IST 2016
Processamento e Recuperação de Informação
Project Report - Part 2 - Group 4

1 - An approach based on graph ranking

Começamos por ler um documento textual em Inglês (parte de um capítulo de Alice In Wonderland), retirando os símbolos de pontuação, menos o apóstrofo (para manter palavras como “don’t” intactas), e depois stopwords. Posteriormente, criam-se sets de Python contendo os termos (n-gramas, com 1<=n<=3) de cada frase, sem repetições, sendo todos colocados numa lista. Também é criada uma lista com todos os termos (sem repetições) do documento e um dicionário de termos para ser usado para indexar os termos em cada array de iteração do PageRank.

O grafo do documento é representado como uma lista de dicionários, em que os índices na lista e as chaves nos dicionários são obtidos pelo dicionário de termos referido acima. Para cada possível par de termos em cada frase, é introduzida (ou *overwritten*) uma entrada com valor 1 no grafo em ambos os sentidos. O dicionário de cada termo só contém as ligações a termos que co-ocorrem consigo, uma vez que só esses influenciam o seu valor de PageRank.

O resultado de cada iteração de PageRank é armazenado como uma lista de valores, que é inserida numa outra lista de comprimento mínimo 1 e máximo 51. A iteração 0 contém os valores iniciais de PageRank (1/N, com N o número de termos). Após cada iteração, é feita uma ordenação dos termos numa lista à parte segundo o seu resultado PageRank. Essa ordenação é comparada com a da iteração anterior (incluindo a de 0) e serve de critério de paragem do cálculo do PageRank - enquanto a ordem for diferente entre iterações, continua-se o cálculo até à última iteração.

Quando terminam as iterações, os 5 melhores resultados ordenados da última iteração são retornados.

2 - Improving the graph-ranking method

A leitura dos ficheiros de texto e de termos relevantes para a colecção FAO30 e comparação dos candidatos com os termos classificados como relevantes são executados como na 1ª parte do projecto.

A extração dos termos candidatos é feita simultaneamente com a tokenização das frases do documento, num tokenizer criado por nós para um TF-IDF Vectorizer. Isto garante que há uma correspondência exacta de termos entre o grafo e o vectorizer.

A primeira melhoria é o preenchimento do grafo. Em vez de se preencher com o valor 1 para cada co-ocorrência de termos, o seu valor vai sendo incrementado.

A segunda melhoria é a utilização da fórmula no enunciado.

$$\mathrm{PR}(p_i) = d \times \frac{\mathrm{Prior}(p_i)}{\sum_{p_j} \mathrm{Prior}(p_j)} + (1-d) \times \sum_{p_j \in \mathrm{Links}(p_i)} \frac{\mathrm{PR}(p_j) \times \mathrm{Weight}(p_j, p_i)}{\sum_{p_k \in \mathrm{Links}(p_j)} \mathrm{Weight}(p_j, p_k)}$$

Para melhorar o desempenho no cálculo do PageRank evitando calcular os somatórios de Prior(p_j) e Weight(p_j,p_k), estes valores foram calculados previamente para cada termo, visto serem constantes. A soma dos pesos dos arcos ligados a cada termo são acumulados à medida que se preenche a matriz do grafo. A soma das probabilidades Prior é feita uma vez para cada termo após o preenchimento da matriz, numa nova lista.

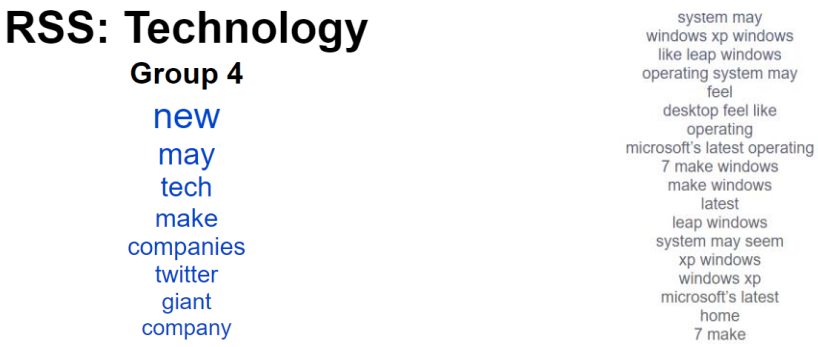
A mean-average-precision foi calculada sobre os cinco primeiros ficheiros, por questões de performance, obtendo-se um valor de 0.0059504847305. Utilizando o método original do 2º exercício do 1º projecto, tinha sido obtida a MAP de 0.00334125921929.

4 - A practical application

Aplicou-se o algoritmo de extração do exercício 1 ao conjunto de títulos e descrições de artigos num *feed* RSS do New York Times intitulado Technology[1].

Em vez de serem seleccionados os cinco melhores candidatos, são seleccionados 100. Os candidatos são depois listados num ficheiro HTML chamado “tech.html”, com os mais relevantes no topo, a cor azul mais vivo e com tamanho maior. Os termos menos relevantes são cinzentos e aparecem com menor tamanho. É possível verificar em que posição na lista cada termo está fazendo *hover* sobre ele.

Para a leitura do ficheiro XML do *feed* foi utilizada a biblioteca “lxml.etree”. O ficheiro HTML foi gerado recorrendo a uma biblioteca chamada “yattag[2]”.



À esquerda, o topo da lista de termos, à direita, o fundo da lista.

[1] http://rss.nytimes.com/services/xml/rss/nyt/Technology.xml
[2] http://www.yattag.org/