Visualização de redes definidas pelas hiperligações entre artigos da Wikipedia

Ângela Mestre[1], Elizabeth Carvalho[1,2]
[1] Universidade Aberta, [2] CIAC- Centro de Investigação em Artes e Comunicação
1101088@estudante.uab.pt, Elizabeth.Carvalho@uab.pt

Resumo

A Wikipedia é um dos portais mais populares da internet, contendo mais de 40 milhões de artigos em qualquer uma das línguas em que está disponível. Os artigos da Wikipedia referenciam outros artigos por meio de hiperligações. As hiperligações traduzem a ligação e interdependência entre artigos. Neste contexto, este trabalho apresenta uma aplicação para a visualização de um grafo de conhecimento definido por hiperligações entre artigos da Wikipedia Inglesa, partindo de um artigo inicial. Dado que, em geral, o número de hiperligações dos artigos da Wikipedia é muito elevado, a aplicação baseia-se num critério natural de seleção em função da sua relevância. Os nodos do grafo obtido têm hiperligações para artigos da Wikipedia, o que proporciona um modo alternativo de navegar na Wikipedia por meio de um grafo.

Palavras-chave: Wikipedia, visualização de informação, grafo de conhecimento.

Title: Visualizing networks defined by links in Wikipedia articles

Abstract

Wikipedia is one of the most popular websites over the Internet with more than 40 million articles in any of the languages in which it is available. Links in Wikipedia articles target related articles. Links translate connections and dependencies upon Wikipedia articles. In this context, this work presents an application to visualize a knowledge graph defined by links in English Wikipedia articles, starting from a base one. Since, in general, the number of links in Wikipedia articles is very large, the application uses a natural criterion for selecting links in terms of their relevance. Moreover, the graph nodes have hyperlinks to Wikipedia articles which gives an alternative way to browse Wikipedia.

Keywords: Wikipedia, information visualization, knowledge graph.

1. Introdução

A Wikipedia é uma enciclopédia online, plurilinguística, de conteúdo gratuito, sem fins lucrativos, baseada num modelo de edição livre por parte de qualquer utilizador. Estima-se que sejam editadas duas páginas por segundo e criados em média 561 novos artigos por dia ["Wikipedia: Statistics", 2018]. A Wikipedia é de facto um dos portais mais populares da internet, sendo que o conteúdo em Inglês contém 5 645 988 artigos num total de cerca de 40 milhões de artigos em qualquer uma das 302 línguas em que está disponível ["Wikipedia:Size of Wikipedia", 2018; "List of Wikipedias", 2018]. Neste trabalho, por simplicidade, considera-se apenas a Wikipedia em Inglês.

A Wikipedia tem sido alvo de estudo por parte de muitos investigadores no âmbito da Estatística e da Inteligência Artificial, e.g., [Warren, Airoldi & Banks, 2008; Bruce, Gao, Andreae & Jabeen, 2012; Yasseri & Kertész, 2013; Nastase & Strube, 2013; Malo, Siitari & Sinha, 2013; Hachey, Radford, Nothman, Honnibal & Curran, 2013, Nothman, Ringland, Radford, Murphy & Curran, 2013]. Os artigos da Wikipedia obedecem a uma organização estrutural definida ["Wikipedia:Manual of Style/Layout", 2018]. A título de exemplo, há uma secção principal, designada de *lead section*, que é a primeira secção e ocorre antes da tabela de conteúdos. Nela é feita uma breve introdução ao artigo ["Wikipedia:Manual of Style/Lead Section", 2018]. O guião da Wikipedia sugere que primeiro parágrafo e a primeira frase devem definir o tópico de modo a elucidar o utilizador não especialista sobre o seu contexto. Opcionalmente, os artigos poderão ter uma tabela *infobox* que contém o sumário dos factos mais importantes ["Help:Infobox", 2018]. As hiperligações entre artigos da Wikipedia são de reconhecida importância no sentido de aumentar a utilidade da Wikipedia ao direcionar o utilizador para outros artigos relevantes para o tópico. Para evitar *overlinking*, de acordo com as orientações da Wikipedia, as hiperligações deveriam aparecer apenas para a primeira ocorrência da palavra ou frase ["Wikipedia:Manual of Style/Linking", 2018]. No entanto, isso nem sempre sucede, sendo que em geral as mesmas hiperligações ocorrem várias vezes dentro do mesmo artigo.

É de esperar que a organização pré-definida dos artigos da Wikipedia permita aos utilizadores encontrar facilmente as hiperligações mais importantes para os tópicos dos artigos. Recentemente, Lamprecht, Lerman, Helic & Strohmaier (2017) analisaram os cliques dos utilizadores na Wikipedia Inglesa durante um mês. Com base num modelo probabilístico, concluíram que a estrutura do artigo influencia a navegação do utilizador, sendo que as hiperligações no início do artigo e na *infobox* são favorecidos em detrimento das restantes hiperligações.

Este trabalho enquadra-se no contexto de organização dos artigos da Wikipedia e da seleção da informação mais importante. Foi desenvolvida uma aplicação que permite visualizar uma rede definida pelas hiperligações entre artigos da Wikipedia. Tendo em conta a extensão dos artigos da Wikipedia e o número elevado de hiperligações neles existentes, um grafo com tantos nodos quantas as hiperligações do artigo, não seria, em geral, acessível e de clara visualização. Assim sendo, a aplicação escolhe apenas as hiperligações mais importantes. Um propósito da aplicação é o de proporcionar uma forma

alternativa para navegar na Wikipedia. Para esse efeito, os nodos da rede têm hiperligações que direcionam o utilizador para os artigos da Wikipedia a eles associados.
A organização deste artigo é a seguinte. A secção 2 descreve algum trabalho relacionado. A secção 3 dedica-se à descrição do protótipo da aplicação. Em particular, a secção fundamenta o critério de relevância das hiperligações, descreve a aplicação em detalhe e a arquitetura do *software*, reporta sobre os testes efetuados e sobre os resultados da implementação e testes. A secção 4 faz algumas considerações sobre a aplicação final, expondo também ideias de trabalho futuro.

2. Trabalho Relacionado

Esta secção descreve duas aplicações encontradas na web no contexto da aplicação que se pretende desenvolver neste trabalho. As aplicações foram desenvolvidas por Cornec (2015) e por Bhuyan (2015), sendo que ambas têm com objetivos muito semelhantes à aplicação aqui proposta. Outros exemplos de aplicações parecidas são a aplicação de Fernandes & Carvalho (2018) que usa a informação das *infoboxes* para criar árvores genealógicas de investigadores, ou a aplicação de Kashcha (2017) que constrói grafos a partir das sugestões de vídeos no YouTube [https://www.youtube.com/].

Figura 1. Exemplo da aplicação de Cornec para o artigo sobre Miles Davis (https://en.wikipedia.org/wiki/Miles_Davis) (Source: Cornec (2014)).

WikiGalaxy

A aplicação desenvolvida por Cornec (2015) recebe um termo de pesquisa, procura o artigo na Wikipedia relativo a esse termo, devolve o número de hiperligações do artigo classificando-as como altamente relacionada ou intimamente relacionada. O grafo obtido a partir das hiperligações é mostrado em 3 dimensões. No grafo também é indicada a classificação dos artigos por categoria (sociedade, geografia, história, política, etc.). É ainda facultada a visualização direta da *infobox* do artigo e a hiperligação para a página do artigo na Wikipedia.
Tal como referido por Cornec, as páginas da Wikipedia foram percorridas com recurso ao analisador WikiXMLJ ["delip/wikixmlj", 2016] (Application Programming Interface (API) de Java [https://www.java.com/en/]). Foi também usada a biblioteca HTTP do Python [https://www.python.org/], Urllib3 [https://urllib3.readthedocs.io/en/latest/], para o acesso directo aos URLs das páginas da Wikipedia. A visualização foi feita com recurso às ferramentas GraphViz [https://www.graphviz.org/] e three.js [https://threejs.org/]. A figura 1 mostra um exemplo de visualização da aplicação de Cornec.

Visualizing Wikipedia as a Graph

A aplicação desenvolvida por Bhuyan (2015) tem como objetivo criar uma nova maneira de navegar na Wikipedia através de um grafo de conhecimento que evidencie as conexões entre tópicos. Dado um artigo da Wikipedia, o autor implementou duas funcionalidades de visualização: (i) quais os tópicos (artigos) para os quais o artigo dado é importante. (ii) quais os tópicos que são importantes para o artigo dado.

A aplicação seleciona as hiperligações mais importantes assumindo que as referidas nos primeiros parágrafos do artigo são mais importantes do que as restantes hiperligações do documento. São assim considerados apenas as hiperligações da secção introdutória. A todas as hiperligações extraídas é dada a mesma importância. Os grafos obtidos são grafos estrela com poucos nodos. Ao clicar no nodo central alterna-se entre a visualização das hiperligações (mais importantes) para artigos que o artigo dado refere, ou a visualização das hiperligações (mais importantes) para artigos em que o artigo dado é referido. Por outro lado, ao clicar nos restantes nodos, o nodo escolhido passa a ser central obtendo-se o grafo para esse nodo.

A aplicação de Bhuyan (2015) usa um programa em Python [https://www.python.org/] para extrair as hiperligações das páginas da Wikipedia com recurso à biblioteca Beautiful Soup [https://www.crummy.com/software/BeautifulSoup/bs4/doc/]. A base de dados para o grafo é criada com o software Neo4j [https://neo4j.com/]. Para a visualização é usada a ferramenta D3.js [https://d3js.org/]. As Figuras 2 e 3 mostram os grafos de Bhuyan relativos aos artigos sobre Religion (https://en.wikipedia.org/wiki/Religion) e India (https://en.wikipedia.org/wiki/India).

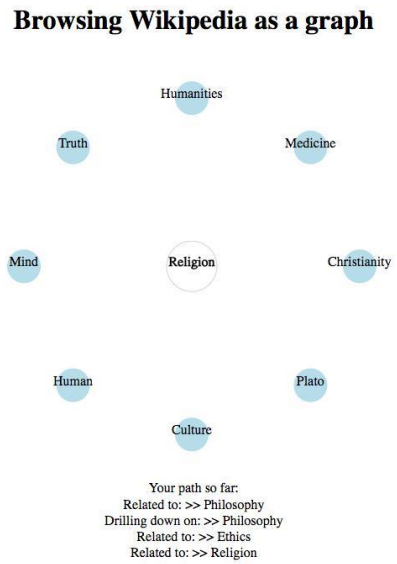

Figura 2. Exemplo da aplicação de Bhuyan para o artigo sobre Religion (https://en.wikipedia.org/wiki/Religion) (Source: Bhuyan (2015)).

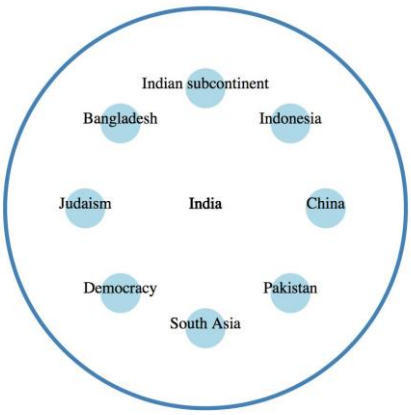

Figura 3. Exemplo da aplicação de Bhuyan para o artigo sobre India (https://en.wikipedia.org/wiki/India) (Source: Bhuyan (2015)).

3. Protótipo

Os artigos da Wikipedia estão ligados entre si por meio de hiperligações. Redes que, por um lado, tornem transparente a interdependência entre artigos; por outro permitam ao utilizador navegar entre artigos por meio das hiperligações são de grande utilidade. De facto, as redes definidas pelas hiperligações têm gerado grande interesse na comunidade científica. Um exemplo é o Wikigame ["Wikipedia:Wiki Game", 2018], um jogo cujo objetivo é chegar a um artigo destino desde um artigo de partida por meio hiperligações, tentando minimizar o número de hiperligações necessárias para o percurso e o tempo despendido.

Esta secção apresenta uma aplicação que permite visualizar ligações relevantes entre artigos da Wikipedia partindo de um artigo inicial. A visualização consiste num grafo de conhecimento cujos nodos têm hiperligações para artigos da Wikipedia. Um objetivo da aplicação, é proporcionar uma maneira alternativa de navegar na Wikipedia por meio de um grafo. Outro propósito da aplicação é dar um conhecimento abrangente sobre o tópico do artigo. A navegação por meio de um grafo (desde que manejável, i.e., com poucos nodos) permite ao utilizador reduzir ainda mais o tempo de consulta do artigo, poupando-lhe o tempo de pesquisa das hiperligações nos artigos.

O software foi desenvolvido em Python [https://www.python.org/] usando a biblioteca Beautiful Soup [https://www.crummy.com/software/BeautifulSoup/bs4/doc/] para extrair os links das páginas da Wikipedia. A visualização do grafo é feita com recurso à biblioteca para visualização de informação D3.js [https://d3js.org/] e à adaptação do software desenvolvido por Bellamy-Royds [Bellamy-Royds, 2018] ao presente trabalho.

Nesta secção, através da análise da estrutura dos artigos da Wikipedia, fundamenta-se a escolha do critério para a relevância das hiperligações. Depois detalha-se a aplicação e a arquitetura do software.

3.1 Critério de relevância das hiperligações

O *layout* típico dos artigos da Wikipedia ["Wikipedia:Manual of Style/Layout", 2018] define a seguinte ordem de apresentação:

(i) *infobox*;
(ii) corpo do texto que inclui a secção principal e a Tabela de Conteúdos (TOC);
(iii) apêndices;
(iv) *bottom matter*.

Tal como descrito em ["Wikipedia:Manual_of_Style/Lead_section", 2018], a secção principal deve proporcionar uma visão geral do tópico, enquanto a *infobox* contém o sumário da informação. A tabela de conteúdos aparece automaticamente em artigos com mais de quatro secções. Dado que atenção dos utilizadores centra-se sobretudo no topo do artigo e na *infobox*, as hiperligações mais importantes encontrar-se-ão provavelmente na secção principal e na *infobox*. Contudo, a análise de artigos específicos, revelou que após o título de dada secção muitas vezes aparecem artigos referidos como *main article* (ou *further*

information ou *see also*) como se vê, por exemplo, na Figura 4. Ora, o tópico desses artigos será certamente importante para o artigo dado, pois não bastou referi-los, foi necessário dedicar-lhes uma secção.


was already referred to as **Portugal**.

History [edit]

Prehistory [edit]

Main article: History of Portugal

The early history of Portugal is shared with the rest of the Iberian Peninsula located in Sou
derives from the joined Romano-Celtic name Portus Cale. The region was settled by Pre-C
Gallaeci, Lusitanians, Celtici and Cynetes, visited by Phoenicians, Ancient Greeks and Cart
Republic dominions as Lusitania and part of Gallaecia, after 45 BC until 298 AD.

The region of present-day Portugal was inhabited by Neanderthals and then by *Homo sapi*
northern Iberian peninsula.[36] These were subsistence societies that, although they did n
organized societies. Neolithic Portugal experimented with domestication of herding anima
or marine fishing.[36]

It is believed by some scholars that early in the first millennium BC, several waves of Celts
inter-married with the local populations, forming different tribes. Modern archeology and

Figura 4. Exemplo de um artigo assinalado como *main article* no artigo sobre Portugal (https://en.wikipedia.org/wiki/Portugal).

Ao analisar os ficheiros HTML dos artigos verifica-se, por exemplo, que os artigos sobre *Monoid* (https://en.wikipedia.org/wiki/Monoid) ou *Macroeconomics* (https://en.wikipedia.org/wiki/Macroeconomics) contêm tabelas parecidas com a *infobox* mas que não são identificadas como tal nos ficheiros HTML. Neste trabalho, essas tabelas são tratadas como *infoboxes*.

Consideram-se, assim, as seguintes listas de hiperligações:

- Hiperligações dos parágrafos do texto:
  - -Quando há TOC:
    - (i) Hiperligações dos parágrafos antes do TOC,
    - (ii) Hiperligações dos parágrafos depois do TOC;
  - -Quando não há TOC: Hiperligações de todos os parágrafos do texto;
- Hiperligações da *infobox* ou tabelas *infobox-like*;
- Hiperligações que ocorrem nos apêndices ou *bottom matter*;
- Hiperligação para artigos de desambiguação ou afins.

Este trabalho leva em linha de conta os resultados de [Lamprecht et al., 2017] e de Bhuyan (2015), considerando as hiperligações da secção principal e da *infobox* como importantes para o documento. No entanto, o número elevado dessas hiperligações obriga a efetuar uma segunda pesagem. Neste sentido, considera-se relevante aquela que verifique o seguinte:

- Quando há TOC:
  - Hiperligações que ocorram simultaneamente na secção principal e na *infobox* ou na lista de *main articles* ou afins.
- Sem TOC:
  - Hiperligações que ocorram simultaneamente em qualquer um dos parágrafos do texto e na *infobox*;
  - Hiperligações que ocorram simultaneamente na *infobox* e nos apêndices ou *bottom matter*.

A aplicação só extrai hiperligações internas, i.e., para artigos da Wikipedia, identificados por https://en.wikipedia.org/wiki/. No entanto, são excluídos os artigos fora do contexto do artigo dado, i.e., hiperligações internas para artigos sobre o funcionamento da Wikipedia (ou similares). Mais precisamente, as pesquisadas serão do tipo https://en.wikipedia.org/wiki/+extensão do artigo, sendo que a extensão do artigo não poderá conter, por exemplo, o seguinte: “Wikipedia:Citation_needed", “Wikipedia:LIBRARY", “Talk:", “Help:", “File:", “Template".

3.2 Descrição da aplicação

Hiperligações entre artigos da Wikipedia pressupõem intersecção dos contextos dos artigos. A aplicação evidencia a conexão entre artigos da Wikipedia por meio de um grafo de conhecimento. O grafo deverá obedecer ao seguinte:

- Os nodos serão associados univocamente a artigos da Wikipedia;
- Os nodos deverão ter rótulos que são as hiperligações para os artigos da Wikipedia a elas associados;
- Os arcos são orientados. A existência de um arco com origem num nodo *A* e destino em um nodo *B* significa que o artigo associado ao nodo *B* é referido no artigo associado ao nodo *A;*
- O grafo pode ter multi-arcos de multiplicidade máxima 2. Assim, pode haver no máximo dois arcos entre dois nodos *A* e *B*, sendo que necessariamente um arco será orientado de *A* para *B* e o outro de *B* para *A;*
- Os arcos são coloridos. Todos os arcos com origem no mesmo nodo devem ter a mesma cor. São atribuídas cores diferentes a arcos provenientes de nodos diferentes.

Na essência pretende-se ainda que a aplicação proporcione o seguinte:

- Um grafo em que os nodos e os arcos sejam facilmente identificados;
- Um grafo com nodos relevantes no sentido da Secção 3.1;
- Um grafo em que as hiperligações sejam unicamente para artigos da Wikipedia no contexto do artigo, havendo assim que excluir hiperligações para páginas de ajuda da Wikipedia, Wikipedias noutras línguas, e outras hiperligações (internas) fora do contexto;

— Grafos com nodos de geração aleatória; o grafo obtido não será sempre o mesmo para múltiplas execuções do mesmo comando.

O grafo será visualizado no navegador. Para que o grafo seja manejável, a aplicação dependerá dos seguintes parâmetros iniciais:

— URL do artigo inicial que corresponde à raiz do grafo;
— Número de iterações *I*, sendo que cada iteração implica expansão de nodos;
— Número de hiperligações a extrair na iteração 0 *n*, i.e., hiperligações retiradas do artigo inicial;
— Número de hiperligações efetivamente extraídas será o mínimo entre *n* e o número de hiperligações relevantes;
— Número das hiperligações *k* a deduzir nas extrações seguintes; espera-se que a atenção do utilizador esteja centrada no nodo inicial pelo que se a iteração 0 extrair *n* hiperligações, considera-se plausível que a iteração $0 \leq j < I$ extraia $\max(1, n - j * k)$ hiperligações.

3.3 Arquitetura

O programa principal do protótipo é codificado em Python 3.0 [“Python 3.0”, 2008] por ser uma linguagem adequada à criação de aplicações web. Mais, a biblioteca Beautiful Soup [https://www.crummy.com/software/BeautifulSoup/bs4/doc/] é das ferramentas mais rápidas e eficientes para *web scraping* [Vargiu & Irru, 2013].

Para a visualização optou-se pela biblioteca em JavaScript D3.js [https://d3js.org/]. Além de ser uma ferramenta vantajosa pela facilidade de escrita do código e permitir visualizar objetos com muitas e variadas características, há muitos algoritmos já desenvolvidos que podem ser facilmente adaptados. Em particular, o grafo *force directed* [Bostock, 2017] é uma *graph tool* muito popular na web. Aqui, adota-se a versão de Bellamy-Royds cujo grafo tem nodos com hiperligações. Para além da biblioteca BeautifulSoup para percorrer os ficheiros HTML dos artigos da Wikipedia, o programa usa os seguintes módulos importados: *itertools* [“9.7. itertools”, 2018], *re* [”6.2. re”, 2018], *webbrowser* [“20.1. webbrowser”, 2018], *requests* [“Requests: HTTP for Humans”, 2018], *sys* [“28.1. sys”, 2018] *e random* [“random”, 2018].

Funções definidas

Recorda-se que o objetivo do programa é extrair as hiperligações relevantes dos artigos no sentido da secção 3.1, escrever os dados para a construção do grafo num ficheiro, seja bd.js, e abrir o navegador para a visualização do grafo. O programa define as seguintes funções para extrair parágrafos dos artigos da Wikipédia:

— Parágrafos da secção principal (antes do TOC): *parasLead();*
— Todos os parágrafos não vazios: *parasTotal()*.

Estas funções recebem como argumento um objeto BeautifulSoup, **wikiSoup**. A primeira localiza a tabela de conteúdos (TOC) nos ficheiros HTML, identificado por **id=toc** na *tag* ** **. De seguida, extrai cada um dos parágrafos acima do TOC para uma lista (que de

seguida é invertida). Os parágrafos são adicionados à lista pela ordem inversa (do TOC para o topo do texto) pelo que depois há que inverter a lista. A função *parasTotal()*, por sua vez, procura a *tag* **<p>** no ficheiro HTML e adiciona todos os parágrafos do texto para uma lista, excluindo os parágrafos vazios. As imagens 5 e 6 mostram trechos de código fonte implementados para estas duas funções.

```
#funcao para obter os paragrafos da seccao principal
def parasLead(wikiSoup):
    listParaLead = []
    #a seccao principal ocorre antes de 'Table of Contents' (TOC)
    #localiza TOC
    TOC = wikiSoup.find('div', id='toc')
    if TOC is not None:
        for text in TOC.previous_siblings:
            if text.name == 'p':
                listParaLead.append(text)
    #inverte a lista
    listParaLead.reverse()
    return listParaLead
```

Figura 5. Função *parasLead().*

```
#sem TOC: funcao para extrair todos os paragrafos do texto
def parasTotal(wikiSoup):
    listaParagrafos = []
    texto = wikiSoup.find_all('p')
    for para in texto:
        listaParagrafos.append(para)
    vazios =[]
    #procura paragrafos vazios
    texto2 = wikiSoup.find_all('p',attrs={'class':'mw-empty-elt'})
    for para2 in texto2:
        vazios.append(para2)
    for para3 in listaParagrafos:
        if para3 in vazios:
            listaParagrafos.remove(para3)
    return listaParagrafos
```

Figura 6. Função *parasTotal*().

O programa define a função *linksPara()* para obter as hiperligações de um parágrafo. A função é usada pelas funções *linksLead()* (hiperligações da *lead section*) e *linksTodos()* (hiperligações de todos os parágrafos) para extrair as hiperligações da secção principal ou de todo o texto para listas. A figura 7 ilustra a função *linksLead*().

```
#com TOC: funcao para extrair links dos paragrafos da seccao
#principal
def linksLead(url,wikiSoup):
    lead = []
    if len(parasLead(wikiSoup))!=0:
        for  para in parasLead(wikiSoup):
            lead += removeDup(linksPara(url,para))
    return lead
```

Figura 7. Função *linksLead*().

Na função acima a função *removeDup()* remove entradas duplicadas de listas.

As seguintes funções (semelhantes à função *linksPara()*) criam as listas de hiperligações das restantes zonas dos artigos:

- *linksMain*() (hiperligações marcadas como *main articles* ou afins);
- *linksInfobox*() (hiperligações da *infobox* ou *infobox-like tables*);
- *linksFurther*() (hiperligações em apêndices ou *bottom matter*) de artigos sem TOC.

Todas as funções recebem como argumento um texto (o URL). Primeiro, procuram as *tags* e atributos para encontrar cada uma das partes do artigo na wikiSoup. Depois, usam *regex matching* [Aho & Ullman,1994, Capítulo 10] para procurar as hiperligações para artigos internos. O processo de *regex matching* exclui ainda as hiperligações para os artigos da Wikipedia (mais comuns) sobre a própria Wikipedia e não no contexto do artigo. As hiperligações selecionadas são por fim adicionadas a listas.

São depois definidas as seguintes funções:

- *relevancia()* para obter a lista (figura 8) de links relevantes de acordo com a secção 3.1; no final a lista é baralhada para que as primeiras ocorrências não sejam privilegiadas para ocorrer no grafo;
- *baseDados()* para obter os nodos e os links do grafo e devolver o resultado para o ficheiro bd.js. A função recebe os seguintes argumentos:
  - Uma cadeia de caracteres (o URL);
  - O número de hiperligações n a extrair na primeira iteração; esse número será depois atualizado para $n \rightarrow \min(n, l)$, onde l designa o número de hiperligações relevantes;
  - O número de hiperligações k a deduzir nas iterações seguintes;
  - O número de iterações I.

```
#funcao para seleccionar links relevantes
#Com TOC: extrai os links que aparecem na seccao principal e
#noutras zonas do
#artigo
#Sem TOC: extrai os links que aparecem nos paragrafos do texto
#e noutras zonas
#do artigo
def relevancia(url):
    rel = []
   #define link completo
   fullLink = "https://en.wikipedia.org/" + url

   #usa requests module para abrir a pagina web
   page_http_response = requests.get(fullLink, verify=False)

   #verifica conexao
   if page_http_response.status_code != 200:
      print("\nErro de conexao ou o artigo nao existe\n")
      sys.exit()
      return None
  #usa Beautiful Soup para percorrer a pagina
```

```
wikiSoup= BeautifulSoup(page_http_response.text, "lxml")

#Com TOC
if  len(parasLead(wikiSoup))!=0:
   for  item in linksLead(url,wikiSoup):
      if item in linksInfobox(url,wikiSoup) \
      or item in linksMain(url,wikiSoup):
         rel.append(item)
   if len(rel)<6:
      rel+=linksLead(url,wikiSoup)

#Sem TOC
if  len(parasLead(wikiSoup))==0:
   #procura matchings entre links dos paragrafos e da infobox
   for  item2 in linksTodos(url,wikiSoup):
      if item2 in linksInfobox(url,wikiSoup):
         rel.append(item2)

   #procura matchings entre linksInfobox e linksFurther
   if  len(linksInfobox(url,wikiSoup))!=0:
      for  item3 in linksInfobox(url,wikiSoup):
         if item3 in linksFurther(url,wikiSoup):
            rel.append(item3)
   if len(rel)<6:
      rel+=linksTodos(url,wikiSoup)
   if len(rel)<6:
     for indexI, infLink in enumerate(linksInfobox(url,wikiSoup)):
         if indexI < 5 :
            rel.append(infLink)
#baralha a lista
baralha= sample(rel, len(rel))
relevantes=removeDup(baralha)
return relevantes
```

Figura 8. Função *relevancia()*.

A função *baseDados*() usa as seguintes funções cujos detalhes são aqui omitidos dada a sua simplicidade:

- *linksRel()* para devolver o número de links relevantes de um artigo;
- *retrieveNlinks()* para obter apenas um número dado de links da lista das hiperligações relevantes;
- *indexar()* para atribuir a cada elemento (cadeia de caracteres) de uma lista o valor (inteiro) da entrada que lhe corresponde.

O *main program* aceita os mesmos argumentos da função *baseDados()*, confere os argumentos do programa, chama a função *baseDados()* e abre o navegador para a visualização do grafo por meio do comando **webbrowser.open().**

Alterações efectuadas ao código de Bellamy-Royds

Além do URL para direcionamento das hiperligações, as alterações efetuadas ao ficheiro index.js de Bellamy-Royds são as seguintes:

- A base de dados do grafo é retirada do ficheiro index.js e importada para o ficheiro bd.js criado pelo programa em Python;
- São aumentados os valores de *var svg* (*viewBox*);
- Os arcos e as esferas para os nodos também são ligeiramente ampliados;
- A gravidade (parâmetro de *var force*) é diminuída para contrariar a sobreposição de nodos;
- Os valores RGB das cores dos arcos são redefinidos para (255*(*a* mod 2), 255*(*a* mod 3)/3, 255*(*a* mod 9)/10), onde para cada arco indexado *a* é o valor da entrada do nodo-origem na lista de nodos final; a redefinição das cores dos arcos permite atribuir aos arcos cores mais diferenciadas (com valores RGB mais distantes uns dos outros); os valores são escolhidos de modo a evitar tonalidades próximas do branco que é a cor de fundo do ecrã.

3.4 Testes e resultados

A aplicação inclui funcionalidades que permitem: (i) extrair todos as hiperligações internas dos artigos (i.e., para artigos da Wikipedia no contexto) e reportar o seu número; (ii) criar múltiplos e ficheiros texto para os quais são reportados todos os passos do programa.

O protótipo foi testado em aproximadamente 50 artigos da Wikipedia, com e sem TOC, *infobox*, e *main articles*, e com muitas ou poucas hiperligações. Em todos eles a aplicação cumpriu as funcionalidades propostas, nomeadamente, a extração das hiperligações relevantes, remoção das externas ou fora do contexto, criação da base de dados para o grafo no ficheiro bd.js e visualização do grafo no navegador.

A coincidência entre listas constantes do algoritmo do programa, permite um equilíbrio entre as hiperligações em termos da sua generalidade. Por outras palavras, para cada nodo do grafo, todos as hiperligações relevantes são equiprováveis para ocorrer neste, pelo que os nodos adjacentes tanto podem corresponder a hiperligações de carácter geral ou particular. Por exemplo, para o artigo sobre Portugal, *Republic* pode considerar-se uma hiperligação geral enquanto a hiperligação *Lisbon* é específica.

A figura 9 mostra um resultado visual, isto é, para o grafo do conhecimento, tendo como base um artigo sobre Portugal (https://en.wikipedia.org/wiki/Portugal).

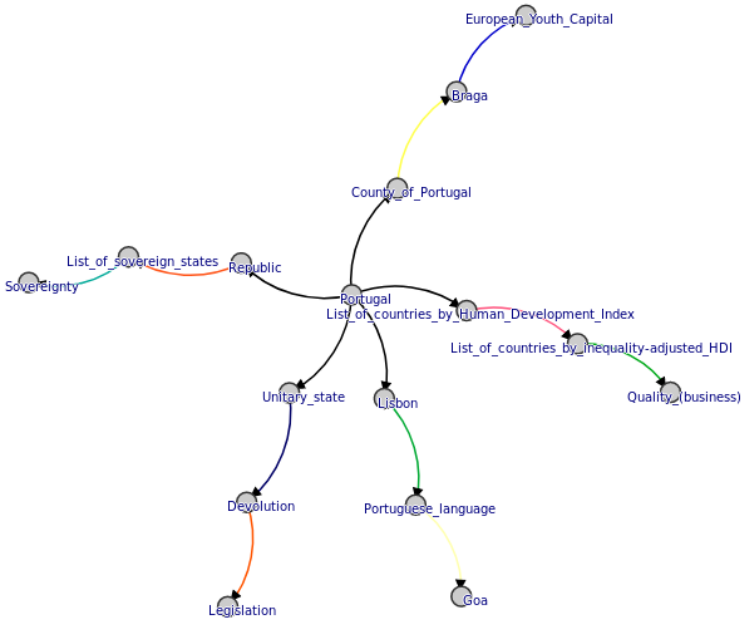

Figura 9. Exemplo de resultado da implementação para o artigo sobre Portugal. (https://en.wikipedia.org/wiki/Portugal).

4. Conclusões e trabalho futuro

Tal como referido anteriormente, o protótipo foi testado em aproximadamente 50 artigos da Wikipedia, sem qualquer ligação entre eles: artigos de várias categorias, com e sem tabela de conteúdos, com e sem *infobox*, com e sem i*nfobox-like tables*, artigos com um número muito elevado de hiperligações, pequenos artigos com poucas hiperligações, artigos sem hiperligações, etc. Em todos os artigos testados, a aplicação cumpriu as funcionalidades propostas. O grafo pareceu sempre bastante apropriado relativamente à relevância das hiperligações do ponto de vista do utilizador não especialista.

O critério para a relevância das hiperligações e coincidências entre listas, pode contudo implicar muito tempo de execução do programa. Nos grafos visualizados, os nodos e os arcos distinguem-se facilmente, bem como os arcos emergentes de cada nodo. As cores dos arcos são bastante variadas. As hiperligações direcionam o utilizador corretamente para o artigo da Wikipedia. Diferentes execuções do mesmo comando geram grafos diferentes.
A geração das hiperligações relevantes aleatoriamente revelou-se uma funcionalidade interessante segundo a qual todas as hiperligações (relevantes) têm a mesma probabilidade de ocorrer no grafo. Hiperligações mais e menos específicas são assim balanceadas sem que seja necessário visualizar muitos nodos para ter acesso a ligações que ocorrem no final da secção principal ou da *infobox*, por exemplo.

A aplicação deu resultados satisfatórios nos exemplos analisados. O critério de seleção das hiperligações reduziu em muito o número a considerar para a geração do grafo. A navegação na Wikipedia através da aplicação desenvolvida permite ao utilizador inteirar-se em poucos segundos do contexto do artigo dado e de artigos neles referidos.

Para além do desenvolvimento de opções ao nível da visualização, e.g., inclusão de *zoom* e eventos com o rato (p. ex. ao passar por cima), um trabalho futuro inclui a visualização de gráficos hierárquicos com a técnica visual *Sunburst* [Rodden, 2014]. Ao considerar a visualização *Sunburst*, o artigo inicial, seja *A*, ocupa o anel central (nível 0), enquanto os artigos nele referidos (diretamente ou em cadeia) ocupam os anéis exteriores. Uma opção seria definir os níveis hierárquicos do gráfico, em função da distância mínima dos artigos ao artigo inicial, de modo que o nível $i \geq 1$ seja ocupado por artigos à distância *i* de *A*. Assim, um artigo *B* referido no artigo *A* estaria à distância 1 do artigo A, enquanto um artigo *C* referido por *B* mas não por A estaria à distância 2 de *A*, e assim sucessivamente.

Naturalmente, que em muitos casos haverá mais do que um caminho entre o artigo *A* e os artigos dos níveis exteriores. Isso sugere que as divisões *a* de cada nível *i* sejam ocupadas por artigos para os quais há *c(a)* caminhos à distância *i* entre eles e o artigo *A*. O número de caminhos proporciona assim um critério de relevância de hiperligações. A navegação na Wikipedia por meio de uma visualização *Sunburst* é possível incluindo hiperligações em cada divisão *a* de cada nível *i*, que direcionem para a lista de hiperligações para artigos para os quais há *c(a)* caminhos à distância *i* do artigo *A*.

Referências

A. Aho e J. Ullman, *Foundations of Computer Science,* http://infolab.stanford.edu/~ullman/focs.html (1994) [9 de Novembro de 2018].

A. Bellamy Royds, D3 Force-directed Node-link diagram with hyperlinks, https://codepen.io/AmeliaBR/pen/AoFHg (2018) [9 de Novembro de 2018].

S. N. Bhuyan, Visualizing Wikipedia as a Graph, http://courses.ischool.berkeley.edu/i247/s15/reports/Wikipedia_Bhuyan.pdf (2015) [9 de Novembro de 2018].

M. Bostock, Force-Directed Graph - blo.cks.org, https://bl.ocks.org/mbostock/4062045 (2017) [9 de Novembro de 2018].

C. Bruce, X. Gao, P. Andreae e S. Jabeen, Query expansion powered by Wikipedia hyperlinks. AI 2012: *Advances in Artificial Intelligence*, 421–432, Lecture Notes in Comput. Sci., 7691, Lecture Notes in Artificial Intelligence, Springer, Heidelberg, 2012.

O. Cornec, WikyGalaxy: Explore Wikipedia in 3D, http://wiki.polyfra.me/# (2015) [9 de Novembro de 2018].

O. Cornec, WikiGalaxy by Owen Cornec | Experiments with Google, https://experiments.withgoogle.com/wikigalaxy (2014) [9 de Novembro de 2018].

D. P. Fernandes, David e E. S. Carvalho, DocGenealogy–Visualizing the doctoral advisors and mentors genealogic tree, *Abakós* 6.2 (2018) 3–20.

B. Hachey, W. Radford, J. Nothman, M. Honnibal e J. R. Curran, Evaluating entity linking with Wikipedia, *Artificial Intelligence* 194 (2013) 130–150.

A. Kashcha, yasiv-youtube, https://yasiv.com/youtube (2017) [9 de Novembro de 2018].

List of Wikipedias, https://en.wikipedia.org/wiki/Wikipedia:List_of_Wikipedias (2018) [9 de Novembro de 2018].

D. Lamprecht, K. Lerman, D. Helic e M. Strohmaier, How the structure of Wikipedia articles influences user navigation, *New Rev. Hypermedia M.* (2017) 29–50.

P. Malo, P. Siitari, e A. Sinha, Automated query learning with Wikipedia and genetic programming, *Artificial Intelligence* 194 (2013) 86–110.

V. Nastase e M. Strube, Transforming Wikipedia into a large scale multilingual concept network, *Artificial Intelligence* 194 (2013) 62–85.

9.7. itertools – Functions creating iterators for efficient looping, https://docs.python.org/2/library/itertools.html (2018) [9 de Novembro de 2018].

J. Nothman, N. Ringland, W. Radford, T. Murphy e J. R. Curran, Learning multilingual named entity recognition from Wikipedia, *Artificial Intelligence* 194 (2013) 151–175.

random - Generate pseudo-random numbers, https://docs.python.org/3/library/random.html (2018) [9 de Novembro de 2018].

Python 3.0 Release, https://www.python.org/download/releases/3.0/ (2008) [9 de Novembro de 2018].

K. Rodden, Applying a sunburst visualization to summarize user navigation sequences, *IEEE computer graphics and applications* 34 (2014) 36-40.

Requests: HTTP for Humans, http://docs.python-requests.org/en/master/ (2018) [9 de Novembro de 2018].

6.2. re – Regular expression operations, https://docs.python.org/3/library/re.html (2018) [9 de Novembro de 2018].

20.1. webbrowser – Convenient Web-browser controller, https://docs.python.org/2/library/

webbrowser.html (2018) [9 de Novembro de 2018].

28.1. sys – System-specific parameters and functions, https://docs.python.org/2/library/sys.html (2018) [9 de Novembro de 2018].

E. Vargiu e M. Urru, Exploiting web scraping in a collaborative filtering-based approach to web advertising, Artificial Intelligence Research 2 (2013) 44–54.

R. Warren, E. Airoldi e D. Banks, *Network analysis of wikipedia. Statistical methods in e-commerce research*, 81–102, Statist. Practice, Wiley, Hoboken, NJ, 2008.

Wikipedia: Help:Infobox, https://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style/Infoboxes (2018) [9 de Novembro de 2018].

Wikipedia: Manual of Style/Linking, http://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style/Linking (2018) [9 de Novembro de 2018].

Wikipedia:Manual of Style/Layout, https://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style/Layout (2018) [9 de Novembro de 2018].

Wikipedia: Manual of Style/Lead Section, http://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style/Lead_section (2018) [9 de Novembro de 2018].

Wikipedia: Size of Wikipedia, https://en.wikipedia.org/wiki/Wikipedia:Size_of_Wikipedia (2018) [9 de Novembro de 2018].

Wikipedia:Statistics, https://en.wikipedia.org/wiki/Wikipedia:Statistics (2018) [9 de Novembro de 2018].

Wikipedia:Wiki Game, https://en.wikipedia.org/wiki/Wikipedia:Wiki_Game (2018) [9 de Novembro de 2018].

T. Yasseri e J. Kertész, Value production in a collaborative environment: sociophysical studies of Wikipedia, *J. Stat. Phys.* 151 (2013) 414–439.

Ângela Mestre, Licenciada em Engenharia Informática pela Universidade Aberta em 2018. Doutorada em Física pela Universidade de Coimbra em 2008. Licenciada em Matemática e em Física também pela Universidade de Coimbra em 2012 e 1998, respetivamente. Atualmente realiza trabalho de investigação na área de Matemática no âmbito de uma bolsa de pós-doutoramento concedida pela Fundação para a Ciência e a Tecnologia.

Elizabeth Simão Carvalho é investigadora do CIAC—Centro de Investigação em Artes e Comunicação e docente no DCeT – Departamento de Ciências e Tecnologia, Universidade Aberta. Ela é doutorada em Tecnologias e Sistemas de Informação e mestre em Ciências da Computação, ambos pela Universidade do Minho, Portugal, e licenciada em Engenharia Eletrotécnica pela Universidade Veiga de Almeida, Brasil. Tem como principais interesses de investigação as áreas de visualização de informação e científica.