Trouver des documents similaires

M. Beigbeder

EMSE

May 15, 2018

Applications

- Documents (sites, pages Web, articles, dépêches) avec des mots similaires (miroir, plagiat, classification)
- Trouver des spectateurs qui ont les mêmes goûts...
- ...ou trouver des films qui ont les mêmes spectateurs
- Résolution d'entité

Exemple d'application: Construction du graphe de citations (1/2)

Extrait des données téléchargées depuis la bibliothèque numérique ISTEX au format Json:

```
{
  "corpusName": "elsevier",
  "author": ...
  "title": "Nuclear antigens in the HeLa cell cycle"...
  "refBibs": [
        "title": "Multiplicaiton and division in Mammalian  Cells",
            ...
             ]
}
...
{
   ...
  "title": "Multiplication and Division in Mammalian Cells",
   ...
}
...
```

Exemple d'application: Construction du graphe de citations (2/2)

7 316 816	titres citants
117 946 803	titres cités
125 263 619	titres

Volume de données: 9,6 Gio (1950–2005, titre de plus de 6 caractères et moins de 300 caractères)

Le but: reconnaître les titres qui se ressemblent.

Le but

3 486	0956 a study of the conditions and mechanisms of the diphenylamine reacti
108 310	1956 a study of the conditions and mechanism ofthe diphenylamine reaction
46 852 687	zzzv and aaaa2 v a decade later he spoke of ashmole as my honoured friend

Le but: reconnaître les titres qui se ressemblent.

Distance entre deux « textes »

- Sur les chaînes elles-mêmes:
  Distance d'édition (ou distance de Levenshtein)
  Différentes versions:
  - nombre d'insertion et de suppression de caractères
  - nombre d'insertion, suppression et remplacement de caractères
  - avec des coûts différents selon les caractères concernés
- Sur des vecteurs (ou multi-ensembles) issus des documents (termes)
  Cf. $tf \cdot idf$
- Sur des ensembles issus des documents (termes, k-grammes)
  Cf. infra

Le moyen. . .

Le but: reconnaître les titres qui se ressemblent

- une première étape, de complexité linéaire ($O(n)$)
  - normaliser
  - c'est-à-dire remplacer les caractères non alphanumériques par des espaces, conversions en minuscules, compactage des espaces
- une deuxième étape, de complexité pseudo-linéaire ($O(n \log n)$)
  - pour trouver les duplicats exacts
  - résultat: 46 852 688 titres normalisés uniques

Le moyen. . . qui n'en est pas un

- une troisième étape, de complexité quadratique ($O(n^2)$)
  - comparer chaque paire selon une similarité (Jaccard, Levenshtein, etc.)
  - temps à raison de 20–30 μs la comparaison: un millénaire
- et choisir un seuil de similarité

Similarité de Jaccard entre deux ensembles A et B:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

`1956 a study of the conditions and mechanism of the diphenylamine [...]`

`1956_a` `956_a_` `56_a_s` `6_a_st` `_a_stu` `a_stud` `_study` `[...]`

`0956 a study of the conditions and mechanisms ofthe diphenylamine [...]`

`0956_a` `956_a_` `56_a_s` `6_a_st` `_a_stu` `a_stud` `_study` `[...]`

Ensembles, similarité

- Similarité de Jaccard (ou indice de Jaccard ou coefficient de Jaccard) entre deux **ensembles** est le cardinal de leur union divisé par le cardinal de leur intersection.

$$Sim(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Exemple

Ensembles et matrices booléennes

- Lignes: éléments de l'univers
- Colonnes: ensembles (sous-ensembles de l'univers)
- Valeur 1 dans la ligne e et colonne S **ssi** e appartient à S
- Similarité de colonnes: similarité de Jaccard des ensembles qu'elles représentent
- Typiquement ces matrices sont creuses

Exemple 011010, 101011

Trois techniques assemblées pour trouver des documents similaires

- Tuilage (*shingling*): conversion de textes en **ensembles** de k-grammes

  ensemble au sens mathématique

- Hachage par minimum (*Minhashing*): conversion de grandes **fonctions caractéristiques** en signatures, en préservant la similarité

  fonction caractéristique au sens mathématique

  grande au sens où l'univers est grand

- Hachage sensible à la localité (*Locality Sensitive Hashing, LSH*): regroupe dans des mêmes alvéoles les signatures similaires

Tuilage

Tuilage

- un *k-gramme* est une séquence de k caractères qui apparaît dans le document
- Exemple: document abcab. Ensemble des bigrammes: {ab, bc, ca, ab} qui est un ensemble à trois éléments plus couramment écrit {ab, bc, ca}
- Le tuilage construit les représentations des documents par leur ensemble des k-grammes.

Tuilage et similarité

- Des documents intuitivement similaires ont beaucoup de k-grammes en commun
- changer un mot affecte uniquement les k-grammes à une distance k du mot.
- changer l'ordre de phrases change uniquement les $2k$ k-grammes qui traversent les frontières de phrases.
- Exemple: Avec k égal à 3. « le chien court devant le chat » et « le chien court derrière le chat ». Les k-grammes du premier texte qui sont remplacés sont: der, err, rri, riè, ièr, ère, e_l.

Construisez les ensembles de 2-grammes pour les documents ABRACADABRA et BRICABRAC.
Quelle est la similarité de Jaccard entre les ensembles de 2-grammes de ces deux documents?

Option de tuilage: compression

- Si k est grand, on peut remplacer les k-grammes par une valeur de hachage sur (disons) 4 octets.
- Le document est représenté par l'ensemble des valeurs de hachage de ses k-grammes.
- Il est possible que deux documents partagent des valeurs de hachage sans partager les k-grammes.

Hachage par minimum

Hachage par minimum

- Soit σ une permutation des lignes de la matrice booléenne
- La fonction de hachage par minimum h_σ associe à une colonne le numéro de la première ligne dans laquelle la colonne C contient un 1
- La signature d'une colonne est le résultat de l'application de plusieurs (disons: 100) fonctions de hachage par minimum indépendantes à une colonne.
- On peut représenter les signatures de la collection par une matrice (pleine ou non-creuse) de signatures

Exemple colonnes: 1100011 0011100 1000011 0111100
permutations: 1376254 4213675 3476125

La propriété

- La probabilité (sur toutes les permutations des lignes) que deux colonnes aient la même valeur de hachage par minimum est la similarité de Jaccard de ces colonnes

$$P(h(C_1) = h(C_2)) = Sim(C_1, C_2)$$

Similarité de signatures

- La similarité de signatures est la fraction des valeurs de hachage par minimum qui sont égales.
- L'espérance de la similarité entre les signatures de deux colonnes est égale à la similarité de Jaccard des ensembles que ces deux colonnes représentent.
  Plus longues sont les signatures, plus petite est l'espérance de l'erreur.

Exemple colonnes: 1100011 0011100 1000011 0111100
permutations: 1376254 4213675 3476125

Implémentation (1/4)

- Avec un milliard de lignes,...
- ... difficile de choisir une permutation de $[1..10^9]$
- ... difficile de représenter en extension une permutation
- Accès aux données

Comment (algorithme) permuter les n valeurs d'un tableau ?

Implémentation (2/4)

- Approximation à la permutation: hachage
  $h_i : \mathbb{N} \to \mathbb{N}$ et $h_i(x)$ est la nouvelle position de la ligne x
- Garder un tableau à deux dimensions indexé par les colonnes et les fonctions de hachage h_i: $M[i, c]$
- ... pour stocker le minimum des $h_i(r)$ pour laquelle la colonne C a un 1 dans la ligne r.

Implémentation (3/4)

for each row r **do**
  for each hash function h_i (*i.e.* each permutation) **do**
    compute and store $h_i(r)$
  end for
  for each column C **do**
    if C has 1 in row r **then**
      for each hash function h_i // each permutation **do**
        if $h_i(r)$ is smaller than $M(i, c)$ **then**
          $M(i, c) = h_i(r)$
        end if
      end for
    end if
  end for
end for

Exemple $h_1(x) = x \mod 5$ $h_2(x) = (2x + 1) \mod 5$

Implémentation (4/4)

- Et si les données arrivent par colonne et non par ligne... par exemple, les colonnes représentent des documents et les lignes des k-grammes
- ... trier les données

Hachage sensible à la localité

Hachage sensible à la localité, HSL

Locality sensitive hashing, LSH

- Idée générale: à partir de la collection (de signatures) construire une petite liste de *paires de candidats*; c.-à-d. des paires d'éléments pour lesquels la similarité sera calculée.
- Calculer les valeurs de hachage des éléments et insérer les éléments dans les alvéoles correspondantes. Les paires d'éléments dans une alvéole seront les *paires de candidats*.

LSH for Minhash

- Idée: calculer plusieurs valeurs de hachage pour les colonnes
- Faire en sorte que les colonnes similaires aient de bonnes chances d'avoir la même valeur de hachage
- Les paires de candidats sont celles qui ont au moins une fois la même valeur de hachage pour une même fonction de hachage.

Partition en bandes

- Partionner les lignes de la matrice de signatures en b bandes
- Chaque bande contient r lignes
  il y a donc $b \times r$ lignes au total
- Pour chaque bande, on construit une table de hachage et on y insère les (identifiants de) colonnes.
  Nota bene: ces tables doivent être les plus grandes possibles
- Les paires de candidats sont celles qui se retrouve dans la même alvéole pour au moins une bande (une table de hachage)
- Question: choisir b et r

Exemple Dessin

Besoin

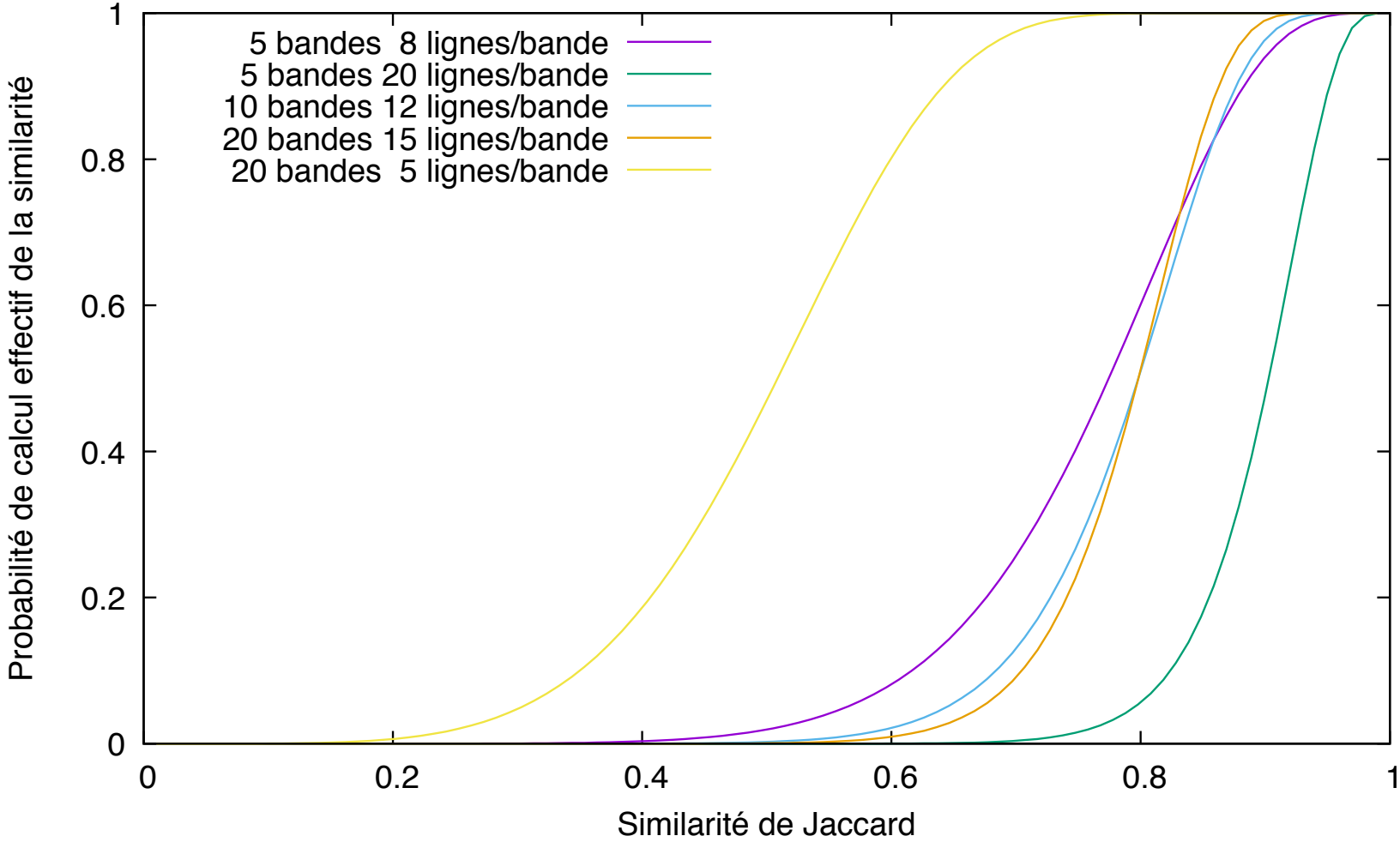
- Nous voulons les paires de documents de similarité supérieure à un seuil: par exemple seuil (*threshold*) $t = 80\% = 0.8$;
- Avec des signatures de (p.ex.) 100 entiers répartis en $b = 20$ bandes de $r = 5$ entiers par bande

Choisir b et r

Supposons que C_1 et C_2 ont une similarité p supérieure à t

- Probabilité que C_1 et C_2 soient identiques dans une bande donnée de r lignes: $p^r \geq (0.8)^5 = 0.328$
- Probabilité que C_1 et C_2 ne soient pas identiques dans cette bande $(1 - p^r) \leq (1 - 0.328) = 0.672$
- Probabilité que C_1 et C_2 ne soient identiques dans aucune bande $(1 - t^r)^b \leq (0.672)^{20} = .00035$
  Autrement dit moins de $0,3\%$ des paires de documents de similarité supérieure à 80% seront des faux négatifs (similarité non détectée car n'étant dans la même alvéole pour aucune bande).
- Probabilité que C_1 et C_2 soient identiques dans au moins une bande $1 - (1 - t^r)^b \geq 1 - (0.672)^{20} = 1 - .00035 = .99965$

Choix du nombre de bandes b et de lignes par bande r



C1	C2	C3	C4	C5	C6	C7
1	2	1	1	2	5	4
2	3	4	2	3	2	2
3	1	2	3	1	3	2
4	1	3	1	2	4	4
5	2	5	1	1	5	1
6	1	6	4	1	1	4

Ci-dessus la matrice de signatures de sept documents. On considère un hachage sensible à la localité avec trois bandes de deux lignes chacune. En supposant qu'il y a assez d'alvéoles disponibles de façon à ce que les fonctions de hachage pour chaque bande soient parfaites, trouvez toutes les paires candidates.

Expliquez le-s compromis à prendre en compte dans le choix du nombre de bandes (b dans la diapo 28 « Partition en bandes ») et du nombre de lignes par bande (r dans la diapo 28).

Quelles sont les similarités de Jaccard entre les ensembles de bigrammes des paires de mots parmi ces quatre mots bricolage, bricoler, abri, col?
Représentez les ensembles de ces quatre mots sous forme d'une matrice avec un bigramme par ligne et une colonne par ensemble.
Calculez les hachages par minimum avec les trois permutations suivantes sur les lignes: $2x + 1 \mod 11$, $3x + 2 \mod 11$, $4x + 6 \mod 11$, $5x + 10 \mod 11$ pour obtenir la signature de ces trois ensembles.
Quelles sont les paires de candidats par le Hachage sensible à la localité en considérant deux bandes de deux lignes?