TOPEXTRACT - TOPONYM EXTRACTION AND DISAMBIGUATION TOOL

LUCAS SIEGL CORREA MACHADO, CELINA MAKI TAKEMURA, AND MARIA FERNANDA MOURA

Abstract. Componente de software para extração e desambiguação de topônimos.

1. Introdução

A Rede AgroHidro tem como objetivo analisar/avaliar os impactos da atividade agrícola e das mudanças climáticas sobre os recursos hídricos de diferentes ecorregiões brasileiras. Para subsidiar ações de investigação e disseminação do conhecimento na rede e organizar sua produção técnico-científica, permitindo o cruzamento dessas informações com outras fontes diversas, se faz necessária a identificação e desambiguação de topônimos e indexação dos documentos textuais pelos mesmos. Particularmente importante nesse cenário, a desambiguação de topônimos representa a ponte entre o mundo do processamento de linguagem natural e os sistemas de informações geográficos utilizados pela Rede.

Como exemplo da amplitude do problema, fazendo uma análise utilizando o banco de dados da Agência Nacional de Águas - ANA [1] que contém a Rede Hidrográfica Brasileira e a Malha Municipal Brasileira do IBGE [2] vemos que 23,16 % dos topônimos referentes a corpos d'água podem ser associados a mais de um lugar - no caso de topônimos referentes a municípios o número entidades ambíguas é 4,6%. A questão da sinonímia em topônimos ainda pode aparecer na alteração de nomes no tempo (e.g. Vila de São Carlos(1797) / Campinas(1842)).

Neste trabalho foi desenvolvido um componente de software com a finalidade de oferecer uma solução que identifique topônimos e resolva casos de desambiguação em uma coleção de textos. O componente recebe como entrada uma coleção de textos, nesta 1ªversão em inglês, encontra os topônimos e desambigua-os, conforme discutido nos próximos itens. O componente foi desenvolvido durante o estágio na Embrapa Informática Agropecuária no período de 04/06/2013 a 10/12/2013 e encontra-se disponível em [3].

2. Metodologia

Dado um documento textual, a metodologia prevê o:

(1) Reconhecimento de entidades nomeadas e a inferência sobre relação das mesmas a acidentes geográficos;
(2) Uso de um gazetteer, i.e., índice de topônimos;
(3) Processo de desambiguação baseado em distâncias;
(4) Geração de metadados geográficos.

Data:. 10 de Dezembro de 2013.

RA 081990 - Matemática Aplicada e Computacional UNICAMP
E-mail: lucas.scmachado@gmail.com

Embrapa Monitoramento por Satélite
E-mail: celina.takemura@embrapa.br

Embrapa Informática Agropecuária
E-mail: maria-fernanda.moura@embrapa.br

A metodologia segue o fluxo ilustrado na Figura 1

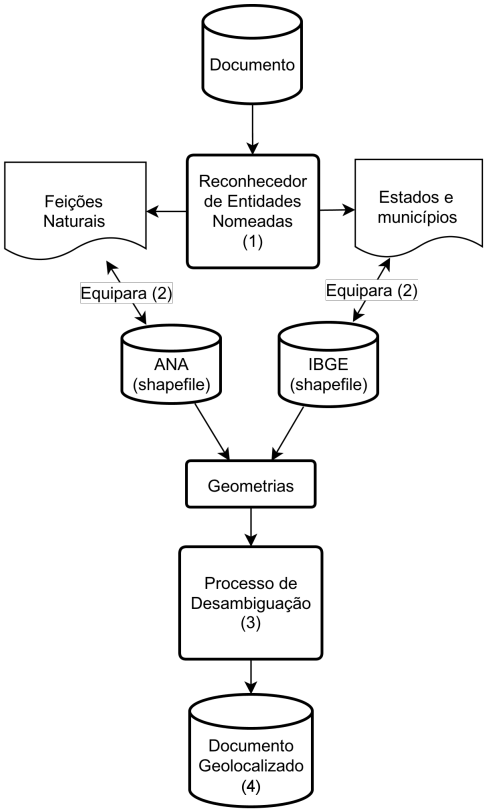

FIGURA 1. Processamento completo de um documento.

2.1. **Reconhecimento de entidades nomeadas.** Existe uma vasta e heterogênea gama de estratégias para identificação automática de entidades nomeadas em textos e é dependente de uma gama de fatores como língua, gênero textual e domínio do conhecimento. Neste componente de software utilizamos o OpenCalais [4], para a língua inglesa.

2.2. **Índice de topônimos.** Neste trabalho, o repertório toponímico foi criado a partir de bases cartográficas. Para os testes detalhados na Seção Experimentos e Resultados 4, utilizou-se documentos em formato Shapefile contendo a Malha municipal brasileira do IBGE [2] e a Rede hidrográfica brasileira ANA [1], no entanto, o processo é extensível a outros documentos em formato Shapefile.

2.3. **Processo de desambiguação.** O processo de desambiguação das entidades nomeadas inicia-se com o cálculo da similaridade de cada entidade com cada item do índice de topônimos. Para o experimento, utilizamos a distância Jaro-Winkler [5] como medida de similaridade. A métrica de distância entre palavras Jaro d_j (2.1) corresponde a soma ponderada do percentual de caracteres correpondentes e transpostos entre duas palavras.

$$d_j = \begin{cases} 0 & \text{if } m = 0 \\ \frac{1}{3}\left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m}\right) & \text{else} \end{cases} \tag{2.1}$$

onde: m é o número de coincidências entre caracteres; $|s_1|$ e $|s_2|$ são os tamanhos das palavras s_1 e s_2; t é o número de transposições;

O método Jaro-Winkler d_w (2.2) é uma variação feita para obter melhores resultados com palavras pequenas, como nomes de pessoas ou cidades. Este método utiliza dois parâmetros de ajuste, sendo p responsável por determinar a quantidade de caracteres comum ao prefixo das palavras um valor maior. A métrica é dada por:

$$d_w = d_j + (\ell p(1 - d_j)) \tag{2.2}$$

onde: d_j é a distância Jaro entre as palavras s_1 e s_2; ℓ é a quantidade de caracteres comum no prefixo de ambas, $\ell \in [0, 4]$ p é o fator de escala relacionado a ℓ, $p \in (0, 0.25]$

Selecionados os topônimos com maior similaridade, caso haja empate, iniciamos o processo de desambiguação, considerando a distância entre dois conjuntos de pontos X e Y , que representam "geograficamente"os topônimos como a distância entre os pontos mais próximos de X e Y:

$$d_N(X, Y) = \min_{(x,y)\in X\times Y} d_E(x, y) = \min_{x\in X} d_E(x, Y) = \min_{y\in Y} d_E(y, X), \tag{2.3}$$

onde d_E denota a distância euclidiana em $\mathcal{S}$.

Sejam $E = \{e_i\},\ i = [1, n]$ o conjunto das entidades nomeadas de um documento $\mathbb{D}$ e $dist(g_1, g_2) = x,\ x \in \mathbb{R}$ a menor distância entre os pontos, tomados dois a dois, de duas geometrias. $\forall\ e_i\ \exists\ G(e_i) = \{g_{j_i}\},\ G(e_i) \neq \emptyset$, onde $G(e_i)$ é denominado Conjunto de Geometrias Ambíguas de e_i. Definimos $\bar{G}$ como Conjunto de Geometrias Desambiguado, onde

$$\bar{G} = \{(\bar{g_1}, ..., \bar{g_n})\} \mid \bar{g_1} \in G(e_1), ..., \bar{g_n} \in G(e_n) \tag{2.4}$$

$$(\bar{g_1}, ..., \bar{g_n}) \in \bar{G} \Leftrightarrow \sum_{i\neq k} dist(\bar{g_i}, \bar{g_k}) = min\left(\sum_{i\neq k} dist(g_{j_i}, g_{j_k})\right), k = [1, n] \tag{2.5}$$

2.4. **Metadados.** A geolocalização (topônimos desambiguados) de cada texto é guardada num padrão específico de metadados. Neste trabalho está sendo utilizado um perfil de metadados com base na ISO 19115:2003. A ISO 19115:2003 é um padrão de metadados para informação geográfica aplicável não só a documentos digitais mas também a documentos textuais.

No Brasil, instituída pelo Decreto N 6.666 de 27/11/2008, a Infraestrutura Nacional de Dados Espaciais INDE tem como propósito catalogar, integrar e harmonizar dados geoespaciais produzidos e mantidos por instituições do governo brasileiro, facilitando sua localização, exploração e acessibilidade para os mais diversos usos. Para tanto, a INDE provê um conjunto integrado de tecnologias; políticas; mecanismos e procedimentos de coordenação e monitoramento; padrões e acordos, necessário para facilitar e ordenar a geração, o armazenamento, o acesso, o compartilhamento, a disseminação e o uso dos dados geoespaciais de origem federal, estadual, distrital e municipal. Em especial, a INDE prevê a utilização de um subconjunto da norma ISO completa, o Perfil de Metadados Geoespaciais Brasileiros - Perfil MGB. Um perfil de metadados contém um conjunto básico e necessário de elementos que retrate as características dos produtos geoespaciais de uma determinada comunidade e garanta sua identificação, avaliação e utilização consistente. No apêndice B, no arquivo "Heriberto.xml", está um exemplo da geração do arquivo após o término do procedimento de desambiguação, utilizando este perfi.

3. Requisitos de Software e Instalação

O componente foi desenvolvido em Python, com base nas bibliotecas:

(1) GDAL (OGR e OSR): <http://trac.osgeo.org/gdal/wiki/GdalOgrInPython>
(2) NUMPY: <http://www.numpy.org/>
(3) NLTK: <http://nltk.org/>
(4) OpenCalais: <http://www.opencalais.com/>
(5) NER (pyner): <https://github.com/dat/pyner>
(6) Geopy: <http://code.google.com/p/geopy/>
(7) Google Translate: <https://github.com/terryyin/google-translate-python>
(8) PDF Miner: <http://www.unixuser.org/ > euske/python/pdfminer/index.html>
(9) Jellyfish: <https://pypi.python.org/pypi/jellyfish/0.2.0>

E os mapas utilizados para a criação do índice de topônimos:

(1) HIntegrada1.shp: Arquivo contendo a hidrografia de todos os rios brasileiros (Projeção EPSG:4618 - SAD69) [1]
(2) 55mu2500gsd.shp: Arquivo contendo a geometria de todos os municípios brasileiros (Projeção EPSG:4618 - SAD69) [2]

4. Experimento e Resultados

No apêndice A, encontra-se o arquivo "experimento.py" [3] com um exemplo de uso. Ao executá-lo, utilizam-se os dados e obteve-se os resultados discutidos nos próximos quatro itens.

4.1. **Importação das bibliotecas do componente.** Para utilização do componente, é necessário a importação das bibliotecas que fazem parte do pacote. Extrato de código:

```
import shpInfo, geomInfo, textualAnalysis, geoRec
```

4.2. **Texto de entrada.** Utilizamos como entrada as seções *Abstract*, *Introduction* e *Materials and methods* do artigo:

- TEIXEIRA, A. H. de C. ; Bastiaanssen, W.G.M. ; Ahmad, M ud D ; Bos, M. G. .Reviewing SEBAL input parameters for assessing evapotranspiration and water productivity for the Low-Middle São Francisco River basin, Brazil Part B: Application to the regional scale Agricultural and Forest Meteorology, v. 149, p. 477-490, 2009 [1]

4.3. **Entidades nomeadas.** Nesta etapa, utilizamos o OpenCalais [4] para fazer o reconhecimento das entidades nomeadas presentes no texto. No experimento em questão, foram encontradas as seguintes entidades nomeadas:

(1) *NaturalFeatures*: Rio São Francisco
(2) *Province/State/City*: 'Idaho', 'Landsat', 'Lagoa Grande', 'Petrolina'

Extrato de código:

```
textualAnalysis.entitiesExtraction(path)
```

4.4. **Topônimos com menor distância de strings e desambiguação.** Utilizando como entrada a saída do item 4.3, damos início ao processo de desambiguação. Utilizando os bancos de dados do IBGE [2] e da ANA [1], juntamente com a métrica de distância entre palavras Jaro-Winkler (2.2), vemos que "Idaho" e "Landsat" não tiveram topônimos com similaridade maior que limiar de corte (para este experimento foi utilizado como limiar de similaridade 0.75 para Hidro Search e 0.85 para Adm Search). Para a entidade "Rio São Francisco" foram encontrados 24 topônimos, para "Petrolina" 1 e para "Lagoa Grande" 2, formando então o Conjunto de Geometrias Ambíguas representado na Figura 2.

(1) Hidro Search: Rio Sao Francisco: Similarity: 1.0 (24 toponym(s))
(2) Adm Search: Idaho not found - Distance < Threshold
(3) Adm Search: Landsat not found - Distance < Threshold
(4) Adm Search: Petrolina: Similarity: 1.0 (1 toponym(s))
(5) Adm Search: Lagoa Grande : Similarity: 1.0 (2 toponym(s))

[1] http://www.alice.cnptia.embrapa.br/bitstream/doc/630907/1/Heriberto.pdf

FIGURA 2. Visualização do Conjunto de Geometrias Ambíguas.

Após a criação do Conjunto de Geometrias Ambíguas, iniciamos a desambiguação das geometrias por distâncias (2.5), obtendo assim o Conjunto de Geometrias Desambiguado (2.4), representado na Figura 3.

FIGURA 3. Visualização do Conjunto de Geometrias Desambiguado.

Extrato de código:

```
# get the geometries from Natural Features
NatFeatLines, NatFeatGeos = geomInfo.getAllGeos(NatFeatFile, NatFeatCode, sad69)
# get the geometries from Province or States or Citys
PrStLines, PrStGeos = geomInfo.getAllGeos(PrStFile, PrStCode, sad69)
# group all the geometries and lines
allLines = NatFeatLines
allGeometries = NatFeatGeos
for x, line in enumerate(PrStLines):
allLines.append(PrStLines[x])
allGeometries.append(PrStGeos[x])
# find the geometries with the minimal sum of distance
choosenLines, choosenGeos = geomInfo.minDistance(allGeometries, allLines)
```

5. Conclusões

O componente gerado implementa a metodologia de desambiguação de topônimos por distância de objetos geográficos, que cumpre seu objetivo de geoespacializar documentos textuais.

O componente desenvolvido é flexível e pode ser extendido ou modificado afim de utilizar outro reconhecedor de entidades nomeadas, outras bases de dados para efetuar as comparações ou até mesmo a metodologia de desambiguação. O foco dos trabalhos futuros são:
(a) Otimizar o reconhecimento de entidades nomeadas;
(b) Adoção de padrões de metadados geoespaciais;
(c) Teste de métodos alternativos de desambiguação.

Bibliografia

[1] **[ANA, 2013]** ANA - Agência Nacional de Águas. *Rede hidrográfica brasileira.*
Disponível em: <http://www.ana.gov.br/bibliotecavirtual/redeHidrografica.asp>.
Acesso em: 09/09/2013.

[2] **[IBGE, 2007]** IBGE - Instituto Brasileiro de Geografia e Estatística. *Malha municipal brasileira.*
Disponível em: <ftp://geoftp.ibge.gov.br/malhas_digitais/municipio_2007/escala_2500mil/proj_geografica_sad69/brasil/55mu2500gsd.zip>.
Acesso em: 09/09/2013

[3] **[Takemura, Moura, Machado, 2013]** TAKEMURA, C. M.; MOURA, M. F.; MACHADO, L. S. C. *TopExtract - Toponym Extraction and Disambiguation Tool: Componente de software para extração e desambiguação de topônimos.* Campinas: Embrapa Monitoramento por Satélite, 2013. 1 CD-ROM
Disponível em: <http://ainfo.cnptia.embrapa.br/digital/bitstream/item/92648/1/manual.pdf>.
Acesso em: 05/12/2013.

[4] **[Thompson Reuters, 2013]** Thompson Reuters *OpenCalais.*
Disponível em: <http://www.opencalais.com/>.
Acesso em: 09/09/2013.

[5] **[Winkler, 1990]** Winkler, W. E. (1990). *"String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage"* Proceedings of the Section on Survey Research Methods (American Statistical Association): 354-359.

Apêndice A. experimento.py

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
import sys
reload(sys)
sys.setdefaultencoding("utf-8")

import shpInfo, geomInfo, textualAnalysis
import numpy as np
import math
import os, ogr, osr

# CONSTANTES
munCSV = 'initialFiles/sorted55mu2500gsd.csv'
munSHP = 'initialFiles/55mu2500gsd.shp'

riversCSV = 'initialFiles/sortedHIntegrada1.csv'
riversSHP = 'initialFiles/HIntegrada1.shp'

path = os.path.abspath('./experimento/')

print '\nInitializing Analysis'

# inicial set of parameters
filename = 'Heriberto.pdf'
filename = filename.split('.')
filename = filename[0]

filelist = [ f for f in os.listdir(path) if f.endswith(".txt") ]
for f in filelist:
    os.remove(path+'/'+f)

textualAnalysis.convert_pdf_all(path)
textualAnalysis.entitiesExtraction(path)

sad69 = 1
NatFeatLines, NatFeatGeos, PrStLines, PrStGeos = [],[],[],[]
NatFeatCode, PrStCode = 'COBACIA', 'GEOCODIG_M'

NatFeatFile = path + '/naturalfeatures.'+ filename
PrStFile = path + '/provinceorstate.' + filename
print 'NatFeatFile', NatFeatFile

# get the geometries from Natural Features
```

```
NatFeatLines, NatFeatGeos = geomInfo.getAllGeos(NatFeatFile, NatFeatCode, sad69)

# get the geometries from Province or States or Citys
PrStLines, PrStGeos = geomInfo.getAllGeos(PrStFile, PrStCode, sad69)

# set the shapefile name
if not os.path.exists(path + '/result/'):
    os.makedirs(path + '/result/')

filename = path + '/result/' + filename

# create all geometries shapefiles
shpInfo.createShapefile(filename+'all.P', PrStCode, PrStGeos, PrStLines, sad69)
print filename+'.allP.shp created successfully'
shpInfo.createShapefile(filename+'all.R', NatFeatCode, NatFeatGeos, NatFeatLines, sad69)
print filename+'.allR.shp created successfully'

# group all the geometries and lines
allLines = NatFeatLines
allGeometries = NatFeatGeos
for x, line in enumerate(PrStLines):
        allLines.append(PrStLines[x])
        allGeometries.append(PrStGeos[x])

# find the geometries with the minimal sum of distance
choosenLines, choosenGeos = geomInfo.minDistance(allGeometries, allLines)

# create the choosen geometries envelope
bBox = geomInfo.boundingBox(choosenGeos)
print 'Bounding Box:', bBox

# get the name of the choosen geometries
geoNames = geomInfo.getGeonames(choosenLines)

# create the XML file
geomInfo.createXML(filename, bBox, geoNames)

# create choosen geometries shapefiles
shpInfo.createShapefile(filename+'.P', PrStCode, choosenGeos[1], choosenLines[1], sad69)
print filename+'.P.shp created successfully'
shpInfo.createShapefile(filename+'.R', NatFeatCode, choosenGeos[0], choosenLines[0], sad69)
print filename+'.R.shp created successfully'

# clear
NatFeatLines, NatFeatGeos = None, None
```

```
PrStLines, PrStGeos = None, None
allGeometries, allLines = None, None
choosenLines, choosenGeos = None, None

print '\n\nAnalysis complete'
```

Apêndice B. Heriberto.xml

```
<root>
  <gmd:extent xmlns:gmd="http://www.isotc211.org/2005/gmd">
    <gmd:EX_Extent>
      <gmd:geographicElement>
        <gmd:EX_GeographicBoundingBox>
          <gmd:westBoundLongitude>
            <gco:Decimal xmlns:gco="http://www.isotc211.org/2005/gco">-46.5264129639</gco:Decimal>
          </gmd:westBoundLongitude>
          <gmd:eastBoundLongitude>
            <gco:Decimal xmlns:gco="http://www.isotc211.org/2005/gco">-36.3931694031</gco:Decimal>
          </gmd:eastBoundLongitude>
          <gmd:southBoundLongitude>
            <gco:Decimal xmlns:gco="http://www.isotc211.org/2005/gco">-20.3587856293</gco:Decimal>
          </gmd:southBoundLongitude>
          <gmd:northBoundLongitude>
            <gco:Decimal xmlns:gco="http://www.isotc211.org/2005/gco">-8.51991939545</gco:Decimal>
          </gmd:northBoundLongitude>
        </gmd:EX_GeographicBoundingBox>
      </gmd:geographicElement>
    </gmd:EX_Extent>
  </gmd:extent>
  <gmd:referenceSystemInfo xmlns:gmd="http://www.isotc211.org/2005/gmd">
    <gmd:MD_ReferenceSystem>
      <gmd:referenceSystemIdentifier>
        <gmd:RS_Identifier>
          <gmd:code>
            <gco:CharacterString xmlns:gco="http://www.isotc211.org/2005/gco">4618</gco:CharacterString>
          </gmd:code>
          <gmd:codeSpace>
            <gco:CharacterString xmlns:gco="http://www.isotc211.org/2005/gco">EPSG</gco:CharacterString>
          </gmd:codeSpace>
          <gmd:version>
            <gco:CharacterString xmlns:gco="http://www.isotc211.org/2005/gco">6.11</gco:CharacterString>
          </gmd:version>
        </gmd:RS_Identifier>
      </gmd:referenceSystemIdentifier>
    </gmd:MD_ReferenceSystem>
  </gmd:referenceSystemInfo>
  <gmd:identificationInfo xmlns:gmd="http://www.isotc211.org/2005/gmd">
    <gmd:descriptiveKeywords>
      <gmd:MD_Keywords>
```

```
        <gmd:keyword>
          <gco:CharacterString xmlns:gco="http://www.isotc211.org/2005/gco">
          Rio Sao Francisco, Codigo_Rio: 74_0
          </gco:CharacterString>
        </gmd:keyword>
        <gmd:keyword>
          <gco:CharacterString xmlns:gco="http://www.isotc211.org/2005/gco">
          Lagoa Grande, Codigo_Munic: 2608750
          </gco:CharacterString>
        </gmd:keyword>
        <gmd:keyword>
          <gco:CharacterString xmlns:gco="http://www.isotc211.org/2005/gco">
          Petrolina, Codigo_Munic: 2611101
          </gco:CharacterString>
        </gmd:keyword>
        <gmd:type>
          <gmd:MD_KeywordTypeCode
          codeList="http://www.isotc211.org/2005/resources/Codelist/gmxCodelists.xml#MD_KeywordTypeCode"
          codeSpace="ISOTC211/19115" codeListValue="place">place</gmd:MD_KeywordTypeCode>
        </gmd:type>
      </gmd:MD_Keywords>
    </gmd:descriptiveKeywords>
  </gmd:identificationInfo>
</root>
```