



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE

UNIVERSITE IBN KHALDOUN - TIARET

MEMOIRE

Présenté à :

FACULTÉ DES MATHÉMATIQUES ET DE L'INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

MASTER

Spécialité : Réseaux et Télécommunications

Par :

Rahmani Fatima Zohra

Sur le thème

Une Approche basée TextING pour la classification des services web

Soutenu publiquement le 29 / 06 / 2022 à Tiaret devant le jury composé de :

Mr Daoud Bachir	MAA Université IBN-KHALDOUN Tiaret	Président
Mr Meghazi Hadj Madani	MAA Université IBN-KHALDOUN Tiaret	Encadreur
Mr Mostfaoui Kadda	MAA Université IBN-KHALDOUN Tiaret	Examineur

2021-2022

Remerciements

Tout d'abord, je remercie Dieu Tout Puissant qui m'a donné le courage et la volonté de faire cet humble travail.

Je remercie mes parents, mes frères et ma sœur qui m'ont soutenu durant mes études. Je remercie Monsieur Meghazi Hadj Madani mon encadreur pour son aide, ses encouragements, ses orientations et ses précieux conseils lors de la réalisation de ce travail.

Je remercie, également, les membres de jury pour avoir accepté de juger mon travail.

Enfin, je voudrais profiter de cette occasion pour exprimer mes remerciements les plus profonds et ma profonde gratitude à Lina Ghali Karim Ismaili et Badreddine Korichi qui m'ont aidé de près ou de loin dans la réalisation de ce message.

Dédicace

Avec un énorme plaisir, un cœur ouvert et une immense joie, que je dédie mon travail à mes très chers, respectueux et magnifiques parents qui m'ont soutenu tout au long de ma vie ainsi à mes sœurs Amina et Khaira, mes frères Amine et Abdellah, et à toute ma famille.

Et mes amis en particulier à Touta Zegaoui, adjilia Meriem, Zahi Narimane, Lamis, Zineb, Asma, Imane Mostfaoui et sarah Massoudi.

A toutes personnes qui m'ont encouragé ou aidé au long de mes études

Résumé

Lorsque nous parlons des services Web, le problème le plus courant est de trouver le meilleur moyen de permettre aux clients de découvrir les services web, dans cette perspective, les approches mises en œuvre, pour la découverte de services Web, sont moins efficaces et manquent de précision. Pour résoudre ce problème, de nombreux travaux de recherche ont été menés et montré que la classification des services web s'avère une solution qui peut réduire la complexité du problème. Ce projet vise à étudier une technique de classification dans le domaine de la "classification de textes" issu de domaine Deep Learning et de l'appliquer sur les descriptions textuelles des services web afin de prédire au mieux leurs classes, les résultats expérimentaux sont satisfaisants et démontrent l'utilité de l'approche.

Mots clé : Service Web, la découverte des services web, Deep Learning, classification de texte

Abstract

When we talk about web services, the most common problem is to find the best way to allow customers to discover web services, in this perspective, the approaches implemented, for the discovery of web services, are less efficient and lack precision. To solve this problem, many research works have been carried and showed that the classification of web services is a solution that can reduce the complexity of the problem. This project aims to study a classification technique in the field of "text classification" from the Deep Learning domain and the possibility of applying it to the textual descriptions of Web Services in order to better predict their classes, the experimental results are satisfactory and demonstrate the usefulness of the approach.

Keywords: Web Service, Web Service discovery, Deep Learning, text classification

ملخص

عندما نتحدث عن خدمات الويب ، فإن المشكلة الأكثر شيوعًا هي العثور على أفضل طريقة للسماح للعملاء باكتشاف خدمات الويب ، ومن هذا المنظور ، فإن الأساليب المطبقة لاكتشاف خدمة الويب تكون أقل كفاءة و تفتقر الى الدقة ، لحل هذه المشكلة. تم الاجراء العديد من الأعمال البحثية و اظهرت أن تصنيف خدمات الويب هو حل يمكن ان يقلل من تعقيد المشكلة ، يهدف هذا المشروع إلى دراسة تقنية تصنيف في مجال "تصنيف النص" من مجال التعلم العميق وتطبيقها على الأوصاف النصية لخدمات الويب من اجل التنبؤ بشكل افضل بفصولها ، والنتائج التجريبية مرضية وتوضح فائدة النهج **الكلمات المفتاحية :** خدمة الويب , اكتشاف خدمة الويب , التعلم العميق , تصنيف النص.

Table des Matières :

Liste des figures	7
Liste des tableaux:.....	9
Introduction Générale :.....	1
Chapitre I : Les fondements de base des Services Web	1
1-Introduction :.....	4
2-Les services web :	4
3- Les Services SOAP :.....	8
4- Les Services REST :	11
5-Grands challenge des services web :.....	13
6- Conclusion :	18
Chapitre II : Les techniques du Machine Learning	19
1-Intoduction :.....	20
2-Les techniques d'analyse des données :.....	20
3-Le machine learning :.....	27
4- Deep Learning :.....	33
5- Conclusion :	38
Chapitre III : Classification de texte pour la découverte des Services Web	39
1 -Introduction :.....	40
2- Les Algorithmes Deep learning pour la classification de texte :	40
3- Quelques travaux connexes pour la classification des services web :	49
4 – Conclusion.....	51
Chapitre IV : Amélioration de processus de découverte des services Web	52
1-Introduction :.....	53
2-Graph neural network (GNN):	53
3- TextING :	57
4- Le modèle TextING pour les services web :.....	60

5-Configuration expérimentale et l'ensemble de données :	65
6- Discussion des résultats :	66
7-Conclusion :	69
Conclusion générale et perspective :	72
Bibliographies :	73
Webographies:.....	76

Liste des figures

Figure 1-1 : Le fonctionnement des Services Web	6
Figure 1-2 : La description de WSDL	7
Figure 1-3 : Le protocole SOAP	8
Figure 1-4 : Structure d'un message SOAP	9
Figure 1-5 : Exemple de Message SOAP	10
Figure 1-6 : L'architecture de REST	12
Figure 1-7 : Les étapes de découverte	13
Figure 1-8 : La Composition de services	16
Figure 2-1 : La régression linéaire simple.....	23
Figure 2-2 : Figure explique le résultat de classification	25
Figure 2-3 : Une visualisation de résultat de Clustering	26
Figure 2-4 : Le fonctionnement de Machine Learning	28
Figure 2-5 : Les algorithmes de Machine Learning	28
Figure 2-6 : L'apprentissage supervisé	29
Figure 2-7 : Algorithme de K-Nearest Neighbors	30
Figure 2-8 : L'apprentissage supervisé vs l'apprentissage non supervisé	31
Figure 2-9 : Le fonctionnement de L'apprentissage par renforcement	32
Figure 2-10 : Une visualisation de Deep Learning.....	33
Figure 2-11: L'architecture d réseau de neurone artificiel.....	34
Figure 2-12 : L'architecture de Perceptron multicouche (MLP).....	35
Figure 2-13 : Architecture d'un réseau de neurone convolutif.....	35
Figure 2-14 : L'architecture des réseaux de neurone récurrent.	36
Figure 2-15 : Unité de base LSTM	37
Figure 3-1 : Classification de texte avec CNN	41
Figure 3-2 : Classification de texte avec RNN	42
Figure 3-3 : Gated Recurrent Unit	42

Figure 3-4: Long Short-Term Memory (LSTM).....	43
Figure 3-5 : Les réseaux de neurones convolutifs récurrents (RCNN) pour la classification de texte	44
Figure 3-6 : La classification de texte par les GNN	45
Figure 3-7 : Text Graph convolution Network	47
Figure 3-8 : La classification de texte à l'aide de LTM TEXT-GCN	48
Figure 3-9 : Architecture ServeNet	50
Figure 4-1 : La visualisation d'un simple Graphe.....	54
Figure 4-2 : Embedding de graphe	55
Figure 4-3 : La classification de nœud	55
Figure 4-4 : La classification ou niveau des liens	56
Figure 4-5 : La classification de graphe	56
Figure 4-6 : La méthode TextING.....	58
Figure 4-7 : La construction de graphe de TextING	59
Figure 4-8 : Processus de notre approche.....	61
Figure 4-9 : La distribution des mots hors vocabulaire.....	62
Figure 4-10 : La construction des graphes pour les services web	62
Figure 4-11 : La figure explique les coocurrence entre les sommets	63
Figure 4-12 : La construction des graphes pour (training, validation + training et teste)...	63
Figure 4-13 : L'enregistrement des graphes.....	64
Figure 4-14 : L'interaction de mots entre les services web	64
Figure 4-15 : La fonction softmax_cross_entropy	65
Figure 4-16 : L'environnement Saturn cloud	65
Figure 4-17 : Ensemble de données utilisées de l'approche TextING	66
Figure 4-18 : Le chargement de graphe	67
Figure 4-19 : L'entraînement de modèle TextING	67
Figure 4-20 : Le graphe d'accuracy	69

Figure 4-21 : Le graphe de perte69

Liste des tableaux:

Tableaux 01 : Comparaison des performances de classification des modèles68

Introduction Générale :

Les services Web sont des briques de base logicielles s'affranchissant de toute contrainte de compatibilité logicielles ou matérielles, les Services Web comportent de nombreux avantages, ils sont utilisables à distance via n'importe quel type de plate-forme, ils peuvent servir au développement d'applications distribuées et sont accessibles depuis n'importe quel type de clients, les Services Web appartiennent à des applications capables de collaborer entre elles de manière transparente pour l'utilisateur, leur mise en œuvre repose sur une architecture décentralisée, cette architecture orientée services (Service Oriented Architecture-SOA) est un style architectural fondé sur la description des services et de leurs interactions, les services sont publiés dans des annuaires par des fournisseurs qui les hébergent, ils sont accessibles via un réseau pour les clients qui les découvrent, les sélectionnent, les invoquent et les utilisent.

Actuellement, les Services Web sont mis en œuvre au travers des technologies de base standards : WSDL, UDDI, SOAP et REST. Ces technologies facilitent la description, la découverte et la communication entre services.

La découverte de services Web est le processus qui consiste à trouver des services Web adaptés à une tâche donnée, la publication d'un service web implique de créer un artefact logiciel et de le rendre accessible aux consommateurs potentiels, les fournisseurs de services Web ajoutent à l'interface d'un point de terminaison de service une description d'interface à l'aide du langage de description des services Web (WSDL) afin qu'un consommateur puisse utiliser le service. Mais il est confronté au défi de choisir et de réaliser la meilleure configuration de service qui répond le mieux à ses besoins.

Problématique :

Vu le nombre important des services Web, le processus de découverte, ce qui représente un axe de recherche émergent consistant à satisfaire les besoins des clients est devenu difficile et manque de précision, le problème consiste à améliorer la découverte sur des données textuelles des services web par des méthodes de classification de texte issu de Deep Learning, beaucoup de travaux de recherche ont été fait pour résoudre cette problématique et ont prouvé que la classification des services web est la solution pour réduire la complexité du problème.

Contribution :

La classification des Services Web est une méthode utilisée afin d'améliorer la vitesse du processus de découverte. Dans le cadre de ce mémoire, nous proposons une approche qui se base sur la classification des services web qui utilisent des descriptions de ces derniers et prédire leurs catégories. Ensuite, nous citerons quelques travaux qui ont contribué au développement du découverte utilisant la classification et les comparerons avec nos résultats pour une étude future plus approfondie ayant pour objectif une nouvelle approche qui améliore le processus de découverte des services web et le rend plus précis, En prenant en compte les spécificités de ces derniers, Pour ce faire, nous avons structuré le présent manuscrit comme suit

Plan de Travail :

Le premier chapitre intitulé «Les fondements de base des Services Web», dans laquelle nous présenterons la définition des services web et ses technologies de base. Ainsi, nous parlerons de ses grands challenges qui contiendront, la découverte, la sélection et la composition.

Dans le deuxième chapitre intitulé « Les techniques du Machine Learning », dans laquelle nous présenterons les algorithmes et les différentes méthodes les plus utilisées dans Machine Learning.

Dans le troisième chapitre intitulé « Classification de texte pour la découverte des Services Web », dans laquelle nous présenterons le développement du processus de découverte par la classification de texte issu de Deep Learning, à la fin, nous mentionnerons quelques travaux notables dans ce domaine.

Dans le quatrième chapitre intitulé « TextING pour améliorer le processus de découverte des Services Web », dans laquelle nous introduirons notre approche adoptée pour la classification des services web. Ensuite nous parlerons des outils utilisés pour la mise en œuvre de notre propre solution, à la fin, en expliqueront toutes les étapes de l'implémentation et nous comparons les résultats obtenus par notre approche avec d'autres travaux.

Chapitre I : Les fondements de base des Services Web

1-Introduction :	4
2-Les services web :	4
2-1 Historique :	4
2-2 Définition :	5
2-3 Les technologies de base des SW :	7
3- Les Services SOAP :	8
4- Les Services REST :	11
5-Grands challenge des services web :	13
5-1 Découverte des SW :	13
5-2 Sélection des SW :	15
5-3 Composition des SW :	16
6- Conclusion :	18

1-Introduction :

L'architecture orientée services (SOA) est apparue comme une approche architecturale visant à améliorer les performances de prestation de services des systèmes hérités existants tout en préservant leurs propriétés les plus importantes. Cette approche en raison de la flexibilité de son adoption a attiré l'attention des universités et des entités commerciales, en particulier dans le développement de technologies avancées telles que le cloud computing (CC) et l'Internet des objets (IoT), bien que de nombreuses études aient listé les facteurs de succès de la SOA quelques petits échecs ont été rapportés dans la littérature malgré l'abondance de matériel SOA est disponible [1].

L'émergence de la technologie des services Web devrait apporter un niveau interopérabilité très élevée et couplage très faible afin d'établir la base d'une architecture orientée services avec une configuration dynamique avancée, qui est une sorte d'architecture d'application distribuée faiblement couplée (ou très couplée) par un ensemble d'applications distribuées autonomes décentralisées (services Web) interagit sur la base de protocoles de communication et met en œuvre travailler en utilisant des techniques ouvertes [2].

2-Les services web :

2-1 Historique :

Les services Web sont nés par les efforts plusieurs organisations ayant un intérêt commun dans le développement et le maintien de «marchés électroniques». Ils veulent pouvoir une communication plus facile pas besoin de négocier entre eux sur chaque transaction pouvoir interpréter leurs différentes données, ils veulent des isolés leurs systèmes informatiques de tous les autres.

C'est ainsi que l'EDI (Electronic Data Interchange) est né en 1975. L'EDI peut être défini comme l'échange de données transactionnelles entre ordinateurs utilisant des réseaux et des formats standardisés. EDI permettait à des personnes d'une langue donnée d'envoyer des messages via câble à des interlocuteurs similaires situés dans un autre lieu. EDI implémente différents applications pour échanger des protocoles de modélisation et de communication.

À la fin des années 1980, il était clair que l'ère des systèmes informatiques isolés arrivait finalement, différents ordinateurs de différentes tailles, les capacités et les formes ont émergé dans la même organisation, le service informatique veut du toute évidence l'utilisation intrinsèquement de la meilleure et la plus bénéfique des précieuses compétences

analytiques dont ils disposent utilisable, il est donc nécessaire de permettre aux applications informatiques de se substituer à leur travail, c'est-à-dire de pouvoir faire un véritable traitement distribué.

Pour faire face à cette nouvelle donne de nouvelles technologies sont apparues telles que CORBA (Common Object Request Broker Architecture) ou la version de Microsoft, le component Object Model (COM). CORBA est une architecture logicielle pour développer des composants. Ces composants sont assemblés pour construire une application complète, la capacité à écrire dans différents langages de programmation à exécuter sur un des processus indépendants, même déployés sur des machines différentes, les composants CORBA utilisent une approche intrinsèquement orientée objet (du point de vue du langage de programmation toutes les méthodes sont virtuelles pas de polymorphisme paramétrique, pas de méthode protégée ou privée pas de surcharge d'opérateur et pas de fonctions de première classe).

Les années 1990 ont non seulement vu une renaissance de l'ordinateur personnel Dans le même temps l'internet s'est considérablement développé avec la demande croissante de normes pouvant s'exécuter sur n'importe quelle plate-forme.

De nos jours, les Services Web suscitent un certain intérêt chez les architectes et les décideurs pour l'instant, les Web Services sont sortis de l'échange entre entreprises pour accueillir la référence au provisionnant des ressources de l'entreprise contre la standardisation complète d'une architecture distribuée de plus, Ce modèle n'est pas exempt de problèmes de performances : les données transmises en ASCII dans le XML lui-même sont intégrées dans une enveloppe SOAP puis HTTP, le problème du choix de la bonne granularité commun à toutes les architectures distribuées est encore plus aigu dans le cas des Web Services même s'ils n'ont pas encore atteint la maturité nécessaire à leur industrialisation. Les Web Services se présentent plus que jamais comme solution appropriée aux problématiques d'échange de données et d'intégration [3].

2-2 Définition :

Un Service Web est un programme informatique qui active et échange des données entre des applications et des systèmes hétérogènes dans un environnement distribué. Il s'agit donc d'un ensemble de fonctions visibles sur Internet ou sur un réseau interne par et pour des applications machines, sans intervention humaine et en temps réel [4].

Le groupe W3C travaillant sur les services Web utilise la définition de service Web suivante dans un document appelé « Architecture de Service Web » :

« Le service Web est un logiciel système conçu pour prendre en charge l'interaction de machine à machine qui peut être interagie via un réseau. Il a une interface décrite dans un format lisible par machine, d'autres systèmes qui interagissent avec le Service Web de manière descriptive utilisent des messages SOAP généralement transmis à l'aide de HTTP avec une sérialisation XML en conjonction avec d'autres normes liées au Web [5].»

La figure 1.1 schématise le fonctionnement des services web :

- Service provider service : les fournisseurs de services implémentent un service et le rendent disponible sur Internet.
- Service Requester : il s'agit d'un utilisateur de service Web, le demandeur exploite un service Web existant en ouvrant une connexion réseau et en envoyant une demande au format XML.
- Le registre de service : le registre de service est un annuaire de services, Le registre fournit un emplacement central aux programmeurs pour publier de nouveaux services ou les trouver [6].

La figure 1-1 illustre le fonctionnement des Services Web.

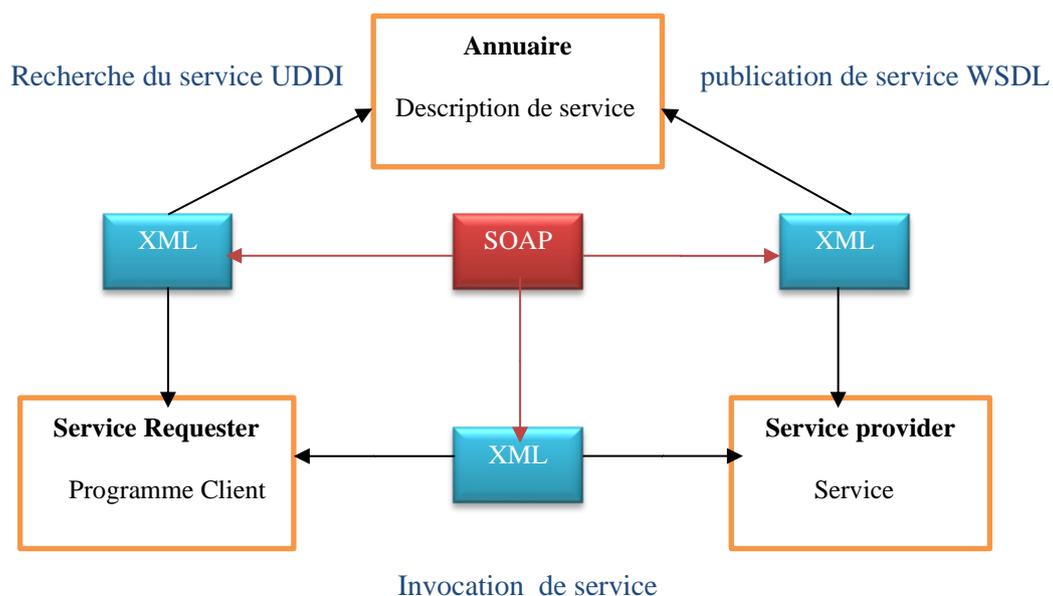


Figure 1-1 : Le fonctionnement des services web

2-3 Les technologies de base des SW :

La technologie des Services Web est un moyen de diffuser rapidement des informations entre les clients, les fournisseurs, les partenaires commerciaux et leurs différentes plateformes. Les services Web sont basés sur le modèle SOA, l'utilisation des technologies standards du Web telles HTTP et XML par les services Web facilite le développement d'applications réparties sur Internet, et permet d'avoir des applications très faiblement couplées. L'intégration est sans doute le facteur essentiel qui favorise l'utilisation des services Web. [6]. Les technologies de base des services Web sont WSDL UDDI, et SOAP.

I. WSDL :

WSDL (Web Services Description Language) est un langage de description standard. Il s'agit de l'interface présentée à l'utilisateur, il montre comment utiliser et interagir avec les services Web. WSDL est basé sur XML et permet de décrire avec précision les détails du service Web tels que le protocole, les ports utilisés, les opérations pouvant être effectuées, les formats de message d'entrée et de sortie et les exceptions pouvant être générées.

La figure 1-2 illustre La description de WSDL.

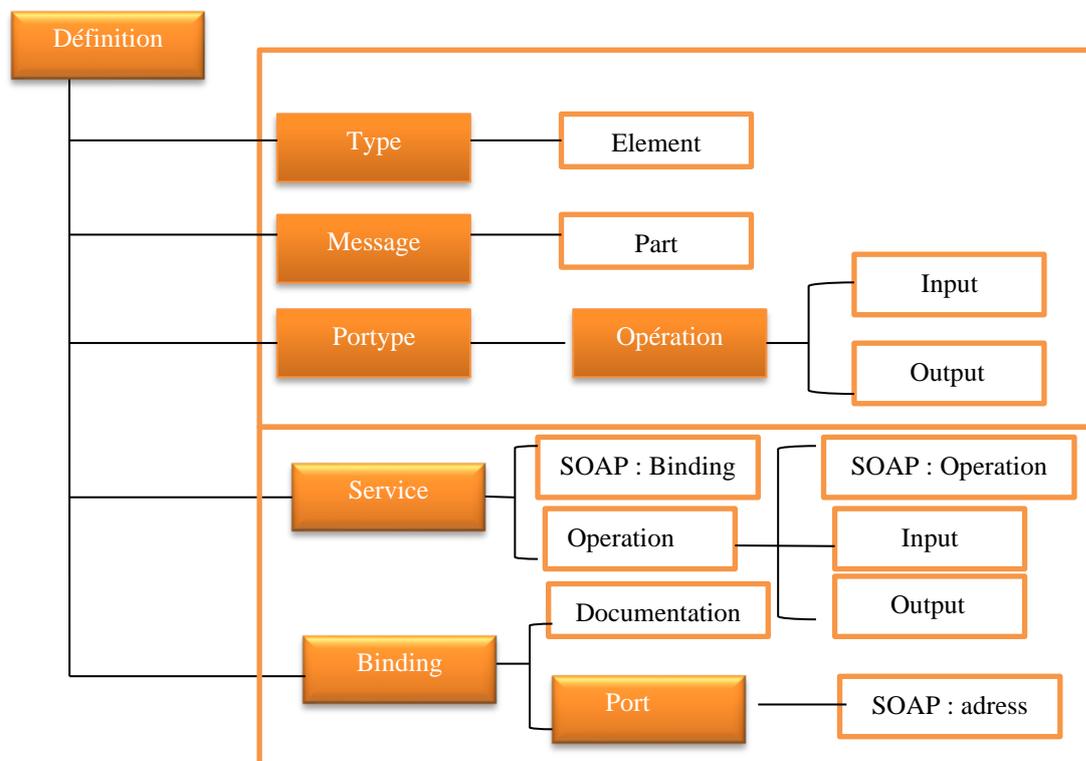


Figure 1-2 : La description de WSDL

UDDI :

Universal Description Discovery and Integration Services Directory (UDDI) est la norme pour la publication et la découverte d'informations sur les services Web. La spécification UDDI est une initiative d'ARIBA, Microsoft et IBM, cette spécification est maintenue par le groupe OASIS et non par le W3C, la spécification UDDI vise à créer un espace de travail ouvert indépendant de la plate-forme (framework) pour décrire, découvrir et intégrer des services des entreprises.

II. SOAP :

SOAP (Simple Object Access Protocol) est un protocole de communication standard. C'est l'épine dorsale des systèmes d'interopérabilité. SOAP est un protocole standardisé W3C écrit en XML. Il peut contenir des données ou des pièces jointes circule sur le protocole HTTP et apparaît comme une enveloppe signalée qui permet des appels de méthode à distance [6].

3- Les Services SOAP :**3-1 Définition :**

SOAP (Simple Object Access Protocol) est un protocole de messagerie. Il permet aux programmes exécutés sur différents systèmes d'exploitation tels que Windows et Linux et de communiquer à l'aide du protocole de transfert hypertexte (HTTP) et son langage le langage de balisage extensible (XML), sachant que les protocoles Web sont installés et mis à disposition sur tous les principaux systèmes d'exploitation, le protocole et le langage XML constituent une solution prête à l'emploi qui permet aux programmes de s'exécuter et de communiquer sur différents systèmes qui coexistent dans un réseau, le protocole http et le langage XML fournissent une solution prête à l'emploi qui permet aux programmes de s'exécuter et de communiquer sur différents systèmes pour coexister dans un réseau.

Le protocole SOAP spécifie comment les en-têtes HTTP et les fichiers XML sont codés afin que deux programmes sur des ordinateurs différents puissent s'appeler et s'échanger des informations. Le protocole SOAP spécifie également comment le programme appelé renvoie une réponse est souvent associé au protocole HTTP, mais prend en charge d'autres protocoles de transport [7].

La figure 1-3 illustre Le protocole SOAP.

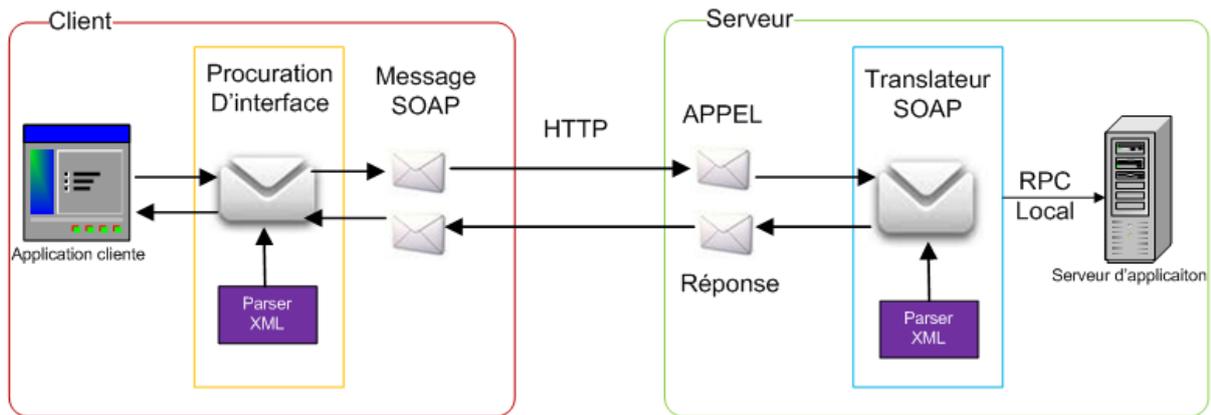


Figure 1-3 : Le protocole SOAP [6].

3-2 Structure d'un message SOAP :

Le message SOAP est assez facile à comprendre il fournit un moyen d'accéder à un objet en appelant une méthode à distance, les deux principales caractéristiques de SOAP sont sa simplicité et le fait que tout le monde s'accorde à l'utiliser. Le message SOAP se compose de deux parties essentielles : l'enveloppe SOAP, le corps SOAP et une partie optionnelle : entête SOAP [6].

La figure 1-4 montre Structure d'un message SOAP.

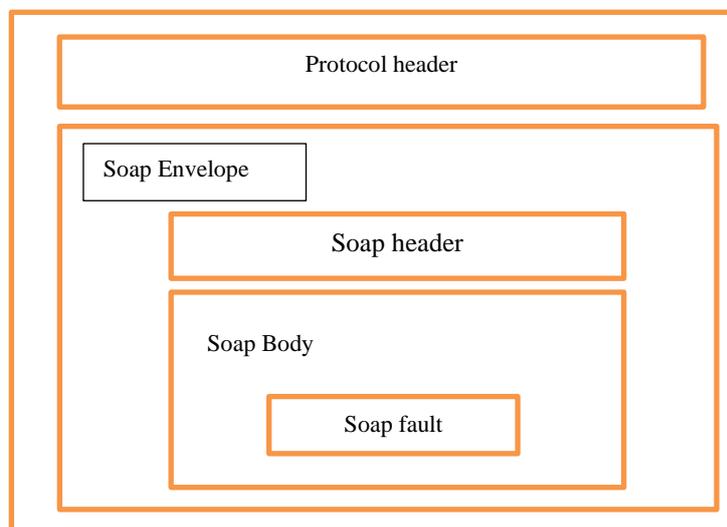


Figure 1-4 : Structure d'un message SOAP

- L'enveloppe : l'enveloppe SOAP est l'élément racine du message SOAP et est indispensable en substance. Les enveloppes contiennent des messages SOAP tout comme les enveloppes traditionnelles contiennent des lettres écrites [8].

- Le header (en-tête) : il s'agit d'une partie facultative qui permet d'ajouter des fonctionnalités aux messages SOAP décentralisé sans accord entre les parties à la communication, on spécifie si les notifications sont obligatoires ou facultatives. Les en-têtes sont particulièrement utiles lorsque le message doit être traité par plusieurs intermédiaires [6].
- Body : le corps SOAP est obligatoire et contient les messages d'application réels qui sont transférés tels que les opérations distantes spécifiques qui sont appelées et les données (paramètres) qui sont échangées [8].
- Fault : le fault est un élément facultatif défini dans le corps SOAP pour signaler les erreurs [6].

La figure 1-5 affiche la Structure d'un message SOAP.

```

<?xml version='1.0' Encoding='UTF-8' ?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
<env:Header>
<m:reservation xmlns:m="http://travelcompany.example.org/reservation"
env:role="http://www.w3.org/2003/05/soap-
envelope/role/next">
<m:reference>uuid:093a2da1-q345-739r-ba5d-pqff98fe8j7d</m:reference>
<m:dateAndTime>2007-11-29T13:20:00.000-05:00</m:dateAndTime>
</m:reservation>
<n:passenger xmlns:n="http://mycompany.example.com/employees"
env:role="http://www.w3.org/2003/05/soap-
envelope/role/next">
<n:name>Fred Bloggs</n:name>
</n:passenger>
</env:Header>
<env:Body>
<p:itinerary xmlns:p="http://travelcompany.example.org/reservation/travel">
<p:departure>
<p:departing>New York</p:departing>
<p:arriving>Los Angeles</p:arriving>
<p:departureDate>2007-12-14</p:departureDate>
<p:departureTime>late afternoon</p:departureTime>
<p:seatPreference>aisle</p:seatPreference>
</p:departure>
<p:return>
<p:departing>Los Angeles</p:departing>
<p:arriving>New York</p:arriving>
<p:departureDate>2007-12-20</p:departureDate>
<p:departureTime>mid-morning</p:departureTime>
<p:seatPreference></p:seatPreference>
</p:return>
</p:itinerary>
</env:Body>

```

Figure 1-5 : Exemple de Message SOAP [9].

3-3 Utilisation de SOAP dans le HTTP :

Les appels SOAP sont plus susceptibles de passer par le serveur pare-feu, en effet le langage HTTP est généralement compatible avec le port 80, alors que d'autres appels peuvent encore être bloqués pour des raisons de sécurité. Étant donné que les pare-feu autorisent le

passage des requêtes HTTP, les programmes qui communiquent à l'aide de SOAP sont assurés de pouvoir communiquer n'importe où sur le réseau.

3-4 Utilisation de SOAP pour les appels de procédure à distance (RPC) :

SOAP est similaire à l'appel de procédure distante (RPC) utilisé dans de nombreuses technologies telles que DCOM et CORBA, mais il supprime la complexité de l'utilisation de ces interfaces. Il permet aux applications d'appeler des fonctions à partir d'autres applications s'exécutant sur n'importe quelle plate-forme matérielle, quels que soient le système d'exploitation et le langage de programmation [7].

4- Les Services REST :

4-1 Définition de Service REST :

REST est un acronyme pour "Representational State Transfer" créé dans l'article de Roy T. Fielding "An Architecture Style of Networked Systems". REST décrit les propriétés du Web qui ont fait son succès, l'explication de la signification de REST par Roy T. Fielding est la suivante. "Representational State Transfer" évoque une image du fonctionnement d'une application Web bien construite, le fait de cliquer sur un lien (transition entre états) amène l'utilisateur à visualiser la page suivante (représentant le nouvel état de l'application), que l'utilisateur peut utiliser.

REST est un style architectural pas une norme, par conséquent il n'y a pas de spécification de REST. C'est un style d'architecture de réseau de services Web qui met l'accent sur la définition des ressources identifiées par les URI et définit la sémantique de communication client, serveur à l'aide de messages de protocole http [10].

4-2 Les contraintes de REST :

- Client-Serveur : la responsabilité est séparée entre le client et le serveur, l'interface d'utilisateur est distinct de l'interface d'utilisateur de stockage de données, cela permet aux deux de se développer indépendamment.
- Sans état (Stateless) : chaque requête client-serveur doit contenir les informations nécessaires pour permettre au serveur de la comprendre sans dépendre du contexte stocké sur le serveur, cela libère beaucoup d'interaction client-serveur.
- Mise en cache : le serveur envoie une réponse fournissant des informations sur la tendance de cette réponse à être mise en cache sa fraîcheur, sa date de création. Si elle

doit être mise en cache dans le futur cela permet au proxy de réduire la charge sur le serveur et le client pour éviter les requêtes inutiles, et cela améliore également l'évolutivité du serveur.

- Interface uniforme : toutes les ressources sont accessibles via l'interface commune (par exemple, HTTP GET, POST, PUT, DELETE).

La figure 1-6 montre l'architecture de REST.

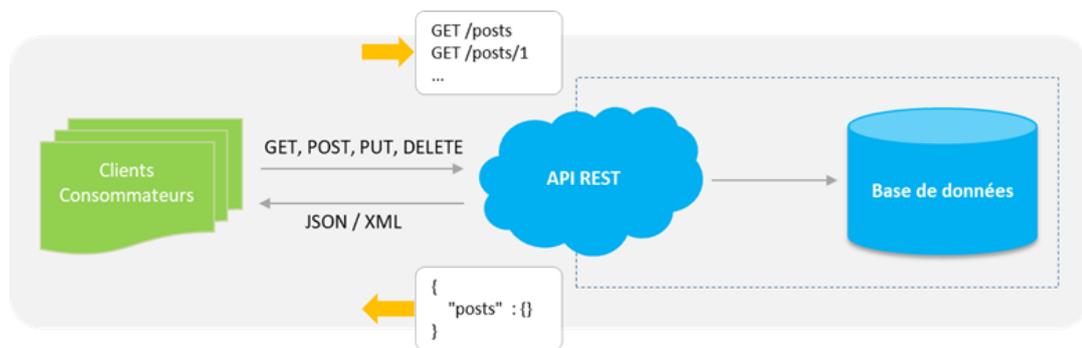


Figure1-6 : L'architecture de REST [11].

- Un système hiérarchisé par couche : les états d'application sont déterminés par les ressources individuelles, toutes les informations ne sont pas envoyées à une seule ressource par conséquent, l'augmentation des demandes, réponses entre le client et le serveur peuvent réduire les performances d'où l'importance de la mise en cache...etc. l'avantage est que cela rend le système de développement plus flexible.
- Code à la demande (facultatif) : exécution de scripts côté client obtenus par le serveur cela rend l'application client plus légère et plus générique, permet d'étendre les fonctionnalités du client en téléchargeant et en exécutant du code en tant qu'application ou script, ce qui simplifie les clients en réduisant le nombre de fonctionnalités qu'ils doivent implémenter par défaut [10].

SOAP VS REST :

1. Le SOAP est un protocole basé sur XML qui fournit des normes pour l'échange d'informations entre plusieurs appareils ou applications. Ainsi que le REST est un style d'architecture qui définit un ensemble de contraintes et de propriétés pour développer des services web.
2. Le REST autorise divers formats de données telles que le texte brut, HTML, JSON. mais le SOAP autorise seulement le format XML.

3. Le SOAP le plus sécurisé par rapport au REST.
4. Le SOAP nécessite plus de bandes passantes et plus de ressources contrairement au REST.
5. Le REST ne pas facile ou flexible comme le SOAP [12].

5-Grands challenge des services web :

5-1 Découverte des SW :

5-1-1 Définition :

La fonction de découverte des services Web est responsable de la recherche. Il est basé sur la description sémantique de ces interfaces. C'est pourquoi le moteur de découverte (en fait du type "Web Service") est extrait de l'ensemble complet des Web Services, un service qui "répond le mieux" aux besoins de cette enquête, " le mieux " Prendre en compte les contraintes de typage sémantique (compatibilité des données...etc.) pas seulement des échanges, mais des restrictions de contexte d'utilisation plus global par exemple la qualité de services (QoS) contexte de qualité de service, politique des utilisateurs et Politique d'utilisation personnelle [13].

La figure 1-7 affiche Les étapes de découverte .

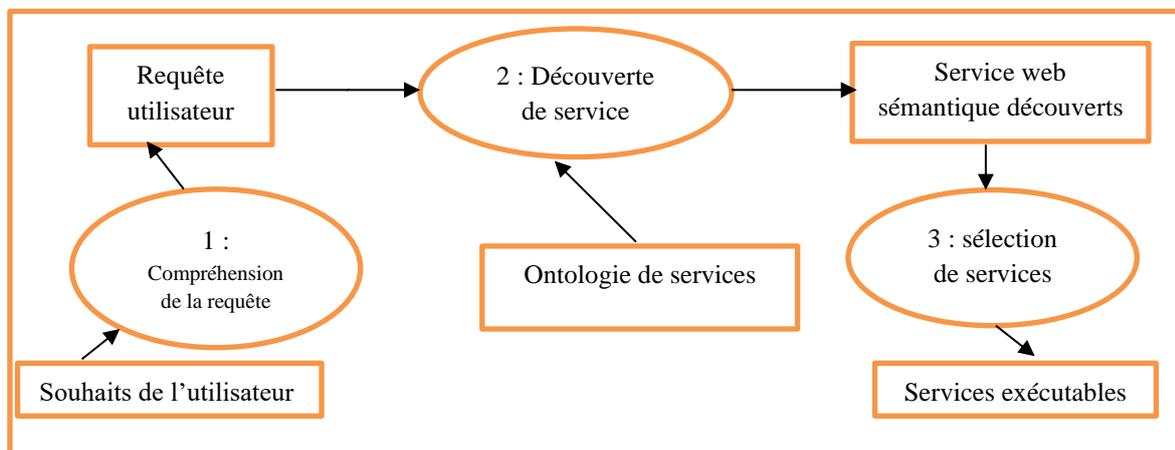


Figure 1-7 : Les étapes de découverte [13].

Le service est conçu pour être sélectionnable à l'aide d'un mécanisme de recherche, la découverte est rendue possible par la description préalable du service et sa publication au registre, cela se fait en deux phases.

- La recherche : cela peut être fait via le registre où le fournisseur expose le service Web.

- La sélection : le client doit sélectionner le service Web qui répond le mieux à ses attentes parmi l'ensemble des services Web obtenus à l'étape de recherche, la recherche et la sélection dans UDDI sont basées sur la publication des services et de leurs fournisseurs. Grâce à UDD les prospects peuvent connaître le fournisseur de services le service fourni par un fournisseur particulier et comment appeler le service. Pour fournir aux clients des réponses à ces questions UDDI divise toutes les informations qu'il contient en trois parties : spécifiés en XML, ces deux éléments peuvent être utilisés pour effectuer une recherche via UDDI [14].

5-1-2 La Découverte sémantique :

La découverte sémantique d'un service Web signifie un raisonnement sémantique sur la base de connaissances. Dans le but de décrire la fonctionnalité de service Web requis en tant que découverte. La découverte sémantique améliore la précision des résultats de recherche par rapport aux techniques traditionnelles de découverte de services Web basées sur des recherches syntaxiques par mots-clés, l'amélioration de la précision des correspondances est coûteuse en termes de puissance de calcul requis, en plus de la précision de sémantique Matchmaking le coût de sémantique Matchmaking a plusieurs aspects qui affectent la conception de l'infrastructure de services Web sémantiques de différentes manières [15].

5-1-3 Les approches :

➤ Approche algébrique :

L'approche de découverte algébrique a ses racines dans les mécanismes algébriques et de recherche d'informations, elle est basée sur le calcul de la similarité du texte à partir d'un graphe structuré dédié ou sur le calcul de la distance (chemin) entre les concepts correspondants. Cette approche appelée algèbre utilise un mécanisme d'appariement structurel, numérique et syntaxique à travers une paire de graphes structurés et en calculant des distances numériques pour vérifier la similarité syntaxique. Pour tirer parti de la sémantique. Ces mécanismes de correspondance utilisent des nombres de termes et des sous-graphes.

➤ Approches déductives :

L'approche déductive est basée sur la logique, les travaux qui choisissent cette catégorie d'approche utilisent la description et les requêtes pour les services spécifiés dans des langages dérivés de formes logiques telles que la logique de description et la logique de prédicat du premier ordre, il utilise également des règles logiques pour rechercher des

services Web et utilise une ontologie pour couvrir les aspects sémantiques. Pour calculer le degré de consensus ils s'appuient sur différentes méthodes prenant en compte la sémantique et ciblant plusieurs éléments descriptifs de service.

➤ **Approche hybride :**

L'approche hybride utilise un mécanisme déductif qui intègre la méthode de calcul de distance. L'idée est de répondre aux limites de chacun de ces deux mécanismes par une combinaison [16].

5-2 Sélection des SW :

5-2-1 Définition :

La sélection de service web consiste à sélectionner le meilleur service parmi un ensemble de services déterminé au regard du descriptif client déclaré à l'aide d'une interface identifiant les préférences pour certifier le service [17].

5-2-2 méthodes de sélection :

La sélection de services Web fait l'objet de quelques travaux et se distingue généralement de trois classes principales (la sélection mono objective, la sélection multi objective, la sélection hybride (mono et multi objective)).

- **La sélection mono objective :** ce type de sélection suppose que k valeurs de QOS (qualité de service) sont agrégées afin d'obtenir un score unique à l'aide d'une fonction objective qui pondère chaque attribut de service. Cette catégorie comprend trois sous-classes : la sélection locale, globale et hybride (global et local).
- **Sélection Multi-Objective :** la sélection multi-objective est une branche de l'optimisation combinatoire, la particularité cherche à optimiser les mêmes plusieurs objectifs (objectif unique pour l'optimisation combinatoire classique) elle se différencie de l'optimisation multidisciplinaire par le fait que les objectifs à optimiser portent sur le même problème.
- **Sélection Mono et Multi-objective :** cette catégorie est basée sur des fonctions de la sélection mono-objective et également des fonctions de la sélection multi-objective [18].

5-3 Composition des SW :

5-3-1 Définition :

La composition des services peut être considérée comme un mécanisme permettant la configuration et la coordination des services existants et l'activation d'applications ou de services à valeur ajoutée, une telle approche présente des avantages significatifs en termes d'économies de coûts et d'efforts de développement, De plus, le service complexe résultant peut être utilisé comme service de base pour de futurs services complexes [19].

La figure 1-8 Explique La Composition de services.

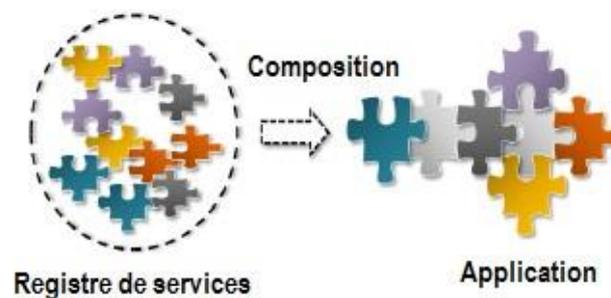


Figure1-8 : La Composition de services

5-3-2 Approches existantes :

Il existe deux approches principales pour configurer les services web : statique et dynamique, on trouve ci-dessous une brève description de chacune de ces approches.

➤ Composition statique des services web :

Avec cette approche. Les services Web qu'on construit sont choisis au moment de l'architecture et de la conception, les composants sont sélectionnés et liés avant d'être compilés et déployés cela fonctionne tant que l'environnement de service web.

➤ Composition dynamique des services web :

Dans cette approche. Le service web à configurer est sélectionné en exécution et la recherche de service est effectuée dans le registre (plateforme de configuration). Pour effectuer une configuration dynamique des services web, StarWSCOP (Star Web Services Composition Platform) on effectue quatre étapes :

- Le fournisseur de services Web publie le service dans le registre.
- StarWSCOP décompose les requêtes des utilisateurs en services abstraits et envoie des requêtes SOAP au registre pour trouver le service approprié.

- Les répertoires de services Web disponibles fournissent une liste de services Web spécifiques.
- StarWSCOP envoie des requêtes SOAP aux services Web trouvés, puis s'y connecte.

5-3-3 Classification des approches de composition de services web :

La composition de service Web nécessite un modèle abstrait pour décrire le flux du service, les approches proposées dans la littérature sont diverses rien n'est reconnu comme une norme en classant ces approches, on peut identifier trois grandes familles :

- Approche sémantique : essentiellement basée sur l'ontologie OWL-S (Web Ontology Language Service) pour les services associatifs.
- Approche industrielle : propose des langages formalisés basés sur XML.
- Approche formelle : utilise des techniques de modélisation de validation de processus (exemple : les réseaux de Petri).

Toutes ces approches permettant de décrire en quelque sorte le flux de services web à générer, ensuite il ne manque plus que les algorithmes de composition des services web et de construction dynamique de ces flux la composante dynamique des services Web concerne essentiellement l'intelligence artificielle et ces techniques de planification. Dans ce qui suit, on décrit les trois approches du composant de service Web mentionnées ci-dessus.

1. Approche sémantique :

Pour explorer une approche sémantique de la composition des services web, la communauté académique a initialement proposé DAML-S (DARPA Agent Markup Language service). L'idée est d'utiliser un langage formel afin de définir ce que fait un service web, puisque WSDL est plus orienté ce qui le rend inadéquat pour la composition de services web, c'est ainsi qu'on a ajouté :

- un profil décrivant la fonctionnalité d'un service.
- un processus décrivant son fonctionnement.
- des descriptions (grounding descriptions) qui font la correspondance de ces composants au WSDL.

Aujourd'hui, OWL-S (web Ontology Language service) a changé pour remplacer DAML-S. on utilise l'ontologie OWL-S pour décrire les flux de services Web agrégés basés sur

l'utilisation de techniques de planification de l'intelligence artificielle, pour rechercher, trier, créer et exécuter des services. OWL-S Étant une ontologie web basée sur OWL qui fournit aux fournisseurs Web un ensemble de balises de base pour décrire les propriétés de ces services web.

2. Approche industrielle :

BPEL4WS (Business Process Execution Language for Web Services) fournit un langage pour spécifier formellement les processus métier et leurs interactions. IBM, Microsoft, a tout contribué au développement de BPEL4WS.

3. Approche formelle :

Le but de l'approche formelle de la configuration des services web est de fournir un mécanisme de validation formelle des services Web complexes, les chercheurs prônant cette approche s'appuient généralement sur une modélisation sous forme de réseaux de Petri, qui sont des modèles de formalisation des processus [20].

6- Conclusion :

Dans ce chapitre nous avons parlé des services web qui sont des programmes informatiques qui activent et échangent des données entre des applications et des systèmes hétérogènes dans un environnement distribué. Avec l'avènement de l'architecture orientée services (SOA) les services Web ont gagné en popularité à mesure que le nombre de services Web augmente, trouver le meilleur service en fonction des besoins des utilisateurs devient un défi. La découverte du service Web sémantique est le processus de recherche du service le plus approprié qui répond à la demande de l'utilisateur et qui doit être développé à l'aide de certaines techniques de machine learning dont nous parlerons dans le chapitre suivant.

Chapitre II : Les techniques du Machine Learning

1-Intoduction :	20
2-Les techniques d'analyse des données :	20
2-1Définition :	20
2-2 L'analyse en composantes principales (ACP) :	21
2-3 La régression linéaire :	21
2-4 La classification :	24
2-5 Clustering :	25
2-6 La réduction de dimension :	26
2-7 L'apprentissage des règles d'association :	27
3- Le machine learning :	27
3-1 Définition :	27
3-2 Les Algorithmes de machine learning :	28
4- Deep learning :	33
4-1 Introduction :	33
4-2 Les réseaux de neurones artificiels :	34
4-3 L'architecture d'un réseau de neurone artificiel :	34
5- Conclusion :	38

1-Introduction :

Le Machine Learning (ML) est l'étude scientifique des algorithmes et des modèles statistiques que les systèmes informatiques utilisent pour effectuer des tâches spécifiques sans être explicitement programmées, les algorithmes d'apprentissage dans de nombreuses applications que nous utilisons tous les jours, chaque fois qu'un moteur de recherche Web comme Google est utilisé pour effectuer des recherches sur Internet, l'une des raisons pour lesquelles il fonctionne si bien est que l'algorithme d'apprentissage a appris à classer les pages web ces algorithmes sont utilisés à des fins diverses tels que l'exploration de données, le traitement d'images et l'analyse prédictive, le principal avantage d'utiliser l'apprentissage automatique c'est qu'une fois que l'algorithme a appris à traiter les données il fait son travail automatiquement.

Depuis l'évolution, les êtres humains ont utilisé plusieurs types pour effectuer différentes tâches plus facilement, la création du cerveau humain a conduit à l'invention de différents types de machines. Ces machines ont grandement facilité la vie humaine en permettant à l'homme de répondre à divers besoins de la vie tels que les voyages, les recommandations et les sites de commerce électronique.

L'apprentissage automatique s'appuie sur divers algorithmes pour résoudre les problèmes de données. Les scientifiques des données concluent qu'il n'y a pas un seul type d'algorithme qui soit le mieux adapté pour résoudre un problème, le type d'algorithme utilisé dépend du type de problème qu'on résolve [21].

2-Les techniques d'analyse des données :**2-1 Définition :**

Grâce à la technologie numérique, les entreprises d'aujourd'hui ont accès à de vastes quantités de données et la compréhension et l'analyse sont un atout précieux.

L'analyse des données est un processus de nettoyage, de transformation et de modélisation des données qui vise à extraire des informations exploitables pour prendre de meilleures décisions au sein de l'entreprise grâce à l'intelligence d'affaires. Il peut être utilisé dans le cadre d'une étude de marché [22].

Cela peut inclure les données suivantes : Journaux du serveur web et données des itinéraires de navigation sur Internet, rapports de contenu et d'activité des réseaux sociaux, textes et réponses aux sondages extraits des e-mails des clients [23].

Il existe de nombreux outils d'analyse de données qui facilitent le traitement et la manipulation de leurs données par les utilisateurs, on peut également utiliser ces outils pour analyser les relations et les corrélations entre les ensembles de données et pour trouver des tendances et des modèles. Il y a beaucoup de "Big data" récurrentes des outils qui peuvent être nommés comme le langage de programmation Python, les logiciels Spark Talend et Apache [22].

Au sens français, le terme analyse de données désigne un sous-ensemble de données dites multivariées ceci est essentiellement inclus [24].

2-2 L'analyse en composantes principales (ACP) :

L'analyse en composantes principales (ACP) est une méthode de la famille d'analyse de données multivariées les plus utilisées [25]. Elle permet de transformer des variables liées entre elles en statistique en nouvelles variables indépendantes les unes des autres. Ces nouvelles variables sont appelées "composantes principales", ou axes, elle permet au praticien de diminuer l'information en dans nombre de composantes plus limitées que le nombre d'origine de variables [24].

L'analyse en composantes principales a plusieurs applications notamment :

- Étudier et visualiser des corrélations entre les variables, afin d'éventuellement limiter le nombre de variables à mesurer par la suite.
- Prendre de facteurs non corrélés qui sont des combinaisons linéaires des variables de départ, afin d'utiliser ces facteurs dans des méthodes de modélisation telles que la régression linéaire, la régression logistique ou l'analyse discriminante.
- Représenter des observations dans un espace à deux ou trois dimensions, afin d'identifier des groupes homogènes d'observations, ou au contraire des observations atypiques.

2-3 La régression linéaire :

L'analyse de régression linéaire, également plus simplement appelée régression linéaire est l'une des méthodes statistiques les plus utilisées en sciences appliquées, sciences humaines et sociales [26]. Se classe parmi les méthodes d'analyse multivariées qui traitent des

données quantitatives. La régression linéaire est une méthode d'investigation sur les données d'observation ou d'expérimentations, où l'objectif principal est de rechercher une liaison linéaire entre une variable Y quantitative et une ou plusieurs variables X également quantitative [27].

Les liaisons entre les variables exercent une influence très importante sur l'efficacité de la méthode [26].

La régression linéaire est la plus utilisée pour deux raisons majeures :

- c'est une ancienne méthode.
- c'est l'outil de base de la plupart des modélisations plus sophistiquée comme la régression logistique, le modèle linéaire généralisé et les méthodes de traitement des séries temporelles [27].

2-3-1 La régression linéaire simple :

Dans une régression linéaire simple, le modèle de régression établit une relation entre la variable d'entrée indépendante (x) et la variable de sortie continue (y), par conséquent le modèle doit commencer par la fonction hypothétique suivante :

$$y = \theta_0 + \theta_1 x \quad (1)$$

La valeur θ_0 correspond à l'intersection 0 et la valeur θ_1 correspond au coefficient de x , dans le cas d'un apprentissage supervisé, il est prévu que les valeurs de (x) et (y) soient données lors de l'entraînement du modèle, Le modèle détermine la meilleure ligne droite, afin de prédire la valeur de (y) en fonction de la valeur de (x) donnée. Pour cela, le modèle doit identifier les meilleures valeurs pour θ_0 et θ_1 , tout en itérant à travers les données d'entraînement. Afin de déterminer les valeurs optimales pour θ_0 et θ_1 . Ensuite évaluer la performance du modèle, on peut utiliser des mesures d'évaluation la plus répandue. Pour mesurer la performance d'un modèle de régression est la racine de l'erreur quadratique moyenne ou REQM qui correspond à la racine carrée de la moyenne des erreurs quadratiques. L'équation est la suivante :

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (2)$$

- n est le nombre total de point de données.
- y_i la valeur de sortie réelle.
- \hat{y}_i la valeur de sortie prédite.

La figure 2-1 explique La régression linéaire simple.

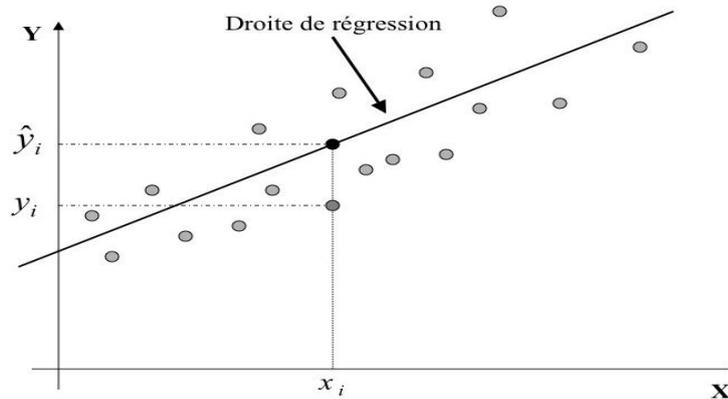


Figure 2-1 : La régression linéaire simple [28].

Une fois la valeur optimale déterminée, le modèle doit pouvoir associer linéairement à la moindre erreur le plus petit REQM.

La régression linéaire simple est le modèle le plus utilisé en finance. Il peut être utilisé pour la gestion de portefeuille, l'évaluation des actifs et l'optimisation. Ainsi on a besoin d'un modèle pour évaluer des situations un peu plus complexes, c'est pour ça on utilise la régression linéaire multiple.

2-3-2 La régression multiple :

Dans la régression multiple, la variable de sortie (y) qu'on essaye de prédire dépend de plusieurs variables, par conséquent on a besoin d'un modèle plus sophistiqué pour expliquer cette dimension supérieure, on peut utiliser un grand nombre de variables indépendantes pour améliorer la précision de modèle tant que les variables ajoutées sont pertinentes pour ce modèle, par exemple un modèle de régression multiple basé sur les trois variables indépendantes présentées dans l'équation suivante :

$$y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$$

De la même manière que pour les mesures de régression linéaire peuvent être utilisées pour déterminer la performance optimale [29].

2-4 La classification :

La classification est une technique de machine learning qui utilise des données connues pour déterminer comment les nouvelles données doivent être classées dans un ensemble de catégories existantes.

Lors de classification d'un ensemble de données donné, le système de classification effectue les actions suivantes :

- premièrement, un nouveau modèle de données est préparé à l'aide de l'un des algorithmes d'apprentissage.
- ensuite, faire des tests sur le modèle préparé.
- plus tard, ce modèle de données est utilisé pour examiner les nouvelles données et déterminer leur classe.

La classification à autre nom, c'est la catégorisation, aussi est une technique de machine learning qui utilise des données connues pour déterminer comment les nouvelles données doivent être classées dans un ensemble d'étiquettes, classes et catégories existantes.

Dans les tâches de classification, le programme doit apprendre à prédire des valeurs discrètes pour les variables dépendantes ou de sortie à partir d'une ou plusieurs variables indépendantes la valeur prédite par le programme, c'est une classe, catégorie ou l'étiquette la plus probable pour les nouvelles observations [30].

La classification est une forme d'apprentissage supervisé, on distingue deux types de classification, la classification binaire et la classification multiclasse.

- **Classification binaire :** c'est un problème d'apprentissage supervisé. Dans lequel l'espace des étiquettes est binaire $y = \{0,1\}$ elle indique l'appartenance à une classe [31]. On va citer un exemple : les fournisseurs de services des messageries telles que Yahoo, Gmail...etc. utilisent cette technique pour classer un nouveau courrier comme spam ou non, l'algorithme de classification s'entraîne en analysant le comportement de l'utilisateur consistant à marquer certains e-mails comme spams sur la base de ces informations, le classificateur décide si un nouveau courrier doit être placé dans la boîte de réception ou dans le dossier spam [30].

- **Classification multi-classes :** c'est un problème d'apprentissage supervisé. Dans lequel l'espace des étiquettes est discret $y = \{1, 2, \dots, C\}$ et correspondant donc à plusieurs classes, par exemple (Identifier à quelle espèce appartient à une plante, identifier en quelle langue un texte est écrit...) [31].

La Figure 2-2 explique le résultat de classification.

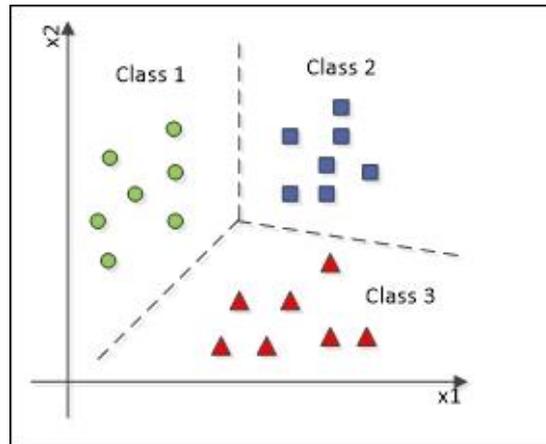


Figure 2-2 : Figure explique le résultat de classification.

2-5 Clustering :

Le clustering est un type de méthode d'apprentissage non supervisé de machine learning. Dans la méthode de l'apprentissage non supervisé les inférences sont tirées des ensembles de données qui ne contiennent pas de variable de sortie étiquetée. Il s'agit d'une technique d'analyse de données exploratoire qui nous permet d'analyser les ensembles de données multivariées.

Le clustering consiste à diviser les ensembles de données en un certain nombre de clusters où les points de données appartenant à un cluster ont des caractéristiques similaires, les clusters ne sont rien d'autre que le regroupement de points de données de sorte que la distance entre les points de données au sein des clusters soit minimale [32].

Le moteur de clustering parcourt complètement les données d'entrée et en fonction des caractéristiques des données il décidera sous quel cluster elles doivent être regroupées.

La Figure 2-3 illustre Une visualisation de résultat de Clustering

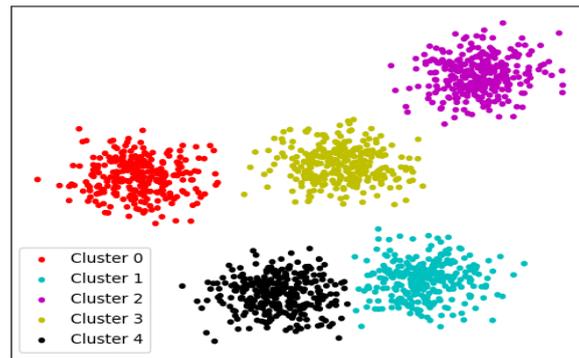


Figure 2-3 : Une visualisation de résultat de Clustering.

Les Algorithmes de clustering :

Les algorithmes les plus utilisés dans clustering sont : Clustering canopy et le Clustering K-means.

- **Clustering canopy** : Le clustering canopy est un algorithme de pré-clustering non supervisé qui est souvent utilisé comme étape de prétraitement pour l'algorithme K-means de clustering hiérarchique, il est utilisé pour accélérer les opérations de clustering sur des grands ensembles de données où l'utilisation directe d'un autre algorithme peut ne pas être possible en raison de la grande taille des ensembles de données [30].
- **Clustering K-means** : K-Mean Clustering est une technique d'apprentissage non supervisé ayant un processus itératif qui vise à partitionner l'ensemble de données K en sous-groupes (Clusters) distincts et non prédéfinis où chaque point de données appartient à un seul groupe, il essaie de rendre les points de données intra-cluster similaire tout en gardant les clusters aussi différents (loin), il attribue des points de données à un cluster de sorte que la somme de la distance au carré entre les points de données et le centroïde du cluster (moyenne arithmétique de tous les points de données appartenant à ce cluster) soit au minimum [33].

2-6 La réduction de dimension :

La taille des données peut être mesurée selon deux dimensions : le nombre de variables et le nombre d'échantillons. Ces deux dimensions peuvent être très importantes et peuvent poser des problèmes lors de l'examen et de l'analyse de ces données, il est donc indispensable de mettre en place un outil informatique qu'on veut du mieux comprendre. La réduction de dimensionnalité est l'une des approches les plus anciennes permettant d'apporter un élément de réponse à cette question, l'objectif est de sélectionner le sous-ensemble optimal

de caractéristiques associées aux critères précédemment définis, le choix d'un sous-ensemble de cette fonctionnalité en fonction des critères qu'on veut utiliser élimine les informations non pertinentes et détaillées, par conséquent cette sélection peut réduire les dimensions de l'espace échantillon et rendre le jeu de données plus représentatifs du problème [34].

La réduction de dimension consiste à retrouver une représentation donnée dans un espace de dimensions inférieures à l'espace dans lequel elles étaient représentées à l'origine, ce qui permet de réduire le temps de calcul et l'espace nécessaire pour stocker les données, mais aussi d'améliorer un algorithme d'apprentissage supervisé entraîné par ces données [31].

On classe les techniques mathématiques de réduction de dimension en deux grandes catégories, la sélection de variables et l'extraction de trait [34].

2-7 L'apprentissage des règles d'association :

Le processus de fouille des données par la recherche de règles d'association se base sur une classe particulière de motifs appelés « itemsets » fréquents ou conjonction d'items fréquents, en s'appuyant sur la particularité de fréquence des itemsets. La technique consiste à mettre en évidence des règles de la forme prémisse (antécédent) implique une conclusion (conséquence). Les règles d'association expriment alors à partir des données de la base des données relationnelles la tendance à impliquer les propriétés de la prémisse et celles qui apparaissent à la conclusion, l'implication étant non mathématique comme la direction de tendance de la conclusion est vraie " lorsque la prémisse est " vrai", la relation implicite donnée par les connexions entre les données donne alors une relation de cause à effet qui produira une inférence jusqu'ici coulée dans l'ensemble de données[35].

3-Le machine learning :

3-1 Définition :

Le machine learning (ML), également appelé apprentissage automatique. En français est un type d'intelligence artificielle qui peut créer des systèmes qui peuvent être appris sans être programmés pour effectuer des tâches spécifiques ou améliorer les performances en fonction des données en cours de traitement [36]. Cette technologie permet le développement de programmes informatiques lorsqu'ils sont exposés à de nouvelles données [37].

Pour comprendre l'apprentissage automatique et son fonctionnement, nous donnons un exemple comme un humain faisant du vélo : au début, on y arrive pas du tout, mais à force d'en faire et bien, on y arrive de mieux en mieux jusqu'au moment où on en fait super bien.

Cet exemple fonctionne bien avec la définition du machine learning donnée par American Tom en 1998, selon lui, une machine apprend lorsque sa **performance P** faire une **T Tâche** s'améliore grâce à une nouvelle **E Expérience**.

- **Tâche T** : faire du vélo.
- **performance P** : rouler droit, ne pas tomber.
- **Expérience E** : chaque fois que l'on fait du vélo, y compris les fois où l'on tombe.

Apprendre à faire T lorsque en améliore P grâce à E [38].

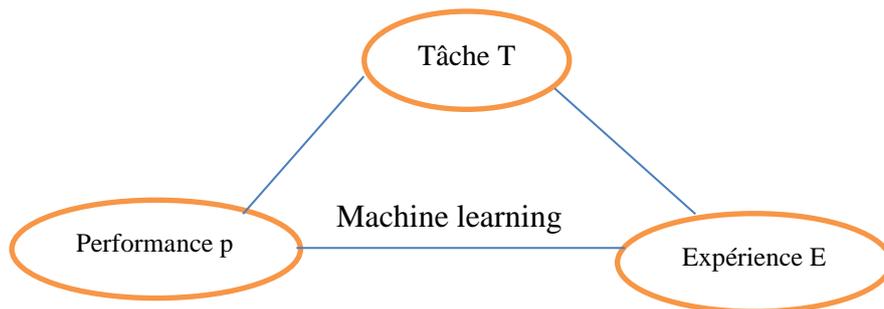


Figure 2-4 : Le fonctionnement de Machine Learning [38].

3-2 Les Algorithmes de machine learning :

Les algorithmes sont les moteurs de l'apprentissage automatique. Ces algorithmes créent des modèles en interagissant avec ces données en fonction de la manière de création du modèle ainsi que la représentation des données d'apprentissage, en présentant les principes algorithmes : l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage par renforcement. La figure suivante affiche Les algorithmes de Machine Learning.

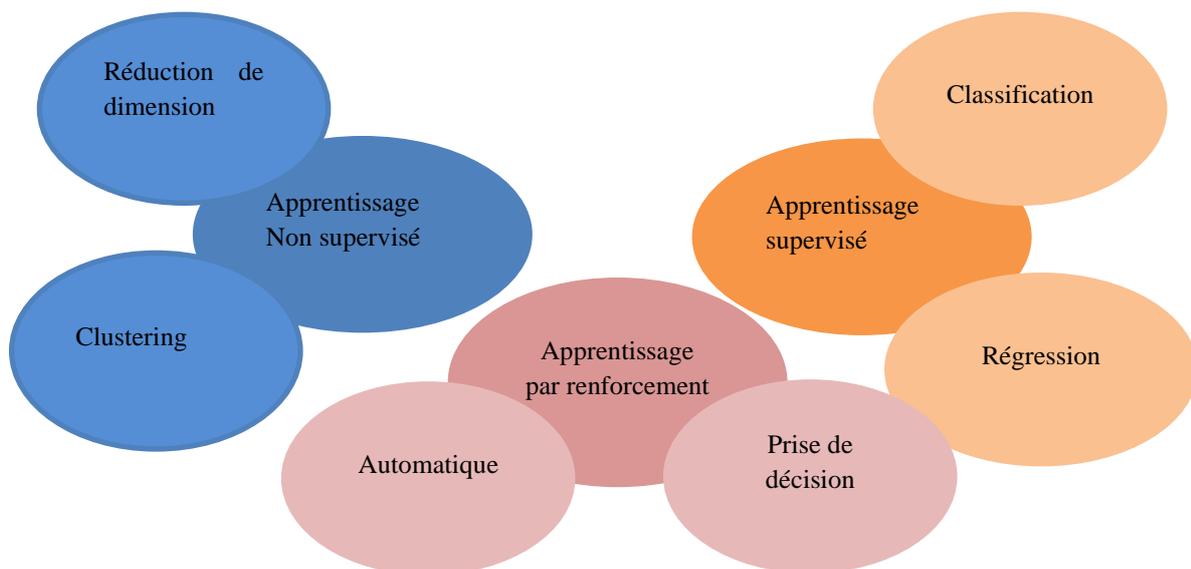


Figure 2-5 : Les algorithmes de Machine Learning [38].

3-2-1 L'apprentissage supervisé :

la première classe de machine learning est nommée les algorithmes d'apprentissage supervisé sont une classe d'algorithmes d'apprentissage automatique qui utilisent un ensemble de données étiquetées (x) afin de classer des données non étiquetées (y) similaires, les données étiquetées sont des données qui ont déjà été classées, tandis que les données non étiquetées sont des données qui n'ont pas encore été étiquetées, les étiquettes de ces derniers peuvent être soit discrètes soit continués [39].

Les algorithmes supervisés sont les algorithmes qui ont besoin d'une aide extérieure, le jeu de données d'entrée est divisé en données d'apprentissage et en données de test, cet ensemble de données de train a une variable de sortie qui doit être prédite ou classifiée, tous les algorithmes apprennent des modèles spécifiques à partir de l'ensemble de données d'apprentissage et les appliquent à l'ensemble de données de test pour la prédiction ou la classification [21].

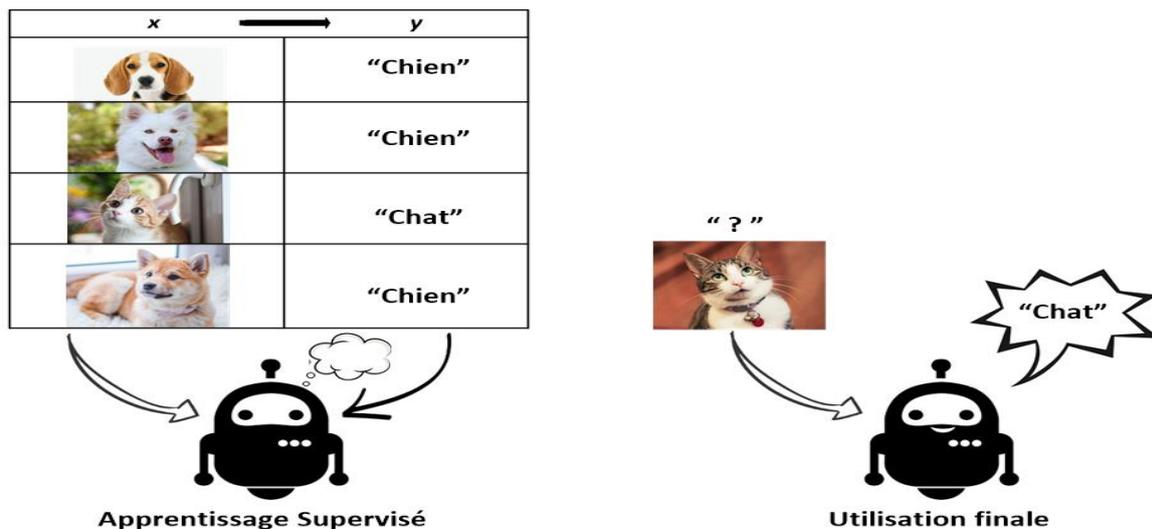


Figure 2-6 : l'apprentissage supervisé [38].

Les algorithmes d'apprentissage supervisé :

On trouve ici quelques algorithmes de l'apprentissage supervisé tels que, support Vector Machine, K-Nearest Neighbors.

1-Support Vector Machine (SVM) : SVM est un algorithme d'apprentissage automatique supervisé utilisé pour la régression et la classification, l'objectif de cet algorithme est de trouver un hyperplan dans un espace N dimensions qui classe distinctement les points de

données, la dimension de l'hyperplan dépend du nombre d'entités, si le nombre de ces derniers est des deux l'hyperplan n'est qu'une ligne et si le nombre d'entrées est des trois l'hyperplan devient un plan 2D. Donc il devient difficile d'imaginer quand le nombre de fonctionnalités dépasse trois [40].

2-K-Nearest Neighbors (KNN) : KNN est un type d'algorithme d'apprentissage supervisé utilisé pour la régression et la classification essaie de prédire la classe correcte pour les données de test, en calculant la distance entre les données de test et tous les points d'apprentissage sélectionnent ensuite le nombre K de points qui est le plus proche des données de test, cet algorithme calcule la probabilité que les données de test appartiennent aux classes de données d'apprentissage "K" et que la classe détient la probabilité la plus élevée qui sera sélectionnée [41].

La Figure 2-7 explique L'Algorithme de K-Nearest Neighbors



Figure 2-7: Algorithme de K-Nearest Neighbors [41].

3-2-2 L'apprentissage non supervisé :

La deuxième classe d'algorithmes de machine learning est l'apprentissage non supervisé, ici on n'étiquette pas les données à l'avance, mais à la place on laisse l'algorithme arriver à cette conclusion, l'un des exemples les plus courants et peut-être les plus simples de l'apprentissage non supervisé est le regroupement, c'est la technique qui tente de séparer les données en sous-ensembles [39].

On appelle ce type de machine learning par l'apprentissage non supervisé, car il n'y a pas de réponse correcte ni d'enseignant, les algorithmes sont laissés à leurs propres mécanismes pour découvrir et présenter la structure intéressante des données [42].

La Figure 2-8 affiche L'apprentissage supervisé vs l'apprentissage non supervisé



Figure 2-8 : L'apprentissage supervisé vs l'apprentissage non supervisé [38].

Deep Learning utilise également des techniques non supervisées bien que différentes du clustering. Dans le traitement du langage naturel (NLP) on utilise des algorithmes non supervisés pour la représentation vectorielle des mots la meilleure façon de le faire s'appelle Word2Vec.

Une autre application intéressante de l'apprentissage non supervisé est dans les modèles génératifs, par opposition aux modèles discriminatifs on forme un modèle génératif avec une grande quantité de données d'un certain domaine comme l'image ou le texte, et le modèle essaiera de générer de nouvelles données similaires à celui que nous avons utilisé pour la formation [39].

3-2-3 L'apprentissage par renforcement :

L'apprentissage par renforcement la troisième technique de machine learning. Grâce à cette technique l'algorithme multiplie les tentatives pour tenter de découvrir quelles actions apportent les plus grandes récompenses. Ce type d'apprentissage reclasse trois principaux composants : **l'agent** (qui apprend ou prend les décisions), **l'environnement** (tout cet avec quoi l'agent interagit), et **les actions** (ce que peut faire l'agent), l'objectif est que l'agent choisisse les actions qui maximisent les récompenses attendues sur une période donnée, l'agent atteindra ce but plus rapidement en suivant des règles bien établies [37].

L'apprentissage par renforcement utilisé pour la robotique, le jeu vidéo, la navigation, les voitures autonomes, et les drones...Etc.

La Figure 2-9 explique Le fonctionnement de L'apprentissage par renforcement

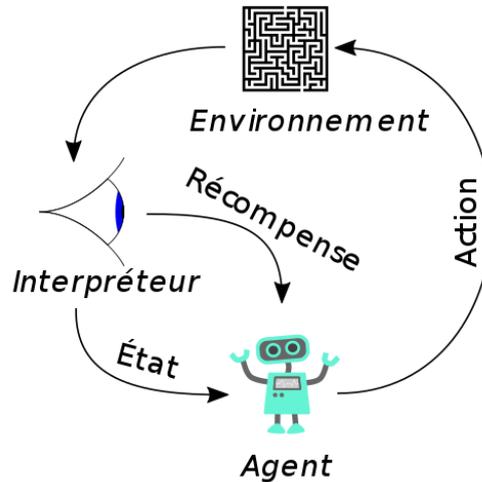


Figure 2-9 : Le fonctionnement de L'apprentissage par renforcement [43].

Les algorithmes de l'apprentissage par renforcement :

Les algorithmes populaires de cette technique sont la recherche de Q-Learning, Temporal Difference Learning et Monte Carlo [38].

1-Q-Learning :

Q-Learning est un algorithme de l'apprentissage par renforcement est à la fois simple, il fonctionne pour résoudre un problème basique qui cherche à trouver la meilleure action à entreprendre compte tenu de l'état actuel, il est considéré comme hors politique, parce que l'algorithme de Q-learning apprend des actions qui sont en dehors de la politique actuelle comme prendre des actions aléatoires. Le "Q" de Q-learning est synonyme de qualité, la qualité représente l'utilité d'une action donnée pour obtenir une récompense future [44].

2-Temporal Difference Learning (TD) :

L'apprentissage par différence temporelle l'un des algorithmes d'apprentissage par renforcement permet à un agent d'apprendre de l'expérience d'un agent sans s'appuyer sur la connaissance de la dynamique de l'environnement, comme les méthodes de programmation dynamique, la méthode TD utilise le bootstrap, c'est-à-dire que les valeurs d'états estimées dépendent des valeurs des états futurs dans la trajectoire [45].

3-Monte Carlo (MC) :

Monte-Carlo, c'est un algorithme de l'apprentissage par renforcement consiste à effectuer des simulations au hasard à retenir les résultats des simulations et à calculer des moyennes de résultats, la technique du Monte Carlo est adaptée lorsque l'on ne connaît pas le modèle du domaine, c'est une stratégie simple, il faut que les tâches de l'agent soient décomposées en épisodes et un épisode correspondra à une simulation [46].

4- Deep Learning :**4-1 Introduction :**

Le Deep learning est issu de la recherche sur l'intelligence artificielle et Machine Learning, qui se concentre sur la création de grands modèles de réseaux de neurones capables de prendre des décisions précises basées sur des données. Le Deep Learning est particulièrement adapté aux contextes où les données sont complexes et où de grands ensembles de données sont disponibles, aujourd'hui, la plupart des entreprises en ligne et des technologies grands public haut de gamme utilisent le Deep Learning entre autres, Facebook utilise l'apprentissage en profondeur pour analyser le texte dans la conversation en ligne, Google et Microsoft utilisent tous l'apprentissage en profondeur pour la recherche d'images et également pour la traduction automatique, tous les téléphones intelligents modernes sont équipés de systèmes d'apprentissage en profondeur. Par exemple, l'apprentissage en profondeur est désormais une technologie standard pour la reconnaissance vocale, ainsi que pour la détection des visages sur les appareils photo numériques [47].

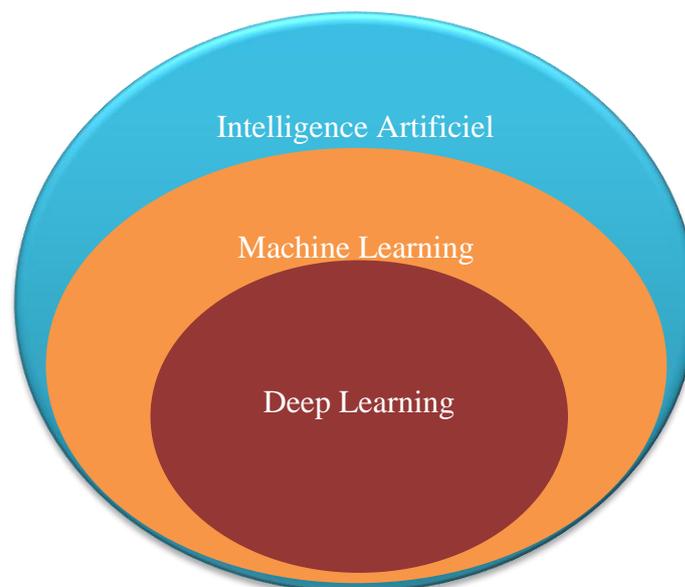


Figure 2-10 : Une visualisation de Deep Learning.

4-2 Les réseaux de neurones artificiels :

4-2-1 Définition :

Les réseaux de neurones trouvent leurs racines dans les tentatives de modélisation mathématique du cerveau humain, la première œuvre remonte en 1943, les travaux de M. Culloch et W. Pitts supposent que les impulsions neuronales sont le résultat de calculs simples effectués par chaque neurone et que la pensée est provoquée par l'action collective des réseaux interconnectés de neurones, les informations sont distribuées et stockées dans le réseau sous forme de coefficients synaptiques ou de fonctions d'activation.

Les réseaux de neurones ne sont pas programmés et sont entraînés par un mécanisme d'apprentissage, les tâches particulièrement adaptées au traitement des réseaux de neurones sont l'association, la classification, l'identification, la prédiction ou l'estimation et le contrôle du processus complexes [48].

4-3 L'architecture d'un réseau de neurone artificiel :

L'architecture d'un réseau de neurones c'est l'organisation de plusieurs neurones entre eux au sein d'un même réseau, la plupart des réseaux de neurone utilisent le même type de neurones, l'architecture dépend à la tâche à apprendre, un réseau de neurone se compose de couche d'entrées appelée (input layer), la couche de sortie (output layer) et des couches cachés (hidden layer) [48], la figure suivante affiche l'architecture d'un réseau de neurone artificiel.

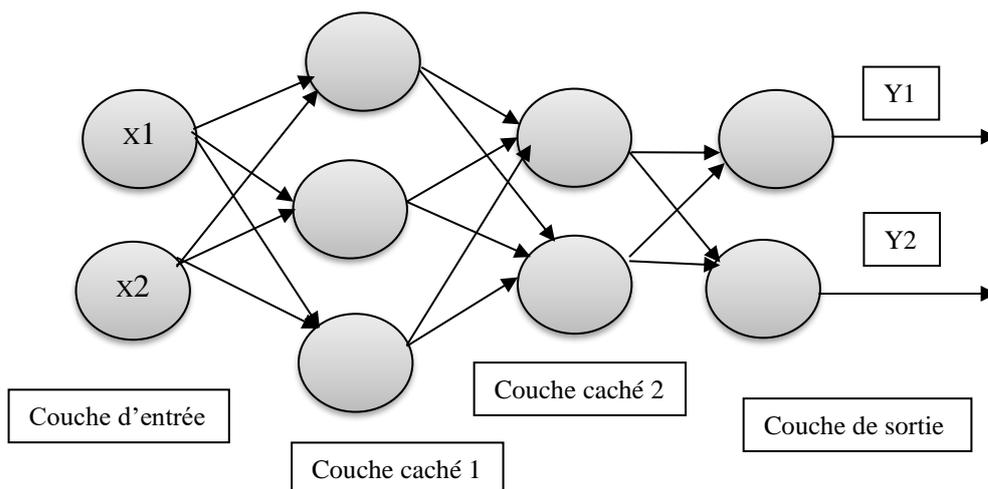


Figure 2-11: L'architecture d'un réseau de neurone artificiel [48]

4-4 Les modèles de réseau de neurone :

4-4-1 Le perceptron multicouche (MLP) :

Les perceptrons multicouches sont des réseaux directionnels de neurones artificiels organisés en couches déplaçant les informations dans une seule direction de la couche d'entrée à la couche de sortie, les neurones sont reliés les uns aux autres par des connexions pondérées en particulier les poids de ces connexions, il s'agit d'une opération de réseau qui utilise une transformation linéaire non pour programmer le mappage de l'espace d'entrée vers l'espace de sortie [49]. On trouve le MLP dans nombreuse application pratique telles que la reconnaissance de fleurs, la détection de fraudes, il est utilisé dans les tâches de classification supervisées, on peut dire que MLP est l'un des modèles les plus populaires est implémenté par de nombreuses bibliothèques comme TensorFlow, Scikit-Learn...Etc. [50].

La Figure 2-12 explique L'architecture de Perceptron multicouche (MLP)

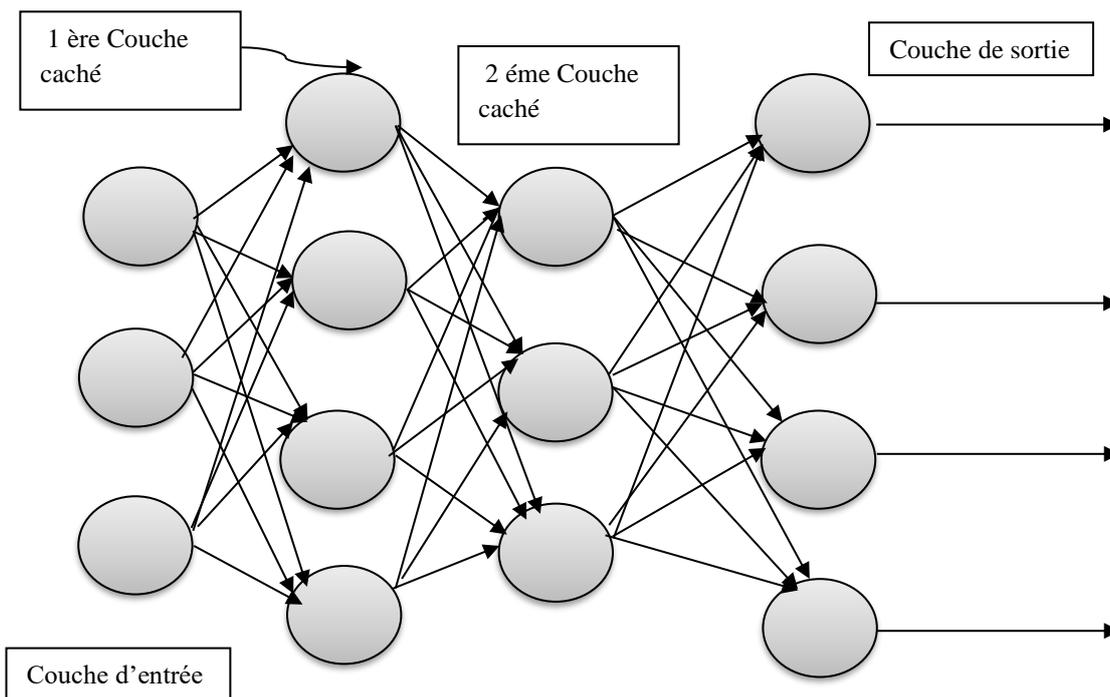


Figure 2-12 : L'architecture de Perceptron multicouche (MLP) [50].

4-4-2 Les réseaux de neurones convolutif (CNN):

Les réseaux de neurones convolutifs sont un sous-type de réseaux de neurones artificiels qui sont biologiquement inspirés par le cortex visuel, ce dernier contenant de petites zones de cellules sensibles à des zones spécifiques du champ visuel.

Il y a deux parties, la première partie est appelée la convolution du modèle (CNN) et la deuxième partie s'appelle la partie de classification d'appariement de modèle (MLP) [51].

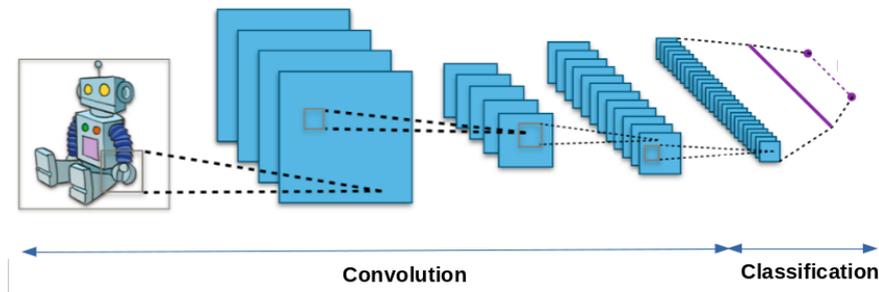


Figure 2-13 : Architecture d'un réseau de neurone convolutif [51].

4-4-3 Les réseaux de neurone récurrent (RNN) :

Les réseaux de neurones récurrents sont un type de réseau de neurones largement utilisé dans l'apprentissage en profondeur. Le RNN est idéal pour le traitement de données séquentielles en utilisant la sortie précédente comme entrée supplémentaire [54].

Contrairement au MLP, les réseaux de neurones récurrents comportent des cycles au sein du graphe de neurones, ce type d'architecture et de pouvoir manipuler des séquences des vecteurs d'entrées représentant chacun un événement temporel en déroulant par rapport au temps. Le RNN peut être considéré comme une suite temporelle de réseaux MLP reliés entre eux à travers leurs couches cachées respectives [53].

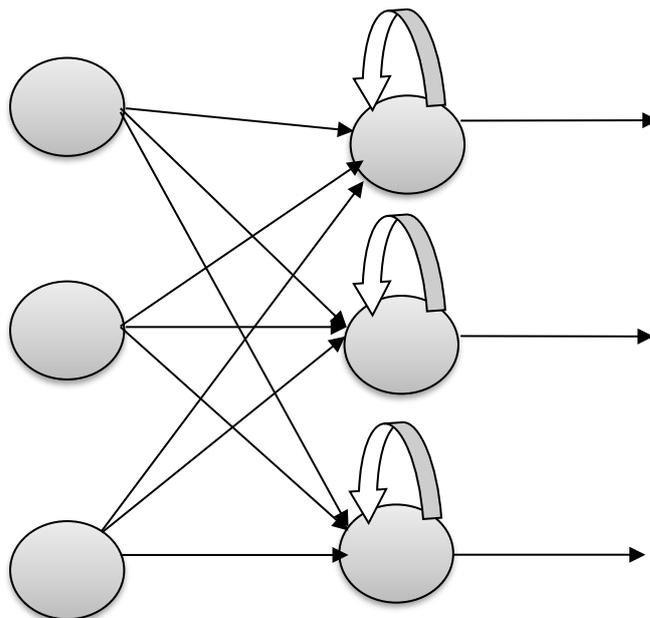


Figure 2-14 : l'architecture des réseaux de neurone récurrent [52].

Les réseaux de neurones récurrents à une relation sur l'état caché comme le schéma illustré dans la figure 2.14, cette contrainte de bouclage assure la capture d'informations séquentielles dans les données d'entrées [52].

L'explosion du gradient :

Les réseaux de neurones récurrents sont difficiles à gérer des séquences relativement longues contenant plus de 10 événements, en fait, les calculs cumulés à long terme réduisent les erreurs obtenues avec la rétropropagation du gradient ou rarement augmentent de façon exponentielle par rapport à l'échelle de temps, cette dernière appelée explosion de gradient existe dans une architecture sans boucle et se dégrade avec le nombre de couches [53].

Parmi les techniques utilisées pour atténuer le problème d'explosion du gradient le LSTM (Long short Term Memory) et GRU (Gated Recurrent Unit).

Long Short Term Memory (LSTM) :

L'une des solutions les plus efficaces pour résoudre le problème d'explosion de gradient Long Short Term Memory est une classe de RNN qui a la même architecture et la même formulation mathématique générale [53].

La figure suivante affiche Unité de base LSTM.

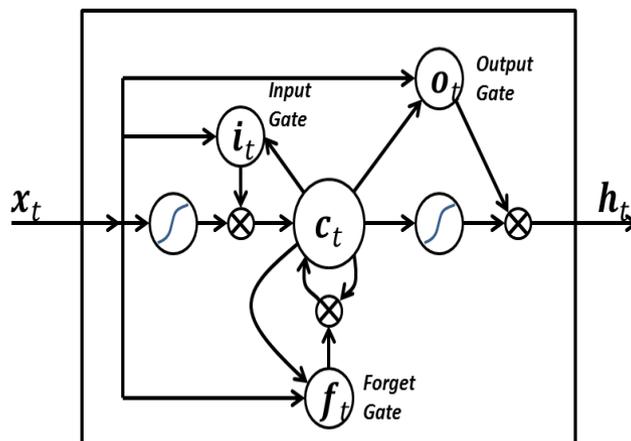


Figure 2-15 : Unité de base LSTM [54].

L'unité LSTM est illustrée dans la figure 2.15 est le composant de base d'une architecture LSTM. C'est une série de portes et de cellules qui coopèrent pour produire un résultat final.

5- Conclusion :

Pour conclure, le Machine Learning ou apprentissage automatique est la principale technique de l'intelligence artificielle, il consiste à entraîner des algorithmes à partir de base d'apprentissage pour leur permettre de faire des prédictions ou automatiser des tâches telles que la classification des textes,

Dans le chapitre suivant nous discuterons de classification des textes à l'aide des algorithmes de Deep Learning avec quelques travaux connexes.

Chapitre III : Classification de texte pour la découverte des Services Web

1 -Introduction :.....	40
2- Les Algorithmes Deep learning pour la classification de texte :	40
3- Quelques travaux connexes pour la classification des services web :	49
4 – Conclusion.....	51

1 -Introduction :

La découverte de services est le processus de recherche de services appropriés à partir d'un ensemble de services qui peuvent être fournis dans un référentiel de services central ou distribué, l'objectif principal de la technologie des services Web est de fournir une intégration et une interaction dynamique de systèmes hétérogènes, la principale limitation de cette technologie est que la découverte et la composition des services nécessitent toujours une intervention humaine accrue, cela pose un obstacle d'autant plus que la charge du service Web augmente. Pour résoudre ce problème la communauté du Web sémantique propose d'enrichir le contenu sémantique des services Web et leurs capacités grâce à des techniques d'apprentissage automatique telles que la classification pour découvrir sémantiquement les services [55].

Vu la nature de description des services web de notre Dataset, voir le chapitre (04) la section (05) on va traiter les algorithmes deep learning pour la classification de texte.

2- Les Algorithmes Deep learning pour la classification de texte :

La classification automatique de texte est un sujet classique du traitement automatique du langage (TAL) et consiste à attribuer des catégories prédéfinies à des documents en texte libre pour une tâche particulière et essaye d'identifier le meilleur algorithme de classification et de définir les meilleures caractéristiques à utiliser en entrée de ces classificateurs, ces modèles sont souvent appliqués avec succès à des données avec des caractéristiques dimensionnelles clairsemées, récemment les méthodes d'apprentissage en profondeur se sont révélées très efficaces et cette tendance est confirmée par le succès de l'expression continue des mots (Word embedding)[56]. L'architecture Next Deep Learning fonctionne avec une très grande précision en utilisant une ingénierie et un calcul de bas niveau, ce qui apporte d'énormes avantages pour la classification de texte [57].

Parmi ces algorithmes on cite :

1-Convolutional Neural Networks (CNN) :

Les réseaux de neurones convolutifs (CNN) sont une autre architecture d'apprentissage en profondeur utilisée pour la classification hiérarchique des documents conçus à l'origine pour le traitement d'images avec une architecture similaire au cortex visuel. Les CNN sont également utilisés efficacement pour la classification de texte, dans un CNN simple pour le traitement d'images le tenseur d'image est réduit à un ensemble de noyaux de taille d par d ,

Chapitre III : Classification de texte pour la découverte des Services Web

ces couches convolutionnelles sont appelées cartes de caractéristiques et peuvent être empilées pour fournir plusieurs filtres pour l'entrée, pour réduire la complexité de calcul le CNN utilise Pooling cela réduit la taille de la sortie d'une couche du réseau à la suivante, diverses techniques de Pooling sont utilisées pour réduire la production tout en préservant les fonctionnalités essentielles.

La méthode de Pooling la plus courante est le Max Pooling qui sélectionne le plus grand élément dans la fenêtre de Pooling, la carte est aplatie en colonnes pour alimenter la sortie groupée des cartes d'entités empilées vers la couche suivante, la dernière couche d'un CNN est généralement une couche étroitement connectée et serrée, normalement l'étape de rétropropagation d'un réseau neuronal convolutif ajuste non seulement les poids, mais également le filtre du détecteur de caractéristiques [58].

La Figure 3-1 affiche la Classification de texte avec CNN.

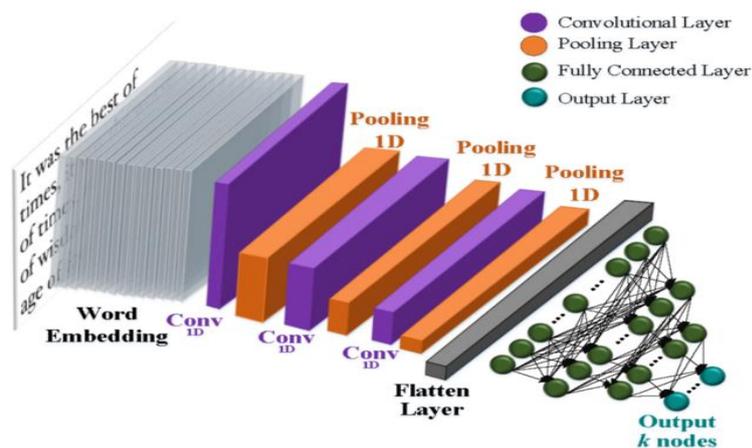


Figure 3-1 : Classification de texte avec CNN [58].

On cite un des travaux qui utilisent le CNN pour la classification de texte. Dans le travail (**Densely Connected CNN with Multi-scale Feature Attention for Text Classification**) de (Shiyao Wang, Minlie Huang, Zhidong Deng), ils proposent un CNN densément connecté avec une attention aux caractéristiques multi-échelles pour la classification de texte, les connexions denses créent des chemins de raccourci entre les blocs convolutifs en amont et en aval ce qui permet au modèle de composer des caractéristiques de plus grandes échelles à partir de celles de plus petites échelles et produire ainsi des caractéristiques variables de n-grammes, de plus une attention aux fonctionnalités multi-échelles est développée pour sélectionner de manière adaptative les multi-échelles caractéristiques pour la classification [59].

2-Recurrent Neural Networks (RNN) :

Les réseaux de neurones récurrents (RNN) sont une autre architecture de réseaux de neurones sur lesquelles travaillent les chercheurs en exploration de texte et en classification, le RNN attribue plus de poids aux points de données précédents dans la séquence par conséquent. Cette technique est un moyen puissant de classer du texte, des chaînes et des données séquentielles, dans les RNN les réseaux de neurones prennent en compte les informations des nœuds précédents d'une manière très sophistiquée qui permettent une meilleure analyse sémantique des structures de l'ensemble de données [58].

La figure 3- 2 affiche la Classification de texte avec RNN.

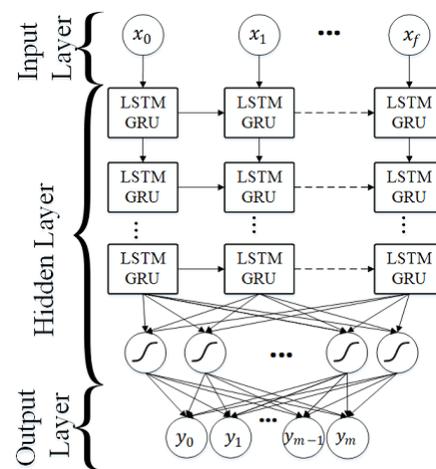


Figure 3- 2 : Classification de texte avec RNN [58].

- **Gated Recurrent Unit (GRU) :**

Une unité de régression fermée (GRU) est un mécanisme fermé pour les RNN, une variante simplifiée de l'architecture LSTM avec les différences suivantes : le GRU a deux portes et pas de mémoire interne (il n'y a pas de deuxième non-linéarité à la fin).

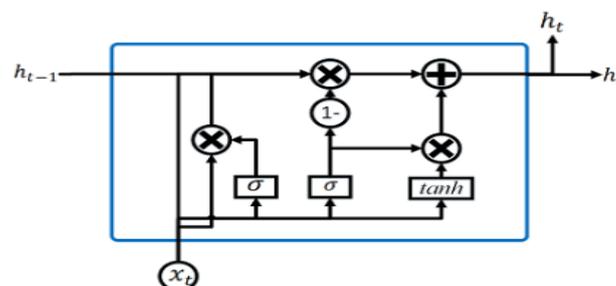


Figure 3-3 : Gated Recurrent Unit [58].

- **Long Short-Term Memory (LSTM) :**

Pour résoudre ces problèmes, la mémoire longue à court terme (LSTM) est un type spécial de RNN qui maintient les dépendances à long terme plus efficacement que les RNN simples, ceci est particulièrement utile pour surmonter les problèmes de gradient de fuite, car les LSTM utilisent plusieurs portes pour ajuster soigneusement la quantité d'informations autorisées dans l'état de chaque nœud. Cette figure montre les cellules de base du modèle LSTM [58].

Figure 3-4 affiche Long Short-Term Memory (LSTM)

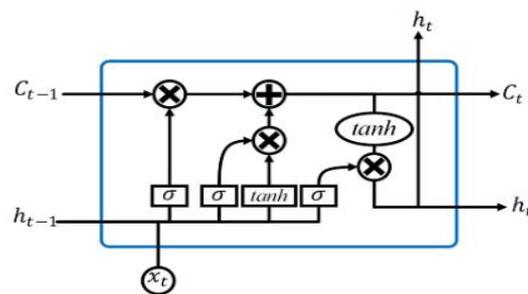


Figure 3-4: Long Short-Term Memory (LSTM) [58].

Parmi les travaux qui utilisent le RNN dans la classification de texte, on cite le travail de (Pengfei Liu, Xipeng Qiu, Xuanjing Huang) dans l'article (**Recurrent Neural Network for Text Classification with Multi-Task Learning**), qui propose trois architectures multitâches pour RNN, bien que l'idée d'apprentissage multitâches ne soit pas nouvelle. Ce travail est nouveau pour intégrer RNN dans le cadre d'apprentissage multiple qui apprend à mapper du texte arbitraire dans des représentations vectorielles sémantiques avec des couches spécifiques à la tâche et partagées [60].

3-Recurrent Convolutional Neural Networks (RCNN) :

Les réseaux de neurones convolutifs récurrents (RCNN) sont également utilisés pour la classification de texte, l'idée principale de cette technique est de capturer des informations contextuelles dans une structure répétitive et d'utiliser un réseau de neurones convolutifs pour construire une représentation textuelle. Cette architecture combine les RNN et les CNN pour tirer parti des deux techniques dans un seul modèle [58].

La Figure 3-5 illustre Les réseaux de neurones convolutifs récurrents (RCNN) pour la classification de texte.

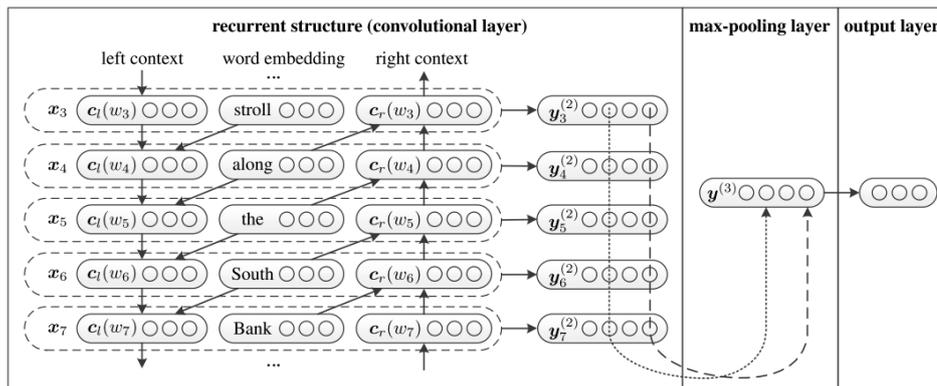


Figure 3-5 : Les réseaux de neurones convolutifs récurrents (RCNN) pour la classification de texte [61].

On cite un des travaux qui utilisent le RCNN dans la classification de texte, c'est l'article **(Recurrent Convolutional Neural Networks for Text Classification)** de (Siwei Lai, Liheng Xu, Kang Liu, Jun Zhao). Tout d'abord, ils appliquent une structure récurrente bidirectionnelle qui peut introduire considérablement moins de bruit par rapport à un réseau neuronal basé sur une fenêtre traditionnelle pour capturer les informations contextuelles dans la plus grande mesure possible lors de l'apprentissage des représentations de mots. De plus, le modèle peut réserver une plus grande plage d'ordre des mots lors de l'apprentissage des représentations de textes, Deuxièmement, ils utilisent une couche de MaxPooling qui juge automatiquement les fonctionnalités qui jouent un rôle clé dans la classification des textes, pour capturer le composant clé dans les textes ils combinent la structure récurrente et la couche de regroupement maximal. Ce modèle utilise à la fois les avantages des modèles neuronaux récurrents et des modèles neuronaux convolutifs, de plus, ce modèle présente une complexité temporelle de $O(n)$ qui est linéairement corrélée avec la longueur du texte [62].

4-Graph neural network (GNN) :

GNN (réseau de neurones graphiques) permet une classification textuelle au lieu de construire un seul graphe au niveau du corpus le GNN produit un graphe au niveau du texte pour chaque texte d'entrée pour un graphe au niveau du texte et relit les nœuds de mots dans une fenêtre raisonnablement petite dans le texte, au lieu de connecter directement tous les nœuds de mots les représentations des poids pour le même nœud et le même bord sont partagées globalement et peuvent être mises à jour dans des diagrammes au niveau du texte via un mécanisme de passage de message où les nœuds obtiennent des informations des nœuds voisins et mettent à jour la représentation.

Chapitre III : Classification de texte pour la découverte des Services Web

Finalement, on résume les représentations d'un nœud dans le graphe pour prévoir les résultats, les graphes éliminent le fardeau de la dépendance entre un seul texte d'entrée et le corpus entier qui prend en charge les tests en ligne, d'ailleurs en connectant des mots dans une petite fenêtre pop-up on a l'avantage d'utiliser moins de mémoire, en effet il exclut de nombreux mots distants du texte et n'a que peu à voir avec le mot courant, ce qui réduit considérablement le nombre d'arêtes, le mécanisme de passage de message permet aux nœuds du graphe de reconnaître les informations environnantes et d'obtenir la signification exacte dans un contexte particulier [63].

La figure suivante illustre la classification de texte à l'aide de réseaux de neurones basés sur le graphe.

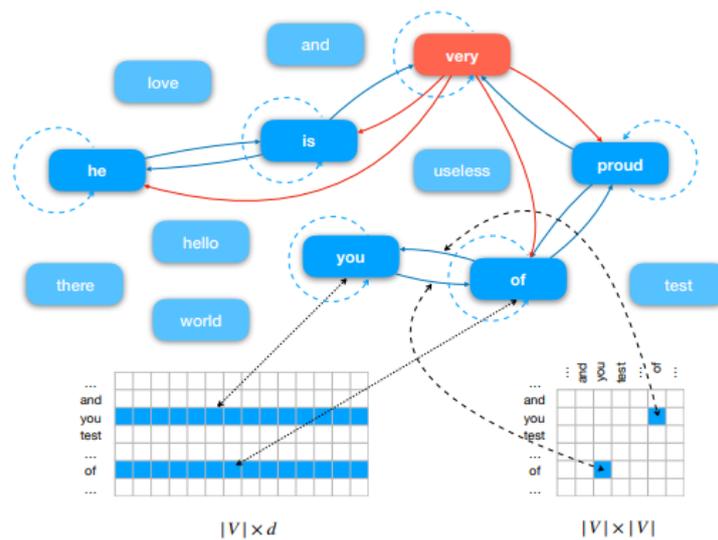


Figure 3-6 : La classification de texte par les GNN [64].

Il existe plusieurs travaux qui utilisent GNN pour la classification de texte on cite trois travaux :

A. Text Level Graph Neural Network for Text Classification:

Les auteurs de l'article [65], au lieu de construire un seul graphe au niveau du corpus, ils produisent un graphe de niveau de texte pour chaque texte saisi, pour un graphe au niveau du texte ils connectent des nœuds de mots dans une fenêtre raisonnablement petite dans le texte plutôt que de connecter directement tous les nœuds de mots, les représentations des mêmes nœuds et les poids des arêtes sont partagées à l'échelle mondiale et peuvent être mis à jour dans le texte de graphe de niveau via un mécanisme de passage de message, où un nœud prend les informations de nœuds voisins pour mettre à jour sa représentation, enfin, ils

Chapitre III : Classification de texte pour la découverte des Services Web

résumant les représentations de tous les nœuds du graphe pour prédire les résultats, avec cette conception les graphes au niveau du texte suppriment le fardeau de la dépendance entre un texte d'entrée unique et l'ensemble du corpus qui prend en charge le test en ligne, de plus, il a l'avantage de consommer moins de mémoire lors de la connexion des mots dans une petite fenêtre contextuelle, car il exclut bon nombre de mots éloignés dans le texte et ayant peu de rapport avec le courant [65].

B. A Sequential Graph Neural Network for Short Text Classification:

Les auteurs de l'article [66], ils proposent une nouvelle méthode basée sur GNN pour la classification de texte court appelée SGNN, Premièrement, ils entraînent chaque document dans le corpus de texte court sous forme de graphes individuels, qui utilisent des fenêtres coulissantes pour modéliser la structure contextuelle des mots et transformer la classification de texte en classification de graphe, pendant ce temps l'incorporation de mots préformés est utilisée comme caractéristique sémantique des mots, Deuxièmement, selon les informations séquentielles distinctives du document ils utilisent BiLSTM pour extraire la caractéristique contextuelle de chaque mot du document pour mettre à jour la représentation du nœud de mot pour chaque graphe de document, par rapport aux modèles précédents basés sur des graphes les informations séquentielles du document sont prises en compte dans la matrice de caractéristiques de chaque graphe de document, Troisièmement, un GCN simplifié a utilisé pour agréger les caractéristiques voisines de chaque nœud de mot pour apprendre les représentations de mots en fonction de leurs structures locales, enfin, les nœuds de mots suffisamment mises à jour sont incorporées en tant que représentations de document, De plus ils étendent un modèle appelé ESGNN qui conserve certaines caractéristiques contextuelles initiales dans le processus d'agrégation du GCN simplifié et atténue efficacement le problème du lissage excessif, au total cette méthode utilise les traits sémantiques de l'incorporation de mots pré-entraînés pour extraire tour à tour les caractéristiques séquentielles et les caractéristiques structurelles de chaque document, ce qui augmente l'échange de caractéristiques entre les mots du document et surmonte les limitations des caractéristiques textuelles dans les textes courts [66].

C. Inductive Text Classification via Graph Neural Networks :

Les auteurs de l'article [67], ils proposent une nouvelle méthode de classification de texte pour les représentations de mots inductifs via des réseaux de neurones de Graphes appelée TextING, contrairement aux approches précédentes basées sur des graphes avec une structure

Chapitre III : Classification de texte pour la découverte des Services Web

globale ils forment un GNN qui peut décrire les relations mot-mot détaillées, ils utilisent uniquement des documents de formation et inféré à nouveaux documents en test, ensuite ils construisent des graphes individuels et appliquer la fenêtre glissante à l'intérieur de chaque document, l'information des nœuds de mots est propagée à leurs voisins via les Gated Graph Neural Networks qui est ensuite agrégé dans le document embedding [67].

Cette méthode sera détaillée dans le chapitre 04.

5- Text Graph convolution Network (Text GCN):

Le réseau de graphe convolutif doit représenter un graphe de texte hétérogène qui peut modéliser une séquence globale de cooccurrence de mots, les graphes textuels hétérogènes contiennent les nœuds et les sommets du graphe, Texte GCN est un modèle qui permet d'utiliser des réseaux de neurones graphiques pour la classification de texte lorsque le type de réseau est la convolution. La figure suivante est un schéma d'un ajustement de graphe de convolution à l'aide d'un GCN de texte, ici la somme de la taille du corpus et du nombre de mots uniques est égale au nombre de nœuds dans le graphe de texte.

La figure suivante illustre Text Graph convolution Network.

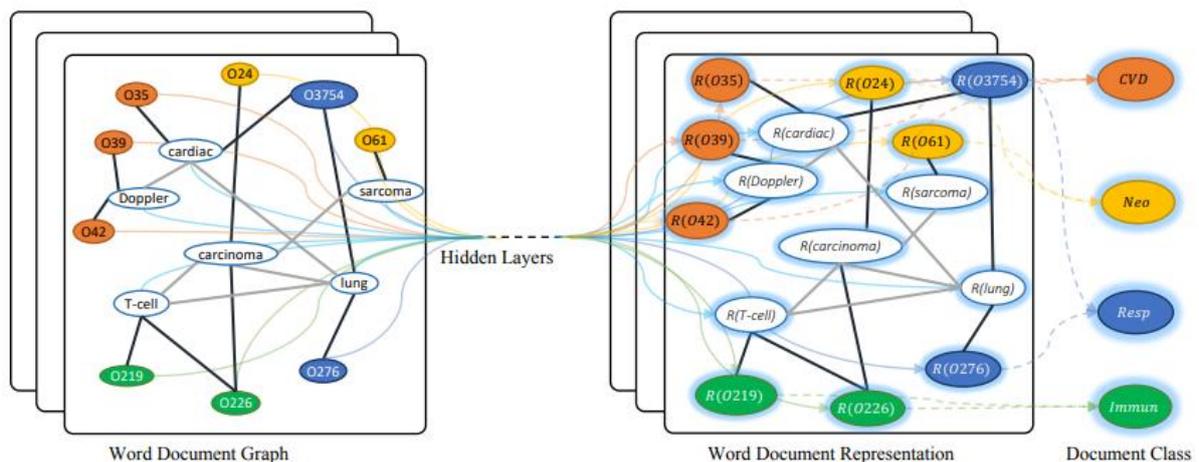


Figure 3-7 : Text Graph convolution Network [64].

Le modèle GCN de texte prend l'entrée comme une matrice d'identité de sorte que chaque mot peut être représenté comme un vecteur one-hot, pour générer le TF-IDF (fréquence de terme-fréquence de document inverse) pour les mots d'un document le modèle crée des arêtes entre les nœuds en fonction de l'apparence des mots dans le corpus global, comme il

Chapitre III : Classification de texte pour la découverte des Services Web

est traditionnel dans TFIDF le terme fréquence fait référence au nombre de fois qu'un mot apparaît dans un document. Pour collecter des statistiques de cooccurrence le modèle fournit une fenêtre de taille fixe sur le document dans le corpus et lorsqu'on fait un glissement de la fenêtre, l'ensemble des informations de cooccurrence de mots aide à prédire et à classer [64]. Il existe plusieurs méthodes de classification de texte basée sur les GCN, d'on peut citer deux travaux :

A. Graph Convolutional Networks for Text Classification:

Les auteurs de l'article [68], ils proposent d'utiliser des réseaux convolutifs de graphes pour la classification de textes, ils construisent un graphe de texte unique pour un corpus basé sur la cooccurrence de mots et les relations de mots de document, puis apprennent un Text Graph Convolutional Network (Text GCN) pour le corpus, le Text GCN est initialisé avec la représentation one-hot pour le mot et le document, il apprend ensuite conjointement les Embedding pour les mots et les documents comme supervisé par les étiquettes de classe connues pour les documents [68].

B. Text Classification Study Based on Graph Convolutional Neural Networks (2021) :

Les auteurs de l'article [69], ils utilisent la fonction d'activation Mish () pour traiter le problème des données expérimentales dans la limite du zéro dur et en modifiant l'élan du moyen mobile à un pas dans l'optimiseur, un certain degré d'amélioration est proposé au problème de mémoire à court terme après l'optimisation des paramètres expérimentaux, qui s'appelle LTM TEXT-GCN [69]. la figure suivante explique la classification de texte avec LTM TEXT-GCN.

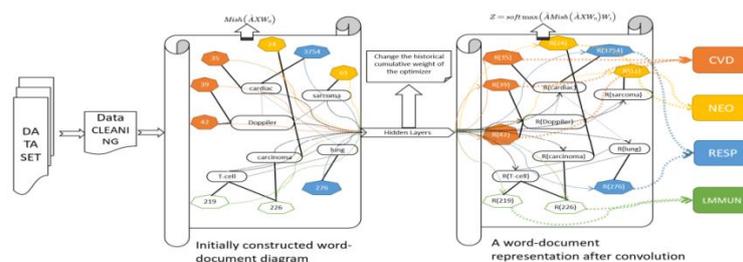


Figure 3-8 : La classification de texte à l'aide de LTM TEXT-GCN [69].

3- Quelques travaux connexes pour la classification des services web :

Avec le développement du Web 2.0 et des services Web, les types et quantités de services Web augmentent rapidement. Trouver un service web qui répond à nos besoins devient de plus en plus difficile, il s'agit d'un moyen efficace de faciliter la découverte de services pour classer les services Web avec des fonctionnalités similaires, les technologies existantes de classification des services Web sont principalement axées sur l'exploitation d'informations fonctionnelles telles que Texte descriptif ou balises pour réaliser la classification des services Web et permet l'accélération de détection des services, il peut également être couramment utilisé pour trouver les services dont les développeurs ont besoin [70].

3.1 A Web Services Classification Method based on GCN (2019):

On trouve dans le travail des auteurs de l'article [70], une approche de classification des services Web par réseau neuronal convolutif de graphes. Cette méthode utilise d'abord le nom, le texte de description, les balises du service Web comme corpus de base pour construire un réseau de graphes hétérogène de "Mots & description de service Web documents" en fonction de la cooccurrence des mots et de la relation des mots entre le document de description du service Web. Dans un réseau de graphes hétérogènes tous les poids de bord entre les nœuds de document et de mot sont calculés à l'aide du terme fréquence de document inverse et les informations de points mutuels sont utilisées pour calculer les poids de bord entre les différents nœuds de document, ensuite, elle applique un réseau de neurones avec des graphes de convolution pour apprendre les informations intégrées des mots et des documents de description de service Web et transformer le problème de classification de document de service Web en un problème de classification de nœud [70].

3.2 A Novel Mixed Wide and PSO-Bi-LSTM-CNN Model for the Effective Web Services Classification (2020):

On trouve dans le travail de (K. Punitha, 2020) un modèle mixte large et PSO-Bi-LSTM-CNN (MW-PSO Bi-LSTM-CNN) est proposé. Cette technique combine les fonctionnalités individuelles d'un document de description de service Web qui exploite un modèle d'apprentissage étendu pour prédire la largeur d'une catégorie de service Web, l'étape suivante consiste à utiliser le modèle PSO-Bi-LSTM-CNN pour extraire les informations contextuelles

Chapitre III : Classification de texte pour la découverte des Services Web

et l'ordre des mots du document de description du service Web et effectuer des prédictions détaillées pour la catégorie de service Web, l'optimisation des essaims de particules (PSO) est intégrée au réseau Bi-LSTM-CNN pour calculer automatiquement divers hyper paramètres, dans la troisième étape les résultats de prédiction de la catégorie, de la profondeur et de la largeur du service web sont intégrés à l'aide d'un modèle de régression linéaire comme résultat final de la classification du service [71].

3.3 Deep Learning for Web Services Classification (2019) :

Les auteurs de l'article [72], proposent un nouveau système de réseau neuronal profond capable d'abstraire automatiquement les représentations de bas niveau des noms de service et des descriptions de service et de les fusionner en fonctionnalités de haut niveau sans ingénierie de fonctionnalités ni restrictions de longueur, puis de prédire la classification des services sur 50 classes de services, ce qui est ServeNet L'architecture est illustrée dans la figure, qui contient la couche d'intégration, la couche d'extraction de fonctionnalités et la couche de tâches [72]. La figure suivante affiche Architecture ServeNet.

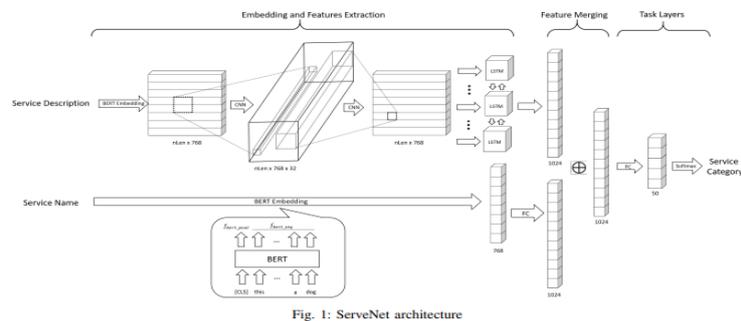


Figure 3. 9 : Architecture ServeNet [72].

3.4 A Spatial and Sequential Combined Method for Web Service Classification (2020):

Les auteurs de l'article [73], Proposent un nouveau réseau de neurone profond qui intègre un réseau convolutif de graphes (GCN) et un réseau bidirectionnel de mémoire à long court terme (BiLSTM) pour extraire automatiquement des caractéristiques d'un document de description fonctionnelle d'un service web, en particulier ils utilisent d'abord un GCN à deux niveaux pour extraire les caractéristiques de l'ensemble de la structure spatiale du service Web cela agit comme un processus d'intégration de mots de préformation, les capacités séquentielles du service Web apprises du modèle BiLSTM sont ensuite intégrées pour l'apprentissage des paramètres partagés [73].

3.5 Graph4Web: A relation-aware graph attention network for web service classification (2022):

Les auteurs de l'article [74], proposent Graph for Web qui utilise un réseau relationnel d'attention aux graphes pour classer les services Web, en particulier ils initialisent le vecteur d'intégration pour chaque nœud du graphe en analysant d'abord la séquence de description de service Web du graphe de dépendances et en ajustant un modèle BERT pré-entraîné, ils proposent également une couche d'attention du graphe de reconnaissance des relations pour apprendre et mettre à jour les vecteurs d'intégration des nœuds en agrégeant les informations des nœuds adjacents et divers types de relations entre les nœuds, de plus, ils introduisent un mécanisme d'auto-conscience pour maintenir un niveau élevé de représentation globale de la classification des services Web [74].

3.6 DeepWSC: A Novel Framework with Deep Neural Network for Web Service Clustering (2019):

Les auteurs de l'article [75], proposent un nouveau cadre avec un réseau neuronal profond pour le clustering de services Web appelé DeepWSC, où un réseau neuronal profond formé sur la base d'un modèle de sujet probabiliste de manière non supervisée est exécuté pour acquérir des informations contextuelles implicites de services Web. Ce cadre tire parti du réseau neuronal convolutif récurrent (RCNN) et atteint une meilleure précision de clustering de services Web, ensuite il stimule fondamentalement la découverte, la composition et la recommandation de services en fournissant efficacement les services web souhaités, une stratégie d'expansion a été proposée pour extraire plus précisément les caractéristiques des services web qui intègre un réseau neuronal convolutif récurrent amélioré (RCNN), un modèle de sujet probabiliste augmenté (WE-LDA) et un modèle d'incorporation de mots formés (Word2vec) [75].

4 – Conclusion :

Dans ce chapitre nous avons présenté la fonction de découverte des services Web qui est responsable de la recherche, il est basé sur la description de ces interfaces pour améliorer la vitesse du processus de découverte et augmente sa précision appropriée à des besoins spécifiés une méthode a été utilisée c'est la classification des services web, qui est basée sur des techniques d'apprentissage en profondeur, nous avons mentionné quelques travaux notables dans ce domaine au moyen des algorithmes de deep learning et de ces travaux, nous discuterons de TextING dans le contenu du chapitre suivant.

Chapitre IV : Amélioration de processus de découverte des services Web

1-Introduction :	53
2-Graph neural network (GNN):	53
3- TextING :	57
4- le modèle TextING pour les services web :	60
5-Configuration expérimentale et l'ensemble de données :	65
6- Discussion des résultats :	66
7-Conclusion :	69

1-Introduction :

La découverte de services vise à rechercher et trouver des services qui répondent aux besoins des demandeurs de services.

Les informations de description de service sont principalement composées de texte court et bruyant et se caractérisent par leur manque de sémantique, ce qui rend difficile l'extraction d'informations contextuelles implicites à partir de la description de service [76]. Dans cette tâche on 'a utilisé TextING pour améliorer le processus de découverte de services Web basé sur les réseaux de neurones de graphe (GNN).

La méthode propose de construire un graphe distinct pour chaque document et utilise GNN pour apprendre des expressions de mots à grain fin basées sur la structure locale, cela permet également de générer efficacement des incorporations de mots cachés pour les nouveaux documents et ceux-ci enrichissent les informations sémantiques du service descriptif.

2-Graph neural network (GNN):

Les réseaux de neurones de graphes (GNN) sont un ensemble de techniques d'apprentissage en profondeur qui fonctionnent dans le domaine des graphes. Ces réseaux ont récemment été appliqués dans plusieurs domaines notamment : optimisation combinatoire, systèmes de recommandation et vision par ordinateur. Ces réseaux peuvent également être utilisés pour modéliser de grands systèmes tels que les réseaux sociaux, les réseaux d'interaction protéine-protéine, les graphes de connaissances et d'autres disciplines de recherche, contrairement à d'autres données telles que les images et les données de graphe fonctionnent dans un espace non euclidien, par conséquent l'analyse de graphes vise la classification des nœuds, la prédiction des liens et le regroupement [77].

2-1 La notion de graphe :

Un graphe est une structure de données composée de deux composants, des sommets et des arêtes, il est utilisé comme structure mathématique pour analyser les relations par paires entre les objets et les entités, les graphes sont généralement définis comme $G = (V, E)$ où V est l'ensemble des sommets et E est l'arête entre eux.

la Figure 4-1 explique La visualisation d'un simple Graphe.

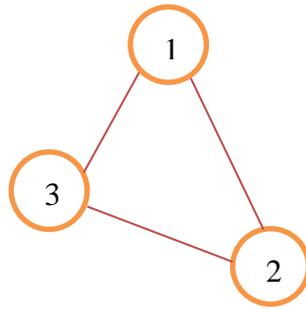


Figure 4-1 : La visualisation d'un simple Graphe.

Les graphes sont souvent représentés par la matrice d'adjacence A , si le graphe a N nœuds, alors A a une dimension $(N \times N)$. Une matrice fonctionnelle différente peut être fournie pour décrire les nœuds dans le graphe, si chaque nœud a des caractéristiques de numéro F , la dimension de la matrice de caractéristiques X est $(N \times F)$ [78].

2-2 les réseaux de neurones de graphe :

Les réseaux de neurones de graphe peuvent être construits comme n'importe quel autre réseau de neurones, en utilisant des couches entièrement connectées, des couches de convolution et des couches de regroupement, le type et le nombre de niveaux dépendent du type et de la complexité des données du graphe et de la sortie souhaitée. Le GNN prend en entrée des données des graphes formatées et produit un vecteur de nombres représentant des informations pertinentes sur les nœuds et leurs relations.

Cette représentation vectorielle s'appelle "Embedding de graphes", l'intégration est couramment utilisée dans l'apprentissage automatique et transforme des informations complexes dans une structure qui peut être apprise séparément, par exemple les systèmes de traitement du langage naturel utilisent embedding de mots pour créer des représentations numériques des mots et de leurs relations.

2-2-1 Embedding de Graphe :

Lorsque les données du graphe sont transmises à GNN, les propriétés de chaque nœud sont combinées avec les propriétés du nœud adjacent, c'est ce qu'on appelle le "passage de messages", si le GNN se compose de plusieurs couches, la couche suivante répète l'opération d'envoi de message en collectant les données des voisins du voisin et en les agrégeant avec les valeurs obtenues de la couche précédente. L'Embedding de Graphe est illustré dans la figure suivante.

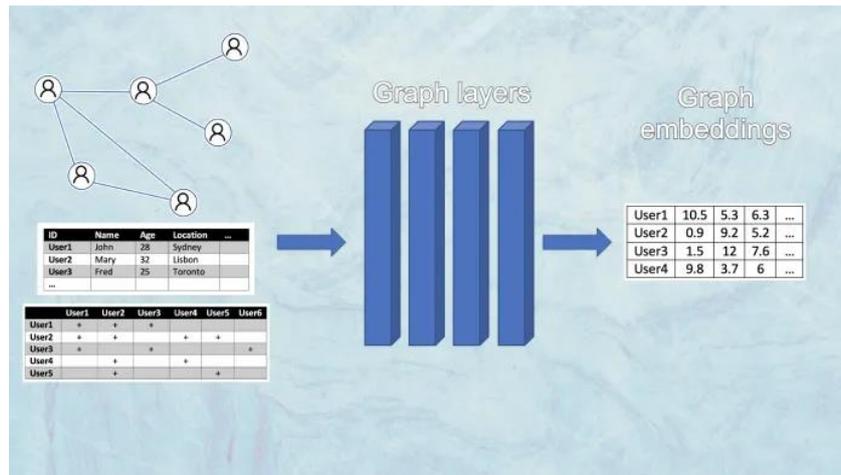


Figure 4-2 : Embedding de graphe [79].

2-2-2 Les tâches de GNN :

- La classification au niveau de nœuds :** GNN permet de déduire le comportement de nœuds individuel dans un diagramme en fonction de leurs relations avec d'autres nœuds, une tâche courante est la classification des nœuds, l'objectif ici est de déduire l'étiquette manquante du nœud en examinant les étiquettes et les entités adjacentes cette méthode est utilisée dans des applications telles que la détection des fraudes financières, la classification des publications et la classification des maladies.

La figure 4-3 explique La classification de nœud.

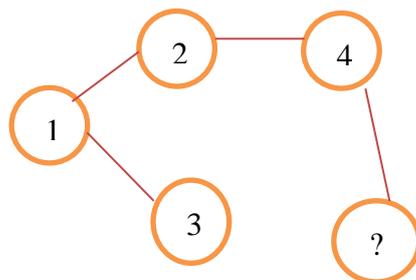


Figure 4-3 : La classification de nœud.

- La classification au niveau des liens :** une autre tâche d'apprentissage importante pour les graphes c'est la prédiction de liens, ceci est au cœur des applications telles que les recommandations de produits et de publicités, les suggestions d'amis...etc. étant donné une relation à deux nœuds le but est de déterminer si les nœuds sont reliés par une relation [80].

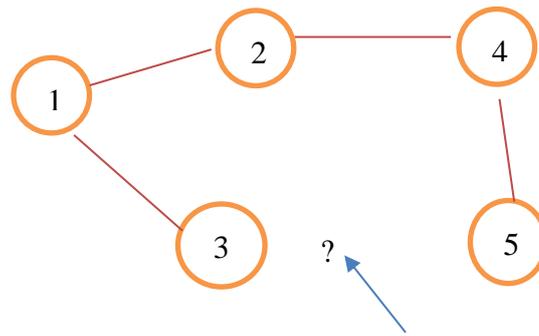


Figure 4-4 : La classification ou niveau des liens.

- **Le regroupement** : GNN peut extraire des nouvelles informations structurelles du graphe [79]. La classification au niveau du graphe consiste à analyser un ensemble de Sous-graphes indépendants [80].

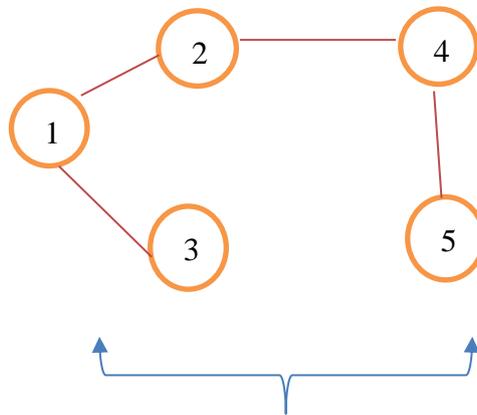


Figure 4-5 : La classification de graphe.

2-3 Les méthodes basées sur le GNN pour la classification de texte :

1-GNN-XML: Graph Neural Networks for Extreme Multi-label Text Classification (2020) :

Les auteurs de l'article [81], proposent GNN-XML, un cadre de réseau de neurones à graphes évolutif adapté aux problèmes XMTC (Extreme multi-label text classification), plus précisément ils exploitent les corrélations d'étiquettes en explorant leurs modèles de cooccurrence et construisent un graphe d'étiquettes basé sur la matrice de corrélation, ils effectuent ensuite le regroupement de graphes attribués en effectuant une convolution de graphe avec un filtre de graphe passe-bas pour modéliser conjointement les dépendances d'étiquettes et les caractéristiques des étiquettes, ce qui induit des grappes d'étiquettes sémantiques, ils proposent en outre un réseau d'isomorphisme de graphes à branches bilatérales pour découpler l'apprentissage de la représentation et l'apprentissage du classificateur pour une meilleure modélisation des étiquettes de queue [81].

2-Weakly-supervised Text Classification Based on Keyword Graph (2021):

Les auteurs de l'article [82], proposent un nouveau cadre appelé ClassKG pour explorer la corrélation mot-clé sur le graphe de mots-clés par GNN. Ce cadre est un processus itératif, dans chaque itération ils construisent d'abord un graphe de mots-clés de sorte que la tâche d'attribution de pseudo-étiquette est transformée en annotation de sous-graphes de mots-clés, pour améliorer la qualité des annotations ils introduisent une tâche auto-supervisée pour pré-entraîner un annotateur de sous-graphe, puis l'affiner avec les pseudo-étiquettes générées par l'annotateur de sous-graphe, ils entraînent ensuite un classificateur de texte pour classer les textes non étiquetés. Enfin, ils ré-extraient les mots-clés des textes classés [82].

3- Tensor Graph Convolutional Networks for Text Classification (2020):

Les auteurs de l'article [83], étudient les réseaux de neurones basés sur des graphes pour un problème de classification de texte. Un nouveau framework TensorGCN (réseaux convolutifs de graphes tensoriels) est présenté pour cette tâche. Un tenseur de graphe de texte est d'abord construit pour décrire des informations contextuelles sémantiques, syntaxiques et séquentielles. Ensuite deux types d'apprentissage de propagation s'effectuent sur le tenseur de graphe de texte, le premier est la propagation intra-graphe utilisé pour agréger les informations des nœuds de voisinage dans un seul graphe, le second est la propagation inter-graphes utilisée pour harmoniser des informations hétérogènes entre graphes [83].

3- TextING :

TextING une méthode de classification de texte pour les représentations inductives de mots via des réseaux de neurones basés sur les graphes, consiste à former un GNN capable de décrire des relations mot-mot détaillé en utilisant uniquement des documents d'apprentissage et d'inférer de nouveaux documents lors des tests, la méthode consiste à construire des graphes séparés en appliquant une fenêtre glissante dans chaque document et les informations des nœuds de mots sont propagées à leurs voisins via un réseau neuronal de graphes fermés, puis agrégées dans des embedding de documents [67].

Les contributions de cette méthode se résument en trois points

- Contrairement aux TextGCN qui construit un graphe unique avec des relations globales entre les documents et les mots, où les interactions de mots au niveau du texte à grain fin ne sont pas prises en compte. Cette méthode propose un nouveau réseau de

neurones à graphes pour classification des textes, où chaque document est un graphe individuel et les représentations de mots à grain fin au niveau du texte sont apprises dans ce graphe par l'utilisation de la fenêtre coulissante.

- Contrairement au TextGCN qui se base sur l'apprentissage transductif en raison de sa structure globale ou les documents d'apprentissages et de teste sont pré-analysés. Le TextING est basée sur l'apprentissage inductif, qui fait l'inférence d'une règle générale à partir de documents d'apprentissage ou cette règle doit classer les nouveaux documents avec une grande précision, c'est-à-dire le modèle doit s'exécuter sur un ensemble de données entièrement nouveau, que le modèle n'a jamais rencontré auparavant, donc dans le cas d'un nouveau point de données il prédit le point instantanément, alors que dans l'apprentissage transductif, il ré-exécute et ré-entraîne le modèle entier, ceci est très coûteux et prend du temps.
- Les résultats démontrent que cette approche est plus performante que les méthodes de classification de texte classique de manière expérimentale (sera expliquée dans la section suivante).

TextING se compose de trois éléments principaux :

La construction de graphes, l'interaction des mots basée sur des graphes et la fonction de lecture. Sont illustrent dans la figure suivante.

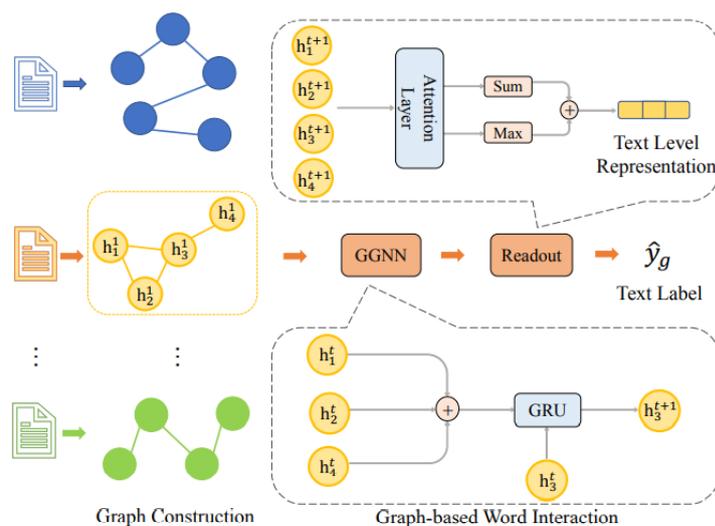


Figure 4-6 : La méthode TextING [67].

- **La construction des graphes :** ils construisent un graphe de documents texte en représentant des mots uniques comme des sommets et des cooccurrences entre mots comme des arêtes, notées $G = (V, E)$ où V est l'ensemble des sommets et E est l'arête la cooccurrence décrit la relation entre les mots qui apparaissent dans un espace de taille fixe, les fenêtres coulissantes (longueur par défaut 3) et elles n'ont pas de direction dans le graphique.

Le texte est prétraité par défaut, comme la tokenisation et la suppression des mots vides, l'incorporation des sommets est initialisée avec la caractéristique des mots représentée par $h \in \mathbb{R}^{|V| \times d}$. Où d est la dimension intégrée, la création d'un graphe distinct pour chaque document propage les informations sur les propriétés des mots et les intègre dans le contexte pendant la phase d'interaction des mots.

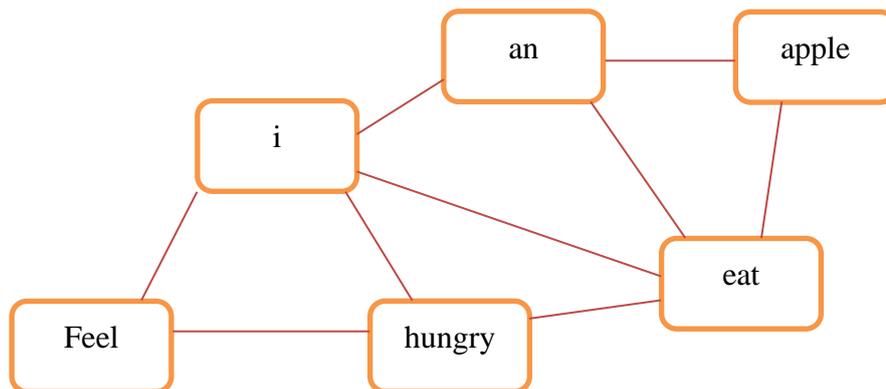


Figure 4-7 : La construction de graphe de TextING.

- **Interaction de mots basée sur un graphique :** sur chaque graphe, ils utilisent un réseau de neurones à graphes fermés (Gated Graph neural network) pour apprendre les embedding de nœuds de mots, un nœud peut recevoir des informations de ses voisins qu'il fusionne ensuite avec sa propre représentation pour la mettre à jour, puisqu'une couche fonctionne sur des voisins de premier ordre ils peuvent empiler la couche t fois pour obtenir des interactions d'entité d'ordre supérieur où un nœud peut atteindre un autre nœud t sauts de distance. Les formules de l'interaction sont :

$$a^t = Ah^{t-1}W_a \quad (3)$$

$$z^t = \sigma(W_z a^t + U_z h^{t-1} + b_z) \quad (4)$$

$$r^t = \sigma(U_r h^{t-1} + b_r) \quad (5)$$

$$\tilde{h}^t = \tanh(W_h a^t + U_h (r^t \odot h^{t-1}) + b_h) \quad (6)$$

$$h^t = \tilde{h}^t \odot z^t + h^{t-1} \odot (1 - z^t) \quad (7)$$

où $A \in \mathbb{R}^{|V| \times |V|}$ est la matrice d'adjacence, σ est la fonction sigmoïde et tous W , U et b sont des pondérations et biais entraînaibles, z et r fonctionnent comme la porte de mise à jour et la porte de réinitialisation respectivement pour déterminer dans quelle mesure les informations de voisinage contribuent à l'intégration actuelle des nœuds.

- **La fonction de lecture :** lorsque les nœuds de mots sont suffisamment mises à jour, ils sont agrégés dans la représentation graphique du document qui est à la base de la prédiction finale. La formule de la fonction de lecture est comme suit :

$$h_v = \sigma(f_1(h_v^t)) \odot \tanh(f_2(h_v^t)) \quad (8)$$

$$h_g = \frac{1}{|v|} \sum_{v \in V} h_v + \text{Maxpooling}(h_1 \dots h_v) \quad (9)$$

Où f_1 et f_2 sont deux perceptrons multicouches (MLP). Le premier agit comme un poids d'attention douce et le second agit comme une transformation de caractéristique non linéaire, en plus de faire la moyenne des caractéristiques des mots ils appliquent également la fonction MaxPooling pour la représentation de graphe h_g , l'idée derrière cela est que chaque mot du texte est important et que les mots-clés doivent contribuer plus clairement, enfin l'étiquette est prédite en alimentant le vecteur de niveau de graphe à la couche softmax, on minimise les pertes grâce à la fonction d'entropie croisée.

$$\tilde{y}_g = \text{softmax}(w h_g + b) \quad (10)$$

$$l = - \sum_i y g_i \log(\tilde{y}_g) \quad (11)$$

Où W sont les poids et b le biais et $y g_i$ est le i -ème élément de l'étiquette one hot [67].

4- Le modèle TextING pour les services web :

Des approches corrélatives ont tenté de regrouper les services web sur la base soit des informations explicites contenues dans les descriptions de service, soit des caractéristiques

sémantiques de fonctionnalité extraites par des modèles thématiques probabilistes, cependant les informations contextuelles implicites des descriptions de service sont ignorées et doivent encore être correctement explorées et exploitées[75], nous proposons un nouveau cadre avec le GNN appelé TextING, nous construisons d'abord des graphes individuels pour chaque service puis nous utilisons le GNN pour apprendre les représentations de mots à grains fin en fonction de leurs structures locales, ce qui peut également produire efficacement des embedding pour des mots invisibles dans le nouveau document, enfin les nœuds de mots sont incorporés en tant que l'incorporation de documents [67].

4-1 Processus de notre approche :

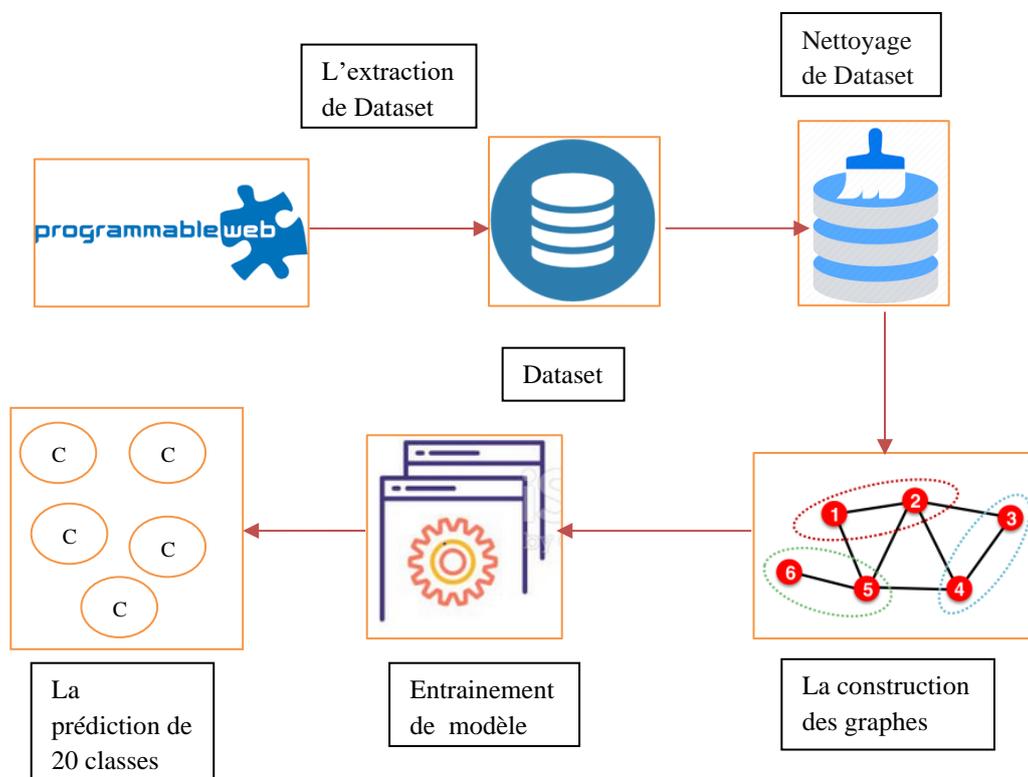


Figure 4-8 : Processus de notre approche

4-1-1 La préparation des données :

Les données de Dataset sont prétraitées de manière standard, y compris la tokenisation et la suppression des mots vides, en ce qui concerne le Word Embedding nous avons utilisé le GloVe et les mots hors vocabulaire (OOV) ont été échantillonnés au hasard à partir d'une distribution uniforme $[-0.01, 0.01]$. la figure suivante affiche les mot (OOV).

```

oov = {}
for v in vocab:
    oov[v] = np.random.uniform(-0.01, 0.01, word_embeddings_dim)
Last executed at 2022-06-04 15:11:56 in 189ms

oov
Last executed at 2022-06-04 15:12:07 in 815ms

{'xfers': array([ 0.00578469, -0.0036941, -0.00987974, -0.00899981,  0.00392832,
 -0.00409926, -0.00892365, -0.00764164, -0.00821813, -0.00019705,
 -0.0022878,  0.00641731, -0.00487705, -0.00247309, -0.00128725,
 -0.00846621, -0.00669672, -0.00278347, -0.00603335,  0.00243962,
  0.00717016,  0.00573644, -0.0066197, -0.00592091,  0.00793881,
 -0.00190953, -0.00588216, -0.00436971, -0.00857519, -0.00947951,
  0.00860187,  0.0060766,  0.00875109,  0.00346387,  0.0080011,
  0.00349947,  0.00256604, -0.00912277,  0.00693834, -0.00964631,
  0.0052322, -0.00839899, -0.00132576, -0.00119939, -0.0064136,
 -0.00886,  0.00331183, -0.00677607,  0.00500363,  0.00159784,
  0.00929911,  0.00208009,  0.00367479,  0.00490607, -0.0013294,
  0.00747267, -0.00259549,  0.00472451,  0.00730309, -0.00235375,
 -0.00242503,  0.00705405, -0.004256, -0.00946762, -0.00173338,
  0.00753089, -0.00096655, -0.00807556,  0.00383298, -0.00302236,
  0.00069225,  0.00194741,  0.00032115, -0.00832327, -0.00506985,
  0.00902219,  0.00509417,  0.0016693, -0.00837703, -0.00730426,
  0.00356375,  0.00117982, -0.00258046, -0.00374343, -0.00518287,
 -0.00330063, -0.00722333, -0.00402457,  0.00045275, -0.00985835,
  0.00478997,  0.00227502, -0.00118555,  0.0095679,  0.00628444,
  0.00070254,  0.00641911,  0.00860191, -0.00091017,  0.00976902]),

```

Figure 4-9 : La distribution des mots hors vocabulaire

4-1-2 La construction des graphes pour les services web :

On conçoit un récipient qui contient tous les mots possibles de description des services pour chaque service web en représentant des mots uniques comme des sommets et des cooccurrences entre les mots comme des arêtes, les cooccurrences décrivent la relation entre les mots apparaissant dans un espace de taille fixe, la fenêtre coulissante (longueur trois par défaut) et ils sont bidirectionnels dans le graphe. La figure suivante explique La construction des graphes pour les services web

```

def build_graph(start, end):
    x_adj=[]
    x_feature=[]
    doc_len_list = []
    y=[]
    vocab_set = set()
    for i in tqdm(range(start, end)):
        doc_words = shuffle_doc_words_list[i].split()
        if truncate:
            doc_words = doc_words[:MAX_TRUNC_LEN]
        doc_len = len(doc_words)
        doc_vocab = list(set(doc_words))
        doc_nodes = len(doc_vocab)
        doc_len_list.append(doc_nodes)
        vocab_set.update(doc_vocab)
        doc_word_id_map = {}
        for j in range(doc_nodes):
            doc_word_id_map[doc_vocab[j]] = j

    # sliding windows
    windows = []
    if doc_len <= window_size:
        windows.append(doc_words)
    else:
        for j in range(doc_len - window_size + 1):
            window = doc_words[j: j + window_size]
            windows.append(window)

```

Figure 4-10 : La construction des graphes pour les services web

```

word_pair_count = {}
for window in windows:
    for p in range(1, len(window)):
        for q in range(0, p):
            word_p = window[p]
            word_p_id = word_id_map[word_p]
            word_q = window[q]
            word_q_id = word_id_map[word_q]
            if word_p_id == word_q_id:
                continue
            word_pair_key = (word_p_id, word_q_id)
            # word co-occurrences as weights
            if word_pair_key in word_pair_count:
                word_pair_count[word_pair_key] += 1.
            else:
                word_pair_count[word_pair_key] = 1.
            # bi-direction
            word_pair_key = (word_q_id, word_p_id)
            if word_pair_key in word_pair_count:
                word_pair_count[word_pair_key] += 1.
            else:
                word_pair_count[word_pair_key] = 1.

```

Figure 4-11 : La figure explique les coocurrence entre les sommets

On découpe notre dataset et on construit des graphes pour (training, validation + training et teste).

- Training : pour optimiser les paramètres de GNN.
- Validation : pour améliorer le modèle.
- Test : pour la performance finale.

```

print('building graphs for training')
x_adj, x_feature, y, _, _ = build_graph(start=0, end=real_train_size)
print('building graphs for training + validation')
allx_adj, allx_feature, ally, doc_len_list_train, vocab_train = build_graph(start=0, end=train_size)
print('building graphs for test')
tx_adj, tx_feature, ty, doc_len_list_test, vocab_test = build_graph(start=train_size, end=train_size+test_size)
doc_len_list = doc_len_list_train + doc_len_list_test

```

```

building graphs for training
100% ██████████ 6471/6471 [00:03<00:00, 1632.79it/s]
building graphs for training + validation
100% ██████████ 7190/7190 [00:04<00:00, 1666.00it/s]
building graphs for test
100% ██████████ 1269/1269 [00:00<00:00, 1669.55it/s]

```

Figure 4-12 : La construction des graphes pour (training, validation + training et teste)

Afin d'enregistrer ces graphes dans des fichiers. et ils sont illustrées dans la figure suivante.

```

with open("data/ind.c.x_adj".format(datasets), 'wb') as f:
    pickle.dump(x_adj, f)

with open("data/ind.c.x_embed".format(datasets), 'wb') as f:
    pickle.dump(x_feature, f)

with open("data/ind.c.y".format(datasets), 'wb') as f:
    pickle.dump(y, f)

with open("data/ind.c.tx_adj".format(datasets), 'wb') as f:
    pickle.dump(tx_adj, f)

with open("data/ind.c.tx_embed".format(datasets), 'wb') as f:
    pickle.dump(tx_feature, f)

with open("data/ind.c.ty".format(datasets), 'wb') as f:
    pickle.dump(ty, f)

with open("data/ind.c.allx_adj".format(datasets), 'wb') as f:
    pickle.dump(allx_adj, f)

with open("data/ind.c.allx_embed".format(datasets), 'wb') as f:
    pickle.dump(allx_feature, f)

with open("data/ind.c.ally".format(datasets), 'wb') as f:
    pickle.dump(ally, f)

```

Figure 4-13 : L'enregistrement des graphes.

4-1-3 l'interaction des mots :

Pour chaque graphe, on utilise Gated Graphe neural network qui utilise GRU (Gated Recurrent Unit) pour apprendre les Embedding des mots, un nœud recueille les informations des voisins directs (message passant) puis combine avec sa propre représentation pour la mettre à jour. Ce message passant est fait par tous les nœuds et dont nous avons de nouveaux Embedding pour chaque sommet de graphe, comme la couche graphique opère sur les voisins de premier ordre nous pouvons empiler cette couche t fois pour obtenir interactions d'entités d'ordre élevé.

```

def gru_unit(support, x, var, act, mask, dropout, sparse_inputs=False):
    """GRU unit with 3D tensor inputs."""
    # message passing
    support = tf.nn.dropout(support, dropout) # optional
    a = tf.matmul(support, x)

    # update gate
    z0 = dot(a, var['weights_z0'], sparse_inputs) + var['bias_z0']
    z1 = dot(x, var['weights_z1'], sparse_inputs) + var['bias_z1']
    z = tf.sigmoid(z0 + z1)

    # reset gate
    r0 = dot(a, var['weights_r0'], sparse_inputs) + var['bias_r0']
    r1 = dot(x, var['weights_r1'], sparse_inputs) + var['bias_r1']
    r = tf.sigmoid(r0 + r1)

    # update embeddings
    h0 = dot(a, var['weights_h0'], sparse_inputs) + var['bias_h0']
    h1 = dot(r*x, var['weights_h1'], sparse_inputs) + var['bias_h1']
    h = act(mask * (h0 + h1))

    return h*z + x*(1-z)

```

Figure 4-14 : L'interaction de mots entre les services web

4-1-5 La fonction de lecture :

Une fois que les nœuds de mots sont suffisamment mises à jour, ils sont agrégés à une représentation graphe pour le document, lorsque l'étiquette est prédite en alimentant le vecteur de niveau de graphe dans une couche softmax, nous minimisons la perte grâce à la fonction d'entropie croisée, la figure suivante affiche La fonction softmax_cross_entropy

```
def softmax_cross_entropy(preds, labels):  
    """Softmax cross-entropy loss with masking."""  
    loss = tf.nn.softmax_cross_entropy_with_logits(logits=preds, labels=labels)  
    return tf.reduce_mean(loss)  
  
def accuracy(preds, labels):  
    """Accuracy with masking."""  
    correct_prediction = tf.equal(tf.argmax(preds, 1), tf.argmax(labels, 1))  
    accuracy_all = tf.cast(correct_prediction, tf.float32)  
    return tf.reduce_mean(accuracy_all)
```

Figure 4-15 : La fonction softmax_cross_entropy

5-Configuration expérimentale et l'ensemble de données :

Nous avons réalisé le travail sur plateforme Saturn Cloud avec (2XLarge – 8cores -64 GB RAM – 40Gi Disk).

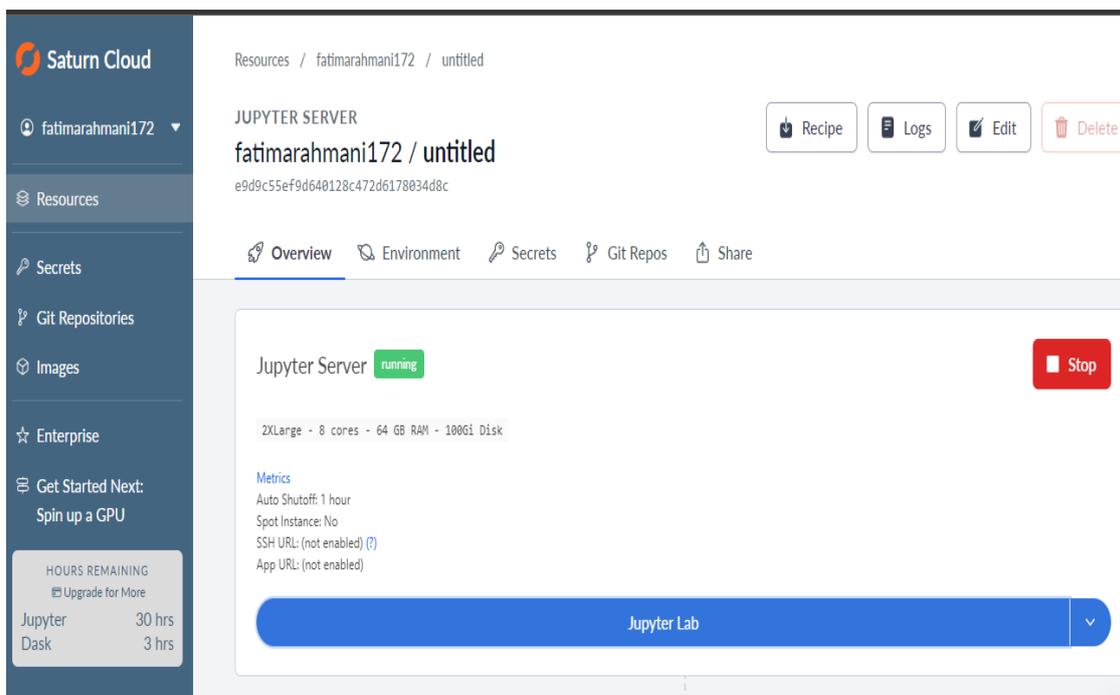


Figure 4-16 : L'environnement Saturn cloud

Pour évaluer la performance de notre approche proposée, nous avons exploré 17 923 service web réels de domaine étiquettes de ProgrammabelWeb jusqu'au premier juillet 2018, nous avons mené les expérimentations sur les 20 premières catégories, l'ensemble de données expérimentales contient 8 459 services Web, après la suppression des mots vides dans les descriptions de services, le moyen nombre de mots dans les descriptions de service est de 38,68 [75].

i prim cate	api_desc	api_name	api_sc_name	api_original_desc
4	webteh croatian base iso/msp/psp regist lic...	Payments	['Financial']	webteh is croatian based isomspssp register...
11	site beenverified.com api oauth protocol de...	Security	['Identity', 'Authentication']	from their site the beenverifiedcom api fol...
4	payplug api integr payment websit notifi cu...	Payments	['Integration', 'Notifications', 'Refunds']	the payplug api integrates payments into we...
11	mozilla persona onlin ident servic websit w...	Security	['Identity', 'Authentication']	mozilla persona is an online identity servi...
7	british oceanograph data centr bodc nation ...	Mapping	['Marine', 'England']	the british oceanographic data centre bodc ...
14	site "mimvi complet commit close collabor d...	Search	['Mobile']	from the site mimvi is completely committed...
0	googl acceler mobil cach api retriev list a...	Tools	['Application Development', 'Open Source', 'Platform-as-a-Service']	the google accelerated mobile pages cache a...
9	knowledg discoveri framework kdf servic u. ...	Science	['Energy']	the knowledge discovery framework kdf servi...
17	core connect repositori facilit free access...	Education	['Library', 'Reference', 'Science']	core connecting repositories facilitates fr...
8	wildnet queensland' databas name conserv st...	Government	['Animals', 'Database', 'Search']	wildnet is queenslands database of the name...
0	webthumbnail.org free webservic captur webs...	Tools	['Screenshots', 'Images', 'Images']	webthumbnailorg is a free webservice for ca...
5	globalgiv onlin marketplac connect countri ...	Social	['Non-Profit']	globalgiving is an online marketplace that ...
8	servic retriev repres u. congress legislatu...	Government	nan	the service retrieves information about rep...
8	shropshir council websit onlin portal shrop...	Government	['England']	the shropshire council website is the onlin...
0	hackerspac community-oper physic peopl meet...	Tools	['Community', 'Collaboration']	hackerspaces are communityoperated physical...
1	mincoin tool crypto-curr mine pool user com...	Financial	['Currency', 'Bitcoin', 'Stocks']	mincoin tools is a cryptocurrency mining po...
0	alau.m io app referral-track platform desig...	Tools	['Analytics', 'Networking', 'Social']	alau.m is an ios app referraltracking platf...
5	gnip onlin provid social data custom data g...	Social	['Activity Streams', 'Data']	gnip is an online provider of social data i...
2	tcs' cloud messag center rest api develop e...	Messaging	['Advertising', 'Aggregation', 'API', 'Appl...	tcs cloud messaging center rest api allows ...
13	250ok softwar compani special email deliver...	Email	['File Sharing']	is a software company that specializes in ...

Figure 4-17 : Ensemble de données utilisées de l'approche TextING

Nous avons utilisé la colonne (api_original_desc) qui contient toutes les descriptions de 20 services web et pour faire la prédiction nous avons utilisé la colonne (api_prim_cate) qui explique le numéro de descriptions de services web.

6- Discussion des résultats :

Pour réaliser nos résultats, nous avons opté pour les paramètres suivants :

- Optimiseur d'erreur : Adam
- Fonction d'erreur : Cross_Entropy.
- Taille de Batch : 64
- Taille de Drop-out : 0.25
- Learning_rate : 0.001

- Nombre d'Epochs : 11

Ensuite, nous avons chargé les fichiers des graphes enregistrés pour faire l'entraînement.

```
train_adj, train_feature, train_y, val_adj, val_feature, val_y, test_adj, test_feature, test_y = load_data(FLAGS.dataset)
```

Figure 4-18 : Le chargement de graphe

Puis faire entraînement, la figure 4.18 illustre l'entraînement de notre modèle.

```
train start...
Epoch: 0001 train_loss= 2.69526 train_acc= 0.18668 val_loss= 2.09548 val_acc= 0.38387 test_acc= 0.37983 test_loss= 2.08379
Epoch: 0002 train_loss= 1.94846 train_acc= 0.41137 val_loss= 1.59154 val_acc= 0.55633 test_acc= 0.53349 test_loss= 1.58848
Epoch: 0003 train_loss= 1.57598 train_acc= 0.52063 val_loss= 1.40417 val_acc= 0.60501 test_acc= 0.57289 test_loss= 1.36646
Epoch: 0004 train_loss= 1.42882 train_acc= 0.56483 val_loss= 1.30122 val_acc= 0.62309 test_acc= 0.62411 test_loss= 1.25116
Epoch: 0005 train_loss= 1.31111 train_acc= 0.60037 val_loss= 1.25380 val_acc= 0.61892 test_acc= 0.62411 test_loss= 1.19553
Epoch: 0006 train_loss= 1.24180 train_acc= 0.62278 val_loss= 1.17852 val_acc= 0.64256 test_acc= 0.64460 test_loss= 1.13617
Epoch: 0007 train_loss= 1.17941 train_acc= 0.63931 val_loss= 1.16523 val_acc= 0.64673 test_acc= 0.66273 test_loss= 1.11494
Epoch: 0008 train_loss= 1.12597 train_acc= 0.65569 val_loss= 1.09789 val_acc= 0.65647 test_acc= 0.67061 test_loss= 1.05170
Epoch: 0009 train_loss= 1.07905 train_acc= 0.66837 val_loss= 1.08938 val_acc= 0.67455 test_acc= 0.68558 test_loss= 1.01596
Epoch: 0010 train_loss= 1.04792 train_acc= 0.67393 val_loss= 1.06865 val_acc= 0.67177 test_acc= 0.68794 test_loss= 1.00593
Epoch: 0011 train_loss= 1.01386 train_acc= 0.68227 val_loss= 1.06920 val_acc= 0.67455 test_acc= 0.68637 test_loss= 1.00175
Early stopping...
```

Figure 4-19 : L'entraînement de modèle TextING

6-1 Comparaison entre les modèles :

Nous évaluons les résultats de la classification multi classe, selon quatre mesures (Recall, f1-score, la pureté, l'information mutuelle normalisée (NMI)).

- **Recall** : le Recall nous permet de connaître le pourcentage de positifs que le modèle a bien prédit.

$$\text{Recall} = \frac{\text{vrai positif}}{\text{vrai positif} + \text{faux négatif}}$$

Plus il est élevé, plus le modèle d'apprentissage automatique peut maximiser le nombre de vrais positifs.

- **Précision** : il peut connaître le nombre de prédictions correctes faites.

$$\text{Précision} = \frac{\text{vrai positif}}{\text{vrai positif} + \text{faux positif}}$$

Plus il est élevé, plus le modèle d'apprentissage automatique peut réduire le nombre de faux positifs.

- **F1_score** : le score F1 fournit une bonne évaluation des performances du modèle [85].

$$\text{F1_score} = 2 * \frac{\text{Recall} * \text{precision}}{\text{Recall} + \text{precision}}$$

- **La pureté** : calcule la proportion de services correctement regroupés par rapport au nombre total de services [84].
- **NMI** : L'information mutuelle normalisée (NMI) est une normalisation du score d'information mutuelle (MI) pour mettre les résultats à l'échelle entre 0 (aucune information mutuelle) et 1 (corrélation parfaite). Dans cette fonction les informations mutuelles sont normalisées par une moyenne généralisée de $H(\text{labels_true})$ et $H(\text{labels_pred})$, définie par la méthode `average_method` [86].

Nous comparerons nos résultats avec les modèles qui ont utilisé la même méthode que la nôtre et qui ont expérimenté sur les descriptions des services web (`api_orignal_desc`) et se prédire les classes (`api_prim_cate`), ces modèles sont :

- DeepWSC (TextCNN, LDA) [84].
- DeepWSC (RCNN, LDA) [84].
- RCNN+WE-LDA+K [75].

Les résultats seront affichés dans le tableau suivant.

Modèles	Pureté	NMI	Recall	F1_score
DeepWSC (TextCNN, LDA)	0.5400	0.4625	0.3484	0.3662
DeepWSC (RCNN, LDA)	0.5492	0.4704	0.3614	0.3784
RCNN+WE-LDA+K	0.5708	0.4856	0.3821	0.3969
TextING	0.6887	0.5986	0.6864	0.6714

Tableau 1 : Comparaison des performances de classification des modèles.

6-2 Discussion de résultats de tableau :

Le tableau présente les performances de notre modèle ainsi que les autres modèles, nous avons comparé nos résultats avec les méthodes de papier [84]. Et la méthode de papier [75]. respectivement, nous avons constaté une amélioration par rapport la méthode DeepWSC (TextCNN, LDA) par avantage moyen de 59.321% pour tous les mesures d'évaluations, de plus le TextING obtient un avantage moyen de 55% sur DeepWSC (RCNN, LDA) et 48.18% sur RCNN+WE-LDA+K, pour tous les mesures d'évaluations, cela prouve la supériorité des

performances de notre approche, et ce qui suggère que notre modèle bénéficie au traitement du texte et plus convivial pour les mots invisibles.

Les deux figures 4.20 et 4.21 affichent respectivement le graphe d'accuracy et de perte de modèle TextING pour la classification des services web,

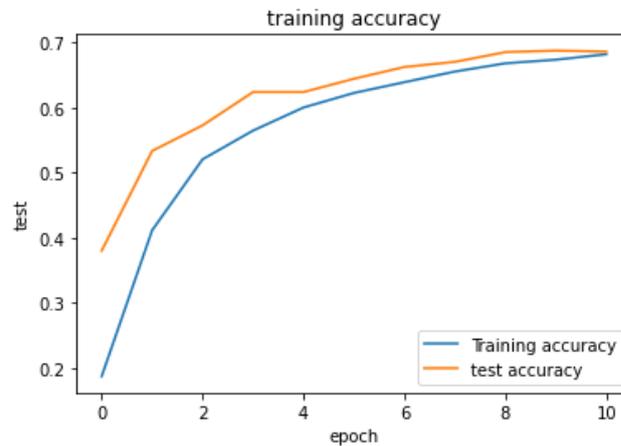


Figure 4-20 : Le graphe d'accuracy

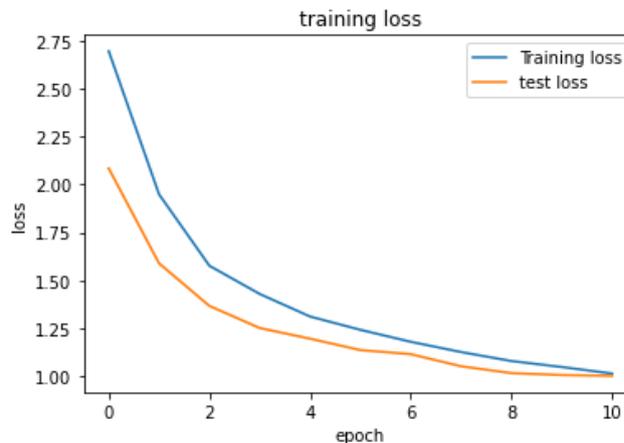


Figure 4-21 : Le graphe de perte

7-Conclusion :

Tout au long de ce chapitre, nous avons présenté notre approche TextING basée sur les réseaux de neurones de graphes (GNN) pour la classification de texte inductive pour améliorer les performances de la classification des Services Web, puis en expliquant toutes les étapes de réalisation, en commençons par l'extraction de Dataset à partir de programmable Web, le nettoyage de nos données, ensuite la construction des graphes pour l'entraînement, la validation et le teste, puis faire l'entraînement de notre modèle pour faire la classification

multiclasse et prédire les classes, finalement nous avons comparé les performances de notre modèle avec des autres méthodes.

Conclusion générale

Conclusion générale et perspective :

Dans notre projet de fin d'étude, on 'a étudié une nouvelle approche pour la classification des services web basée sur la description textuelle de chaque service. Tout d'abord nous avons présenté un état de l'art sur les services web qui comprend sa définition et ses technologies, ainsi que ces grands challenges, tels que la fonction de découverte des services Web qui est le responsable de la recherche. Ensuite, nous avons fait un survol sur plusieurs techniques différentes de Machine Learning en se focalisant plus précisément sur l'apprentissage profond (Deep Learning).

Pour commencer, nous avons utilisé la méthode (TextING) pour la classification de texte inductive basée sur les réseaux de neurones de graphes (GNN), qui construit des graphes, où les interactions de mots dans chaque graphe sont prises en compte, Pour faire une classification « multi- Classes ». Ce modèle permet de prédire les classes dont un service a de fortes chances d'y appartenir, en l'entraînant avec une Dataset issue du monde réel collectée à partir du répertoire des APIs « ProgrammableWeb.com ». Par la suite nous avons présenté nos résultats qui étaient satisfaisants après les avoir comparées avec d'autres travaux.

Perspective :

Comme perspective à notre travail, nous essaierons de combiner notre approche avec les nouvelles techniques d'Embedding telle que Bert, qui est un modèle de représentation de textes écrit en langage naturel, avec un contexte bidirectionnel, qui prend en considération le contexte des mots des textes plus longs, toujours en gardons la structure des graphes.

Références Bibliographiques

Bibliographies :

- [1]: NaghmehNiknejad^aWaidahIsmail^aImranGhani^bBehzadNazari^cMahadiBahari^cAb Razak Bin CheHussin, Understanding Service-Oriented Architecture (SOA): A systematic literature review and directions for further investigation, July 2020, 101491,
- [2]: Nicolas Salatge. Conception et mise en oeuvre d'une plate-forme pour la sûreté de fonctionnement des services Web. Réseaux et télécommunications [cs.NI]. Institut National Polytechnique de Toulouse - INPT, 2006. Français. fftel-00135748.
- [3]: Cyrielle Lablanche Florens Seine Sébastien Gastaud, Université de Nice-Sophia Antipolis Licence d'Informatique — 3ème année, Les Web Services,2004–2005
- [13]: Mohand-Said Hacid, Freddy Lécué, Alain Léger, Christophe Rey, Farouk Toumani, * Université Claude Bernard Lyon, Les web services sémantiques, automate et intégration. I. Introduction, éléments et scénarios, découverte de services web, mars 2009.
- [14]: THESE Présentée pour obtenir le grade de DOCTORAT EN SCIENCES EN INFORMATIQUE Par REZEG khaled THEME Découverte des services dans les systèmes d'information géographiques répartis sous réseau ad-hoc, Université Mohamed Khider Biskra (18/01/2011).
- [15]: A. Friesen, M. Altenhofen, «SEMANTIC DISCOVERY OPTIMIZATION Matching composed semantic web services at publishing time», WEBIST pp 347-350,2005.
- [16]: Thèse de Doctorat CBR4WSD : Une approche de découverte de services Web par Raisonnement à Partir de Cas Présentée et soutenue publiquement le 13 Novembre 2014 par Ibrahim EL BITAR, UNIVERSITE MOHAMMED V ECOLE MOHAMMADIA D'INGENIEURS RABAT Centre d'Etudes Doctorales Sciences et Techniques pour l'Ingénieur.
- [19]: Thèse en Doctorat en Science présentée pour l'option du grade de Docteur en Informatique Temglit Nacera, La composition dynamique des services web, Université des sciences et de la technologie Houari Boumediene, faculté d'Electronique et d'Informatique (19/10/2019).
- [20]: Mémoire présenté comme exigence partielle de la matière en informatique, Aniss Alkamari, Composition de service web par appariement de signature, Université du Québec à Montréal (01/2008).
- [21] Batta Mahesh, Machine Learning Algorithms – A Review, ResearchGate impact Factor (2018) :0.28 | SJIF (2018) : 7.426
- [26] Thierry Foucart, COLINÉARITÉ ET RÉGRESSION LINÉAIRE, 01/03/2006
- [27] Josiane Confais, Monique Le Guen, Premiers pas en régression linéaire avec SAS®,22/10/2007.
- [31] Chloé-Agathe Azencott, Introduction au machine learning.
- [34] Y. Bennani, S. Guérif, E. Viennet Université Paris 13, LIPN - CNRS UMR 7030 99, avenue J.B. Clément, F-93430 Villetaneuse, Réduction des dimensions des données en apprentissage artificiel
- [35] Dominique HERVOUET, Mémoire présenté en vue d'obtenir le diplôme d'ingénieur en informatique Option système d'information, Visualisation des règles d'association en environnement virtuel 3D interactif, (18/02/2011).
- [39] Python Deep Learning. Valentino Zocca, Gianmario Spacagna, Daniel Slater, Peter Roelants. 2017

Références Bibliographiques

- [45] A thesis submitted to McGill University in partial fulfilment of the requirements of the degree of Master of Science. Raviteja Chunduru, Attentive Temporal Abstraction in Reinforcement Learning,(23/11/2020).
- [46] Bruno Bouzy, Apprentissage par renforcement (3),12/10/2010
- [47] Deep learning JOHN D. KELLEHER,2019
- [48] Youcef djeriri, Université de Sidi-Bel-Abbès, Les réseaux de neurones Artificiels 09/2017.
- [49] Marc Parizeau, Département de génie électrique et de génie informatique l'Université Laval, Le perceptron multicouche et son algorithme de rétropropagation des erreurs,10/09/2004.
- [53] Mohamed Bouaziz, Réseaux de neurones récurrents pour la classification de séquences dans des flux audiovisuels parallèles,23/04/2018.
- [55] Martha Varguez-Moo, Francisco Moo-Mena et Victor Uc-Cetina Facultad de Matematicas, Universidad Autonoma de Yucatan, Merida, Mexique, Use of Classification Algorithms for Semantic Web Services Discovery, JOURNAL OF COMPUTERS, VOL. 8, NO. 7, JULY 2013.
- [56] Yves Mercadier. Classification automatique de textes par réseaux de neurones profonds : application au domaine de la santé. Intelligence artificielle [cs.AI]. Université Montpellier, 2020. Français. fNNT : 2020MONTS068ff. fftel-03145856f
- [59] Shiyao Wang, Minlie Huang, Zhidong DengDensely Connected CNN with Multi-scale Feature Attention for Text Classification
- [60] Pengfei Liu, Xipeng Qiu, Xuanjing Huang Recurrent Neural Network for Text Classification with Multi-Task Learning.
- [62] Siwei Lai, Liheng Xu, Kang Liu, Jun Zhao,Recurrent Convolutional Neural Networks for Text Classification
- [63]Lianzhe Huang, Dehong Ma, Sujian Li, Xiaodong Zhang and Houfeng WANG MOE Key Lab of Computational Linguistics, Peking University, Beijing, 100871, China {hlz, madehong, lisujian, zxdc, Text Level Graph Neural Network for Text Classification
- [65] Lianzhe Huang, Dehong Ma, Sujian Li, Xiaodong Zhang and Houfeng WANG, Text Level Graph Neural Network for Text Classification
- [66] Ke Zhao, Lan Huang, Rui Song, Qiang Shen, et Hao Xu,A Sequential Graph Neural Network for Short Text Classification.
- [67] Yufeng Zhang, Xueli Yu, Zeyu Cui, Shu Wu, Zhongzhen Wen et Liang Wang,Every Document Owns Its Structure: Inductive Text Classification via Graph Neural Networks.
- [68] Liang Yao, Chengsheng Mao, Yuan Luo, Graph Convolutional Networks for Text Classification.
- [69]G. Xuyang, Y. Junyang and X. Shuwei, "Text Classification Study Based on Graph Convolutional Neural Networks," 2021 International Conference on Internet, Education and Information Technology.
- [70]Hongfan Ye 1 , Buqing Cao1, Junjie Chen1 , Jianxun Liu1 , Yiping Wen1 , Jinjun Chen1, 2 Australia,A Web Services Classification Method based on GCN,2019

Références Bibliographiques

- [71]K. Punitha,A Novel Mixed Wide and PSO-Bi-LSTM-CNN Model for the Effective Web Services Classification
- [72] Yilong Yang, Wei Ke, Weiru Wang†, Yongxin Zhao Faculty Deep Learning for Web Services Classification,Webology, Volume 17, Number 2, December, 2020
- [73]Wang, X., Liu, J., Liu, X., Cui, X., Wu, H. (2020). A Spatial and Sequential Combined Method for Web Service Classification,2020.
- [74]Bing Li, Zhi Li, YilongYang ,Residual attention graph convolutional network for web services classification,14/06/2021
- [75]Guobing Zou, Zhen Qin, Qiang He, Pengwei Wang, Bofeng Zhang1, Yanglan Gan, 2019,DeepWSC: A Novel Framework with Deep Neural Network for Web Service Clustering.
- [76] Yang Yu, Jun Zeng, Juan Yao, Junhao Wen, Bin Xing, Web Service Discovery Based on Knowledge Graph and Similarity Network, 2020 IEEE World Congress on Services (SERVICES).
- [81]Daoming Zong,Shiliang Sun,2020, GNN-XML: Graph Neural Networks for Extreme Multi-label Text Classification.
- [82]Lu Zhang, Jiandong Ding, Yi Xu, Yingyao Liu, Shuigeng Zhou, 2021, Weakly-supervised Text Classification Based on Keyword Graph.
- [83] Xien Liu, Xinxin You, Xiao Zhang, Ji Wu, Ping Lv, 2020, Tensor Graph Convolutional Networks for Text Classification.
- [84] Guobing Zou, Zhen Qin, Qiang He, Pengwei Wang, Bofeng Zhang, Yanglan Gan,2020, DeepWSC: Clustering Web Services via Integrating Service Composability into Deep Semantic Features

Références Bibliographiques

Webographies:

[4]: Service Web - Définition et Explications, <https://www.techno-science.net/glossaire-definition/Service-Web.html> (dernier consultation 20/03/2020)

[5]: W3C World Wide Web Consortium; “Web Services Architecture”; W3C Working Group

Note 11; February 2004, <https://www.w3.org/TR/ws-arch/> (dernier consultation 20/03/2020)

[6]: Les Services Web, 12/10/2016, <https://openclassrooms.com/courses/lesservicesweb> (dernière consultation 26/04/2022).

[7]: La Rédaction TechTarge, DEFINITION SOAP (Simple Object Access Protocol), <https://whatis.techtarget.com/fr/definition/SOAP-Simple-Object-Access-Protocol> (dernier consultation 20/03/2020)

[8]: F5 GLOSSARY, SOAP (Simple Object Access Protocol), https://www.f5.com/fr_fr/services/resources/glossary/simple-object-access-protocol-soap

(dernier consultation 20/03/2020)

[9]: IBM Integration Bus, Version 10.0.0.24, Last Updated: 2022-04-07, The structure of a SOAP message, <https://www.ibm.com/docs/en/integration-bus/10.0?topic=soap-structure-message> (dernier consultation 20/03/2020)

[10]: RESTFUL web service, <http://mgharzouli.e-monsite.com/medias/files/rest-et-restful-ws.pdf>: (dernier consultation 20/03/2020)

[11]: L’architecture REST expliquée en 5 règles : dernier consultation 20/03/2020

<https://blog.nicolashachet.com/developpement-php/larchitecture-rest-expliquee-en-5-regles/>

(dernier consultation 20/03/2020)

[12] 2009 - 2022 by Thomas ROSSIER, SOAP vs. REST : choisir la bonne architecture web services, <http://thomasrossier.fr/papers/API-soap-vs-rest-les-principales-differences.php>

[17]: 29/04/2022, Sélection des services Web composites basée sur QoS, <https://www.rapport-gratuit.com/selection-des-services-web-composites-basee-sur-qos/>

(dernier consultation 20/03/2020)

[18]: 29/04/2022, Sélection des services web, <https://www.rapport-gratuit.com/selection-des-services-web/> (dernier consultation 20/03/2020)

[22] Laura P, La Data Analyse ou Analyse de données, 22/02/2012, <https://datascientest.com/data-analysis-tout-savoir> (Dernière consultation 24/04/2022)

[23] par La rédaction Tech Target, Analytique Big Data, copyright 2007-2022, <https://www.lemagit.fr/definition/Analytique-Big-Data> (Dernière consultation 24/04/2022)

[24] Analyse des données, http://www.statelem.com/analyse_des_donnees.php (Dernière consultation 24/04/2022)

[25] XLSTAT liste des fonctionnalités, <https://www.xlstat.com/fr/solutions/fonctionnalites/> (Dernière consultation 24/04/2022)

Références Bibliographiques

- [28] RegressionLineaireSimple1, https://user.ocstatic.com/upload/2017/11/07/15100617101762_RegressionLineaireSimple1.jpeg (Dernière consultation 24/04/2022)
- [29] Amandine Allmang, les principaux algorithmes de régression pour l'apprentissage supervisé, 2022, <https://fr.linedata.com/les-principaux-algorithmes-de-regression-pour-lapprentissage-supervise> (Dernière consultation 24/04/2022)
- [30] Machine learning with python –technique , [Machine Learning with Python Tutorial \(tutorialspoint.com\)](https://www.tutorialspoint.com/machine-learning-with-python/) (Dernière consultation 24/04/2022)
- [32] Rohit Sharma, what is clustering and different types of clustering methods, 01/12/2020, <https://www.upgrad.com/blog/clustering-and-types-of-clustering-methods/> (Dernière consultation 24/04/2022)
- [33] Imad Dabbura, 17/09/2018, K-means Clustering :Algorithm, Application, Evaluation methods, and Drawbacks, [K-means Clustering: Algorithm, Applications, Evaluation Methods, and Drawbacks | by Imad Dabbura | Towards Data Science](https://www.analyticsinsights.io/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-by-imad-dabbura-towards-data-science/) (Dernière consultation 24/04/2022)
- [36] Qu'est-ce que le machine learning , <https://www.oracle.com/dz/data-science/machine-learning/what-is-machine-learning/> (Dernière consultation 24/04/2022)
- [37] Machine Learning – Définition, fonctionnement et secteur d'application, par la rédaction, 11/01/2022, 10h00, <https://intelligence-artificielle.com/machine-learning-definition> (Dernière consultation 24/04/2022)
- [38] Guillaume Saint-Cirgue, 11/03/2019, Comment fonctionne le machine learning ?, <https://machinelearnia.com/comment-fonctionne-machine-learning/> (Dernière consultation 24/04/2022)
- [40] Introduction to Support Vector Machine (SVM) dernier mise à jour, 16/07/2020, <https://www.geeksforgeeks.org/introduction-to-support-vector-machines-svm/> (Dernière consultation 24/04/2022)
- [41] Antony Christopher 02/02/2021, K-Nearest6Neighbor, <https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4> (Dernière consultation 24/04/2022)
- [42] Zakaria Ismaili, machine learning ou Apprentissage Automatique, <https://analyticsinsights.io/apprentissage-supervise-vs-non-supervise/> (Dernière consultation 24/04/2022)
- [43] introduction à l'apprentissage automatique, https://projeduc.github.io/intro_apprentissage_automatique/introduction.html (Dernière consultation 24/04/2022)
- [44] David s, Q-learning – Le machine learning avec apprentissage par renforcement, 29/03/2021, <https://datascientest.com/q-learning-le-machine-learning-avec-apprentissage-par-renforcement> (Dernière consultation 24/04/2022)
- [50] Gundamotoko, Le perceptron multicouches | Le perceptron multi-couches – Deep learning –le perceptron multicouche, 05/12/2019, <https://kongakura.fr/article/Le-perceptron-multicouches> (Dernière consultation 24/04/2022)
- [51] Les réseaux de neurones convolutifs, 17/04/2018, <https://www.natural-solutions.eu/blog/la-reconnaissance-dimage-avec-les-reseaux-de-neurones-convolutifs> (Dernière consultation 24/04/2022)

Références Bibliographiques

- [52] Priyal Walipita, Recurrent Neural Networks in Deep Learning-part 1,26/03/2020, <https://medium.datadriveninvestor.com/recurrent-neural-networks-in-deep-learning-part-1-df3c8c9198ba> (Dernière consultation 24/04/2022)
- [54] Yassine Touzani, Deep learning, Les réseaux de neurones récurrents,25/08/2021, <https://datavalue-consulting.com/deep-learning-reseaux-neurones-recurrents-rnn> (Dernière consultation 24/04/2022).
- [57]Text classification: What is and why it matters ,MonkeyLearn Inc. All rights reserved 2022<https://monkeylearn.com/text-classification/>(dernier consultation 27/05/2022)
- [58]SWAYAM , MITTAL Deep Learning Techniques for Text Classification,Aug 16, 2019,<https://medium.datadriveninvestor.com/deep-learning-techniques-for-text-classification-9392ca9492c7/>(dernier consultation 27/05/2022)
- [61]<https://ychai.uk/notes/images/RCNN.png>(dernier consultation 27/05/2022)
- [64] Yugesh Verma ,How to Use Graph Neural Networks for Text Classification?, publier en 07/11/2021 , dans DEVELOPERS CORNER,<https://analyticsindiamag.com/how-to-use-graph-neural-networks-for-text-classification/>(dernier consultation 27/05/2022)
- [77] DERRICK MWITI, The Essential Guide To GNN(Graph Neural Network),<https://cnvrg.io/graph-neural-networks/?fbclid=IwAR3Hoh5F4nQqI36beZhLgfz69o4PHK1I7B1tzVQFwxwe6sd0pT4LVld-01yg> (dernier consultation 27/05/2022)
- [78] : Shanon Hong , An Introduction to Graph neural Network (GNN) for Analysing Structered Data , 05/03/2020, <https://towardsdatascience.com/an-introduction-to-graph-neural-network-gnn-for-analysing-structured-data-afce79f4cfdc> (dernier consultation 27/05/2022)
- [79] Ben Dickson , What are Graph neural Network (GNN) ? 11/10/2021,<https://bdtechtalks.com/2021/10/11/what-is-graph-neural-network/>
- [80] Zichen Wang, Vassilis N.Ioannidis, How AWS uses graph neural networks to meet customer needs, 24/03/2022, <https://www.amazon.science/blog/how-aws-uses-graph-neural-networks-to-meet-customer-needs> (dernier consultation 27/05/2022)
-
- [85] TOM KELDENICH,02/09/5/2021, Recall, Precision, F1 Score – Explication Simple Métrique en ML, <https://www.inside-machinelearning.com/recall-precision-f1-score/> (dernier consultation 13/06/2022)
- [86]Sklearn.metrics.normalized_mutual_info_score, https://scikitlearn.org/stable/modules/generated/sklearn.metrics.normalized_mutual_info_score.html (14/06/2022)