

Autoclippping

Recolha automática de notícias para uma taxonomia de um tópico específico

José Oliveira^{1,2}, Nuno Escudeiro^{1,3}, e Ricardo Almeida^{1,4}

¹ Instituto Superior de Engenharia do Porto, Porto, Portugal

² 1130763@isep.ipp.pt

³ nfe@isep.ipp.pt

⁴ ral@isep.ipp.pt

Resumo A monitorização dos media com o objetivo de compilar notícias sobre determinado assunto, processo denominado *clipping*, exige cada vez mais recursos conforme aumenta a quantidade de informação *online*. Usar soluções de aprendizagem automática para auxiliar os editores de boletins temáticos pode ser uma maneira muito eficiente de oferecer suporte ao recorte automático de documentos na *web*. Este artigo apresenta soluções para a recolha automática de páginas *web* de *seed websites* de interesse para recolher notícias potencialmente interessantes para o boletim da European Association of ERASMUS Coordinators. O processo de recolha devolve dados não estruturados que são pré-processados para serem depois explorados por técnicas de aprendizagem automática. Em particular, são usados classificadores de texto para rotular notícias recentes sobre uma taxonomia que representa o tópico de interesse. O *web crawling* responsável por fazer a recolha de notícias também recolhe estatísticas sobre a qualidade das notícias extraídas de cada *seed website* para que o modelo possa adaptar automaticamente a sua frequência de procura para evitar o desperdício de recursos ao recuperar dados de sites estáticos. A avaliação preliminar mostra que esse processo pode recolher notícias relevantes com uma redução significativa no tempo e no esforço exigidos ao editor do boletim informativo.

Palavras-chave: *Web Crawling*, *Text Mining*, Aprendizagem supervisionada, Classificação.

1 Introdução

A European Association of ERASMUS Coordinators (EAEC), uma associação com mais de 150 membros, na sua maioria universidades europeias, promove o desenvolvimento da educação no espaço europeu, em particular, por meio do programa Erasmus+, tem como uma das atividades a para a produção de um boletim mensal no qual exploram conteúdos publicados no último mês em *websites* de interesse. O processo de pesquisa e análise de informação é feito de forma manual pelos seus colaboradores. A EAEC tem a vontade de automatizar a recolha de informação e disponibilizá-la em diretorias associadas a um tópico. A

diminuição do tempo despendido na recolha de notícias, permitirá uma redução na duração da produção do boletim.

A procura automática de conteúdos exige a utilização de um *web crawler* que perante a indicação de determinadas sementes (*URL*), faz o *download* dessas páginas *web*, extrai as hiperligações nelas contidas, e recursivamente continua o *download* das páginas identificadas nessas hiperligações. A sobrecarga no servidor *web* e a não duplicação do conteúdo descarregado, em múltiplos ciclos, impõem desafios adicionais no desenho do programa [1]. O *web scraping*, um tipo de ferramenta muito similar ao *web crawler* - como a Scrapy, uma *framework* de *open-source* [6]-, tem a vantagem de conseguir a extração exclusiva do corpo de texto da notícia, significando à posteriori um processamento do texto mais simples. No entanto, as classes e/ou as *tags* de HTML onde os textos de interesse se encontram embutidos, necessitam de ser manualmente indicadas para cada domínio. A navegação para outras hiperligações também pode ficar comprometida caso esses *websites* não se encontrem na configuração do *spider* [2].

As páginas *web* uma vez descarregadas são processadas para o apuramento dos tópicos abordados. Inicialmente existe a necessidade da filtração, transformação e representação do conteúdo dos ficheiros recolhidos. De seguida, um conjunto de modelos é treinado e feita uma avaliação dos resultados. Por fim, os dados são apresentados ao utilizador numa plataforma web.

Para a criação de dados estruturados para a aplicação do classificador, torna-se necessário o pré-processamento dos ficheiros de texto simples. O texto é dividido em partes (*tokenization*), são removidos todos os sinais de pontuação e substituídos os espaços vazios e caracteres não-textuais por um único espaço vazio (remoção das *stop words*). Às palavras resultantes com recurso a dicionários as palavras são reduzidas à sua forma mais simples, seja utilizando o seu radical (*stemming*) ou um sinónimo mais comum que a própria palavra (*lemmatization*). Por último, aplicam-se técnicas para a redução da dimensão dos termos, onde os termos com maior significado são mantidos e os mais irrelevantes e redundantes são eliminados (*feature selection*) [3,4].

A classificação de texto é um método supervisionado de aprendizagem que aprende a determinar a classe de um dado texto com base num modelo treinado previamente. Esta classe pode ser o tópico focado no texto. Um conjunto de modelos de classificação, com uso de métodos estatísticos ou de *deep learning*, é treinado sobre parte dos dados de teste. A precisão do modelo é avaliada, categorizando a parte do conjunto de dados de testes não usados para treino e comparando os resultados com as categorias atribuídas manualmente.

Um *website* torna-se a solução mais simples para a visualização dos resultados do projeto, em qualquer dispositivo, em qualquer lugar. Pretende-se através da aplicação: visualizar todos os documentos resultantes dos *crawlings* e a categoria atribuída pelo classificador, bem como validar e/ou modificar a categoria atribuída.

Este artigo propõe um sistema de recolha de páginas *web* de domínios de interesse para o boletim da EAEC e seu tratamento. O conteúdo descarregado

necessita de ter algumas partes do texto eliminadas ou transformadas, para posterior uso de um classificador, onde é feita a atribuição da categoria a que a notícia pertence. Uma plataforma *online* deve servir como ferramenta de visualização, e manipulação, dos ciclos de *crawling* e das notícias recolhidas e categorizadas.

Neste documento encontram-se na Secção 2 as principais técnicas e métodos aplicados, na fase de recolha de documentos e o respetivo processamento, seguindo-se a conclusão na Secção 3.

2 Metodologia

Neste capítulo encontram-se descritos os principais componentes da aplicação. No primeiro capítulo, descreve-se a forma como o processo de obtenção de dados ocorre. De seguida, são apresentadas as técnicas aplicadas para a transformação dos dados para uma forma reduzida e organizada. Por fim, no último capítulo são descritos os algoritmos de classificação usados no projeto para a produção dos resultados. Na Figura 1 encontra-se o diagrama da visão geral dos componentes e atividades envolvidas.

2.1 Web Crawling

O algoritmo de *web crawling* descarrega páginas *web* em massa. A sua utilização é comum na indexação das páginas *web* para posterior pesquisa, por parte de motores de pesquisa. O *download* em massa e a capacidade do *web crawler* navegar para as hiperligações presentes nas páginas intervencionadas, recursivamente, obriga a alguns cuidados. No presente projeto adotaram-se as regras:

- Não procurar páginas nos diretórios ascendentes. Exemplo, para o *seed URL* `www.exemplo.pt/news`, só as páginas *web* presentes dentro de "news" e diretórios descendentes são acedidas. Locais como `www.exemplo.pt/about` ou `www.exemplo.pt/agenda` não são abrangidos;
- Utilização de um valor temporal aleatório de espera entre reaquisição;
- Filtragem de formatos de ficheiros não desejáveis através da sua extensão. Exemplos: `js`, `css`, `iso` e `img`.

A utilização de filtros reduz a taxa de transferência no disco e na rede, bem como rejeita conteúdos não utilizados na classificação. Um tempo de espera entre requisições deve ser tido em conta para não existir sobrecarga no servidor *web* remoto, nem para mecanismos contra ataques de *denial-of-service* sejam ativados [5].

A frequência de publicações é diferente para cada *website*, como tal, executar o processo diariamente para um *website* que publica duas vezes por semana, por exemplo, despende recursos de modo pouco eficaz. Como tal, a frequência de conteúdos disponibilizados para cada *website* deve ser analisada com base nos registos do *web crawler* e devem ser agendados ciclos diferentes de *crawling* para cada *seed URL*.

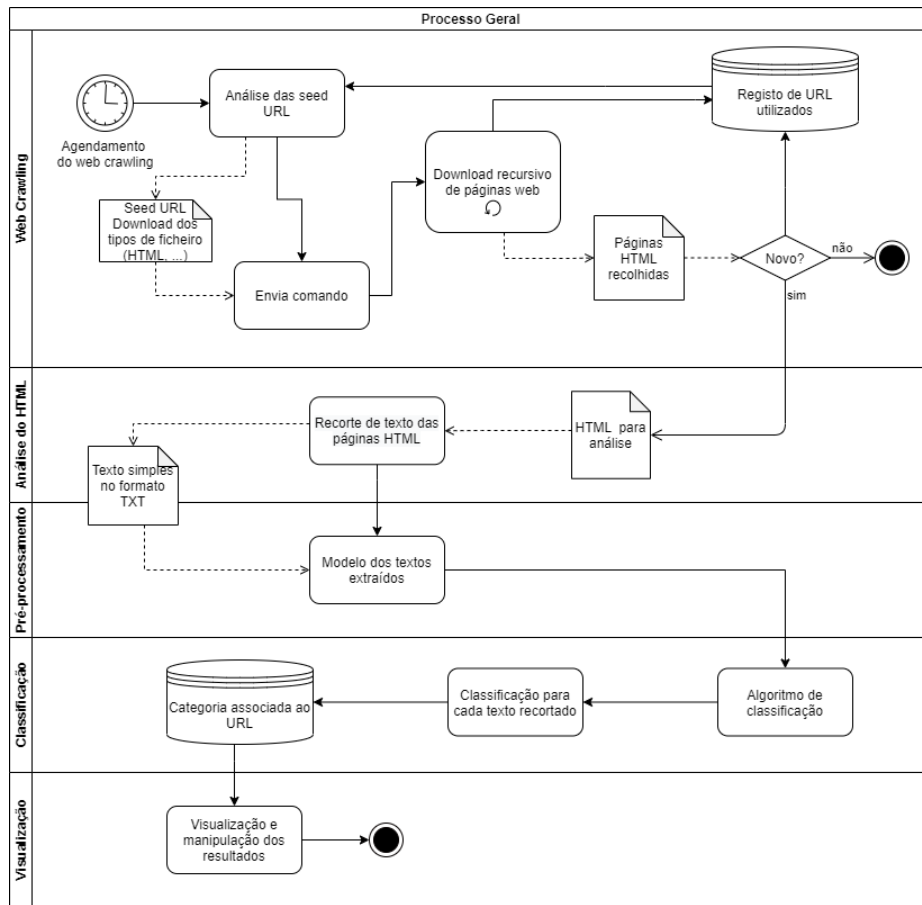


Fig. 1. Visão geral do processo de recolha e classificação das páginas web.

2.2 Pré-processamento

Em *text mining*, os dados são pré-processados para a extração de termos com valor e conhecimento de dados não estruturados. A convergência dos termos para uma forma mais simples e comum é um dos propósitos. Termos com pouco valor ou redundantes são eliminados.

Para a limpeza e preparação do texto são aplicadas funções como, conversão de todas as palavras para minúscula, remoção dos sinais de pontuação, *lemmatization* e remoção de *stop words*.

O texto normalizado é guardado num vetor TF-IDF. As frequências dos termos em cada documento são ponderadas, penalizando as palavras que aparecem com mais frequência no conjunto de texto dos documentos. Os termos mais raros têm maior probabilidade de serem representativos do tópico.

2.3 Classificadores

Nesta secção é enumerado um conjunto de algoritmos de classificação, uns com abordagens estatísticas e outros que usam redes neuronais artificiais, usados para a criação de um modelo de classificação para o conjunto de dados baseado em *websites* de interesse para a EAEC.

Os hiperparâmetros dos modelos de classificação são ajustados com recurso das funções Radomized Search e Grid Search, num cenário de 3-Fold Cross Validation, obtendo-se uma otimização dos parâmetros do modelo, resultando em melhores previsões.

Algumas sugestões de algoritmos de classificação: K- Nearest Neighbour (KNN), Multinomial Naive Bayes, Gradient Boosting, Random Forest, wSupport Vector Machines (SVM), Logistic Regression e Long Short Term Memory (LSTM).

3 Conclusão

Um sistema de procura e classificação de conteúdos é uma necessidade para muitas organizações. O uso de tecnologias para a extração de conhecimento de informação não estruturada é uma tarefa complexa, que envolve diferentes processos consoante a informação de origem e a informação pretendida.

O sistema desenvolvido faz recolha, análise e disponibilização de notícias de relevo para a produção do boletim da European Association of ERASMUS Coordinators. Tratando-se de um sistema de aprendizagem ativo, com a classificação de um número maior de registos na plataforma *web*, por parte do utilizador, as previsões tendem a ter uma menor taxa de erro.

Para trabalho futuro, sugere-se a continuação do treino dos diferentes modelos de classificadores e caso em algum ponto algum obtenha melhores resultados que o predefinido, seja feita a alteração de classificador usado de forma automática.

Bibliografia

1. Najork, M. (2017). Web Crawler Architecture. In Encyclopedia of Database Systems (pp. 1–4). https://doi.org/10.1007/978-1-4899-7993-3_457-3
2. vanden Broucke, S., & Baesens, B. (2018). Practical Web Scraping for Data Science. Practical Web Scraping for Data Science, 155–172. <https://doi.org/10.1007/978-1-4842-3582-9>
3. Hotho, A., Nürnberger, A., Paaß, G., & Ais, F. (2005). A Brief Survey of Text Mining. Retrieved from <http://www.crisp-dm.org/Process/index.htm>
4. Ting, S. L., Ip, W. H., & Tsang, A. H. C. (2011). Is Naïve Bayes a Good Classifier for Document Classification? International Journal of Software Engineering and Its Applications (Vol. 5).
5. Olston, C., & Najork, M. (2010). Web Crawling. Foundations and Trends R in Information Retrieval, 4(3), 175–246. <https://doi.org/10.1561/15000000017>
6. Scrapy developers. (2008). Scrapy — A Fast and Powerful Scraping and Web Crawling Framework. <https://scrapy.org/>