



Université Mohammed V - Rabat
École Nationale Supérieure
d'Informatique et d'Analyse
des Systèmes



Rapport

FILIÈRE

Business Intelligence & Analytics

SUJET :

Moteur de Recherche sur le Saint Quran

Réalisé par :

Fatima Zahra BELHAMRA

Ilham TIRACH

Oumaima DGUIGOU

Safaa LHAKKAOUI

Zaïneb EL HASSANI

Encadré par :

Pr. Nouredine KERZAZI

Introduction générale

Dans le cadre de ce projet, nous sommes chargés d'évaluer et d'améliorer un moteur de recherche sur le saint Coran, développé par nos collègues. Ce moteur de recherche vise à faciliter l'accès et la consultation de textes sacrés, y compris le Coran, le Tafsir et d'autres documents tels que le Hadith. Pour cela, le code source complet est mis à notre disposition, permettant une exploration approfondie des mécanismes de traitement, de recherche et d'analyse des textes religieux. L'objectif est non seulement de tester et valider le déploiement de la solution avec Flask, mais aussi d'examiner la fonction de scoring pour comprendre son impact sur les résultats. En parallèle, nous testerons des analyzers alternatifs, définirons des stopwords pertinents, et proposerons des améliorations techniques et fonctionnelles pour optimiser la performance de la solution.

Chapitre 1

1.1 Architecture de la Solution Proposée

L'architecture de la solution pour la création d'un moteur de recherche syntaxique et sémantique pour le Coran repose sur une série d'étapes bien définies, comme illustré dans la figure ci-dessous. Cette architecture est structurée en quatre principales phases : *Source des données*, *Traitement et Nettoyage*, *Indexation*, et *Déploiement*. Voici une description détaillée de chaque phase :

1.1.1 Source des données

Cette étape regroupe l'acquisition des données à partir de différentes sources :

- **Fichiers JSON :** Incluent des données brutes telles que les sourates (`surahs.json`) et les versets (`verses.json`).
- **Base de données SQL Server :** Contient des informations enrichies sur le Coran, y compris des tafsirs, accessibles via des requêtes SQL.
- **Fichiers CSV :** Incluent d'autres types de données brutes comme les Hadiths (`hadiths.csv`).

1.1.2 Traitement et Nettoyage

Une fois les données collectées, elles passent par une étape de prétraitement pour garantir leur qualité et leur uniformité :

- Les scripts Python sont utilisés pour charger, nettoyer et structurer les données (gestion des encodages, suppression des doublons, correction des anomalies).
- Les données sont consolidées en un fichier JSON final, prêtes pour l'indexation.

1.1.3 Indexation

Dans cette étape, Elasticsearch est utilisé pour indexer les données. Chaque document est enrichi de métadonnées supplémentaires pour optimiser les capacités de recherche. Les index sont conçus pour prendre en charge des recherches syntaxiques et sémantiques.

1.1.4 Déploiement

Enfin, une interface utilisateur est développée à l'aide de Flask pour permettre aux utilisateurs d'accéder aux fonctionnalités du moteur de recherche. L'application Web permet une interaction intuitive avec le corpus indexé.

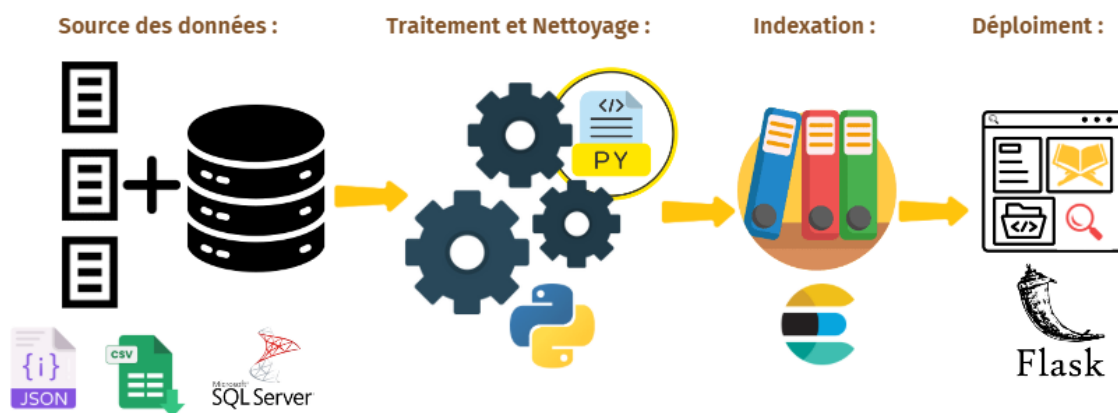


FIG. 1.1 : Vue d'ensemble de l'architecture technique

1.2 Justification des Choix Technologiques

Les choix technologiques adoptés dans cette solution répondent à des critères de performance, de scalabilité et d'adéquation aux besoins spécifiques du projet. Voici une justification détaillée des outils utilisés :

1.2.1 Python

Python est choisi pour le traitement des données en raison de sa richesse en bibliothèques adaptées à l'analyse de données (`pandas`, `json`, `codecs`). Sa flexibilité et sa large communauté en font un outil incontournable pour manipuler des corpus complexes comme le Coran.

1.2.2 Elasticsearch

Elasticsearch est utilisé comme moteur d'indexation et de recherche pour ses fonctionnalités avancées :

-
- Recherche rapide et précise grâce à un moteur basé sur Apache Lucene.
 - Support des index vectoriels pour intégrer des recherches sémantiques avec des modèles d'apprentissage automatique.
 - Facilité d'intégration avec des outils tiers pour améliorer la pertinence des résultats.

1.2.3 Flask

Pour le déploiement, Flask est choisi pour sa simplicité et sa flexibilité en développement d'applications Web. Il permet de concevoir une interface utilisateur légère, adaptée à des projets de petite ou moyenne taille.

1.2.4 SQL Server

SQL Server est utilisé comme source de données relationnelles en raison de sa robustesse et de son intégration facile avec Python (`pyodbc`). Cela facilite l'extraction et la transformation des données critiques telles que les tafsirs.

1.2.5 Formats JSON et CSV

Les formats JSON et CSV sont privilégiés pour leur simplicité et leur compatibilité avec les outils de traitement et d'indexation. Ils permettent de structurer efficacement des données hétérogènes.

En résumé, ces choix technologiques assurent une implémentation optimale de la solution tout en permettant une évolutivité pour de futures améliorations, telles que l'intégration de recherches sémantiques avec des modèles de type BERT.

1.3 Phase de Préprocessing

1.3.1 Chargement des Données

- Importation des fichiers JSON avec l'encodage `utf-8-sig`.
- Connexion à la base SQL pour récupérer les tafsirs depuis la table `tafsir1`.
- Lecture des données depuis `hadiths.csv`.

1.3.2 Prétraitement des Données du Coran

- Construction d'une structure de données unifiée incluant :
 - Indices des versets.
 - Noms des sourates et leurs indices.

-
- Textes des versets (avec et sans diacritiques).
 - Tafsirs associés.
 - Sauvegarde dans le fichier `quran.json` avec les clés suivantes : `aya_index_quran`, `aya_index_sura`, `sura_name`, `sura_index`, `total_verses`, `aya_chakl`, `aya_no_chakl`, et `tafsir_aya`.

1.3.3 Prétraitement des Données des Hadiths

- Conversion du fichier nettoyé `hadith_processed.csv` vers `json_hadith.json` à l'aide de la fonction `csv_to_json`.
- Justification : Format JSON choisi pour sa facilité de stockage et d'intégration.

1.3.4 Nettoyage des Données

- Suppression des colonnes inutiles : `id`, `hadith_id`, `chain_indx`.
- Remplacement des noms des sources en anglais par leur version arabe à l'aide de `replace` avec `regex=True`.
- Élimination des caractères indésirables (HTML, Unicode) dans `text_ar` et `text_en`.
- Gestion des doublons : Identification et suppression des doublons dans `text_en`.
- Réorganisation des colonnes sélectionnées : `source`, `source_english`, `chapter_no`, `chapter`, `text_ar`, et `text_en`.

1.3.5 Fichiers de Sortie

- **`quran.json`** : Données structurées du Coran, incluant les tafsirs.
- **`json_hadith.json`** : Hadiths nettoyés, prêts pour des analyses ultérieures.

Mapping et Création d'Index

Dans cette section, nous décrivons le processus de mapping et de création d'index pour les données du saint Coran et du Hadith, dans le cadre de notre projet de moteur de recherche. Nous avons travaillé avec plusieurs versions d'index pour le Coran et le Hadith, permettant une analyse comparative de leurs performances et de leur pertinence.

1.4 Préparation à elasticsearch

Avant l'indexation, les fichiers JSON bruts ont été transformés pour :

- Extraire les champs pertinents pour les Hadiths (source, chapitre, texte en arabe et en anglais) et le Coran (index des sourates, versets, texte avec et sans voyelles, et tafsir).
- Renommer les champs pour correspondre aux besoins du projet.
- Créer une nouvelle structure de données plus adaptée aux traitements Elasticsearch.

Structure Générale du Code de Mapping

Le code de mapping dans Elasticsearch est structuré en deux grandes sections :

- **Section `settings`** : Cette section contient les paramètres de configuration, tels que :
 - **Les paramètres de base** :
 - * `number_of_shards` : Définit le nombre de partitions logiques pour l'index.
 - * `number_of_replicas` : Spécifie le nombre de copies de chaque shard pour assurer la redondance.
 - **Analyse linguistique** :
 - * **Filtres** : Définissent des étapes spécifiques comme la suppression des mots vides (`stopwords`), la normalisation, le lemmatisation (*stemming*) et la création de n-grams.
 - * **Analyseurs** : Assemblages de filtres réutilisables pour analyser les textes en fonction des besoins.
- **Section `mappings`** : Cette section définit la structure des données en spécifiant :
 - **Les types des champs** : (`text`, `integer`, `keyword`, etc.).
 - **L'utilisation des analyseurs linguistiques** : Les champs textuels peuvent être analysés avec les analyseurs définis dans `settings`.

```
mapping = {  
  "settings" : {  
    "number_of_shards":1,  
    "number_of_replicas":1,  

```

FIG. 1.2 : Paramètres de base

```

"analysis":{
  "filter":{
    "arabic_stop":{
      "type":"stop",
      "stopwords_path":"merged-stopwords.txt" #
    },
    "arabic_keywords":{
      "type":"keyword_marker",
      "keywords_path":"keywordmarker-words.txt"
    },
    "stop_complex_queries":{
      "type":"stop",
      "stopwords_path":"merged-stopwords-1.txt"
    },
    "arabic_stemmer":{
      "type":"stemmer",
      "language":"arabic"
    },
    "arabic_shingle":{
      "type":"shingle",
      "min_shingle_size":2,
      "max_shingle_size":3
    }
  },
},

```

FIG. 1.3 : Filtres

```

"analyzer":{
  "rebuilt_arabic":{
    "tokenizer":"standard",
    "filter":[
      "arabic_stop",
      "arabic_normalization",
      "arabic_keywords",
      "arabic_shingle",
      "arabic_stemmer"
    ]
  },
  "stem_noshingle":{
    "tokenizer":"standard",
    "filter":[
      "arabic_stop",
      "arabic_normalization",
      "arabic_keywords",
      "arabic_stemmer" #for one word aggregations
    ]
  }
}

```

FIG. 1.4 : Analyseurs


```
"mappings":{
  "properties":{
    "source_arabic":{
      "type":"text",
      "analyzer" : "simple",
      "fielddata" : "true",
      "fields":{
        "keyword":{
          "type":"keyword"
        }
      }
    },
    "source_english":{
      "type":"text",
      "fields":{
        "keyword":{
          "type":"keyword"
        }
      }
    },
    "chapter_no":{
      "type":"integer"
    }
  }
}
```

FIG. 1.5 : Mappings

1.5 Index et Mapping pour les Hadiths

1.5.1 Création des Index

Deux index ont été créés pour les Hadiths :

- `hadith_no_mapping` : Utilise le mapping par défaut d'Elasticsearch.
- `hadith_with_mapping` : Inclut un mapping personnalisé adapté aux champs des Hadiths et aux analyzers nécessaires.

1.5.2 Mapping Personnalisé

Pour l'index `hadith_with_mapping`, le mapping a été conçu pour :

- Définir des analyzers spécifiques, comme un analyzer arabe pour le texte `hadith_arabic`.
- Ajouter des sous-champs permettant le stemming, les recherches exactes, et les agrégations.
- Gérer les textes en anglais et en arabe de manière distincte.

Critère	Mapping Simple	Mapping Personnalisé
Paramètres de configuration	Basique : définit uniquement number_of_shards et number_of_replicas.	Avancé : inclut des paramètres pour l'analyse linguistique (analysis, filter, analyzer).
Analyse linguistique	Utilise des analyseurs généraux, comme "arabic".	Inclut des analyseurs et filtres personnalisés (stemming, stopwords, n-grammes, etc.).
Structure des champs	Chaque champ est défini simplement avec un type de base (text, keyword, long).	Les champs ont plusieurs sous-champs utilisant différents analyseurs pour divers besoins.
Flexibilité de recherche	Convient aux recherches basiques.	Optimisé pour des recherches complexes et adaptées à des langues ou corpus spécifiques (ex. : arabe).
Cas d'utilisation	Pour des applications simples ou généralistes.	Pour des cas nécessitant une analyse linguistique approfondie et des optimisations spécifiques.

FIG. 1.6 : La différence entre le mapping simple et le mapping personnalisé

1.5.3 Envoi des Données

Les données des Hadiths transformées ont été indexées à l'aide de la méthode `bulk`, garantissant une ingestion rapide et efficace.

1.5.4 Validation

Des requêtes, comme des recherches sur des mots spécifiques dans le champ `hadith_arabic`, ont été utilisées pour valider la pertinence du mapping et des analyzers.

```

• analyze_body = {
  "analyzer": "shingle_nostem", # Or whatever analyzer is applied to hadith_arabic.shingle_nostem
  "text": "الإيمان"
}
analysis = es.indices.analyze(index="hadith_with_mapping", body=analyze_body)
print(analysis)

{'tokens': [{'token': 'الإيمان', 'start_offset': 0, 'end_offset': 7, 'type': '<ALPHANUM>', 'position': 0}]}

```

FIG. 1.7 : Requêtage 1 pour la validation

```

'_shards': {'total': 1, 'successful': 1, 'skipped': 0, 'failed': 0},
'hits': {'total': {'value': 158, 'relation': 'eq'},
'max_score': 9.508631,
'hits': [{'_index': 'hadith_with_mapping',
'_type': '_doc',
'_id': 'mzQI-JIBhcI7irrYckT',
'_score': 9.508631,
'_source': {'source_arabic': 'سنن الترمذي',
'source_english': 'Jami' al-Tirmidhi ',
'chapter_no': '37',
'chapter': 'Chapters on the description of the Day of Judgement, Ar-Riqaq, and Al-Wara' - كتاب صفة القيامة والرقائق والورع عن رسول الله صلى الله عليه وسلم حتى يخبره من أى حلال الإيمان شاء بليغها هذا حديث حسن ومعنى قوله حلال الإيمان يعني ما يعطى أهل الإيمان من حلال الجنة',
'hadith_arabic': 'Sahl bin Muadh bin Anas Al-Juhani narrated from his father, that the Messenger of Allah (s.a.w)said: "whoever"',
'hadith_english': 'Sahl bin Muadh bin Anas Al-Juhani narrated from his father, that the Messenger of Allah (s.a.w)said: "whoever"',
'_index': 'hadith_with_mapping',
'_type': '_doc',
'_id': 'qTQI-JIBhcI7irrYTHz',
'_score': 9.508631,
'hits': [{"token": "الإيمان", "start_offset": 0, "end_offset": 7, "type": "<ALPHANUM>", "position": 0}]}]}]}

```

FIG. 1.8 : Résultat de la requête

1.6 Index et Mapping pour le Coran

1.6.1 Création des Index

Trois index distincts ont été créés pour le Coran :

- `quran_no_mapping` : Utilise le mapping par défaut d'Elasticsearch.
- `quran_old_mapping` : Utilise un mapping personnalisé de base pour des tests initiaux.
- `quran_final_mapping` : Utilise un mapping personnalisé final, optimisé pour les données et les requêtes sur le texte coranique.

1.6.2 Mapping Personnalisé

Pour l'index `quran_final_mapping`, le mapping a été conçu pour :

- Gérer des champs spécifiques comme `aya_chakl` (texte avec voyelles) et `aya_no_chakl` (texte sans voyelles).
- Inclure des analyzers arabes et des filtres tels que le stemming et les stopwords.
- Faciliter les recherches agrégées et les recherches précises dans les textes coraniques.

1.6.3 Envoi des Données

Les données coraniques, une fois transformées, ont également été indexées avec la méthode `bulk`.

1.6.4 Validation

Des requêtes de validation ont été effectuées, notamment sur des mots arabes spécifiques dans les champs `aya_chakl` et `aya_no_chakl`, pour s'assurer de la pertinence des analyzers et des résultats.

```
{
  'took': 16,
  'timed_out': False,
  '_shards': {
    'total': 1,
    'successful': 1,
    'skipped': 0,
    'failed': 0
  },
  'hits': {
    'total': {
      'value': 18,
      'relation': 'eq'
    },
    'max_score': 8.450897,
    'hits': [
      {
        '_index': 'quran_no_mapping',
        '_type': '_doc',
        '_id': 'aTQ9-ZIBhcI7irrYUer5',
        '_score': 8.450897,
        '_source': {
          'aya_index_quran': 4799,
          'aya_index_sura': 15,
          'sura_name': 'سورة النجم',
          'sura_index': 53,
          'total_verses': 62,
          'aya_chakl': 'عَنْدَمَا جِئْتُ أَلْمَأُؤَةَ',
          'aya_no_chakl': 'عندما جئة الماوى',
          'tafsir_aya': 'وقال مقاتل والكلبي : بأوي إليها أرواح الشهداء ( عندما جئة الماوى )'
        }
      }
    ]
  },
  '_index': 'quran_no_mapping',
  '_type': '_doc'
}
```

FIG. 1.10 : Résultat avec $quran_{no_mapping}$

```
{
  'took': 48,
  'timed_out': False,
  '_shards': {
    'total': 1,
    'successful': 1,
    'skipped': 0,
    'failed': 0
  },
  'hits': {
    'total': {
      'value': 166,
      'relation': 'eq'
    },
    'max_score': 6.0696745,
    'hits': [
      {
        '_index': 'quran_final_mapping',
        '_type': '_doc',
        '_id': 'yJ09-ZIBhcI7irrY9RdP',
        '_score': 6.0696745,
        '_source': {
          'aya_index_quran': 3946,
          'aya_index_sura': 158,
          'sura_name': 'سورة الصافات',
          'sura_index': 37,
          'total_verses': 182,
          'aya_chakl': 'وَجَعَلُوا بَيْنَهُ وَبَيْنَ الْجَنَّةِ نَهْجًا وَلَقَدْ عَلِمْتِ الْإِنْتِ إِهْمَ لَخُضْرُونَ',
          'aya_no_chakl': 'وجعلوا بينه وبين الجنة نسيبا ولقد علمت الجنة إهم لمحضرون',
          'tafsir_aya': 'ثم نزه نفسه عما قالوا فقال ( وجعلوا بينه وبين الجنة نسيبا )'
        }
      }
    ]
  },
  '_index': 'quran_final_mapping',
  '_type': '_doc'
}
```

FIG. 1.9 : Résultat avec $quran_{final_mapping}$

Indexes disponibles

Après validation, les indexes configurés et leurs caractéristiques respectives sont présentés ci-dessous :

<input type="checkbox"/> quran_final_mapping	● yellow	open	1	1	9735	89mb
<input type="checkbox"/> quran_no_mapping	● yellow	open	1	1	12470	22.3mb
<input type="checkbox"/> quran_old_mapping	● yellow	open	2	1	6235	11mb
<input type="checkbox"/> hadith_wiith_mapping	● yellow	open	1	1	33357	105.5mb
<input type="checkbox"/> hadith_noo_mapping	● yellow	open	1	1	33357	33.8mb

FIG. 1.11 : Mapping

- **quran_final_mapping** : Cet index contient le mapping final pour le Coran, avec un total de 9,735 documents, occupant un espace de 89 MB.
- **quran_no_mapping** : Index du Coran utilisant le mapping suggéré par Elasticsearch, contenant 12,470 documents pour un total de 22.3 MB.

-
- **quran_old_mapping** : Version ancienne de l'index du Coran avec un mapping précédent, comprenant 6,235 documents, pour un espace de 11 MB.
 - **hadith_with_mapping** : Index du Hadith avec notre mapping personnalisé, totalisant 33,357 documents et occupant un espace de 105.5 MB.
 - **hadith_no_mapping** : Index du Hadith utilisant le mapping suggéré par Elasticsearch, contenant également 33,357 documents mais avec un espace réduit à 33.8 MB.

1.6.5 Analyse des Mappings

Les mappings personnalisés jouent un rôle crucial dans l'optimisation des performances des requêtes et la précision des résultats. Cette section détaille les décisions techniques prises pour le mapping des indexes `hadith_with_mapping` et `quran_final_mapping`, ainsi que leur impact.

1.6.5.1 Mapping pour `hadith_with_mapping`

- ****Analyzer Arabe**** : Un analyzer personnalisé a été utilisé pour le champ `hadith_arabic`. Cet analyzer applique une normalisation spécifique pour gérer les variations linguistiques comme les formes d'accentuation (*tashkeel*), améliorant ainsi les résultats des recherches en arabe.
- ****Sous-champs**** :
 - **nostem** : Permet des recherches exactes en désactivant le stemming.
 - **stem** : Applique un stemming léger pour prendre en compte les dérivations des mots.
- ****Texte Bilingue**** : Le mapping distingue les champs anglais (`hadith_english`) et arabes (`hadith_arabic`) pour permettre des analyses linguistiques distinctes.

Impact : Ce mapping améliore la précision pour les requêtes sur des mots spécifiques en arabe tout en offrant une flexibilité pour les recherches multilingues.

1.6.5.2 Mapping pour `quran_final_mapping`

- ****Séparation des Champs**** :
 - **aya_chakl** : Champ avec les voyelles (*tashkeel*), utilisé pour les analyses linguistiques exactes.
 - **aya_no_chakl** : Champ sans voyelles, permettant des recherches plus générales et tolérantes.
- ****Tafsirs (Explications)**** : Inclusion du champ `tafsir_aya`, offrant une recherche sur le sens contextuel des versets.

- **Gestion des Index** :
 - Les index des sourates (**sura_index**) et des versets (**aya_index_quran**) sont configurés pour des recherches rapides et des agrégations.

1.6.5.3 Comparaison avec les Mappings par Défaut

Conclusion : L'utilisation de mappings personnalisés permet d'exploiter pleinement les capacités d'Elasticsearch pour des cas d'utilisation spécifiques, en particulier pour des langues complexes ou des corpus bilingues.

Pour la phase de déploiement, nous avons ajusté légèrement le design en présentant les résultats sous forme de tableau. Une colonne dédiée au scoring a été ajoutée, utilisant la fonction par défaut d'Elasticsearch, BM25 (Best Matching), nous l'avons déjà vu en cours. Et voici des exemples de requêtes :

Quran Verse Search Engine		
Search Query: "الجنة"		
Number of Ayat: 54		
Surah	Ayah	Score
سورة الصفات	وَجِبُوا أَن تَكُونَ ذُنُوبًا مِّنَ الْجَنَّةِ لَمَّا بَلَغَ الْهُدَىٰ لِمَ كُنْتُمْ تَكْفُرُونَ قَالَ مُجَاهِدٌ وَقَدْ رَأَىٰ آيَةَ الْجَنَّةِ لَمَّا تَنَزَّلَتْ عَلَىٰ السَّمَاءِ وَقَالَ ابْنُ (وَجِبُوا بَيْنَهُ وَبَيْنَ الْجَنَّةِ نَسِيًا) عَبَّاسٍ : حِينَ تَرَى الْمَلَائِكَةَ يَقُولُ لَهُمُ الْجَنَّةُ ، وَمَعَهُمُ الْإِيمَانُ ، هُمْ بَنَاتُ اللَّهِ ، وَقَالَ الْكَلْبِيُّ : أَتْلُوهُمُ اللَّهَ ، بَلْ تَزُوجُ مِنْ الْجَنِّ فَخَرَجَ مِنْهَا الْمَلَائِكَةُ تَعْلَمُ أَنَّ اللَّهَ عَنْ نَفْسِكَ ، وَقَدْ كَانَ زَعَمَ بَعْضُ قُرَيْشٍ أَنَّ الْمَلَائِكَةَ بَنَاتُ اللَّهِ تَعْلَمُ اللَّهُ ، هَذَا أَبُو بَكْرٍ الصَّدِيقُ : لَمَّا سَمِعَهُمْ ؟ قَالُوا : سُرَاتُ الدِّينِ وَقَالَ الْحَسَنُ : مَعْنَى النَّسَبِ أَنَّهُمْ أَشْرَكُوا بِالْإِلَهِاتِ فِي عِبَادَةِ اللَّهِ ، (وَلَقَدْ : عَلِمَتِ الْجَنَّةُ أَنَّهُمْ) يَعْنِي قَوْلًا هَذَا الْقَوْلُ (لِمَحْضُورُونَ) فِي الدَّارِ ، ثُمَّ زَعَمَتْ نَفْسُهُ صَاعًا قَالُوا قَالُوا	7.94
سورة الحشر	لَا يَسْتَوِي أَسْمَاءُ كَثِيرٌ وَأَسْمَاءُ ثَلَاثُ الْجَنَّةِ لَمَّا كُنْتُمْ كَافِرِينَ "لَا يَسْتَوِي أَسْمَاءُ كَثِيرٌ وَأَسْمَاءُ ثَلَاثُ الْجَنَّةِ لَمَّا كُنْتُمْ كَافِرِينَ"	7.94

Une recherche sur "Al-jannah" (paradis) dans le Coran a identifié 54 versets pertinents. Les résultats sont présentés dans un tableau avec les colonnes Sourate, Verset, et Score de pertinence (ex. : 7,94).

Quran Verse Search Engine		
Search Query: "المال"		
Number of Ayat: 4		
Surah	Ayah	Score
سورة الفجر	وَلْيُكُونَ لِلْمَالِ حَيْثُ جَاءَ وَلْيُكُونَ لِلْمَالِ حَيْثُ جَاءَ، أَي كَثِيرًا، يَحْيَى: تَحْيَوْنَ جَمْعَ الْمَالِ وَتَوَلَّوْنَ بِهِ، يَقَالُ: جَمَعَ الْمَالُ فِي الْحَوْضِ، إِذَا كَثُرَ وَاجْتَمَعَ	11.12
سورة التكوير	الْمَالِ وَالْبَنُونَ زِينَةُ الْحَيَاةِ الدُّنْيَا وَالْبَاقِيَاتُ الصَّالِحَاتُ خَيْرٌ عِنْدَ رَبِّكَ ثَوَابًا وَخَيْرٌ أَمَلًا التي يتقتر بها عبدة وأصحابه الأعداء (زينة الحياة الدنيا) ليست من زاد الآخرة قال علي بن أبي طالب (المال والبنون) رضى الله عنه : المال والبنون حوث الدنيا والأصل المسألة حوث الآخرة وقد يجمعها الله لأقوام (والباقيات الصالحات) اختلوا فيها قال ابن عباس وعكرمة ومجاهد : هي قول : سبحان الله والحمد لله ولا إله إلا الله والله أكبر . وقد روي أن النبي صلى الله عليه وسلم قال : " أفضل الكلام أربع كلمات : سبحان الله والحمد لله ولا إله إلا الله والله أكبر " . أخرنا أبو القاسم عبد الله بن محمد الحافظ أنبا أبو بكر محمد بن الحسن البحري أخرنا أبو جعفر عبد الله بن إسماعيل الهيثمي أنبا أحمد بن عبد الجبار المالازي أخرنا أبو معاذة عن الأعمش عن أبي صالح عن أبي هريرة رضى الله عنه قال : قال رسول الله صلى الله عليه وسلم : " لأن أقول سبحان الله والحمد لله ولا إله إلا الله والله أكبر أحب إلي مما طلعت عليه الشمس " . أخرنا	9.23

FIG. 1.13 : Quran verse search engine

Une recherche sur "Al-mal" (argent) dans le Coran a identifié 4 versets pertinents, présentés dans un tableau avec les colonnes Sourate, Verset, et Score de pertinence (ex. : Al-Fajr avec un score élevé de 11,12).

Requêtes Hadith

Hadith Search Engine		
Search Query: "الحياء"		
Number of Hadiths: 26		
Chapter	Hadith	Score
صحيح البخاري	حدثنا ابنه، حدثنا شعبه، عن قتادة، عن أبي السوار الهذلي، قال سمعت عمران بن حصين، قال قال النبي صلى الله عليه وسلم الحياء لا يأتي إلا بخير قال بشر بن كعب مكراب في الحكمة أن من الحياء وفاء، وأن من الحياء سقيفة قال له عمران أحكك عن رسول الله صلى الله عليه وسلم وتحدثني عن مسجدهم كتاب الأدب - (Al-Adab) - Good Manners and Form	13.07
سنن الترمذي	حدثنا ابن أبي عمير، وأحمد بن منيع، - المعنى واحد قال حدثنا سليمان بن عيينة، عن الأزهر، عن سلمة، عن أبيه، أن رسول الله صلى الله عليه وسلم مر برجل وهو يمشي في الحياء قال رسول الله صلى الله عليه وسلم الحياء من الإيمان قال أحمد بن منيع في حديثه أن النبي صلى الله عليه وسلم سمع رجلاً يمشي في الحياء قال هذا حديث حسن صحيح وفي الباب عن أبي هريرة وأبي ثوبة وأبي أمامة كتاب الإيمان عن رسول الله صلى الله عليه وسلم - The Book on Faith	12.76

FIG. 1.14 : Hadith search engine

Search Query: "الإيمان"		
Number of Hadiths: 158		
Chapter	Hadith	Score
سنة الترمذي	حدثنا عباس بن محمد الدوري، حدثنا عبد الله بن يزيد المقرئ، حدثنا سعيد بن أبي أيوب، عن أبي مرحوم عبد الرحيم بن محبوب، عن سهل بن معاذ بن أبي الجهم، عن أبيه، أن رسول الله صلى الله عليه وسلم قال من ترك اللباس أو أضعفه له وهو يكثر عليه دعاء الله يوم القيامة على رؤوس الخلائق حتى يخيره من أي حلال الإيمان شاء فليأخذه فذلك حسن ومعنى قوله حلال الإيمان يعني ما يعطى أهل الإيمان من حلال الجنة وإرفاقه بالخروج عن رسول الله صلى الله عليه وسلم	9.51
صحيح مسلم	حدثنا عبد الله بن سعيد، وعبد بن حميد، قال حدثنا أبو عمار الطقي، حدثنا سليمان بن بلال، عن عبد الله بن دينار، عن أبي صالح، عن أبي هريرة، عن النبي صلى الله عليه وسلم قال الإيمان بضع وسبعون شعبة والحياء شعبة من الإيمان	9.31
سنة الترمذي	أخبرنا محمد بن عبد الله بن المبارك، قال حدثنا أبو عمار، قال حدثنا سليمان بن بلال، عن عبد الله بن دينار، عن أبي صالح، عن أبي هريرة، عن النبي صلى الله عليه وسلم قال الإيمان بضع وسبعون شعبة والحياء شعبة من الإيمان	9.31

FIG. 1.15 : Search query

Une recherche sur "AL-Iman" (Foi) dans une base de données de hadiths a identifié 158 résultats. Ils sont classés par chapitre, texte du hadith, et score indiquant la pertinence ou l'authenticité.

La fonction de Scoring : BM25 - Best Matching 25

$$BM25(t, d) = \sum_{t \in q} IDF(t) \cdot \frac{f(t, d) \cdot (k_1 + 1)}{f(t, d) + k_1 \cdot (1 - b + b \cdot \frac{|d|}{avgdl})}$$

where:

- $f(t, d)$ is the frequency of t in d ,
- $|d|$ is the length of document d ,
- $avgdl$ is the average document length in the collection,
- k_1 and b are hyperparameters that control the weighting.

FIG. 1.16 : la fonction BM25_Scoring

Quran Verse Search Engine		
Number of Ayat: 54		
Surah	Ayah	Score
سورة الصفات	وَجَعَلُوا بَيْنَهُ وَبَيْنَ الْجَنَّةِ لُتْفًا وَلَقَدْ عَلِمْتُمُ الَّذِينَ يُؤْتِمِرُونَ قال مجاهد وقتادة : أراد بالجنة الملائكة ، نسوا جنة لاجتماعهم عن الأيسار . وقال ابن عباس : هي من الملائكة قال لهم لهم الجن ، ومنهم إبليس ، قالوا : هم بنات الله وقال الكلبي : (وجعلوا بينه وبين الجنة نسبا) قالوا - لعنهم الله - بل تزوج من الجن فخرج منها الملائكة تعالى الله عن ذلك ، وقد كان زعم بعض قريش أن الملائكة بنات الله تعالى الله . فقال أبو بكر الصديق : فمن أينهم ؟ قالوا : سورات الجن . وقال : الحسن : معنى النسب أنهم الشركوا الشياطين في عبادة الله ، (ولقد علمت الجنة أنهم) يعني قذافي هذا القول (لمحضون) في النار ، ثم نزه نفسه عما قالوا فقال	7.94
سورة الحشر	لَا يَسْتَوِي أَسْخَابُ النَّارِ وَأَسْخَابُ الْجَنَّةِ أَسْخَابُ الْجَنَّةِ هُمْ الْفَائِزُونَ "لا يستوي أصحاب النار وأصحاب الجنة أصحاب الجنة هم الفائزون"	7.94
سورة الشرح	وَأَزَلَّتِ الْجَنَّةُ لِلنَّارِ قرئت (الجنة للنار) (وأزلت)	7.46

FIG. 1.17 : TEST_scoring

- Elasticsearch utilise la fonction BM25 pour le scoring.
- BM25 prend en compte le nombre d'occurrences du mot (ou de la phrase) dans les documents.
- Comme illustré dans l'image ci-dessus, la requête est : "Al-Jannah."
 - Dans les deux premiers documents, ce mot apparaît 4 fois (exactement tel quel), ce qui leur donne le même score.
 - Dans le troisième document, il apparaît 2 fois, ce qui entraîne un score inférieur à celui des deux premiers.

Conclusion ELK

En conclusion, ce projet de moteur de recherche pour les textes du Coran et du Hadith constitue une première itération visant à optimiser la pertinence et l'accessibilité des résultats. Grâce aux mappings personnalisés et à une ingestion de données rapide, l'outil offre une expérience de recherche fluide et précise. Cette version initiale est un point de départ, avec des améliorations prévues dans les prochaines itérations pour enrichir encore davantage la performance et la convivialité de l'interface.

Chapitre 2

Système RAG et Développement du Modèle

2.1 Le Système Retrieval-Augmented Generation (RAG) : Concept et Application

Le système Retrieval-Augmented Generation (RAG) est une méthode avancée qui combine la récupération d'informations et la génération de texte, permettant de produire des réponses précises et enrichies à partir de données externes. Ce système est particulièrement adapté aux tâches de question-réponse, où les réponses reposent sur des informations actualisées et pertinentes.

Dans cette mise en œuvre, le processus s'appuie sur les composants suivants :

- **Récupération d'information** : Identification des documents ou passages pertinents dans un corpus, réalisée à l'aide de modèles d'embedding spécialisés pour la langue et le domaine.
- **Génération de texte** : Production de réponses contextuelles et cohérentes, en s'appuyant sur les informations récupérées et un modèle génératif puissant.

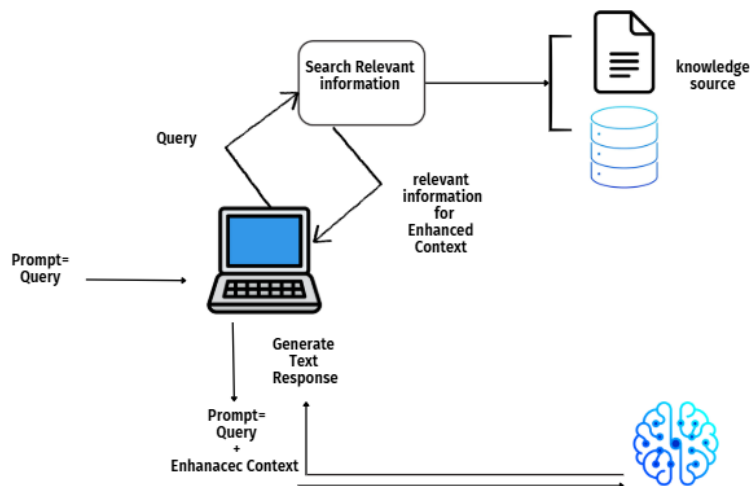


FIG. 2.1 : RAG

2.2 Composants du Système RAG

Le système est structuré autour de trois modules principaux :

- **Module de Récupération (Retrieval Module)**

Ce module recherche les documents les plus pertinents à partir d'une base de données vectorisée. Dans ce projet, **Chroma**, un système de stockage vectoriel performant, est utilisé pour gérer les vecteurs et effectuer des recherches efficaces.

- **Modèle d'Embedding**

Le rôle de ce module est de représenter les textes sous forme de vecteurs sémantiques. Deux modèles spécifiques, entraînés pour des applications en arabe, sont utilisés :

- `Omartificial-Intelligence-Space/Arabic-Triplet-Matryoshka-V2`
- `omarelshehy/Arabic-STs-Matryoshka-V2`

Ces modèles offrent des embeddings optimisés pour capturer les nuances sémantiques des textes en arabe, facilitant une récupération précise et pertinente.

- **Modèle Génératif (Generative Model)**

Pour la génération de texte, le système utilise le modèle **Qwen/Qwen2-1.5B-Instruct**, un modèle génératif avancé conçu pour comprendre et répondre de manière fluide et informative. Ce modèle, combiné avec les données récupérées, produit des réponses précises et contextualisées.

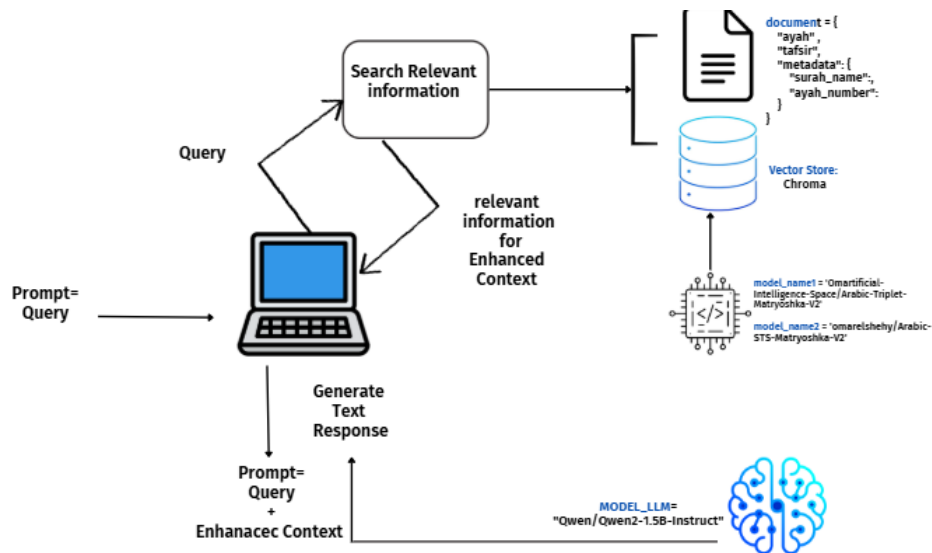


FIG. 2.2 : RAG réalisé

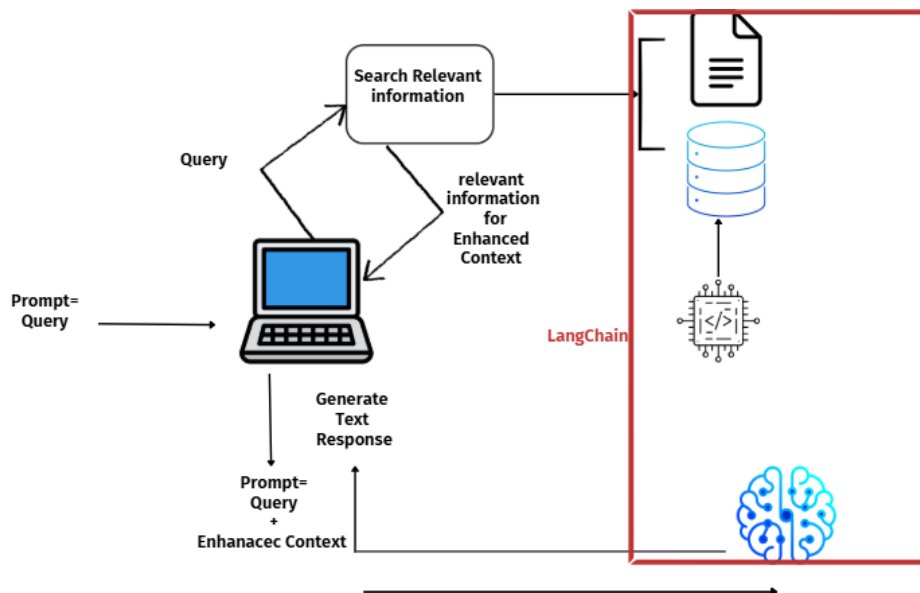


FIG. 2.3 : langchain

Cette architecture, associant des modèles spécialisés pour l'arabe et des outils avancés de stockage et de recherche vectorielle, démontre l'adaptabilité et la puissance du système RAG dans des contextes linguistiques et thématiques spécifiques.

2.3 Étapes de Prétraitement pour la Deuxième Partie

L'étape de prétraitement pour la deuxième partie se concentre sur la transition de la recherche basée sur la syntaxe avec ELK à la recherche sémantique, garantissant la cohérence des données et leur alignement avec les nouvelles exigences du projet.

2.3.1 Cohérence des Données

La structure du fichier `quran.json` est restée inchangée afin de maintenir la compatibilité avec les deux parties du projet.

2.3.2 Remplacement du Tafsir

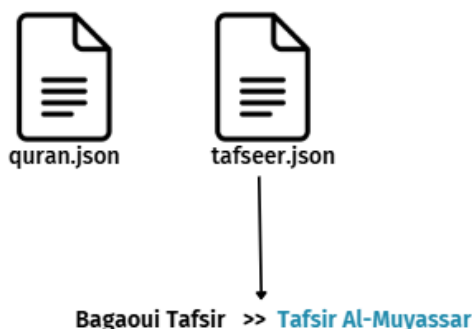


FIG. 2.4 : Tafasser

- Le Tafsir original (Bagaoui) a été remplacé par le Tafsir Al-Muyassar, un commentaire simplifié, pour mieux s'aligner avec l'approche de recherche sémantique.
- Les champs de métadonnées clés tels que `ayah` (numéro du verset) et `number` (numéro du chapitre) ont été préservés afin d'assurer la cohérence des données.

2.3.3 Transformation des Champs

Les transformations suivantes ont été appliquées aux données :

- Le champ `number` (représentant le numéro du chapitre) et le champ `ayah` (représentant le numéro du verset) ont été standardisés en type `int` pour améliorer le traitement et les opérations de recherche.

2.4 Modèle de Génération des Embeddings

Les embeddings jouent un rôle essentiel dans le système RAG, car ils permettent de convertir des documents ou des parties de texte en vecteurs de caractéristiques. Ces vecteurs sont ensuite utilisés pour comparer et retrouver les informations les plus pertinentes dans une base de données.

Le modèle utilisé pour générer ces embeddings est basé sur des architectures de réseaux neuronaux pré-entraînés, telles que BERT ou RoBERTa, qui sont adaptées à la tâche de génération de représentations vectorielles des documents.

$$\mathbf{E}_{doc} = f_{\text{embeddings}}(D) \quad (2.1)$$

Où \mathbf{E}_{doc} représente l'embedding d'un document D , et $f_{\text{embeddings}}$ est la fonction qui génère l'embedding à partir du texte.

Pour le choix des modèles d'embeddings nous avons utilisé **MTEB Leaderboard** (Massive Text Embedding Benchmark) pour sélectionner le modèle d'embeddings qui va être convenable pour la langue Arabe : [1]

Rank	Model	Model Size (Million Parameters)	Memory Usage (GB, fp32)	Embedding Dimensions	Max Tokens	Average	STS17 (ar-ar)	STS17 (en-ar)	STS17 (en-de)	STS17 (en-fr)
280	Arabic-STS-Matryoshka-V2	135	0.5	768	512		86.86	13.34		
388	sima_embedding-sts-v0.1	135	0.5	768	512		85.59	47.7	31.86	9.1
130	Arabic-Triplet-Matryoshka-V2	135	0.5	768	512		85.31			
50	Arabic_text_embedding_for_sts	135	0.5	768	512		85.05			
51	Arabic_text_embedding_sts_ar	135	0.5	768	512		84.96			

FIG. 2.5 : MTEB_EMBEDDING_Models_Arabic_STS

2.4.1 Arabic-STS-Matryoshka-V2

Le modèle **Arabic-STS-Matryoshka-V2**, basé sur le framework **SentenceTransformer**, est dérivé du modèle de base **aubmindlab/bert-base-arabertv02**. Ce modèle est spécialement conçu pour l'embedding sémantique des phrases et des paragraphes en arabe, en produisant des représentations vectorielles denses de dimension 768.

Principales caractéristiques

- **Architecture** : SentenceTransformer utilisant un modèle BERT.
- **Fonction de similarité** : Similarité cosinus pour comparer les embeddings.
- **Flexibilité des dimensions d'embedding** : Matryoshka embeddings permettant des tailles ajustables (768, 512, 256, 128, 64) afin d'optimiser la performance et la mémoire en fonction des besoins.
- **Longueur maximale de séquence** : 512 tokens.

Applications

Le modèle peut être utilisé dans des tâches variées telles que :

- La similarité textuelle sémantique (STS).
- La recherche sémantique.

- Le regroupement (*clustering*).

[2]

2.4.2 Arabic-Triplet-Matryoshka-V2

Ce modèle est basé sur **Sentence-Transformers** et est un affinement du modèle **aubmindlab/bert-base-arabertv02**. Il est conçu pour projeter des phrases et des paragraphes en un espace vectoriel dense à 768 dimensions. Ce modèle est adapté pour diverses tâches NLP, notamment :

- La similarité sémantique.
- La recherche sémantique.
- Le paraphrase mining.
- La classification de texte.
- Le clustering.

Particularités

- **Données d'entraînement** : Entraîné sur un jeu de données contenant 1M échantillons provenant de la base *akhooli/arabic-triplets-1m-curated-sims-len*.
- **Objectif d'entraînement** : Perte finale basée sur *MatryoshkaLoss* évaluée à 0.718 après 3 époques.
- **Dimensions des sorties** : 768 dimensions (troncation possible pour des tailles inférieures : 512, 256, 128, 64).
- **Optimisation des embeddings** : Utilisation d'embeddings *Matryoshka* pour équilibrer performances et mémoire.

Caractéristiques Techniques

- **Taille du modèle** : 135M paramètres.
- **Type de tenseur** : F32.
- **Évaluation des performances** : Sur la tâche *MTEB MintakaRetrieval (ar)*, les scores auto-reportés incluent un *ndcg_at_10* de 20.059.
- **Licence** : Apache 2.0.

[3]

Matryoshka Representation Learning

Les modèles d'embeddings que nous avons choisis repose sur une nouvelle approche : Matryoshka Representation Learning.

Matryoshka Representation Learning - MRL : c'est une approche d'apprentissage qui encode des informations à différentes granularités, permettant à un vecteur d'adaptation d'être utilisé pour des contraintes computationnelles variées.



FIG. 2.6 : Matryoshka doll

Architecture

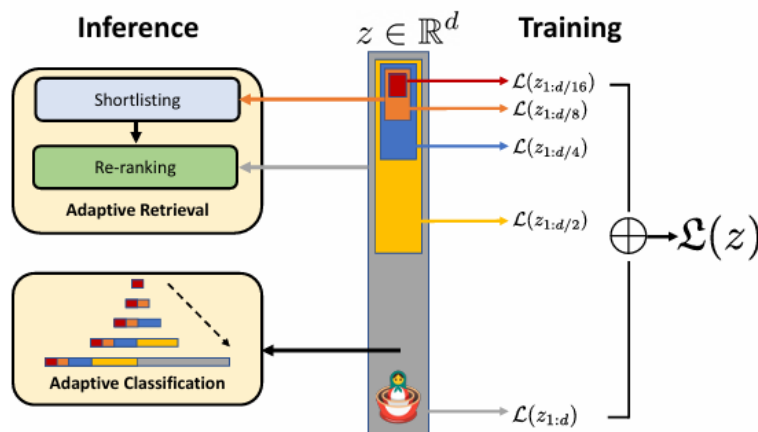


Figure 1: 🧸 Matryoshka Representation Learning is adaptable to any representation learning setup and begets a Matryoshka Representation z by optimizing the original loss $\mathcal{L}(\cdot)$ at $O(\log(d))$ chosen representation sizes. Matryoshka Representation can be utilized effectively for adaptive deployment across environments and downstream tasks.

FIG. 2.7 : MRL Architecture

- MRL s'appuie sur des représentations de vecteurs hiérarchiques avec des dimensions imbriquées, inspirées des poupées russes (d'où le nom Matryoshka).

- Chaque dimension inférieure (par exemple, les 8 premières dimensions d'un vecteur 2048-D) capture une granularité spécifique, offrant des vecteurs à granularité grossière ou fine selon les besoins.

Formulation de la fonction de perte

Dans le cadre de l'apprentissage par représentations Matryoshka (MRL), la fonction de perte globale est définie comme suit :

$$\mathcal{L}_{\text{MRL}} = \frac{1}{N} \sum_{i=1}^N \sum_{m \in M} c_m \cdot \mathcal{L}_{\text{cls}} \left(W^{(m)} \cdot F(x_i; \theta_F)_{1:m}, y_i \right)$$

où :

- N : Nombre d'exemples dans l'ensemble d'entraînement.
- $M = \{m_1, m_2, \dots, m_k\}$: Ensemble des dimensions imbriquées ($m_1 < m_2 < \dots < m_k \leq d$).
- c_m : Poids d'importance pour la dimension m , généralement $c_m = 1$.
- \mathcal{L}_{cls} : Perte de classification pour chaque sous-représentation.
- $F(x_i; \theta_F)$: Réseau de base générant un vecteur de représentation $z \in \mathbb{R}^d$.
- $F(x_i; \theta_F)_{1:m}$: Sous-vecteur contenant les m premières dimensions de z .
- $W^{(m)} \in \mathbb{R}^{Lm}$: Poids du classifieur linéaire spécifique à m .
- $y_i \in [L]$: Label de l'exemple i .

[4]

2.5 Le Modèle de Génération des Réponses (LLM)

Une fois les informations pertinentes récupérées, elles sont utilisées par un modèle de génération de texte, tel qu'un LLM (Large Language Model), pour produire une réponse finale. Le modèle LLM, qui peut être un GPT ou un T5, prend en entrée les embeddings des documents récupérés et génère des réponses contextualisées et informées.

Le processus peut être formalisé comme suit :

$$\mathbf{R} = f_{\text{LLM}}(\mathbf{E}_{\text{retrieved}}) \quad (2.2)$$

Où \mathbf{R} est la réponse générée, et $\mathbf{E}_{\text{retrieved}}$ est l'ensemble des embeddings des documents pertinents récupérés.

2.5.1 Présentation de Qwen et des Modèles Sélectionnés

Qwen est une série de modèles de langage et multimodaux développée par l'équipe Qwen du groupe Alibaba. Les modèles de cette série, désormais à leur version Qwen2.5, ont été conçus pour exceller dans des tâches variées allant de la compréhension du langage naturel à la génération de texte, en passant par la compréhension multimodale (vision et audio) et l'utilisation d'outils. Ces modèles sont entraînés sur des données multilingues et multimodales à grande échelle, puis ajustés avec des données de qualité pour s'aligner sur les préférences humaines.

2.5.1.1 Caractéristiques Générales de Qwen2.5

La dernière version, Qwen2.5, introduit les fonctionnalités suivantes :

- **Modèles denses et faciles à utiliser** : Basés sur une architecture de type *decoder-only*, disponibles en différentes tailles (0.5B, 1.5B, 3B, 7B, 14B, 32B, et 72B) et en variantes *base* et *instruct*.
- **Données d'entraînement étendues** : Pré-entraînement sur un ensemble de données contenant jusqu'à 18 trillions de tokens.
- **Améliorations significatives** :
 - Suivi des instructions.
 - Génération de textes longs (jusqu'à plus de 8K tokens).
 - Compréhension et génération de données structurées (par exemple, des tableaux ou des JSON).
 - Résistance accrue à la diversité des prompts système, facilitant les applications comme les jeux de rôle ou les agents conversationnels.
- **Support de longue contextes** : Capacité à traiter des contextes allant jusqu'à 128K tokens et à générer jusqu'à 8K tokens.
- **Multilingue** : Support de plus de 29 langues, incluant le chinois, **l'arabe**, l'anglais, le français, l'espagnol, le portugais, l'allemand, l'italien, le russe, le japonais, le coréen, le vietnamien, le thaï, et d'autres.

2.5.1.2 Les Modèles Sélectionnés : Qwen/Qwen2.5-3B-Instruct et Qwen/Qwen2.5-7B-Instruct

Dans cette étude, nous avons choisi d'expérimenter deux modèles spécifiques de la famille Qwen :

- **Qwen/Qwen2.5-3B-Instruct** :
 - Taille : 3 milliards de paramètres.

- Conçu pour des applications nécessitant des performances solides tout en restant légers en termes de calcul.
- Adapté aux environnements où la latence et les coûts computationnels sont des facteurs critiques.
- La position du modèle Qwen/Qwen2.5-3B-Instruct dans le leaderboard Hugging Face est illustrée ci-dessous, mettant en évidence ses performances sur des tâches variées.

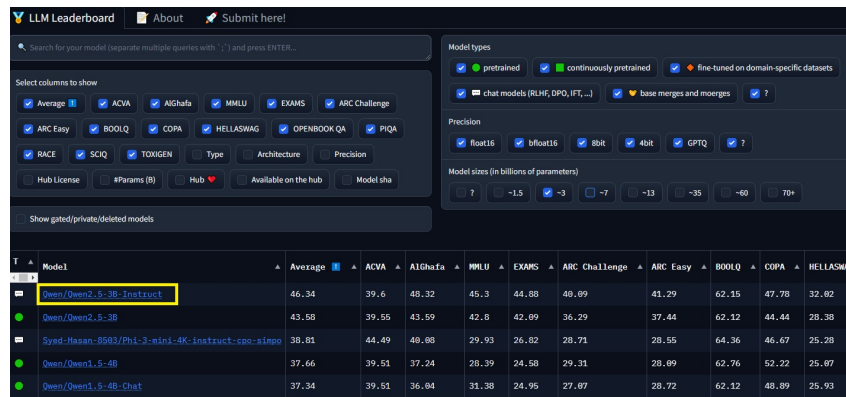


FIG. 2.8 : Classement du modèle Qwen/Qwen2.5-3B-Instruct sur le leaderboard Hugging Face

• Qwen/Qwen2.5-7B-Instruct :

- Taille : 7 milliards de paramètres.
- Offre des capacités améliorées pour des tâches plus complexes grâce à une compréhension approfondie et un plus grand nombre de paramètres.
- Idéal pour des applications où des performances optimales priment sur les contraintes de calcul.
- Le classement du modèle Qwen/Qwen2.5-7B-Instruct sur le leaderboard Hugging Face est présenté ci-après, soulignant sa compétitivité et sa capacité à traiter des tâches complexes.

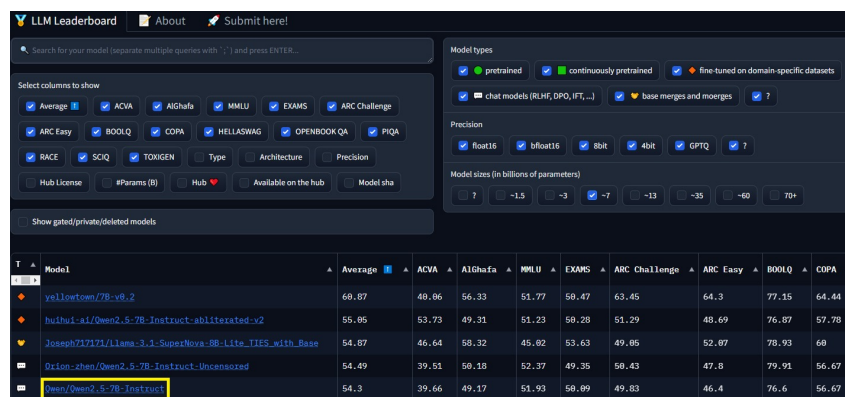


FIG. 2.9 : Classement du modèle Qwen/Qwen2.5-7B-Instruct sur le leaderboard Hugging Face

Ces deux modèles ont été sélectionnés en raison de leur réputation dans la communauté et de leur classement élevé dans les leaderboards de Hugging Face. Les tests comparatifs que nous avons réalisés mettent en lumière leurs forces et faiblesses respectives, comme détaillé dans la section suivante.

2.5.2 Paramètres de Génération Utilisés

Lors de l'évaluation des modèles Qwen/Qwen2.5-3B-Instruct et Qwen/Qwen2.5-7B-Instruct, plusieurs paramètres de génération ont été configurés pour optimiser les réponses générées. Ces paramètres et leur rôle sont décrits ci-dessous :

- **temperature=0.2** : Ce paramètre contrôle la créativité des réponses générées. Une valeur basse, comme 0.2, favorise des réponses plus déterministes et précises, en limitant la diversité des choix. Cela est particulièrement utile pour des tâches nécessitant des réponses cohérentes et factuelles.
- **repetition_penalty=1.1** : Ce paramètre réduit la probabilité que le modèle répète les mêmes phrases ou mots. En appliquant une légère pénalité lors des répétitions, il améliore la fluidité et l'originalité des réponses.
- **return_full_text=False** : Indique que seul le texte généré doit être retourné, sans inclure le prompt d'entrée. Cela permet d'obtenir des sorties plus claires et directement exploitables.
- **max_new_tokens=500** : Spécifie le nombre maximal de nouveaux tokens que le modèle peut générer. Ce paramètre contrôle la longueur des réponses produites, assurant un contenu suffisant tout en évitant une génération excessive.

Ces réglages jouent un rôle crucial dans l'adaptation du comportement des modèles aux besoins spécifiques du projet. Ils permettent d'atteindre un équilibre entre précision, créativité et concision, en fonction des exigences des tâches.

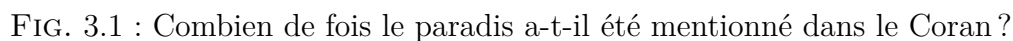
2.6 Réalisation Finale

Dans la réalisation finale du système, nous avons intégré les différentes étapes du RAG dans une solution cohérente et efficace. Le flux de travail commence par la récupération d'informations à partir de documents externes, suivi par la génération des embeddings, et enfin, l'utilisation d'un modèle LLM pour fournir une réponse générée de manière optimale. Le système a été évalué en termes de précision, pertinence et temps de réponse, démontrant ainsi son efficacité dans des tâches complexes de question-réponse.

Réalisation

Par ailleurs, nous avons utilisé **Ngrok** en tant que tunnel afin de permettre un accès sécurisé et rapide aux services hébergés localement. Cette solution a facilité l'interaction avec les modèles et l'intégration des outils nécessaires à notre analyse.

3.1.1 En utilisant : Qwen/Qwen2.5-3B-Instruct



3.1.2 En utilisant : Qwen/Qwen2.5-7B-Instruct



FIG. 3.5 : Qui sont les hypocrites ?



FIG. 3.6 : Qui sont les habitants de l'Enfer ?



FIG. 3.7 : Qui sont les habitants du paradis dans le Coran ?


```

Question: من هو الله ؟
Answer:
الله هو الخالق لكل شيء. وهو رب السموات والأرض. ولا يعلم من العباد وجهه. ولكنه على كل شيء. يحب غلظه وحده دون غيره. والله أن ينطق الناس به في مقامهم
Source docs:
Document 0
*SJIN=0.408351
*Content:
الآية: [وقالوا اتخذ الله رداً عليه بل لي آله من السموات والأرض كل له فائقون:]
التفسير: [وقلت اليهود والنصارى والمشركون: اتخذ الله لنفسه رداً. قلوا لله سبحانه. عن هذا القول الظل: أي كل من في السموات والأرض عليه وجهه وهم سبحانه. فليسوا له أولاداً ولا أولاداً لهم. بل هم آلهة]
*Metadata:
[الصدرة: "صدرة الفكرة" و "رقم الآية": "102"]
*****
Document 1
*SJIN=0.409279
*Content:
الآية: [لئن لم يكن له آية إلا هو خالق كل شيء فاعلموه وهو على كل شيء وكيل:]
التفسير: [لئنكم -أيها المشركون- هو ربكم كل رداً لا يجوز بحق ربوا خالق كل شيء. فاعلموا واعلموا له بلطاعة وإحساناً له ولطاعة وإحساناً له على كل شيء وكل وجهه. بين أمور خلقه [الصدرة: "صدرة الأثر" و "رقم الآية": "102"]
*****
Document 2
*SJIN=0.422197
*Content:
الآية: [وهو الله في السموات وعلى الأرض يعلم سركم ومخبركم ويعلم ما تكفرون:]
التفسير: [هو سبحانه هو إله الصوري في السموات والأرض. ومن خلال أفرجه له جميع ما تعلموه -أيها الناس- وما تظنونه ويعلم جميع أصنافكم من غير أن تراه. وإلهة غيره -أيها المشركون- الباطل]
*Metadata:
[الصدرة: "صدرة الأثر" و "رقم الآية": "13"]
*****
Document 3
*SJIN=0.422213
*Content:
الآية: [إن من آية الله أن لا آية إلا أنا فاعلموا أنكم تكفرون:]
التفسير: [إن من آية الله أن لا آية إلا أنا. لا شيء إلا أنا. لا شيء إلا أنا فاعلموا وحده. وأنكم تكفرون تكفرون أيها المشركون]
*Metadata:
[الصدرة: "صدرة علم" و "رقم الآية": "14"]
*****

```

FIG. 3.8 : Qui est Dieu ?

3.1.3 Interface

ما هي الحكمة من فريضة الصيام وفقاً للقرآن؟

وفقاً للقرآن، فرض الصيام لعل الناس يتقون الله. وهذا يعني أن الصيام يجعل الناس يتجنبون المعاصي ويحافظون على طاعة الله وعبادة الله وحده.

Retrieved Documents

Document 1 (Score: 0.420)

Content:

الآية: [يا أيها الذين آمنوا كتب عليكم الصيام كما كتب على الذين من قبلكم لعلكم تتقون] التفسير: يا أيها الذين صدقوا الله ورسوله وعلما بشره، فرض الله عليكم الصيام كما فرضه على الأمم قبلكم؛ لعلكم تتقون ربكم، فتجعلون بينكم وبين المعاصي وقاية بطاعته وعبادته وحده

Enter your query - Arabic:

FIG. 3.9 : Quelle est la sagesse derrière l'obligation du jeûne dans le Coran ?

ما أهمية الصبر في مواجهة المصاعب في القرآن؟

أهمية الصبر في مواجهة المصاعب في القرآن تتجلى في عدة نقاط:

1. الصبر هو سلاح المؤمن في وجه الأذى والإساءة.
2. الصبر يؤدي إلى الفوز برضوان الله ونجاح في الدنيا والآخرة.
3. الصبر يساعد على الانتصار على الأعداء وعدم الاستسلام لهم.
4. الصبر يجعل الإنسان يثق في الله ويستعاض عنه في الشدة.
5. الصبر يؤدي إلى العفو عن الأذى وتجاوزة، وهو فعل حميد.

فالصبر في القرآن الكريم هو طريق المؤمنين نحو النصر والسعادة.

FIG. 3.10 : Quelle est l'importance de la patience face aux épreuves dans le Coran ?



FIG. 3.11 : Qui sont les habitants de la caverne?

RAG-based Quran Chatbot

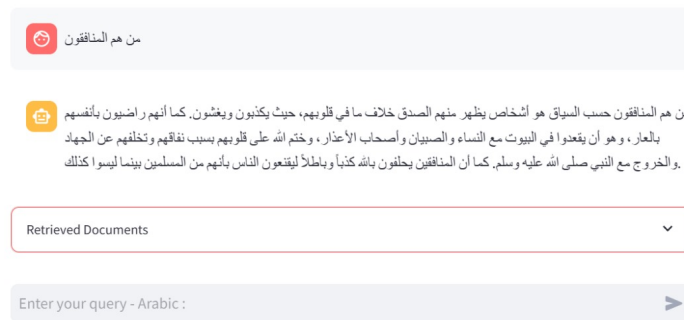


FIG. 3.12 : Qui sont les hypocrites?

RAG-based Quran Chatbot

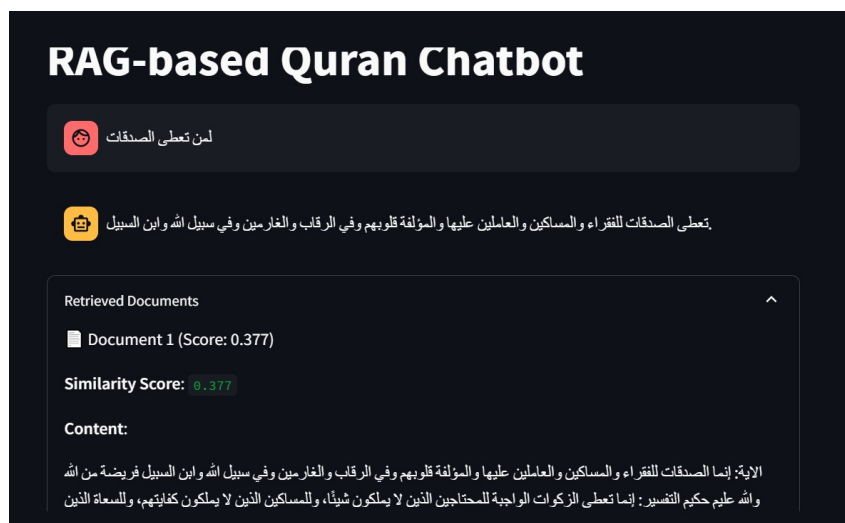


FIG. 3.13 : À qui sont destinées les aumônes?

Chapitre 4

Bibliographie

- 1 [MTEB Learderboard](#)
- 2 [ArabicSTSMatryoshkaV2 model](#)
- 3 [ArabicTripletMatryoshkaV2 model](#)
- 4 [Matryoshka Representation Learning Paper](#)