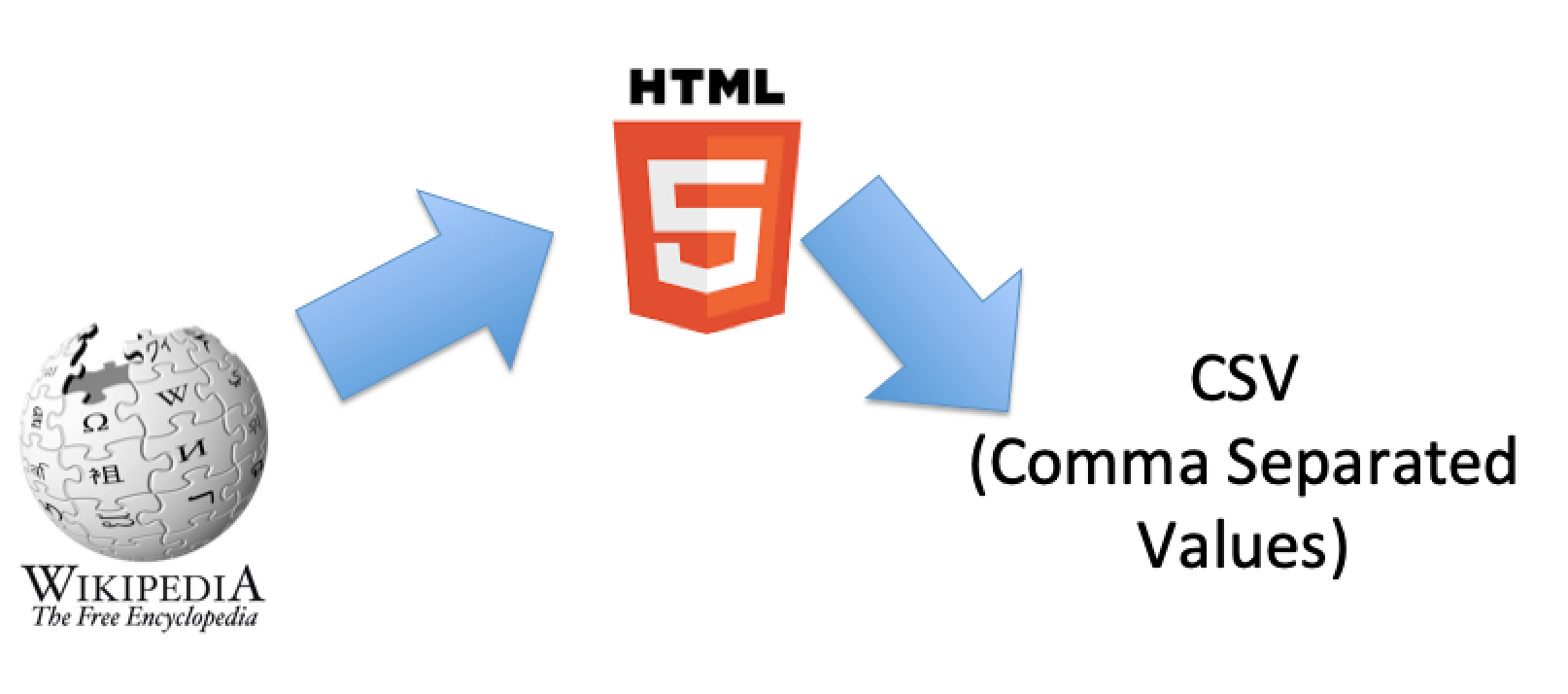
|  |  |  |
| --- | --- | --- |
|  | Développement Logiciel –  Responsable : Mathieu Acher  Contact : [mathieu.acher@irisa.fr](mailto:mathieu.acher@irisa.fr)  2019/2020 | Description: isticFR2012.jpg |

**Projet: « Wikipedia Matrix»**

Wikipedia est une fantastique source de données, principalement composée d’articles écrits en langage naturel (e.g., français, anglais).



L’objectif de ce projet est d’extraire des tableaux au format CSV à partir de pages Wikipedia.

**Pourquoi extraire des tableaux dans Wikipedia?**

La motivation générale est que les tableaux Wikipedia sont difficiles à exploiter par des outils statistiques, de visualisation ou n’importe quel outil capable d’exploiter les tableaux (e.g., Excel, OpenOffice, RStudio, Jupyter). Ces tableaux sont en effet écrits dans une syntaxe (Wikitext) difficile à analyser et non nécessairement conçue pour la spécification de tableaux. De plus, il y a une forte hétérogénéité dans la manière d’écrire des tableaux, ce qui complique encore plus le traitement des données tabulaires de Wikipedia. Le même constat peut être fait pour le format HTML qui peut être utilisé pour présenter un tableau dans un navigateur Web: il n’est pas facilement exploitable par des outils statistiques ou des tableurs. L’objectif est donc d’extraire les tableaux Wikipedia et de les traduire dans un format plus simple et adapté. De manière générale, l’extraction de données est une activité difficile et pourtant cruciale.

Le choix de CSV <https://fr.wikipedia.org/wiki/Comma-separated_values> a été effectué car il a le mérite d’être très simple et il est supporté par de nombreux outils.

La principale difficulté du projet sera de développer une procédure ***robuste et la plus générale possible.*** Pour évaluer les qualités de votre solution, un concours sera organisé : l’objectif sera d’extraire le plus possible de tableaux Wikipedia (et évidemment l’extraction devra produire des fichiers CSV bien formés et corrects).

**Quelques défis**

Wikipedia contient énormément de données (du texte, des figures, des sections, etc.) qui ne sont pas des données tabulaires. Il est même possible qu’une page Wikipedia ne contienne tout simplement pas de données tabulaires. Un défi sera donc d’occulter ce genre d’informations et ne considérer que les données pertinentes pour l’extraction de tableau. Il faudra aussi être en mesure d’extraire plusieurs tableaux sur une même page Wikipedia.

Une autre difficulté est que certaines données tabulaires sont difficiles à convertir au format CSV (par exemple, les tableaux imbriqués). Une barrière supplémentaire concerne l’hétérogénéité des données tabulaires qui complique la tâche d’extraction.

La conception et l’implémentation d’un algorithme général et robuste nécessitent de nombreux essais et erreurs sur différents cas réels. Il faudra veiller, lors de l’évolution de votre solution, à ne pas casser ce qui marchait précédemment (test de non régression). Typiquement, vous devez vérifier que certaines extractions qui fonctionnaient sur certaines pages Wikipedia sont toujours à même d’y parvenir.

**Quelques contraintes supplémentaires**

Une page Wikipedia peut être analysée de deux manières différentes :

* En allant chercher le code Wikitext correspondant
* En exploitant le rendu HTML de la page Wikipedia

Pour ce projet, il est demandé d’effectuer l’extraction via HTML.

**Auto-évaluation de votre approche**

En plus d’une solution technologique (en Java) pour extraire des données tabulaires 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, à la difficulté d’extraire des données tabulaires, ou à la qualité de votre solution.

**Comment commencer ?**

* “Fork” de <https://github.com/acherm/wikipediamatrix-bench>
  + inscription sur github
  + mise en place technique avec git, github, fork, git clone, et un commit simple
* exécuter et étudier la classe BenchTest
  + maven (pom.xml), JUnit, etc.

L’objectif long terme est d’intégrer votre extracteur dans BenchTest et ainsi démontrer la robustesse et l’applicabilité de votre solution… autrement dit **à la fin du projet, un “mvn test” doit permettre de lancer votre extracteur sur l’ensemble d’URLs proposé.**

Avant d’y parvenir, on se propose d’attaquer le problème avec un seul exemple et une seule page Wikipedia. Aussi, à l’intérieur du projet cloné, ajouter des classes pour:

* Implémenter l’extraction via HTML en utilisant jsoup <https://jsoup.org/> (utiliser Maven pour ajouter les dépendances) ; obtenir rapidement une solution « basique » qui étant donné une et une seule URL d’une page Wikipedia produit un CSV sur cet exemple : <https://en.wikipedia.org/wiki/Comparison_of_Canon_EOS_digital_cameras>
* Tester automatiquement (avec JUnit) votre solution « basique »
  + Vérifier que le CSV est bien valide[[1]](#footnote-0) en utilisant le parser CSV : <https://commons.apache.org/proper/commons-csv/apidocs/org/apache/commons/csv/CSVParser.html> (utiliser Maven pour ajouter les dépendances: <https://commons.apache.org/proper/commons-csv/>)
    - Lien intéressant: <https://www.callicoder.com/java-read-write-csv-file-apache-commons-csv/>
  + Ecrire des assertions sur le nombre de lignes et colonnes attendues
  + Ecrire des assertions sur les noms de colonnes attendues

Logging (pas de System.out.println!): <https://logging.apache.org/>

Ensuite, il s’agit d’étendre votre solution pour supporter non plus une seule URL mais plusieurs. Il faut ainsi expérimenter dans le large, sur plusieurs cas possibles (cf la liste des URLs dans le projet cloné). Vous devez songer à implémenter des techniques de test « génériques » (eg pour toute URL, les CSVs sont valides)

**Evaluation**

Rendu sous la forme d’un projet Github comprenant:

* le code source (incluant les tests)
* un README.md documentant le projet incluant:
  + à quoi sert votre logiciel?
  + license
  + comment exécuter le projet (Maven?)
  + architecture (diag. UML)
* un rapport sur les résultats obtenus:
  + combien de tableaux ont été extraits?
  + combien de colonnes, de lignes, et de cellules sont présentes dans les tableaux extraits? (min/max/mean/std)
  + quels sont les noms de colonnes les plus fréquents?
  + qualité et faiblesse de votre extracteur
  + synthèse générale: est-ce que les tableaux de Wikipedia que vous avez extrait sont exploitables par des outils statistiques?

Date limite: 20 décembre 2019

[mathieu.acher@irisa.fr](mailto:mathieu.acher@irisa.fr)

1. Extrait de Wikipedia: “CSV is a delimited text file that uses a comma to separate values (many implementations of CSV import/export tools allow other separators to be used). Simple CSV implementations may prohibit field values that contain a comma or other special characters such as newlines. More sophisticated CSV implementations permit them, often by requiring " (double quote) characters around values that contain reserved characters (such as commas, double quotes, or less commonly, newlines). Embedded double quote characters may then be represented by a pair of consecutive double quotes, [13] or by prefixing a double quote with an escape character such as a backslash (for example in Sybase Central).” [↑](#footnote-ref-0)