EADW Political Personalities

Grupo 3

Artur Balanuta - 68206
Instituto Superior Técnico
artur.balanuta@ist.utl.pt

Dário Nascimento - 68210
Instituto Superior Técnico
dario.nascimento@ist.utl.pt

ABSTRACT

A extracção e análise de feeds de notícias online permite inferir inúmeros aspectos de uma determinada sociedade ou amostra populacional. Caracterizando aspectos como a relevância de determinado acontecimento, qual o sentimento geral da população relativamente a um dado acontecimento ou personalidade. Este projecto permite-nos obter informações como a opinião sobre os políticos portugueses, sobre o respectivo partido e qual a evolução da sua notiaridade ao longo do tempo. Analisamos ainda quais os países que têm mais relevância no nosso paradigma político e quais as personalidades mais citadas. Deste modo podemos inferir aspectos como a opinião que os média propagam sobre um determinado político ou o partido com melhor reputação ou que acontecimentos melhoram a opinião dos políticos nos média.

Categories and Subject Descriptors

Information Retrial, Sentiment Analysis [

Keywords

]: Information Retrial, Sentiment Analysis, Politics,Opinion

1. INTRODUÇÃO

O objectivo deste trabalho consiste na construção de um sistema de recolha, extração e analise de feeds (RSS) de notícias. Vamo-nos focar na recolha de personalidades Politicas Portuguesas. Como tambem a analise e clasificação dessas opinioes. Vamos tambem criar um motor de busca onde iremos poder efectuar pesquisas sobre esas personalidades.

2. REQUISITOS FUNCIONAIS

A nossa solução é modular afim permitir a combinação de funcionalidades e a rápida extensabilidade. A Figura1 explicita a arquitectura da nossa solução. Consultamos vários sites de feeds, extraimos os links para a notícia original, descarregamos a noticia, extraimos as entidades presentes e analisamos o sentimento de cada uma das frases da noticia e associamos esse sentimento à entidade de cada frase. Em todos os casos a informação é guardada em varias tabelas utilizando a faramenta **sqlite3**. As tabelas foram desenhadas por foram a poderemos efectuar pesquisas eficientes e complexas sobrea a base de dados. A linguagem de programação utilizada foi o **Python**. Utilizamos tambem algumas faramentas para processar a informação como é o caso do **Whoosh**, o **Feedparser**, o **Beautiful Soup** e o **NTLK**.

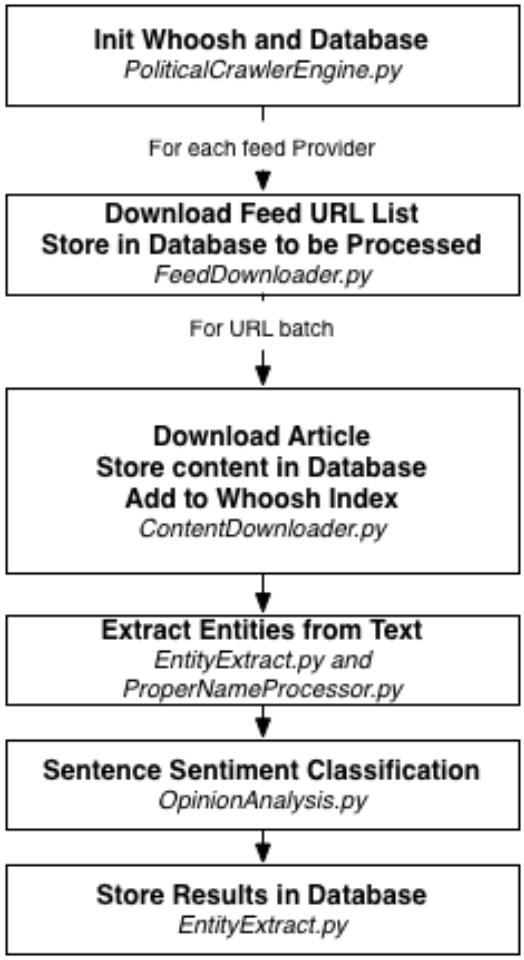

Figure 1: Arquitectura do Sistema

2.1 Colecção e Armazenamento de Notícias

A recolha de noticias é iníciada descarregando os feeds dos prinicípais jornais politicos portugueses: Diário de Notícias, Jornal de Notícias, Diário Económico, Sol. De cada um destes feeds utilizando a faramenta **feedparser**, extraimos o URL da notícia original e a data do feed. Estes dados são guardados na base de dados.
Imediatamente a seguir ou à posteriori, descarregamos o contéudo da notícia através do URL. Para tal utilizando a farramenta **Beautiful Soup**. Construímos um HTML parser para cada um dos websites. O contéudo é separado em Título, Sumário e Article de modo a atribuírmos relevância distinta a cada parte do texto. A entrada na base de dados é actualizada com o contéudo e marcado como processado.
Estes dados são então utilizados para realizar as pesquisas e para extracção de informação, assuntos abordados nas próximas secções.
Criamos tambem um sistema de caching das paginas descaregadas. Ao longo do projecto verificamos que exestia um bottleneck na extração da informação da Web, logo para acelerar o processo da aquisição da informação guardamos as paginas em disco utilizando varias threads e depois efectuamos o seu processamento. Para distinguir os ficheros, o seu nome é dado em função de uma função de resumo (SHA-1) do URL original.

2.2 Pesquisa de Notícias

Despois da obtenção do artido do seu sumario e titulo procedemos a indexaçao desta infoemação. A indexação é efectuada utilizando a ferramenta **Whoosh** utilizando o modelo **BM25**. Para diferenciar e modificar o pesso de cada parte da messagem por forma a obter melhores resultados utilizamos a seguinte metodologia: O ID sera o URL original do Artigo que é o nosso identificador unico no Sistema. E para cada par **(ID, TEXTO)** que introduzimos no Whoosh a parte do texto sera composta pelas tres partes do artigo de seguinte modo.

$$\textbf{TEXTO} = \textbf{10 x Titulo} + \textbf{2 x Sumario} + \textbf{Artigo}$$

O Resultado de uma pesquisa é um conjunto de links por ordem crescente de relevancia. Como temos o as entidades presentes na base de dados para cada noticia apresentamos os 7 Entidades mais relevantes. Tambem apresentamos o Sentimento da noticia em relação a cada entidades, como da noticia em si.

2.3 Extração de Entidades

A extracção de nomes de entidades no nosso sistema é suportada por uma lista nomes de personalidades previamente conhecida. O objectivo é que novos nomes sejam adicionados a esta lista e que nomes já existentes sejam reconhecidos com boa confiança.
Com base numa lista de entidades previamente conhecida, gerámos uma tabela de nomes próprios designada de "ProperNounTable"e uma tabela de Entidades com uma popularidade pré-definida e a popularidade adquirida nas noticías politicas designada de "EntitiesTable".
Para cada nome que o NLTK indentifica como potêncial nome próprio, verificamos:

- Se o nome não pertence à lista de nomes próprios errados conhecida ("O","A", "Desde", entre outros)
- Se o nome pertente à lista de nomes próprios conhecidos. Caso pertença, é concatenado com o nome em formação. Caso contrário é definido como candidato a nome próprio. Se o nome seguinte for também um nome próprio, então ambos são adicionados à "ProperNounTable", expandindo a lista de nomes próprios conhecida.
- Se os nomes próprios constituintes do nome da entidade desconhecida correspondem em mais de 60% ao nome de uma das entidades conhecidas.De todas as entidades conhecidas candidatas a este nome, é seleccionada a que tem maior reputação nas noticias e a que tem maior reputação pré-definida. Caso não haja nenhuma entidade conhecida com o nome, este nome é adicionado como nova entidade.

Ao nível do processamento de texto, obtámos por unificar todos os nomes sem acentos graves ou agudos de modo a que palavras como: "Luís"ou "Luis"fossem equivalentes. Esta situação permitiu aumentar em 3 terços o número de entidades detectadas. Em particular o número de ocorrências do ministro "Vitor Gaspar"aumentou imenso.
No caso se novos nomes o NLTK não foi capaz de identificar com precizão se um determinado nome é ou não um nome proprio, o que provocava um grande numero de falsos positivos. Isso acontece pelo facto de o NLTK classificar as plavaras como Inglezas, o que resultava em resultados de pouca precisão. Para ultrapassar este problema criamos a nossa propria base de dados de Nomes Proprios. A Base de dados é formada por uma combinação de nomes extraidas de tres fotes: A lista de Politicos fornecida para o projecto, NLTK Corpus Floresta TreeBank (Portugues) e NLTK Corpus MacMorpho TreeBank (Brasileiro/Portugues). Apesar de conseguiremos reultados muito bons tinhamos muitos nomes proprios que apareciam muitas vezs e não se adequavam ao nosso caso. Mais a frente vamos explicar como conseguimos melhorar os resultados retirando as palavras desnecessarias atravez de um fitro.

2.4 Analise do Sentimento

A análise de sentimento permite determinar a reputação e ter um feedback em tempo real do panorama político nacional. A opinião que os média publicam sobre um determinado político ou partido político é extremamente relevante e decisiva para o futuro político-partidário de um país. Em 2013, as eleições em Itália sofreram uma forte influência por parte dos média. A maioria dos média televisivos italianos são propriedade de Silvio Berlusconi, então candidato à liderança, fizeram campanhas de opinião posítiva ao político e fazendo com que a opinião de um político envolvido em vários escandalos se altera-se e deste modo obtivesse quase o mesmo número de votos que Luigi Bersani.

1º - Pré-processamos o texto e extraímos as entidades de cada frase, como explicado na secção anterior.

2.4.1 Sentence-Level Sentiment Analysis

Através do mecanismo da secção anterior, extraimos as entidades de cada frase. Assumimos que existe apenas 1 opinião

por cada frase. No entanto, caso exista mais do que uma entidade, tentámos dividir a frase em subfrases.
Uma das maiores difículdades, especialmente em notícias políticas, é a análise de opiniões contídas nas citações de outros políticos. Estas citações são muitas vezes sarcásticas e dificeis de analisar. A frase é objectiva ou subjectiva, isto é, a frase contém alguma opinião, visão ou crença subjacente? Apenas considerámos frases subjectivas porque frases objectivas tem uma análise muito mais complexa.

2.4.2 *Document-Level Sentiment Analysis*

Admitimos um método de análise de ıSupervied Learning em que classificamos o documento em 3 classes: Positivo, Negativo e Neutro. Ao nível do documento

2.4.3 *Sentiment Lexicon Acquisition*

Existem 3 formas de obter o léxico de sentimentos: ıManualmente, ıDictionary-based e ıCorpus-based. Uma hipótese possível de criação do léxico seria criar uma pequena lista de adjectivos manualmente. Utilizando um dicionário online, como a infopedia [**?**], verificar que se trata de um adjectivo, extrair os seus sinónimos e antónimos e realizar a mesma análise iterativamente para cada um. Um processo semelhante a um url-crawler mas com palavras num dicionário. Este processo geraria um léxico classificado como palavras positivas ou palavras negativas.
Em alternativa, utilizámos o léxico Sentilex [**?**]. Processámos este léxico para uma lista de expressão:Part-of-Speech:Gênero:Opinião. A opinião das expressões foi dividida em Positiva, Negativa ou Neutra.