Inteligência Artificial

Trabalho 2

Gustavo Alves Rodrigues (121151605)

Marcos Vinícius Treviso (121150107)

Thayson Rafael Karlinski (111151990)

1 – Análise do programa:

O corretor ortográfico implementado depende das probabilidades à priori, probabilidade à posteriori, e a probabilidade de erro dado uma letra. Para isto foi criado uma classe específica denominada *gerador_probabilidades*.

1.1 – Gerador de probabilidades:

Nessa classe é possível gerar três tipos de probabilidades:

- Dado que o alfabeto permite letras e bigramas minúsculas e maiúsculas.
- Dado que o alfabeto permite letras e bigramas acentuadas.
- Dado que o alfabeto permite números.

Isso porque o trabalho foi feito para analisar erros ortográficos na língua portuguesa, além disso, a possibilidade dos números é acrescentada devido ao fato de muitos usuários cometerem erros de digitar um número ao invés de uma letra.

As probabilidades à priori (letras) e as probabilidades à posteriori (bigramas) foram calculadas em relação a uma base de texto, utilizando métodos simples de amostragem. As bases de texto utilizadas foram retirados de livros em modo texto na Internet, e encontram-se no diretório *books/* na raiz da pasta do trabalho. Segue abaixo uma tabela indicando seus respectivos nomes e tamanhos.

Nome	**Tamanho**
A Metamorfose	120 KB
O Apanhador no Campo de Centeio	404 KB
Sherlock Holmes – Obra Completa	3.6 MB

No entanto, as probabilidades dos erros foram aleatóriamente inseridas a partir da posição da letra relativa ao teclado português no formato ABNT2, onde os integrantes do grupo optaram pelos valores mostrados na tabela abaixo.

Acertar a letra:	Letras ao lado:	Outras letras:
60%	35%	05%

Tabela 1. Probabilidade de erro dada uma letra.

Segundo os dados da Tabela 1 é possível observar que a probabilidade de acertar a letra é maior do que errar, e errar uma letra que está logo ao lado é maior do que uma que está longe. Dito isso, é importante salientar também que esses dados não representam sua real veracidade de probabilidade num texto escrito em português.

1.2 – Corretor ortográfico:

Para o corretor ortográfico funcionar, ele necessita das probabilidades geradas pelo gerador de probabilidades, e de um arquivo contendo pares de palavras, tal que a primeira palavra corresponde a errada e a segunda corresponde a correta, onde toda palavra deve estar dentro do alfabeto das probabilidades geradas, pois senão elas não teram probabilidades associadas (tanto à priori como à posteriori).

Cada instância do corretor ortográfico tem como objetivo analisar um par de palavras do arquivo de palavras, por isso, foi criado um método de ajuda para colocar todas as palavras do arquivo para uma lista, desse modo a iteração dos pares ocorre sobre a lista e não sobre o arquivo em aberto.

Após setada as dependências do corretor ortográfico, é inicializado o algoritmo de Viterbi, com a opção booleana de normalizar as probabilidades no modelo.

O algoritmo de Viterbi se resume em fazer uma filtragem com maximização sobre os estados num instante *t. .....*

Ao fim do algoritmo de viterbi, é feito a procura do erro através do caminho mais provavél de um estado x_1 até o estado $x_{t-1.}$ E com isso é encontrado a palavra gerada pelo corretor, indicando qual seria a sequência de estados mais provavél (palavra gerada) dada a evidência (palavra errada).

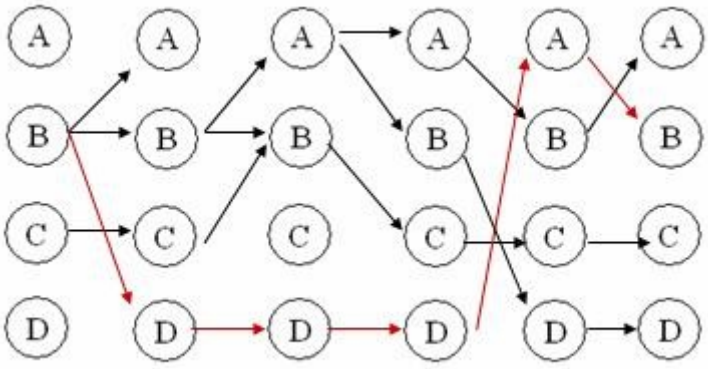

Figura 1. Representação do algoritmo de Viterbi.

Para a exibição dos resultados na tela, são utilizadas funções para medir a similaridade de duas palavras. As duas funções foram extraídas de repositórios públicos na Internet.

- *Levenshtein distance:* *http://rosettacode.org/wiki/Levenshtein_distance#Python*

- *Dice's coefficient:* *http://en.wikibooks.org/wiki/Algorithm_Implementation/Strings/Dice%27s_coefficient#Python*

Suas funcionalidades no programa se resumem a comparar a palavra gerada com a errada e com a correta, identificando assim estatísticas de erro do programa.

2 – Resultados:

A probabilidade de erro está levando em consideração o algorítmo de Levenshtein para analisar a similaridade entre a correta e a gerada pelo Viterbi. O alfabeto usado foi sensitivo ao caso, acentuado e com possibilidade de usar dígitos.

Palavras/Base Texto	**A Metamorfose**	**O Apanhador ...**	**Sherlock Holmes ...**
caza | casa	casa [100%]	caza [75.00%]	caza [75.00%]
m3sa | mesa	mesa [100%]	mesa [100%]	mesa [100%]
inferno | interno	interno [100%]	interno [100%]	interno [100%]
pezo | peso	pezo [75.00%]	pezo [75.00%]	peso [100%]
gogador | jogador	górador [57.14%]	gogador [85.71%]	gonador [71.43%]
gakeria | galeria	gameria [85.71%]	gaoeria [85.71%]	gameria [85.71%]
tijela | tigela	tinela [83.33%]	tinela [83.33%]	tinela [83.33%]
cadeifa | cadeira	cadeira [100%]	cadeira [100%]	cadeifa [85.71%]
senjora | senhora	senhora [100%]	senhora [100%]	senhora [100%]
af | ar	ar [100%]	ar [100%]	ar [100%]
musica | música	musica [83.33%]	musica [83.33%]	musica [83.33%]
mae | mãe	mae [66.67%]	mar [33.33%]	mar [33.33%]
negocio | negócio	negócio [100%]	negocio [85.71%]	negocio [85.71%]

De acordo com os resultados obtidos, fica evidente o fato do algoritmo conseguir corrigir palavras cujo bigrama da palavra errada não seja muito comum e caso o bigrama da palavra correta seja bastante comum. Entretanto, devemos também levar em consideração a posição do caractere errado no bigrama, onde seguem as regras das probabilidades de erro dado uma evidência.

Sendo assim, é correto afirmar que o programa se comportou de maneira esperada, que é corrigir apenas os erros mais grotescos e se atrapalhar com os outros. Os principais motivos é que a comparação está sendo apenas acima de bigramas, e que as probabilidades obtidas não são coerentes com o domínio da língua portuguesa.

Podemos visualizar mais claramente essa conclusão através da obtenção das probabilidades gerais de aparecimento de um bigrama, onde o bigrama acentuado é maior que seu não acentuado correspondente.

A Metamorfose	
gó: 0.3068%	go: 0.1698%
ãe: 0.1109%	ae: 0.0000%
ão: 0.8929%	op: 0.0247%
óp: 0.0288%	op: 0.0247%
õe: 0.0767%	oe: 0.0123%

O Apanhador no Campo de Centeio	
áx: 0.0154%	ax: 0.0030%
ãe: 0.0503%	ae: 0.0009%
ão: 0.9782%	ao: 0.0793%
ôx: 0.0047%	ox: 0.0034%

Sherlock Holmes – Obra Completa	
áx: 0.0026%	áx: 0.0026%
ãe: 0.0050%	ãe: 0.0050%
ão: 0.9446%	ão: 0.9446%
õe: 0.0638%	õe: 0.0638%

Na última linha da tabela de resultados, a palavra desejada era ne**gó**cio, onde a probabilidade do bigrama **gó** era uma parte fundamental para a corretude do algoritmo, como no caso da base A Metamorfose esse bigrama acentuado foi maior que o não acentuado (além das outras regras já citadas), o algoritmo conseguiu descobrir o melhor caminho correto. Já nas outras duas bases, isso não aconteceu.