constituição genômica. O esquema abaixo (fig. 1) mostra as ferramentas e os conceitos básicos nos quais está baseado o projeto:

Figura 1 – Bases técnicas do projeto



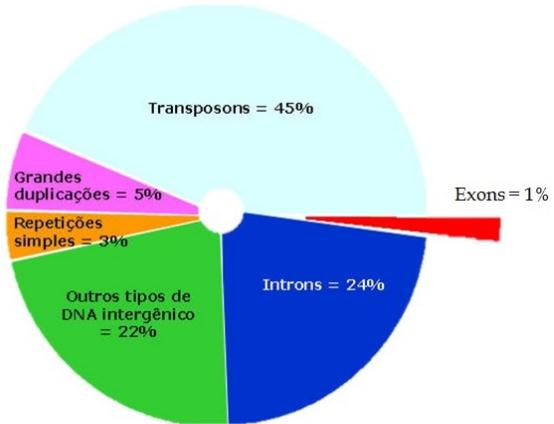
Fonte: Autoria própria

#### 1.3.1 Base da Genética

Na genética, encontramos dois conceitos fundamentais para compreender a constituição molecular que é utilizada para o desenvolvimento do modelo: as sequências ALU (sequências biológicas repetitivas) e as árvores filogenéticas (expressão gráfica da relação de proximidade entre espécies). O genoma humano tem uma estrutura dividida em diferentes tipos de sequências (fig. 2), onde os elementos repetitivos têm uma grande presença porcentual, e portanto, são considerados como agentes importantes para a natureza evolutiva.

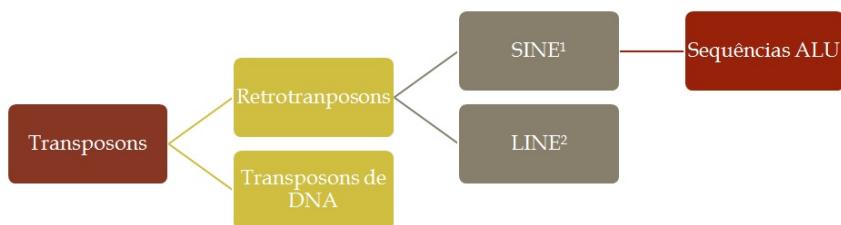
Pela sua natureza, tais elementos repetitivos (Fig. 3) têm sido considerados como alguns dos agentes mais importantes na diversificação e evolução de espécies através dos anos. Os mais importantes, segundo a porcentagem de sua presença dentro do genoma, são aqueles conhecidos como Transposons ([HäSLER; STRUB, 2006](#)).

Figura 2 – Distribuição dos diferentes tipos de sequências de nucleotídeos



Fonte: ([AMORIM; PORTELA, 2006](#))

Figura 3 – Tipos de Sequências repetitivas

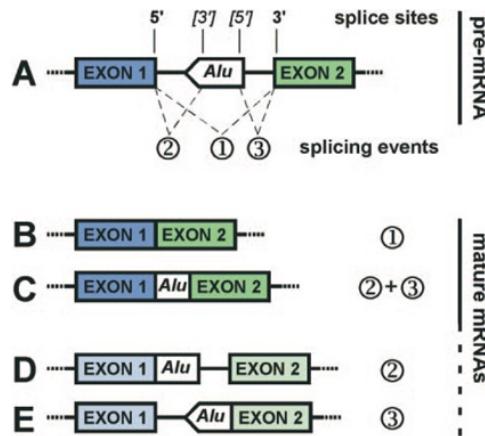


Fonte: Autoria própria

As sequências ALU são um tipo de elementos repetitivos dentro do genoma, e pertencem à grande família dos Elementos Nucleares de Inserção curta (*SINE* pela sua sigla em inglês ([HÄSLER; STRUB, 2006](#)). São os elementos repetitivos mais abundantes no genoma humano, com aproximadamente um milhão de cópias (10% do total do genoma); emergiram faz 65 milhões de anos de uma fusão do gene 7SL RNA que teve lugar no genoma dos supraprimatas, e foram amplificados ao longo do genoma humano através do efeito de retrotransposição. Eles têm sido confirmados com um papel importante dentro do desenvolvimento para diferentes tipos de mutações e doenças genéticas. Apresentam um efeito conhecido como *splicing* alternativo (fig. 4), o qual define-se como a capacidade de influenciar no processo de criação de diferentes tipos de proteínas a partir de um mesmo gene ([HÄSLER; STRUB, 2006](#)).

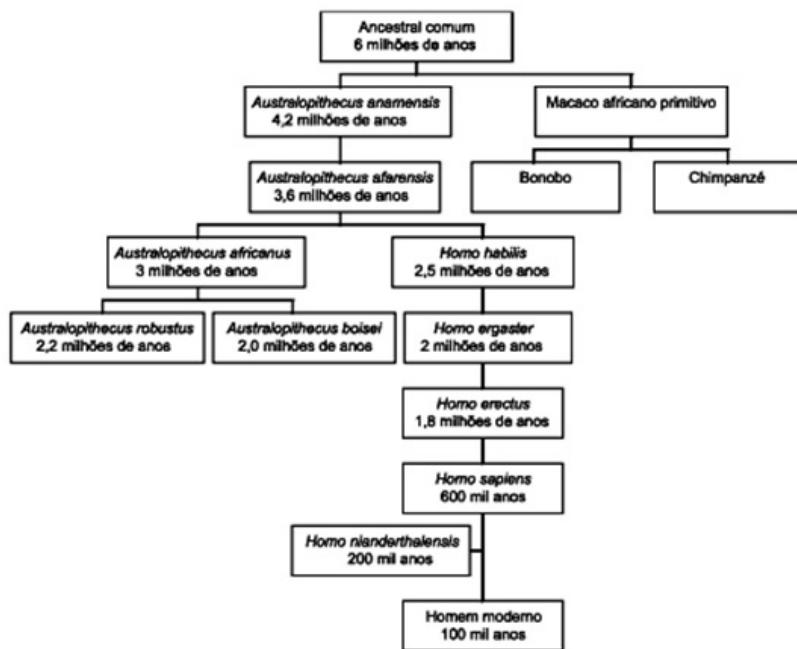
Vale lembrar que somente 2% da estrutura do genoma de fato codifica proteínas; o restante, conhecido desdenhosamente como *junk* (DNA lixo), é composto de trechos de comprimento variável, muitos deles repetidos e sem função aparente. As sequências ALU foram consideradas por muito tempo como *junk*, mas atualmente, as pesquisas tem corrigido essa noção. Como conceito complementar, e utilizando informação biológica,

Figura 4 – Possíveis eventos de splicing alternativo de uma sequência ALU

Fonte: ([HÄSLER; STRUB, 2006](#))

como a descrita acima, as árvores filogenéticas são basicamente ferramentas gráficas pelas quais é possível estabelecer relações evolutivas de diferentes espécies, através de características específicas. Para a construção de uma árvore filogenética confiável (fig. 5), é necessário definir a características dos organismos na qual será baseado o estudo, e uma métrica correspondente que trabalhe com dados estatísticos específicos ([FABRI et al., 2000](#)).

Figura 5 – Árvore filogenética para a provável evolução da linhagem humana

Fonte: ([FABRI et al., 2000](#))

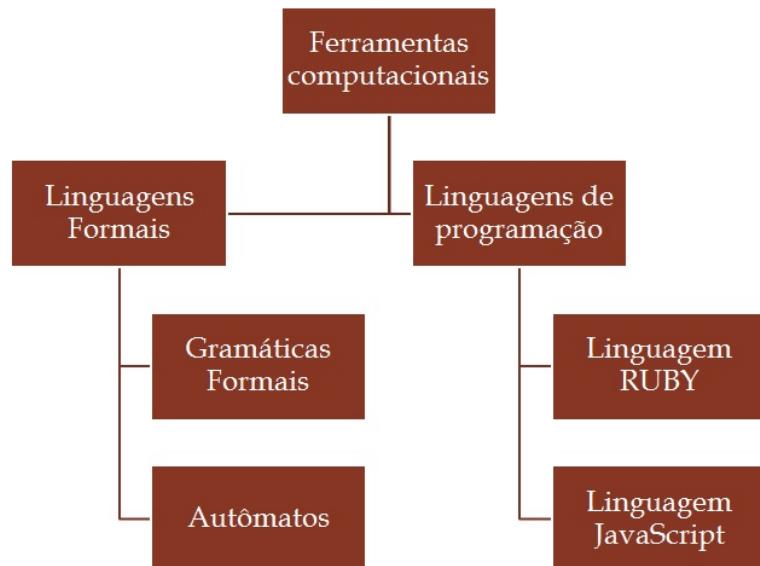
A partir de modelos computacionais orientados à bioinformática, seria possível obter informações precisas da diferenciação das sequências ALU em várias espécies de

primatas, inclusive dos seres humanos, as quais podem ser utilizadas para a construção de diferentes árvores filogenéticas.

### 1.3.2 Base da Ciência da Computação

Na base geral da Ciência da computação, são apresentadas duas ferramentas fundamentais que são utilizadas para desenvolver o modelo computacional: linguagens formais e as suas respectivas estruturas sintáticas encontradas na simulação do autômato, expressas através de uma linguagem de programação. Neste projeto, foi utilizada a linguagem RUBY, pela disponibilidade de uma plataforma baseada na teoria de linguagens formais e de livre acesso que pode ser adaptadas às nossas necessidades e finalidade desta pesquisa ([SOUZA, 2014](#)). É possível desenhar uma estruturação das ferramentas no ambiente computacional e sintático (fig. 6):

Figura 6 – Ferramentas computacionais

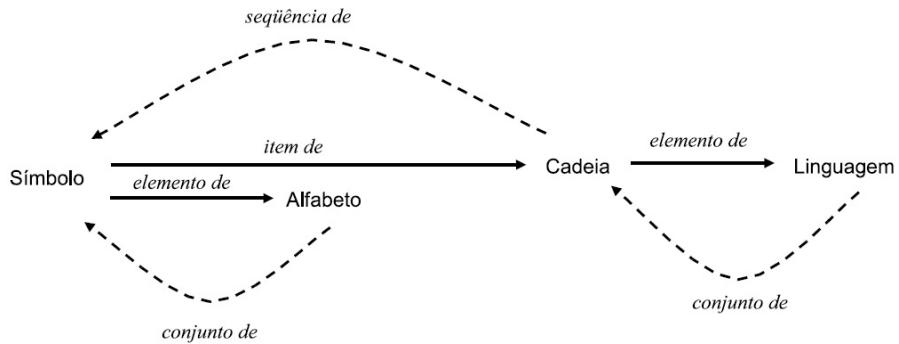


Fonte: Autoria própria

Em conceito, as linguagens Formais são estudos de modelos matemáticos que possibilitam a geração e o reconhecimento de linguagens (no sentido amplo da palavra), suas classificações, estruturas, propriedades, características, e inter-relacionamentos ([MIDENA et al., 2009](#)). Uma linguagem formal pode ser definida de duas formas: por meio de dispositivos geradores (gramáticas) ou de dispositivos reconhecedores (autômatos ou máquinas), onde cada dispositivo tem um papel importante para o contexto específico para o qual será desenvolvida a respectiva linguagem. Para cada linguagem, temos uma série de elementos básicos que a constituem (fig. 7).

Conforme a natureza dos dispositivos requeridos para a sua definição, as linguagens formais podem ser agrupadas em diferentes categorias; elas são dispostas em um

Figura 7 – Elementos de uma linguagem formal



Fonte: ([MIDENA et al., 2009](#))

esquema conhecido como *Hierarquia de Chomsky* ([MIDENA et al., 2009](#)).

- Tipo 3 ou Linguagens Regulares
- Tipo 2 ou Linguagens Livres de Contexto
- Tipo 1 ou Linguagens Sensíveis ao Contexto
- Tipo 0 ou Linguagens Enumeráveis Recursivamente

Cada uma das classes de linguagens acima numeradas, possui os seus próprios modelos de Dispositivos Geradores (Gramáticas) e Dispositivos Reconhecedores (Autômatos).

A ferramenta complementar na área de Ciência da Computação a ser utilizada é a Linguagem de programação RUBY, uma linguagem orientada a objetos, a qual tem similaridade na escrita com linguagens como: Lisp, Smalltalk, Eiffel, Perl e Python ([MATSUMOTO, 2001](#)). Esta é considerada como uma Linguagem de alto nível; foi selecionada pela disponibilidade de múltiplas plataformas de livre acesso desenhadas pelo professor Ícaro Andrade Souza da Universidade Federal de Bahia ([SOUZA, 2014](#)), prontas para utilização para efeitos do projeto. Tal plataforma foi desenhada em um projeto com objetivo de aprimorar os programas existentes baseadas no livro de Linguagens Formais: Teoria, Modelagem e Implementação ([MIDENA et al., 2009](#)), e ela está baseada nos conceitos do mesmo livro. Por exemplo, abaixo é mostrado um programa de busca da cadeia específica 'aca' dentro de uma sequência ALU, simulando um autômato finito determinístico (fig. 8).

Figura 8 – Exemplo de um autômato finito determinístico programado na linguagem RUBY aplicado en sequências ALU para busca de palavra 'aca' e o seu respectivo diagrama de transições

```

rdf = ReconhecedorDeterministico.new("q0", ["q3"])
rdf.automato.adicionarTransicao( {"[\"q0\", \"c\"] => \"q0\""} )
rdf.automato.adicionarTransicao( {"[\"q0\", \"g\"] => \"q0\""} )
rdf.automato.adicionarTransicao( {"[\"q0\", \"t\"] => \"q0\""} )

rdf.automato.adicionarTransicao( {"[\"q0\", \"a\"] => \"q1\""} )
rdf.automato.adicionarTransicao( {"[\"q1\", \"g\"] => \"q0\""} )
rdf.automato.adicionarTransicao( {"[\"q1\", \"t\"] => \"q0\""} )

rdf.automato.adicionarTransicao( {"[\"q1\", \"a\"] => \"q1\""} )
rdf.automato.adicionarTransicao( {"[\"q1\", \"c\"] => \"q2\""} )

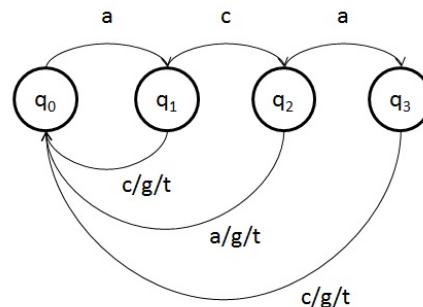
rdf.automato.adicionarTransicao( {"[\"q2\", \"c\"] => \"q0\""} )
rdf.automato.adicionarTransicao( {"[\"q2\", \"g\"] => \"q0\""} )
rdf.automato.adicionarTransicao( {"[\"q2\", \"t\"] => \"q0\""} )

rdf.automato.adicionarTransicao( {"[\"q2\", \"a\"] => \"q3\""} )
rdf.automato.adicionarTransicao( {"[\"q3\", \"a\"] => \"q1\""} )

rdf.automato.adicionarTransicao( {"[\"q3\", \"c\"] => \"q0\""} )
rdf.automato.adicionarTransicao( {"[\"q3\", \"g\"] => \"q0\""} )
rdf.automato.adicionarTransicao( {"[\"q3\", \"t\"] => \"q0\""} )

rdf.iniciar( "ggccggcgcggtggctacgcctgtaatccagcacacttggaggccgaggcggtggatcacctgaggtcaggagttcgagaccgcctggccaacatggtaaacccccgtctct"

```



Fonte: Autoria própria, baseado em ([MIDENA et al., 2009](#))

## 2 Linguagens Formais e Sequências Genômicas

No presente capítulo, é efetuada uma revisão da literatura, sobre a qual estão apoiados os conceitos técnicos e a fundamentação bibliográfica do projeto. A revisão é baseada na sua maioria na teoria de sequências ALU em diferentes espécies de primatas, assim como na sua função dentro do genoma. Fazendo a relação com as ferramentas computacionais, são revisados artigos e informações sobre algoritmos de diferentes naturezas, utilizados na busca de palavras e padrões em sequências de formas sentenciais, assim como as definições das linguagens formais nesse contexto.

### 2.1 Revisão de literatura

Nesta seção, são discutidos vários trabalhos de pesquisa feitos anteriormente nas bases do presente projeto (Genética e Ferramentas Computacionais).

#### 2.1.1 Elementos ALU como reguladores de expressão gênica

O sequenciamento inicial do genoma humano mostrou que 55 % da sua sequência de nucleotídeos é composto de elementos repetitivos (do qual, 45% são Transposons) (HÄSLER; STRUB, 2006). Existem quatro classificações principais dos Transposons nos mamíferos (fig. 9), cada uma com média de comprimento expressa em bases, número aproximados de cópias e porcentagem da presença no genoma.

Figura 9 – Classificações principais dos Transposons nos mamíferos

	Tamanho	Cópias	% Presença
LINEs	6-8 kb	850,000	21%
SINEs	100-300 bp	1,500,000	13%
Elementos retrovírus	6-11 kb 1.5-3 kb	450,000	8%
Transposons Fóseis	2-3 kb 80-3,000 bp	300,000	3%

Fonte: Autoria própria, baseado em ([CONSORTIUM, 2001](#))

Dentro das famílias dos mencionados elementos, existe a grande classificação ALU, tanto em porcentagem de presença como em importância bioquímica. Com diferentes possíveis inserções no genoma, têm sido provadas na assistência de múltiplas

proteínas, inclusive a partir de um gene único. Estão presentes em mais de um milhão de cópias. Representam até 85% da família dos SINE (*Short Interspersed Nuclear Elements*) e emergiram há 55 milhões de anos com a radiação dos primatas pela fusão das terminais 3' e 5' do gene 7SL RNA, o qual codifica a variação de RNA da Partícula de Reconhecimento de Sinais (SRP pela suas siglas em inglês) ([HÄSLER; STRUB, 2006](#)).

Um membro típico da família ALU está composto por dois braços relacionados, também conhecidos como monómeros. O elemento inteiro tem um tamanho aproximado de 300 bases nitrogenadas. É rodeado por repetições curtas que correspondem à duplicação do sítio de inserção. ([QUENTIN, 1992](#)). No fim da sequências, podemos encontrar uma região rica em Adenina (Fig. 10).

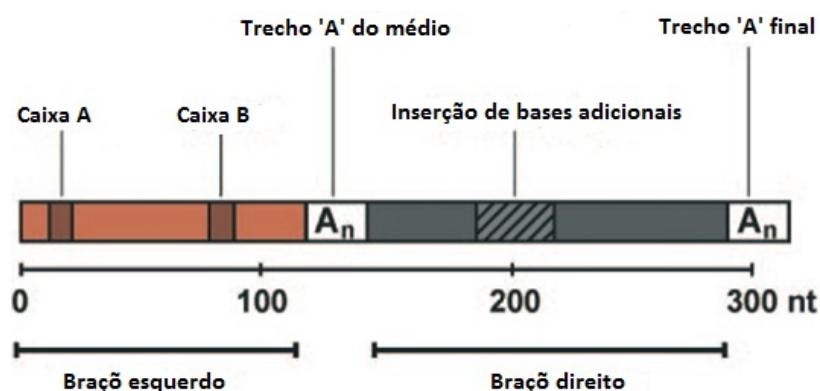
Figura 10 – Exemplo de uma sequência ALU

```
GGCCGGGCGCGGTGGCTACGCCGTAAATCCCAGCACTTGG
GAGGCCGAGGCCGGTGGATCACCTGAGGTAGGAGTTGAGA
CCAGCCTGGCCAACATGGTGAAACCCCCGTCTCTACTAAAAAT
ACAAAAAATTAGCCGGCGTGGTGGCGGGCGCTGTAATCCCA
GCTACTCGGGAGGCTGAGGCAGGAGAATCGCTTGAACCCGGG
AGGCGGAGGTTGCAGTGAGCCGAGATCGGCCACTGCACTCC
AGCCTGGGCAACAAGAGCGAAACTCCGTCTCAAAAAAA
```

Fonte: ([WATSON, 2008](#))

Elementos ALU diméricos (Fig. 11) são exclusivos dos primatas, e eles foram replicados ao longo dos genomas dos primatas através de intermediários de RNA, por um mecanismo de retrotransposição. No entanto, sua amplificação tem sido dependente do mecanismo de outros retrotranposons, pois os elementos ALU não codificam proteínas ([HÄSLER; STRUB, 2006](#)).

Figura 11 – Arquitetura dos elementos ALU



Fonte: Autoria própria, baseada em ([HÄSLER; STRUB, 2006](#))

Atualmente, há muitos trabalhos desenvolvidos com sequências ALU. Tais sequências apresentam mais de 1.100.000 cópias ao longo do genoma humano ([DRIDI, 2012](#)), pelo qual Dridi definiu seu projeto de pesquisa orientado para a investigação do papel das mesmas, tentando refutar sua caracterização como DNA lixo. Ele descobriu que alguns tipos de sequências ALU (entre eles Polimerase III e RNA B1/2) podem funcionar como reguladores de expressão morfológica, as quais têm um papel importante no desenvolvimento das espécies. Basicamente, o enfoque do estudo foi identificar a influência de tais sequências e seu efeito de retrotransposição na existência de doenças genéticas nos humanos e diferentes primatas.

Muitas funções dos elementos ALU já foram descobertas, mas as três mais importantes são:

a) *Splicing* alternativo: é um mecanismo pelo qual o uso de sítios alternativos em mRNA gera múltiplas variações de proteínas a partir de um mesmo gene. Por exemplo, o gene responsável da orientação de receptores de Drosophila (moscas das frutas) pode gerar, potencialmente, até 38.000 isomorfos diferentes, através de *splicing* alternativo. Uma das formas mais comuns para um gene adquirir novos sítios de *splicing* alternativo é um processo chamado Exonização (Fig. 4). É uma mutação das sequências intrônicas pré-existentes e que resulta no recrutamento de sequências intrônicas dentro das regiões de codificação do RNAm ([SCHMUCKER et al., 2012](#)).

b) Edição de RNA: é um processo pelo qual a sequência de nucleotídeos de moléculas de RNA é mudada durante ou após da transcrição. Entre ditas modificações, conversões de bases parecem ser o maior tipo de edição. As conversões de bases mais características são aquelas pelas quais Citosina é convertida em Uracila e a Adenosina em Inosina. A edição A-I é um mecanismo muito difundido ao longo do genoma, e mais surpreendentemente, cerca de um 90% de todas as substituições ocorrem em elementos ALU contidos em RNAm ([HÄSLER; STRUB, 2006](#)). As enzimas catalisando Adenosina a Inosina são conhecidas como Desaminases de Adenosina (ADAR) e nos seres humanos estão presentes em 3 genes diferentes (ADAR1, ADAR2 e ADAR3) ([SCHAUB; KELLY, 2002](#)).

c) Influência na tradução de proteínas: Tem sido encontrado frequentemente que RNA ALU, transcrito de elementos ALU, está presente no citosol de células de primatas. Elementos ALU contêm as caixas internas A e B do RNA polimerase III, mas são muito fracas como para levar a cabo o processo de transcrição dos elementos ALU ([HÄSLER; STRUB, 2006](#)). Embora sendo os elementos ALU parte de uma grande família de sequências, cada uma delas encontra-se em um contexto diferente. Cada um dos *loci* ALU apresentam um padrão único de expressão em diferentes linhas celulares nos seres humanos, em resposta a diferentes tipos de estresse. O contexto de sequências circundante a cada elemento ALU controla sua regulação de transcrição de forma única

(LI; SCHMID, 2001).

Mais focados nas desordens genéticas, alguns cientistas fazem uma interpretação da relação de diferentes tipos de sequências ALU presentes em primatas, e de forma geral, o desenvolvimento de diferentes doenças, como por exemplo a obesidade nos seres humanos (KUEHNEN; KRUDE, 2012).

Desde o descobrimento dos elementos transponíveis no ano 1940, novas técnicas de análise foram criadas para identificar a distribuição e regulação destes elementos dentro do genoma das plantas, de vertebrados e de invertebrados. Em específico, nos seres humanos existe uma relação direta entre esses elementos e o fenômeno da evolução e da diversidade genética.

Muitas espécies de primatas apresentam elementos ALU únicos e que os diferenciam geneticamente de outras espécies, baseada nos contextos das sequências (MARTINS et al., 2015). Atualmente, os autores trabalham estudando diferentes espécies de macacos, tentando distingui-las com base na estrutura dos elementos ALU, e de fato, foi constatado que existem muitas repetições (especificamente SINE AluSx3 e AluSc8) que estão presentes só em uma espécie, neste caso nos macacos capuchinhos. Ainda dentro das raças de capuchinhos (*Cebus* e *Sapajus*), temos uma estrutura genômica variável.

Com base nesses resultados, Schneider e Sampaio no ano 2015 começaram uma pesquisa sobre a evolução dos primatas do novo mundo, e a sistemática envolvida. Eles estudam a taxonomia dos primatas de propostas originárias dos anos 80, considerando a morfologia demonstrada em muitos estudos baseados nos dados moleculares para o esclarecimento da filogenia dos macacos do novo mundo. Com todo este trabalho, o objetivo deles é definir a relação evolutiva dos primatas, e como este fenômeno também é demonstrado no desenvolvimento da diversidade genética nos seres humanos (SCHNEIDER; SAMPAIO, 2015).

Para o estudo baseado no genoma de seres humanos, Kelley e sua equipe explicam o desempenho dos elementos ALU transponíveis, e a sua influência na evolução das redes regulatórias do genoma humano. O estudo deles é focado no reconhecimento dos sites de conexão do RNA e a produção de diferentes tipos de proteínas baseados no *splicing* alternativo provocado pelo efeito das diferentes sequências ALU presentes ao longo da estrutura genômica (KELLEY et al., 2014).

### 2.1.2 Sequências alvo em genomas dos Neanderthais

As sequências ALU, sendo restritas em espécies de primatas, são marcadores moleculares para estabelecer diferenças evolutivas, tanto no comportamento social como em características morfológicas, dos organismos onde possam ser encontradas. Por exemplo, considerando a proximidade genética entre seres humanos e Neanderthais,

as divergências proteicas resultando em mutações físicas têm origem em genes afetados pelo contexto ALU. Para estudos baseados em comparações dos seres humanos e outras espécies de primatas, existem muitas referências importantes. As características morfológicas típicas dos Neandertais apareceram pela primeira vez nos registros dos fósseis europeus, que aproximadamente remontam 400,000 anos. Alguns cientistas têm encontrado muito poucos argumentos a favor de inter-relacionamento genético entre as duas espécies (GREEN et al., 2010). Segundo estudos feitos recentemente, foi encontrado que a porcentagem da inserção dos genes dos Neandertais no genoma dos Europeus Paleolíticos pode ser estimada entre 0,02% e 0,09% (CURRAT; EXCOFFIER, 2004).

No ano 1991 foi feita a primeira extração do DNA de um espécime de Neandertal encontrado perto de Düsseldorf, Alemanha no ano 1856. Através de análises filogenéticas e comparações feitas com DNA mitocondrial de seres humanos (Fig. 12), onde os pontos indicam as bases consistentes no alinhamento e os asteriscos indicam as regiões conservadas entre a sequência referência e as sequências das amostras, foi determinado que não tinham alta similaridade, portanto, as espécies não poderiam apresentar inter-relacionamento (KRINGS et al., 1997).

Figura 12 – Sequências de DNA de clones derivados da região de controle mitocondrial de um fóssil de Neandertal

refseq	ACAGCAATCAACCCCTCAACTATCAGCATCAACTGAACTCCANAGGCCACCCCT-CACCCAC
A10.1	.....T.....T.....A.....A.G...T.R.....
A10.2	.....T.....G.....T.....A.....A.G...T.G...T.R.....
A10.3	.....T.....G.....T.....A.....A.G...T.R.....
A10.4	.....T.....G.....T.T.....A.....A.G...T.R.....
A10.5	.....T.....G.....T.....A.....A.G...T.R.....
A10.6	.....T.....G.....T.....A.....A.G...T.R.....
A10.7	.....T.....G.....T.....G.....A.....A.G...T.R.....
A10.8	.....T.....G.....T.....A.....A.G...T.R.....
A10.9	.....T.....G.....T.....A.....A.G...T.R.....
A10.10	.....T.....G.....T.....A.....A.G...T.R.....A.....
A10.11	.....T.....G.....T.T.....A.....A.G...T.R.....
A10.12	..A.....T.....G.....T.....A.....A.G...T.R.....
A10.13	.....T.....T.G.....T.T.....A.....A.G...T.R.....
A10.14	.....T.....G.....T.....A.....A.G...T.R.....
A10.15	.....T.....G.....T.....A.....A.G...T.R.....
A10.16	.....T.....G.....T.....A.....A.G...T.R.....
A10.17	.....T.....G.....T.....A.....A.G...T.R.....
A10.18	.....G.....T.....A.....A.G...T.R.....
A17.1	.....T.....G.....T.....A.....A.G...T.R.....
A17.2	.....T.....G.....T.....A.....A.G...T.R.....
A17.3	.....T.....G.....T.....T.RT.....A.G...T.R.....
A17.4	.....T.....G.....T.....A.....A.G...T.R.....
A17.5	.....T.....G.....T.....T.RT.....A.G...T.R.....
A17.6	.....T.....G.....T.....A.....A.G...T.R.....
A17.7	.....T.....G.....T.....A.....A.G...T.R.....
A17.8	.....T.....G.....T.....A.....A.G...T.R.....
A17.9	.....T.....T.G.....T.....A.....A.G...T.R.....T.....
A17.10	.....T.....G.....T.....A.....T.....A.G...T.R.....
A17.11	.....T.....G.....T.....A.....A.G...T.R.....
A17.12	.....T.....G.....T.....A.....A.G...T.R.....
B11.1	.....T.....G.....T.....A.....A.G...T.R.....
B11.2	.....T.....G.....T.....A.....A.G...T.R.....
B11.3	.....T.....G.....T.....A.....A.G...T.R.....
B11.4	.....T.....T.....T.....G.....A.....A.G...T.R.....
B11.5	.....T.....T.....T.....G.....A.....A.G...T.R.....T.....
B11.6	.....T.....T.....T.....G.....A.....A.G...T.R.....T.....
B11.7	.....T.....T.....G.....T.....A.....A.G...T.R.....T.....
B11.8	.....T.....CT.....T.....G.....A.....A.G...T.R.....
B11.9	.....T.....T.....T.....G.....A.....A.G...T.R.....
B11.10	.....T.....T.....G.....T.....A.....A.G...T.R.....
B11.11	.....C.....T.....G.....T.....A.....A.G...T.R.....T.....
B11.12	.....T.....G.....T.....A.....A.G...T.R.....
B11.13	.....T.....G.....T.....A.....A.G...T.R.....
B11.14	.....T.....G.....T.....A.....A.G...T.R.....
C16.1	.....T.....A.....A.G...T.R.....
C16.2	.....T.T.....A.....A.G...T.R.....
C16.3	.....T.....A.....A.G...T.R.....
C16.4	.....T.....A.....A.G...T.R.....
C16.5	.....T.T.....A.....A.G...T.R.....

Fonte: (KRINGS et al., 1997)

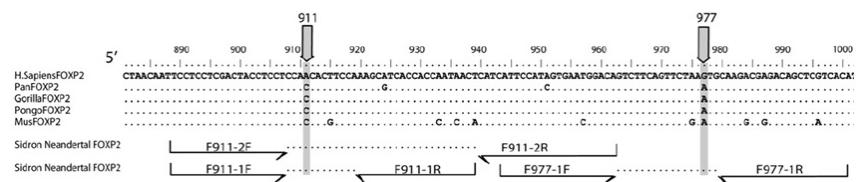
Existem muitos estudos feitos com diversos tipos de Neandertais (Fig.13). Por exemplo, o DNAmt do Neandertal do Cáucaso (espécimen de há 29.000 anos) tem uma divergência de apenas 3,49% com o DNAmt do Neandertal Feldhofer (Alemanha), o que os coloca no mesmo ramo das árvores filogenéticas. ([OVCHINNIKOV et al., 2000](#)).

Figura 13 – Sítios variáveis nas sequências de DNA dos fragmentos obtidos através de sequenciamento direto

Fonte: (OVCHINNIKOV et al., 2000)

Existem vários exemplos de estudos feitos com diferentes genes, para tentar esclarecer o relacionamento entre as espécies. O gene FOXP2 é um dos maiores responsáveis pela aquisição da linguagem e o desenvolvimento da fala. É considerado entre 5% das proteínas melhor conservadas no genoma dos mamíferos. Os seres humanos divergem na constituição do gene FOXP2 em dois sítios do Exon 7 (911 e 977) (Fig. 14), segundo a comparação com o genoma dos chimpanzés ([KRAUSE et al., 2007](#)).

Figura 14 – Alinhamento de sequências de posições 880-1000 de nucleotídeos do gene FOXP2. As substituições de nucleotídeos na linhagem humana são indicadas pelas setas cinzentas. Posições iguais no alinhamento são indicadas por pontos. Os três pares de primers utilizados para recuperar as duas substituições dos neandertais Sidron são indicados por setas



Fonte: (KRAUSE et al., 2007)

Como mostrado acima, o gene FOXP2 pode representar um dos motivos evolutivos diferenciais entre espécies de primatas, assim como vários genes controlando mais outras características morfológicas e de comportamento.

### 2.1.3 Estresse diferencial em locos Alu e seus efeitos na expressão gênica

Enquanto elementos repetitivos ALU em humanos são considerados membros de uma família grande de genes, cada um deles encontra-se em um contexto diferente que modifica sua transcrição (Fig. 15). Esse contexto pode variar segundo o tipo de estresse ao qual estejam expostos ditos elementos (LI; SCHMID, 2001).

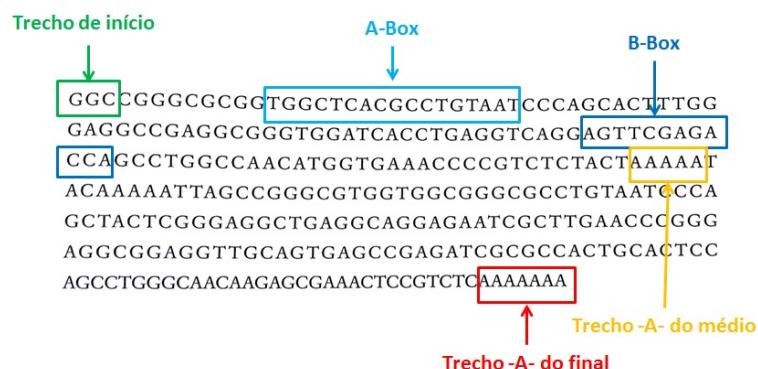
Figura 15 – Contexto de duas sequências ALU, encontradas no gene precursor de Beta-amiloide (produtos da Proteína Precursora Amiloide), os quais são peptídeos observados no cérebro de pacientes com doença de Alzheimer

1 tgctaaatct aagaacttta atttttagat attatgatct catctctaca attttgaatt  
61 tcatgtccaa taaaagttcc ttactcttctt tttttttttt tgagacggag tctcgctcgt  
121 tcgcggcaggc tggagtgcag tggcgcgatc tcggctact tcaagctcag cctccgggt  
181 tcacccatt ctccgcctc agcccccga gtagctggga ctacaggcgc ccggccacgac  
241 gcccgctaa tttttgtat ttttagttaga gacggggtt caccgttta gccaggatgg  
301 tgttgatctc ctgacccgt gatccggcccg cctcagccctc ccaaagaaaaa gtccctcact  
361 cttaaagtgtt cctcccttccc cccagggttg gcttcatggg catgcaaccc tggagatgt  
421 cacaggccct gcgggtggag gagccccatg ctgggtttaa cgctctgca ttgcacatctt  
481 aaaatctta atttaatttt ttttctttttt ttttgagggtg gagtctcgct ctgtcgccca  
541 ggctggagtg caatggcaca atcttggctc actgcaacctt ccgcctccca ggttcaagcg  
601 atttccctgc ctcagcccttggatgtctggattacagg caggatataac cacgcgtccgg  
661 taatttttgtc atttttagatagatgggggg ttccacatgttggccaggc tggcttagaa  
721 ctccgtaccct cagggtatct gcccacccatgg gcctctaaatgtctggat tacaggccatg  
781 agccaccagg cccggccatc aaatctttaa taatgtaaaca aagggtctca cgttgcatt  
841 ttgcgtggaa ctctgcaaga tttgttagcttggaccacgt ttctctttgc attcagatac  
901 ctctt

Fonte: (INFORMATION, 2015a)

Vale lembrar que todos os elementos SINE eucarióticos foram derivados ancestralmente dos genes do RNA da Partícula de Reconhecimento de Sinais (em roedores e primatas). Assim como esse gene, todos os elementos SINE contêm caixas internas (A e B) promotoras para RNA polimerase III (Fig. 16) ([LI; SCHMID, 2001](#)).

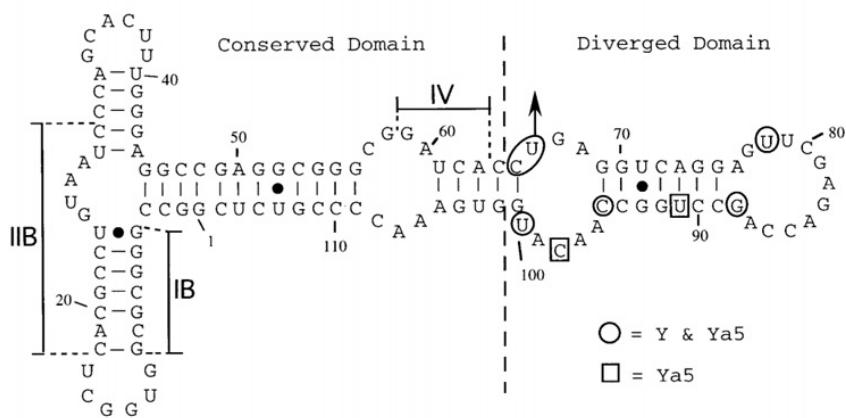
Figura 16 – Exemplo de uma sequência ALU de uma longitude de 290 pares de bases com uma terminação de 7 bases de Adenina e suas principais seções, incluindo as caixas internas



Fonte: (WATSON, 2008)

Um conceito importante a considerar é a metilação. A metilação em massa de repetições ALU em certas linhas celulares e tecidos pode reprimir as suas atividades modelos. A atividade promotora de várias unidades de transcrição de Polimerase III é estimulada ou ainda direcionada por sequências flanqueadoras (CHESNOKOV; SCHMID, 1996). RNA ALU é instável, mas pode ser dirigido para uma transcrição do monómero esquerdo estável conhecida como RNA ALU citoplasmático (scAlu), através do processamento do RNA 3'. Tais transcrições interagem com a sub unidade de união RNA ALU da Partícula de Reconhecimento de Sinais (SPR), conhecido como SRP9/14 (Fig.17) (SARROWA et al., 1997).

Figura 17 – União de SRP9/14 com sequência ALU no seu monómero esquerdo. Mudanças de nucleotídeos que acompanharam a evolução de elementos ALU é mostrada com círculos para AluY e com quadros para AluYa5. Os sítios de união com SRP9/14 está indicados por IIB, IB e IV. A região conservada está presente em AluSx, AluY, e AluY5



Fonte: (SARROWA et al., 1997)

O dado anterior (fig. 17) fala da conexão entre os elementos ALU e seu contexto. Mutações que acompanham ao RNA ALU no seu processo de evolução levou a mudanças em uma estrutura conservada também encontrada em Partículas de Reconhecimento de Sinais do RNA. Elas estão associadas com a desestabilização termodinâmica e afinidade reduzida dos elementos ALU por SRP9/14.

Comparadas com o seu extraordinário número de cópias, elementos ALU em humanos geralmente têm um baixo nível de expressão (SCHMID, 1998). Apesar de ter estrutura bem definida com as caixas internas promotoras necessárias para a tradução, não têm sido comprovados como agentes independentes de tradução. No entanto, transcrição ALU é fortemente influenciada por uma variedade de tipos de estresse celular. Os tipos de estresse mais comuns são: Choques de calor, Exposição de celular a cicloheximida (inibidor de síntese de proteínas) e Infecções virais (LIU et al., 1995).

Embora esteja comprovado através de análises de diferentes sequências que

vários membros ALU são transcritos tanto em condições normais como de estresse, nas últimas existe uma estimulação maior da transcrição (LIU et al., 1995). Vale lembrar que esse efeito está presente em outras espécies como roedores, coelhos e bichos da seda (LI; SCHMID, 2001).

Vale lembrar que além da expressão e regulação ALU pelo estresse celular, transcrição é também um pré-requisito para a retrotransposição de sua linha germinal. 'Genes ALU' podem ser definidos como sendo 'fonte' que codificam ativamente novos membros da família. Existe forte evidência de que algumas sequências LINE1 providenciam fatores que são essenciais para retrotransposição ALU. Estresse celular pode induzir retrotransposição, pelo qual existe possibilidade de que alguns tipos desses estresses induzam transcrição de elementos LINE1 e genes ALU (LI; SCHMID, 2001).

#### 2.1.4 Algoritmos de busca de palavras

Tentativas para encontrar padrões de símbolos têm sido desenvolvidas por muito tempo através de programas e algoritmos de busca de palavras. Existe uma grande variedade deles, para busca de palavras ou *strings* dentro de um texto, cada um com diferentes métodos e estratégias variáveis, para trabalhar segundo o contexto. Os três algoritmos mais conhecidos são: Knuth-Morris-Pratt, Boyer-Moore e Aho-Corasick.

O algoritmo Knuth-Morris-Pratt, também conhecido como algoritmo KMP, procura a ocorrência de uma palavra "W" dentro de um texto principal ou *text string* denotado por "S". Analisando cada símbolo desde o início do texto, emprega a observação de que quando ocorre uma incompatibilidade, o mesmo texto oferece suficiente informação para determinar onde poderia iniciar o seguinte pareamento correto ou *match* (KNUTH et al., 1977). Um algoritmo de busca de palavras procura encontrar o índice de início "m" no texto "S[]" que seja compatível com a palavra de busca "W" (KNUTH et al., 1977). Os elementos que compõem o algoritmo são:

1. W = palavra buscada
2. S = texto de busca
3. m = posição em "S" onde começa a primeira possibilidade de *match* para "W"
4. i= denotando o índice do caractere atualmente considerado em "W"

Considere-se uma sequência de bases nitrogenadas **atgttgatctggaccatgatcg**, a qual seria nosso texto "S" e uma palavra "W" composta por **atgat**, a qual será procurada no texto "S" Na figura 18, temos a presença de todos os elementos do algoritmo: a construção de uma tabela parcial (denotada por "T"), que indica cada reconhecimento do algoritmo e estabelece un valor *score* [i], pelo qual se houver um *mismatch* (cor vermelha), infere o início de um novo *match* (cor azul), pulando uma quantidade [i] de

Figura 18 – Exemplo de busca de palavras segundo o algoritmo Knuth-Morris Pratt

	1	2	
m:	0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2		
S:	a t g t t g a t c t g g a c c a t g a t c g c		
W:	a t g a t		
i:	0 1 2 3 4		

	1	2	
m:	0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2		
S:	a t g t t g a t c t g g a c c a t g a t c g c		
W:	a t g a t		
i:	0 1 2 3 4		

...

	1	2	
m:	0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2		
S:	a t g t t g a t c t g g a c c a t g a t c g c		
W:		a t g a t	
i:		0 1 2 3 4	

Fonte: Autoria própria

símbolos sem possibilidade alguma de serem aceitos (KNUTH et al., 1977). A eficiência do algoritmo está baseado em uma complexidade de  $O(n)$ , onde ‘n’ é o tamanho do texto “S” e O é a notação O-Grande, a qual é uma função matemática para calcular os recursos para a execução do algoritmo.

No segundo algoritmo, o Boyer-Moore é um eficiente método de busca de palavras que é considerado o padrão de referência na literatura de buscas práticas. O algoritmo pre-processa a palavra a ser procurada (o padrão), mas não a cadeia onde será procurada (o texto). Ótimo para aplicações onde o padrão é mais curto que o texto (BOYER; MOORE, 1977). A ideia básica por trás do algoritmo é que mais informação é obtida comparando o padrão desde a direita que desde a esquerda.

O caractere no Texto na posição correspondente ao caractere mais à direita no padrão é denotado como *Char* ou *C*. Existe um valor delta que, se tivessemos ocorrência do *char* no padrão, é calculado matematicamente da seguinte forma:

$$\delta = (C) - (\text{ocorrência da } C \text{ no padrão})$$

Consideremos a mesma sequência de bases nitrogenadas **atgttgatctggaccat-gatcg**c como nosso texto “T” e um padrão “P” composto por **atgat**, o qual será procurada no texto “T” (fig. 19). Também temos a construção de uma tabela parcial “T”, que indica onde deve-se procurar o início de um novo *match*. O algoritmo procura o primeiro

Figura 19 – Exemplo de busca de palavras segundo o algoritmo Boyer-Moore

The figure illustrates the Boyer-Moore string search algorithm through three stages of comparison between a Text (T) and a Pattern (P).

- Stage 1:** The pattern "atgat" is compared against the text starting at index 4. A mismatch is found at index 5 (text '4' vs pattern '5'). The pattern is shifted to the right, starting its next comparison at index 6.
- Stage 2:** The pattern "atgat" is compared against the text starting at index 6. A mismatch is found at index 7 (text '7' vs pattern '6'). The pattern is shifted to the right, starting its next comparison at index 8.
- Stage 3:** The pattern "atgat" is compared against the text starting at index 8. All symbols match (text '8', '9', '0' vs pattern '8', '9', '0'). A 'match' is declared.

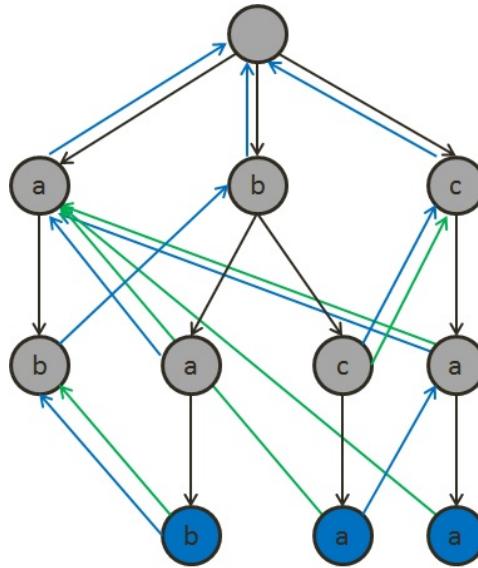
Fonte: Autoria própria

símbolo da direita do padrão e compara com a locação correspondente no texto. Se houver *match*, percorre e analisa todos os símbolos do padrão de esquerda para direita até consumir todos os símbolos. Se encontrar algum *mismatch* no processo, como mostrado na tabela (fig. 19), movimenta o padrão para direita o número de espaços correspondentes à quantidade de símbolos comparados, e repete o processo progressivamente até encontrar um *match* total.

Finalmente, o Aho-Corasick é um algoritmo de busca de palavras que funciona algo assim como uma verificação dentro de um dicionário de palavras preestabelecidas. Localiza elementos de um finito número de palavras (ou “dicionário”) dentro de um texto de entrada. Compara todos os padrões simultaneamente, procurando as palavras do dicionário inteiro. Poderiam existir resultados repetidos, pois todas as coincidências são encontradas se cada subpalavra concorda. (por exemplo, com um dicionário **a**, **aa**, **aaa**, **aaaa** e a palavra procurada é **aaaa**). O algoritmo constrói uma máquina finita de estados (**AHO; CORASICK, 1975**).

Existe um “dicionário” onde encontram-se as palavras chave. No seguinte exemplo será denotado como **K**. A palavra que será o padrão para a busca é conhecido como texto; é onde serão procuradas subpalavras que coincidam com as palavras chave. Considere-se um dicionário **K = a, ab, bab, bc, bca, c, caa**. Essas palavras exatas poderiam ser subpalavras de cadeias maiores (bab, bcb, bca, bcc, bcab, etc). A melhor forma de entender as transições do Aho-Corasick é um diagrama de estados.

Figura 20 – Diagrama de estados no funcionamento geral do algoritmo Aho-Corasick

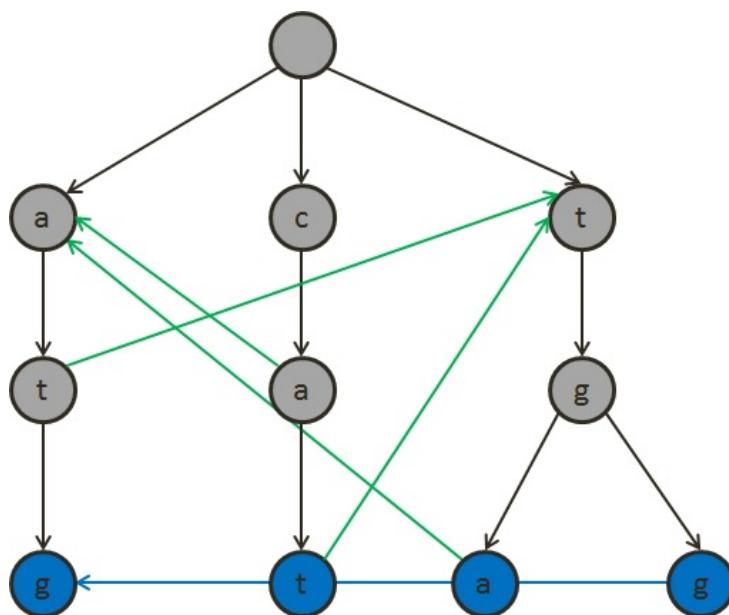


Fonte: Autoria própria

No diagrama de estados (fig. 20), os nodos azuis são as palavras em  $K$ , os demais são nodos cinza. Setas negras são transições filhas. Setas azuis são sufixos de palavras no dicionário. Setas verdes são transições alternativas.

Considere-se a mesma sequência de bases nitrogenadas **atgttgatctggaccatgtcg** a qual seria nosso texto e um dicionário  $K$  composto por uma série de palavras chave (fig. 21). ( $K = \text{atg, tga, tgg, cat}$ ).

Figura 21 – Diagrama de estados correspondente às palavras chave



Fonte: Autoria própria

Figura 22 – Exemplo de tabelas de busca de palavras para o algoritmo Aho-Corasick

T:	a	t	g	t	t	g	a	t	c	t	g	g	a	c	c	a	t	g	a	t	c	g	c
K:	a	t	g		t	g	a		t	g	g		c	a	t								
	↑	↑	↑		↑	↑	↑		↑	↑	↑		↑	↑	↑		↑	↑	↑				

T:	a	t	g	t	t	g	a	t	c	t	g	g	a	c	c	a	t	g	a	t	c	g	c
K:	a	t	g		t	g	a		t	g	g		c	a	t								
	↑	↑	↑		↑	↑	↑		↑	↑	↑		↑	↑	↑		↑	↑	↑				

Fonte: Autoria própria

A tabela parcial “T” (fig. 22) indica onde deve-se procurar o início de um novo *match*. Como mostrado, todas as palavras são procuradas simultaneamente, o que demonstra uma eficiência maior quando o objetivo da busca realizada são múltiplas palavras de comprimento curto em relação ao texto onde estesjam sendo procuradas.

Como a maioria das ferramentas computacionais, a eficiência de tais algoritmos depende do contexto de trabalho. Por exemplo, para busca padrões de comprimento longo em textos maiores (como por exemplo, busca de sequências biológicas específicas dentro de genomas completos), o método mais apropriado é aquele implementado pelo Knuth-Morris-Pratt, devido a sua natureza de desconsiderar rapidamente opções erradas. No entanto, o Boyer-Moore e o Aho-Corasick desenvolvem-se com alta eficiência na busca de padrões curtos, tipo subsequências ou palavras chaves, os quais são exemplo clássicos do uso de tais algoritmos (BOYER; MOORE, 1977).

### 2.1.5 Linguagens Formais: Teoria, Modelagem e Implementação

A área das Linguagens Formais concentra o estudo de modelos matemáticos que possibilitam a especificação e o reconhecimento de linguagens (no sentido amplo da palavra), suas classificações, estruturas, propriedades, características e inter-relacionamentos (MIDENA et al., 2009). A importância dessa teoria na Ciência da Computação é dupla: apoia outros aspectos teóricos da Ciência da Computação (como decidibilidade, computabilidade e complexidade computacional) e fundamenta diversas aplicações computacionais tais como processamento de linguagens, reconhecimento de padrões, modelagem de sistemas.

Uma linguagem formal é um conjunto de cadeias sobre um alfabeto. Os elementos que definem uma linguagem são:

1. Símbolo: é o elemento básico de menor tamanho que forma parte da uma linguagem. É uma representação gráfica, única e indivisível, que tem um significado diferenciado

de acordo ao contexto onde ele é utilizado.

2. *Alfabeto*: é um conjunto finito e não-vazio de símbolos.

3. *Cadeia*: em um alfabeto, é uma sequência finita de símbolos deste alfabeto.

Quando consideradas como conjuntos, diversas operações podem ser aplicadas sobre linguagens, entre elas:

1. *Intersecção*: define-se a intersecção de dois conjuntos A e B como sendo a coleção de todos os elementos comuns aos dois conjuntos.

2. *Diferença*: Define-se a diferença entre dois conjuntos A e B (nesta ordem) como sendo o conjunto formado por todos os elementos de A não-pertencentes ao conjunto B.

3. *Complementação*: Define-se a complementação de um conjunto A em relação ao B, como sendo o conjunto de todos os elementos de B que não pertencem a A.

Para uma linguagem formal (finita ou infinita) ser definida com todos seus elementos, existem dois tipos básicos de formalismos matemáticos:

1. *Dispositivos Geradores*: conjunto de regras que podem ser empregadas para gerar as cadeias de uma linguagem.

2. *Dispositivos Reconhecedores*: recebem uma cadeia qualquer e retornam um valor que informa se ela pertence ou não à linguagem.

Os dispositivos geradores e reconhecedores são conhecidos também como Gramática e Autômatos, respectivamente.

### 2.1.6 Gramáticas formais para expressão de estruturas intermoleculares

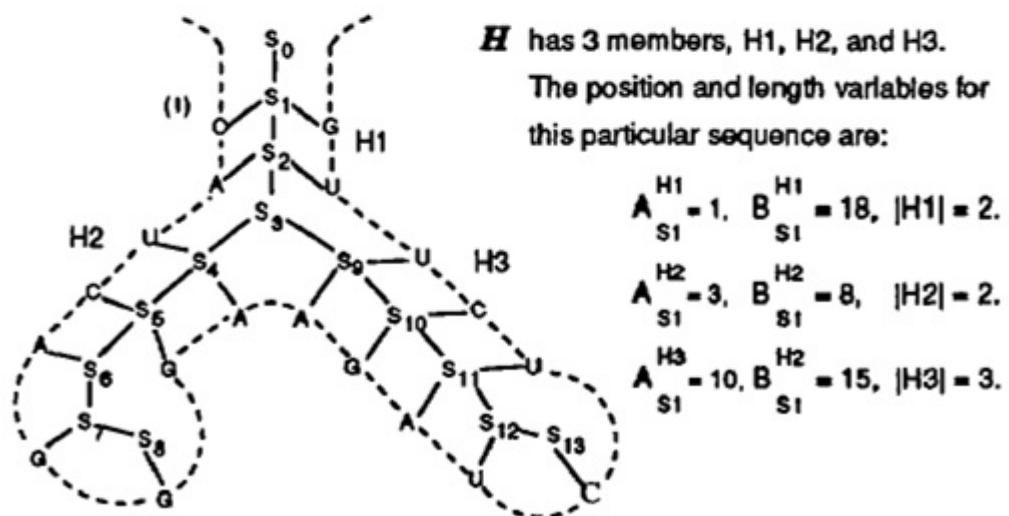
Gramáticas formais podem ser utilizadas para modelar formas gerais de estruturas intramoleculares, tais como estruturas secundárias de ácidos nucléicos ([SEARLS, 1995](#)).

Searls tinha sugerido, inicialmente na década dos 90, que derivações e árvores de derivação de gramáticas formais simples poderiam ser usadas para modelar estruturas secundárias (fig. 23). Muitos trabalhos de pesquisa foram desenvolvidos utilizando este modelo baseado na teoria de linguagens formais, empregando técnicas de aprendizado de máquina para induzir formas estocásticas de tais gramáticas no conjuntos de sequências de exemplo ([GRATE et al., 1994](#)).

É interessante notar o fato que as forças que fazem a união das fitas de DNA são basicamente aquelas envolvidas na criação de estruturas secundárias no RNA. Embora seja assim, não temos uma forma óbvia de usar uma gramática para descrever as

Figura 23 – Regras de produção da Gramática e correspondente árvore de derivação para sequência de RNA **caucagggaagaucucuug**

$$P = \{ \begin{array}{ll} S_0 \rightarrow S_1, & S_7 \rightarrow G S_8, \\ S_1 \rightarrow C S_2 G, & S_8 \rightarrow G, \\ S_1 \rightarrow A S_2 U, & S_8 \rightarrow U, \\ S_2 \rightarrow A S_3 U, & S_9 \rightarrow A S_{10} U, \\ S_3 \rightarrow S_4 S_9, & S_{10} \rightarrow C S_{10} G, \\ S_4 \rightarrow U S_5 A, & S_{10} \rightarrow G S_{11} C, \\ S_5 \rightarrow C S_6 G, & S_{11} \rightarrow A S_{12} U, \\ S_6 \rightarrow A S_7, & S_{12} \rightarrow U S_{13}, \\ S_7 \rightarrow U S_7, & S_{13} \rightarrow C \end{array} \}$$



Fonte: ([GRATE et al., 1994](#))

dependências dentro de uma fita dupla de DNA, mesmo que nas estruturas secundárias de RNA ([SEARLS, 1995](#)). A intuição básica por trás da ideia de linguagens de clivagens é que uma estrutura secundária intramolecular pode ser convertida em intermolecular por recortes simples ( $\delta$ ) em pares de bases, sendo expressa em uma gramática como segue:

$$G = (N, \Sigma, P, S)$$

P é uma série de produções onde:  $(N \cup \Sigma)^* N (N \cup \Sigma)^* \times (N \cup \Sigma \cup \delta)^*$

Para colocar ditas gramáticas regulares de clivagens no contexto biológico, temos que considerar as enzimas de restrição, a quais são agentes genéticos que literalmente cortam a fita de DNA em um sítio específico chamado de “sequência de reconheci-

mento”. Por exemplo a enzima **MboI** corta justo antes da sequência de reconhecimento **gatc**. Uma gramática para representar cortes no DNA é formulada segundo a estrutura geral das gramáticas formais (fig. 24).

Figura 24 – Gramática para produção de cortes em uma fita de DNA, onde  $\Sigma = a, c, g, t$ , o símbolo  $\delta$  representa o símbolo de corte e o símbolo  $\epsilon$  representa a palavra vazia

$$\begin{aligned} G : S &\rightarrow aS \mid cS \mid gS \mid tS \\ S &\rightarrow \delta gatc S \\ S &\rightarrow \epsilon \end{aligned}$$

Fonte: ([SEARLS, 1995](#))

Essa gramática geraria formas sentenciais que são cortadas justo antes da sequência de reconhecimento. Embora a gramática esteja clara, ainda existe a possibilidade de que gere uma sequência **gatc** sem ser cortada ou reconhecida pela enzima. Isso pode ser corrigido com uma gramática mais elaborada (Fig. 25).

Figura 25 – Gramática otimizada para produção de cortes em uma fita de DNA, onde  $\Sigma = a, c, g, t$ , o símbolo  $\delta$  representa o símbolo de corte e o símbolo  $\epsilon$  representa a palavra vazia

$$\begin{aligned} S &\rightarrow gG \mid aS \mid tS \mid cS \mid \delta gatc S \mid \epsilon \\ G &\rightarrow gS \mid aA \mid tS \mid cS \\ A &\rightarrow gS \mid aS \mid tT \mid cS \\ T &\rightarrow gS \mid aS \mid tS \end{aligned}$$

Fonte: ([SEARLS, 1995](#))

O acima descrito corresponde ao contexto biológico em que a enzima atua sobre o DNA até cortar cada sequência de reconhecimento. Nesse caso, a função de corte modela com precisão aquela ação da enzima em formas sentenciais geradas pelas derivações ordinárias da gramática.

### 2.1.7 Autômatos para busca de segmentos ALU em sequências genômicas

As sequências ALU, na sua estrutura, têm características específicas:

1. Região inicial (marcador molecular ‘ggcc’)
2. Caixas promotoras A e B (agentes biológicos que assistem na produção de proteínas)

3. Região do meio rica em Adenina

4. Inserção de 31 bases no braço direito (braço com maior comprimento)

5. Região terminal rica em Adenina

Cada uma dessas características estruturais podem ser representadas através de algum tipo de linguagem formal (Fig. 26). Segundo o contexto biológico e a natureza das sequências, as gramáticas formais utilizadas apresentam um nível de complexidade que permite a construção (e reconhecimento através dos diferentes autômatos) de basicamente qualquer tipo de estrutura encontrada na constituição genômica das espécies.

Figura 26 – Classes de Linguagens formais utilizadas para expressão de sequências genômicas

<i>Language</i>	<i>Grammar</i>	<i>Automaton</i>	<i>Recognition</i>	<i>Dependency</i>	<i>Operations</i>	<i>Biology</i>
Recursively Enumerable Languages	Unrestricted $Baa \rightarrow A$	Turing Machine 	Undecidable ?	Arbitrary ?	diagonalization	Unknown ?
Context-Sensitive Languages	Context-Sensitive $At \rightarrow aA$	Linear-Bounded 	Exponential? /	Crossing 	duplication inversion transposition	Pseudoknots (parallel) 
Context-Free Languages	Context-Free $S \rightarrow gSc$	Pushdown (stack) 	Polynomial 	Nested 	insertion	Hairpins (antiparallel) 
Regular Languages	Regular $A \rightarrow cA$	Finite-State Machine 	Linear 	Strictly Local 	concatenation disjunction iteration (*)	Transcription (processive) 

Fonte: ([SEARLS, 1995](#))

Para o reconhecimento básico de segmentos ALU contidos em uma sequência aleatória de DNA (ou RNA), são utilizados Autômatos finitos, para identificar conjuntos específicos de caracteres. Por exemplo, existem mais de 200 subfamílias de sequências ALU, divididas em 7 principais: AluJo, AluSx, AluSq, AluSp, AluY, AluSc e AluSb. Cada uma dessas famílias apresenta características diferenciáveis nas suas estruturas sintáticas (Fig. 27).

Segundo o tipo de sequência, autômatos podem ser programados para procurar cadeias de símbolos específicos que determinam a estrutura sintática diferenciável para cada família ALU. Segue um exemplo (Fig. 28) o diagrama de transição de estados para o autômato reconhecedor de um tipo específico de família ALU.

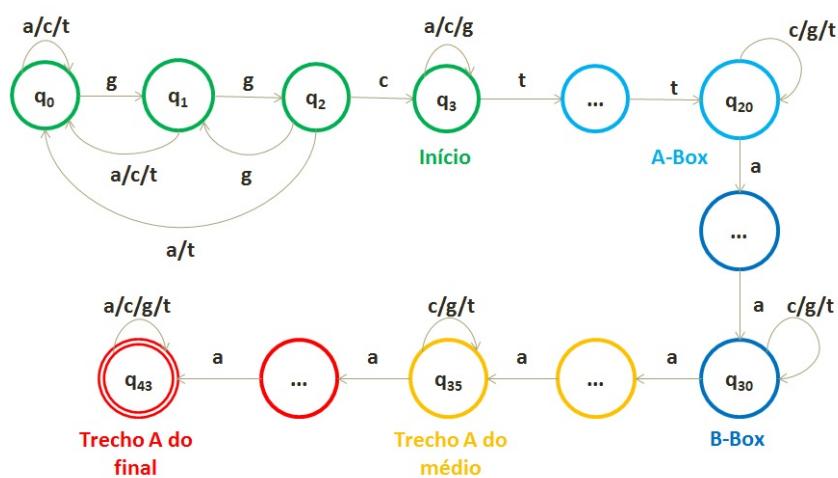
Figura 27 – Caixas internas para família consenso AluSx

- >gi|551543|gb|U14574.1|HSU14574 \*\*\*ALU WARNING:  
Human Alu-Sx subfamily consensus sequence

**ggcgggcggaggccggcgcg**tggctcacgcctgtaatcccagcacttgggaggaagatcacctgagg  
 tcaggagttcgagaccggcctggccaacatggtaatccagctactcgggaggctgaggcaggagaatcgcttgaacccggg  
 ggcgtggtggcgcgccgtaatccagctactcgggaggctgaggcaggagaatcgcttgaacccggg  
 aggccggaggttgcagtgagccgagatcgcgccactgcactccagcctggcgacagagcggagactccgtc  
 tcaaaaaaaaaa

Fonte: ([INFORMATION](#), 2015a)

Figura 28 – Diagrama de transição de estados para autômato reconhecedor da família AluSx



Fonte: Autoria própria

### 3 Pipeline para Expressão, reconhecimento e diferenciação de Sequências ALU

A ideia fundamental do projeto é desenhar uma sequência de processamentos, conhecida também como *Pipeline*, através do qual seja possível, tendo entradas iniciais (sequências genômicas), obter um resultado desejado (distância genética expressa em estruturas de linguagens formais), através dos dados utilizados.

#### 3.1 Construção do *Pipeline* para Expressão e Reconhecimento de Sequências ALU através de Linguagens Formais

O presente projeto foi desenvolvido em 5 etapas básicas, com o auxílio das ferramentas anteriormente mencionadas. A figura 29 mostra um esquema geral do desenvolvimento do trabalho de pesquisa:

Figura 29 – Construção do *Pipeline* do projeto



Fonte: Autoria própria

**1. Consulta de bancos de dados de sequências genômicas:** existem múltiplos bancos de dados para obter as sequências ALU das espécies selecionadas, sendo um dos mais importantes (pela disponibilidade e variedade de plataformas de busca) aquele proporcionado pelo Centro Nacional de Informação Biotecnológica (NCBI, pela sua sigla em inglês). A pesquisa foi realizada com arquivos em um formato tipo FASTA, o qual é basicamente um arquivo de texto contendo sequências tanto de nucleotídeos como de aminoácidos. Vale lembrar que além do banco de dados proporcionado pelo NCBI, existem outros que também podem ser utilizados pelo seu livre acesso, como por exemplo o banco de dados genéticos de Japão (DDBJ ou aquele disponível no site do Laboratório Europeu de Biología Molecular (EMBL). O objetivo dessa consulta inicial foi obter as sequências consensos das famílias ALU a serem consideradas.

**2. Definição da linguagem formal:** para construir linguagens formais que possam ser utilizadas para fins do projeto, é necessário definir gramáticas formais com os

seus respectivos dispositivos reconhecedores. As linguagens geradas têm como base os símbolos representando as bases nitrogenadas ('a' para Adenina, 'c' para Citosina, 'g' para Guanina e 'u' para Uracilo), a partir das quais estão conformados os diferentes tipos de sequências de RNA dentro do genoma. Tais símbolos com as respectivas regras de produção são definidas dentro da gramática. Devido à natureza do comportamento das estruturas das sequências, e a forma em que elas foram desenvolvidas e estudadas no projeto, são utilizadas linguagens do tipo 2 (Livres de Contexto), as quais inherentemente definiram as gramáticas e autômatos.

**3. Escrita de programas vários baseados nas linguagens formais:** na linguagem de programação seleccionada (Linguagem RUBY) são implementados os diferentes dispositivos (tanto gramáticas livres de contexto como autômatos de pilha pela classe de linguagem utilizada) que fazem a tarefa da análise das formas sentenciais que representam as sequências ALU, e elas são aceitadas ou rejeitadas, evidenciando as diferenças das sequências, expressas em estruturas sintáticas das linguagens. Temos um exemplo da programação de um autômato finito na linguagem RUBY (SOUZA, 2014), onde é indicado, além da sequência de entrada para reconhecimento, cada estado do autômato, o símbolo a ser reconhecido e os estados a serem adoptados em forma de transições:

Figura 30 – Exemplo de escritura de código RUBY para autômato finito determinístico

```
$: << "C:/ruby teoria/lab/03/afd"
require 'ReconhecedorDeterministico'

rdf = ReconhecedorDeterministico.new("q0", ["q3"])

rdf.automo.adicionarTransicao( [{"q0", "t"} => "q1"] )
rdf.automo.adicionarTransicao( [{"q1", "c"} => "q0"] )
rdf.automo.adicionarTransicao( [{"q0", "a"} => "q2"] )
rdf.automo.adicionarTransicao( [{"q2", "a"} => "q3"] )
rdf.automo.adicionarTransicao( [{"q3", "a"} => "q3"] )
rdf.automo.adicionarTransicao( [{"q3", "c"} => "q4"] )
rdf.automo.adicionarTransicao( [{"q3", "t"} => "q4"] )
rdf.automo.adicionarTransicao( [{"q3", "g"} => "q4"] )
rdf.automo.adicionarTransicao( [{"q4", "c"} => "q0"] )
rdf.automo.adicionarTransicao( [{"q4", "a"} => "q3"] )

rdf.iniciar( "tctcaaaaaa" )

automatos = rdf.analizar()
puts rdf.reconheceu?

#ou
automatos.each do |automato|
  puts automato.configuracao?
end
```

Fonte: Autoria própria

**4. Execução dos programas, mineração e verificação de dados:** uma vez prontos os dispositivos reconhecedores das linguagens formais construídas, a seguinte etapa

é a execução dos respectivos programas. Temos algoritmos para gerar inicialmente as sequências em diferentes arquivos (Linguagem RUBY) e um programa web para construir as sequências inteiras a partir das partes iniciais (chamado de *Strings Generator*). Uma vez com as sequências geradas, segue o processo de mineração de dados, o qual visa construir uma amostra de sequências com potencial para serem consideradas dentro de alguma família ALU. Tais sequências são ingressadas nas simulações dos autômatos (fig. 31) para serem aceitadas ou rejeitadas para cada família ALU. Finalmente, o objetivo foi validar o resultado do processamento dos autômatos no banco de dados do NCBI, através do algoritmo BLAST ([INFORMATION, 2015a](#)).

Figura 31 – Exemplo de execução de código RUBY para autômato finito determinístico reconhecendo cada símbolo da sequência **tctcaaaaaa**, sendo aceita com valor *true*

```
C:\ruby teoria\lab\03\afd\casouso>ruby automato1.rb
true
(q0, (1, <{t}ctcaaaaaa>)
 (q1, (2, <t{c}tcaaaaaa>)
 (q0, (3, <tc{t}caaaaaa>)
 (q1, (4, <tct{c}aaaaaa>)
 (q0, (5, <tctc{a}aaaaa>)
 (q2, (6, <tctca{a}aaaa>)
 (q3, (7, <tctcaa{a}aaa>)
 (q3, (8, <tctcaaa{a}aa>)
 (q3, (9, <tctcaaaa{a}a>)
 (q3, (10, <tctcaaaaa{a}>)
 (q3, (11, <tctcaaaaaa{>}))
```

Fonte: Autoria própria

**5. Definição de métrica para distância genômica:** uma vez obtidos os resultados da execução dos respectivos programas torna-se necessário definir uma métrica para encontrar a distância genômica, expressa nas diferentes estruturas sintáticas das linguagens formais. Devidamente processada e parametrizada, essa informação pode ser utilizada para a construção posterior de árvores filogenéticas baseadas nas diferenças das sequências ALU presentes em várias espécies.

### 3.2 Sequências ALU em bancos de dados

Para iniciar o processamento, é necessário obter as sequências alvo depositadas em um banco de dados. Na área das ciências biológicas, existem múltiplos bancos de dados para fazer pesquisas sobre sequências genômicas, sendo a mais importante aquela proporcionada pelo Centro Nacional de Informação Biotecnológica (NCBI, pelas suas siglas em inglês), com parceria constante com o Banco de Dados do DNA Japonês (DDBJ) e o Laboratório Europeu de Biologia Molecular (EMBL). O objetivo dessa consulta inicial foi obter as sequências consensos das famílias ALU a serem consideradas. Definidas com antecedência, foram procuradas as famílias de sequências ALU apropriadas para o

desenvolvimento do projeto. Vale lembrar que a família AluY ([HÄSLER; STRUB, 2006](#)) foi considerada amostra inicial para o processamento posterior no *Pipeline*. A partir dela, foi inferida a estrutura secundária do resto das famílias estudadas e analisadas, as quais apresentam uma construção sintática similar à original.

Figura 32 – Sítio de pesquisa de sequências através do site do NCBI

The screenshot shows the NCBI Nucleotide search interface. The search term 'ALU' is entered in the search bar. The results page displays a list of 4,943 nucleotide sequences. The first few results are listed as follows:

- \*\*\*ALU WARNING: Human Alu-Sb1 subfamily consensus sequence**
- 1. 288 bp linear DNA  
Accession: U14569.1 Gl: 551538  
[GenBank](#) [FASTA](#) [Graphics](#)
- 2. 290 bp linear DNA  
Accession: U14574.1 Gl: 551543  
[GenBank](#) [FASTA](#) [Graphics](#)
- 3. 291 bp linear DNA  
Accession: U14573.1 Gl: 551542  
[GenBank](#) [FASTA](#) [Graphics](#)

Fonte: ([INFORMATION, 2015b](#))

Em pesquisa realizada no NCBI (fig. 32), foram encontradas sequências consensos (resultado de vários alinhamentos) das seguintes famílias ALU:

1. AluJ
2. AluSb
3. AluSc
4. AluSp
5. AluSq
6. AluSx

Cada um dos consensos das famílias ALU, excepto a AluSx, são considerados para construção de autômatos e inferência de estruturas secundárias. Também é considerada a família AluY, que embora não foi encontrada como consenso geral de família no banco de dados GenBank, ela tem sido encontrada no gene da  $\alpha$ -Fetoproteína ([HÄSLER; STRUB, 2006](#)). A família AluSx apresenta, segundo o alinhamento múltiplo feito no site do Clustal Omega (Fig. 33) uma grande divergência na sua constituição interna e, portanto, na sua estrutura secundária.

Figura 33 – Alinhamento múltiplo dos consensos das famílias ALU

Alu-Sx	GGCGGGCGGAGGCCGGCGGGUGGUCCACGCCUGUAUCCCAGCACUUUGGGAGG-----
Alu-J	-----GGCGGGCGCGGGUGGUCCACGCCUGUAUCCCAGCACUUUGGGAGGCCGA
Alu-Sp	-----GGCGGGCGCGGGUGGUCCACGCCUGUAUCCCAGCACUUUGGGAGGCCGA
Alu-Sq	-----GGCGGGCGCGGGUGGUCCACGCCUGUAUCCCAGCACUUUGGGAGGCCGA
Alu-Y	-----GGCGGGCGCGGGUGGUCCACGCCUGUAUCCCAGCACUUUGGGAGGCCGA
Alu-Sb	-----GGCGGGCGCGGGUGGUCCACGCCUGUAUCCCAGCACUUUGGGAGGCCGA
Alu-Sc	-----GGCGGGCGCGGGUGGUCCACGCCUGUAUCCCAGCACUUUGGGAGGCCGA *****
Alu-Sx	-----AAGAUCCCCUGAGGUCCAGGAGUUCAGGACCGCCUGGCCAACAUUGGUGAAACCC
Alu-J	GGCGGGAGGAUCACUUGAGCCAGGGAGUUCAGGACCGCCUGGCCAACAUUGGUGAAACCC
Alu-Sp	GGCGGGCGGAUCACCUUGAGGUCCAGGGAGUUCAGGACCGCCUGGCCAACAUUGGUGAAACCC
Alu-Sq	GGCGGGUGGAUCACCUUGAGGUCCAGGGAGUUCAGGACCGCCUGGCCAACAUUGGUGAAACCC
Alu-Y	GGCGGGCGGAUCAC - GAGGUCCAGGAGAUCAGGACCAUCCUGGUUAACAUUGGUGAAACCC
Alu-Sb	GGCGGGCGGAUCAC - GAGGUCCAGGAGAUCAGGACCAUCCUGGUUAACACGGUGAAACCC
Alu-Sc	GGCGGGCGGAUCAC - GAGGUCAAGAGAUCAGGACCAUCCUGGCCAACAUUGGUGAAACCC *****
Alu-Sx	CGUCUCUACAUAAAAAU -- ACAAAAUUAGCCGGCGUGGGCGCGCCUGUAUCCC
Alu-J	CGUCUCUACAUAAAAAU -- ACAAAAUUAGCCGGCGUGGGCGCGCCUGUAUCCC
Alu-Sp	CGUCUCUACAUAAAAAU -- ACAAAAUUAGCCGGCGUGGGCGUGGGCGCCUGUAUCCC
Alu-Sq	CGUCUCUACAUAAAAAU -- ACAAAAUUAGCCGGCGUGGGCGUGGGCGCCUGUAUCCC
Alu-Y	CGUCUCUACAUAAAAAU -- ACAAAAUUAGCCGGCGUGGGCGUGGGCGCCUGUAUCCC
Alu-Sb	CGUCUCUACAUAAAAAU -- ACAAAAUUAGCCGGCGUGGGCGUGGGCGCCUGUAUCCC
Alu-Sc	CGUCUCUACAUAAAAAU -- ACAAAAUUAGCUGGGCGUGGGCGUGGGCGCCUGUAUCCC *****

Fonte: ([LABORATORY, 2015a](#))

### 3.3 Linguagens livres de contexto para expressão e reconhecimento de sequências ALU

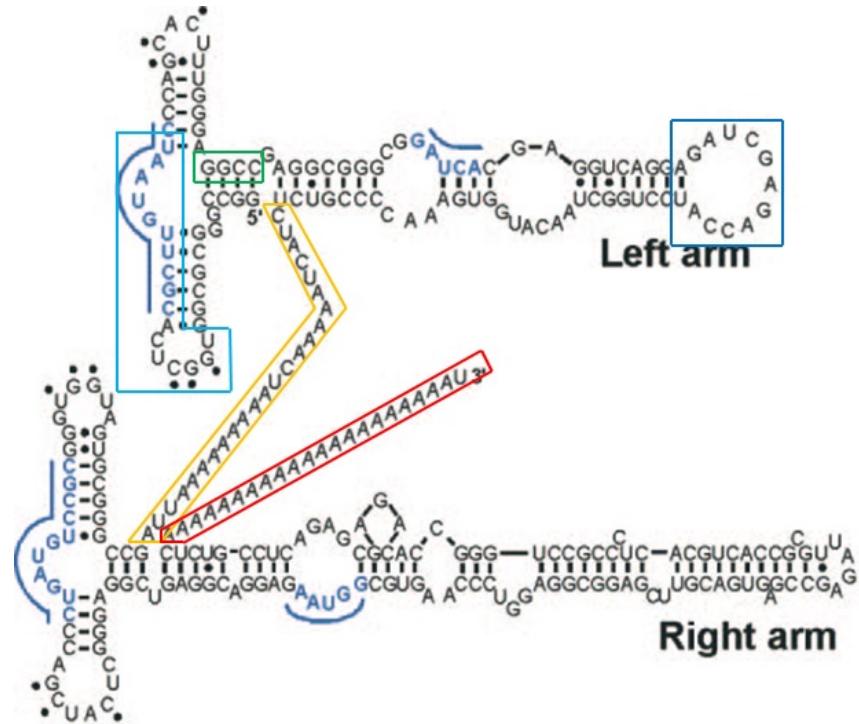
Considerando cada uma das famílias encontradas no Banco de Dados, é preciso a definição de uma (ou várias) Linguagens formais para expressão, geração e reconhecimento de sequências. A hierarquia de Chomsky (Fig. 26) ([SEARLS, 1995](#)) prevê a utilização de gramáticas livres de contexto como dispositivos geradores e autômatos de pilha como dispositivos reconhecedores quando trata-se de estruturas secundárias em sequências genômicas. Cada uma das famílias consenso apresentam caixas típicas de sequências ALU (Fig. 34), as quais podem ser consideradas como estruturas sintáticas desde o ponto de vista gramatical.

O dogma central da Biología ([KLUG et al., 2006](#)) considera os pareamentos entre as bases nitrogenadas **a-u** (RNA) ou **a-t** (DNA) e **c-g**, tanto em DNA (formação de fitas duplas) como em RNA (formação de estruturas secundárias).

Mesmo assim como na família AluY, foi utilizada a informação disponibilizada para inferir as estruturas secundárias do resto das famílias consenso. Todas, exceto a família AluSx, foram encontradas com estruturas secundárias similares à sequência AluY padrão (Fig. 35).

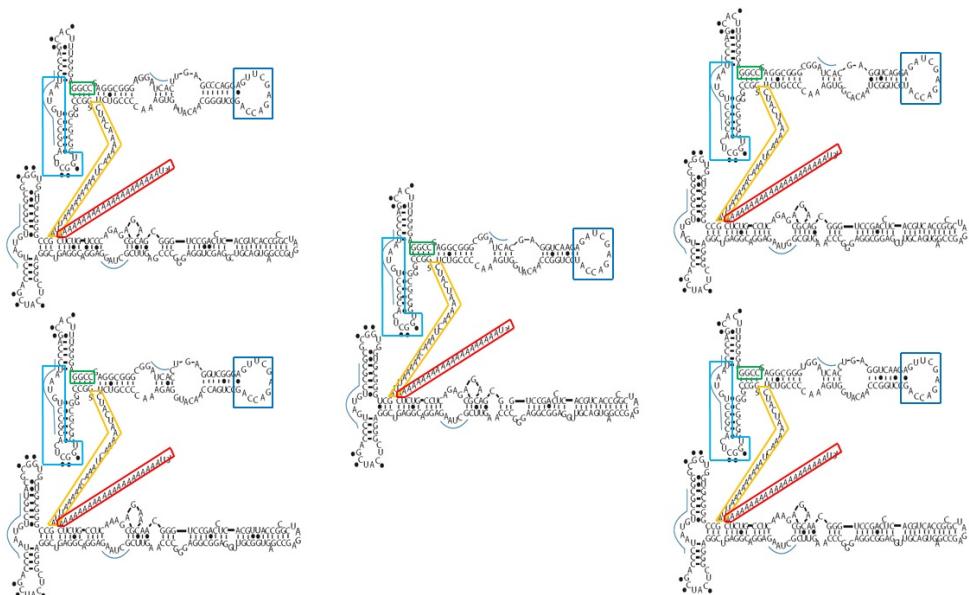
Segundo as estruturas sintáticas mostradas (Fig. 34 e Fig. 35), é possível inferir uma gramática livre de contexto apropriada para geração de cadeias com forma

Figura 34 – Estrutura Secundária da sequência RNA AluY, com suas caixas internas correspondentes. De cor verde, é marcada a região inicial. A cor azul clara indica a caixa promotora A-Box. A cor azul escura indica a caixa promotora B-Box. A cor amarela indica a região Poly-A média. A cor vermelha indica a região Poly-A final



Fonte: Autoria própria

Figura 35 – Estruturas Secundárias inferidas para os consensos das famílias ALU encontradas



Fonte: Autoria própria

estrutural simulando sequências genômicas de natureza Alu.

Uma gramática livre de contexto adaptada para geração de sequências genômicas, pode ser definida como segue:

$$G = (V, \Sigma, P, S)$$

$$V = \{S, A_n, B_n, D_n, E_n, F_n, H_n, W, Y, Z, a, g, t, c\}$$

$$\Sigma = \{a, g, t, c\}$$

P = conjunto de regras de produção

S = S (Símbolo inicial)

Os símbolos capitalizados representam não-terminais, ou seja, aqueles que serão substituídos na aplicação de regras de produção da gramática. O símbolo **A** é um não-terminal utilizado para substituição em cada *subscript* onde exista uma produção livre de contexto (pareamento de bases). O símbolo **B** é um não-terminal para produções regulares (ausência de pareamento de bases). Os símbolos {D, E, F, H, W, Y, Z} são não-terminais utilizados para concatenação sequencial entre os *subscript* escritos para gerar as palavras finais. A partir da gramática, o autômato de pilha é definido pela sua estrutura de sete elementos como segue:

$$M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$$

$$Q = \{q_0, q_1, q_2, \dots, q_n\}$$

$$\sigma = \{a, g, t, c\}$$

$$\Gamma = \{Z_0, A_n, C_n, G_n, U_n, R_n, P_n\}$$

$\delta$  = conjunto de transições

$q_0$  = estado inicial

$Z_0$  = símbolo inicial da pilha

$$F = \{q_f, \dots, q_n\} \text{ (Conjunto de estados finais)}$$

O conjunto **Q** contém todos os estados possíveis que o autômato pode adotar ao analisar uma sequência específica. O conjunto **F** contém os estados finais do autômato, geralmente definidos como  $q_f$ , mas é considerada a possibilidade de adicionar mais algum estado em casos especiais  $q_n$ . A pilha contém o símbolo inicial **Z<sub>0</sub>**, símbolos que permitem identificação de produções regulares (**R<sub>n</sub>**, **P<sub>n</sub>**) e símbolos que permitem identificar produções livres de contexto (**A<sub>n</sub>**, **C<sub>n</sub>**, **G<sub>n</sub>**, **U<sub>n</sub>**).

### 3.4 Escrita de *scripts* para gramáticas, autômatos e gerador de sequências

O desenvolvimento do projeto incluiu o uso de três grupos de *scripts*:

1. Gramáticas formais livres de contexto: utilizando *scripts* de uma plataforma disponibilizada na linguagem RUBY pelo Eng. Ícaro Andrade Souza (SOUZA, 2014), foram redesenhados para representar uma gramática livre para expressão de estruturas secundárias da família AluY (Fig. 36). O funcionamento do *script* foi desenvolvido para gerar não apenas sequências AluY, mas todas as diferentes possibilidades de palavras considerando 4 diferentes possibilidades para produções regulares (a, c, g, u) e 4 possibilidades para produções livres de contexto dos pareamentos biológicos possíveis (a-u, u-a, c-g, g-c). Isso para comprovar a hipótese de que uma gramática livre de contexto modelada para uma família AluY pode gerar sequências de outras famílias ALU. Por exemplo, tal gramática pode gerar as possibilidades:

$S \rightarrow aRu / uRa / cRg / gRc$

$R \rightarrow aRu / uSa / cSg / gSc / aR / uR / cR / gR$

Figura 36 – Exemplo da escrita de código para Gramática Livre de Contexto utilizada para gerar sequências ALU

```

1 $: << "C:/ruby_teoría/lab/05/gsc"
2 require 'GramaticaSensivelContexto'
3
4 gramSensivContex = GramaticaSensivelContexto.new()
5
6 gramSensivContex.adicionarProducao( {"S" => ["A1Z"] } )
7 gramSensivContex.adicionarProducao( {"A1" => ["gA2c", "cA2g", "aA2u", "uA2a"] } )
8 gramSensivContex.adicionarProducao( {"A2" => ["gA3c", "cA3g", "aA3u", "uA3a"] } )
9 gramSensivContex.adicionarProducao( {"A3" => ["gA4c", "cA4g", "aA4u", "uA4a"] } )
10 gramSensivContex.adicionarProducao( {"A4" => ["gB1D1D2c", "cB1D1D2g", "aB1D1D2u", "uB1D1D2a"] } )
11 gramSensivContex.adicionarProducao( {"B1" => ["aB2", "cB2", "gB2", "uB2"] } )
12 gramSensivContex.adicionarProducao( {"B2" => ["aB3", "cB3", "gB3", "uB3"] } )
13 gramSensivContex.adicionarProducao( {"B3" => ["aB4", "cB4", "gB4", "uB4"] } )
14 gramSensivContex.adicionarProducao( {"B4" => ["aD3", "cD3", "gD3", "uD3"] } )
15
16
17 derivacoes = gramSensivContex.derivar( "S", 10 )
18
19 derivacoes.each do |sentencias|
20   sentencias.each do |w, substituicoes|
21     puts "#{w.inspect()}=>#{substituicoes.inspect()}"
22   end
23 end

```

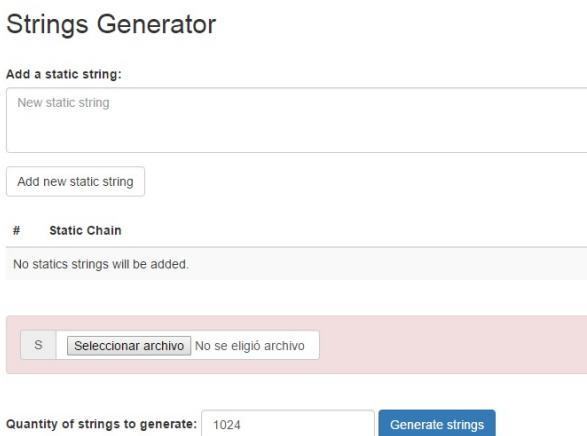
Fonte: Autoria própria

Considerando a capacidade de processamento no computador padrão (Acer Aspire E 15, Inter Core i5, 1.7GHz) utilizado para executar os *scripts*, foi concluído que para atingir um tempo prático e apropriado para gerar as formas sentenciais (330-340 segundos), a melhor estratégia era dividir as gramáticas por partes. No total foram escritos 35 *scripts*, cada uma com a quantidade entre 8 e 10 derivações da geração, para

produzir formas sentenciais com símbolos não-terminais para serem substituídos de forma sequencial no gerador de sequências.

2. Gerador de sequências: cada um dos 35 arquivos gerados através da gramática livre de contexto contêm formas sentenciais que são interpretadas como sub-sequências genômicas. Nesse passo é necessário um *script* intermediário para construir palavras finais de forma aleatória, a partir da substituição sequencial dos símbolos não-terminais em cada arquivo. Tal objetivo foi conseguido através de uma plataforma Web de livre acesso, de autoria própria em parceria com o Eng. Luis Alberto Nuñez, escrita na linguagem JavaScript ([\(NUÑEZ; MACHADO, 2016\)](#)) e nomeada como *Strings Generator* (Fig. 37).

Figura 37 – Plataforma Web AURELIA para programa *Strings Generator*



Fonte: ([\(NUÑEZ; MACHADO, 2016\)](#))

Tal plataforma pede ao usuário cada um dos 35 arquivos iniciais, os processa e constrói de forma aleatória um número de sequências finais indicado pelo usuário. Ainda, apresenta a possibilidade de adicionar cadeias estáticas para considerações do estudo em andamento.

3. Autômatos de pilha: utilizando outro grupo scripts da mesma plataforma disponibilizada na linguagem RUBY, o programa referente ao autômato de pilha foi adaptado para reconhecer especificamente sequências genômicas, já seja aceitando para uma família ALU ou rejeitando (Fig. 38). Assim, foi desenhado um autômato para cada família considerada, sendo um total de 6 autômatos.

Cada uma das sequências geradas no *Strings Generator* foi processada em cada um dos autômatos, para verificar possível associação com alguma das famílias ALU, e as saídas dos processamentos foram impressas em arquivos texto (.txt).

Figura 38 – Exemplo de *script* para Autômatos de Pilha para reconhecimento de sequências pertencentes a alguma família ALU

```

1 $: << `C:/ruby teoria/lab/04/apnd`
2 require 'ReconhecedorAPND.rb'
3
4 #Automato de pilha
5 rpnd = ReconhecedorAPND.new( "q0", ["qf"] )
6
7 /Região Inicial/
8 rpnd.automato.adicionarTransicao([["q0", "g", "Z0"] => [[["q0", ["Z0", "C1"]]]])
9 rpnd.automato.adicionarTransicao([["q0", "a", "Z0"] => [[["q0", ["Z0"]]]]])
10 rpnd.automato.adicionarTransicao([["q0", "c", "Z0"] => [[["q0", ["Z0"]]]]])
11 rpnd.automato.adicionarTransicao([["q0", "u", "Z0"] => [[["q0", ["Z0"]]]]])
12 rpnd.automato.adicionarTransicao([["q0", "\n", "Z0"] => [[["q0", ["Z0"]]]]])
13
14 rpnd.automato.adicionarTransicao([["q0", "g", "C1"] => [[["q0", ["C2"]]]]])
15 rpnd.automato.adicionarTransicao([["q0", "u", "C1"] => [[["q0", []]]]])
16 rpnd.automato.adicionarTransicao([["q0", "c", "C1"] => [[["q0", []]]]])
17 rpnd.automato.adicionarTransicao([["q0", "u", "C1"] => [[["q0", []]]]])
18 rpnd.automato.adicionarTransicao([["q0", "\n", "C1"] => [[["q0", ["C1"]]]]])
19
20 rpnd.automato.adicionarTransicao([["q0", "c", "C2"] => [[["q0", ["G1"]]]]])
21 rpnd.automato.adicionarTransicao([["q0", "a", "C2"] => [[["q0", []]]]])
22 rpnd.automato.adicionarTransicao([["q0", "g", "C2"] => [[["q0", ["C2"]]]]])
23 rpnd.automato.adicionarTransicao([["q0", "u", "C2"] => [[["q0", []]]]])
24 rpnd.automato.adicionarTransicao([["q0", "\n", "C2"] => [[["q0", ["C2"]]]]])
25
26 rpnd.automato.adicionarTransicao([["q0", "c", "G1"] => [[["q1", ["C", "C", "G", "G"]]]]])
27 rpnd.automato.adicionarTransicao([["q0", "a", "G1"] => [[["q0", []]]]])

```

Fonte: Autoria própria

### 3.5 Execução de programas: amostragem e validação de dados

A parte fundamental do *Pipeline* é a sua execução através dos *scripts* baseados na gramática livre de contexto e dos autômatos de pilha (Fig 39).

Figura 39 – Passos para execução real do *Pipeline*



Fonte: Autoria própria

Uma primeira amostragem de dados foi feita de forma manual, através da identificação do marcador molecular iniciador 'ggcc' específico das sequências ALU, para considerar apenas aquelas palavras que apresentassem possibilidades de serem definidas dentro de alguma família, e assim fazer uma filtragem para acrescentar a eficiência no reconhecimento nos autômatos.

Após o reconhecimento de sequências nos autômatos respectivos para cada família, segue a validação dos resultados no Banco de Dados. Utilizando o algoritmo BLAST (*Basic Local Alignment Search Tool*), cada sequência reconhecida em alguma família ALU foi processada para confirmar os resultados.

### 3.6 Formulação de proposta de métrica para distância genômica

A técnica tradicional para medir distância genômica consiste na construção de árvores filogenéticas, utilizando como base características fenotípicas ou genotípicas das espécies a serem consideradas. O uso das gramáticas formais oferece a possibilidade de se-definir uma nova métrica para medir a distância genômica, através das estruturas sintáticas encontradas tanto na série de aplicação de produções da gramática como na sequência de movimentações do autômato. No presente caso, propomos uma métrica através dos autômatos, analisando os resultados da execução dos mesmos sobre as sequências consideradas.

## 4 Resultados obtidos: execução do *Pipeline* e proposta para métrica para distância genômica

No presente capítulo, serão expostos os resultados da execução do *Pipeline* planejado, e a respectiva análise inferida a partir deles, aceitando ou rejeitando as hipóteses do trabalho.

### 4.1 Geração de formas sentenciais

Analizando o tempo de processamento dos scripts das gramáticas livres de contexto, foi determinado que o número ótimo de passos para geração de formas sentenciais para cada algoritmo é entre 8 e 10 derivações (330-340 seg.). Um número maior apresenta um estouro de memória do computador utilizado pelo número de substituições feitas pelo programa, apresentado-se a possibilidade de um processamento mais eficiente, utilizando um computador com um processador com maior capacidade (recomenda-se utilizar o computador CACAU, disponibilizado no NBCGIB nas instalações da UESC). Considerando a taxa de aumento exponencial (mais quatro possibilidades em produções livres de contexto **a-u**, **u-a**, **c-g**, **g-c** ou em produções regulares **a**, **u**, **c**, **g** com cada derivação acrescentada), o número de possibilidades geradas em cada script pode ser calculado por:

$$X = 4^n \quad (1)$$

onde 'X' representa o número de substituições feitas e 'n' é o número de passos aplicados. Por exemplo, considere-se uma gramática  $G$  com as seguintes derivações:

$$S \rightarrow aS / cS / gS / uS$$

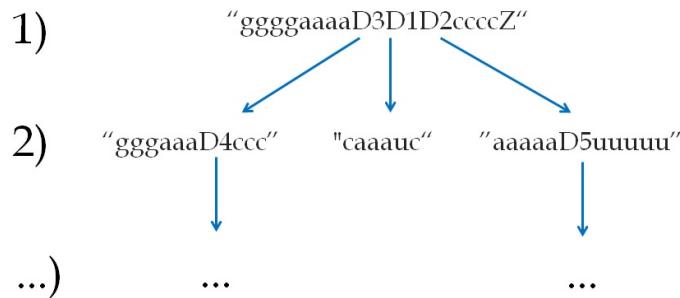
Na primeira derivação obtemos **aS** / **cS** / **gS** / **uS** (4 possibilidades). Se for acrescentada mais uma derivação, obteríamos **aaS** / **acS** / **agS** / **auS** / **caS** / **ccS** / **cgS** / **cuS** / **gaS** / **gcS** / **ggS** / **guS** / **uaS** / **ucS** / **ugS** / **uuS** (16 possibilidades) e assim sucessivamente.

No final, foram obtidos 35 scripts baseados na gramática, cada um contendo formas sentenciais de sequências ALU. Esses arquivos foram nomeados com símbolos não-terminais (definidos dentro de gramática), para posterior processamento no Gerador de sequências ou *Strings Generator*.

## 4.2 Geração de palavras

Uma vez obtidos os arquivos contendo as formas sentencias parciais, é necessário construir palavras finais representando, potencialmente, sequências ALU. Tal objetivo é atingido através da plataforma *Strings Generator*, disponibilizada na plataforma Aurelia para livre acesso. O funcionamento do programa, e assim como mostrado anteriormente na interface gráfica, é pedir para um usuário fornecer um por um os arquivos resultantes das produções da gramática, localizados em uma pasta local do computador. Logo, pede para o usuário fornecer um número de sequências desejadas, e de forma aleatória, seleciona esse número de formas sentencias de cada um dos 35 arquivos para randomizar a construção das palavras (Fig. 40). De forma sequencial, as formas sentenciais dentro de cada arquivo servem como entrada para arquivos posteriores, substituindo símbolos não-terminais específicos. O processo é repetido até ter consumido todos e cada um dos arquivos.

Figura 40 – Procedimento para geração de palavras através da substituição de símbolos não-terminais



Fonte: Autoria própria

O resultado é uma lista de 'n' palavras finais, sendo 'n' o número desejado de sequências fornecido pelo usuário. Dita lista é impressa em um arquivo de texto (saída.txt) para acesso e consulta (Fig. 41).

As amostras através da geração de palavras foram construídas em tamanhos de  $2^n$  (Tabela 1), até chegar em uma quantidade onde era apresentado um estouro da plataforma web, sendo mostrada uma mensagem de erro.

## 4.3 Amostragem de dados

A gramática livre de contexto definida para gerar as palavras, inferida a partir da família AluY, considera não apenas só sequências dessa família, mas também qualquer outra possibilidade de sequência que atenda a estrutura secundária correspondente à dita família. Colocando-o em termos matemáticas, e fazendo análise da quantidade

Figura 41 – Parcada da lista de palavras finais geradas pelo programa *Strings Generator*

Fonte: Autoria própria

de produções necessárias para gerar sequências inteiras para um consenso de uma sequência AluY, encontrada no gene -Fetoproteína na sua estrutura secundária, temos:

- 73 produções livres de contexto.
  - 159 produções regulares.

Com um total de 232 produções e considerando o dogma central da biología, temos um número total de possibilidades de:

$$4^{232} = 4,76x4^{139} \quad (2)$$

Sendo um universo de possibilidades de sequências grande demais para análises práticas, é feita uma amostragem manual de dados para serem processados nos autômatos. O primeiro filtro é identificar aquelas sequências que apresentem o prefixo **ggcc**, o marcador molecular inicial para encontrar palavras ALU no genoma (Tabela 1).

Após amostragem inicial, cada uma das 271 sequências resultantes são processadas nos Autômatos de Pilha escritos para cada uma das 6 famílias ALU. O objetivo é identificar se é encontrada pelo menos uma sequência relacionada às famílias, na busca de comprovar a hipótese de que uma gramática livre de contexto inferida de uma caracterização ALU específica pode gerar sequências de outras famílias.

## 4.4 Reconhecimento de sequências

Foram programados autômatos independentes para reconhecer cada uma das 6 famílias. Em cada autômato foram processadas as 271 sequências pertencentes à amostra com prefixo inicial **ggcc**, com um total de 1.626 reconhecimentos (Fig. 42). Cada processamento registrou um tempo de entre 190 e 220 seg.

Tabela 1 – Tamanho da geração inicial de palavras e amostragem de sequências com prefixo iniciador.

Sequências geradas no <i>Strings Generator</i>	Sequências ALU potencias (prefixo 'ggcc')	Sequências ALU encontradas
2	0	0
4	0	0
8	0	0
16	0	0
32	0	0
64	0	0
128	0	0
256	0	0
512	0	0
1024	2	0
2048	2	0
4096	11	1
8192	17	4
16384	64	10
32768	114	17
65536	271	57

Fonte: Autoria própria.

Figura 42 – Exemplo da saída com valor *True* da simulação de um autômato

```
true
((q0, (1, <{g}ccaugcgcguggcucacgccuguaaucccaguaccaugggaggccgaggcgggauuuucacuagcgc
ccaggaguucgagaccagccugggcaguuuccgugacgccccgucuggacaaaaaaagacaagacguaucag
ccgggcgugguggcgcgcgcugaguucccagcuacucugggaggcugaggcaggaggaucgcuugagccc
gggaggucgagggcugcagugagccugacgcgcacugcucuccagccugggcgacagagcgcagacccu
gucucaaaaaaaaaacaaccauaaaa>),["Z0"])
((q0, (2, <g{g}ccaugcgcguggcucacgccuguaaucccaguaccaugggaggccgaggcgggauuuucacuagcgc
ccaggaguucgagaccagccugggcaguuuccgugacgccccgucuggacaaaaaaagacaagacguaucag
ccgggcgugguggcgcgcgcugaguucccagcuacucugggaggcugaggcaggaggaucgcuugagccc
gggaggucgagggcugcagugagccugacgcgcacugcucuccagccugggcgacagagcgcagacccu
gucucaaaaaaaaaacaaccauaaaa>),["Z0", "C1"])
((q0, (3, <gg{c}caugcgcguggcucacgccuguaaucccaguaccaugggaggccgaggcgggauuuucacuagcgc
```

Fonte: Autoria própria

Um total de 57 sequências foram retornadas com valor *true* nas saídas das simulações dos autômatos (Tabelas 1 e 2), interpretando-se como que tais sequências pertencem a uma família ALU.

Cada uma das sequências reconhecidas precisam de ser verificadas em um Banco de Dados, no caso, aquele disponibilizado pelo NCBI.

Tabela 2 – Sequências reconhecidas por Autômatos associados com sequências ALU.

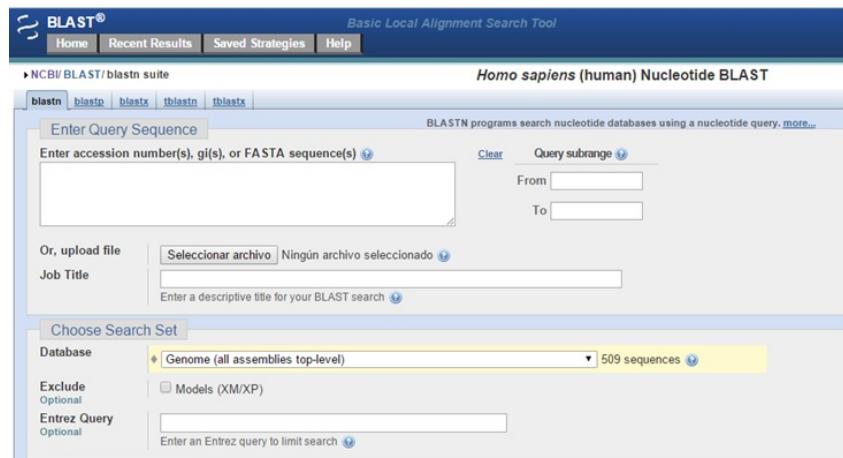
Família ALU	Num. de sequências reconhecidas	Porcentagem representativa (em relação à amostra com prefixo inicial)
AluJ	12	4.43%
AluSb	8	2.95%
AluSc	7	2.58%
AluSp	10	3.69%
AluSq	5	1.85%
AluY	15	5.54%
<b>TOTAL</b>	<b>57</b>	<b>21.03%</b>

Fonte: Autoria própria.

## 4.5 Validação de dados

Para validar os dados obtidos nos processamentos dos autômatos, cada uma das sequências reconhecidas é fornecida como entrada para o algoritmo BLAST, o qual retorna como resultado os trechos genômicos com maior similaridade registrado no Banco de Dados. O acesso é pelo site do NCBI (Fig. 43).

Figura 43 – Site de pesquisa de sequências através do BLAST



Fonte: ([INFORMATION, 2015a](#))

57 sequências foram consultadas no GenBank, com resultados como segue:

Tabela 3 – Validação de sequências reconhecidas no Banco de Dados GenBank do NCBI, através do algoritmo BLAST.

No. de sequências da amostra <i>Strings Generator</i>	Família reconhecida pelo autômato	Família reconhecida pelo BLAST	Porcentagem de identidade segundo BLAST
817	AluJ	AluJ	91%
974	AluSb	AluSb	86%
1608	AluSp	AluSp	83%
3596	AluY	AluY	86%
5161	AluY	Alu (Macaca Mulatta)	98%
6126	AluJ	AluJ	85%
6224	AluY	Alu (Pan Troglodytes)	82%
6289	AluSc	AluSc	86%
7047	AluY	AluY	84%
8225	AluSq	AluSq	87%
9170	AluSb	AluSb	95%
10603	AluSq	AluSq	89%
10948	AluSp	AluSp	91%
11879	AluSc	AluSc	86%
13755	AluJ	AluJ	95%
14648	AluJ	AluJ	86%
15693	AluSp	AluSp	98%
16623	AluSb	Crom. 3	85%
17821	AluSc	AluSc	82%
18307	AluJ	AluJ	86%
18728	AluY	AluY	84%
20157	AluSp	AluSp	87%
22231	AluY	AluY	95%
22842	AluSb	Crom. 5	89%
23378	AluJ	AluJ	91%
24837	AluSq	AluSq	89%
25867	AluY	AluY	91%
27389	AluJ	AluJ	94%
Continua			

Continuação da tabela			
No. de sequências da amostra <i>Strings Generator</i>	Família reconhecida pelo autômato	Família reconhecida pelo BLAST	Porcentagem de identidade segundo BLAST
27913	AluSc	AluSc	86%
28217	AluSp	AluSp	91%
30872	AluY	AluY	92%
32619	AluSp	AluSp	91%
35156	AluSb	AluSb	90%
38001	AluY	Aly	89%
40076	AluJ	AluJ	90%
43330	AluSq	AluSq	89%
44940	AluSc	AluSc	87%
47031	AluY	AluY	94%
48643	AluSp	AluSp	90%
50186	AluSb	Alu (Pan Troglodytes)	89%
51268	AluY	AluY	92%
51777	AluJ	AluJ	85%
53746	AluSc	AluSc	86%
54626	AluSp	AluSp	88%
55442	AluJ	AluJ	88%
56290	AluY	AluY	91%
58031	AluSq	AluSq	86%
59014	AluSb	AluSb	89%
59641	AluJ	AluJ	85%
60328	AluSp	AluSp	86%
61026	AluY	AluY	93%
62637	AluSc	AluSc	85%
62948	AluJ	AluJ	87%
64104	AluY	AluY	92%
64164	AluSb	AluSb	85%
65169	AluSp	AluSp	84%
65437	AluY	AluY	92%

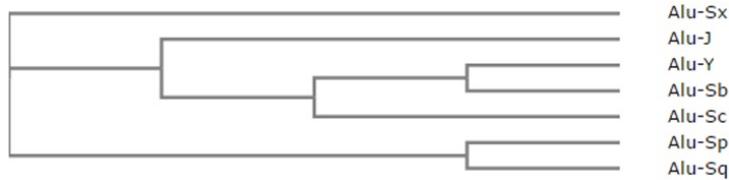
Fonte: Autoria própria.

Um total de 57 sequências (100% foram verificadas pelo Banco de Dados como sequências de natureza ALU. No entanto, as sequências que foram reconhecidas para outras espécies além dos consensos para *Homo Sapiens* não apresentam informação disponibilizada para sobre a família às quais pertencem, pelo qual as mesmas não são indicadas na tabela de acima.

## 4.6 Formalização de proposta de métrica para distância genômica

Existem vários fatores bases para construir as métricas para distância genômica, geralmente codificadas em árvores filogenéticas; um dos grupos de árvores mais usados nos estudos de genética são aqueles desenhados através de Alinhamento Múltiplo de Sequências (MSAs pela sua sigla em inglês). Essa técnica atribui uma pontuação específica (conhecida como *Value*) para cada *Match*, *Mismatch* e *Gap* encontrados no resultado do Alinhamento Múltiplo ([THOMPSON et al., 1994](#)). O *Value* é uma métrica numérica que fornece informação para gerar uma árvore gráfica (Fig. 44).

Figura 44 – Árvore filogenética dos Consensos das 6 famílias ALU construída no Clustal Omega



Fonte: ([LABORATORY, 2015a](#))

A nova proposta para estabelecer uma métrica para distância genômica através da análise da transição de estados dos autômatos de pilha apresentada neste trabalho procura um padrão que possa ser interpretado como confiável para estabelecer dita métrica.

Nas simulações dos autômatos, a notação de transição de estados é definida por:

$$(q_i, \sigma, X) \rightarrow (q_j, \gamma) \quad (3)$$

onde, para cada transição,  $q_i$  é o estado de origem,  $\sigma$  é o símbolo na fita de entrada,  $X$  é o símbolo no topo da pilha,  $q_j$  é o estado final e  $\gamma$  é o símbolo que substitui aquele no topo da Pilha. As palavras associadas com as transições de estados para reconhecimento de uma palavra podem ser interpretadas como uma linguagem formal de diferentes tipos. Por exemplo, para uma sequência da família J reconhecida pelo seu

respectivo autômato, temos as seguintes transições de estados, com suas respectivas repetições:

$$(q_0)^4.(q_1)^7.(q_2)^5.q_3.q_2.q_3.q_2.q_3.q_4.(q_5)^{13}.q_6.(q_5)^4.q_6.q_7.q_8.q_9.q_{10}.q_{11}.q_{12}.q_{13}.(q_{14})^{13}.(q_{15})^5.(q_{16})^8.\\(q_{17})^6.(q_{18})^7.(q_{19})^{11}.q_{20}.q_{21}.q_{22}.q_{23}.q_{24}.q_{25}.q_{26}.(q_{27})^7.q_{28}.q_{29}.q_{30}.q_{31}.(q_{32})^3.q_{33}.q_{34}.q_{35}.q_{36}.q_{37}.\\q_{38}.q_{39}.(q_{40})^{28}.(q_{41})^5.(q_{42})^7.q_{43}.q_{44}.q_{45}.q_{42}.q_{43}.q_{44}.q_{45}.(q_{46})^9.(q_{47})^8.q_{48}.q_{49}.q_{50}.q_{51}.q_{52}.q_{53}.q_{54}.\\(q_{55})^4.(q_{56})^2.(q_{57})^5.(q_{58})^8.(q_{59})^6.(q_{60})^{12}.(q_{61})^9.(q_{62})^4.q_{63}.q_{64}.q_{65}.q_{66}.(q_{65})^3.q_{66}.q_{67}.q_{66}.q_{67}.q_{68}.\\q_{69}.q_{70}.q_{71}.q_{72}.q_{73}.q_{74}.q_{75}.q_{76}.q_{77}.q_{78}.q_{79}.q_{80}.q_{81}.q_{82}.q_{83}.q_{84}.q_{85}.q_{86}.q_{87}.q_{88}.q_{89}.(q_{90})^3.q_{91}.q_{92}.\\(q_{93})^4.q_{94}.q_{95}.q_{96}.q_{97}.(q_{98})^2.q_{99}.q_{100}.q_{101}.q_{102}.(q_{103})^5.(q_f)^{32}$$

Na lista de transições, são encontradas diversas passagens de estados que podem ser interpretadas como palavras associadas da linguagem original definida para expressão das sequências, onde ditas palavras constroem linguagens formais tanto livres de contexto (da forma  $a^n b^n$ ) como sensíveis ao contexto (da forma  $a^n b^n c^n$ ).

### 1) Estruturas livres de contexto, com n=2:

$$(q_0)^4.(q_1)^7.(q_2)^5.q_3.q_2.q_3.q_2.q_3.q_4.(q_5)^{13}.q_6.(q_5)^4.q_6.q_7.q_8.q_9.q_{10}.q_{11}.q_{12}.q_{13}.(q_{14})^{13}.(q_{15})^5.(q_{16})^8.\\(q_{17})^6.(q_{18})^7.(q_{19})^{11}.q_{20}.q_{21}.q_{22}.q_{23}.q_{24}.q_{25}.q_{26}.(q_{27})^7.q_{28}.q_{29}.q_{30}.q_{31}.(q_{32})^3.q_{33}.q_{34}.q_{35}.q_{36}.q_{37}.\\q_{38}.q_{39}.(q_{40})^{28}.(q_{41})^5.(q_{42})^7.q_{43}.q_{44}.q_{45}.q_{42}.q_{43}.q_{44}.q_{45}.(q_{46})^9.(q_{47})^8.q_{48}.q_{49}.q_{50}.q_{51}.q_{52}.q_{53}.q_{54}.\\(q_{55})^4.(q_{56})^2.(q_{57})^5.(q_{58})^8.(q_{59})^6.(q_{60})^{12}.(q_{61})^9.(q_{62})^4.q_{63}.q_{64}.q_{65}.q_{66}.(q_{65})^3.q_{66}.q_{67}.q_{66}.q_{67}.q_{68}.\\q_{69}.q_{70}.q_{71}.q_{72}.q_{73}.q_{74}.q_{75}.q_{76}.q_{77}.q_{78}.q_{79}.q_{80}.q_{81}.q_{82}.q_{83}.q_{84}.q_{85}.q_{86}.q_{87}.q_{88}.q_{89}.(q_{90})^3.q_{91}.q_{92}.\\(q_{93})^4.q_{94}.q_{95}.q_{96}.q_{97}.(q_{98})^2.q_{99}.q_{100}.q_{101}.q_{102}.(q_{103})^5.(q_f)^{32}$$

### 2) Estruturas livres de contexto, com n=6:

$$(q_0)^4.(q_1)^7.(q_2)^5.q_3.q_2.q_3.q_2.q_3.q_4.(q_5)^{13}.q_6.(q_5)^4.q_6.q_7.q_8.q_9.q_{10}.q_{11}.q_{12}.q_{13}.(q_{14})^{13}.(q_{15})^5.(q_{16})^8.\\(q_{17})^6.(q_{18})^7.(q_{19})^{11}.q_{20}.q_{21}.q_{22}.q_{23}.q_{24}.q_{25}.q_{26}.(q_{27})^7.q_{28}.q_{29}.q_{30}.q_{31}.(q_{32})^3.q_{33}.q_{34}.q_{35}.q_{36}.q_{37}.\\q_{38}.q_{39}.(q_{40})^{28}.(q_{41})^5.(q_{42})^7.q_{43}.q_{44}.q_{45}.q_{42}.q_{43}.q_{44}.q_{45}.(q_{46})^9.(q_{47})^8.q_{48}.q_{49}.q_{50}.q_{51}.q_{52}.q_{53}.q_{54}.\\(q_{55})^4.(q_{56})^2.(q_{57})^5.(q_{58})^8.(q_{59})^6.(q_{60})^{12}.(q_{61})^9.(q_{62})^4.q_{63}.q_{64}.q_{65}.q_{66}.(q_{65})^3.q_{66}.q_{67}.q_{66}.q_{67}.q_{68}.\\q_{69}.q_{70}.q_{71}.q_{72}.q_{73}.q_{74}.q_{75}.q_{76}.q_{77}.q_{78}.q_{79}.q_{80}.q_{81}.q_{82}.q_{83}.q_{84}.q_{85}.q_{86}.q_{87}.q_{88}.q_{89}.(q_{90})^3.q_{91}.q_{92}.\\(q_{93})^4.q_{94}.q_{95}.q_{96}.q_{97}.(q_{98})^2.q_{99}.q_{100}.q_{101}.q_{102}.(q_{103})^5.(q_f)^{32}$$

### 3) Estruturas sensíveis ao contexto, com n=3:

$$(q_0)^4.(q_1)^7.(q_2)^5.q_3.q_2.q_3.q_2.q_3.q_4.(q_5)^{13}.q_6.(q_5)^4.q_6.q_7.q_8.q_9.q_{10}.q_{11}.q_{12}.q_{13}.(q_{14})^3.(q_{15})^5.(q_{16})^8.\\(q_{17})^6.(q_{18})^7.(q_{19})^{11}.q_{20}.q_{21}.q_{22}.q_{23}.q_{24}.q_{25}.q_{26}.(q_{27})^7.q_{28}.q_{29}.q_{30}.q_{31}.(q_{32})^3.q_{33}.q_{34}.q_{35}.q_{36}.q_{37}.\\q_{38}.q_{39}.(q_{40})^{28}.(q_{41})^5.(q_{42})^7.q_{43}.q_{44}.q_{45}.q_{42}.q_{43}.q_{44}.q_{45}.(q_{46})^9.(q_{47})^8.q_{48}.q_{49}.q_{50}.q_{51}.q_{52}.q_{53}.q_{54}.\\(q_{55})^4.(q_{56})^2.(q_{57})^5.(q_{58})^8.(q_{59})^6.(q_{60})^{12}.(q_{61})^9.(q_{62})^4.q_{63}.q_{64}.q_{65}.q_{66}.(q_{65})^3.q_{66}.q_{67}.q_{66}.q_{67}.q_{68}.\\q_{69}.q_{70}.q_{71}.q_{72}.q_{73}.q_{74}.q_{75}.q_{76}.q_{77}.q_{78}.q_{79}.q_{80}.q_{81}.q_{82}.q_{83}.q_{84}.q_{85}.q_{86}.q_{87}.q_{88}.q_{89}.(q_{90})^3.q_{91}.q_{92}.\\(q_{93})^4.q_{94}.q_{95}.q_{96}.q_{97}.(q_{98})^2.q_{99}.q_{100}.q_{101}.q_{102}.(q_{103})^5.(q_f)^{32}$$

4) Estruturas sensíveis ao contexto, com n=4:

$$(q_0)^4 \cdot (q_1)^7 \cdot (q_2)^5 \cdot q_3 \cdot q_2 \cdot q_3 \cdot q_2 \cdot q_3 \cdot q_4 \cdot (q_5)^{13} \cdot q_6 \cdot (q_5)^4 \cdot q_6 \cdot q_7 \cdot q_8 \cdot q_9 \cdot q_{10} \cdot q_{11} \cdot q_{12} \cdot q_{13} \cdot (q_{14})^{13} \cdot (q_{15})^5 \cdot (q_{16})^8 \cdot (q_{17})^6 \cdot (q_{18})^7 \cdot (q_{19})^{11} \cdot q_{20} \cdot q_{21} \cdot q_{22} \cdot q_{23} \cdot q_{24} \cdot q_{25} \cdot q_{26} \cdot (q_{27})^7 \cdot q_{28} \cdot q_{29} \cdot q_{30} \cdot q_{31} \cdot (q_{32})^3 \cdot q_{33} \cdot q_{34} \cdot q_{35} \cdot q_{36} \cdot q_{37} \cdot q_{38} \cdot q_{39} \cdot (q_{40})^{28} \cdot (q_{41})^5 \cdot (q_{42})^7 \cdot q_{43} \cdot q_{44} \cdot q_{45} \cdot q_{42} \cdot q_{43} \cdot q_{44} \cdot q_{45} \cdot (q_{46})^9 \cdot (q_{47})^8 \cdot q_{48} \cdot q_{49} \cdot q_{50} \cdot q_{51} \cdot q_{52} \cdot q_{53} \cdot q_{54} \cdot (q_{55})^4 \cdot (q_{56})^2 \cdot (q_{57})^5 \cdot (q_{58})^8 \cdot (q_{59})^6 \cdot (q_{60})^{12} \cdot (q_{61})^9 \cdot (q_{62})^4 \cdot q_{63} \cdot q_{64} \cdot q_{65} \cdot q_{66} \cdot (q_{65})^3 \cdot q_{66} \cdot q_{67} \cdot q_{66} \cdot q_{67} \cdot q_{68} \cdot q_{69} \cdot q_{70} \cdot q_{71} \cdot q_{72} \cdot q_{73} \cdot q_{74} \cdot q_{75} \cdot q_{76} \cdot q_{77} \cdot q_{78} \cdot q_{79} \cdot q_{80} \cdot q_{81} \cdot q_{82} \cdot q_{83} \cdot q_{84} \cdot q_{85} \cdot q_{86} \cdot q_{87} \cdot q_{88} \cdot q_{89} \cdot (q_{90})^3 \cdot q_{91} \cdot q_{92} \cdot (q_{93})^4 \cdot q_{94} \cdot q_{95} \cdot q_{96} \cdot q_{97} \cdot (q_{98})^2 \cdot q_{99} \cdot q_{100} \cdot q_{101} \cdot q_{102} \cdot (q_{103})^5 \cdot (q_f)^{32}$$

5) Estruturas sensíveis ao contexto, com n=5:

$$(q_0)^4 \cdot (q_1)^7 \cdot (q_2)^5 \cdot q_3 \cdot q_2 \cdot q_3 \cdot q_2 \cdot q_3 \cdot q_4 \cdot (q_5)^{13} \cdot q_6 \cdot (q_5)^4 \cdot q_6 \cdot q_7 \cdot q_8 \cdot q_9 \cdot q_{10} \cdot q_{11} \cdot q_{12} \cdot q_{13} \cdot (q_{14})^{13} \cdot (q_{15})^5 \cdot (q_{16})^8 \cdot (q_{17})^6 \cdot (q_{18})^7 \cdot (q_{19})^{11} \cdot q_{20} \cdot q_{21} \cdot q_{22} \cdot q_{23} \cdot q_{24} \cdot q_{25} \cdot q_{26} \cdot (q_{27})^7 \cdot q_{28} \cdot q_{29} \cdot q_{30} \cdot q_{31} \cdot (q_{32})^3 \cdot q_{33} \cdot q_{34} \cdot q_{35} \cdot q_{36} \cdot q_{37} \cdot q_{38} \cdot q_{39} \cdot (q_{40})^{28} \cdot (q_{41})^5 \cdot (q_{42})^7 \cdot q_{43} \cdot q_{44} \cdot q_{45} \cdot q_{42} \cdot q_{43} \cdot q_{44} \cdot q_{45} \cdot (q_{46})^9 \cdot (q_{47})^8 \cdot q_{48} \cdot q_{49} \cdot q_{50} \cdot q_{51} \cdot q_{52} \cdot q_{53} \cdot q_{54} \cdot (q_{55})^4 \cdot (q_{56})^2 \cdot (q_{57})^5 \cdot (q_{58})^8 \cdot (q_{59})^6 \cdot (q_{60})^{12} \cdot (q_{61})^9 \cdot (q_{62})^4 \cdot q_{63} \cdot q_{64} \cdot q_{65} \cdot q_{66} \cdot (q_{65})^3 \cdot q_{66} \cdot q_{67} \cdot q_{66} \cdot q_{67} \cdot q_{68} \cdot q_{69} \cdot q_{70} \cdot q_{71} \cdot q_{72} \cdot q_{73} \cdot q_{74} \cdot q_{75} \cdot q_{76} \cdot q_{77} \cdot q_{78} \cdot q_{79} \cdot q_{80} \cdot q_{81} \cdot q_{82} \cdot q_{83} \cdot q_{84} \cdot q_{85} \cdot q_{86} \cdot q_{87} \cdot q_{88} \cdot q_{89} \cdot (q_{90})^3 \cdot q_{91} \cdot q_{92} \cdot (q_{93})^4 \cdot q_{94} \cdot q_{95} \cdot q_{96} \cdot q_{97} \cdot (q_{98})^2 \cdot q_{99} \cdot q_{100} \cdot q_{101} \cdot q_{102} \cdot (q_{103})^5 \cdot (q_f)^{32}$$

6) Estruturas sensíveis ao contexto, com n=7:

$$(q_0)^4 \cdot (q_1)^7 \cdot (q_2)^5 \cdot q_3 \cdot q_2 \cdot q_3 \cdot q_2 \cdot q_3 \cdot q_4 \cdot (q_5)^{13} \cdot q_6 \cdot (q_5)^4 \cdot q_6 \cdot q_7 \cdot q_8 \cdot q_9 \cdot q_{10} \cdot q_{11} \cdot q_{12} \cdot q_{13} \cdot (q_{14})^{13} \cdot (q_{15})^5 \cdot (q_{16})^8 \cdot (q_{17})^6 \cdot (q_{18})^7 \cdot (q_{19})^{11} \cdot q_{20} \cdot q_{21} \cdot q_{22} \cdot q_{23} \cdot q_{24} \cdot q_{25} \cdot q_{26} \cdot (q_{27})^7 \cdot q_{28} \cdot q_{29} \cdot q_{30} \cdot q_{31} \cdot (q_{32})^3 \cdot q_{33} \cdot q_{34} \cdot q_{35} \cdot q_{36} \cdot q_{37} \cdot q_{38} \cdot q_{39} \cdot (q_{40})^{28} \cdot (q_{41})^5 \cdot (q_{42})^7 \cdot q_{43} \cdot q_{44} \cdot q_{45} \cdot q_{42} \cdot q_{43} \cdot q_{44} \cdot q_{45} \cdot (q_{46})^9 \cdot (q_{47})^8 \cdot q_{48} \cdot q_{49} \cdot q_{50} \cdot q_{51} \cdot q_{52} \cdot q_{53} \cdot q_{54} \cdot (q_{55})^4 \cdot (q_{56})^2 \cdot (q_{57})^5 \cdot (q_{58})^8 \cdot (q_{59})^6 \cdot (q_{60})^{12} \cdot (q_{61})^9 \cdot (q_{62})^4 \cdot q_{63} \cdot q_{64} \cdot q_{65} \cdot q_{66} \cdot (q_{65})^3 \cdot q_{66} \cdot q_{67} \cdot q_{66} \cdot q_{67} \cdot q_{68} \cdot q_{69} \cdot q_{70} \cdot q_{71} \cdot q_{72} \cdot q_{73} \cdot q_{74} \cdot q_{75} \cdot q_{76} \cdot q_{77} \cdot q_{78} \cdot q_{79} \cdot q_{80} \cdot q_{81} \cdot q_{82} \cdot q_{83} \cdot q_{84} \cdot q_{85} \cdot q_{86} \cdot q_{87} \cdot q_{88} \cdot q_{89} \cdot (q_{90})^3 \cdot q_{91} \cdot q_{92} \cdot (q_{93})^4 \cdot q_{94} \cdot q_{95} \cdot q_{96} \cdot q_{97} \cdot (q_{98})^2 \cdot q_{99} \cdot q_{100} \cdot q_{101} \cdot q_{102} \cdot (q_{103})^5 \cdot (q_f)^{32}$$

7) Estruturas sensíveis ao contexto, com n=8:

$$(q_0)^4 \cdot (q_1)^7 \cdot (q_2)^5 \cdot q_3 \cdot q_2 \cdot q_3 \cdot q_2 \cdot q_3 \cdot q_4 \cdot (q_5)^{13} \cdot q_6 \cdot (q_5)^4 \cdot q_6 \cdot q_7 \cdot q_8 \cdot q_9 \cdot q_{10} \cdot q_{11} \cdot q_{12} \cdot q_{13} \cdot (q_{14})^{13} \cdot (q_{15})^5 \cdot (q_{16})^8 \cdot (q_{17})^6 \cdot (q_{18})^7 \cdot (q_{19})^{11} \cdot q_{20} \cdot q_{21} \cdot q_{22} \cdot q_{23} \cdot q_{24} \cdot q_{25} \cdot q_{26} \cdot (q_{27})^7 \cdot q_{28} \cdot q_{29} \cdot q_{30} \cdot q_{31} \cdot (q_{32})^3 \cdot q_{33} \cdot q_{34} \cdot q_{35} \cdot q_{36} \cdot q_{37} \cdot q_{38} \cdot q_{39} \cdot (q_{40})^{28} \cdot (q_{41})^5 \cdot (q_{42})^7 \cdot q_{43} \cdot q_{44} \cdot q_{45} \cdot q_{42} \cdot q_{43} \cdot q_{44} \cdot q_{45} \cdot (q_{46})^9 \cdot (q_{47})^8 \cdot q_{48} \cdot q_{49} \cdot q_{50} \cdot q_{51} \cdot q_{52} \cdot q_{53} \cdot q_{54} \cdot (q_{55})^4 \cdot (q_{56})^2 \cdot (q_{57})^5 \cdot (q_{58})^8 \cdot (q_{59})^6 \cdot (q_{60})^{12} \cdot (q_{61})^9 \cdot (q_{62})^4 \cdot q_{63} \cdot q_{64} \cdot q_{65} \cdot q_{66} \cdot (q_{65})^3 \cdot q_{66} \cdot q_{67} \cdot q_{66} \cdot q_{67} \cdot q_{68} \cdot q_{69} \cdot q_{70} \cdot q_{71} \cdot q_{72} \cdot q_{73} \cdot q_{74} \cdot q_{75} \cdot q_{76} \cdot q_{77} \cdot q_{78} \cdot q_{79} \cdot q_{80} \cdot q_{81} \cdot q_{82} \cdot q_{83} \cdot q_{84} \cdot q_{85} \cdot q_{86} \cdot q_{87} \cdot q_{88} \cdot q_{89} \cdot (q_{90})^3 \cdot q_{91} \cdot q_{92} \cdot (q_{93})^4 \cdot q_{94} \cdot q_{95} \cdot q_{96} \cdot q_{97} \cdot (q_{98})^2 \cdot q_{99} \cdot q_{100} \cdot q_{101} \cdot q_{102} \cdot (q_{103})^5 \cdot (q_f)^{32}$$

Dentro das estruturas repetitivas nas simulações, expressas como múltiplas passagens dos estados dos autômatos (Fig. 45), padrões podem ser definidos para encontrar medidas de semelhança entre sequências de diferentes famílias. Pela associação com caixas biológicas, a análise das diferenças e similaridades entre as repetições dos estados é uma potencial medida para distância biológica, considerando as transições dos autômatos definidos para cada família ALU, e seu resultado com valor *True*, indicando que a palavra processada de fato pertence a determinada família. Por exemplo, considere-se uma potência de n = 4 para os estados repetitivos, onde a linguagem associada é de tipo

sensível ao contexto, segundo as listas de palavras associadas. Foi analisada a possível distância genômica entre uma sequência de cada família reconhecida pelo respectivos autômatos, através de uma análise visual que representa a proposta para medições iniciais.

Figura 45 – Exemplo de repetição de estados como estruturas sintáticas nos resultados dos Autômatos para sequências reconhecidas para 6 famílias ALU

AluJ		AluSb		AluSp		AluY		AluSc		AluSq	
Est	Est <sup>n</sup>										
q <sub>0</sub>	4										
q <sub>5</sub>	4	q <sub>17</sub>	4	q <sub>41</sub>	4	q <sub>41</sub>	4	q <sub>17</sub>	4	q <sub>41</sub>	4
q <sub>55</sub>	4	q <sub>55</sub>	4	q <sub>55</sub>	4	q <sub>54</sub>	4	q <sub>55</sub>	4	q <sub>42</sub>	4
q <sub>62</sub>	4	q <sub>62</sub>	4	q <sub>62</sub>	4	q <sub>61</sub>	4	q <sub>62</sub>	4	q <sub>55</sub>	4
q <sub>93</sub>	4	q <sub>93</sub>	4	q <sub>93</sub>	4	q <sub>92</sub>	4	q <sub>93</sub>	4	q <sub>62</sub>	4
										q <sub>93</sub>	4

Fonte: Autoria própria

A classificação visual das estruturas sintáticas apresenta um agrupamento das sequências de esquerda à direita, em ordem de semelhança segundo o número de repetições dos estados reconhecendo estruturas biológicas específicas.

Também pode ser feita uma busca de diferenças entre sequências da mesma família, procurando o mesmo objetivo das árvores filogenéticas. Mesmo sendo um autômato único para determinada família, as sequências reconhecidas dentro dele podem apresentar diferenças sintáticas, sendo mais um argumento para métrica proposta (Tabela 4). Por exemplo, considere-se a família AluSq, para a qual o autômato correspondente reconheceu 5 sequências.

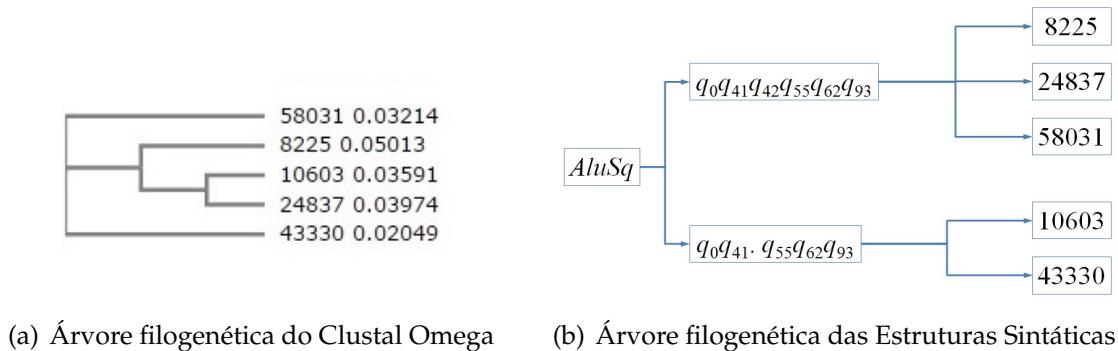
Tabela 4 – 5 sequências da família AluSq, identificadas pelo número de sequência no arquivo de saída do *Strings Generator*, agrupadas através das estruturas sintáticas expressas nas palavras associadas sobre o alfabeto dos estados do autômato, com potência de repetição  $n = 4$

8225		10603		24837		43330		58031	
St	<i>n</i>								
q <sub>0</sub>	4								
q <sub>41</sub>	4								
q <sub>42</sub>	4	q <sub>55</sub>	4	q <sub>42</sub>	4	q <sub>55</sub>	4	q <sub>42</sub>	4
q <sub>55</sub>	4	q <sub>62</sub>	4	q <sub>55</sub>	4	q <sub>62</sub>	4	q <sub>55</sub>	4
q <sub>62</sub>	4	q <sub>93</sub>	4	q <sub>62</sub>	4	q <sub>93</sub>	4	q <sub>62</sub>	4
q <sub>93</sub>	4			q <sub>93</sub>	4			q <sub>93</sub>	4

Segundo a tabela anterior, as sub-classes das sequências 8225, 24837 e 58031 e o grupo das sequências 10603, 43330 apresentam dentro delas as mesmas palavras

associadas sobre o alfabeto de estados, definidas como estruturas sensíveis ao contexto, com um número de repetições  $n = 4$ . A idéia fundamental é fornecer dados que possam ser codificados em um tipo de árvore filogenética, assim como propor uma alternativa para as árvores já conhecidas (Fig. 46).

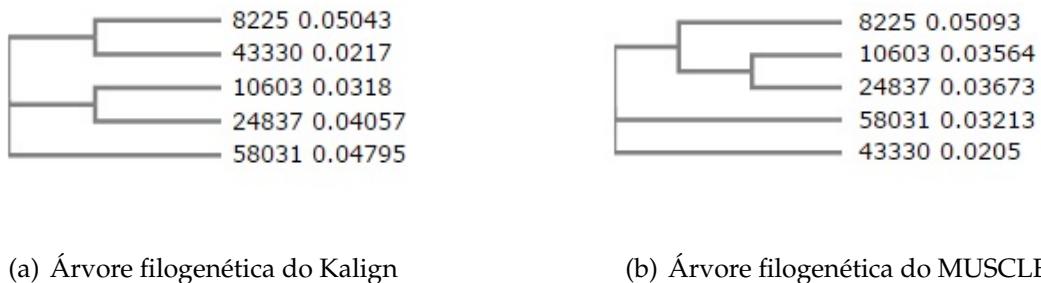
Figura 46 – Árvores filogenéticas, tanto do Clustal Omega como aquela inferida através das estruturas sintáticas da linguagem associada, para 5 sequências reconhecidas pelo autômato correspondente à família AluSq



Fonte: (a): ([LABORATORY, 2015a](#)), (b): Autoria própria

Vale lembrar que o Clustal Omega é apenas uma das múltiplas ferramentas de alinhamento e inferência de árvores filogenéticas. Existem várias técnicas (Fig. 47) que utilizam critérios diferentes para estabelecer métricas e analisar as similaridades entre sequências. Portanto, as árvores construídas a partir desses alinhamentos podem ser divergentes.

Figura 47 – Outras técnicas de alinhamento múltiplo de sequências e construção de árvores filogenéticas aplicadas nas 5 sequências reconhecidas pelo autômato para família AluSq



Fonte: (a): ([LABORATORY, 2015b](#)), (b): ([LABORATORY, 2015c](#))

No modelo proposto, a métrica para construção de árvores será obtida da análise das palavras associadas, pelo qual é possível analisá-las desde varios pontos de vista, como mostrado na análise da potência  $n = 4$  (Fig. 46). No resultado da simulação do autômato, é encontrado um grupo maior de potências, mas com apenas produções regulares (um único estado associado), ou com produções sensíveis ao contexto mas sem possibilidades de encontrar sub-classes para diferenciar as sequências reconhecidas. Nesse caso, não é possível a inferência de uma árvore filogenética.

Assim como foi feita uma árvore a partir das estruturas encontradas na potência diferencial  $n = 4$ , considere-se uma segunda potência diferencial  $n = 2$ , como segue na tabela 5:

Tabela 5 – 5 sequências da família AluSq, identificadas pelo número de sequência no arquivo de saída do *Strings Generator*, diferenciadas através das estruturas sintáticas expressas nas palavras associadas sobre o alfabeto dos estados do autômato, com potência de repetição  $n = 2$

8225	10603	24837	43330	58031
St q6 q18 q18 q42 q56 q91 q97	St q6 q18 q18 q42 q56 q91 q96	St q5 q6 q18 q18 q18 q56 q91 q96	St q6 q18 q18 q43 q42 q56 q56 q91	St q6 q18 q18 q42 q56 q91 q97 q97
$n$ 2 2 2 2 2 2 2	$n$ 2 2 2 2 2 2 2	$n$ 2 2 2 2 2 2 2	$n$ 2 2 2 2 2 2 2	$n$ 2 2 2 2 2 2 2

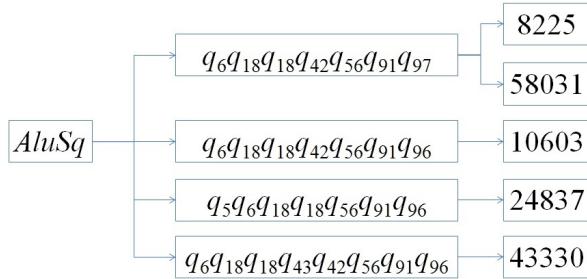
Segundo a tabela anterior, as sub-classes são definidas como segue: a primeira com as sequências **8225** e **58031**, a segunda com a sequência **10603**, a terceira com a sequência **24837** e a quarta com a sequências **43330**, fornecendo os dados para a respectiva árvore (Fig. 48).

E para concluir, considere-se a potência diferencial  $n = 6$ , assim como mostrado na tabela 6.

Segundo a tabela anterior, as sub-classes são definidas como segue: a primeira com as sequências **8225**, **58031** e **43330**, a segunda com a sequência **10603** e a terceira com a sequência **24837**, fornecendo os dados para a respectiva árvore (Fig. 49).

No entanto, as árvores nas figuras 46b, 48 e 49 são construídas utilizando, para cada uma, apenas uma das potências encontradas nas palavras associadas sobre o alfabeto de estados. É proposto um sistema de *Score*, onde um ponto para comparação

Figura 48 – Árvore filogenética inferida através das estruturas sintáticas da linguagem associada, para 5 sequências reconhecidas pelo autômato correspondente à família AluSq com  $n = 2$

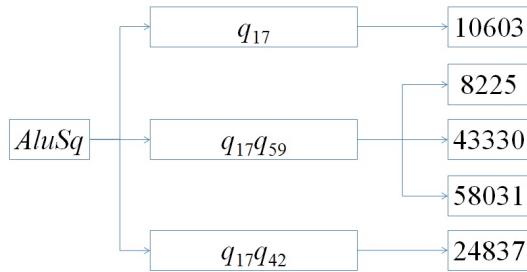


Fonte: Autoria própria

Tabela 6 – 5 sequências da família AluSq, identificadas pelo número de sequência no arquivo de saída do *Strings Generator*, diferenciadas através das estruturas sintáticas expressas nas palavras associadas sobre o alfabeto dos estados do autômato, com potência de repetição  $n = 6$

8225	10603	24837	43330	58031
St q17 q59	St q17 q59	St q17 q42	St q17 q59	St q17 q59
$n$ 6	$n$ 6	$n$ 6	$n$ 6	$n$ 6

Figura 49 – Árvore filogenética inferida através das estruturas sintáticas da linguagem associada, para 5 sequências reconhecidas pelo autômato correspondente à família AluSq com  $n = 6$



Fonte: Autoria própria

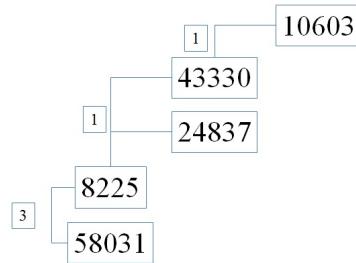
entre duas sequências é somado quando houver total igualdade e pertencem à mesma sub-classe dentro da análise de cada potência (Tabela 7).

Tabela 7 – Cálculo do *Score* para construção da árvore filogenética da família AluSp

	8225	10603	24837	43330	58031
8225	-	0	1	1	3
10603	-	-	0	1	0
24837	-	-	-	0	0
43330	-	-	-	-	0
58031	-	-	-	-	-

Uma árvore composta (Fig. 50) é inferida através da consideração de todas as potências diferenciais na linguagem associada, com  $n = \{2, 4, 6\}$ . Note-se que os números nos quadros à esquerda de cada ramificação da árvore é o respectivo *Score*.

Figura 50 – Árvore filogenética composta das sequências da família AluSq



Fonte: Autoria própria

Mesmo assim, é possível calcular uma árvore composta através do sistema *Scores* para cada família ALU reconhecida pelos autômatos.

## 5 Considerações finais

No presente capítulo serão desenvolvidos os pontos referentes às Conclusões do projeto de pesquisa, a discussão dos resultados obtidos e as propostas para trabalhos futuros baseado no modelo experimental apresentado.

### 5.1 Conclusões

Analizando os resultados obtidos no desenvolvimento do modelo experimental proposto, conlui-se o seguinte:

1. A hipótese inicial do trabalho, sobre uma gramática livre de contexto da forma  $G = (V, \Sigma, P, S)$ , inferida apartir da estrutura secundária de família AluY, que fosse capaz de gerar não apenas sequências dessa família, mas de outros consensos conhecidos, foi confirmada ao obter 57 sequências distribuídas nos 6 consensos restantes, incluso o consenso original. Biológicamente, é provada a natureza ancestral conservada das sequências ALU, além de associar diferenças nas características genéticas com estruturas sintáticas das linguagens formais.
2. Os resultados das simulações nos autômatos de pilha definidos para reconhecimento das diferentes sequências, utilizando a teoria de linguagens formais, foram validados com os dados obtidos GenBank, o qual confirma o correto funcionamento dos dispositivos reconhecedores da Linguagem Formal planejada para o modelo apresentado.
3. Uma gramática formal livre de contexto pode ser utilizada para expressar sequências genômicas de qualquer natureza com estruturas secundárias previamente definidas.
4. As estruturas sintáticas encontradas nas linguagens formais, tanto nos dispositivos geradores como nos dispositivos reconhecedores, podem ser utilizadas para definir restrições nas métricas de distância genômica.
5. Desde o ponto de vista computacional, é possível utilizar o modelo proposto para analisar amostras maiores de sequências, acrescentando o tamanho de gerações de sequências, sendo analisadas em um núcleo computacional com maior capacidade de processamento.

## 5.2 Discussão dos resultados

No final do século XX, ambas ciência da computação e biologia molecular estavam evoluindo rapidamente como disciplinas, e prever a estrutura, função e dobramento de macromoléculas por meio de estudos teóricos ou experimentais era um problema desafiador. Combinando as áreas, foi desenvolvido um modelo computacional para expressar sequências de RNA em diferentes formas, utilizando gramáticas formais livres de contexto de natureza estocástica ([SAKAKIBARA et al., 1994](#)). Os autores do trabalho, utilizando modelos ocultos de Markov, tentaram prever estatisticamente a formação de determinados fenômenos biológicos tratando-se de sequência de tRNA. Naquele momento, os estudos de formação de estruturas secundárias não estavam refinados como estão atualmente, e a informação tinha que ser inferida a partir de probabilidades estatísticas. Mas tarde, outro modelo, considerando gramáticas formais em forma de árvores adjacentes, foi desenvolvido para determinar uma sub-classe das gramáticas que fosse adequada para prever estruturas secundárias([UEMURA et al., 1999](#)) . A eficiência do algoritmo foi definida através da medida de tempo  $O(n^6)$ , onde  $n$  é o comprimento da sequência de entrada. O resultado foi um algoritmo para Árvores Adjacentes Lineares, onde são construídos pareamentos dentro das sequências procurando regiões das estruturas que sejam altamente complementares. Nesse modelo, a inferência de estruturas considera todas as formações biologicamente possíveis, como laços internos ou *hairpins*.

O modelo apresentado no presente trabalho considera apenas informações disponíveis sobre estruturas secundárias já definidas através de estudos genômicos rigorosos. O objetivo é eliminar as variáveis probabilísticas para acrescentar precisão nos resultados de expressão e reconhecimento de sequências, considerando construir algoritmos eficazes e eficiente desde o ponto de vista de tempo de processamento computacional. Utilizando a informação obtida dos alinhamentos múltiplos, as estruturas secundárias inferidas para os consensos avaliados apresentam um elevado grau de confiabilidade, pelo qual as gramáticas livres de contexto definidas conseguem reduzir a aleatoriedade encontrada em modelos anteriores. Aliás, a proposta de utilizar as estruturas sintáticas como base para estabelecer uma métrica para distância genômica apresenta possibilidades para aplicações várias na busca de padrões para fenômenos genotípicos e biológicos.

Outras das novidades apresentadas no modelo é a utilização de autômatos de pilha para reconhecimento de sequências. Nossa foco, desde o ponto de vista dos dispositivos reconhecedores, é propor uma métrica alternativa para consideração de sequências de diferentes naturezas, analisando os elementos característicos das transições do autômato, tentando definir as diferenças em nível de estruturas sintáticas e linguagem associada às transições entre estados.

A métrica alternativa apresenta diversas vantagens que, para estudos de padrões específicos em sequências, poderiam representar uma facilidade na análise de aplicações biológicas. Recentemente, múltiplas ferramentas para construção de árvores têm sido desenvolvidas ou aprimoradas, tentando aumentar a eficiência do processo. Por exemplo, elementos do algoritmo CLUSTAL incluem estimativa rápida de distância, alinhamento progressivo, e refinamento utilizando o particionamento restrito das árvores (CHENNA et al., 2003). Assim como CLUSTAL, existem vários outros como o MUSCLE (EDGAR, 2004) e o Kalign (LASSMANN; SONNHAMMER, 2005) que através de diferentes critérios biológicos, estabelecem as relações evolutivas desde diferentes pontos de vista, o qual poderia representar uma dificuldade na hora de escolher a metodologia mais apropriada. A métrica proposta procura padronizar a construção das árvores, eliminando a variação entre métodos sem esquecer a relação entre estruturas sintáticas e seu significado biológico. Tal rigor fornece uma maior facilidade para reduzir a margem de erro em estudos genéticos, assim como uma técnica mais apropriada para busca de padrões para diferentes objetivos de pesquisa, sendo baseada em métodos matemáticos e computacionais precisos.

### 5.3 Trabalhos futuros

Baseados no modelo experimental desenvolvido, são propostos os trabalhos a seguir:

1. Reconhecimento de padrões na sequência aplicação de regras de produção (gramática) e na sequências de transição de transições (autômatos) para identificar estruturas sintáticas características das linguagens.
2. Reconhecimento de padrões nas estruturas secundárias conhecidas para inferir a estrutura secundária da família AluSx, e mesmo assim, poder ser modelada através de uma linguagem formal.
3. Utilização do modelo apresentado para identificação de sequências associadas com doenças genéticas, identificando estruturas sintáticas que possam ser associadas com presença de fenómenos biológicos característicos.
4. Utilização de Linguagens Formais para geração, reconhecimento e diferenciação de outros tipos de sequências genômicas, incluso em outras espécies.
5. Automatização do Pipeline proposto, construindo uma linha integrada de processamentos.

## Referências

- AHO, A.; CORASICK, M. Efficient string matching: An aid to bibliographic search. *Communications of the ACM*, v. 18, p. 333–340, 1975.
- AMORIM, L.; PORTELA, R. **Genoma humano**. 2006. Disponível em: <[https://www.ufpe.br/biolmol/genomas\\_inicial.htm](https://www.ufpe.br/biolmol/genomas_inicial.htm)>. Acesso em: 25 de Junho de 2015.
- BOYER, R.; MOORE, J. S. A fast string searching algorithm. *Communications of the ACM*, v. 20, p. 762–772, 1977.
- CHENNA, R.; SUGAWARA, H.; KOIKE, T.; LOPEZ, R.; GIBSON, T.; HIGGINS, D.; THOMPSON, J. Multiple sequence alignment with the clustal series of programs. *Nucleic Acids Research*, v. 31, p. 3497–3500, 2003.
- CHESNOKOV, I.; SCHMID, C. Flanking sequences of an alu source stimulate transcriptions in vitro by interacting with sequence-specific transcription factors. *Journal of Molecular Evolution*, v. 42, p. 30–36, 1996.
- CONSORTIUM, I. H. G. S. Initial sequencing and analysis of the human genome. *Nature*, v. 409, p. 860–921, 2001.
- CURRAT, M.; EXCOFFIER, L. Modern humans did not admix with neanderthals during their range expansion into europe. *PLOS Biology*, v. 2, p. 2264–2274, 2004.
- DRIDI, S. Alu mobile elements: From junk dna to genomic gems. *Scientifica*, v. 1, p. 1–12, 2012.
- EDGAR, R. Muscle: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*, v. 32, p. 1792–1797, 2004.
- FABRI, R.; CELESTINO, L. C.; SILVA, A. D.; SEGURASSE, E. A filogênese da lingaugem. *Arquivos de Neuropsiquiatria*, v. 58, p. 188–194, 2000.
- GRATE, L.; HERBSTER, M.; HUGHEY, R.; HAUSSLER, D. Rna modeling using gibbs sampling and stochastic context free grammars. In: **Proc. of Second Int. Conf. on Intelligent Systems for Molecular Biology**. Menlo Park, CA, USA: AAAI/MIT Press, 1994. p. 138–146. ISBN 978-0-929280-68-4.
- GREEN, R.; KRAUSE, J.; BRIGGS, A.; MARICIC, T. A draft sequence of the neandertal genome. *Science*, v. 328, p. 710–722, 2010.
- HÄSLER, J.; STRUB, K. Alu elements as regulators of gene expression. *Nucleic Acids Research*, v. 34, p. 5491–5497, 2006.
- INFORMATION, N. C. for B. **BLAST Search Site**. 2015. Disponível em: <<http://blast.ncbi.nlm.nih.gov/>>. Acesso em: 02 de setembro de 2015.
- INFORMATION, N. C. for B. **NCBI Search Site**. 2015. Disponível em: <<http://www.ncbi.nlm.nih.gov/nuccore>>. Acesso em: 02 de outubro de 2015.

- KELLEY, D.; HENDRICKSON, D.; TENEN, D.; RINN, J. Transposable elements modulate human rna abundance and splicing via specific rna-protein interactions. **Genome Biology**, v. 15, p. 537–553, 2014.
- KLUG, W.; CUMMINGS, M.; SPENCER, C. **Conceitos de Genética**. California: Pearson Education, 2006.
- KNUTH, D.; MORRIS, J.; PRATT, V. Fast pattern matching in strings. **SIAM Journal on Computing**, v. 2, p. 323–350, 1977.
- KRAUSE, J.; LALUEZA-FOX, C.; ORLANDO, L.; ENARD, W.; GREEN, R.; BURBANO, H.; HUBLIN, J.-J.; HäNNI, C.; FORTEA, J.; RASILLA, M. de la; BERTRANPETIT, J.; ROSAS, A.; PääBO, S. The derived foxp2 variant of modern humans was shared with neandertals. **Current Biology**, v. 17, p. 1908–1912, 2007.
- KRINGS, M.; STONE, A.; SCHMITZ, R.; KRAINITZKI, H.; STONEKING, M.; PääBO, S. Neandertal dna sequences and the origin of modern humans. **Cell**, v. 90, p. 19–30, 1997.
- KUEHNEN, P.; KRUDÉ, H. Alu elements and human common diseases like obesity. **Mobile Genetic Elements**, v. 2, p. 197–201, 2012.
- LABORATORY, E. M. B. **Clustal Omega alignment tool**. 2015. Disponível em: <<http://www.ebi.ac.uk/Tools/msa/clustalo/>>. Acesso em: 02 de setembro de 2015.
- LABORATORY, E. M. B. **Kalign alignment tool**. 2015. Disponível em: <<http://www.ebi.ac.uk/Tools/msa/kalign/>>. Acesso em: 14 de julho de 2016.
- LABORATORY, E. M. B. **MUSCLE Omega alignment tool**. 2015. Disponível em: <<http://www.ebi.ac.uk/Tools/msa/muscle/>>. Acesso em: 14 de julho de 2016.
- LASSMANN, T.; SONNHAMMER, E. Kalign – an accurate and fast multiple sequence alignment algorithm. **BCM Bioinformatics**, v. 6, p. 298–306, 2005.
- LI, T.-H.; SCHMID, C. Differential stress induction of individual alu loci: implications for transcription and retrotransposition. **Gene**, v. 276, p. 135–141, 2001.
- LIU, W.-M.; CHU, W.-M.; CHOUDARY, P.; SCHMID, C. Cell stress and translational inhibitors transiently increase the abundance of mammalian sine transcripts. **Nucleic Acids Research**, v. 23, p. 1758–1765, 1995.
- MARTINS, A.; AMORIM, N.; CARNEIRO, J.; ANTUNES, P.; SAMPAIO, I.; SCHNEIDER, H. Alu elements and the phylogeny of capuchin (cebus and sapajus) monkeys. **American Journal of Primatology**, v. 77, p. 368–375, 2015.
- MATSUMOTO, Y. **Ruby Programming Language**. 2001. Disponível em: <<https://www.ruby-lang.org/es/about/>>. Acesso em: 14 de setembro de 2015.
- MIDENA, M.; NETO, J.; VEJA Ítalo. **Linguagens Formais: Teoria, Modelagem e Implementação**. São Paulo: Bookman, 2009.
- NUñEZ, L. A.; MACHADO, A. E. **Strings Generator web platform**. 2016. Disponível em: <<https://alu-seq-strings-generator.herokuapp.com/>>. Acesso em: 02 de fevereiro de 2016.

- OVCHINNIKOV, I.; GÖTHERSTRÖM, A.; ROMANOVALL, G.; KHARITONOV, V.; LIDÉN, K.; GOODWIN, W. Molecular analysis of neanderthal dna from the northern caucasus. **Nature**, v. 404, p. 490–493, 2000.
- QUENTIN, Y. Origin of the alu family: A family of alu-like monomers gave birth to the left and right arms of alu elements. **Nucleic Acids Research**, v. 20, p. 3397–3401, 1992.
- SAKAKIBARA, Y.; BROWN, M.; HUGHEY, R.; MIAN, S.; SJÖLANDER, K.; UNDERWOOD, R.; HAUSSLER, D. Stochastic context-free grammars for tRNA modeling. **Nucleic Acids Research**, v. 22, n. 23, p. 5112–5120, 1994.
- SARROWA, J.; CHANG, D.-Y.; MARAIA, R. The decline in human alu retroposition was accompanied by an asymmetric decrease in srp9/14 binding to dimeric alu rna and increased expression of small cytoplasmic alu rna. **American Society for Microbiology**, v. 17, p. 1144–1151, 1997.
- SCHAUB, M.; KELLY, W. Rna editing by adenosine deaminases generates rna and protein diversity. **Biochimie**, v. 84, p. 791–803, 2002.
- SCHMID, C. Does sine evolution preclude alu function? **Nucleic Acids Research**, v. 26, p. 4541–4550, 1998.
- SCHMUCKER, D.; CLEMENS, J.; SHU, H.; WORBY, C.; XIAO, J.; MUDA, M.; DIXON, J.; ZIPUSKY, L. Drosophila dscam is and axon guidance receptor exhibiting extraordinary molecular diversity. **Scientifica**, v. 1, p. 1–12, 2012.
- SCHNEIDER, H.; SAMPAIO, I. The systematics and evolution of new world primates - a review. **Molecular Phylogenetics and Evolution**, v. 82, p. 348–357, 2015.
- SCHNEK, A.; MASSARINI, A. **Biología**. Buenos Aires: Editorial Medica Panamericana, 2008.
- SEARLS, D. Formal grammars for intermolecular structure. In: INBS. Herndon, Virginia, USA: IEEE, 1995. p. 30–37. ISBN 0-8186-7116-5.
- SOUZA Ícaro A. **GitHub.Inc.** 2014. Disponível em: <<https://github.com/icaro-andrade1>>. Acesso em: 02 de outubro de 2015.
- THOMPSON, J.; HIGGINS, D.; GIBSON, T. Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. **Nucleic Acids Research**, v. 22, p. 4673–4680, 1994.
- UEMURA, Y.; HASEGAWA, A.; KOBAYASHI, S.; YOKOMORI, T. Tree adjoining grammars for rna structure prediction. **Theoretical Computer Science**, v. 210, p. 277–303, 1999.
- WATSON, J. **DNA: O segredo da vida**. São Paulo: Schwarcz, 2008.

## Apêndices

## APÊNDICE A – Glossário de termos de Biología e Genética

Os termos aqui definidos foram obtidos do Livro 'Biología' ([SCHNEK; MASSARINI, 2008](#)) :

**Análise proteômica:** análise do conjunto de todas as proteínas produzidas por um genoma em particular.

**Adenina:** base nitrogenada do grupo das purinas; moléculas componentes de ácidos nucleicos e de energia capazes de estabelecer ligações com grupos fosfato.

**Base nitrogenada:** a molécula básica da composição de sequências de DNA/RNA contendo nitrogênio, que tem propriedades básicas (tendência para adquirir um ião H +); são classificadas como purinas ou pirimidinas.

**Citosina:** base nitrogenada do grupo das pirimidinas; moléculas componentes de ácidos nucleicos e de energia capazes de estabelecer ligações com grupos fosfato.

**Enzima de restrição:** agentes biológicos que cortam o DNA de dupla hélice em sequências de nucleótidos específicas.

**Exon:** sequência genômica presente no mRNA após o corte e eliminação dos intrôns.

**Genoma:** totalidade do material genético de uma célula ou indivíduo. O conjunto completo de cromossomos de uma célula ou indivíduo com seus genes associados.

**Intron:** segmento de DNA que é transcrito em RNA, mas é finalmente eliminado pelas enzimas da molécula para criar mRNA.

**LINE:** sigla em inglês para Elementos Nucleares de Inserção Longa. São sequências de nucleotídeos repetitivas dentro de genoma, de comprimento longo (mais de 500 bases nitrogenadas).

**Monómero:** uma molécula simples, relativamente pequena, que pode se ligar a outras e formar um polímero.

**mRNA:** RNA do tipo mensageiro. É um tipo de moléculas de RNA que desempenham a função de carregar a informação genética desde o cromossomo aos ribossomas, onde é traduzida em proteínas.

**Nucleotídeo:** molécula bioquímica que consiste em um grupo fosfato, um açúcar de cinco carbonos (ribosa ou desoxirribosa) e uma base de purina ou pirimidina. Nucleotídeos são os blocos de construção de ácidos nucleicos.

**Retrotransposição:** efeito de uma sequência genômica para se copiar e inserir em outros lugares do genoma.

**RNA:** ácido nucleico que caracteriza-se pela presença de açúcar ribose e a pirimidina uracilo. O RNA é o material genético de muitos vírus, denominados retrovírus.

**Sequências ALU:** sequência genômica de natureza repetitiva, pertencente à família dos elementos SINE, restrita às espécies de primatas.

**SINE:** sigla em inglês para Elementos Nucleares de Inserção Curta. São sequências de nucleotídeos repetitivas dentro de genoma, de comprimento curto (máximo de 500 bases nitrogenadas).

**SRP (Partícula de reconhecimento de sinais):** sequência de nucleótidos específica em que a enzima de restrição corta a molécula de DNA.

**Splicing:** do inglês, cortar ou clivar, processo de eliminação de introns e ligação de exons do mRNA. enzima de restrição corta a molécula de DNA.

**Transposon:** uma sequência de DNA transportando um ou mais genes, e que é capaz de se mover de um lugar para outro dentro do genoma.