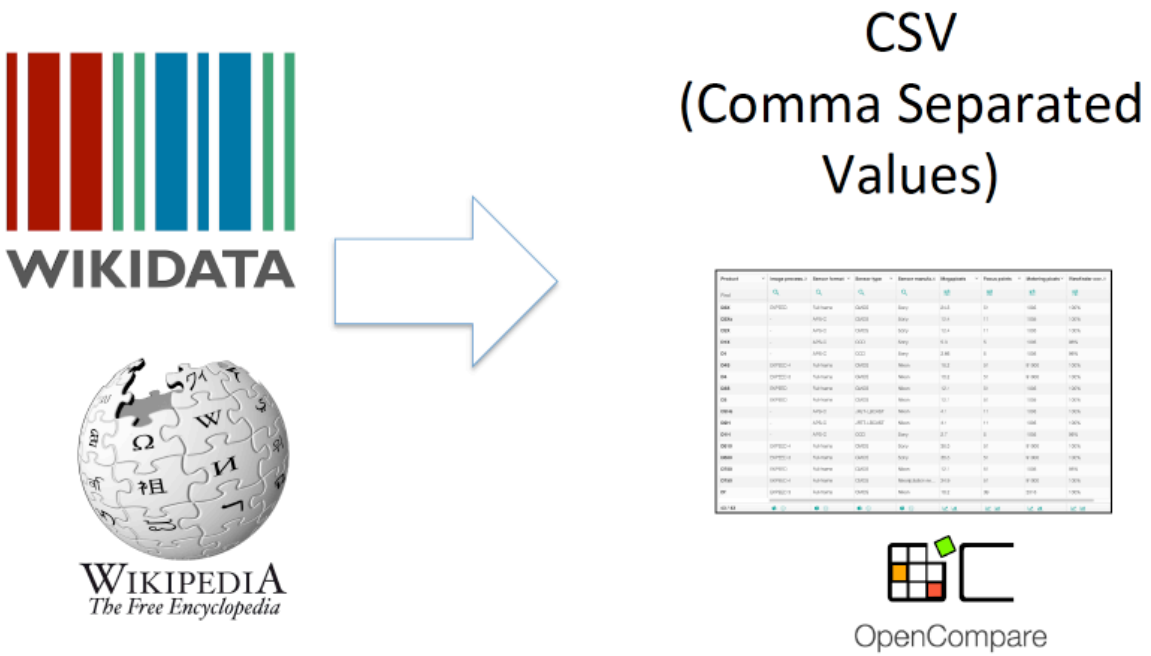
MIAGE 1
Projet Développement Logiciel –
Responsable : Mathieu Acher
Contact : mathieu.acher@irisa.fr
2017/2018

Gestion de projet – Projet Développement Logiciel (PDL)

Projet #3: « MatrixSynthesizerWikipedia »

Ressources :
http://mathieuacher.com/teaching/PDL/

Wikipedia est une fantastique source de données, principalement composée d'articles écrits en langage naturel (e.g., français, anglais). Wikidata est une initiative plutôt récente, complémentaire, qui vise à davantage structurer l'information.



L'objectif de ce projet de PDL est de créer des matrices de comparaisons (au format CSV) à partir de Wikipedia et/ou Wikidata. Ces matrices seront ensuite visualisées et exploitées sur opencompare qui supporte l'import CSV.

Par exemple, on voudrait comparer des « pays » (France, USA, Espagne, Chine, etc.). On pourrait alors fabriquer automatiquement une matrice avec comme « features » le nombre d'habitants, la superficie, le PIB, le drapeau, la langue principale pratiquée, etc. L'information sur le nombre d'habitants, le PIB, etc. ne sera pas rentré à la main, mais par l'exploitation des données de Wikipedia ou Wikidata. L'identification même des « features » sera réalisée automatiquement. Le tout sera écrit en Java.

Evidemment, il n'y a pas de sens à comparer des « choses » qui n'ont pas de points communs (e.g., comparer des appareils photos avec des acteurs). Aussi nous ferons l'hypothèse qu'un utilisateur fournit en entrée de votre procédure une liste de noms de produits (par exemple : « France, Argentine, Russie, Espagne »).

Wikipedia contient énormément de données (du texte, des figures, des sections, etc.) On pourrait considérer le texte, mais il est écrit en langage naturel, et il est par exemple difficile d'extraire une information précise comme le PIB ou le nombre d'habitants. Aussi, on s'intéressera uniquement aux « infobox » de Wikipedia :

https://fr.wikipedia.org/wiki/Aide:Infobox

Cette information est bien mieux structurée et se prête donc à une extraction automatique.

Pour Wikidata, l'information est dores et déjà structurée. Il faut par contre maîtriser l'API et la manière de traverser les différentes propriétés d'une « entité ».

En plus d'une solution technologique (en Java) pour produire des matrices de comparaison à partir de Wikipedia, l'objectif de ce projet est aussi d'explorer la pertinence de l'idée : obtient-on des matrices de qualité ? est-ce utile de fabriquer des matrices de comparaison à partir de Wikipedia ?

Vous devez adresser ces deux questions en utilisant votre solution sur plusieurs exemples concrets. Vous devez ainsi démontrer les aspects positifs et la plus value de votre solution mais aussi les limites actuelles, qu'elles soient liées à la qualité des données de Wikipedia/Wikidata, à la difficulté de regrouper des produits similaires, ou à la qualité de votre solution. L'évaluation de votre approche comptera pour 50% de la note : ce n'est donc pas qu'un travail d'implémentation.

L'implémentation complète et de bout en bout n'est pas triviale. Aussi il est fortement conseiller :

- De ne considérer que Wikipedia ou Wikidata dans un premier temps
- D'obtenir rapidement une solution « basique » qui étant donné un nom de produit (et un seul) retourne les « features » intéressantes ainsi que les valeurs

associés. A noter que le résultat sera finalement un CSV avec un seul produit.

- D'étendre votre solution pour supporter non plus un seul produit en entrée, mais plusieurs. Vous pouvez alors songer à implémenter des techniques pour classer la pertinence des « features » (e.g., certaines « features » ne sont pas très intéressantes, car peu de produits ont l'information associée dans Wikipedia ou Wikidata).
- De supporter Wikipedia et/ou Wikidata uniquement lorsque vous avez au moins une des deux sources de données qui fonctionne. A minima on souhaite pouvoir choisir entre l'un ou l'autre pour construire la matrice.
- Certains noms de produits méritent d'être « désambiguer ». Par exemple, le nom de produit « Python » est ambiguë car ce peut-être le langage de programmation ou le serpent. A minima fournir les moyens pour l'utilisateur de préciser quelle ressource Wikipedia / Wikidata est la plus pertinente. En bonus, vous pouvez désambiguer automatiquement (à l'aide du contexte des autres produits).
- La *combinaison* de Wikipedia et de Wikidata est en « bonus ». Un cas particulièrement intéressant concerne le possible « conflit » entre les informations de Wikipedia et Wikidata (e.g., la population du pays n'est pas la même sur Wikipedia que sur Wikidata)

Une attention particulière devra également être portée à la conception de l'API Java pour que votre solution soit la plus réutilisable et paramétrisable possible.

Comment commencer ?

Etudier Wikipedia/Wikidata et notamment les APIs qui sont proposés. Prototyper une première transformation « simple » (cf ci-dessus). Itérer.

Tester. Evaluer la pertinence de votre solution. Itérer, tester.

En parallèle de vos expérimentations, écrire le document de spécification.