

IC05 2004 une introduction à l'Information Retrieval

Eustache DIEMERT

Université de Technologie de Compiègne - Réseaux, Territoires et Géographie de
l'Information

11 mai 2004

Outline

Techniques classiques d'Information Retrieval

- Requêtes booléennes et Index inverse
- Classement par mesure de Pertinence
- Recherche par Similarité

Outline

Techniques classiques d'Information Retrieval

- Requêtes booléennes et Index inverse
- Classement par mesure de Pertinence
- Recherche par Similarité

Outline

Techniques classiques d'Information Retrieval

- Requêtes booléennes et Index inverse
- Classement par mesure de Pertinence
- Recherche par Similarité

Le paradigme de l'Information Retrieval (IR)

- "L'information existe, il suffit de la trouver..."
- Définition:
 - 1 Préparation d'un index de mots-clefs pour un corpus donné
 - 2 Réponse à une requête de mots-clefs ...
 - 3 sous la forme d'une liste ordonnée
- Maturité de l'IR traditionnelle dans les années 80 (Salton, etc)

Le paradigme de l'Information Retrieval (IR)

- "L'information existe, il suffit de la trouver..."
- Définition:
 - ① Préparation d'un index de mots-clefs pour un corpus donné
 - ② Réponse à une requête de mots-clefs ...
 - ③ sous la forme d'une liste ordonnée
- Maturité de l'IR traditionnelle dans les années 80 (Salton, etc)

Le paradigme de l'Information Retrieval (IR)

- "L'information existe, il suffit de la trouver..."
- Définition:
 - 1 Préparation d'un index de mots-clefs pour un corpus donné
 - 2 Réponse à une requête de mots-clefs ...
 - 3 sous la forme d'une liste ordonnée
- Maturité de l'IR traditionnelle dans les années 80 (Salton, etc)

Situation actuelle

Les techniques d'IR sont utilisées comme base de la plupart des systèmes d'information peu/pas structurés

- Systèmes d'aide pour utilisateurs finaux (MS Windows, Word)
- Moteurs de Recherche tradi (Google)
- Systèmes de Veille multi-sources (IBM's WebFountain)
- Mais
- limites dans un contexte *hypertexte à grande échelle*
- Nécessité de coupler ces techniques à des algorithmes tiers

Situation actuelle

Les techniques d'IR sont utilisées comme base de la plupart des systèmes d'information peu/pas structurés

- Systèmes d'aide pour utilisateurs finaux (MS Windows, Word)
- Moteurs de Recherche tradi (Google)
- Systèmes de Veille multi-sources (IBM's WebFountain)
- **Mais**
- limites dans un contexte *hypertexte à grande échelle*
- Nécessité de coupler ces techniques à des algorithmes tiers

Situation actuelle

Les techniques d'IR sont utilisées comme base de la plupart des systèmes d'information peu/pas structurés

- Systèmes d'aide pour utilisateurs finaux (MS Windows, Word)
- Moteurs de Recherche tradi (Google)
- Systèmes de Veille multi-sources (IBM's WebFountain)
- **Mais**
- limites dans un contexte *hypertexte à grande échelle*
- Nécessité de coupler ces techniques à des algorithmes tiers

Situation actuelle

Les techniques d'IR sont utilisées comme base de la plupart des systèmes d'information peu/pas structurés

- Systèmes d'aide pour utilisateurs finaux (MS Windows, Word)
- Moteurs de Recherche tradi (Google)
- Systèmes de Veille multi-sources (IBM's WebFountain)
- **Mais**
- limites dans un contexte *hypertexte à grande échelle*
- Nécessité de coupler ces techniques à des algorithmes tiers

Situation actuelle

Les techniques d'IR sont utilisées comme base de la plupart des systèmes d'information peu/pas structurés

- Systèmes d'aide pour utilisateurs finaux (MS Windows, Word)
- Moteurs de Recherche tradi (Google)
- Systèmes de Veille multi-sources (IBM's WebFountain)
- **Mais**
 - limites dans un contexte *hypertexte à grande échelle*
 - Nécessité de coupler ces techniques à des algorithmes tiers

Situation actuelle

Les techniques d'IR sont utilisées comme base de la plupart des systèmes d'information peu/pas structurés

- Systèmes d'aide pour utilisateurs finaux (MS Windows, Word)
- Moteurs de Recherche tradi (Google)
- Systèmes de Veille multi-sources (IBM's WebFountain)
- **Mais**
- limites dans un contexte *hypertexte à grande échelle*
- Nécessité de coupler ces techniques à des algorithmes tiers

Situation actuelle

Les techniques d'IR sont utilisées comme base de la plupart des systèmes d'information peu/pas structurés

- Systèmes d'aide pour utilisateurs finaux (MS Windows, Word)
- Moteurs de Recherche tradi (Google)
- Systèmes de Veille multi-sources (IBM's WebFountain)
- **Mais**
- limites dans un contexte *hypertexte à grande échelle*
- Nécessité de coupler ces techniques à des algorithmes tiers

1 Requêtes booléennes et Index inverse

Les Requêtes

Requêtes booléennes:

- documents contenant le terme "Java"
↔ Java
- ou le terme "Java" mais pas "coffee"
↔ Java -coffee

Requêtes de proximité:

- ou bien: la phrase "Java beans" ou le terme "API"
↔ (Java beans) OR API
- ou "Java" et "island" dans la même phrase
↔ Java NEAR island

Les Requêtes

Requêtes booléennes:

- documents contenant le terme "Java"
↔ Java
- ou le terme "Java" mais pas "coffee"
↔ Java -coffee

Requêtes de proximité:

- ou bien: la phrase "Java beans" ou le terme "API"
↔ (Java beans) OR API
- ou "Java" et "island" dans la même phrase
↔ Java NEAR island

Les Requêtes

Requêtes booléennes:

- documents contenant le terme "Java"
↪ Java
- ou le terme "Java" mais pas "coffee"
↪ Java -coffee

Requêtes de proximité:

- ou bien: la phrase "Java beans" ou le terme "API"
↪ (Java beans) OR API
- ou "Java" et "island" dans la même phrase
↪ Java NEAR island

Les Requêtes

Requêtes booléennes:

- documents contenant le terme "Java"
↪ Java
- ou le terme "Java" mais pas "coffee"
↪ Java -coffee

Requêtes de proximité:

- ou bien: la phrase "Java beans" ou le terme "API"
↪ (Java beans) OR API
- ou "Java" et "island" dans la même phrase
↪ Java NEAR island

Indexation

- Phase d'Extraction (*tokenization in english*)
 - token* : séquence de caractères sans ponctuation ni espace
 - filtrage des tags HTML
 - mise en minuscules, transcodage
- Compression destructive (élimination des mot-outils, des pluriels etc)
- Phase de Codage : création d'un *index inverse*
 - attribution d'un identifiant (*id*) à chaque token
 - codage d'un document en une liste (ordonnée) d'id
 - stockage dans un SGBD des couples (token,document)

- Résultat:

Dictionnaire = ((id1 \mapsto token1), ..., (idn \mapsto tokenn))

Index = ((id1 \mapsto Doc3, Doc17, ...), ..., (id2 \mapsto Doc14))

Indexation

- Phase d'Extraction (*tokenization in english*)
 - token* : séquence de caractères sans ponctuation ni espace
 - filtrage des tags HTML
 - mise en minuscules, transcodage
- Compression destructive (élimination des mot-outils, des pluriels etc)
- Phase de Codage : création d'un *index inverse*
 - attribution d'un identifiant (*id*) à chaque token
 - codage d'un document en une liste (ordonnée) d'id
 - stockage dans un SGBD des couples (token,document)

- Résultat:

Dictionnaire = ((id1 \mapsto token1), ..., (idn \mapsto tokenn))

Index = ((id1 \mapsto Doc3,Doc17,...), (id2 \mapsto Doc14))

Indexation

- Phase d'Extraction (*tokenization in english*)
 - token* : séquence de caractères sans ponctuation ni espace
 - filtrage des tags HTML
 - mise en minuscules, transcodage
- Compression destructive (élimination des mot-outils, des pluriels etc)
- Phase de Codage : création d'un *index inverse*
 - attribution d'un identifiant (*id*) à chaque token
 - codage d'un document en une liste (ordonnée) d'id
 - stockage dans un SGBD des couples (token,document)

- Résultat:

Dictionnaire = ((id1 \mapsto token1), ..., (idn \mapsto tokenn))

Index = ((id1 \mapsto Doc3,Doc17,...), (id2 \mapsto Doc14))

Indexation

- Phase d'Extraction (*tokenization in english*)
 - token* : séquence de caractères sans ponctuation ni espace
 - filtrage des tags HTML
 - mise en minuscules, transcodage
- Compression destructive (élimination des mot-outils, des pluriels etc)
- Phase de Codage : création d'un *index inverse*
 - attribution d'un identifiant (*id*) à chaque token
 - codage d'un document en une liste (ordonnée) d'id
 - stockage dans un SGBD des couples (token,document)
- Résultat:
Dictionnaire = ((id1 \mapsto token1), ..., (idn \mapsto tokenn))
Index = ((id1 \mapsto Doc3,Doc17,...), (id2 \mapsto Doc14), ...)

Stop-words & Stemming

Il s'agit de compresser de manière destructive l'information.

- Stop-words

mots-outils ne renseignant pas sur le sujet du discours,
mais sur son articulation (de, la, donc, pour)
suppression ou bien marquage des "emplacements vides"

- Exemple

brutal: il a vu le chat \mapsto vu chat

fin: X X vu X chat (polysémie réduite)

- Stemming ou racinisation

goes, go, went gone \mapsto go

analyse combine morphosyntaxe (Porter) et dictionnaire

problème: Porter(université) = Porter(universel) =

univers

- Phase de tuning vis-à-vis de l'application : adaptez le ratio

Stop-words & Stemming

Il s'agit de compresser de manière destructive l'information.

- Stop-words

mots-outils ne renseignant pas sur le sujet du discours,
mais sur son articulation (de, la, donc, pour)
suppression ou bien marquage des "emplacements vides"

- Exemple

brutal: il a vu le chat \mapsto vu chat

fin: X X vu X chat (polysémie réduite)

- Stemming ou racinisation

goes, go, went gone \mapsto go

analyse combine morphosyntaxe (Porter) et dictionnaire

problème: Porter(université) = Porter(universel) =

univers

- Phase de tuning vis-à-vis de l'application : adapter le ratio

Stop-words & Stemming

Il s'agit de compresser de manière destructive l'information.

- Stop-words

mots-outils ne renseignant pas sur le sujet du discours,
mais sur son articulation (de, la, donc, pour)
suppression ou bien marquage des "emplacements vides"

- Exemple

brutal: il a vu le chat \mapsto vu chat

fin: X X vu X chat (polysémie réduite)

- Stemming ou racinisation

goes, go, went gone \mapsto go

analyse combine morphosyntaxe (Porter) et dictionnaire


problème: Porter(université) = Porter(universel) =

univers

- Phase de tuning vis-à-vis de l'application : adapter le ratio

Stop-words & Stemming

Il s'agit de compresser de manière destructive l'information.

- Stop-words
mots-outils ne renseignant pas sur le sujet du discours,
mais sur son articulation (de, la, donc, pour)
suppression ou bien marquage des "emplacements vides"
- Exemple
brutal: il a vu le chat \mapsto vu chat
fin: X X vu X chat (polysémie réduite)
- Stemming ou racinisation
goes, go, went gone \mapsto go
analyse combine morphosyntaxe (Porter) et dictionnaire
problème: Porter(université) = Porter(universel) =
univers
- Phase de tuning vis-à-vis de l'application : adapter le ratio 

Précision / Rappel

- Mesures de qualité (*IR tradi.*)

- Précision

$$\frac{NbDocsPertinents}{NbDocsRamenes}$$

- Rappel

$$\frac{NbDocsRamenes}{NbDocsPertinentsExistants}$$

- peu (précision) ou pas (rappel) adaptés au Web, mais importants pour mesurer l'efficacité sur un corpus-test

Précision / Rappel

- Mesures de qualité (*IR tradi.*)
- Précision

$$\frac{NbDocsPertinents}{NbDocsRamenes}$$

- Rappel

$$\frac{NbDocsRamenes}{NbDocsPertinentsExistants}$$

- peu (précision) ou pas (rappel) adaptés au Web, **mais** importants pour mesurer l'efficacité sur un corpus-test

Précision / Rappel

- Mesures de qualité (*IR tradi.*)

- Précision

$$\frac{NbDocsPertinents}{NbDocsRamenes}$$

- Rappel

$$\frac{NbDocsRamenes}{NbDocsPertinentsExistants}$$

- peu (précision) ou pas (rappel) adaptés au Web, **mais** importants pour mesurer l'efficacité sur un corpus-test

Précision / Rappel

- Mesures de qualité (*IR tradi.*)

- Précision

$$\frac{NbDocsPertinents}{NbDocsRamenes}$$

- Rappel

$$\frac{NbDocsRamenes}{NbDocsPertinentsExistants}$$

- peu (précision) ou pas (rappel) adaptés au Web, **mais** importants pour mesurer l'efficacité sur un corpus-test

Vector Space Model

On dispose déjà d'un index inverse pour sélectionner les documents répondant à une requête. Il s'agit de les classer par **une** mesure de "pertinence".

- Vector-Space Model (Salton 1980)

document = point dans un espace de mots

doc7=(1,3,0,5)

dico=("chambre", "air", "microsoft", "vélo")

approche géométrique → mesure de *distance*

- Application

requête "cyclisme vélo" est un point de l'espace

similarité = distance euclidienne(document, requête)

↔ produit scalaire(document, requête)

- Améliorations

Vector Space Model

On dispose déjà d'un index inverse pour sélectionner les documents répondant à une requête. Il s'agit de les classer par **une** mesure de "pertinence".

- Vector-Space Model (Salton 1980)

document = point dans un espace de mots

doc7=(1,3,0,5)

dico=("chambre", "air", "microsoft", "vélo")

approche géométrique → mesure de *distance*

- Application

requête "cyclisme vélo" est un point de l'espace

similarité = distance euclidienne(document, requête)

↔ produit scalaire(document, requête)

- Améliorations

Vector Space Model

On dispose déjà d'un index inverse pour sélectionner les documents répondant à une requête. Il s'agit de les classer par **une** mesure de "pertinence".

- Vector-Space Model (Salton 1980)

document = point dans un espace de mots

doc7=(1,3,0,5)

dico=("chambre", "air", "microsoft", "vélo")

approche géométrique → mesure de *distance*

- Application

requête "cyclisme vélo" est un point de l'espace

similarité = distance euclidienne(document, requête)

⇔ produit scalaire(document, requête)

- Améliorations

Les requêtes "documents similaires"

- Trouver des documents similaires à un document connu
- Vector-Space Model
 - documents = points dans un espace de mots
 - docs similaires \Leftrightarrow proches dans l'espace
 - \Leftrightarrow produit scalaire(document, requête) $\rightarrow 1$
- Améliorations
 - éliminer les miroirs
 - ...

Les requêtes "documents similaires"

- Trouver des documents similaires à un document connu
- Vector-Space Model
 - documents = points dans un espace de mots
 - docs similaires \Leftrightarrow proches dans l'espace
 - \Leftrightarrow produit scalaire(document, requête) $\rightarrow 1$
- Améliorations
 - éliminer les miroirs
 - ...

Les requêtes "documents similaires"

- Trouver des documents similaires à un document connu
- Vector-Space Model
 - documents = points dans un espace de mots
 - docs similaires \Leftrightarrow proches dans l'espace
 - \Leftrightarrow produit scalaire(document, requête) $\rightarrow 1$
- Améliorations
 - éliminer les miroirs
 - ...

Conclusion

Un paradigme puissant...

- Usage généralisé pour les couches basses des Systèmes d'Information
- Techniques robustes, notamment pour le Web (peu d'hypothèses)

mais peu convaincant seul !

- Limitation à une recherche d'un "déjà-connu"
- Limitations inhérentes aux listes (polysémie,...)

Donc:

- Utilisation de données contextualisantes:
 - liens hypertextes
 - hypothèses fortes (ontologies d'un domaine,...)
 - classification automatique
 - surcouche ergonomique spécialisée ("cartes" etc)

Conclusion

Un paradigme puissant...

- Usage généralisé pour les couches basses des Systèmes d'Information
- Techniques robustes, notamment pour le Web (peu d'hypothèses)

mais peu convaincant seul !

- Limitation à une recherche d'un "déjà-connu"
- Limitations inhérentes aux listes (polysémie,...)

Donc:

- Utilisation de données contextualisantes:
 - liens hypertextes
 - hypothèses fortes (ontologies d'un domaine,...)
 - classification automatique
 - surcouche ergonomique spécialisée ("cartes" etc)

Conclusion

Un paradigme puissant...

- Usage généralisé pour les couches basses des Systèmes d'Information
- Techniques robustes, notamment pour le Web (peu d'hypothèses)

mais peu convaincant seul !

- Limitation à une recherche d'un "déjà-connu"
- Limitations inhérentes aux listes (polysémie,...)

Donc:

- Utilisation de données contextualisantes:
 - liens hypertextes
 - hypothèses fortes (ontologies d'un domaine,...)
 - classification automatique
 - surcouche ergonomique spatialisante ("cartes" etc.)

Conclusion

Un paradigme puissant...

- Usage généralisé pour les couches basses des Systèmes d'Information
- Techniques robustes, notamment pour le Web (peu d'hypothèses)

mais peu convaincant seul !

- Limitation à une recherche d'un "déjà-connu"
- Limitations inhérentes aux listes (polysémie,...)

Donc:

- Utilisation de données contextualisantes:
 - liens hypertextes
 - hypothèses fortes (ontologies d'un domaine,...)
 - classification automatique
 - surcouche ergonomique spatialisante ("cartes", etc.)

Conclusion

Un paradigme puissant...

- Usage généralisé pour les couches basses des Systèmes d'Information
- Techniques robustes, notamment pour le Web (peu d'hypothèses)

mais peu convaincant seul !

- Limitation à une recherche d'un "déjà-connu"
- Limitations inhérentes aux listes (polysémie,...)

Donc:

- Utilisation de données contextualisantes:
 - liens hypertextes
 - hypothèses fortes (ontologies d'un domaine,...)
 - classification automatique
 - surcouche ergonomique spatialisante ("cartes", etc.)

Conclusion

Un paradigme puissant...

- Usage généralisé pour les couches basses des Systèmes d'Information
- Techniques robustes, notamment pour le Web (peu d'hypothèses)

mais peu convaincant seul !

- Limitation à une recherche d'un "déjà-connu"
- Limitations inhérentes aux listes (polysémie,...)

Donc:

- Utilisation de données contextualisantes:
 - liens hypertextes
 - hypothèses fortes (ontologies d'un domaine,...)
 - classification automatique
 - surcouche ergonomique spatialisante ("cartes" etc.)