ASAP - Análise Sentimental de Artigos Políticos

Jérôme de Figueiredo
Instituto Superior Técnico - Tagus Park
Av. Prof. Doutor Aníbal Cavaco Silva
Porto Salvo, Portugal
jerome.figueiredo@ist.utl.pt

Nádia Gonçalves
Instituto Superior Técnico - Tagus Park
Av. Prof. Doutor Aníbal Cavaco Silva
Porto Salvo, Portugal
nadia.goncalves@ist.utl.pt

ABSTRACT

1. INTRODUÇÃO

Este projecto foi desenvolvido no âmbito da UC (Unidade Curricular) de Extracção e Análise de Dados Web, com o intuito de extrair e analisar documentos do natureza política. O desenvolvimento deste projecto teve também como finalidade, a aplicação de conceitos e tecnologias abordadas na UC durante as aulas teóricas e práticas.

2. EXTRACÇÃO DE FEEDS

Nesta secção, será abordada a forma como as notícias e respectivas necessárias informações são extraídas e armazenadas de forma se efectuar a análise das mesmas. Também será abordado o mecanismo de detecção de notícias duplicadas.

2.1 Método de Extracção

No início da execução deste projecto é pedido ao utilizador para se inserir o endereço de feed pretendido para análise (ex.: http://feeds.dn.pt/DN-política). Note-se que é necessário inserir o link completo como descrito no exemplo.

Um vez o feed inserido, para a extracção dos títulos das notícias é usado a ferramenta feedparser com a função *post.title*. Esta função faz parte da ferramenta feedparser e retira directamente os títulos dos artigos. A extracção do conteúdo dos artigos é feita com a ferramenta BeautifulSoup. Esta ferramenta inclui as necessárias funções para analisar paginas HTML e extrair a informação pedida segundo uma serie de critérios. Neste projecto, a ferramenta procura nas tags HTML o o critério $id = Article$ e extrai o seu conteúdo.

2.2 Indexação

A indexação da notícia é feita com a auxilio da ferramenta Whoosh. Esta função é invocada no módulos *feeds.feeds*, e tem como parâmetro o ficheiro *feeds.txt* que contém as notícias extraídas a partir do feedparser. O objectivo desta função é indexar cada notícia, para depois se efectuarem pesquisas e ser retornado o id das notícias correspondentes à pesquisa, assim como a frequência a que ela ocorre.

2.3 Armazenamento

É importante sublinhar, que a extracção do conteúdo dos artigos das notícias é feita para o ficheiro *feeds.txt* e da mesma forma, os títulos das notícias para o ficheiro *title_file.txt*.

No ficheiro $feeds.txt$, são armazenados os conteúdos de cada artigo e para cada um deste é criado e adicionado no início um id. No ficheiro *title_file.txt*, são armazenados os títulos da cada notícia com o respectivo equivalente id. Este id é colocado no início de cada artigo e no início de cada respectivo título por fim a se poderem processar algumas operações e associar cada notícia com o seu respectivo título.

Este armazenamento é feito no módulo feeds, dentro da função feeds. Esta módulo contém mecanismos para evitar o armazenamento de artigos duplicados que serão descritos na subsecção 2.4 deste relatório.

2.4 Tratamento de notícias Duplicadas

Como é comum acontecer, algumas vezes notícias repetem-se. Apesar de não existir maneira 100% precisa de evitar estes casos, este projecto inclui métodos para evitar armazenar a mesma notícia repetidamente.

Aquando da extração das notícias e dos respectivos títulos, no módulo *feeds.py* é criado um hash SHA-1 do título da notícia e armazenado no ficheiro *hash_file.txt*. A cada extracção, após ler-se o título, é criado um hash deste e lê-se o ficheiro *hash_file.txt* para se verificar se o hash já existe. Se não existir, então a notícia ainda não foi extraída e pode-se proceder a extracção descrita anteriormente. De forma inversa, se a hash existir no ficheiro *hash_file.txt*, significa que a notícia já foi anteriormente extraída e é ignorado o processo de extracção.

O algoritmos de criptografia escolhido foi o SHA-1 devido a sua robusteza. O SHA-1 produz digests de160bits levando a uma muito baixa probabilidade de repetição. No entanto o mesmo conteúdo de fonte irá criar sempre o mesmo digest, i. e., um título repetido irá gerar sempre o mesmo digest, bastando uma ligeira mudança (ex.: um acento ou uma letra), para daí ser gerado um digest completamente diferente.

3. RANKING
Nesta secção, será explicado como foi feito o ranking das personalidades em cada notícia assim como as ferramentas para geração de gráficos representantes destes valores.

4. MÉTODO DE RANKING
Uma vez extraídas a notícias e devidamente armazenadas, estas são indexadas usando a ferramenta Whoosh. Para tal, as notícias armazenadas no ficheiro *feeds.txt* são percorridas e partidas em palavras, e para cada notícia é associado um id gerado pela pela função de indexação. É de notar que as notícias no ficheiro *feeds.txt* se encontram no estado bruto com ainda alguma tags HTML. Apesar destas não influenciarem qualquer tipo de resultado, as notícias são devidamente "limpas"antes de serem indexadas e assim seguem de forma esperada. Os ficheiros do index são armazenados na subdirectoria *indexdir_proj*.

Uma vez indexadas as notícias, é invocado no script inicial o módulo *search.py* que usa também a ferramenta Whoosh. Esta ferramenta percorre os ficheiros de index e retira todos os valores de score em BM25.

Neste projecto também são criados gráficos para fácil visualização de alguns dados. Para tal, sao calculadas a medias dos rankings de cada entidade e as 20 entidades com valores mais relevantes são escritos para o ficheiro *plotBM25.dat* no subdirectório *plots* na forma ***Valor Entidade***. Adicionalmente, é criado o ficheiro *plotBM25.gp* ainda no subdirectório *plots* que contem toda a informação necessária para a formatação e criação do gráfico de ranking.

A ferramenta para a criação de gráficos é explicada com mais precisão na secção 7.

5. ANÁLISE DAS ENTIDADES
Nesta secção, será detalhado o modo como é feita a extracção das entidades políticas encontradas nos feeds de notícias.

5.1 Método de Extracção
No script inicial é invocado a função *personalidadesTratamento*, passando como parâmetro a *list_personalities*, que se encontra no módulo FIndEntities. Esta função extrai os políticos do ficheiro *personalities.txt* e passa os nome dos políticos representados nesse ficheiro, para a *list_personalities*. A outra função do módulo *FIndEntities*, é invocada a partir do módulos *sentiments.sentimentos*, antes do tratamento da notícia. Nessa função, *_SearchEntities*, passa-se como parâmetros a notícia e a *list_persolnalities*. O objectivo desta função é extrair os nomes dos políticos encontrados na notícia. De uma forma mais detalhada, pode-se dizer que com o auxilio do nltk conseguiu-se extrair as entidades da seguinte forma:

- Para cada frase da notícia (*sent_tokenize(news))* e para cada marcação de entidade da palavra *(ne_chunk(nltk.pos_tag(nltk.word_tokenize(sentence))))*
- É verificado se a marca é um node. Se for este verifica se a entidade é uma "PERSON", e em caso positivo este começa a construir o nome da pessoa buscando os valores nas folhas
- Após a junção este verifica se o nome se encontra na *list_personalities* e adiciona-o a uma lista

Esta função retorna a lista de personalidades encontradas de cada notícia.

6. ANÁLISE DE SENTIMENTOS
Nesta secção será abordada a forma de extração do sentimento captado na notícia.

6.1 Método de Análise
Este método foi divido em duas partes: **Tratamento dos Sentimentos (Subsecção 6.1.1)** e **Sentimentos da notícia (Subsecção 6.1.2)**

6.1.1 Tratamento dos Sentimentos
Numa primeira fase, ao correr o script inicial, *scriptInicial.py*, é invocada a função *sentilexTratamento*, passando como parâmetros *lexiDic*, *tuple*. Após a invocação da função, este lê o ficheiro sentilex e retém as informações mais importantes do ficheiro tais como a root, que é a palavra que pode aparecer no texto, e a polaridade da root. Após a extração da informação importante do ficheiro, este passa a primeira palavra da root como key do dicionário *lexiDic*, e na tuple coloca o resto da expressão, o comprimento da expressão e a polaridade. O motivo pelo o qual se criou este dicionário foi para facilitar a pesquisa dos sentimentos e das expressões que se pode encontrar na notícia, facilitando assim a comparação entre palavras.

6.1.2 Sentimento da notícia
Após o tratamento dos sentimentos, já se está apto para analisar se a notícia tem uma conotação positiva ou negativa em relação ao tema que se está a analisar, a política. A função *sentimentos*, tem como parâmetros *lexiDic* e *opinionDic*. A função extrai os sentimentos da notícia fazendo com que se consiga fazer um cálculo relativo à polaridade sentimental, determinando assim se a notícia é sentimentalmente positiva ou negativa. Esta função foi realizada com auxilio da ferramenta nltk, para a separar a notícia em palavras. A comparação foi feita a partir do dicionário que continha os sentimentos já tratados com cada palavra da notícia. Quando se encontra uma correspondência, é verificado se aquela palavra tem continuação, i.e, se faz parte de uma expressão. Caso não tenha quer dizer que a palavra é única e é somada à polaridade da notícia a polaridade da palavra. Se a palavra for negativa, é subtraído um valor ao total da polaridade da notícia, se for positiva é somado um valor e em caso neutro o valor de polaridade total não é alterado. Caso o dicionário verifica que a lista contém uma expressão, então é comparado a próxima palavra com a próxima palavra da expressão, e assim sucessivamente. Este ciclo só termina se caso encontrar correspondência ou se percorrer todos os elementos da lista e não encontrando nenhuma correspondência. Caso este encontre correspondência, então verifica a polaridade da expressão, subtraindo-se à polaridade da noticia um valor se for negativa, somando se for positivo e mantendo o valor em caso da ser neutra. Após a análise este passa ao dicionário *opinionDic*, o id com o respectivo titulo da notícia como key e passa como value, uma tupla com uma lista de políticos encontrados na notícias *politiciansList* e o peso do sentimento da notícia.

7. FERRAMENTA DE PLOTTING

A fim de apresentar alguns valores relevantes recolhidos durante o projecto, foi usada a ferramenta ***Gnuplot***.

O Gnuplot é altamente flexível e permite criar variados tipos de gráficos. Este programa só necessita de um ficheiro de entrada (no caso deste projecto do tipo *.dat*) que recebe os valores a serem tratados e outro ficheiro do tipo *.gp* com as definições necessárias para o tratamento do ficheiro *.dat* e para a apresentação e criação do gráfico.

Devido a alguma limitações na apresentação final dos valores de ranking, o número de valores apresentados teve de ser reduzido ao top 20 por forma a ser possível a leitura dos resultados. Isto foi devido à existência de mais de 1500 valores aquando dos teste efectuados, tornando-se impossível perceberem-se tantos pontos num gráfico. Adicionalmente, a quantidade de valores de ranking obtidos são sempre muito elevados e podem aumentar ainda a valores bem superior aos obtidos nos testes.

Os gráficos obtidos durante os testes pode ser visto no Apêndice, nas figuras 3 e 4.

8. RESULTADOS OBTIDOS

Nesta seção serão apresentados os resultados obtidos relativos a:

- Sentimento da notícia (secção 8.1),
- Score de cada notícia (secção 8.2).

8.1 Sentimentos das Notícias

Os resultados obtidos para na análise de sentimentos relativo às entidade são devolvidos na forma: ***id notícia - título notícia***, ***políticos***, ***resultado final da soma da análise de sentimentos***.

Ex.:

1 - Passos promete para breve guião da reforma do Estado

políticos:
Paulo Portas
Passos Coelho

Valor da notícia: 4

Os resultados obtidos nos testes encontram-se demonstrados em anexo na figura 2

8.2 Score de cada Entidade

Os resultados obtidos no scoring BM25 são devolvido na forma de percentagem sendo apresentados os valores para cada notícia e um valor médio geral.

Ex.:

Vítor Gaspar
Score Doc: 5 ==> 4.77950871124
Score Doc: 3 ==> 4.04907450246
Average rank: 4.41429160685

Este output imprime o nome da personalidade seguido de cada notícia com o seu respectivo id e score. Finalmente, para cada político, no final de serem levantados todos os scores, é calculado e apresentado o valor médio total.

Os resultados obtidos nos testes encontram-se demonstrados em anexo na figura 1

9. CONCLUSÕES

Na realização deste projecto, foi interessante perceber como alguma técnicas de recolha de informação e análise das mesmas é efectuado. Existe uma variada quantidade de ferramentas e o estudo e escolha das que pareceram mais adequadas demonstrou-se recompensador e interessante.

É de notar que para o cálculo de ranking, a recolha de personalidades vem de um ficheiro já pré-definido. Assim, este calculo não é feito de forma "inteligente"em que são estudadas variações dos nomes das entidades (ex.: Pedro Passos Coelho = Passos Coelho) não reflectindo assim a precisão esperada. Este ponto poderá ser melhorado no futuro aumentando a precisão desta ferramenta.

Também é importante referir que a ferramenta foi pensada e criada em especifico para os feeds de notícias de ordem política do Diário de Notícias e Jornal de Notícias, não garantindo assim o funcionamento com outros feeds.

Finalmente, a precisão geral da ferramenta poderia ser globalmente melhorada usando técnicas mais avançadas para o reconhecimento das variações de nomes das personalidades e com o uso de técnicas mais avançadas de ranking e análise de sentimentos. Estes poderão ser pontos a pegar e desenvolver com mais profundidade no futuro.

APPENDIX
A. RESULTADOS DE TESTES

```
----------------------------------------
Joaquim Raposo
----------------------------------------
Vítor Gaspar
Score Doc: 5 ==> 4.77950871124
Score Doc: 3 ==> 4.04907450246
Average rank:  4.41429160685
----------------------------------------
Artur Figueiredo
Score Doc: 18 ==> 2.94679918401
Average rank:  2.94679918401
----------------------------------------
José Mendes Bota
Score Doc: 17 ==> 4.7487128394
Average rank:  4.7487128394
----------------------------------------
Manuel Fino
Score Doc: 17 ==> 3.83507408522
Score Doc: 22 ==> 3.4924700975
Score Doc: 7 ==> 2.92305738058
Average rank:  3.41686718776
----------------------------------------
Seruca Emídio
----------------------------------------
Raul Soares Veiga
Score Doc: 13 ==> 2.81908673438
Score Doc: 7 ==> 2.69887732357
Score Doc: 17 ==> 2.52769107738
Score Doc: 18 ==> 2.19732448129
Average rank:  2.56074490415
----------------------------------------
António Melo Pires
Score Doc: 14 ==> 4.11937270289
Score Doc: 8 ==> 3.94605590548
Score Doc: 3 ==> 2.19023250699
Average rank:  3.41855370512
----------------------------------------
```

Figure 1: Exemplo: Output de ranking

```
10 - Marques Guedes chama CDS "principal partido da oposição"

Politicos:
Pedro Passos Coelho
Paulo Portas
Paulo Portas
Paulo Portas

Valor da noticia:  -1


3 - Uma semana negra

Politicos:
Manuela Ferreira Leite
Pacheco Pereira
Rui Rio
Carlos Abreu Amorim
Passos Coelho
Passos Coelho

Valor da noticia:  1


17 - Cavaco convoca Conselho de Estado para dia 20 às 17h

Politicos:
Cavaco Silva
Jorge Sampaio
Marcelo Rebelo Sousa
Francisco Balsemão
Cavaco Silva

Valor da noticia:  0


5 - Portugal está no "caminho certo" para sair do programa

Politicos:

Valor da noticia:  3
```

Figure 2: Exemplo: Output da análise de sentimentos

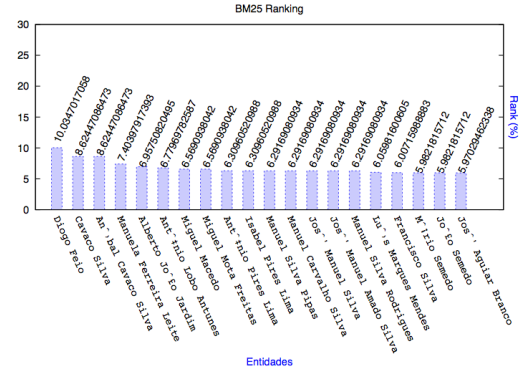

Figure 3: Exemplo: Gráfico de rankings

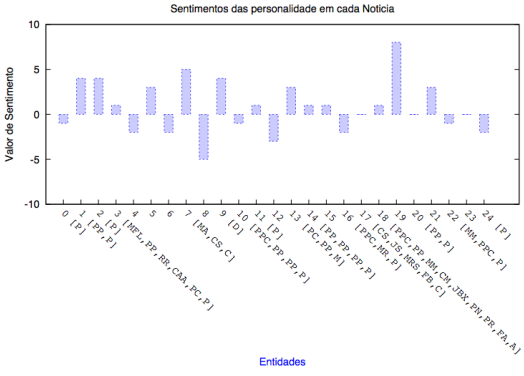

Figure 4: Exemplo: Gráfico análise de sentimentos