INF344: Données du Web

Les quatre « v » de la datamasse du Web

2 mai 2016

TELECOM ParisTech

La datamasse

La datamasse, le Big Data, les masses de données :

- Données collectées pour certaines applications, par certaines entreprises, données librement disponibles, etc.
- Très grande valeur à analyser (fouille, prédiction)
- Significativement plus complexe que les données traditionnelles :

  Volume : ordres de grandeur au-dessus de ce qu'on peut traiter de manière centralisé

  Variété : types de données (texte, multimédia, graphes, structurées) variés, organisation des données variées

  Vélocité : données changeantes ou produites à grande vitesse (LHC : 100 millions de collision par seconde), parfois plus que ce qu'on est capable de stocker

  Véracité : qualité de l'information très variable ; imprécision dans l'information

Affronter la datamasse

- Impacte des domaines variés : fouille de données, apprentissage, visualisation, respect de la vie privée. . . et bien sûr gestion de données
- On a besoin de nouveaux algorithmes, de nouveaux outils, de nouveaux modèles
- Ce cours : focus sur les problèmes de gestion de données issues du Web
- On va bien au-delà de SQL sur des SGBD relationnel. . . mais on retrouve les mêmes concepts de base

Plan

Volume

Variété

Vélocité

Véracité

Conclusion

Applications traditionnelles de gestion de données

- Données d'un forum Web populaire
  - 1000 posts par jour
  - 5 Kio de données par post
  - 10 ans de durée de vie

Applications traditionnelles de gestion de données

- Données d'un forum Web populaire
  - 1000 posts par jour
  - 5 Kio de données par post
  - 10 ans de durée de vie

$\Rightarrow$ $\sim$20Gio de données. Tient facilement sur n'importe quel système de gestion de données (p. ex., MySQL).

Applications traditionnelles de gestion de données

- Données d'un forum Web populaire
  - 1000 posts par jour
  - 5 Kio de données par post
  - 10 ans de durée de vie

  $\Rightarrow$ $\sim$20Gio de données. Tient facilement sur n'importe quel système de gestion de données (p. ex., MySQL).
- Comptes d'une banque
  - 10 millions de clients
  - 5 transactions par jour
  - 100 octets par transaction
  - 1 an d'historique dans le système de production

Applications traditionnelles de gestion de données

- Données d'un forum Web populaire
  - 1000 posts par jour
  - 5 Kio de données par post
  - 10 ans de durée de vie

  $\Rightarrow$ $\sim$20Gio de données. Tient facilement sur n'importe quel système de gestion de données (p. ex., MySQL).
- Comptes d'une banque
  - 10 millions de clients
  - 5 transactions par jour
  - 100 octets par transaction
  - 1 an d'historique dans le système de production

  $\Rightarrow$ $\sim$2 Tio de données. Tient dans un système de bases de données classiques, sur un serveur, ou, mieux, distribué sur quelques serveurs (p. ex., Oracle, DB2, PostgreSQL).

Données massives

- Google Search : 850 Tio de données (2006) [Chang et al., 2006]
- Google Earth : 70 Tio de données (2006) [Chang et al., 2006]

Données massives

- Google Search : 850 Tio de données (2006) [Chang et al., 2006]
- Google Earth : 70 Tio de données (2006) [Chang et al., 2006]
- Facebook
  - 1,5 milliards d'utilisateurs
  - $\sim$10 Mio de données par utilisateurs

Données massives

- Google Search : 850 Tio de données (2006) [Chang et al., 2006]
- Google Earth : 70 Tio de données (2006) [Chang et al., 2006]
- Facebook
  - 1,5 milliards d'utilisateurs
  - $\sim$10 Mio de données par utilisateurs

$\Rightarrow$ $\sim$15 Pio de données

Données massives

- Google Search : 850 Tio de données (2006) [Chang et al., 2006]
- Google Earth : 70 Tio de données (2006) [Chang et al., 2006]
- Facebook
  - 1,5 milliards d'utilisateurs
  - $\sim$10 Mio de données par utilisateurs

$\Rightarrow$ $\sim$15 Pio de données

Besoin d'autres formes de stockage et d'indexation de données sur une grappe de serveurs.

Données massives

- Google Search : 850 Tio de données (2006) [Chang et al., 2006]
- Google Earth : 70 Tio de données (2006) [Chang et al., 2006]
- Facebook
  - 1,5 milliards d'utilisateurs
  - $\sim$10 Mio de données par utilisateurs

$\Rightarrow$ $\sim$15 Pio de données

Besoin d'autres formes de stockage et d'indexation de données sur une grappe de serveurs.

Pas seulement une question de taille :

- Très grand nombre de requêtes par seconde
- Réponse rapide aux requêtes, où qu'on soit dans le monde

Principes généraux du stockage sur grappe

- Deux grandes stratégies :
  - Arbre de recherche distribué. Par exemple, BigTable [Chang et al., 2006] (Google), Apache HBase.
  - Table de hachage distribuée [Karger et al., 1997]. Par exemple, Dynamo (Amazon), Apache Cassandra, Project Voldemort.

Principes généraux du stockage sur grappe

- Deux grandes stratégies :
  - Arbre de recherche distribué. Par exemple, BigTable [Chang et al., 2006] (Google), Apache HBase.
  - Table de hachage distribuée [Karger et al., 1997]. Par exemple, Dynamo (Amazon), Apache Cassandra, Project Voldemort.
- Réplication des données pour
  1. Pas de perte de données suite à une faille matérielle
  2. Répartir la charge des lectures de données
  3. Éventuellement, plusieurs copies à différents emplacements pour une localité géographique

Principes généraux du stockage sur grappe

- Deux grandes stratégies :
  - Arbre de recherche distribué. Par exemple, BigTable [Chang et al., 2006] (Google), Apache HBase.
  - Table de hachage distribuée [Karger et al., 1997]. Par exemple, Dynamo (Amazon), Apache Cassandra, Project Voldemort.
- Réplication des données pour
  1. Pas de perte de données suite à une faille matérielle
  2. Répartir la charge des lectures de données
  3. Éventuellement, plusieurs copies à différents emplacements pour une localité géographique
- Limitations : requêtes moins expressives que dans les systèmes classiques, perte de cohérence du système

Principes généraux du stockage sur grappe

- Deux grandes stratégies :
  - Arbre de recherche distribué. Par exemple, BigTable [Chang et al., 2006] (Google), Apache HBase.
  - Table de hachage distribuée [Karger et al., 1997]. Par exemple, Dynamo (Amazon), Apache Cassandra, Project Voldemort.
- Réplication des données pour
  1. Pas de perte de données suite à une faille matérielle
  2. Répartir la charge des lectures de données
  3. Éventuellement, plusieurs copies à différents emplacements pour une localité géographique
- Limitations : requêtes moins expressives que dans les systèmes classiques, perte de cohérence du système
- Voir INF728 et les séances INF344 sur stockage distribué, HBase pour plus de détails

Plan

Volume

Variété

Vélocité

Véracité

Conclusion

Différentes sources organisent différemment les mêmes données

2007		
240	EE	Foto N. Afrati, Chen Li, Jeffrey D. Ullman: Using views to generate efficient evaluation plans for queries. J. Comput. Syst. Sci. 73(5): 703-724 (2007)
2005		
239	EE	Jeffrey D. Ullman: Gradiance On-Line Accelerated Learning. ACSC 2005: 3-6
238	EE	Serge Abiteboul, Rakesh Agrawal, Philip A. Bernstein, Michael J. Carey, Stefano Ceri, W. Bruce Croft, David J. DeWitt, Michael J. Franklin, Hector Garcia-Molina, Dieter Gawlick, Jim Gray, Laura M. Haas, Alon Y. Halevy, Joseph M. Hellerstein, Yannis E. Ioannidis, Martin L. Kersten, Michael J. Pazzani, Michael Lesk, David Maier, Jeffrey F. Naughton, Hans-Jörg Schek, Timos K. Sellis, Avi Silberschatz, Michael Stonebraker, Richard T. Snodgrass, Jeffrey D. Ullman, Gerhard Weikum, Jennifer Widom, Stanley B. Zdonik: The Lowell database research self-assessment. Commun. ACM 48(5): 111-118 (2005)
237	EE	Serge Abiteboul, Richard Hull, Victor Vianu, Sheila A. Greibach, Michael A. Harrison, Ellis Horowitz, Daniel J. Rosenkrantz, Jeffrey D. Ullman, Moshe Y. Vardi: In memory of Seymour Ginsburg 1928 - 2004. SIGMOD Record 34(1): 5-12 (2005)
2003		
236	EE	Jeffrey D. Ullman: A Survey of New Directions in Database System. DASFAA 2003: 3-
235	EE	Jeffrey D. Ullman: Improving the Efficiency of Database-System Teaching. SIGMOD Conference 2003: 1-3
234	EE	Jim Gray, Hans-Jörg Schek, Michael Stonebraker, Jeffrey D. Ullman: The Lowell Report. SIGMOD Conference 2003: 680
233	EE	Serge Abiteboul, Rakesh Agrawal, Philip A. Bernstein, Michael J. Carey, Stefano Ceri, W. Bruce Croft, David J. DeWitt, Michael J. Franklin, Hector Garcia-Molina, Dieter Gawlick, Jim Gray, Laura M. Haas, Alon Y. Halevy, Joseph M. Hellerstein, Yannis E. Ioannidis, Martin L. Kersten, Michael J. Pazzani, Michael Lesk, David Maier, Jeffrey F. Naughton, Hans-Jörg Schek, Timos K. Sellis, Avi Silberschatz, Michael Stonebraker, Richard T. Snodgrass, Jeffrey D. Ullman, Gerhard Weikum, Jennifer Widom, Stanley B. Zdonik: The Lowell Database Research Self Assessment CoRR cs.DB/0310006: (2003)

Différentes sources organisent différemment les mêmes données

Querying websites using compact skeletons - all 11 versions »
A Rajaraman, **JD Ullman** - Journal of Computer and System Sciences, 2003 - Elsevier
Several commercial applications, such as online comparison shopping and process automation, require integrating information that is scattered across multiple w ebsites or XML documents. Much research has been devoted to this problem, **...**
Cited by 13 - Related Articles - Web Search

[BOOK] Wprowadzenie do teorii automatów, jezyków i obliczen
JE Hopcroft, **JD Ullman**, B Konikow ska - 2003 - Wydaw . Naukow e PWN
Cited by 15 - Related Articles - Web Search

Improving the efficiency of database-system teaching - all 3 versions »
JD Ullman - Proceedings of the 2003 ACM SIGMOD international conference …, 2003 - portal.acm.org
ABSTRACT The education industry has a very poor record of produc- tivity gains. In this brief article, I outline some of the w ays the teaching of a college course in database systems could be made more ecient, and sta time used **...**
Cited by 4 - Related Articles - Web Search

A survey of new directions in database systems - all 5 versions »
JD Ullman - Database Systems for Advanced Applications, 2003.(DASFAA …, 2003 - ieeexplore.ieee.org
A survey of new directions in database systems. Ullman, JD Stanford University; This paper appears in: Database Systems for Advanced Applications, 2003. (DASFAA 2003). Proceedings. Eighth International **...**
Cited by 3 - Related Articles - Web Search

Intégration des données

- But : obtenir des données de différentes sources de données avec une interface/requête unique
- Exemples :
  - Science : interroger différentes bases de données génomiques
  - Commerce : interroger les catalogues de différents vendeurs
  - Administration : intégrer les données financières de différentes branches
  - Web : trouver des données sur une personne de nombreuses sources Web
- Tâche complexe : décrire des connections potentiellement complexes entre les sources de données, utiliser de la sémantique
- Buzz word : Web sémantique

Sémantique : la colle entre sources

- Les sources de données :
  - ont été développées indépendamment
  - sont autonomes
  - très hétérogènes
- De la sémantique est nécessaire pour relier les concepts et leurs structures
- De la logique est utilisée pour décrire cette sémantique

Exemple

- Où puis-je voir un film de Woody Allen aujourd'hui à Paris ?
  - Woody Allen *plays_in* un film X
  - X *is_shown_at_theater* Y
  - Y *is_located_in* Paris
- Ignorer les sources non pertinentes : Air France, etc.
- Trouver les sources pertinentes et comprendre comment les utiliser :
  - IMDB (Internet Movie Database) : films avec beaucoup d'informations ; fournit la liste des films de Woody Allen
  - Allociné : dit quand un film a lieu à Paris
- Combiner leurs résultats

Deux approches principales

- Poser les requêtes dans un schéma global
- Pour répondre, utiliser des données sur les schémas locaux
- Dans les deux approches, les formules relient les schémas locaux au schéma global
- Approche entrepôt
  - L'instance globale est matérialisée
  - Les données sont transformées depuis les instances locales et chargées dans l'instance globale
  - Les requêtes sont évaluées sur l'instance globale
- Approche médiateur
  - L'instance globale est virtuelle
  - Les requêtes sont évaluées en utilisant des requêtes aux instances locales

Vues

- L'intégration peut être approchée comme une vue sur les bases de données locales
- Une vue est une requête nommée qui peut être utilisée dans d'autres requêtes
- Exemple

  View1(X,Y1,Y2) : Flight(X) $\wedge$ DepartureAirport(X,Y1)
  $\wedge$ ArrivalAirport(X,Y2)

  View2(X,Y) : Place(X) $\wedge$ Located(X,Y) $\wedge$ Capital(Y)
- Vue matérialisée : calculée à l'avance et stockée, dans l'esprit de l'entrepôt
  - En mémoire ou en cache
  - Mises à jour coûteuses
  - Maintenance : propager les mises à jours pour actualiser la vue
- Vue virtuelle : à la demande, dans l'esprit de la médiation
  - Les requêtes sont coûteuses
  - La vue est recalculée à chaque utilisation

Deux principales approches : comparaison

- Approche entrepôt
  - Création : coût du calcul et du stockage
  - Évaluation des requêtes très efficace
  - Mises à jour coûteuses : besoin de propager les mises à jour locales vers l'entrepôt
    Sinon les données stockées peuvent êtres obsolètes
- Approche médiateur
  - Création : pas de coût
  - Requêtes : coût de la reformulation, peut-être du calcul, peut-être de la communication
  - Mises à jour : pas de coût
- Compromis classique en bases de données entre mises à jour et requêtes

L'approche médiateur – détails

- Schéma global : Définit un schéma médiateur
  - Vocabulaire structuré servant comme interface de requête pour les requêtes utilisateur
  - Typiquement, un schéma par domaine
- Schémas locaux : Déclare une source de données
  - Modèle le contenu de la source à intégrer en termes du schéma médiateur
  - Connecte les concepts/relations de la source à celles du schéma médiateur
- Traitement des requêtes
  - Reformuler et décomposer une requête utilisateur sur le schéma global en des requêtes sur le schéma local qui sont évaluées sur les sources de données
  - Combiner les réponses des requêtes locales pour construire la réponse à la requête globale

L'usage de la logique dans l'intégration

- Définir un schéma médiateur
  - Un schéma de bases de données
  - Contraintes : formules de logique du premier ordre
- Déclarer une source de données
  - Une source est une instance de bases de données
  - Liens avec le schéma médiateur : formules de logique du premier ordre
- Requêtes
  - Exprimées comme des formules de logique du premier ordre
  - L'évaluation de la requête globale peut utiliser un optimiseur de requêtes
  - Chaque évaluation de requête locale peut utiliser un optimiseur de requêtes

Pour plus de détail, voir les cours sur extraction d'informations, Web sémantique, fouille d'opinions, Web scraping

Plan

Volume

Variété

Vélocité

Véracité

Conclusion

Volatilité de l'information du Web

- La demi-vie du contenu du Web est de quelques années [Koehler, 2003]
- Sur les réseaux sociaux comme Twitter, l'information ne peut parfois plus être recherchée au bout d'une semaine [Twitter, 2011]
- Extrême diversité de taux de rafraîchissement du contenu des URLs, de la fraction de seconde à des dizaines d'années
- Indispensable d'archiver l'information du Web d'aujourd'hui pour les historiens de demain [Masanés, 2005]
- Utile pour un moteur de recherche comme Google de permettre de rechercher dans les actualités récentes

Rafraîchissement des URL

- Le contenu du Web change
- Taux de changements variables :

  page principale d'un site d'actualités : toutes les minutes

  article sur arXiv : essentiellement pas de changement
- Collecte continue et identification des taux de changements pour une collecte adaptative : comment déterminer la date de dernière modification d'une page Web ?

Estampille HTTP

Deux mécanismes d'estampille temporelle en HTTP : balises entités et dates de modification. Peuvent être fournies à chaque requête :

```
ETag: "497bef-1fcb-47f20645"
Last-Modified: Tue, 01 Apr 2008 09:54:13 GMT
```

Etag : identifiant unique pour le document fourni, change si le document change ; peut être utilisé dans des requêtes avec `If-Match` et `If-None-Match`.

Last-Modified : heure de dernière modification ; peut être utilisé dans des requêtes avec `If-Modified-Since` et `If-Unmodified-Since`.

- Information généralement fournie et fiable pour le contenu statique (p. ex., fichiers image)
- Information rarement fournie (ou avec une date fausse *maintenant*) pour le contenu dynamique

Estampilles dans le contenu des pages Web

- Très fréquent dans les sites Web dynamiques :
  - soit comme une estampille global (*Last modified :*) ;
  - soit sur des entrées individuelles : articles, commentaires, etc. (est-ce que l'estampille globale est le maximum des estampilles individuelles ?) ;
  - parfois également dans des méta-données de la page Web : commentaires HTML, balises `<meta>` Dublin Core.
- Relativement facile à identifier et à extraire de la page Web (mots-clefs, expressions rationnelles pour les dates).
- Informel : parfois partiel (pas d'indication de temps), souvent sans fuseau horaire.
- Pas nécessairement fiable.

Estampilles sémantiques aditionnelles

Fichiers d'autres types que HTML peuvent avoir des mécanismes d'estampille temporelle sémantique :

PDF, documents Office, etc. : date de création et de modification disponible en méta-données. Assez fiable.

Flux RSS : estampilles sémantiques fiables.

Images, sons : méta-données EXIF (ou similaire). Pas toujours fiable, et la date de capture d'une image peut ne pas avoir de rapport avec la date de publication.

Contenu sémantique externe utilisé pour dater une page Web :

- Possibilité d'apparier un flux RSS au contenu d'une page Web
- Sitemap fournie par le propriétaire du site.

Voir cours sur crawl Web (y compris crawl de contenus complexes) pour plus de détails.

Plan

Volume

Variété

Vélocité

Véracité

Conclusion

Données incertaines

Sources nombreuses de données incertaines :

- Erreurs de mesure
- Intégration de données de sources contradictoires
- Correspondances imprécises entre schémas hétérogènes
- Processus automatique incertaine (extraction d'information, traitement du langage naturel, etc.)
- Jugement humain imparfait
- Mensonges, opinions, rumeurs

Données incertaines

Sources nombreuses de données incertaines :

- Erreurs de mesure
- Intégration de données de sources contradictoires
- Correspondances imprécises entre schémas hétérogènes
- Processus automatique incertaine (extraction d'information, traitement du langage naturel, etc.)
- Jugement humain imparfait
- Mensonges, opinions, rumeurs

Cas d'étude : Extraction d'information Web

instance	iteration	date learned	confidence
arabic, egypt	406	08-sep-2011	(Seed) 100.0
chinese, republic_of_china	439	24-oct-2011	100.0
chinese, singapore	421	21-sep-2011	(Seed) 100.0
english, britain	439	24-oct-2011	100.0
english, canada	439	24-oct-2011	(Seed) 100.0
english, england001	439	24-oct-2011	100.0
arabic, morocco	422	23-sep-2011	100.0
cantonese, hong_kong	406	08-sep-2011	100.0
english, uk	436	19-oct-2011	100.0
english, south_vietnam	427	27-sep-2011	99.9
french, morocco	422	23-sep-2011	99.9
greek, turkey	430	07-oct-2011	99.9

Never-ending Language Learning (NELL, CMU),

`http://rtw.ml.cmu.edu/rtw/kbbrowser/`

Cas d'étude : Extraction d'information Web

Google squared labs

comedy movies | Square it | Add

comedy movies

	Item Name	Language	Director	Release Date
	The Mask	English	Chuck Russell	29 July 1994
	Scary M			
	Superba			
	Music			
	Knocked			

English
language for the mask
www.infibeam.com - all 9 sources »

Other possible values

- **English Language** Low confidence
  language for Mask
  www.freebase.com
- **english, french** Low confidence
  languages for the mask
  www.dvdreview.com
- **Italian Language** Low confidence
  language for The Mask
  www.freebase.com

Search for more values »

Chuck Russell
directed by for The Mask
www.infibeam.com - all 9 sources »

Other possible values

- **John R. Dilworth** Low confidence
  director for The Mask
  www.freebase.com
- **Fiorella Infascelli** Low confidence
  directed by for The Mask
  www.freebase.com - all 2 sources »
- **Charles Russell** Low confidence
  directed by for The Mask
  www.freebase.com - all 2 sources »

Search for more values »

Google Squared (terminé), capture d'écran de [Fink et al., 2011]

Cas d'étude : Extraction d'information Web

Sujet	Prédicat	Objet	Confiance
Elvis Presley	diedOnDate	1977-08-16	97.91%
Elvis Presley	isMarriedTo	Priscilla Presley	97.29%
Elvis Presley	influences	Carlo Wolff	96.25%

YAGO, `http://www.mpi-inf.mpg.de/yago-naga/yago`

Incertitude dans l'extraction d'information Web

- Le système d'extraction d'information est imprécis
- Le système a une certaine confiance dans l'information extraite, qui peut être :
  - une probabilité que l'information soit vraie (p. ex., champs aléatoires conditionnels)
  - un score de confiance numérique ad-hoc
  - un niveau discret de confiance (faible, moyen, haut)
- Et si cette information incertaine n'est pas quelque chose de final, mais est utilisée comme une source de données, p. ex., dans un système d'interrogation ?

Différents types d'incertitude

Deux dimensions

- Différent types :
  - Valeur inconnue : NULL dans les SGBD
  - Alternative entre plusieurs possibilités : soit A soit B soit C
  - Imprécision sur une valeur numérique : un capteur donne une valeur qui est une approximation de la valeur réelle
  - Confiance dans un fait dans son ensemble : cf. extraction d'information
  - Incertitude structurelle : le schéma des données lui-même est incertain
- Incertitude qualitative (NULL) ou quantitative (95%, faible confidence, etc.)

Systèmes

Trio `http://infolab.stanford.edu/trio/`, calcule à la fois la véracité et la lignée des données

MayBMS `http://maybms.sourceforge.net/`, SGBD relationnel probabiliste complet au-dessus de PostgreSQL, utilisable pour des applications pratiques.

Voir les séances sur gestion d'incertitude, MayBMS pour plus de détails

Plan

Volume

Variété

Vélocité

Véracité

Conclusion

Conclusion

- Les 4 v de la datamasse sont des défis pour le traitement des données du Web
- INF344 couvre :
  - Acquisition et enrichissement de données Web (crawl, extraction d'informations, recherche d'informations, ranking Web, fouille d'opinions).
  - La modélisation et le raisonnement sur les données Web (Web sémantique, données probabilistes).
  - Le stockage et le calcul sur les données Web (MapReduce, HBase).

Bibliography I

Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach, Mike Burrows, Tushar Chandra, Andrew Fikes, and Robert E. Gruber. Bigtable : A Distributed Storage System for Structured Data. In *Intl. Symp. on Operating System Design and Implementation (OSDI)*, 2006.

Robert Fink, Andrew Hogue, Dan Olteanu, and Swaroop Rath. SPROUT^2 : a squared query engine for uncertain web data. In *SIGMOD*, 2011.

David R. Karger, Eric Lehman, Frank Thomson Leighton, Rina Panigrahy, Matthew S. Levine, and Daniel Lewin. Consistent Hashing and Random Trees : Distributed Caching Protocols for Relieving Hot Spots on the World Wide Web. In *Proc. ACM SIGACT Symp. on the Theory of Computing (STOC)*, pages 654–663, 1997.

Wallace Koehler. A longitudinal study of web pages continued : a consideration of document persistence. *Inf. Res.*, 9(2), 2003.

Bibliography II

Julien Masanés. Web archiving methods and approaches : A comparative study. *Library Trends*, 54 :72–90, 2005. doi : 10.1353/lib.2006.0005.

Twitter. Historical data not working. `https://dev.twitter.com/discussions/2483`, 2011.

Licence de droits d'usage

Contexte public } avec modifications