**Tópicos Especiais em Sistemas Inteligentes**

**Documentação Técnica da análise de algoritmos relacionados ao Graphia**

André Felipe de Paiva Cardozo

Karen Torres Teixeira

**Sumário**

[1. Introdução 3](#_Toc405206171)

[2. Descrição 3](#_Toc405206172)

[a. Visão Geral 3](#_Toc405206173)

[b. Textos Escolhidos 3](#_Toc405206174)

[3. Tecnologias 3](#_Toc405206175)

[a. Analisadas 3](#_Toc405206176)

[i. Tagger 3](#_Toc405206177)

[ii. Analise Sintática 4](#_Toc405206178)

[b. Usadas na análise 4](#_Toc405206179)

[i. Algoritmo para separação de sentenças 4](#_Toc405206180)

[ii. Algoritmo para análise geral do projeto 4](#_Toc405206181)

[4. Resultados 8](#_Toc405206182)

[5. Dificuldades Encontradas 10](#_Toc405206183)

[6. Trabalho Futuros 10](#_Toc405206184)

**Figuras**

[Figura 1 - Tela de Menu 5](#_Toc404961339)

[Figura 2 - Tela de menu ampliada 5](#_Toc404961340)

[Figura 3 - Tela de Análise da sentença antes do final 6](#_Toc404961341)

[Figura 4 - Tela de Análise da sentença após o final 6](#_Toc404961342)

[Figura 5 - Tela de Correção da Sentença 7](#_Toc404961343)

[Figura 6 - Gráfico estatístico de comparação das sentenças 7](#_Toc404961344)

1. Introdução

Monografia destinada a descrever o projeto de análise do algoritmo Graphia, expondo uma visão geral do projeto, os textos que foram utilizados nas análises, os objetivos desse projeto, as tecnologias analisadas e as que foram usadas para analisá-las, os resultados encontrados e os possíveis trabalhos futuros para estender o projeto.

1. Descrição
   1. Visão Geral

O projeto tem como objetivo analisar os resultados de extração de relações e entidades usando os algoritmos de pos-tagger e analise sintática baseados no Graphia, para atribuir semântica a essas relações.

* 1. Textos Escolhidos
* Rio de Janeiro
* Rede Globo
* Roberto Marinho

Inspiração: Escolher textos em português que envolvessem uma pessoa, um lugar e uma empresa que possuíssem relações. No nosso trabalho, utilizamos textos retirados da Wikipedia em português, para que pudéssemos extrair automaticamente os textos, primeiramente fizemos um dump da Wikipedia no dia 04/11/2014 e usamos o script em python chamado WikiExtractor.py criado por um italiano que pega o dump que vem em XML e transforma num texto limpo, descartando qualquer tipo de informação ou anotação presentes na página da Wikipedia, tais como imagens, referências, tabelas e listas.

1. Tecnologias
   1. Analisadas
      1. Tagger

Algoritmo que recebe como entrada:um texto ou uma frase e associa “tags” a cada palavra indicando o sentido semântico de cada uma.

Exemplos de “tags” são:

"V": verbo

"N": substantivo

"ADP": adposição

"ADJ": adjetivo

"DET": determinante

"PUNC": pontuação

"PRON": pronome

"NUM": número

"ADV": advérbio

"CJ": conjunção

"INTJ": interjeição

"P": preposição

"A": adjetivo

"ART": artigo

"PNT": pontuação

"PRS": pronome (não possessivo)

"CARD": cardinal

"CONJ": conjunção

* + 1. Analise Sintática

O parser é um script baseado no algoritmo do Graphia, que recebe um arquivo texto com uma sentença em seu corpo e retorna um texto com as palavras tagueadas e devidamente formatadas para que parecessem uma árvore (foram usados parênteses para essa representação).

* 1. Usadas na análise
     1. Algoritmo para separação de sentenças

O algoritmo para separação de sentenças é um algoritmo autoral, feito em JAVA, criado para automatizar a separação de sentenças pertencentes em um texto.

O funcionamento dele exige um conhecimento técnico da linguagem de programação, já que não possui uma interface visual. O processo para executá-lo exige que o usuário troque uma “tag” que define qual texto que será carregado e executado.

O resultado da execução é um arquivo texto composto pelas sentenças extraídas do texto selecionado pelo usuário, separadas por linhas.

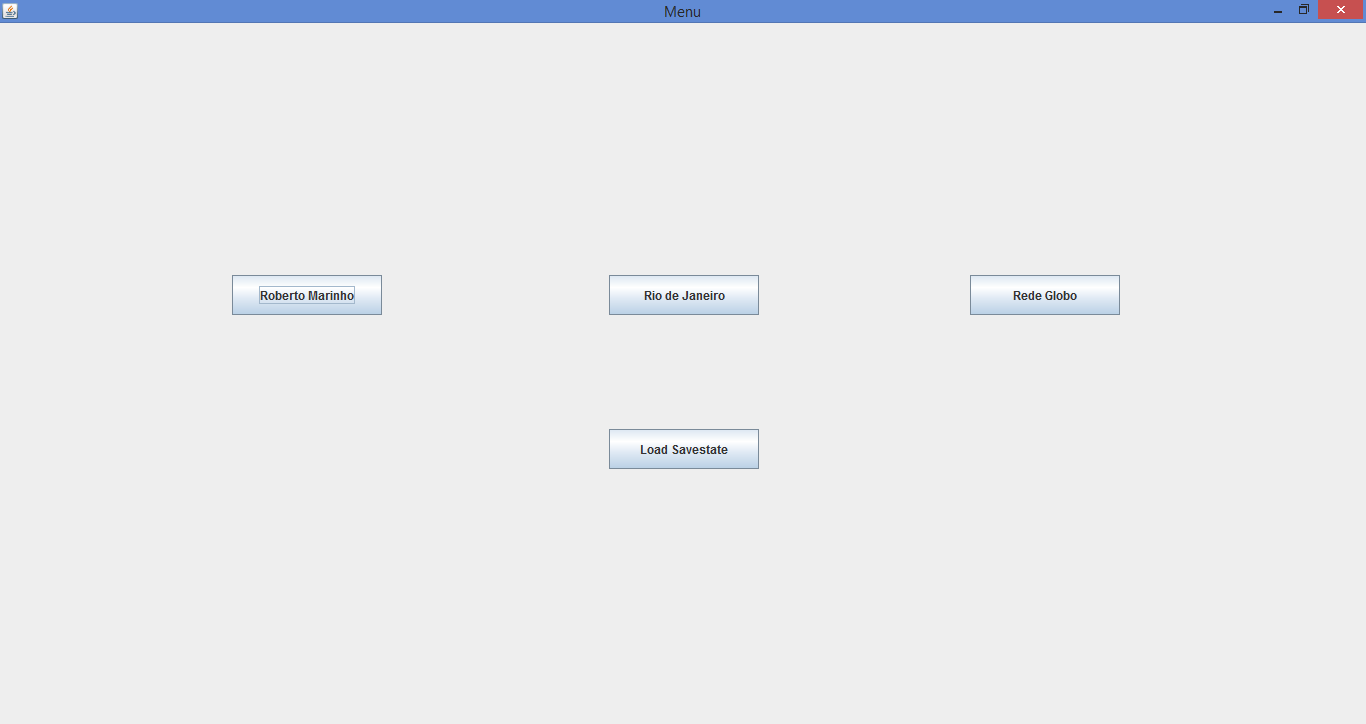
* + 1. Algoritmo para análise geral do projeto

O algoritmo usado na análise do projeto tem como objetivo facilitar a visualização dos textos, exibição de legendas em relação à “tags” geradas pelo Tagger, salvamento de estado para continuação da análise posterior dado que são muitas sentenças, exibição de dados estatísticos e salvamento dos resultados, podendo ser utilizado por um “usuário comum”.

Suas telas se dividem em:

* **Menu Inicial**

A tela de Menu Inicial possui somente 4 botões, onde 3 são pra seleção de qual texto será analisado e o outro é usado para carregar o estado que foi salvo anteriormente.

****

**Figura 1 - Tela de Menu**

****

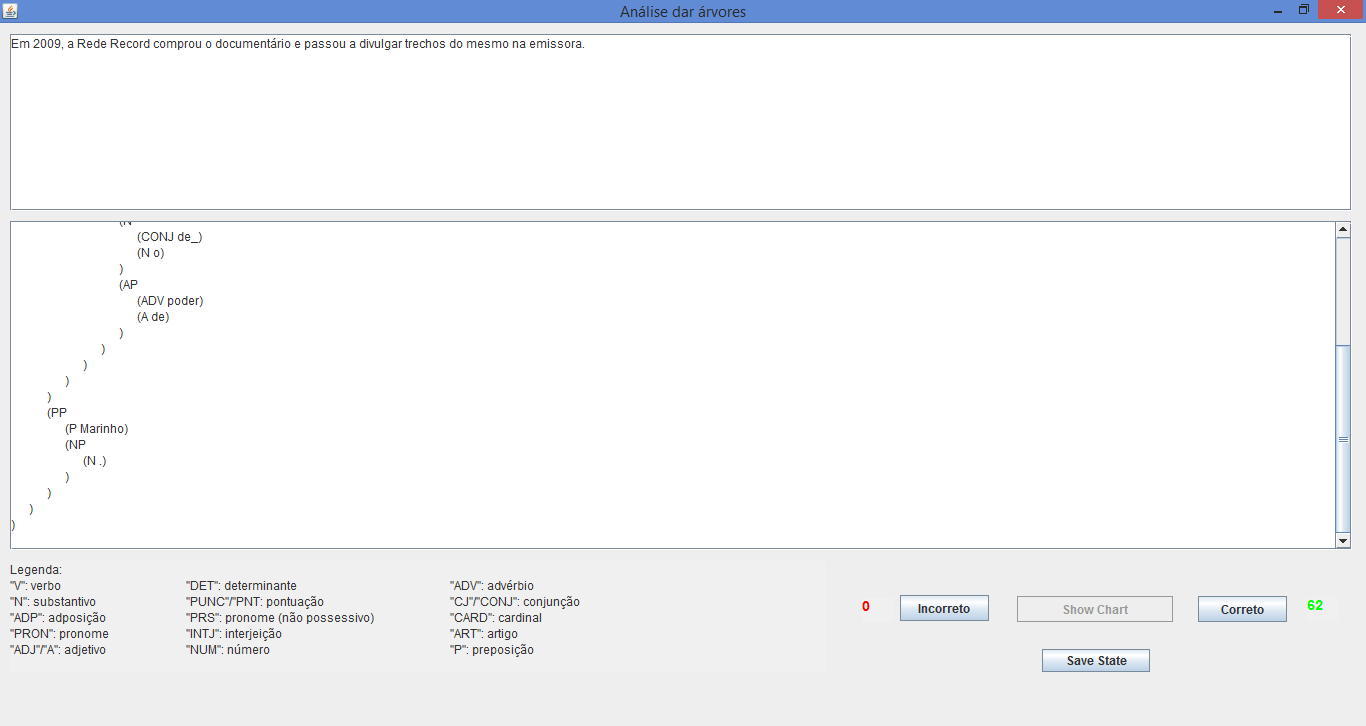
**Figura 2 - Tela de Menu ampliada**

* **Tela de Análise de Sentenças**

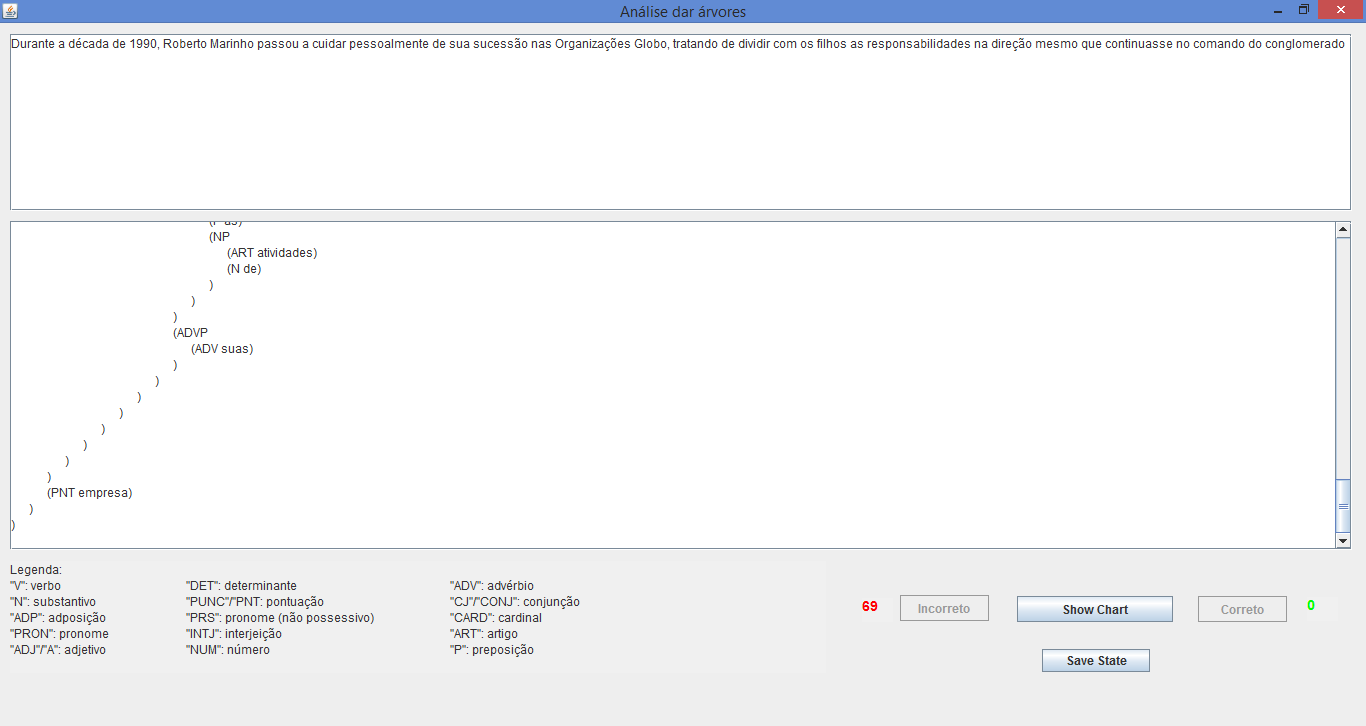
A tela de Análise de Sentenças é dividida em duas caixas de texto onde a sentença original e a árvore de sentença gerada pelo parser são exibidas, um campo com a legenda de todas as “tags” marcadas em cada termo do texto pelo Tagger, dois valores, um verde e um vermelho, que indicam o número de sentenças corretas e incorretas, respectivamente, um botão que serve para indicar que a sentença está correta e outra para indicar se está incorreta, um botão para visualização de um gráfico estatístico e um botão para salvar o estado da análise, caso seja preciso parar.

Os botões de correto e incorreto permanecem ativos até as sentenças acabarem, o contrário do botão de visualizar gráfico que só é ativado quando todas as sentenças do mesmo texto são analisadas. Ao clicar em qualquer um dos dois, o valor que indica o número de corretos ou incorretos é incrementado. Entretanto, se o botão clicado for o de incorreto, o usuário é redirecionado para a tela de Correção de Texto.

Ao final da Análise de cada texto, além da troca da forma de exibição de alguns botões, é gerado um relatório, que informa o número total de sentenças corretas e incorretas no texto junto com sua porcentagem, todas as sentenças corretas e todas as incorretas.



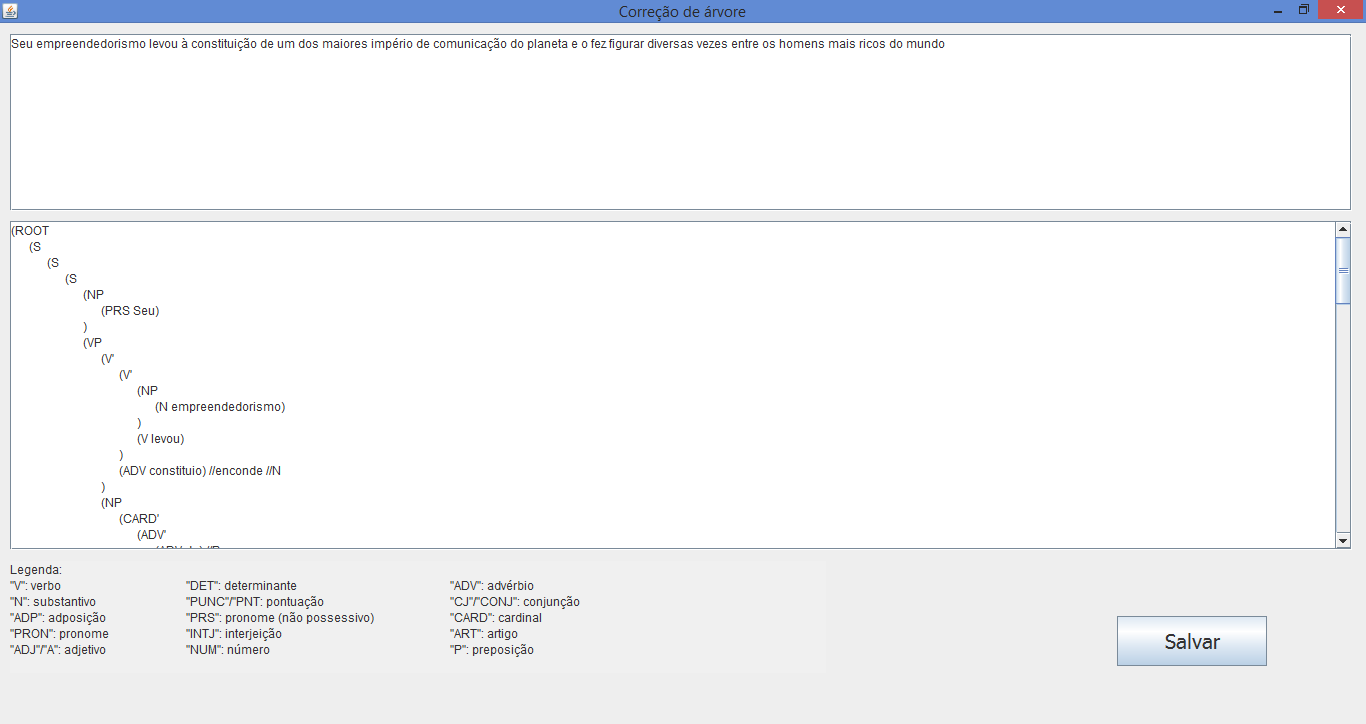
**Figura 3 - Tela de Análise da sentença antes do final**



**Figura 4 - Tela de Análise da sentença após o final**

* **Tela de Correção de Sentenças**

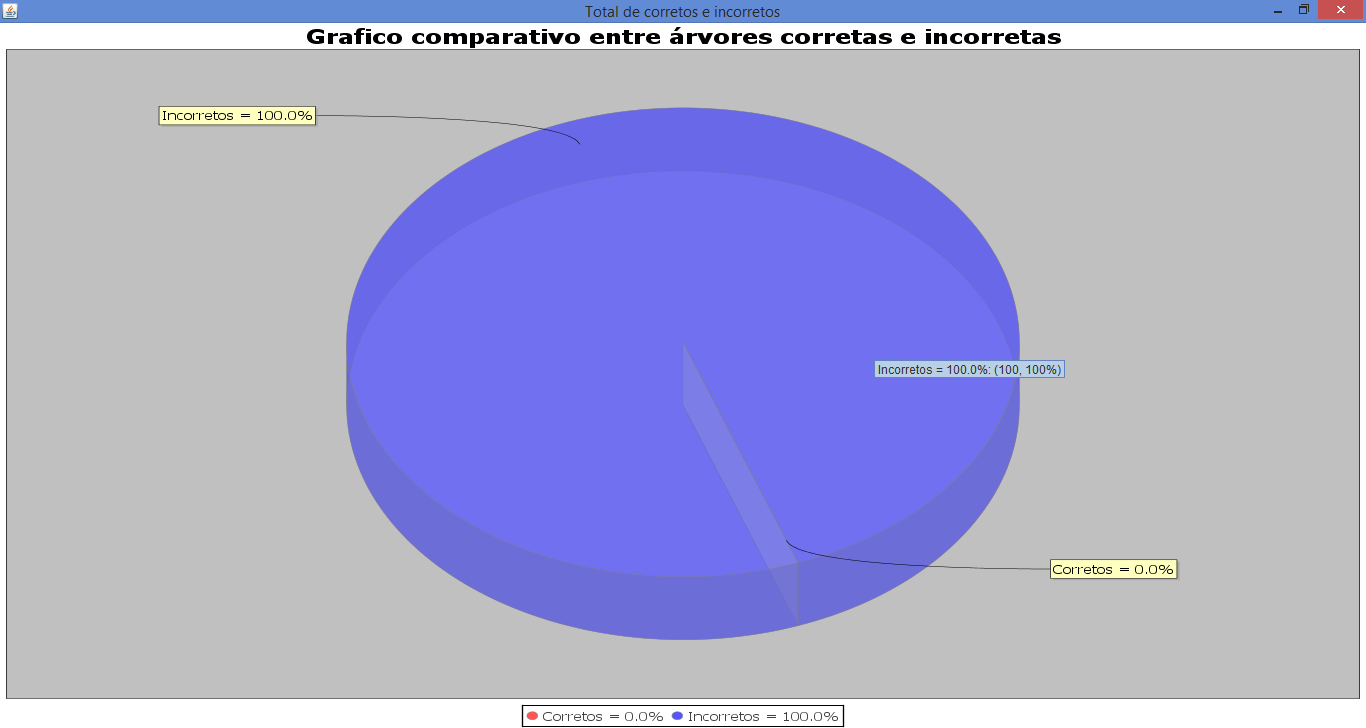
A tela de Correção de Sentenças é semelhante à tela de Análise de Sentenças, entretanto, os botões que existem na tela de análise não existem nessa tela, ela possui apena um botão de salvamento, onde, ao clicar nesse botão, é gerado um arquivo texto informando dados da sentença como um número identificador, a sentença original, a sentença parseada incorreta e a sentença parseada correta.



**Figura 5 - Tela de Correção da Sentença**

* **Tela de exibição de um gráfico estatístico**

Essa tela exibe um resultado estatístico em forma de gráfico de pizza da comparação entre sentenças corretas e incorretas definidas pelo usuário.



**Figura 6 - Gráfico estatístico de comparação das sentenças**

1. Resultados

O resultado foi unânime, pois todas as sentenças que passaram nos algoritmos de análise sintática apresentavam erros, salvo por sentenças que possuíam apenas uma palavra que estavam corretas, apesar de elas serem dispensáveis para a análise.

Algumas sentenças possuíam erros de codificação, não reconhecendo caracteres acentuados, e isso pode afetar na classificação.

Um exemplo é a palavra “organizações” que ele reconhece como “organizaes” no exemplo abaixo, classificando como pronome. Que não faz sentido, pois a palavra se quer existe. Entretanto, apesar desse problema, palavras corretas, acabaram sendo classificadas erradas, como por exemplo, “1995” no exemplo abaixo, ele classifica como nome, mas na verdade deveria ser um cardinal.

Exemplo extraído de Sentença incorreta:

(ROOT

(S

(NP

(N'

(N Proprietrio)

(PP

(P de\_)

(NP

(N'

(N as) //artigo

(PP

(P Organizaes) //encoding //N

(NP

(ART Globo) //N

(N'

(N de) //P

(N 1925) //CARD

(N a) //P

(N 2003) //CARD

(N ,) //PNT

(N foi) //V

(N um) //ART

(N de\_) //P

(N os) //A

)

)

)

)

)

)

)

)

(VP

(V homens) //V

(ADVP

(ADVP

(ADV mais)

(PP

(P poderosos) //ADJ

(NP

(ART e) //CONJ

(N influentes) //ADJ

)

)

)

(ADVP

(ADV de\_) //P

(PP

(P o) //ART

(NP

(ART pas) //enconde //N

(N'

(N em\_) //P

(PP

(P o) // ART

(NP

(ART sculo) //encoding //N

(N XX) //cardinal

)

)

)

)

)

)

)

)

)

)

1. Dificuldades Encontradas

* A separação de sentenças foi errada em casos que pontuação não indicassem final de frase, porém foi indiferente nessa fase da análise.
* Alguns bugs foram encontrados no parser, pois ele retornava um erro quando executava com o texto do Rio de Janeiro.
* Dificuldade para relembrar todas as regras gramaticais para realizar a análise.

1. Trabalho Futuros

Como trabalho futuro, possuímos a intenção de remodelar o algoritmo do Graphia para que funcione corretamente para textos em português, pois os resultados dessa análise foram negativos e a proposta desse trabalho era avaliar a necessidade de fazer essa remodelagem.