Classification des résultats d'un moteur de recherche

Anthony Hessab, Nirina Andriananja, Kim Leng Chhun

Plan

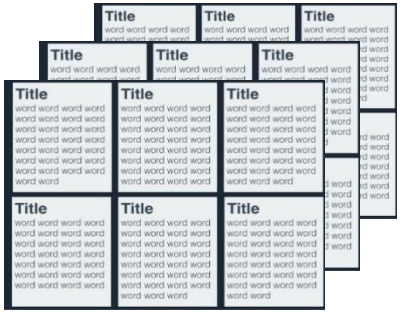
- Contexte et objectif
- Collection et traitement des données
- Modeling LDA
- Interface
- Démo
- Améliorations

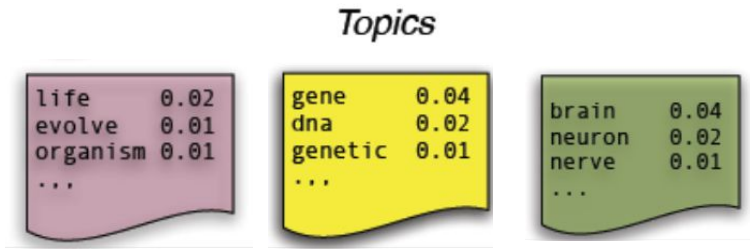
1

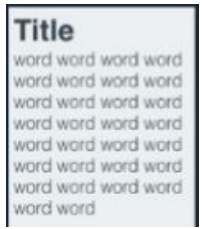
Contexte & Objectif

Contexte & objectif

Classer les résultats d'un moteur de recherche dans différents topics







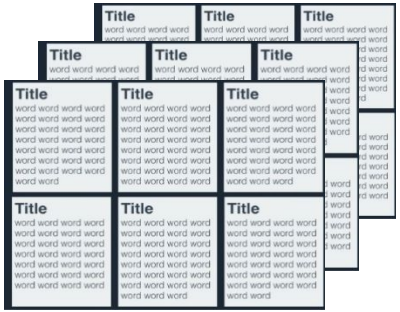
topic 0 , topic 1

Google

2

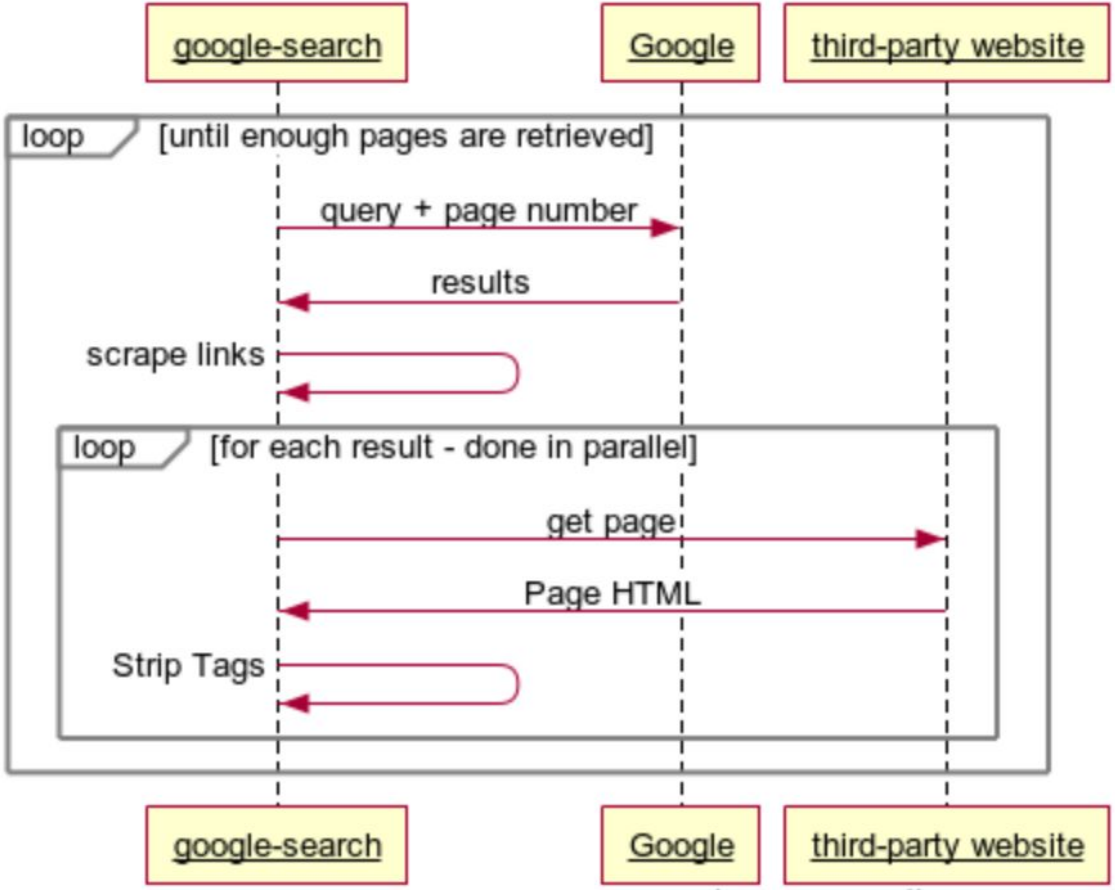
Collecte des Données

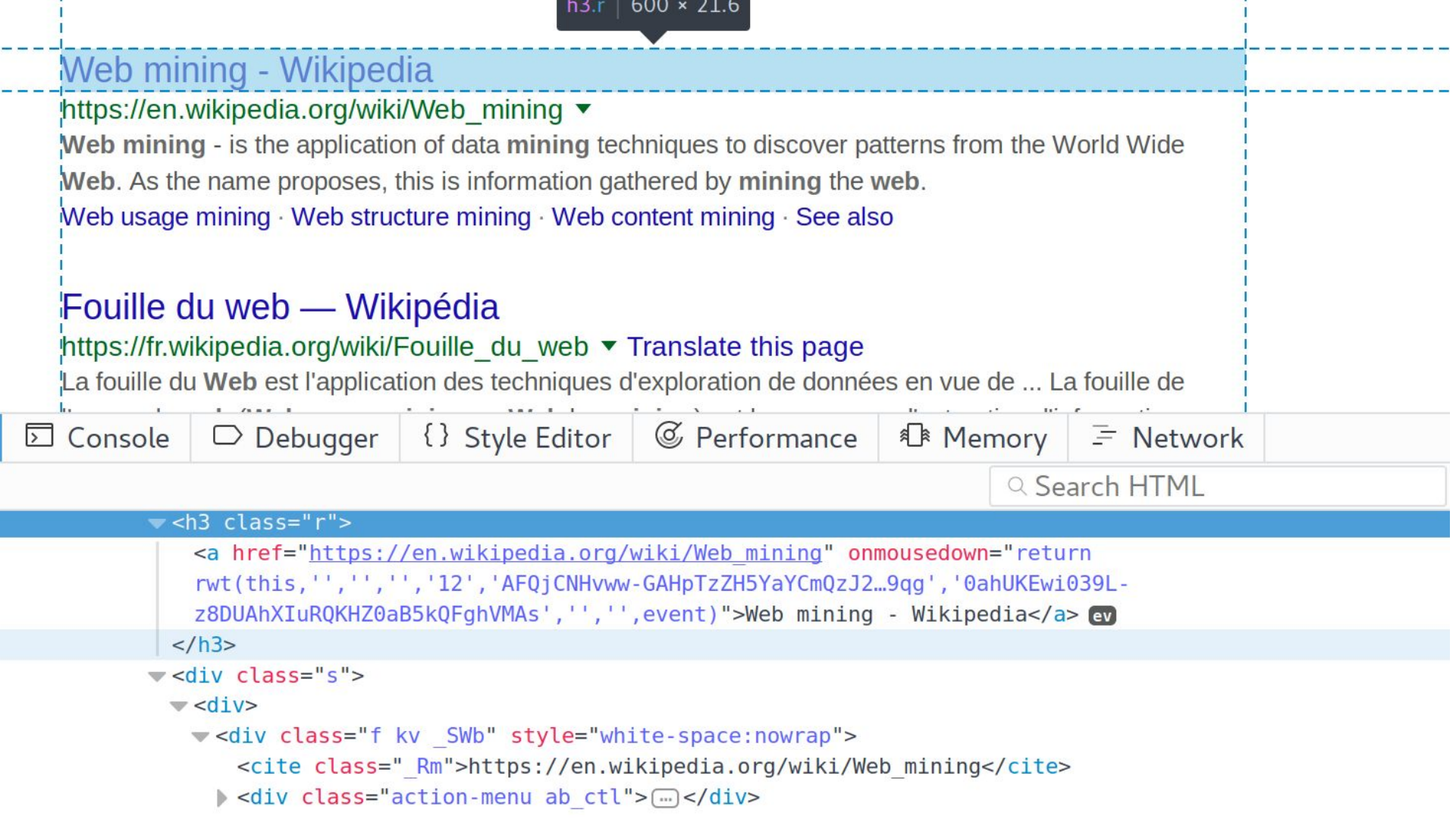
Collecte des données



GoogleSearch:
beautifulsoup – python

Collecte des données




h3.r | 600 × 21.6
Web mining - Wikipedia
https://en.wikipedia.org/wiki/Web_mining
Web mining - is the application of data mining techniques to discover patterns from the World Wide Web. As the name proposes, this is information gathered by mining the web.
Web usage mining · Web structure mining · Web content mining · See also
Fouille du web — Wikipédia
https://fr.wikipedia.org/wiki/Fouille_du_web Translate this page
La fouille du Web est l'application des techniques d'exploration de données en vue de ... La fouille de
Console
Debugger
Style Editor
Performance
Memory
Network
Search HTML
<h3 class="r">
Web mining - Wikipedia ev
</h3>

<cite class="_Rm">https://en.wikipedia.org/wiki/Web_mining</cite>
…

Collecte des données

https://www.github.com/anthonyhseb/topics

google-search

pypi v1.0.2 | build passing | docs latest | pyup 6 updates

Library for scraping google search results.

- Usage:

```
from googlesearch.googlesearch import GoogleSearch
response = GoogleSearch().search("something")
for result in response.results:
    print("Title: " + result.title)
    print("Content: " + result.getText())
```

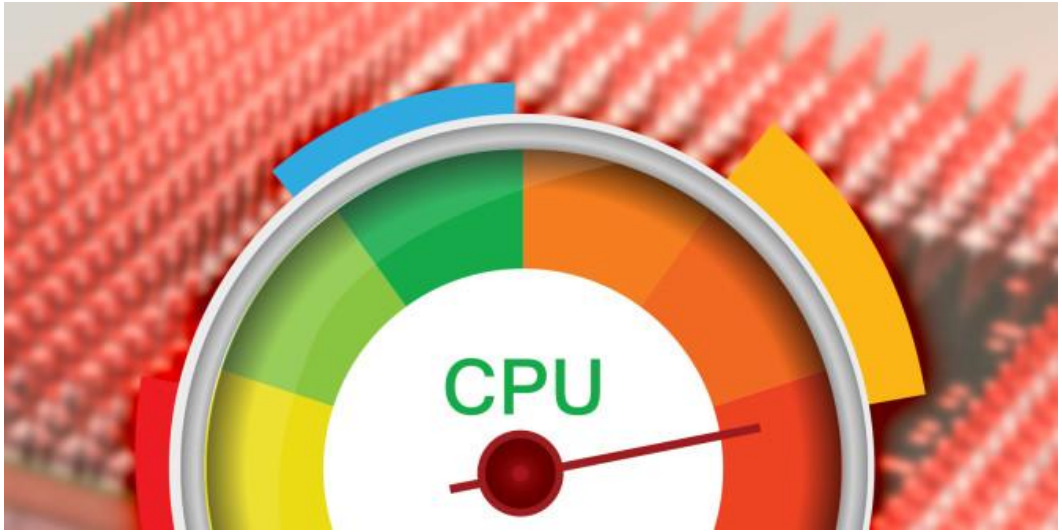
- Free software: MIT license

Collection des données - Problèmes

- BeautifulSoup est lourd sur le CPU
- Plus de temps pour analyser le DOM que la requête HTTP pour télécharger l'HTML!

- requêtes HTTP parallélisables

- Requêtes CSS et extraction du text en parallèle ne fait que ralentir les threads encore plus
- Améliorations:
  - Eviter les sélecteurs CSS (parcours manuel du DOM)



3

Traitements des données

Traitements des données

Enlever les stops words

Enlever les ponctuations

Enlever les valeurs numériques

Découper en n-gram

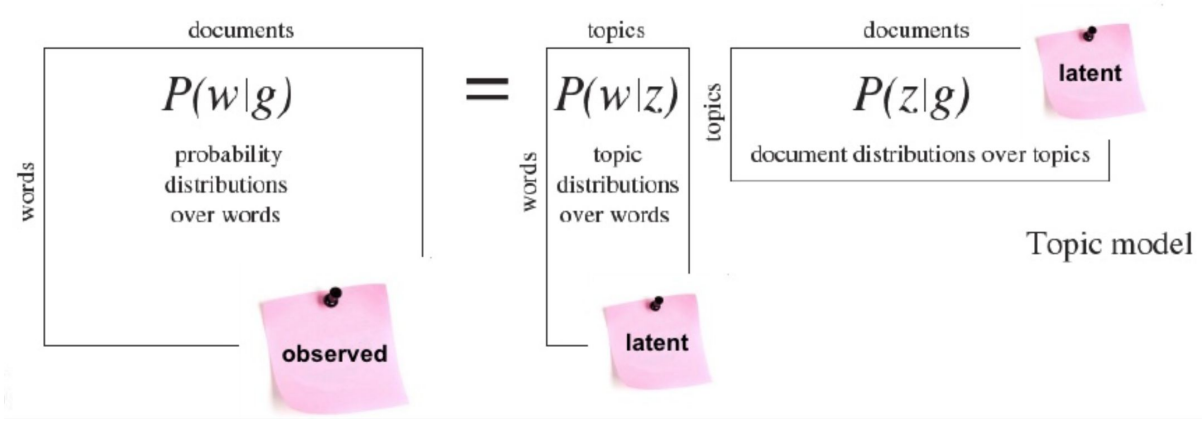
Enlever les mots qui n'apparaissent qu'une seule fois dans l'ensemble des documents

4

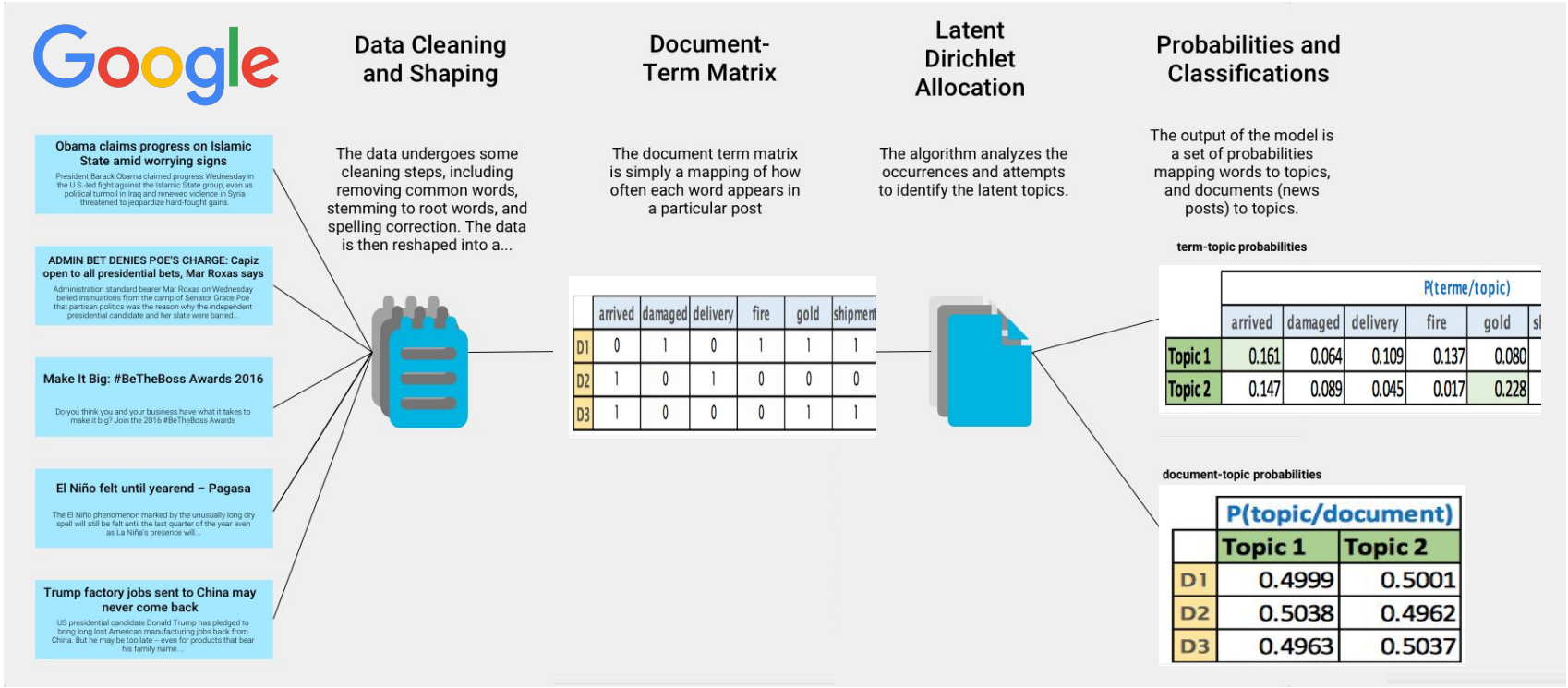
Modeling - LDA

Modeling - LDA (Latent Dirichlet Allocation)

LDA est une technique qui permet de découvrir automatiquement les abstraits thèmes(topics) cachés dans la collection des documents non structurés



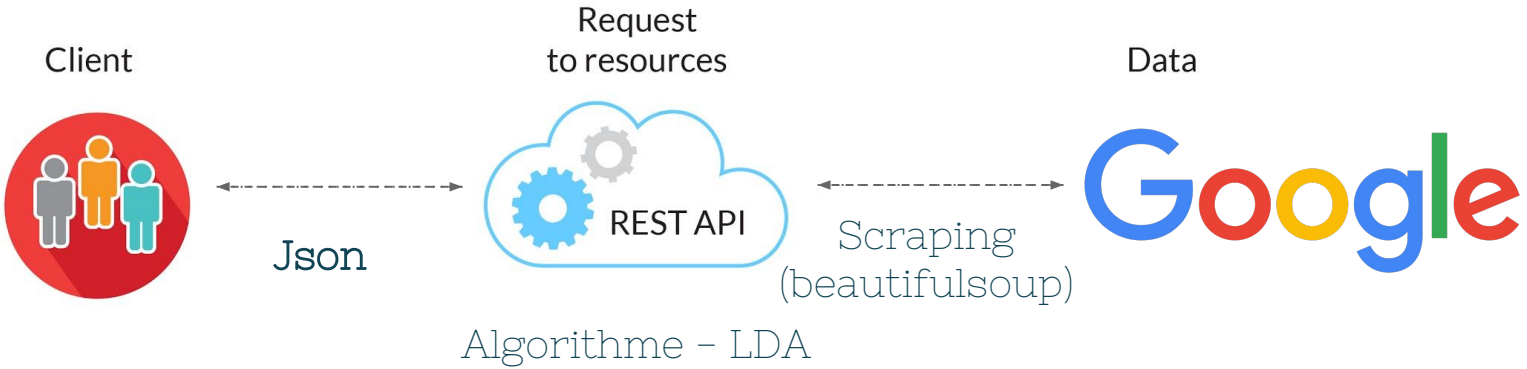
Modeling - LDA (Latent Dirichlet Allocation)



5

Interface

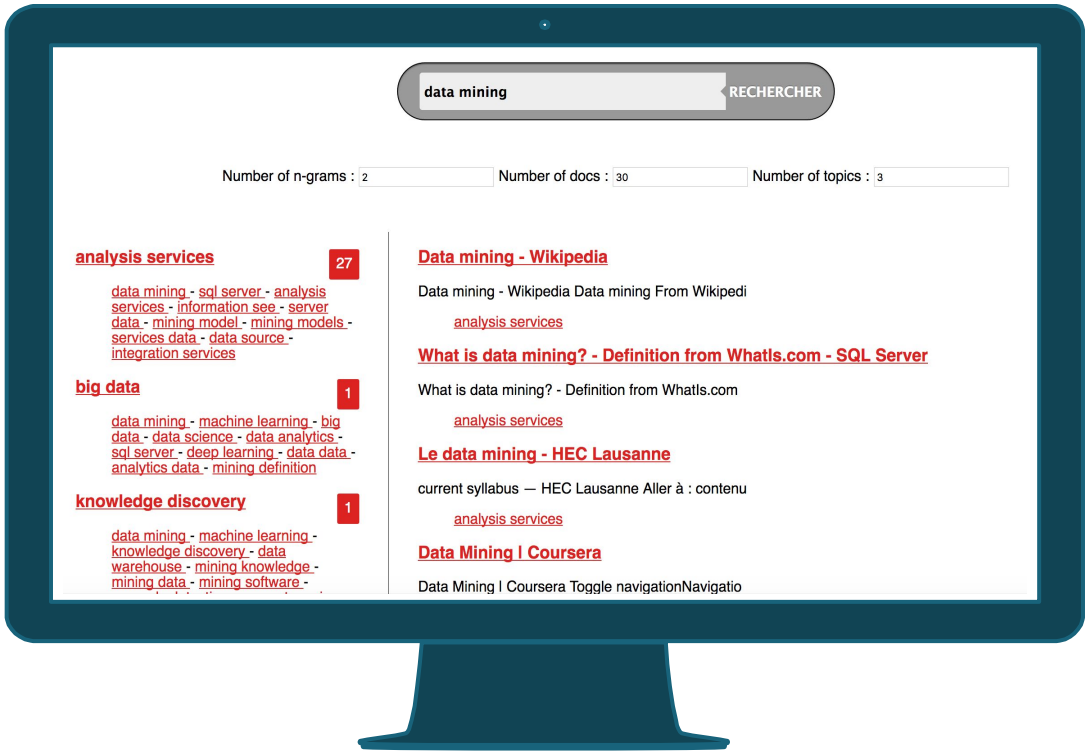
Interface


Client
Request
to resources
Data
REST API
Json
Scraping
(beautifulsoup)
Google
Algorithme – LDA

6

Démo

Interface web (Application)



7

Nommage des Topics

Nommage des Topics

- LDA n’associe pas de noms aux clusters

```
[(0, u0.005*data science + 0.005*decision making + 0
.005*sequential patterns + 0.005*rights reserved + 0
.005*raw data + 0.005*contact us + 0.005*center data
 + 0.005*r u + 0.005*k l + 0.005*j k), (1, u0.113*da
ta mining + 0.071*pattern discovery + 0.064*text dat
a + 0.030*mine data + 0.023*discovery data + 0.016*s
equential patterns + 0.016*different ways + 0.016*mi
ning also + 0.016*solve problem + 0.016*transactiona
l data), (2, u0.233*data mining + 0.032*data warehou
se + 0.014*mining tools + 0.014*mining data + 0.012*
data preparation + 0.012*data data + 0.012*mining te
chniques + 0.010*mining process + 0.010*analysis dat
a + 0.009*large data)]
```

Nommage des Topics

- Pas de solution standard
- Approche simple et efficace:
  - Utiliser l' n-gram au plus grand poid qui ne figure pas dans d'autres topics
- Problèmes?
  - Si beaucoup de termes sont communs, beaucoup de clusters => nom peu pertinent

web usage 27

web mining - data mining - mining web - web usage - web content - usage mining - structure mining - content mining - web structure - web data

ieee xplore 7

ieee xplore - username password - personal sign - terms use - contact us - rights reserved - world largest - use web - terms conditions - digital library

web data 1

data mining - web mining - web data - mining web - big data - mining software - e mail - web site - data analytics - web usage

8

Améliorations

Améliorations

- Améliorer l'interface
  - présenter les topics sous forme graphique (wordcloud)
  - catégoriser le contenu des documents par topic (colorisation)
- Optimiser la performance de la collection des données
- Tester les autres algorithmes (ex: LSA)
- Améliorer l'algorithme de nommage du topic

Merci !

Questions ?

Github : https://github.com/anthonyhseb/topics

Annexes

LDA

Suppose you have the following set of sentences:

- I ate a banana and spinach smoothie for breakfast
- I like to eat broccoli and bananas.
- Chinchillas and kittens are cute.
- My sister adopted a kitten yesterday.
- Look at this cute hamster munching on a piece of broccoli.

LDA : automatically discovering topics that these sentences contain.

- Sentences 1 and 2: 100% Topic A
- Sentences 3 and 4: 100% Topic B
- Sentence 5: 60% Topic A, 40% Topic B

- Topic A: 30% broccoli, 15% bananas, 10% breakfast, 10% munching, … (at which point, you could interpret topic A to be about food)
- Topic B: 20% chinchillas, 20% kittens, 20% cute, 15% hamster, … (at which point, you could interpret topic B to be about cute animals)

The question, of course, is: how does LDA perform this discovery?

LDA - Pseudocode

- For each document, randomly assign each word in the document to one of the K topics.

Note: this random assignment gives us topic representations of all the documents and word distributions of all the topics

To improve :

- For each document d...
- ....Go through each word w in d...
- ..... for each topic t:
- ..... p(topic t | document d)
- ..... p(word w | topic t)
- ..... Reassign w a new topic

  => choose topic t with probability p(topic t | document d) * p(word w | topic t)
- Repeating previous steps a large number of times => convergence