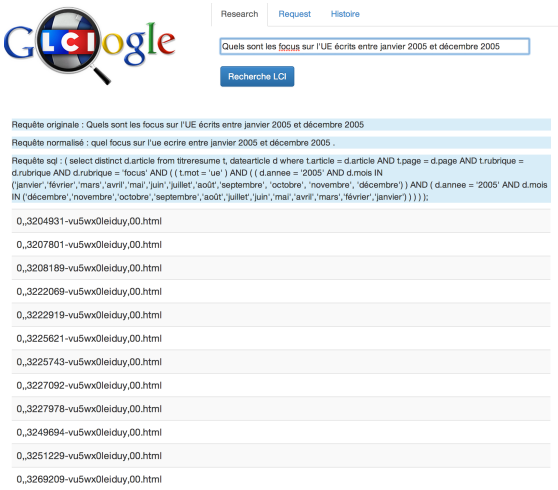
Vous voulez savoir qu'est-ce qui se passe pour le pape en 2005 ? Vous voulez trouver des nouvelles du **président** des Etats-Unis?

Why not try GoogleLCI??

c'est peut-être la meilleure façon de trouver des nouvelles sur LCI.



GoogleLCI est:

Un moteur de recherche dédié pour le site LCI.

Il a archivé tous les pages de LCI news depuis **25/02/2005** jusqu'a **02/03/2006**. les éléments indexés sont titre résumé, les rubriques comme **Une**, **Focus** et **voir aussi**, **les gros titres**, **les rappels**.

Comment utiliser le moteur de recherche?

Vous pouvez **taper votre requête en langage naturel** qui destine de trouver des informations mentionné ci-dessus et laisser GoogleLCI pour faire toutes les choses restante, même si vous faites des **erreurs orthographiques** sur votre requête, le moteur de recherche va corriger des erreurs en utilisant des **correcteurs orthographiques**.

Comment trouver des résultats?

Les résultats sont affichés en bas au centre de la page. Pour les requêtes qui demandent de compter le nombre de nouvelle, un nombre va afficher directement sur la page, sinon,
tous les résultats concernant la requêtes va être affiches sur la page. et si aucun information a été trouvée par le moteur de recherche, une notification va sortir pour informer l'utilisateur.Si vous cherchez des pages, les résultats sont données avec **un lien cliquable** sur le page pour y accéder.
La requête **originale, normalisée et SQL** générée sont aussi affichés sur la page.

Quelles sont les fonctionnalités?

1. **Chercher des informations générales.**
2. **Chercher des pages/articles ou rubriques entre certains date.**
3. **Compter le nombre de page/article sur des sujets.**
4. **Chercher des articles par le nom(email) de auteur.**

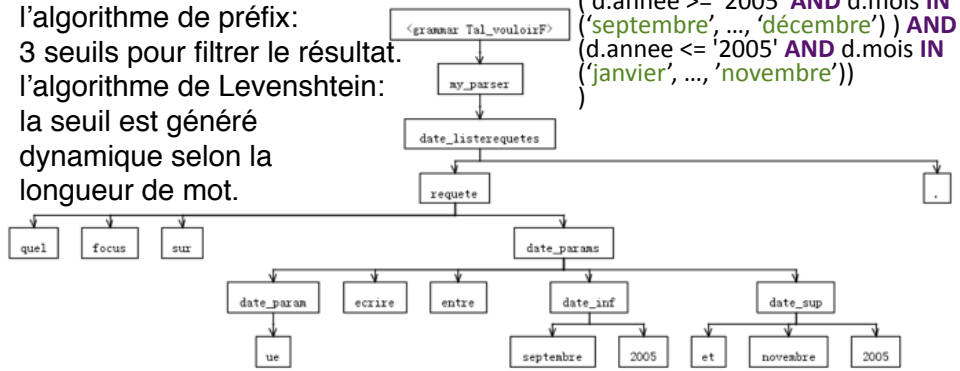
Notre système comporte 3 parties:

Une interface Web qui contient des HTML, CSS et javascript.
Un Servlet qui s'occupe de recevoir des requêtes de l'utilisateur et générer des résultats.
Une base de donnée qui stocke les tableaux inverses en format de tableaux SQL pour le site LCI.

Technique

D'aboard, on utilise des scripts **perl** pour extraire des informations depuis des pages LCI à l'aide des **expressions régulières**, les grouper dans plusieurs catégories et éventuellement générer un seul fichier structuré.**Ensuite**, on utilise des **commandes bash** et des scripts perl pour **diviser** des titre, resume, etc depuis fichier structuré ver les **termes,** ensuite. on calcule des occurrences de chaque terme afin de calcule son **poids rare**. et éventuellement, on utilise ce pois pour générer le **stop-list. Après,** on calcule des lemmes en utilisant l'**algorithme de calcul des successeurs** et on utilise cette liste de lemme pour filtrer notre corpus en **remplaçant les termes par son lemme**. **Enfin**, on crée la **liste inverse** a partir de la liste de mot filtrer par stop-list et par lemme.

Un correcteur orthographique nous permet de corrigés des erreurs ci-dessus, nous utilisons deux type de correcteurs : correcteur **par préfix** et correcteur **levenshtein**.
Type de d'erreur : manqué des lettres à la fin ou n'importe où du mot, ajouté des lettres supplémentaire, permuté deux lettres.
l'algorithme de préfix:
3 seuils pour filtrer le résultat.
l'algorithme de Levenshtein:
la seuil est généré dynamique selon la longueur de mot.




```
Saisie : je vuex les article traitent proc ordinateur
Corrections : O:je  L:(vue;veux) <les> O:article
O:traitent P:(proche;procès) N:ordinateur
```

Quels sont les focus sur l'UE érits entre septembre 2005 et novembre 2005

```
SELECT DISTINCT  d.article
FROM titreresume t, datearticle d
WHERE
t.article = d.article AND t.page =
d.page AND t.rubrique = d.rubrique
AND d.rubrique='focus' AND
(
  (t.mot = 'ue' ) AND
( d.annee >= '2005' AND d.mois IN
('septembre', ..., 'décembre') ) AND
(d.annee <= '2005' AND d.mois IN
('janvier', ..., 'novembre'))
)
```

l'API java.sql nous permet de intérroger différents type de base de données en utilisant la méthode **executeQuery(requete),** les résultats sont sous forme d'objet **ResultSet,** et pour récupérer les noms de colonnes dans les résultats, on utilise **ResultSetMetaData.**

Tomcat est un conteneur web de servlets et JSP. Afin de permettre des utilisateur d'**interroger la base de données via web,** on a créé notre propres **Servlet** qui hérite la classe **HttpServlet,** et il va traiter les **requête GET** depuis l'utilisateur, extraire les requêtes SQL, l'exécute sur la base de données en utilisant l' API ci-dessus, et convertit les résultats au format HTML, et les met dans la HTTP réponse (HttpServletResponse)

Préparation et Indexation du Corpus

Dans cette partie, car les nombre de pages à traiter sont considérable, il faut être sur que chaque étape sont bien passé avant de commencer dans l'étape suivant, du coup, **un processus de vérification** dans chaque étape est essentielle.
une autre chose à réfléchir c'est l**e choix du seuil** pour la génération de stop-list, car cette étape est totalement subjectif, du coup, il faut réfléchir soigneusement pour fixer le seuil.

Correcteur orthographique

Pendent le test, on a trouver que les deux correcteurs orthographique **ne sont pas suffisant** pour corriger certains type d'erreur, il faut implémenter d'autre type de correcteur, par exemple, le correcteur de Norvig, pour corriger des erreur ou des fautes de frappe.

Analyse Syntaxique

Pour cette partie, on a cherché les informations en donnant la date, l'email de auteur ou le mot clé sur le corpus, on peut aussi compter le nombre d'article concernant les informations cherchées, mais il existe encore d'autre grammaire qui n'est pas réalisée, du coup , il y a **des limités** sur notre moteur de recherche.

Interrogation d'une base de données

Nous travaillons directement sur une base de donnée déjà créé, et il y a des structures de donnée qu'on trouve déraisonnable, par exemple, les dates sont stocké séparément, et le mois est stocké en français, il introduit des travaux supplémentaires pour générer des requête SQL dans l'étape précédent.

Analyse critique

utc Université de Technologie Compiègne
HU Kewei / WANG Yujue