Viviane Santos da Silva

Olá! Sou formada em Ciências Moleculares e, desde o início de 2014, mestranda pelo programa de pós-graduação em Linguística Geral da FFLCH, com enfoque em Linguística Computacional. Minha tese trata do problema de desambuiguação lexical automática de substantivos do português brasileiro.

Exercícios

Nesta seção, você tem acesso aos links para a página com os meus exercícios resolvidos.

Trabalho Final

Como vocês podem imaginar, eu não entendo muita coisa de Eco. Então, minhas propostas estão relacionadas a outros assuntos.

Plano A - As Colocações

Colocações são formadas por palavras que ocorrem frequentemente próximas a uma dada palavra-alvo (*node word*) e podem fornecer dicas úteis sobre o significado e o uso das palavras-alvo.

Para extrair e analisar essas colocações, eu partirei de um texto qualquer fornecido pelo usuário que será lido e transformarei as frases do texto em dicionários: os dicionários são vetores indexados — nesse caso, pelas palavras que compõem as frases. Na prática, o dicionário será usado para acessarmos as frequências das palavras de uma maneira mais fácil, em vez de usarmos as posições dos vetores, usaremos as próprias palavras como índices (ou chaves) de acesso às frequências.

Juntamente ao texto fornecido pelo usuário, a função também receberá como parâmetro uma palavra-alvo, que será o termo central das colocações a serem extraídas do texto. Uma vez que todas as frases foram transformadas em vetores, para cada termo que co-ocorrer novamente com a palavra-alvo, eu somarei 1 no dicionário para a frequência desse termo.

Depois que as frequências dos termos co-ocorrentes tiverem sido anotadas, eu irei destacar das co-ocorrências mais significativas as que, mais provavelmente, formam colocações. Para isso, vamos estabelecer o modelo nulo e o modelo alternativo: o modelo nulo é o de que as palavras seguem apenas suas distribuições de acordo com o texto como um todo, isto é, uma palavra pode estar co-ocorrendo mais com a palavra-alvo pelo simples fato de ser uma palavra com alta frequência no texto em geral. O modelo alternativo, por sua vez, é o de que as colocações são fatores importantes para a ocorrência das palavras e influenciam suas distribuições no texto.

Como saída, a função gerará um arquivo contendo as n palavras mais frequentes do texto e será impressa na tela a frequência da palavra-alvo escolhida pelo usuário. Um outro arquivo será gerado

contendo as n co-ocorrências mais frequentes em relação à palavra-alvo. A função plotará um gráfico comparando o modelos nulo e o alternativo.

Side quest: em vez de fornecer um texto, o usuário poderá pedir que as colocações sejam retiradas de um corpus default, o Floresta Sintá(c)tica, que possui interface facilmente acessível em Python pelo Natural Languege ToolKit e que eu pretendo descobrir como acessar pelo R usando alguma interface entre R e Python.

Plano B - Loteca

Vamos ~~fingir~~ supor que a Loteria é um mecanismo de sorteio honesto. Minha proposta para o plano B é de criar uma função que calcula qual a probabilidade de uma determinada aposta ser ganhadora. Pretendo calcular também a relação custo-benefício da aposta ou, em outras palavras, o prejuízo esperado; em qual caso se perderia mais: em uma aposta alta ou em várias tentativas com apostas baixas.

A partir dos dados de sorteios anteriores disponibilizados no site das Loterias (acesse-os aqui), pretendo analisar se os números sorteados não possuem viés. Caso haja algum *bias*, tenho a intenção de comparar as probabilidades de se ganhar um sorteio considerando as duas situações: sorteio enviesado e sorteio justo. Para encontrar o *bias*, seria feito um teste de hipótese tendo como alternativa a atribuição de maiores probabilidades de sorteio para os números que mais foram sorteados desde 1996 (ano a partir do qual se tem registro na tabela disponibilidade pelas Loterias).

A função receberá como entrada 1 argumento: um vetor com os valores que irão compor a aposta, sendo 15 o limite de números que se podem escolher. Como saída, a função retornará a probabilidade de a aposta ser ganhadora nas duas situações já descritas e um gráfico que compara esses dois casos.

Obs.: O R tem um pacote que permite transformar uma tabela em html em uma tabela que pode ser lida em R. Como a tabela da Mega-Sena está em html, eu usarei os pacotes RCurl e XML para acessar os dados de sorteios anteriores das Loterias.

> Viviane, a proposta A é interessantíssima e eu adorei, mas vai dar um trabalho do cão  ... principalmente a parte de criar os dicionários. Até onde eu sei, não existem no R estruturas de dados equivalentes a dicionários (ou maps usando strings como chave), vc teria que ou implementá-las ou utilizar um pacote de análise textual ou mineração de dados (que provavelmente já teria implementado o resto da sua proposta). Só isso pra mim já seria uma função excelente. Repare na complexidade: Cada frase de tamanho N seria indexada por N chaves, sendo que cada chave pode ocorrer em Z frases de todo o corpus do texto, e talvez mais de uma vez em cada frase. Assim, vc teria uma lista(corpus) de vetores de palavras (frases), e uma outra lista (dicionario) com todas as palavras de todas as frases indicando em quais frases do corpus elas ocorrem e em que posições... para

acessar isso vc precisaria de uma função que recebesse uma string e retornasse uma lista das frases, para só então poder começar a fazer o resto da sua proposta... Enfim, isso seria praticamente um pacote de análise textual. Se vc está a fim de encarar esse desafio, vai fundo que vai ficar bom pra dedéu :)

Para fins práticos e não da disciplina, se vc tem os pacotes do Python que funcionam e sabe usá-los, use o Python. Não reinvente a roda ;)

Sobre a função B, ela é mais prática e mais rápida de se fazer, mas também é um exercício interessante de cálculos de probabilidade. —- *Vitor Rios*

Olá, Vitor! Depois de ler seu comentário, eu fui pesquisar um pouco melhor quais alternativas eu teria para implementar dicionários ou algo similar em R. Um dos links em que eu cheguei foi este aqui: http://opendatagroup.wordpress.com/2009/07/26/hash-package-for-r/. Vou fazer alguns testes e ver se isso dá conta da minha proposta sobre os dicionários.

Obrigada, Viviane

Entrega do trabalho final

Acesse as informações aqui.