LETTRES SORBONNE UNIVERSITÉ

Master 2 Langue et Informatique
NOM : XIE
Prénom : Shilin
M3SOL034
N°étudiant : 21102552

Rapport du Projet SOLR

Dans l'ère actuelle de l'information, les moteurs de recherche jouent un rôle indispensable, occupant une place de plus en plus cruciale dans la vie quotidienne des utilisateurs. Leur développement ne représente pas seulement une avancée technologique, mais constitue également une composante essentielle de la société de l'information. Grâce à leur capacité à répondre rapidement aux besoins des utilisateurs, à leurs algorithmes de recherche intelligents et à une couverture réseau étendue, les moteurs de recherche favorisent constamment la facilité d'accès et la démocratisation de l'obtention d'informations. Ces outils puissants offrent aux utilisateurs une expérience de recherche d'informations efficace et précise, devenant ainsi le principal moyen pour les individus d'obtenir les informations nécessaires.

Dans le cadre de ce projet, nous avons choisi d'utiliser Solr comme moteur de recherche. Solr est un serveur de recherche plein texte à haute performance développé en Java, basé sur Lucene. Grâce à ce moteur de recherche robuste, nous pouvons facilement créer localement un moteur de recherche flexible et personnalisable, stockant différents types de fichiers et ajustant les poids en fonction des besoins pour fournir des résultats de recherche plus précis. Cela nous permet de tirer pleinement parti des fonctionnalités de Solr, en construisant un système de recherche d'informations puissant et efficace, répondant aux divers besoins des utilisateurs dans un environnement d'information complexe.

Le corpus de ce rapport de projet consiste en un enregistrement détaillé des informations sur les films de TMDB, comprenant les 10 000 premiers films jusqu'au 26 juillet 2022. Les données sont présentées au format CSV et incluent neuf colonnes clés :

- ID : Identifiant unique de chaque film sur le site TMDB, offrant un suivi exclusif des films.
- title : Nom du film, constituant une identification fondamentale.
- genre : Genre du film, englobant plusieurs catégories telles que le crime, l'aventure.
- original_language : Langue originale utilisée lors de la publication du film.
- overview : Résumé succinct de chaque film.
- popularity : Popularité du film, reflétant l'intérêt global des spectateurs.
- release_date : Date de sortie du film.
- vote_average : Note moyenne du film, fournissant une évaluation quantitative de la qualité du film.
- vote_count : Nombre de votes du film, reflétant la participation à l'évaluation du film.

Il est important de noter que ce jeu de données ne comprend pas de champs de type booléen. Pour compenser cette lacune, une nouvelle colonne nommée "recommended" a été introduite. Dans cette colonne, si la valeur de vote_average d'un film est supérieure à 8, la valeur correspondante est définie comme TRUE.

Après avoir téléchargé le fichier SOLR, utilisez le terminal cd jusqu'à l'adresse du fichier bin dans le fichier solr et entrez *./solr start*. Nous pouvons utiliser l'URL *http://localhost:8983/solr/#/* pour utiliser ce serveur. Utilisez la commande *./solr create -c Movies* pour créer la collection SOLR de ce projet. Ce corpus est un fichier csv qui peut être téléchargé directement dans la collection sur la

page du serveur : cliquez sur la collection *Movies* que je viens de créer sur la page, recherche Documents, cliquez dessus, sélectionnez le téléchargement de fichier du type de document, puis sélectionnez le fichier à télécharger et la soumission du document est réussie. Bien sûr, nous pouvons également télécharger le fichier depuis le terminal via la commande *bin/post -c Movies example/exampledocs/Movies.csv.*

Une fois le téléchargement du fichier terminé, j'entre dans la phase de création du schéma d'indexation de Solr. C'est une étape cruciale qui garantit une gestion efficace des données dans le projet.

Voici la démarche que nous avons suivie :
Identification des Champs Essentiels :
J'ai déterminé les champs essentiels qui seront indexés par Solr pour assurer une représentation complète des données cinématographiques. Ces champs correspondent aux 10 colonnes présentes dans le fichier CSV.
Garantie d'une Indexation Cohérente et Précise :
J'ai veillé à ce que la structure du schéma permette une indexation cohérente et précise des données. Chaque champ a été défini avec soin pour s'assurer qu'il correspond aux types de données appropriés, garantissant ainsi une recherche efficace.
En définissant ce schéma d'indexation, j'ai posé les bases nécessaires pour permettre à Solr de gérer et d'interroger les données cinématographiques de manière optimale.

Pour ce faire, accéder à serveur > solr > Films > conf sur la machine sur laquelle se trouve le dossier SOLR et recherchez le fichier de *managed-schema*. Dans le fichier, ajoutez les éléments suivants pour définir le type approprié pour chaque champ :

```
<field name="genre" type="text_general" indexed="true" stored="true"/>
<field name="id" type="string" multiValued="false" indexed="true" required="true" stored="true"/>
<field name="original_language" type="string" indexed="true" stored="true"/>
<field name="overview" type="text_general" indexed="true" stored="true"/>
<field name="popularity" type="pdoubles"  indexed="true" stored="true"/>
<field name="recommended" type="booleans" indexed="true" stored="true"/>
<field name="release_date" type="pdate" indexed="true" stored="true"/>
<field name="title" type="text_general" indexed="true" stored="true"/>
<field name="vote_average" type="pdoubles" indexed="true" stored="true"/>
<field name="vote_count" type="plongs" indexed="true" stored="true"/>
```

Ensuite, rechercher *solrconfig.xml* dans le dossier conf, ouvrez-le et ajoutez-le à la ligne 1131 :

```
<schemaFactory class="ManagedIndexSchemaFactory">
  <bool name="mutable">true</bool>
  <str name="managedSchemaResourceName">managed-schema</str>
</schemaFactory>
```

Ouvrir le terminal et entrer la commande ./solr restart pour redémarrer le serveur. A ce moment, ouvrez la collection et cliquez sur le schéma pour voir les champions que vous venez de créer.

Afin de rendre le moteur de recherche plus convivial, l'étape suivante consiste à copier le dossier *Velocity* dans conf et à continuer d'ajouter le code pertinent dans *solrconfig.xml* :

```
<lib dir="${solr.install.dir:../../../..}/contrib/velocity/lib" regex=".*\.jar" />
<lib dir="${solr.install.dir:../../../..}/dist/" regex="solr-velocity-\d.*\.jar" />

<queryResponseWriter name="velocity" class="solr.VelocityResponseWriter" startup="lazy">
  <str name="template.base.dir">${velocity.template.base.dir:}</str>
</queryResponseWriter>

<requestHandler name="/browse" class="solr.SearchHandler">…
```

Tout d'abord, modifiez le poids de la recherche. Dans ce corpus, j'ai choisi le poids du titre du film à 10, le poids du résumé du film à 5 et le poids du type de film à 2.

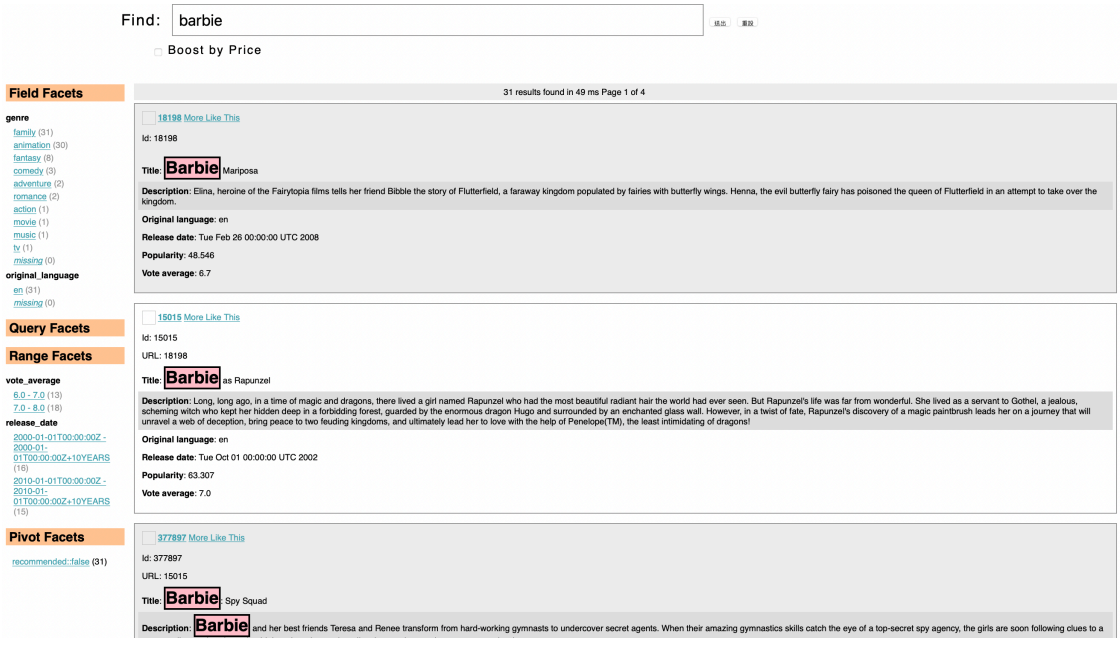
Ensuite, en ajoutant le code suivant dans la balise requestHandler de *solrconfig.xml*, j'ai créé cinq facettes (2 facettes de champ, y compris le genre et la langue d'origine, 2 facettes d'intervalle, y compris la note moyenne et la date de publication, et une facette pivot, ce qui est recommandé).

Pour le réaliser, rechercher le fichier *richtext_doc.vm* dans le dossier *Velocity*, ouvrez et écrivez le contenu du corpus qui doit être affiché dans la page de recherche, et définissez la police et le format souhaités dans le fichier main.css.

Redémarrer le serveur et utiliser le lien *http://localhost:8983/solr/Movies/browse* pour voir une page de recherche concise et claire.

Bientôt, nous sommes arrivés à la dernière étape : la mise en évidence des termes de recherche. Ce qui doit être modifié, ce sont les fichiers *solrconfig.xml* et main.css. Ajoutez le style de balisage pour les termes de recherche dans le fichier main.css, écrivez la classe personnalisée dans le fichier css dans *hl.simple.pre* dans Mise en évidence des valeurs par défaut dans *solrconfig.xml*, puis ajoutez les paramètres que vous devez utiliser, comme le mien. Le projet se concentre principalement sur la récupération de contenu dans les parties titre et présentation, c'est pourquoi les points forts tombent également ici.

Redémarrer maintenant le serveur et ouvrez la page du navigateur. Entrer des mots-clés tels que : *barbie* et nous verrons que les caractères pertinents dans le titre et la description sont spécialement marqués.



Le serveur Solr dans ce projet fournit une solution fiable pour la gestion et la récupération des données cinématographiques. En définissant des schémas et des facettes appropriés, un moteur de recherche flexible et efficace est créé, offrant une bonne expérience de recherche. L'ensemble du processus ne demande pas beaucoup d'efforts et est très flexible. Il peut être utilisé sur divers ensembles de données avec un grand nombre de données et d'attributs. C'est un outil très pratique.