



Royaume du Maroc
Ministère de l'Enseignement Supérieur,
de la Recherche Scientifique et de l'Innovation
École Nationale des Sciences Appliquées
ENSA – Tanger

Projet : Chatbot Web Sémantique (Domaine Santé)

Introduction au Web Sémantique

Module : IA Avancée & Web Sémantique

Réalisé par :
Abdelilah BENCHINE
Génie Informatique

Encadré par :
M.AMECHNOUE KHALID
Enseignant à l'ENSA de Tanger

Année universitaire : 2025 – 2026

Table des matières

1	Introduction générale	2
2	Fondamentaux du Web Sémantique	3
2.1	Principe général	3
2.2	RDF, OWL et SPARQL	3
2.3	De l'ontologie aux connaissances	3
3	Base de connaissances utilisée	4
3.1	Source de l'ontologie	4
3.2	Concepts représentés	4
4	Implémentation du chatbot sémantique	5
4.1	Architecture générale	5
4.2	Chargement de la base de connaissances	5
4.3	Analyse NLP (Natural Language Processing)	6
4.4	Génération automatique de requêtes SPARQL	6
4.5	Interaction avec l'utilisateur	7
5	Résultats obtenus	8
5.1	Exemple d'exécution	8
	Capture d'écran	9
6	Analyse critique et perspectives	10
6.1	Compréhension acquise	10
6.2	Améliorations possibles	10
	Conclusion	11

Chapitre 1

Introduction générale

Le Web a évolué d'un simple ensemble de pages destinées aux humains vers un espace où les données doivent être comprises, reliées et exploitées automatiquement par les machines. Cette vision est appelée **Web sémantique**.

L'objectif de ce projet est de mettre en pratique ces concepts en développant :

- une **base de connaissances** décrivant un domaine réel (la santé),
- une **ontologie OWL** structurant ce domaine,
- un **graphe RDF** représentant les faits,
- des requêtes **SPARQL** pour interroger la connaissance,
- un **chatbot** capable de transformer une question en langage naturel en requête SPARQL.

Le domaine choisi est la **santé**, car il contient naturellement des entités (médecins, patients, diagnostics...) et des relations (soigne, prescrit, diagnostique...) adaptées au Web sémantique.

Chapitre 2

Fondamentaux du Web Sémantique

2.1 Principe général

Web classique + IA (NLP) + RDF/OWL = Web sémantique

Contrairement au Web classique, le Web sémantique cherche à :

- rendre les données **compréhensibles par les machines**,
- permettre des **raisonnements automatiques**,
- faciliter l'interopérabilité entre applications.

2.2 RDF, OWL et SPARQL

- **RDF** représente les connaissances sous forme de triplets **Sujet-Prédicat-Objet**.
- **OWL** permet de définir une ontologie : classes, relations et contraintes.
- **SPARQL** est le langage d'interrogation des graphes RDF.

2.3 De l'ontologie aux connaissances

Nous distinguons :

- la **TBox** : schéma du domaine (OWL),
- la **ABox** : faits concrets (RDF).

Dans ce projet, ces deux aspects sont stockés dans un fichier RDF/XML issu d'une ontologie réelle.

Chapitre 3

Base de connaissances utilisée

3.1 Source de l'ontologie

Nous avons utilisé une ontologie open-source disponible sur GitHub :

`https://github.com/Ahmedmessoudi/Project-WebSemantique`

Remerciement : nous remercions **Ahmed Messoudi** pour la mise à disposition des fichiers :

- `sante_ontologie.owl` : ontologie éditée sous Protégé,
- `sante_ontologie.rdf` : base de connaissances RDF utilisée par le chatbot.

3.2 Concepts représentés

L'ontologie modélise notamment :

- **Medecin**, **Patient**, **EtablissementSante**,
- **Diagnostic**, **ActeMedical**, **Traitement**.

Quelques relations clés :

- `aPourPatient` : un médecin suit un patient,
- `TravailleDans` : un médecin travaille dans un établissement,
- `aPourDiagnostic` : un patient a reçu un diagnostic,
- `prescrit` : un médecin prescrit un traitement.

Ces relations ont été interrogées via SPARQL dans le chatbot.

Chapitre 4

Implémentation du chatbot sémantique

Cette partie présente l’implémentation pratique du projet. L’objectif est de montrer comment les concepts théoriques (RDF, OWL, SPARQL et NLP) sont intégrés dans une application fonctionnelle sous forme de chatbot.

Le code a été développé en Python, en utilisant la bibliothèque `rdflib` pour manipuler le graphe RDF et exécuter les requêtes SPARQL.

4.1 Architecture générale

L’architecture logicielle adoptée repose sur une organisation en couches simples :

Utilisateur → NLP → SPARQL → Graphe RDF → Réponse

- l’utilisateur pose une question en langage naturel ;
- le module NLP analyse la question et identifie l’intention ;
- une requête SPARQL correspondante est générée automatiquement ;
- la requête est exécutée sur la base de connaissances RDF ;
- le résultat est reformulé de manière lisible.

4.2 Chargement de la base de connaissances

Le cœur du projet repose sur le fichier `sante_ontologie.rdf`, qui contient :

- les classes,
- les relations (propriétés),
- les individus (médecins, patients, établissements, diagnostics...).

Ce fichier est chargé en mémoire sous forme de graphe RDF grâce à la bibliothèque `rdflib`. Le code suivant crée un graphe puis importe la base :

```
from rdflib import Graph
```

```
g = Graph()
g.parse("ontology/sante_ontologie.rdf", format="xml")

print(f"Graph chargé avec {len(g)} triplets.")
```

Une fois chargé, le graphe peut être interrogé avec SPARQL.

```
PS C:\Projets\web-semantique-chatbot> python .\src\test_kb.py
[KB] Graph chargé avec 694 triplets.
http://www.co-ode.org/ontologies/ont.owl#DrAymen
http://www.co-ode.org/ontologies/ont.owl#DrHicham
http://www.co-ode.org/ontologies/ont.owl#DrKhadija
http://www.co-ode.org/ontologies/ont.owl#DrMedAli
http://www.co-ode.org/ontologies/ont.owl#Manipulateur_IRM
```

(Espace réservé pour une capture de la console montrant le nombre de triplets chargés)

4.3 Analyse NLP (Natural Language Processing)

Le chatbot doit être capable de comprendre, au moins partiellement, les questions en français.

Pour simplifier, un module NLP **basé sur des règles** a été implémenté. Il se contente de détecter :

- **l'intention** : ce que l'utilisateur cherche (patients d'un médecin, établissements, diagnostic, etc.);
- **l'entité** concernée : par exemple DrHicham ou le diagnostic Diagnostique_COVID19.

Exemple simplifié de détection :

```
if "patient" in question and "dr_hicham" in question:
    intent = "patients_medecin"
    entity = "DrHicham"
```

Bien que simple, ce mécanisme permet déjà de simuler le fonctionnement d'un véritable module NLP.

4.4 Génération automatique de requêtes SPARQL

Une fois l'intention et l'entité identifiées, le chatbot construit dynamiquement une requête SPARQL.

Par exemple, pour récupérer les établissements où travaille un médecin :

```
PREFIX ont: <http://www.co-ode.org/ontologies/ont.owl#>

SELECT DISTINCT ?etab WHERE {
    ont:DrHicham ont:TravailleDans ?etab .
}
```

Cette requête interroge le graphe RDF et retourne les ressources liées.

Le programme exécute ensuite la requête :

```
results = g.query(sparql_query)
```

4.5 Interaction avec l'utilisateur

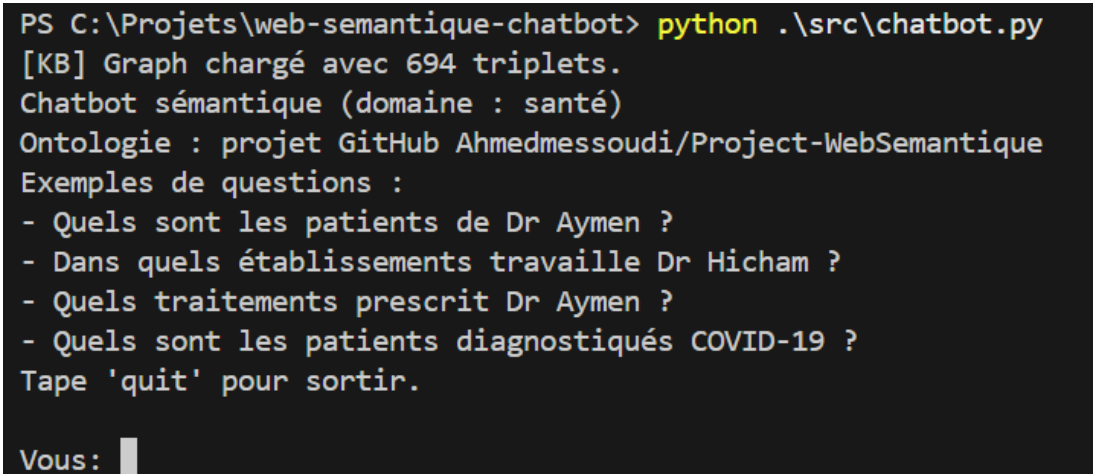
Le chatbot fonctionne en boucle interactive :

1. lecture de la question,
2. analyse NLP,
3. génération SPARQL,
4. exécution,
5. affichage de la réponse.

Un exemple d'exécution :

```
Vous : Dans quels tablissemments travaille Dr Hicham ?
Bot : DrHicham travaille dans : Cabine_DrHicham, Hopital_IbnSina.
```

Le rôle principal du chatbot est donc de servir d'interface entre l'utilisateur et la base de connaissances sémantique.



```
PS C:\Projets\web-semantique-chatbot> python .\src\chatbot.py
[KB] Graph chargé avec 694 triplets.
Chatbot sémantique (domaine : santé)
Ontologie : projet GitHub Ahmedmessoudi/Project-WebSemantique
Exemples de questions :
- Quels sont les patients de Dr Aymen ?
- Dans quels établissements travaille Dr Hicham ?
- Quels traitements prescrit Dr Aymen ?
- Quels sont les patients diagnostiqués COVID-19 ?
Tape 'quit' pour sortir.

Vous: █
```

(Espace réservé pour une capture d'écran du chatbot en fonctionnement)

Chapitre 5

Résultats obtenus

Cette section présente les principaux résultats obtenus lors de l'exécution du chatbot sémantique. L'objectif n'est pas uniquement d'afficher une réponse, mais de vérifier que toutes les étapes du pipeline fonctionnent correctement :

- compréhension de la question en langage naturel,
- génération correcte d'une requête SPARQL,
- interrogation de la base de connaissances RDF,
- production d'une réponse cohérente et interprétable.

5.1 Exemple d'exécution

Un premier scénario de test a consisté à interroger les établissements dans lesquels exerce un médecin présent dans la base.

Question posée par l'utilisateur :

Dans quels établissements travaille Dr Hicham ?

Le module NLP identifie que :

- l'entité concernée est le médecin **DrHicham**,
- l'intention correspond à une recherche d'établissements associés,
- il faut exploiter la propriété **TravailleDans**.

Une requête SPARQL est alors construite automatiquement :

```
PREFIX ont: <http://www.co-ode.org/ontologies/ont.owl#>

SELECT DISTINCT ?etab WHERE {
    ont:DrHicham ont:TravailleDans ?etab .
}
```

Après exécution, le chatbot renvoie la réponse suivante :

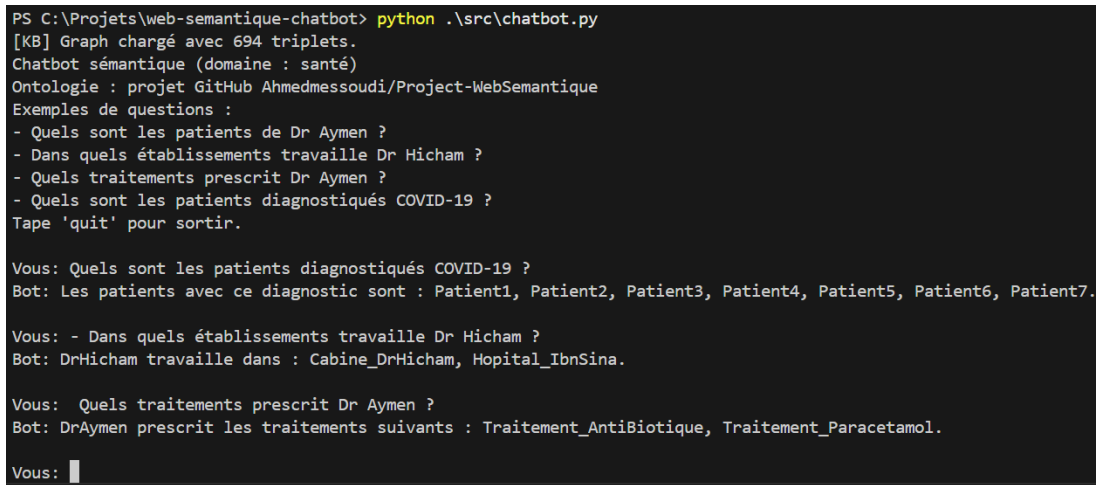
DrHicham travaille dans : Cabine_DrHicham, Hopital_IbnSina.

Cette réponse n'est pas codée en dur : elle provient directement du graphe RDF, démontrant que :

- la base de connaissances est correctement chargée ;
- la requête SPARQL correspond exactement à la structure de l'ontologie ;
- le formatage final de la réponse est correct.

Capture d'écran

L'image suivante illustre une exécution réelle du programme :



```
PS C:\Projets\Web-semantic-chatbot> python .\src\chatbot.py
[KB] Graph chargé avec 694 triplets.
Chatbot sémantique (domaine : santé)
Ontologie : projet GitHub Ahmedmessoudi/Project-WebSemantique
Exemples de questions :
- Quels sont les patients de Dr Aymen ?
- Dans quels établissements travaille Dr Hicham ?
- Quels traitements prescrit Dr Aymen ?
- Quels sont les patients diagnostiqués COVID-19 ?
Tape 'quit' pour sortir.

Vous: Quels sont les patients diagnostiqués COVID-19 ?
Bot: Les patients avec ce diagnostic sont : Patient1, Patient2, Patient3, Patient4, Patient5, Patient6, Patient7.

Vous: - Dans quels établissements travaille Dr Hicham ?
Bot: DrHicham travaille dans : Cabine_DrHicham, Hopital_IbnSina.

Vous: Quels traitements prescrit Dr Aymen ?
Bot: DrAymen prescrit les traitements suivants : Traitement_AntiBiotique, Traitement_Paracetamol.

Vous: █
```

(Espace réservé pour une capture d'écran de la console affichant la question et la réponse)

Cet exemple confirme que la chaîne complète :

$$\text{Langagenaturel} \Rightarrow \text{NLP} \Rightarrow \text{SPARQL} \Rightarrow \text{GrapheRDF} \Rightarrow \text{Rponse}$$

fonctionne correctement sur une base de connaissances réelle.

Chapitre 6

Analyse critique et perspectives

6.1 Compréhension acquise

Ce projet m’a permis de comprendre :

- la différence entre TBox et ABox,
- le rôle des ontologies dans la structuration des connaissances,
- la modélisation RDF sous forme de graphes,
- l’interrogation via SPARQL,
- l’intégration d’un module NLP au-dessus d’une base de connaissances.

6.2 Améliorations possibles

- utilisation d’un NLP avancé (spaCy, Transformers),
- ajout d’un moteur d’inférence OWL,
- extension du domaine médical,
- création d’une interface web complète.

Conclusion

Le travail réalisé au cours de ce projet m’a permis de passer de la compréhension théorique du Web sémantique à une mise en pratique concrète à travers le développement d’un chatbot capable d’interroger une base de connaissances dans le domaine de la santé.

À travers l’utilisation combinée de RDF, OWL, SPARQL et d’un module simple de traitement du langage naturel, j’ai pu constater comment les données, lorsqu’elles sont modélisées sous forme de graphe et liées par des relations sémantiques, peuvent être exploitées de manière beaucoup plus intelligente qu’un stockage classique.

Le projet a notamment montré que :

- une ontologie permet de structurer clairement un domaine ;
- une base RDF rend les connaissances interopérables et exploitables par des programmes ;
- SPARQL offre un puissant langage d’interrogation ;
- même un NLP simple peut servir d’interface entre un humain et une base sémantique.

Bien que le prototype mis en place reste volontairement limité (en termes de types de questions, vocabulaire reconnu et taille de la base de connaissances), il constitue une preuve de concept solide. Il démontre qu’il est possible de concevoir des systèmes capables de comprendre des requêtes naturelles et d’y répondre sur la base de connaissances structurées.

Dans une perspective future, plusieurs pistes d’amélioration s’ouvrent naturellement : intégration d’un NLP plus avancé, ajout d’un moteur d’inférence OWL, enrichissement de l’ontologie, ou encore déploiement d’une interface web accessible.

Ainsi, ce projet confirme l’intérêt et le potentiel du Web sémantique dans la construction d’applications intelligentes capables de relier l’information, de la structurer et de la rendre exploitable par les machines autant que par les utilisateurs humains.