Projet innovant

Liens sémantiques

M1 Informatique
Université Paris 8
2021-2022

Céline Lecomte

1 Introduction

Ce projet innovant a pour but la création de liens sémantiques entre textes.
Il s'agit de trouver des liens sémantiques qui peuvent relier des passages ressemblants entre textes.

Je vais vous présenter ici les fonctionnalités développées pour ce projet.
Il permet de comparer 2 textes ensemble et de déterminer si ces 2 textes ont des mots en commun.

2 Solutions proposées

Je vais vous présenter ici les solutions proposées pour la création de liens sémantiques entre textes.

2.1 Téléversement des textes

L'utilisateur a la possibilité de choisir les textes qu'il souhaite comparer et de les téléverser.
Par un formulaire method="post" enctype="multipart/form-data".

2.2 Indexation des textes

function lire($texte, $separateurs)
function compter($source)
function indexer($name, $path)
function insertSource($langue, $nbMots, $source)
function insertIndexation($mot, $occurence, $source)

Les textes choisis par l'utilisateur sont indexés.
Les informations sont stockées dans 2 tables : source et indexation
Voici les étapes de cette indexation.

2.2.1 Normalisation du texte

Les mots sont transformés en minuscule grâce à la fonction strtolower().

Sont exclus de l'indexation les mots faisant moins de 3 lettres. strlen()
On considère que les mots dont la taille est inférieure à 3 lettres n'apportent pas de sens (comme les articles par exemple).
De même les "mots vides" sont également exclus de l'indexation. Car les "mots vides" ne sont pas porteurs d'information. Exemple de mots vides : article, adverbe, verbes être et avoir.

2.2.2 Définition de la langue du texte

$motVideFR et $motVideUK sont 2 tableaux contenant la liste des mots vides pour les langues française et anglaise.

Lors de la comparaison (in_array()) des mots du texte avec ces tableaux de mots vides, définition de la langue.
Si un mot est trouvé dans $motVideFR, alors la langue est FR.
Si un mot est trouvé dans $motVideUK, alors la langue est UK.
Sinon la langue est "Autre".

Le cas est prévu où FR et UK soient trouvés dans le même texte. Dans ce cas, c'est la langue dénombré le + grand nombre de fois qui est considéré comme la langue du document. array_count_values()

2.2.3 Calcul du nombre de mots du texte
Dénombrement du nombre de mots du texte. Via un compteur.

2.2.4 Calcul de la fréquence des mots du texte
Dénombrement de la fréquence des mots du texte.
array_count_values()

2.2.5 Indexation
Les informations de l'indexation sont stockées dans 2 tables : source et indexation.

Table source
```
langue ENUM('FR', 'UK', 'Autre'),
nbMots INT,
source VARCHAR(100)
```

Table indexation
```
mot VARCHAR(100),
occurence INT,
source VARCHAR(100)
```

2.3 Affichage des textes disponibles à la comparaison
function explorerDir($path)
Lecture récursive du répertoire où sont stockés les textes téléversés.
Affichage de ces textes sous forme de checkbox.

2.4 Comparaison des textes

Si l'utilisateur choisit, via la checkbox, 2 textes, la comparaison des 2 textes commence.
Le cas est prévu où l'utilisateur choisirait + ou – de 2 textes.

La comparaison se déroule en 2 étapes.

2.4.1 Comparaison de la langue des textes

function getSource($texte)
La table source contient les langues des textes.
strcmp()
SSI les textes sont dans la même langue on passe à l'étape 2.
Sinon $diagnostiqueFinal : Pas de lien sémantique.

2.4.2 Comparaison des 5 mots les + fréquents des textes

function getIndexation($texte)
SSI les textes sont dans la même langue comparaison des 5 mots les + fréquents des textes.
La table indexation contient les mots des textes et leurs fréquences.

```php
$indexationTA = getIndexation($texteA);
$indexationTB = getIndexation($texteB);
foreach($indexationTA as $key => $row) {
    $mot = $row['mot'];
    $tabMotsA[] = $mot;
}
foreach($indexationTB as $key => $row) {
    $mot = $row['mot'];
    $tabMotsB[] = $mot;
}

$compteur = 0;
foreach($tabMotsA as $motA) {
    if (in_array($motA, $tabMotsB)) {
        $tabMotsCommuns[] = $motA;
        $compteur++;
    }
}
echo "Sur les 5 mots les + fréquents des 2 textes, il y a $compteur mots communs trouvés <br>";

if($compteur>0){
    echo "Mots communs trouvés : <br>";
    foreach($tabMotsCommuns as $motCommun) {
        echo $motCommun;
    }
    $implodeMotsCommuns = implode(",", $tabMotsCommuns);
    $insertLienSemantique = insertLienSemantique($texteA, $texteB, $implodeMotsCommuns);
    $diagnostiqueFinal = 'Liens sémantiques trouvés. <br>';
}
else{
    echo "Les 2 textes n'ont pas de mot en commun. On ne peut donc pas créer de lien sémantique. <br>";
    $diagnostiqueFinal = 'Pas de lien sémantique trouvé. <br>';
}
```

SSI $compteur>0, insertion dans la table lienSemantique des liens sémantiques trouvés.
$diagnostiqueFinal : Liens sémantiques trouvés.

Sinon $diagnostiqueFinal : Pas de lien sémantique.

2.5 Création de liens sémantiques
SSI $compteur>0, insertion dans la table lienSemantique des liens sémantiques trouvés.

function insertLienSemantique($sourceA, $sourceB, $motsCommuns)

Table lienSemantique
sourceA VARCHAR(100),
sourceB VARCHAR(100),
motsCommuns TEXT

3 Conclusion

Ce projet innovant m'a permis d'aborder la création de liens sémantiques entre 2 textes.
J'ai pu proposer une application qui permet de téléverser les documents.
De rechercher des informations sur ces textes (langue du document, nombre de mots du document, calcul de la fréquence des mots du document).
Et de comparer 2 documents entre eux.
Afin de trouver (ou pas) des mots communs entre les 2 documents.

Ce travail est une première étape.
Il pourrait bien sûr être amélioré et complété. Ce qui permettrait une application + aboutie et + complète.

4 Bibliographie

https://www.php.net/
https://www.mysql.com/fr/
https://sql.sh/

5 Annexes

Voir captures d' écran de l'application créée dans le document joint.