Alignement d'ontologies

Emmanuel Coquery

Contexte

- Intégration de plusieurs bases de connaissances
  - A partir de sa propre ontologie, permettre de réutiliser l'information disponible depuis une ontologie externe
  - Permettre la traduction de concepts lors de l'interaction avec un autre système

Peu de sources, beaucoup d'objets

- Faire dialoguer des systèmes d'information de différentes organisations
  - Santé: hôpitaux entre eux, avec le système de la CPAM, etc
  - Industries: mise en correspondance client-fournisseur sur les données techniques
- Enrichissement de données: utiliser les données ouvertes et liées (Linked Open Data) pour enrichir des données internes

Nombreuses sources peu volumineuses

- Réseaux de connaissances en pair-à-pair
  - Chaque utilisateur possède quelques connaissances
    - Avec son vocabulaire propre
- Cherche à utiliser les connaissances du réseau pour répondre à une question

A-Box et T-Box

A-Box = instances

T-Box = classes

- Pas tout à fait les mêmes objectifs
  - A-Box: identifier les instances entre elles (=)
  - T-Box: relations plus souples [Euzenat2008]:

$$\{\subseteq, \supseteq, (), \perp\}$$

A-Box

- Étant donné deux instances dans une A-Box
  - quelle chance (probabilité) qu'elles représentent le même objet réel
  - autrement dit: un lien sameAs a-t-il du sens ?
- Comparer les attributs
  - distance basée sur un vecteur des comparaisons de chaque attribut
    - ex: [score-nom, score-prenom, score-date-naissance, score-ville-naissance]
  - score: booléen ou valeur numérique (distance, probabilité d'être les mêmes)

Quelques techniques de comparaison de valeurs textuelles

- Distance d'édition, e.g. Levenstein
- ngram: nombre de sous-chaînes communes de taille n
- techniques basées sur le traitement du langage: e.g. lemmatisation

Comparer les vecteurs de comparaison

- Moyenne pondérée des score des attributs

- Règles logiques ad-hoc:
  - Si le nom est le même à 0,8 et le prénom à 0,9 et que la date de naissance est la même (à 1,0) alors on considère que c'est la même personne.

- Difficile d'écrire l'ensemble des règles

- Possibilité d'apprentissage via e.g. le machine learning

Utiliser des contraintes

- La correspondance est transitive:
  - Si A ↔ B et B ↔ C alors A ↔ B
- La correspondance peut être exclusive (moins vrai dans les ontologies car monde ouvert):
  - Il y a au plus 1 A qui correspond à 1 B
- Utilisation de dépendances fonctionnelles pour certaines propriétés:
  - Si P est fonctionnelle:
    si A P OA, si B P OB et A ↔ B alors OA ↔ OB

Déductions négatives

- Transitivité: Si A ↔ B et B ↮ C alors A ↔ C

- DF:

  - Si P est fonctionnelle:
    si A P OA, si B P OB et OA ↮ OB alors A ↮ B

T-Box

- Faire correspondre les classes / concepts des ontologies
  - Pas toujours de correspondance exacte
  - $\subseteq, \supseteq$: une classe est plus spécifique/générale
  - $\not(), \perp$: intersection non vide ou vide
- Même chose pour les propriétés

Techniques par extension

- Utiliser l'alignement des instances
- Inférer les correspondances entre classes à partir des ensembles d'instances
  - on peu déduire $=, \subseteq, \supseteq, \not(), \perp$ entre classes
- Moins clair pour les propriétés

Utilisation des axiomes pour déduire des nouvelles correspondances

- Etant données certaines correspondances
- Et les axiomes définissant en intention certaines classes
- Poser la question au niveau de la T-Box union:

$$C_1 \stackrel{?}{\subseteq} C_2$$

Composition d’alignements

$$A \leftrightarrow B \leftrightarrow C$$

Composition

	=	>	<	◊	⊥
=	=	>	<	◊	⊥
>	>	>	>=<◊	>◊	>◊⊥
<	<	Γ	<	<◊⊥	⊥
◊	◊	>◊⊥	<◊	Γ	>◊⊥
⊥	⊥	⊥	<◊⊥	<◊⊥	Γ

[Euzenat2008]

Alignements incohérents

- Trouver une instance commune à deux classes disjointes (ou une instance d'une classe vide)

- Violer des contraintes comme la fonctionnalité (au sens DF) d'une propriété

- Problème particulièrement présent dans les scénario pair à pair

    - Nécessité de travailler avec une sous partie cohérente de l'union des ontologies

Alignements de grandes ontologies

- Calcul de distance pour alignement de base coûteux: quadratique dans le nombre d'entités à comparer

⇒ Découper les instances en des ensembles plus petits et comparer au sein de ces ensembles (blocking)

- Identifier une clé de blocking (compliqué !)
    - Utiliser les classes
    - Hash (doit être compatible avec la notion de distance)
    - Ngrams
- Possibilité de distribuer une entité sur plusieurs noeuds de calcul
    - Communiquer les correspondances trouver aux autres noeuds

Références

- [Euzenat2008] Jérôme Euzenat. Algebras of ontology alignment relations. Amit Sheth, Steffen Staab, Mike Dean, Massimo Paolucci, Diana Maynard, Timothy Finin, Krishnaprasad Thirunarayan. Proc. 7th international semantic web conference (ISWC), Oct 2008, Karlsruhe, Germany. Springer Verlag, 5318, pp.387-402, 2008, Lecture notes in computer science. <10.1007/978-3-540-88564- 1 25>. <hal-00793543>

- [ES2009] Jérôme Euzenat, Pavel Shvaiko. Ontology Matching Tutorial

- [GM2012] Lise Getoor, Ashwin Machanavajjhala. Entity Resolution: Tutorial