REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITÉ MOHAMED BOUDIAF - M'SILA

Faculté des Mathématiques et de l'Informatique

Département d'Informatique

N° : …….

DOMAINE : Mathématiques et Informatique

FILIERE : Informatique

OPTION : Systèmes d'Information et Génie Logiciel

Mémoire présenté pour l'obtention
Du diplôme de Master Académique

Par : BERROUBI ABDELAZIZ

BEN LATRACHE SAMIYA

Intitulé

Vers un Système pour le Résumé Automatique des Textes Arabes

Soutenu devant le jury composé de :

……	Université de M'sila	Président
Dr. Mahmoud BRAHIMI	Université de M'sila	Rapporteur
……	Université de M'sila	Examinateur

Année universitaire : 2021/2022

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ MOHAMED BOUDIAF - M'SILA

Faculté des Mathématiques et de l'Informatique

Département d'Informatique

N° : …….

DOMAINE : Mathématiques et Informatique

FILIERE : Informatique

OPTION : Systèmes d'Information et Génie Logiciel

Mémoire présenté pour l'obtention
Du diplôme de Master Académique

Par : BERROUBI ABDELAZIZ

BEN LATRACHE SAMIYA

Intitulé

Vers un Système pour le Résumé Automatique des Textes Arabes

Soutenu devant le jury composé de :

……	Université de M'sila	Président
Dr. Mahmoud BRAHIMI	Université de M'sila	Rapporteur
……	Université de M'sila	Examinateur

Année universitaire : 2021/2022

TABLE DES MATIERES

Chapitre II : Résumé automatique du texte

Chapitre III : Conception et implémentation du système

LISTE DES FIGURES

LISTE DES TABLEAUX

INTRODUCTION GENERALE

Introduction Générale

Face à l'émergence d'Internet et à l'évolution des moteurs de recherche, l'information textuelle au format électronique s'est accumulée rapidement et en grande quantité. Il est donc intéressant de proposer des outils informatiques de visualisation rapide de texte, comme le résumé automatique (en compressant le texte de manière pertinente), afin que les utilisateurs puissent apprécier la pertinence des documents par rapport à l'information recherchée.

Un résumé est un texte concis qui reflète le contenu "de base" d'un autre texte, appelé texte source. En fait, le but du résumé est d'aider le lecteur à déterminer si le document source contient l'information recherchée. Il se peut également que le lecteur n'ait pas besoin de lire l'intégralité du document source simplement parce que l'information recherchée est présente dans le résumé.

Notez que le résumé automatique motive différentes directions. En fait, plusieurs approches ont été explorées en linguistique et en statistique. Cependant, la plupart des travaux dans le domaine du résumé reposent sur des extractions plus pratiques et plus utilisées (par exemple, l'indexation).

Dans le cadre du traitement automatique du langage naturel (TALN), plus précisément, du résumé automatique de documents arabes pour lesquels le sujet de notre travail est approprié. Compte tenu des caractéristiques et des difficultés de l'arabe, notre objectif est d'étudier et de proposer une méthode hybride (statistique) pour résumer automatiquement les documents rédigés en arabe.

Aujourd'hui, la plupart des systèmes de résumé automatique traitent des textes en langues indo-européennes (anglais, français, etc.). Compte tenu de l'augmentation du contenu rédigé en arabe, la nécessité de développer un système de résumé automatique dédié à l'arabe est devenue de plus en plus important ces dernières années

La question qui doit être résolue à partir de là est de savoir comment générer des résumés qui répondent aux besoins des utilisateurs ? Compte tenu du choix de l'utilisateur, quelles méthodes d'extraction ou combinaisons de méthodes conviennent au texte arabe ? Pour résoudre ce problème et mettre en place un système de résumé automatique de texte en arabe, nous utilisons des techniques d'extraction déjà utilisées dans (TALN). Pour aboutir à des réponses à toutes ces questions, le reste de ce manuscrit sera organisé comme suit :

Dans le premier chapitre, nous allons présenter le domaine du traitement automatique des langages naturels (TALN) avec toutes ses spécificités, ses techniques, ses défis et ses

diverses exploitations tout en mettant l'accent sur la langue arabe avec ses difficultés propres.

Nous focalisons dans le deuxième chapitre sur notre étude consacré au résumé automatique des textes avec ses approches et ses techniques fournies.

La conception et l'implémentation de notre système seront présenté dans le troisième chapitre avec la présentation des outils utilisés et des résultats obtenus.

Finalement, nous concluons ce travail par une conclusion et quelques suggestions pour rehausser les performances du système développé au future.

CHAPITRE I

TRAITEMENT AUTOMATIQUE DU LANGAGE NATUREL

Chapitre I : traitement automatique du langage naturel (TALN)

1. Introduction :

Les dernières années ont connu une évolution très rapide dans le contenu du Web. La masse informationnelle est devenu de plus en plus volumineuse où le texte constitue la partie essentielle de cette masse. De ce fait, l'utilisateur du Net doit avoir des outils rapides lui permettant de bien chercher, bien classer et bien comprendre ce contenu volumineux et même compliqué.

Par conséquent, toute contribution à la classification, au traitement de documents texte et à l'extraction d'informations devient l'objectif principal. C'est dans cette perspective que le domaine du « traitement automatique de langage naturel » est né.

Dans ce chapitre, nous allons présenter ce domaine avec ses axes d'applications et ses spécificités techniques et pratiques.

2. Historique de traitement automatique de langage naturel (TALN) :

Historiquement, les premiers grands travaux dans le domaine du TAL ont porté sur la traduction automatique, avec le développement des premiers traducteurs automatiques (très rudimentaires) dès 1954. Certaines phrases russes présélectionnées sont automatiquement traduites en anglais. Bien que la taille du vocabulaire ne soit que de 250 mots et que la grammaire soit de 6 règles, cette expérience a suscité beaucoup de travail sur le terrain. C'était en effet l'époque où l'Union soviétique réussissait dans la course à l'espace et où l'armée américaine était si désireuse de suivre les publications techniques soviétiques qu'aucun de ses ingénieurs n'apprenait le russe.

Dans les années 1960, SHRDLU était l'un des premiers programmes informatiques pour la compréhension du langage naturel et le meilleur logiciel pour permettre des conversations interactives avec des utilisateurs basées sur la terminologie anglaise. Blocks world, un système de langage naturel basé sur un vocabulaire relativement restreint, fonctionne très bien, garder les chercheurs optimistes.

Les progrès réels, cependant, ont été beaucoup plus lents, avec beaucoup moins d'ambition après que le rapport ALPAC de 1966 ait constaté que les objectifs de recherche sur dix ans n'avaient pas été atteints.

ELIZA est un programme qui simule des entretiens avec des psychiatres, écrit par Joseph Weizenbaum entre 1964 et 1966. Utilisant peu d'informations sur les pensées ou les émotions

Humaines, ELIZA parvient parfois à fournir une représentation étonnante de l'interaction humaine. Lorsque le patient va au-delà de la base de connaissances.

Dans les années 1970, de nombreux programmeurs ont commencé à écrire des "ontologies conceptuelles", dont le but était de structurer les informations en données compréhensibles par les ordinateurs. MARGIE (Schank, 1975), SAM (Cullingford, 1978), PAM (Wilensky, 1978), TaleSpin (Meehan, 1976), SCRUPULE (Lehnert, 1977), Polities (Carbonell, 1979), Plot Units (Lehnert, 1981) D'autres systèmes similaires à ELIZA ont été créés tels que PARADE, Racter et Jabberwacky.

Dès les années 1980, alors que la puissance de calcul augmentait et que les coûts diminuaient, les modèles statistiques pour la traduction automatique ont reçu une attention croissante. [1]

3. Définition de TALN :

Le TALN est l'ensemble des méthodes et des programmes qui permettent un traitement par l'ordinateur des données langagières, mais quand ce traitement tient compte des spécificités du langage humain. Il y a des traitements de données langagières (écritures sur fichiers, sauvegardes ou autres) qui ne font pas partie du traitement automatique des langues. [2]

4. Objectifs de TALN :

Le but du traitement automatique du langage naturel (TALN) est de concevoir le traitement automatique des données linguistiques, c'est-à-dire Une langue.

Ces données linguistiques peuvent être du texte écrit, ou Dialogue écrit ou parlé, encore plus petit que les unités dites linguistiques Généralement du texte (par exemple des phrases, des déclarations, des expressions ou juste des mots isolés). [3]

5. Outils de TALN :

Le traitement automatique des langues naturel nécessite évidemment des outils divers que l'on peut grouper en trois catégories distinctes :

- ✓ **Linguistiques** : ils décrivent les différentes connaissances relatives à la langue. [6]
- ✓ **Formels** : ils expriment les connaissances linguistiques dans un formalisme qui convient à un traitement automatique. [6]

- ✓ **Informatiques** : ils utilisent la description formelle des connaissances dans une application informatique concrète. [6]

6. Domaines de recherche de TALN :

La diversité des outils utilisés pour le traitement automatique a conduit à la diversité du TAL, qui implique des recherches dans différents domaines. [4]

6.1. La linguistique :

C'est une discipline scientifique qui étudie le langage. Il n'est pas prescriptif, mais descriptif. Les prescriptions correspondent à des normes, c'est-à-dire à ce que les grammairiens considèrent comme linguistiquement correct. Au lieu de cela, la linguistique se contente de décrire le langage tel qu'il est, et non tel qu'il devrait être. [4]

6.2. L'informatique :

Est un domaine d'activité scientifique, technique et industriel concerné par le traitement automatique de l'information numérique par l'exécution de programmes informatiques hébergés par des équipements électriques et électroniques : systèmes embarqués, ordinateurs, robots, automates, etc. [4]

6.3. La linguistique informatique :

C'est un domaine interdisciplinaire de la modélisation statistique du langage naturel basée sur des symboles (à base de règles) ou dans une perspective informatique. [4]

6.4. L'intelligence artificielle (IA) :

Toutes les théories et techniques utilisées pour produire des machines capables de simuler l'intelligence humaine. [4]

7. Applications de TALN :

Parmi les applications de TALN on cite :

7.1. L'indexation automatique et la recherche documentaire :

L'essentiel de l'information étant sous forme de texte en langage naturel (références, livres, revues, articles, etc.), l'intérêt d'une automatisation de la recherche bibliographique est évident, qui doit pouvoir retrouver automatiquement l'information, la littérature pertinente, ou les citations de littérature pour répondre aux questions des utilisateurs. [5]

7.2. Le résumé automatique :

Nous allons travailler à l'élargissement de l'indexation et de la recherche automatisée, qui est désormais une discipline à part entière. Il est utilisé pour générer une version compressée du texte d'ascendance africaine qui conserve des informations à son sujet. [5]

7.3. La traduction automatique :

Ou la traduction, l'une des premières applications de la PNL, a été définie comme l'application informatique de la traduction d'un texte parlé ou d'un langage naturel sortant (ou langue source) vers une langue d'entrée (ou langue cible). [5]

7.4. Les correcteurs :

Généralement les correcteurs orthographiques, syntaxiques et stylistiques, qui se limitent à l'analyse linguistique et aident les humains à convertir un texte en un autre, avant tout la correction de texte. Ce sont des outils très populaires, et ils font déjà partie intégrante de la plupart des logiciels de traitement de texte. [5]

7.5. La génération de textes en langue naturelle :

Il est conçu pour générer du texte en langage naturel à partir de données non linguistiques telles que des graphiques, des croquis, des dessins ou des données numériques. [5]

7.6. Les systèmes de dialogue homme-machine :

C'est ce qui permet aux humains de parler aux ordinateurs. Par exemple, il permet d'interroger des bases de données en langage naturel, de contrôler des robots ou des machines, et même de parler à des systèmes experts, etc. [5]

8. Les difficultés du TALN :

Il existe principalement deux types de difficultés rencontrées en TALN, et il y a Soit l'ambiguïté du langage, soit la quantité de contenu implicite communiqué naturellement.

8.1. Ambiguïté :

Le langage naturel est ambigu, peu importe comment nous le comprenons. Cette ambiguïté, loin d'être limite, est une de ses caractéristiques. On y voit aussi le résultat inéluctable du compromis, d'une part le pouvoir expressif quasi illimité, d'autre part les contraintes liées aux contraintes des ressources physiologiques mises en œuvre (taille de la mémoire à court et long terme) la densité des mot, espace phonétique, restrictions de prononciation, etc.). Cette

ambiguïté se manifeste dans les Multiples interprétations possibles de chaque entité linguistique associée au niveau de traitement, comme le montre l'exemple suivant :

- L'ambiguïté du graphème (lettre) lors de l'encodage orthographique : comparer la prononciation de i dans lit, poire et maison
- Ambiguïté des terminaisons de mots lors de la conjugaison et de l'inflexion. Compte tenu de l'ambiguïté des propriétés syntaxiques et sémantiques de la forme
- Graphique (c'est-à-dire par rapport à son sens : ainsi mange est morpho-syntaxiquement ambigu puisqu'il correspond aux formes démonstratives et subjonctives du verbe manger), mais aussi sémantiquement ambigu.
- L'ambiguïté de la fonction grammaticale d'une phrase s'explique par une phrase.
- L'ambiguïté des quantificateurs, des conjonctions et des prépositions.
- L'ambiguïté sur l'interprétation à donner en contexte à un énoncé. [1]

8.2. Implicite :

L'activité langagière s'inscrit toujours dans le contexte d'interaction entre deux personnes et est censée avoir une connaissance du monde et de ses fonctions, de sorte que la désambiguïsation et la grande majorité des éléments contextuels nécessaires à la compréhension des énoncés naturels peuvent rester implicites. Une fois que la machine essaie de s'insérer dans le processus naturel de communication avec les humains, la situation change complètement : La machine n'a pas cette connaissance de base, ce qui rend difficile, voire impossible, la compréhension complète de la plupart des énoncés si vous n'avez pas un base de connaissances complémentaire, avec accès à des connaissances générales sur le monde (ou domaine) (connaissances statiques) et des connaissances sur le contexte des énoncés (connaissances dynamiques). Sans cette connaissance, de nombreux autres problèmes de compréhension deviennent presque insurmontables : Pensez par exemple aux ellipses, aux métaphores et plus généralement aux figures de style.

Heureusement, il existe de nombreuses applications qui limitent dans une large mesure ces difficultés. Dès lors que le cadre d'analyse des textes est restreint à un sous-domaine précis (textes juridiques, textes scientifiques, serveurs d'information spécialisés dans l'information sportive, etc.), d'une part de nombreuses ambiguïtés peuvent être ignorées, notamment sémantiquement (par exemple dans le contexte des textes juridiques) , on pourrait négliger la

possibilité qu'un avocat brun indique que le fruit est un peu trop mûr) ; en revanche, le formel indique une grande connaissance nécessaire pour comprendre l'énoncé dans le domaine considéré. En réalité, le contexte de certains domaines d'activité ou d'interactions spécifiques semble limiter considérablement l'ensemble des déclarations possibles (ou acceptables), simplifie grandement la manipulation de ces vrais sous-langages machine. [1].

9. Les défis de TALN :

La complexité de TALN (Le traitement automatique du langage nature) est due à la nature du langage humain. Règles utilisées Parce que l'échange d'informations en langage naturel n'est pas facile ordinateur, parce qu'il n'est pas facilement compréhensible par celui-ci. Certaines règles peuvent être de haut niveau L'abstraction et certaines règles peuvent être de bas niveau. Dans un sens plus large, comprendre Le langage humain et ses concepts sont essentiel pour transmettre les informations souhaitées. Le traitement automatique du langage nature est difficile à mettre en œuvre avec les ordinateurs car il Complexité et caractéristiques imprécises, bien que les humains puissent facilement le saisir. [7]

10. Les niveaux de traitement dans un système TALN :

Pour traiter le langage naturel, nous avons besoin d'informations linguistiques coordonnées et corrélées différents niveaux. Dans la plupart des cas, quatre niveaux de connaissances linguistiques sont utilisés : vocabulaire morphologique, syntaxe, sémantique et pragmatique. Ces niveaux se chevauchent Chacun apporte des problématiques spécifiques liées à un niveau donné. Cela nous donne hiérarchie suivante :

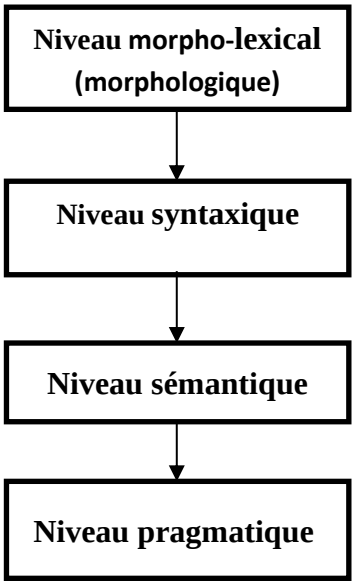

Figure 1.1 Niveaux de traitement de langage naturel.

10.1. Niveau morpho-lexical (morphologique) :

La morphologie explique la structure des mots et ce qu'ils font dans la phrase. L'analyse consiste à segmenter le texte en unités de base Quelle connaissance est attachée au système une fois cette Scission Après exécution, ce n'est plus du texte qui est manipulé, mais une liste ordonnée d'unités. Pour Traitement du texte numérique On commence par une chaîne de caractères imprimés, puis on essaye de le segmenter pour que chaque partie corresponde à une classe dans système. [8**].**

10.2. Niveau syntaxique :

Cela fait partie de la grammaire, comment gérer les mots Combinez pour former des clauses et des clauses de liens.

il consiste à découper la chaîne en unités, reliant les représentations des groupements les relations fonctionnelles entre ces unités et qui unissent ces groupes l'unités (Figure 1.2).[8].

Reprenons l'exemple précédant : «يكتب عمر الدرس», et sa représentation morphologique:

- **U1** =يكتب
- **U2** =عمر
- **U3** = الدرس[8].

Le résultat de l'analyse syntaxique pourra être par exemple l'arbre suivant :

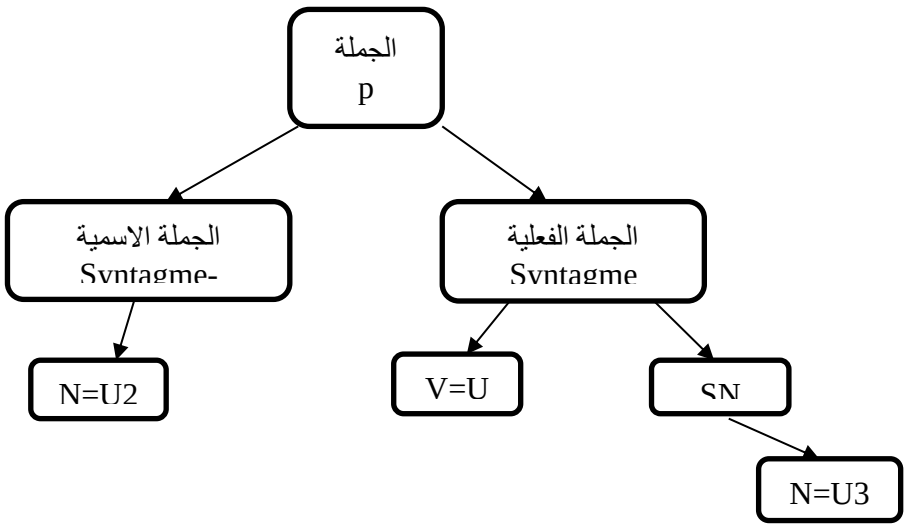

Figure 1.2 Arbre syntaxique de la phrase P

- **P** = « يكتب عمر الدرس»
- **SN** =عمر
- **SV** =يكتب الدرس
- **SN** =الدرس
- **N** =عمر
- **V** =يكتب
- **N** =الدرس

10.3. Niveau sémantique :

L'analyse sémantique est encore beaucoup plus complexe que la description et la formalisation niveaux précédemment prescrits. De ce fait, peu d'outils de traitement restent opérationnels ou Moins, se concentrant sur des applications très limitées où l'analyse sémantique se limite à champ complètement étroit ; cependant, sur la façon dont Construire un analyseur sémantique générique complet couvrant tous arabe, et sera indépendant d'un domaine d'application spécifique. Les phrases sont la principale unité d'analyse soutenue par le traitement sémantique, de sorte qu'en représente une partie importante. Ces phrases, le sens que l'analyseur sémantique doit décrire, sont constituées d'un certain nombre de mots identifiés par l'analyse morphologique, et Il est regroupé en structures par analyse syntaxique. [8]

10.4. Niveau pragmatique :

Ce type de traitement supprime l'ambiguïté qui ne peut pas être éliminée. Par traitement sémantique, en raison de certains problèmes liés au contexte Les phrases sont prononcées en (donnant aux mots des significations contextuelles) ça arrive), c'est-à-dire qu'il est chargé de placer le mot dans tous les contextes En utilisant des informations hors contexte (géographie, sports, travail, …ETC.). L'enchaînement de ces traitements est une idéalisation.

En pratique, le meilleur Considérez ces niveaux de traitement comme des processus coopératifs qui échangent Information bidirectionnelle (du niveau "bas" au niveau "haut", et sens inverse)

Il est donc souvent nécessaire de rappeler des informations sémantiques Trouver la structure syntaxique "correcte" d'une phrase, etc. [8]

11. La langue arabe :

L'arabe (al arabiya en transcription traditionnelle) a été écrit à l'origine par Arabe. C'est une langue sémitique (comme l'akkadien et l'hébreu). Ensemble, il appartient au sous-groupe sémite du sud. En raison de l'expansion Territoire médiéval et diffusion du Coran, la langue de Toute l'Afrique du Nord et l'Asie Mineure. Ainsi, parler arabe, c'est parler un ensemble complexe dans lequel des variantes écrites et parlées se déploient en réponse à une Usage social très diversifié, du plus académique au plus populaire. Mais sinon Cette diversité, la société arabe est profondément consciente d'appartenir à une communauté Linguistique homogène. [12]

12. Particularité de la langue arabe :

En raison de ses propriétés morphologiques et syntaxiques, l'arabe est considéré comme le Langages difficiles à maîtriser dans le domaine du PNL. Les premiers travaux de recherche, commencés vers les années 1970, ont porté sur Dictionnaire et morphologie arabe. Avec l'avènement d'Internet et des moteurs de recherche, le nombre de documents en arabe Le contenu disponible sous forme électronique est devenu très important. Ainsi, de nombreux travaux de recherche Le traitement automatique de l'arabe commence à apparaître. Ce travail implique de multiples directions de recherche telles que la syntaxe et la traduction. Indexation automatique et automatique des documents, recherche d'informations, etc. Ce travail sur le traitement automatique de l'arabe a rencontré des problèmes En raison de la nature cohésive de la langue, de la richesse des changements flexionnels, il y a eu de nombreuses variantes de l'arabe, La plupart des textes arabes écrits sont muets, etc.

Dans la section suivante, nous essaierons de couvrir brièvement ces questions, Cela rend le traitement automatique de l'arabe difficile à maîtriser. [9]

12.1. Les voyelles :

En arabe écrit, les voyelles (diacritiques) sont omises, ce qui donne Les omissions sont que les mots ont tendance à être très ambigus. Cela peut donner Il existe certaines ambiguïtés à deux niveaux : ¬ Vocabulaire. ¬ Difficulté à reconnaître sa fonction dans une phrase, (distinguer sujet du sujet Remplir,…). Cela affecte les fréquences des mots parce qu'ils Calculé après détection de la racine ou lemmatisation des mots, c'est-à-dire Basé sur la suppression des préfixes et des suffixes. Lors du calcul des scores titre, il peut arriver que l'on pense que les mots viennent du même Le concept de quand ils ne le sont pas. Dans l'exemple, le mot distribution est utilisé avec ou sans titre lemmatisé, alors la phrase 3 obtient le score le plus élevé Les phrases 1 et 2 semblent plus intéressantes, alors que pour un texte vocalique.

العنوان: ا ثر العلم. 1- **العلماء....** 2- **علميا....** 3- **بين العلم الوطني والعلم الاجنبي....**	Titre : import de la science 1- Les scientifiques…. 2- Scientifiquement…. 3- Entre le drapeau national et le drapeau étranger…

Figure 1.3 Exemple sur l'effet du mot non voyelle « العلم » sur les extraits. [10]

L'ambiguïté vient du mot العلم science ou signe, alors que la voyelle nous aurons العِلْمُ La science et العَلَمُ sont les drapeaux.

Dans certains cas, cette ambiguïté peut être résolue en Une analyse plus approfondie des phrases ou des statistiques (par exemple, plus probable Il y a " العلم الوطني " le drapeau national que la science nationale).

De plus la capitalisation n'est pas employée dans l'arabe ce qui rend l'identification des noms propres, des acronymes, et des abréviations encore plus difficile.

Comme la ponctuation est rarement utilisée, nous devons ajouter une étape de paragraphe Phrases pour l'analyse de texte. [10]

12.2. Agglutination :

Contrairement au latin, dans les articles arabes, les prépositions, les pronoms, Les ETC s'en tiennent aux adjectifs, noms, verbes et particules auxquels ils se réfèrent. Par rapport à En français, un mot arabe peut parfois correspondre à une phrase française. Exemple : Le mot arabe "أتتذكروننا " correspond en Français à la phrase "Est-ce que vous souvenez de nous ?".

Cette caractéristique peut conduire à une ambiguïté au niveau morphologique. Efficace, Il est parfois difficile de faire la distinction entre les caractères proclitiques ou enclitiques et primitifs ce mot. Par exemple, le caractère " و"dans le mot "وصل"(il est arrivé) est un caractère original alors que dans le mot "وفتح" (il a ouvert), il s'agit d'une proclitique. [11]

12.3. Irrégularité de l'ordre des mots dans la phrase :

L'ordre des mots en arabe est relativement libre. D'une manière générale, on met au début de la phrase le mot sur lequel on veut attirer l'attention et l'on termine sur le terme le plus long ou le plus riche en sens ou en sonorité. Cet ordre provoque des ambiguïtés syntaxiques

artificielles, dans la mesure où il faut prévoir dans la grammaire toutes les règles de combinaisons possibles d'inversion de l'ordre des mots dans la phrase.

Ainsi par exemple, on peut changer l'ordre des mots dans la phrase (Tableau 1.1) pour obtenir deux phrases ayant le même sens. [9]

Verbe + sujet + complément	فعل + فَاعل + متمم	Est allé le garçon à l'école	ذهب الولدُ إِلَى المدرسة
Sujet + Verbe + complément	فَاعل + فعل + متمم	Le garçon est allé à l'école	الولدُ ذهب إِلَى المدرسة
Complément + Verbe + sujet	متمم + فعل + فَاعل	A l'école est allé garçon	إِلَى المدرسة ذهب الولدُ

Tableau 1.1 Exemple de combinaisons possibles d'inversion de l'ordre des mots dans La phrase. [9]

12.4. Absence de ponctuation régulière :

L'arabe n'est pas principalement basé sur la ponctuation et Marques de composition ; il convient de noter qu'elles ne sont pas utilisées d'une certaine manière sont régulières dans les textes arabes courants, et même si elles y figurent, elles ne sont pas Régi par des règles d'utilisation spécifiques. Par contre, on peut trouver un paragraphe entier en arabe sans symboles Signes de ponctuation, à l'exception du point à la fin de ce paragraphe. Par conséquent, il convient de noter que, La présence de ponctuation ne guide pas la segmentation comme elle le fait Pour les autres langues latines comme le français ou l'anglais. Par conséquent, en divisant Le texte arabe ne doit pas être guidé uniquement par la ponctuation et les marques typographie, mais aussi par des petits mots et certains mots comme les conjonctions Coordination. [9]

12.5. Détection de racine :

Pour détecter la racine d'un mot, il est nécessaire de savoir comment il est dérivé et Supprimer les éléments de flexion qui ont été modifiés (préfixe, préfixe, suffixe, suffixe) Ajouter à. J'utilise la liste des préfixes et suffixes proposée par (voir Tableau 1.2). Certains d'entre eux ont été utilisés pour la lemmatisation des mots arabes ; ils sont Déterminés par calcul de fréquence des collections d'articles arabes par les institutions françaises Presse (AFP).

L'analyse morphologique devra donc séparer et identifier des morphèmes semblables aux mots préfixés comme les conjonctions Wa-'و ' et fa- 'ف ', des prépositions préfixées comme bi-'ب ' et li- 'ل', l'article défini 'ال', des suffixes de pronom possessif. La phase d'analyse morphologique détermine un schème possible. Les préfixes et suffixes sont trouvés en enlevant

progressivement des préfixes et des suffixes et en essayant de faire correspondre toutes les racines produites par un schème afin de retrouver la racine.

Préfixes							
لا	في	للا	كم	بم	وت	بت	وال
با	وا	لي	فم	له	ست	يت	فال
	فا	وي	ال	وم	نت	مت	بال
suffixes							
ا	ة	ين	ية	هم	ته	وه	ات
	ه	يه	تك	هن	تم	ان	وا
	ي	ية	نا	ها	كم	تي	ون

Tableau 1.2 La liste de Préfixes et suffixes les plus fréquents.

Lorsqu'un mot peut être dérivé de plusieurs radicaux différents, la détection des radicaux est Encore plus dure, surtout sans voyelles. [11].

12.6. Le caractère '_' :

Les typographes utilisent souvent le caractère "-" (appelé Kashida), qui permet Allonger les lignes entre les mots pour améliorer la lisibilité pour limiter Même pour des raisons purement esthétiques, l'espace blanc peut être laissé sur une ligne raisonnable.

Mais ça Utiliser l'analyse automatique de mal : ce caractère ne fait pas partie de l'alphabet Arabe, considéré comme un intrus par les systèmes d'analyse automatisés. Doit donc Utilisez un sous-programme spécifique pour l'éliminer. Exemple : mot الكتاب : peut-être Écrit de plusieurs façons : ..., الكتاب , الكتـــاب , الـكتاب etc. [11]

12.7. Mots étrangers translittérés en arabe :

Il y a des problèmes avec la translittération arabe des mots étrangers parce qu'ils ne Il n'y a pas de racine en arabe. L'analyseur considère que le mot translittéré est inconnu. Certains projets étrangers méritent une attention particulière en raison de leur fréquence Haute. Exemple : ... أورو ,دولار etc. [11]

12.8. Segmentation de phrase :

Identifier la fin d'une phrase est délicat car la ponctuation n'est pas Systématiquement, parfois des particules délimitent des phrases. Pour la segmentation de texte Objet :

- Segmentation morphologique basée sur la ponctuation,

- basés sur l'identification de marqueurs morphosyntaxiques ou Mots fonctionnels tels que : حتى ,لكن ,أي ,و ,اوou, et, c.à.d. mais, quand.

Cependant, ces particules peuvent servir plus que séparer des phrases. [11]

13. Difficultés de traitement automatique de la langue arabe :

13.1. La segmentation de textes :

La segmentation du texte est l'étape fondamentale de son traitement automatique ; ce qu'il fait est de diviser le texte en types spécifiques d'unités que nous définirons, déterminé auparavant. En fait, les opérations de segmentation de texte incluent la délimitation Diviser ses éléments de base (c'est-à-dire les caractères) en différents éléments constitutifs Niveaux de structure, tels que : paragraphes, phrases, syntaxe, mots graphiques, formes de mots, morphèmes etc. Cependant, la particularité de l'arabe rend la segmentation de l'arabe toujours La différence est qu'il n'y a pas de majuscule au début de la nouvelle phrase. Et, Les signes de ponctuation sont rarement utilisés. Selon les recherches menées Belguith, certaines particules comme "et | و", "donc | ف", etc. est un facteur principal Dans la séparation des phrases, cela peut aider à guider la segmentation. [9]

13.2. L'analyse morphologique :

La morphologie est un niveau essentiel dans les systèmes de traitement automatiques de la langue. L'opération de l'analyse morphologique tient à étudier la forme d'un mot (unités lexicales) en faisant une analyse interne de la structure de ce dernier. Le but étant de décomposer un mot à des éléments plus petits (préfixes, suffixes, etc.) selon des règles de combinaison relatives à ces derniers. À proprement parler, l'analyse morphologique ne fait que la séparation et l'identification des morphèmes semblables aux mots préfixés (comme les conjonctions "wa | وَ " et "fa | فَ " ,etc.), des prépositions préfixées (comme "bi | بِ" et "li | لِ ", l'article défini "ال ", etc.), des suffixes de pronom possessif. La phase d'analyse morphologique détermine un schéma possible. Les préfixes et suffixes sont trouvés en enlevant progressivement des préfixes et des suffixes et en essayant de faire correspondre toutes les racines produites par un schème afin de retrouver la racine.

Ce Le principal problème de cette analyse est l'agglutination et l'absence de voyelles. Pour l'agglutination, contrairement au latin, en arabe les pronoms, prépositions, articles, conjonctions et autres particules adhèrent aux noms, verbes, Les adjectifs et les particules auxquels ils se réfèrent. Comparé au français, un mot arabe peut Correspond parfois à une phrase française.

Cette caractéristique crée une ambiguïté morphologique lors de l'analyse. Par conséquent, l'identification des unités lexicales qui composent les unités morphologiques n'est pas toujours facile à repérer. Le problème est de reconnaître la bonne segmentation D'où la difficulté de distinguer entre proclitique ou enclitique et caractères le texte original du mot. Par exemple, le caractère "و "dans le mot "il est arrivé | وصل"est un caractère original alors que dans le mot "et il a ouvert | وفتح", il s'agit plutôt d'une proclitique.

L'absence de voyellation pose un autre problème important. En effet, les mots non voyellés engendrent beaucoup de cas ambigus au cours de l'analyse (e.g. le mot non voyelle " فصل" pris hors contexte peut être un verbe au passé conjugué à la troisième personne du singulier "il a licencié |فَصَلَ " , ou un nom masculin singulier "chapitre/ saison | فَصلٌ",ou encore une concaténation de la conjonction de coordination "puis |فَـ " avec le verbe "صل ": impératif du verbe lier conjugué à la deuxième personne du singulier masculin).[9]

13.3. L'étiquetage grammatical :

L'étiquetage grammatical est une opération qui inclut l'attribution de chaque mot La catégorie du texte à laquelle il appartient (non, verbe, adjectif, article défini, etc.) Le contexte dans lequel il apparaît. La difficulté du balisage grammatical augmente lorsque le texte cible est présent Dans leur forme, pas des voyelles, mais des voyelles partielles ou même pleines Pas de voyelles, correspondant aux cas les plus courants. Dans ces conditions, le but général du balisage grammatical est de répondre à La question suivante : Comment associer aux différents mots qui composent le texte Des étiquettes appropriées pour eux compte tenu du contexte dans lequel ils se produisent ? Par conséquent, le problème lorsque le texte souhaité est dans sa non- ou des voyelles partielles au lieu de leurs formes vocaliques.

Le problème vocalique d'un mot se pose donc car le choix L'accentuation appropriée du mot est difficile et dépend largement du contexte. [9]

Le tableau 1.3 présente le problème d'ambiguïté grammaticale rencontrée lors de l'attribution catégorique d'un mot non voyelle "ktb | كتب ", qui admet au moins cinq étiquettes grammaticales qui sont les suivantes :

Exemple de voyellation	Etiquettes grammaticales

كُتُبٌ| kutubun :des livres	Substantif, masculin, pluriel
كَتبٌ| kutbun :un écrit	Substantif, masculin, singulier
كَتَبَ| kataba :il a écrit	Verbe, 3eme, personne masculin, singulier de l'accompli actif
كُتِبَ|kutiba :il a été écrit	Verbe, 3eme, personne masculin, singulier de l'accompli passif
كَتِبْ |kattib :fais écrire	Verbe à l'impératif, 2eme, personne masculin, singulier

Tableau 1.3 : Exemple d'étiquettes grammaticales attribuées selon la voyellation [9].

13.4. L'analyse syntaxique :

L'analyse syntaxique permet d'associer sa structure syntaxique à des phrases. Eventuellement, en identifiant ses différentes composantes et les rôles qu'elles jouent entre eux. Cependant, l'analyse syntaxique prend le résultat de l'analyse lexicale comme entrée (éventuellement un jeton morphosyntaxique) et produire une structure Hiérarchie des groupements structurels et des relations fonctionnelles groupé. Enfin, il convient de noter que l'ambiguïté phonologique et grammaticale, par opposition à Le mutisme des mots crée des difficultés au niveau de l'analyse syntaxique. Par conséquent, un Les phrases, en l'absence de voyelles, peuvent être interprétées et traduites selon plusieurs Les deux sont des interprétations grammaticalement correctes. [9]

14. Outils de traitement automatique de la langue arabe :

Il existe 3 principaux outils en arabe pour le traitement automatique de l'arabe, Les analyseur morphologiques, les concordances, les racineurs.

14.1. Analyseurs morphologiques :

Les analyseurs morphologiques segmentent les unités lexicales pour identifier différentes composants et prouver qu'ils appartiennent au langage. [12]

14.2. Les concordanciers :

Index écrit mis en œuvre à la main est un effort à grande échelle uniquement possible avec des œuvres pérennes. Le traitement automatisé facilite cette tâche et étend son champ d'application à de nombreuses disciplines scientifiques. Dans le cas de l'arabe, la mise en place du coordinateur électronique nécessite un travail préalable à l'aide de ressources lexicales et d'outils de tokenisation morphosyntaxiques. L'approche classique de construction d'indexeurs

basée sur la reconnaissance de formes d'éléments dans le texte est inefficace lorsqu'il s'agit de l'arabe, dont l'écriture est non vocale et dont la structure unitaire lexicale peut être décrite comme cohésive et fortement fléchie. Par conséquent, l'outil Arçon développé pour l'arabe est conçu pour fournir le contexte et la fréquence, et permettent l'exploration du corpus à partir des caractéristiques proposées par l'analyse morphologique et des informations graphiques trouvées dans le texte. L'index arabe final s'articule autour d'un trio : unité lexicale, analyse positionnelle et morphologique. L'outil prend du texte ou un ensemble de textes en entrée. Il permet de:

- Construire une liste de fréquences d'items, de radicaux ou de toute autre caractéristique d'analyse morphosyntaxe par ordre alphabétique ou de fréquence.
- Construction du vocabulaire, consultable par item, racine, cardinalité ou analyse morphosyntaxique. [12]

14.3. Racineurs :

Les racineurs se veulent d'abord un outil utile au TAL, ce type d'analyse «simpliste », traite de façon identique affixes flexionnels et dérivationnels. Les algorithmes de racinisation en arabe les plus connus sont ceux de Racineur de 'larkey' et Racineur de 'Khoja'. [12]

15. Conclusion :

Dans ce chapitre, nous avons présenté le domaine du TALN, ses objectifs, ses domaines d'application, ses outils ainsi que les principes de la langue arabe et les difficultés rencontrées lors de son traitement automatique. L'arabe se caractérise par la variabilité, l'articulation et l'ambiguïté, posant de nombreux défis à divers domaines tels que le traitement du langage naturel ou la recherche d'informations.

Dans le chapitre suivant, nous allons présenter la discipline du résumé automatique de texte qui constitue notre objectif derrière ce travail.

CHAPITRE 02

RESUME AUTOMATIQUE DU TEXTE

Chapitre II : Résumé automatique du texte

1. Introduction :

L'objectif d'un système de résumé automatique est de produire un document condensé à partir d'un ou plusieurs documents sources. Comme les informations textuelles disponibles en ligne augmentent rapidement, le résumé automatique a connu un fort essor ces dernières années. De ce fait, ce chapitre sera consacré à la présentation de ce domaine tout en mettant l'accent sur les différentes approches et techniques contribuées.

2. Définitions :

2.1. Résumé :

Est le texte qui est produit à partir d'un ou plusieurs textes, qui transmet des informations importantes dans le(s) texte(s) original(s), et qui ne dépasse pas la moitié du ou des textes originaux généralement. [13]

2.2. Le résume automatique de texte :

Est le processus de génération d'un résumé cohérent significatif en couvrant le plus Informations importantes autant que possible. Récemment, des méthodes et des techniques ont été proposées pour le texte automatique résumé et appliqué largement dans divers domaines. [14]

3. Les types des résumés automatiques :

Il existe plusieurs types de résumés selon leur longueur, leur style et leur subjectivité :

3.1. Résumé informatif :

Le résumé informatif fournit un ensemble informations de donne un gros panorama du contenu d'un texte. Plus cela, l'ensemble des principaux sujets doit être rapporté.

Par conséquent, le résumé des médias, il contient toutes les informations pertinentes en réalité, qui visent à tend à conserver l'organisation générale du texte source.

Ainsi, les sujets principaux qui sont rappelés dans le résumé sont répartis de manière fidèle par rapport à L'organisation initiale afin de donner un juste aperçu du texte source. [9].

3.2. Résumé indicatif :

Le rôle d'un résumé indicatif est de fournir au lecteur suffisamment informations afin qu'il puisse décider s'il doit consulter le fichier source.

Ce résumé indicatif implique le concept de thermalisation, c'est-à-dire qu'il ne Aucun commentaire n'est accepté pour les thèmes développés que dans la documentation source. Des résumés de ce type peuvent être rapprochés du catalogue.

A cette fin, le résumé indicatif ne contient qu'Un résumé informatif, mais surtout en réponse à sa fonctionnalité. Les résumés indicatifs sont souvent utilisés dans les collections bibliographiques parce qu'ils sont bons Bon pour les documents avec des descriptions plus longues : il fournit au lecteur un bref aperçu Que peut-il trouver dans le dossier.

Par conséquent, des résumés indicatifs sont également utiles Pour les articles courts car cela donne un aperçu immédiat du contenu. [9]

3.3. Résumé d'opinion :

La Synthèse d'Avis a pour objectif de présenter et d'identifier l'offre, jugements et opinions. Par conséquent, le but des résumés d'opinion n'est pas seulement de Générer une synthèse des informations contenues dans le texte, tout en identifiant Tendances, identifier les opinions exprimées. [9]

3.4. Résumé de conclusions :

Un résumé des conclusions est parfois appelé « résumé des résultats » et est également appelé « résumé des conclusions » Pour résumer Une conclusion ou un résultat est défini comme "un bref énoncé dans un document (généralement placé à la fin de ce document) destiné à compléter Les lecteurs qui ont lu l'article précédent".

Pour cette raison, ces résumés ne comprennent que Résultats et conclusions présentés dans le texte source. [9]

3.5. Résumé critique :

Ce résumé contient non seulement des informations pertinentes Mais il contient également les commentaires de l'abstracteur (autre que l'auteur du texte).

Cette « summarizer » évalue de manière critique la qualité et les principales affirmations Fichier d'origine. A cette fin, ce type de résumé combine un texte source condensé A apporté une contribution importante au contenu de cet article. [9]

3.6. Résumé synthétique :

Il consiste à emprunter certains termes au texte source et à les rendre une interprétation qui fait que le nouveau texte n'est pas un sous-ensemble de ce texte source. [9]

3.7. Résumé scolaire :

Les résumés académiques suivent des critères de construction spécifiques, tels que Baisses de taux standardisées, interdictions de prêt ou utilisation systématique synonyme.

En fait, ce type de résumé est conçu à des fins éducatives Évaluer et valider les compétences d'analyse et de compréhension des étudiants Texte et écrit. Par conséquent, le résumé scolaire doit être fidèle au texte original, Conservez le plan et la structure générale du sujet. [9]

4. Types d'utilisateurs des résumés :

Il y a plusieurs types de résumés selon leur but :

4.1. Le résumé mono-document :

C'est le résumé d'un seul document isolé.

4.2. Le résumé multi-document:

C'est le résumé d'un groupe de documents, pas forcément hétérogènes, portant souvent sur une thématique bien précise. Les Résumés Automatiques des Documents Textuels. [15]

4.3. Résumé automatique multi-documents :

Les systèmes de résumé multi-documents peuvent générer des résumés d'une collection Texte en donnant une description de son idée principale.

Cependant, certains D'autres aux résumés multi-documents. Par exemple, selon la méthode La programmation linéaire a eu plus de succès que les méthodes basées sur des graphes. [15] Les trois principaux problèmes du résumé multi-documents sont

- Identifiez les éléments saillants redondants.
- Identifier les différences entre les documents.
- Cohérence des résumés même si le contenu provient de documents sources différents.

5. Les différentes approches de résumé automatique :

Il existe deux approches principales pour générer des résumés de texte :

5.1. Abstraction :

Alors que nous nous **intéressons** principalement aux systèmes de synthèse extractive, Les systèmes abstraits partagent une certaine forme d'abstraction dynamique Modéliser le contenu du document, même si les critères d'extraction dans le document Les cas dynamiques sont généralement sémantiquement superficiels.

Méthode de la synthèse abstraite imite quelque peu le processus humain naturel de synthèse de documents. Par conséquent, ils génèrent Le résumé est plus similaire au résumé manuel. Ce processus peut être décrit En deux étapes principales : comprendre le texte source et générer le résumé. Les deux tâches sont assez complexes.

C'est pourquoi ils sont simplifiés. La première étape vise à analyser sémantiquement le contenu du texte et déterminer quelle partie du résumé doit être exprimée. Elle a parfois Prendre la forme d'une tâche d'extraction d'informations pertinentes pour le domaine concerné ou des regroupements de phrases du texte source. La génération de texte est un domaine en soi. Une approche simplifiée consiste à appliquer des techniques de génération de texte à texte : Utilisez la paraphrase ou combinez et compressez Phrases. Une autre façon est d'introduire un modèle de texte Domaine (modèle) et instanciation lors de la génération. [16]

5.2. Extraction :

Le point fort du résumé par extraction est qu'il évite la génération de texte. Ceci permet d'une part, de se concentrer sur la sélection du contenu pertinent et d'autre part, d'obtenir un résumé lisible et linguistiquement correct. La cohérence n'est en revanche pas garantie. Par exemple, si le système de résumé sélectionne des phrases contenant des références (acronyme, pronom personnel, etc.) et ne sélectionne pas les phrases contenant leurs antécédents, il est fort probable que le résumé produit soit incompréhensible. Pour pallier ce problème, certains travaux considèrent le paragraphe comme unité d'extraction au lieu de la phrase Ceci permet de garder la cohérence du texte source mais ne peut pas être applicable dans le cas de résumés courts. De plus, il est évident que cette méthode réduit la précision du résumé en y incluant des phrases peu importantes juste pour améliorer la cohérence. D'autres chercheurs procèdent à des étapes de pré/post-traitement du texte qui améliorent partiellement la cohérence globale du résumé. Le processus principal dans le résumé extractif est la sélection des segments de textes (généralement les phrases) pertinents et non redondants sans dépasser une taille limite du résumé. Ce principe limite la couverture des informations apportées par le texte source. Les résumés abstractifs souffrent moins de ce problème puisque l'information peut y être reformulée. [16]

6. Les méthodes de résumé automatique :

Dans cette section, nous présentons brièvement l'extraction de phrases clés, qui reposent essentiellement sur le calcul de scores de pertinence à chaque phrase pour estimer son

importance dans le texte. Le résumé final ne restera que les phrases avec les scores les plus élevés. [17]

La figure 2.1 regroupe les différentes méthodes de résume automatique de texte :

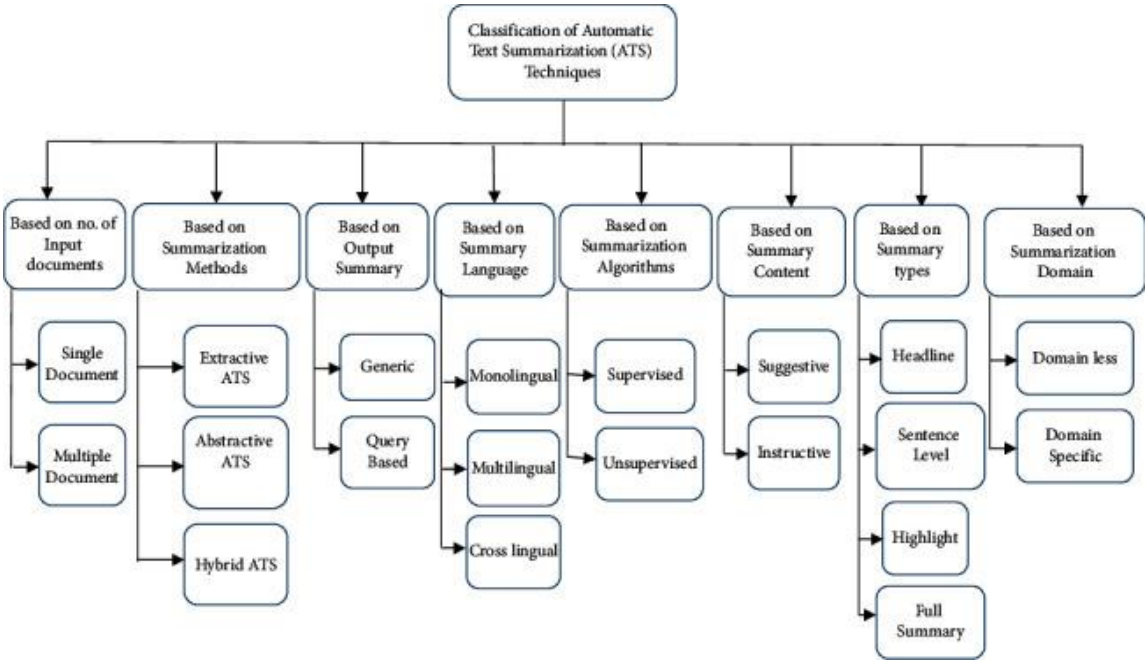

Figure 2.1 les différentes méthodes de résume automatique de texte [42]

6.1. Méthodes à base de mots clés :

Cette méthode est basée sur les faits utilisés par l'auteur (exprimant son idée principale) Quelques mots-clés faciles à répéter dans le texte [Pardo et aI, 2002].

Le résumé automatique est Il est ensuite généré en recherchant dans le texte source la plus petite unité de texte qui le rassemble Mots clés.

Ce principe est souvent appliqué dans les différentes variantes présentées dans les sous-sections suivantes.

- **Mots-clés prédéfinis :**

Pour calculer le score de chaque phrase S en fonction des mots clés qu'elle contient, on peut calculer les Scores ci-dessous :

Score $_{\text{mot-clé}}$**(S) =** $a(t)$ ***** $F(t)$

F(t) est la fréquence du terme t dans la phrase S

$$a(t) = \begin{cases} A \text{ si } t \in \text{liste de mots} - \text{clés} (A > 1) \\ 1 \qquad\qquad\qquad \text{sinon} \end{cases}$$

Les listes de mots clés peuvent être introduites ou composées par les utilisateurs (domaines d'intérêt) Mots clés définis par l'auteur. L'importance du poids du terme t est donnée par A×F(t), où A>1.

- **Titre :**

Etant donné que le titre est l'expression la plus significative et qui résume le mieux un document en quelques mots, on peut dire que la phrase qui ressemble le plus au titre est la plus marquante du document. Par conséquent, on peut attribuer à chaque phrase un poids en fonction de sa ressemblance avec le titre [Ishikawa et al. 2001].

Dans ce cas, nous traitons les mots des titres de texte comme des mots-clés et générons résumé en sélectionnant des phrases qui couvrent certains mots qui apparaissent dans le titre.

Score titre(S) = b (*t*) * F (*t*)

F(*t*) est la fréquence du terme *t* dans la phrase S

$$\boldsymbol{b(t)} = \begin{cases} \boldsymbol{A\ si\ t \in liste\ de\ mots\ du\ titre} (\boldsymbol{A} > 1) \\ \mathbf{1} \qquad\qquad \boldsymbol{sinon} \end{cases}$$

- **Distribution des termes :**

L'idée de cette approche est de considérer des phrases importantes qui contiennent mots importants dans le texte. Un mot est considéré comme important s'il est utilisé assez fréquemment dans le texte. Identique au principe de mesure TF-IDF [Nobata et Sekine, 2003].

$$Score_{if.idf}(S) = \frac{1}{|S|}\sum_{w \in s} if.idf(w)$$

$$tf.idf(w) = \frac{tf(w)-1}{tf(w)} log\frac{DN}{df(w)}$$

S= nombre de mots dans la phrase S.

if (w) est la fréquence du terme w dans le document.

df (w) est le nombre de documents du corpus où le terme w apparaît.

DN est le nombre de documents dans le corpus.

$$Score(S) = \frac{1}{|S|}\sum_{w \in s} Score(w)$$

$$Score(S) = F(w) \times \frac{\log(|S|)}{S(w)}$$

F(w) est la fréquence du terme w dans la phrase.

S(w) nombre de phrases dans lesquelles w apparaît.

6.2. Méthode à base de position :

Cette méthode suppose que la position d'une phrase dans le texte indique son importance Dans le contexte. Par exemple, la première et la dernière phrase d'un paragraphe peuvent être Transmettre l'idée principale et devrait donc faire partie du résumé. En variante, cette méthode

peut être appelée méthode Lead ; c'est une méthode de détermination d'une phrase En extrayant les plus importants qui sont en tête.

Cette méthode peut efficacement résumer des articles de journaux, car les phrases importantes ont tendance à apparaître dans La première phrase de l'article [Ishikawa et al, 2001].

Nous définissons le score de la phrase S à la position i comme suit :

Score lead (Si) = βi

$$\boldsymbol{\beta i} = \begin{cases} \boldsymbol{B} > 0 \; si \; i > 0 \\ \boldsymbol{0} \; \boldsymbol{si} \quad \boldsymbol{i} \geq \boldsymbol{N} \end{cases}$$

βi Est une fonction rectangulaire qui modélise la distribution des phrases importantes en fonction de leur distribution positionnement dans l'article.

Si les dernières phrases sont importantes, introduisez une La nouvelle plage de valeurs i. L'inconvénient de cette approche est qu'elle repose sur La nature du texte à résumer et le style de l'auteur.

6.3. Méthode dépendant de la longueur de phrase :

La méthode attribue des poids aux phrases en fonction du nombre de mots dans la phrase les scores peuvent être calculés à l'aide de deux techniques : [Nobata et Sekine, 2003]

- La longueur (Li) de chaque phrase est relative à la longueur maximale de la phrase.

Score long (Si) = $\boldsymbol{L_i/L_{max}}$

- Attribuez zéro point aux phrases plus courtes qu'une certaine longueur (minimum L). [17]

$$\textbf{Score long(S)} = \begin{cases} \boldsymbol{0 \quad si \;\; L_i \; si \; L_i \leq L_{min}} \\ \boldsymbol{\frac{L_i - L_{min}}{L_{min}} \; si \; L_i > L_{min}} \end{cases}$$

6.4. Méthode à base d'expressions indicatives (cue methods) :

Cette méthode sélectionne des cellules de texte avec une indication ou une expression spécifique. Par exemple, pour un texte scientifique, nous avons son but comme l'expression Le travail..., ce papier présente..., les résultats et les conclusions sont de bons candidats Indiquez les phrases à inclure dans le résumé. Différents types de texte peuvent avoir différentes expressions indicatives. On peut dériver un score pour une phrase d'un texte Tout ce qui est analysé en fonction de la similitude qu'il présente pour une caractéristique donnée.

On peut définir le score d'une phrase S correspondant à un motif comme :

$$\beta i = \begin{cases} \textbf{1 si S correspond à un motif} \\ \textbf{0 sinon} \end{cases}$$

6.5. Méthode basée sur les relations (cohésion lexicale) :

L'utilisation de la fréquence des mots est un excellent moyen de mettre en évidence les termes clés dans un texte, mais il ne tient pas compte de la relation entre les différents mots partie du texte. L'extraction de phrases basée sur la fréquence des mots entraîne généralement Manque de cohésion. Pour pallier ce problème, ceux déployés sur le terrain ont développé Une méthode basée sur la cohésion grammaticale (i.e. référence, substitution, conjonctions) et la cohésion lexicale (c'est-à-dire les mots sémantiquement liés). Cette méthode montre que plus une phrase n'a de liens avec une autre dans le texte, plus Approprié dans ce cas, c'est-à-dire qu'il exprime le même sujet. Par conséquent, la phrase Les liens doivent être sélectionnés ensemble pour former un résumé. Manque certains Des phrases très liées peuvent produire un texte incohérent. Identifier un telle La pertinence repose généralement sur un thésaurus ou un dictionnaire informatisé, ce qui permet Identifier les relations entre les mots. Nous construisons des chaînes lexicales à partir de mots Candidats pour le texte, ces chaînes regroupent des mots qui sont passés de thésaurus. Extrayez les phrases les plus pertinentes pour la chaîne lexicale [Chali et Pinchak, 2001] [Pardo et al, 2002].

6.6. La méthode d'exploration contextuelle :

Les méthodes d'exploration contextuelle visent à identifier les connaissances linguistiques dans les textes, en les replaçant dans leur contexte et en les organisant dans des tâches spécialisées. La méthode est basée sur la construction manuelle de base de données de balises de langue et expression de règles d'exploration contextuelle.

Ces règles appliquées aux phrases du texte source seront utilisez des étiquettes sémantiques hiérarchiques pour filtrer les informations sémantiques indépendantes du domaine, telles que :

Phrases structurées, définitions, causalité, etc. Stratégies de sélection d'unités proéminentes en fonction des besoins des utilisateurs [Farzindar, 2003].

L'analyse de l'exploration contextuelle repose sur la connaissance. A cet effet, quatre niveaux de connaissance sont distingués [Bern, 1996]:

- Connaissance de la langue (grammaire et vocabulaire), indépendante de connaissance du monde extérieur ;
- Connaissances spécifiques au domaine : cela implique Expertise liée au domaine de compétence et aux règles d'organisation de ce domaine ;

- Connaissances socioculturelles en fonction du milieu social, usages, coutumes, etc.
- Connaissances encyclopédiques universelles et communes à tous personnes.

La politique de décision pour l'exploration du contexte s'exprime comme suit :
Déterminer d'abord les règles heuristiques des indicateurs pertinents, Caractéristiques du problème à résoudre. Puis recherchez le contexte linguistique. Rechercher des indices linguistiques pour prendre les décisions appropriées.

L'avantage de cette technique est : L'indépendance entre les connaissances, Compétences linguistiques et connaissances accumulées nécessaires au système zone spécifique. Il permet une évolutivité incrémentale, complétant Listes qui ont été construites (chercher des indices plus fins) et affiner les règles en Explorer (Désolés). Ainsi, le système d'exploration contextuelle est plus ou moins efficace selon la richesse des indices considérés et la compétence du calcul exploré.

6.7. Méthode hybride :

Les méthodes décrites dans les sections précédentes utilisent des fonctionnalités (fréquence, emplacement, expressions indicatives, etc.), les résultats seuls ne peuvent être garantis Meilleurs.

Ces caractéristiques sont souvent combinées, par exemple, L'équation suivante :

$$\boldsymbol{Score_{hybride}(S) = a_1 * Score_{if*idf}(S) + a_2 * Score_{lead}(S) + a_3 * Score_{cue}(S) + a_4 * Score_{titre}(S)}$$

Le poids $\boldsymbol{a_i}$ peut-être fixé arbitrairement ou déterminé de cette façon Expérimental (par exemple par apprentissage).

Certaines expériences de [Edmundson, 1969] sur un corpus hétérogène de 200 documents ont montré que si on combine les méthodes cue, titre et position (poids zéro pour la méthode mots-clés), On obtient de meilleurs résultats que si on les combine avec la méthode mot-clés.

Dans le cas de textes journalistiques ont combiné les méthodes de distribution de termes, du titre, de la position et de la cue, en considérant la spécificité du texte [Strzalkowski et al.1998]. Ils ont fait ressortir que les phrases qui commencent par des nominaux ou contiennent des mots du titre semblent être plus pertinentes que des phrases n'ayant pas ce caractère. De plus, les mots ou les phrases n'apparaissant que dans quelques paragraphes sont plus importants que ceux mentionnés dans tous les paragraphes. Pour garder la cohérence du texte, le résumé est composé d'une sélection de paragraphes pertinents.

7. Conclusion

L’objectif de ce chapitre était de présenter l’état de l'art concernant le résumé automatique du texte. Ce domaine connait un nombre considérable de contributions et de techniques vu son importance aux utilisateurs du Net et aux traitement intelligent de l’information.

Dans ce qui suit, nous allons présenter la démarche de notre système proposé avec la méthode, les algorithmes et l’environnement technique adopté dans cette phase de réalisation.

CHAPITRE III

CONCEPTION ET IMPLEMENTATION DU SYSTEME

Chapitre III : Conception et implémentation du système

1. Introduction :

Dans ce chapitre nous allons présenter une étude évaluative sur les méthodes extractives puis nous aborderons les détails techniques liés à notre système (ESA text Extractive Summarization of Arabic Text). Par la suite, nous allons présenter notre approche et l'environnement technique de développement avec quelques captures d'écran comme démonstration d'exécution de notre système proposé.

2. Etude évaluative sur les méthodes extractives :

Le résumé extractif, au niveau le plus élémentaire, peut être abordé en utilisant la technique de notation de phrase qui obtient le mot-clé du texte [26]. Cela se fait en analysant et en filtrant les mots les plus fréquemment utilisés dans le texte. Les phrases avec une fréquence élevée de ces mots sont utilisées pour générer un résumé du texte original en utilisant les phrases avec des scores élevés dans l'ordre décroissant des scores [27]. Pour de meilleures performances et une meilleure efficacité, des méthodes basées sur des graphes ont été introduites, rendant les modèles capables de considérer des attributs plus complexes de l'information textuelle et de présenter des informations concises avec une meilleure précision.

Dans les approches basées sur les graphes, les mots sont considérés comme des nœuds et leur relation avec d'autres mots est basée sur leur fréquence, qui est représentée par des arêtes. Les arêtes sont pondérées et sont analysées pour choisir les mots de la requête pour générer un résumé [28]. Plusieurs algorithmes comme PageRank , TextRank , TexRank , etc., peuvent être utilisés pour des techniques efficaces de résumé de texte [29]. Un graphique bipartite est créé pour représenter les phrases et les sujets séparément. Des scores sont attribués à chaque phrase, et les phrases en scores décroissants sont ajoutées au résumé. Plusieurs techniques telles que la distance de (Levenshtein), la similarité sémantique et la similarité cosinus sont utilisées pour déterminer la relation entre les phrases et les mots, ce qui ouvre la voie à une génération de résumé efficace [30].

2.1 L'algorithme TF-IDF :

Dans l'algorithme TF-IDF, les textes volumineux sont convertis en phrases, puis la fréquence des termes pondérés, et la fréquence des phrases inverses est calculée où la fréquence des phrases est définie comme le nombre de phrases du document, qui impliquent ces termes [31]. Les vecteurs des phrases sont calculés et comparés aux autres phrases et sont ensuite notés. Le

produit de TF et IDF calcule la valeur TF-IDF d'un mot/terme, où TF (fréquence du terme) est défini comme le nombre de fois qu'un mot apparaît dans un document et IDF est la fréquence inverse du document [31]. Les phrases avec le score le plus élevé sont considérées comme les phrases concluantes pour le résumé [32].

L'estimation TF-IDF de tout et le mot d'action seraient alors déterminés à partir du récapitulatif prétraité des mots. Les calculs de TF-IDF peuvent être effectués à l'aide de l'équation (3).

Calcule de TF (fréquence de terme) :

$$tf_w = \frac{\textit{Nombre de terme w dans le document}}{\textit{Nombre totale des termes de document}} \qquad (1)$$

Calcule d' idf (fréquence inverse de document) :

$$Idf_i = log \quad \frac{|D|}{|(d_j : t_i \in \boldsymbol{d_j})|} \qquad (2)$$

Où :

- ✓ |D| : nombre total de document dans le corpus.
- ✓ $|(d_j : t_i \in \boldsymbol{d_j})|$ **:** nombre de document ou le terme t_i apparait.

Calcule de tf-idf :

Tf- idf = tf * idf (3)

2.2 TextRank Algorithme :

TextRank est utilisé pour le prétraitement du texte afin de déterminer les mots-clés et les phrases pertinentes dans un texte donné. Il s'agit d'un modèle de classement basé sur des graphiques non supervisé. Ensuite, ces phrases sont utilisées pour générer le résumé du texte. Étant donné que l'algorithme TextRank est basé sur un graphe, la signification d'un sommet est déterminée en fonction des informations complètes fournies par le graphe. L'algorithme TextRank prend cette décision en fonction des "votes" ou des "recommandations" d'un sommet. Tous les sommets sauf celui pris en compte voteront pour un sommet [33]. L'importance ou la valeur d'un sommet est calculée en fonction des votes reçus par le sommet. De plus, le vote de chaque sommet a son importance calculée en considérant la valeur du sommet qui vote. Une fois que tous les sommets sont notés ou évalués, les sommets avec des scores maximum sont ensuite choisis comme mots-clés importants. Ces mots clés sont utilisés pour déterminer le contexte clé du texte et des phrases, qui doivent être ajoutés au résumé généré.

Formellement, supposons qu'un graphe orienté avec l'ensemble de sommets V et l'ensemble d'arêtes E soit représenté par $G = (V , E)$, où E est un sous-ensemble de VxV . Soit ln (V_i) l'ensemble des sommets d'un sommet donné V_i qui pointent vers lui, et soit Out(V_i) l'ensemble des sommets vers lesquels pointe le sommet V_i . Le score d'un sommet est défini à l'aide de l'équation (7) [33].

$$S(V_i) = (1 - d) + d * \sum_{j \in In(Vi)} S(V_j) / |Out(V_j)| \quad (7)$$

Ici, d est le facteur d'amortissement dont la valeur est comprise entre 0 et 1. Il intègre la probabilité de sauter d'un sommet donné à un autre sommet dans le graphe.

2.3 Algorithme BertSum :

BertSum est un algorithme de résumé abstrait basé sur BERT (Représentations d'encodeurs bidirectionnels de transformateurs), une architecture d'apprentissage non supervisée construite au-dessus de l'architecture Transformer. L'architecture BERT a réussi à fonctionner plus efficacement pour un large éventail de tâches que les modèles existants dans le domaine NLP [34].

2.4 Algorithme LexRank :

LexRank est une technique extractive utilisée pour le synopsis de texte. Méthode LexRank pour le résumé de texte où une autre méthode bébé utilisée est la méthode PageRank avec un frère TextRank . Cette technique d'apprentissage est basée sur le graphe non supervisé. La notation des phrases est terminée en utilisant la stratégie du diagramme. LexRank est utilisé pour déterminer la signification des phrases en fonction de l'idée de la centralité des vecteurs propres dans une représentation graphique des phrases. Selon cet algorithme, si une phrase est similaire à de nombreuses autres phrases, on suppose qu'elle est plus importante dans le document.

Ce modèle a un cadre de réseau dépendant de la ressemblance du cosinus intraphrase, qui est utilisé comme grille continue de la représentation schématique des phrases [35]. Cette extraction de phrase tourne de manière significative autour de l'arrangement des phrases avec le même plan. Par exemple, une phrase centroïde est choisie, remplissant la moyenne de toutes les phrases restantes dans l'enregistrement. Plus tard, les phrases sont disposées selon leurs similitudes.

2.5 Algorithme PEGASUS :

‘’Pretraining with Extracted Gap sentences for Abstractive Summarization’’ (PEGASUS) est un algorithme de résumé abstrait qui utilise le cadre séquence à séquence, qui utilise des

RNN, basés sur des architectures d'encodeur-décodeur. Il utilise des modèles séquence-à-séquence pré-entraînés avec des phrases masquées puis transmises à l'encodeur-décodeur.

3. Métriques d'évaluation :

En général, il existe trois types d'évaluations : l'évaluation basée sur la Co-sélection (avec un résumé de référence), l'évaluation basée sur des documents (avec le document original) et l'évaluation basée sur le contenu (sans résumé de référence) [36]. Nous brièvement discuter eux comme suit.

a. *basées sur la cosélection* . Cette technique d'évaluation est basée sur des mots-clés dans le résumé du système, et elle nécessite une comparaison des résumés de référence des documents. Les mots communs du résumé de référence et du résumé système sont choisis et évalués séparément. Le rappel , le score F et la précision sont les mesures

b. *Métriques d'évaluation basées sur le contenu.* Cette technique évalue le système de synthèse en termes largement compris. Le plan ne peut pas obtenir un réseau de pensées, un flux de phrases, la relation des phrases avec des phrases précédentes ou la curiosité du contenu. Chacune de ces difficultés peut être résolue en utilisant une approche basée sur le contenu. Nous montrons quelques méthodologies d'évaluation basées sur le contenu qui tiennent compte des caractéristiques variées d'un texte. Cela nécessite simplement une vue d'ensemble du système, qui contient des mesures telles que la cohésion, la non- redondance et la lisibilité

c. *Métriques d'évaluation basées sur des documents.* Lorsque deux phrases d'un document ont la même pertinence, mais qu'aucune n'est incluse dans le résumé de référence, ces mesures d'évaluation ne permettent pas d'évaluer correctement le résumé du système

3.1 La métrique ROUGE :

Depuis le milieu des années 2000, la métrique ROUGE a été largement utilisée pour l'évaluation programmée des contours [37]. Il est appelé Recall-Oriented Understudy for Gisting Evaluation (ROUGE), et il a présenté diverses mesures qui aident à décider naturellement de la nature d'un plan en le comparant avec des synopsis humains (de référence) considérés principalement comme la vérité fondamentale.

Différents types de ROUGE sont utilisés pour comparer différentes phrases. La granularité des textes comparés entre les résumés système et les résumés de référence peut être considérée comme ROUGE-L, ROUGE-N et ROUGE-S.

a. *ROUGE-N* identifie le chevauchement entre les unigrammes, les bigrammes, les trigrammes et les n-grammes d'ordre supérieur

b. *ROUGE-L* utilise la phrase commune la plus longue (LCS) pour déterminer la séquence de termes correspondante la plus étendue. LCS a l'avantage d'exiger des correspondances en séquence qui capturent l'ordre des mots au niveau de la phrase plutôt que des correspondances séquentielles. Vous n'avez pas besoin de spécifier une longueur de n-grammes car elle contient les n-grammes typiques les plus longs en séquence *par* défaut

c. *ROUGE-S* est n'importe quelle paire de mots dans le bon ordre d'une phrase, en tenant compte des lacunes. C'est ce qu'on appelle l'accord de saut de gramme. Skip-bigram, par exemple, teste le chevauchement entre des paires de mots avec une limite de deux espaces entre eux. Par exemple, les sauts-bigrammes pour le terme "chien dans le panier" seront "chien dedans, chien le panier du chien, dans le, dans un panier, le panier" ROUGE-1 fait référence au chevauchement des unigrammes entre la description de l'appareil et le résumé de référence concernant cette étude. ROUGE-2 fait référence au chevauchement des bigrammes entre la méthode et les résumés de comparaison. Généralement, il y a trois métriques [38] que ROUGE génère pour analyser les résultats.

- **(i) *Rappel***. Le rappel est un aspect de la métrique ROUGE qui peut être considéré comme la quantité de données originales fournies au modèle qui a été utilisée pour générer le résumé.
- **(ii) F-*Score***. Le F-score est une valeur numérique dérivée en utilisant la précision et le rappel. Il est utilisé pour exprimer la bonne combinaison de rappel et de précision.

*F-score= 2 * rappel * précision/(rappel+précision*)

(8)

- **(iii) *Précision***. La précision fait référence à la quantité mesurable de résumé généré qui était essentiellement nécessaire ou requise pour générer un résumé efficace.

L'ensemble de données et les sorties de notre algorithme sont fournies dans la fonction ROUGE, qui est utilisée pour évaluer la similarité de deux phrases en comptant le nombre de mots qui se chevauchent, puis en générant un résultat sous la forme de trois mesures appelées rappel, f-score et précision.

3.2 Ensemble de données MultiNews :

Cet ensemble de données contient un résumé généré par l'homme des différents articles de presse cités sur https://newser.com [39]. Des éditeurs professionnels ont rédigé ces résumés et incluent des liens vers les articles originaux cités.

Pour cet ensemble de données, le résumé moyen généré par tous les exemples contient en moyenne trois phrases dans le résumé résultant. Par conséquent, pour une meilleure génération de résultats, nous avons conservé un résumé en trois phrases comme référence. Les résultats générés après avoir utilisé cet ensemble de données comme fournisseur de synthèse de référence sont présentés dans le tableau 3.1.

N	Algorithm	Rog-1-f	Rog-1-p	Rog-1-r	Rog-2-f	Rog-2-p	Rog-2-r	Rog-lf	Rog-lp	Rog-lr
1	TF-IDF	0,2971	0,35273	0,25663	0,0821	0,0987	0,0703	0,2495	0,2849	0,222
2	LexRank	0,2941	0,4203	0,22619	0,0765	0,1077	0,0593	0,2307	0,3306	0,1772
3	BertSum	0,2584	0,42442	0,18581	0,0745	0,1325	0,0519	0,2268	0,3501	0,1678
4	TextRank	0,5948	0,60544	0,58456	0,1112	0,0736	0,2276	0,2828	0,2041	0,4605
5	PÉGASE	0,438	0,49796	0,39095	0,1998	0,2261	0,179	0,3734	0,4296	0,3303

Tableau 3.1 ROUGE pour l'ensemble de données MultiNews . [40]

Nous pouvons voir dans le tableau 3.1 que sur l'ensemble de données MultiNews , TextRank donne le meilleur résultat de tous les algorithmes sur les métriques ROUGE-1, et PEGASUS offre les meilleures performances pour les métriques ROUGE-2 et ROUGE-L de tous les algorithmes comparés. Si nous comparons la moyenne globale du score F, alors PEGASUS a le meilleur score F pour tous, et TextRank a le deuxième meilleur score F moyen et le meilleur parmi les algorithmes basés sur l'extraction.

Voici une représentation visuelle des données recueillies ci-dessus, qui analysera les performances des différents algorithmes de la (figure 3.1).

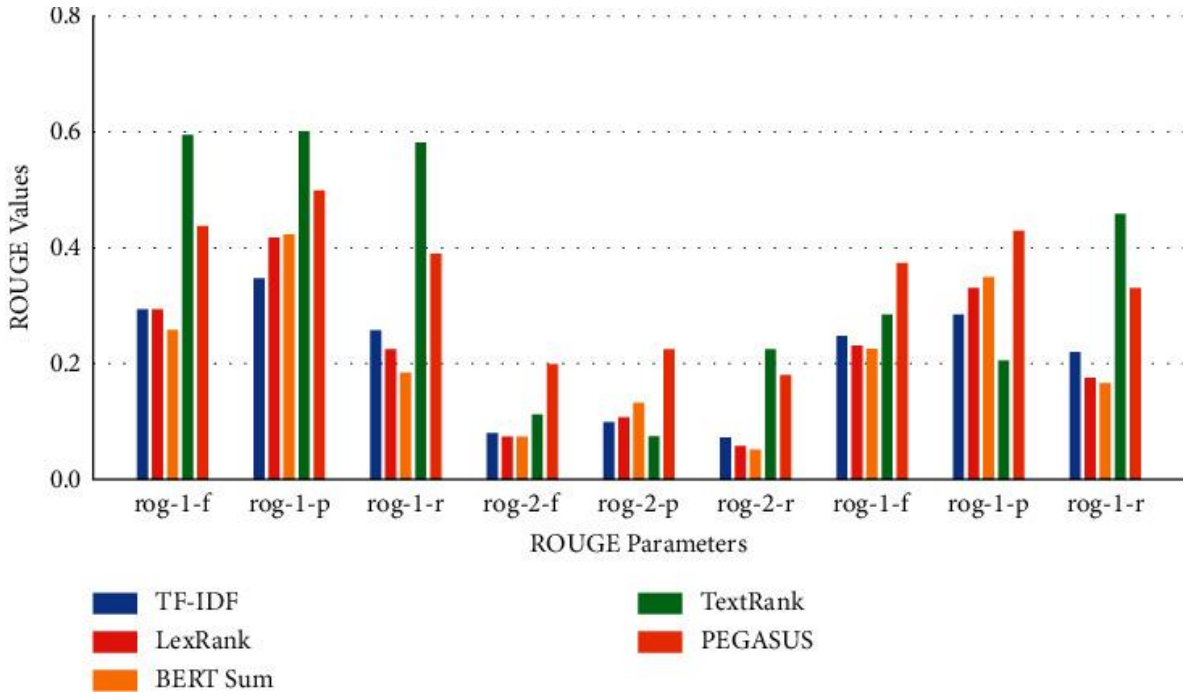

Figure 3.1 performances des différents algorithmes. [40]

3.3 Ensemble de données Reddit –TIFU :

Cet ensemble de données contient les échantillons de l'ensemble de données Reddit, et TIFU représente ici le nom du sous- reddit. Il contient également des résumés manuscrits des échantillons présents dans l'ensemble de données, qui sont utilisés à titre de référence. Pour cet ensemble de données, le résumé moyen généré pour chaque échantillon était d'une longueur de 3 phrases. Par conséquent, lors de la récupération des résultats, nous avons utilisé des résumés de trois phrases comme résumé généré à partir de nos algorithmes. Les résultats ont ensuite été comparés à l'aide de la bibliothèque ROUGE implémentée en Python et sont présentés dans le tableau 3.2 pour les cinq algorithmes. [40]

Non.	Algorithm	Rog1f	Rog1p	Rog1r	Rog2f	Rog2p	Rog2r	Rog-lf	Rog-lp	Rog-lr
1	TF-IDF	0,2095	0,1819	0,4208	0,1251	0,1578	0,1839	0,1282	0,1835	0,3525
2	LexRank	0,2199	0,1183	0,3312	0,1275	0,2034	0,1806	0,1442	0,1709	0,2713
3	BertSum	0,2261	0,1165	0,3887	0,1209	0,1263	0,1832	0,1356	0,1905	0,3362
4	TextRank	0,2159	0,1098	0,5056	0,1258	0,1555	0,2215	0,1258	0,1784	0,4279
5	PÉGASE	0,2376	0,2139	0,3293	0,1845	0,2766	0,2023	0,2175	0,1974	0,3846

Tableau 3.2 ROUGE pour l'ensemble de données Reddit –TIFU [40]

Il ressort du tableau 3.2 que, pour l'ensemble de données Reddit , l'algorithme TextRank donne les meilleurs résultats possibles sur les quatre algorithmes basés sur l'extraction, qui a la moyenne la plus élevée de F-score et PEGASUS les surpasse tous, comme le montre le graphique. Ci-dessous aussi, l'un ou l'autre de ces algorithmes peut être utilisé pour générer des résumés de textes longs, qui sont similaires aux échantillons de l'ensemble de données Reddit -TIFU.

Voici une représentation visuelle des données recueillies ci-dessus, qui analysera les performances des différents algorithmes de la figure 3.2.

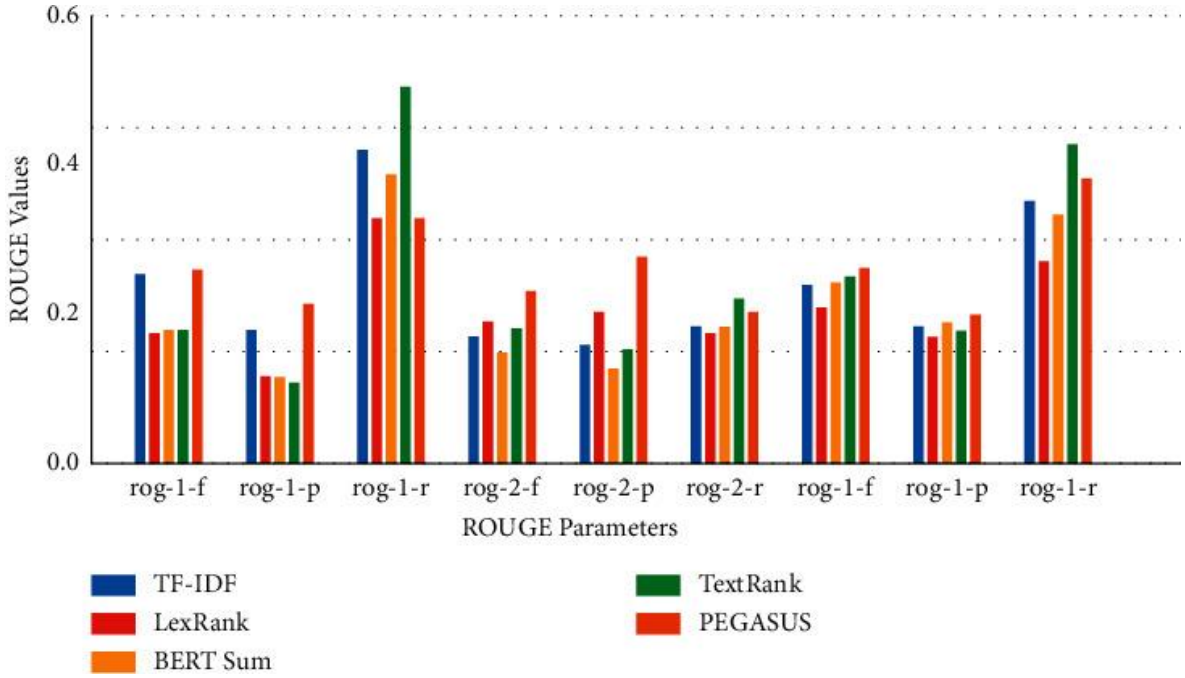

Figure 3.2 performances des différents algorithmes [40]

3.4 Comparaison des résultats :

Des algorithmes de synthèse sur le jeu de données REDDIT-TIFU :

Les valeurs ROUGE générées sont présentées dans le tableau 3.3

Algorithme	***F* –mesure**	**Rappel**	**Précision**
TextRank	0,133	0,085	0,382
LexRank	0,19	0,148	0,331

Tableau 3.3 Précision, rappel et *F* -mesure des algorithmes [40]

3.5 Conclusion d'évaluation :

Il est visible que ces deux algorithmes TextRank et LexRank donnent de meilleurs résultats sur l'ensemble de données Reddit -TIFU et MultiNews par rapport au résultat généré par l'ensemble de données. TextRank a obtenu de meilleurs résultats que les autres algorithmes de synthèse extractive pour diverses raisons. L'algorithme TextRank suit un apprentissage non supervisé car il n'y a aucune exigence d'ensemble de données de formation et aucune entrée générée par l'homme, ce qui permet à l'algorithme de fournir de meilleurs résultats par rapport aux autres algorithmes. L'algorithme TextRank est conçu de telle manière qu'en raison de son implémentation interne de l'algorithme PageRank et de la génération de la matrice de similarité, ses performances sont meilleures que celles de LexRank et de l'algorithme BERT.

4. L'Approche proposée :

L'approche que nous proposons est une approche extractive, hybride et non-supervisée pour l'identification automatique des mots clés dans des articles sources. Notre approche est qualifiée hybride par rapport à la combinaison de plusieurs critères statistiques et d'autres linguistiques pour la détection des mots clés, les critères statistiques sont la distribution des occurrences des termes (fréquences) et la position des termes dans les paragraphes du texte. L'indice linguistique est la catégorie grammaticale des termes.

L'objectif du résume extractif est de sélectionner les phrases les plus pertinentes du texte, un algorithme de clustering est appliqué sur une matrice de similarité pour regrouper l'ensemble des phrases similaires dans des clusters. Puis, on applique des paramètres de sélection pour déterminer la ou les phrases les plus pertinentes dans chaque cluster. Enfin, construction de résumé en utilisant les phrases extraites.

Donc, le système de résumé comprend les étapes suivantes :

- Prétraitement.
- Extraction de termes de fonctionnalité (mots des, mots de titre, termes fréquents).
- Sur l'ensemble des phrases du texte appliquer l'algorithme de classification
- Réorganiser les phrases pour avoir le résumé.

5. Architecture de système :

L'architecture globale de notre système est représentée par la figure 3.3 suivante :

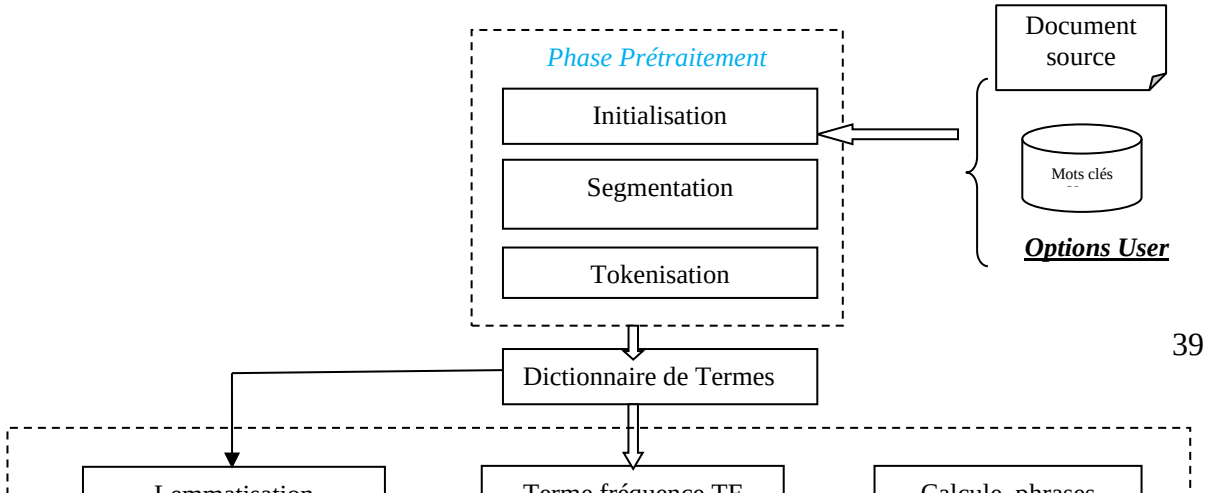

Figure 3.3 architecture du Système

5.1 Prétraitements :

5.1.1 Initialisation :

Chargement de texte source, nous avons proposé à l'utilisateur De choisir d'introduire un fichier au format texte, ou une URL d'un article sur le WEB. En plus, de choisir une méthode de résumé, éventuellement d'introduire des mots clés, la longueur de résumé, et prendre ou non en considération le titre d'article.

Le système fonctionne avec des paramètres par défaut qui sont les mots clés cherchés dans le texte avec la méthode de fréquence, le titre en considération, et longueur de résumé est de trois ligne.

5.1.2 La normalisation :

Afin de manipuler les variations du texte qui peuvent être représentées en arabe, on applique plusieurs genres de normalisation sur le texte. Par exemple, dans l'arabe écrit, les voyelles sont souvent omises dans les textes, néanmoins, on peut parfois trouver quelques voyelles présentes avec les mots. Alors, l'élimination de ces voyelles est nécessaire pour fin de normalisation. Certaines lettres subissent une simple modification dans l'écriture qui n'influe

pas considérablement sur le sens du mot. Mais l'encodage de ces lettres change d'un mot à un autre. Une autre raison pour ce prétraitement est que l'on a tendance fréquemment à mal écrire ces différentes formes de ‘hamza’. Ce genre d'erreurs est très répandu dans les textes arabes. Par exemple, le mot «أكل» est généralement écrit « اكل ». Aussi la lettre « ة » à la fin des mots qui peut être écrite de deux façons : « ة » ou « ه ». Les deux mots arabes « عادة » et « عاده » signifient le même mot (habitude) malgré que leur dernière lettre soit représentée différemment.

La normalisation concerne les étapes suivantes :

- Enlever la ponctuation.
- Retirer les signes diacritiques (principalement voyelles faibles).
- Retirer les non-lettres arabes.
- Remplacer le إ ou le أ initial par l'alif nu ا.
- Remplacer le آ par le ا.
- Remplacer le ىء d'ordre par le ى
- Remplacer le ي final par le ى
- Remplacer le ة final par le ه
- La liste des signes de ponctuation, des signes diacritiques, et des non-lettres

5.1.3 Encodage uniques des textes :

L’encodage unique des textes en format standard, permet de représenter les textes sans aucune déformation au niveau de caractère lors de lecture. Tous les textes de notre corpus sont représentés avec un encodage UTF-8.

5.1.4 La segmentation :

La segmentation est une étape nécessaire et signifiante dans tout traitement de la langue naturelle. La fonction d'un segmenteur est de couper un texte courant en segments, de sorte qu'ils puissent être introduits dans un capteur morphologique ou dans un étiqueteur de position pour un traitement ultérieur. Dans notre approche nous avons opté à une segmentation en paragraphe. Le paragraphe étant le segment qui garantit la couverture d’une unité thématique et un sens complet. Une frontière d’un paragraphe est facilement détectée par un point et/ou un saut de ligne.

5.1.5 Tokenization :

Consistant à diviser les phrases en mots en identifiant les espaces, les virgules et les symboles spéciaux entre les mots. La liste des phrases et des mots est maintenue pour un traitement ultérieur.

5.2 Phase traitement :

Dans cette phase on fait les opérations nécessaires pour trouver les phrases pertinentes.

5.2.1 La lemmatisation (Stemming) :

un mot arabe est formé généralement par une séquence de {antéfixe, préfixe, noyau, suffixe, postfixe}. Ainsi un mot arabe peut avoir une forme plus compliquée si tous ces affixes sont attachés à sa forme standard. De plus la lemmatisation des mots nous donne une valeur exacte de la distribution des mêmes termes dans le texte. [41]

5.2.2 Le filtrage (élimination de stopwords) :

Le filtrage consiste à prendre chaque segment de l'étape précédente (des paragraphes) et éliminer tous les mots non significatifs. Pour chaque mot reconnu, on le compare avec un des éléments dans la base de données qui contient tous les mots non-significatifs. Si un mot en fait partie, il ne sera pas pris en considération pour le calcul de sa fréquence. La base de données regroupe tous les particules et/ou les mots vides (stopwords).

5.2.3 Extraction :

Une fois qu'un document de saisie est divisé en une collection de phrases ces dernières sont classées en fonction de quatre caractéristiques importantes : la fréquence, la valeur de la position des phrases, les mots clés et la similarité avec le titre. Ces caractéristiques sont les critères de sélection des phrases pertinentes. [41]

- **La fréquence :**

La fréquence est le nombre de fois qu'un mot se produit dans un document. Si la fréquence d'un mot dans un document est élevée, on peut dire que ce mot a un effet significatif sur le contenu du document. La valeur de fréquence totale d'une phrase est calculée en résumant la fréquence de chaque mot dans le document. Pour réaliser cette étape on a employé l'algorithme TF-IDF qui est le plus connu et le plus utilise pour extraire les termes fréquents.

- **Position de phrase :**

Position de la phrase dans le texte, décide de son importance. Les phrases au début définissent le thème du document tandis que les phrases finales concluent ou résument le document. La valeur de position d'une phrase est calculée en attribuant la valeur la plus élevée à la première phrase et la valeur la plus basse jusqu'a la dernière phrase du document.

- **Les articulateurs :**

Les articulateurs sont des expressions conjonctives (par conséquent **لذلك** , enfin **اخيرا** etc..) qui relie les limites de la communication et signale les relations sémantiques dans un texte. Il faut les déterminer pour décomposer les phrases complexes.

- **La similitude avec le titre :**

La similitude avec le titre comprend les mots dans les titres et les en-têtes (en cas d'un sous sommaire comme le cas des articles long, exemple les articles du site Wikipedia). Ces mots sont considérés comme ayant des poids supplémentaires dans la notation de la phrase pour la synthèse. Au plus les mots du titre sont des termes fréquents par défaut.

- **Le score des phrases :**

On a ajoute ce critère pour favorise les phrases ayant le plus nombre d'apparition des termes fréquents. Le score final d'une phrase est une combinaison linéaire de fréquence, la valeur de position de la phrase, les poids des mots clés et Similarité avec le titre du document.

5.2.4 La Clustering :

C’est la regroupement des phrases les plus similaires dans des groupe appelés Cluster, ils existent plusieurs algorithmes de clusternig (K-mean, Algorithme Aprior, Clustering hiérarchique, Analyse en composantes principales (ACP), Décomposition en valeurs singulières (SVD) etc..) , plusieurs parmi les sont implémentés dans les bibliothèques de plusieurs plateformes de développement, donc le choix de module impose l’algorithme de clustering, dans notre système on a utilisé le K-means qui est plus populaire, simple et efficace, en plus il est non supervisé.

Pour qu’on peut regrouper les phrases les plus similaires on a besoin d'une mesure de similarité. Dans une classification non supervise utilisant l’une des techniques classiques, la similarité entre deux phrases peut être mesurée par plusieurs métriques telles que la distance euclidienne, la distance du cosinus. On a choisi la similarité cosinus qui permet de calculer la similarité entre deux vecteurs à N dimensions en déterminant le cosinus de l'angle entre eux. Cette métrique est fréquemment utilisée dans la fouille de textes. [41]

5.3 Génération de résumé :

Pour chaque cluster on a choisi une ou plusieurs phrases pertinentes de l'étape précédente. On a un ensemble non ordonnées de phrases. Le résumé sera donc la réorganisation des phrases en fonction ses positions dans le texte original.

6. Implémentations :

On présente dans ce stade l'implémentation de notre système , on commence tout d'abord par la présentation de l'environnement de développement, en détaillant les différents outils utilisés, puis on explique le déroulement de l'application, et enfin on interprète et on commente les résultats obtenus.

6.1 Environnement de développement :

On présente dans cette section, le langage de programmation Python utilisé, et l'environnement de développement.

6.1.1 Python :

Il existe un certain nombre de raisons pour lesquelles le langage de programmation Python est populaire auprès des professionnels qui travaillent sur des systèmes d'apprentissage automatique. L'une des raisons les plus souvent citées est la syntaxe de Python, qui a été décrite à la fois comme «élégante» et aussi «mathématique». Les experts soulignent que la sémantique de Python a une correspondance particulière avec de nombreuses idées mathématiques courantes, de sorte que il ne faut pas autant de courbe d'apprentissage pour appliquer ces idées mathématiques dans le langage Python. [42]

Python est également souvent décrit comme simple et facile à apprendre, ce qui constitue une grande partie de son attrait pour toute utilisation appliquée, y compris les systèmes d'apprentissage automatique. Certains programmeurs décrivent Python comme ayant un «compromis complexité / performances» favorable et décrivent comment l'utilisation de Python est plus intuitive que d'autres langages, en raison de sa syntaxe accessible.

Python possède également des outils particuliers qui sont extrêmement utiles pour travailler avec des systèmes d'apprentissage automatique. Certains citent un éventail de Framework et de bibliothèques, ainsi que des extensions comme NumPy, où ces accessoires facilitent la mise en œuvre des tâches Python. Le contexte du langage de programmation lui-même est donc également important dans sa popularité pour ces utilisations appliquées. Une autre ressource est un module scikit appelé «machine learning en Python», qui peut guider les professionnels vers l'utilisation de Python à ce titre.

Python est décrit favorablement pour l'apprentissage automatique par rapport à des langages comme Java, Ruby on Rails, C ou Perl. [42]

Le choix de Python a été motivé par les raisons suivantes :

- L'une des principales langues parmi les langues appropriées pour la programmation de problèmes d'apprentissage profond.
- Il dispose un grand nombre de bibliothèques pour le traitement du langage naturel, telles que NLPnet, NLTK
- Un langage simple, productif et utilisable dans presque tous les domaines et systèmes.

6.1.2 Version de Python :

Python a évolué à travers différentes versions. Chaque passage à une version supérieure s'est faite avec une compatibilité ascendante. Cela signifie p. ex. que du code Python s'exécutant avec Python 3.2, marchera également avec Python 3.3 ou versions suivantes. Il y a cependant une exception à cela : le passage de Python 2.x (2000) à Python 3.x (dès 2008). Nous avons utilisé la version (3.7). [43]

6.1.3 Interpréteurs Python :

Il existe de nombreuses implémentations de Python et par conséquents d'interpréteurs. Les plus connus sont :

- ***l'interpréteur de base*** : interpréteur par défaut intégré à toute distribution Python, écrit en C et ainsi parfois dénommé ***CPython***
- IPython : interpréteur interactif très évolué
- IDLE : mini IDE s'appuyant sur l'interpréteur de base et le toolkit graphique tkinter
- Jython : interpréteur écrit en Java (donc tournant sur la VM Java), permettant l'utilisation d'objets Java dans du code Python
- PyPy : interpréteur écrit lui-même en Python et qui a vocation d'être très rapide (JIT compiler)
- IronPython : implémentation du langage Python dans les environnements Microsoft .NET et Mono ; notez que cet interpréteur est resté à la version 2.7 … ce qui est un signe de retard, que le projet n'est plus à jour. [42]

Il faut noter que l'interpréteur de base est l'interpréteur de référence du langage Python, et que les autres interpréteurs n'implémentent souvent pas la toute dernière version du langage.

Nous n'allons présenter ici que les deux plus courants : l'interpréteur de base CPython, et l'interpréteur interactif IPython à travers Jupyter.

6.1.4 L'interpréteur de base CPython :

Lorsque Python est distribué avec le système d'exploitation (ce qui est le cas sous GNU/Linux et macOS), on dispose alors de l'interpréteur de base et de la librairie standard. L'interpréteur CPython permet d'exécuter des scripts/programmes en frappant, depuis une fenêtre terminale : python script.py. [43,42]

L'option -i est particulièrement intéressante : en frappant python -i script.py , juste après l'exécution du script, on entre dans le mode mode interactif de l'interpréteur, ce qui donne la possibilité d'examiner les variables globales où tracer le stack d'erreurs. L'interpréteur de base peut donc aussi être utilisé en mode interactif, et c'est ce qui se passe lorsqu'on lance la commande python sans passer de script en argument. Le mode interactif est alors signalé par le prompt >> . On quittera l'interpréteur en passant la commande quit() ou exit() , ou en frappant <ctrl-D> sous Linux et macOS, ou <ctrl-Z> sous Windows (qui envoie un EOF , c-à-d. un caractère de fin de fichier).

6.1.5 L'interpréteur interactif IPython :

Bien que l'utilisation de IPython soit la majorité du temps faite à travers l'application Jupyter (décrite plus bas), il est également possible de lancer IPython de façon native.

IPython est l'interpréteur Python interactif le plus utilisé actuellement en raison de sa convivialité, de ses vastes possibilités interactives et graphiques.

Google Colab ou Colaboratory que nous avons l'utilisé est un service cloud, offert par Google (gratuit), basé sur Jupyter Notebook et destiné à la formation et à la recherche dans l'apprentissage automatique. [44]

6.1.6 IDE's pour Python :

Utiliser un bon éditeur pour programmer en Python c'est bien. Recourir à un environnement de développement (IDE, Integrated Development Environment) c'est encore plus confortable et puissant. Pour un usage scientifique de Python, Spyder figure parmi les plus répandus.

- **Spyder** : Spyder (Scientific Python Development EnviRonment) est un IDE orienté vers un usage scientifique de Python et doté de fonctionnalités avancées d'édition, debugging,

introspection et profiling. L'utilisateur MATLAB ou GNU Octave GUI se retrouvera dans un environnement familier, avec notamment des fenêtres Console, Editor, Variable explorer, File explorer, History, Online help…etc Disponible sur tous les systèmes d'exploitation, cet IDE est lui-même écrit en Python et s'appuie sur le toolkit graphique multiplateforme Qt (nécessitant les packages PyQt et PySide). [42]

- **PyCharm** : un IDE Python de plus en plus en vogue, existant en versions Community (libre) et Professional (payante, incluant support du développement HTML, JavaScript et SQL)
- **Eclipse** : l'historique IDE libre et multiplateforme, se prête bien entendu aussi au développement Python, complété par l'extension PyDev
- **Pyzo** : IDE libre interagissant bien avec Anaconda
- **Eric Python IDE** : IDE libre basé Qt/Scintilla et architecture de plugins
- **PyScripter** : IDE Python libre spécifiquement Windows

6.1.7 Jupyter Notebook :

L'interpréteur IPython versions 0.12 à 3.x offrait une fonctionnalité appelée Ipython Notebook qui permettait, à la façon d'autres logiciels scientifiques (Mathematica, Maple…), de créer des documents interactifs composés de code Python vivant, de texte formaté (Markdown, HTML et LaTeX) et de graphiques. Cette fonctionnalité a été sortie du projet IPython pour constituer un outil indépendant dénommé Jupyter. L'objectif de cette scission a été de rendre cette technologie de notebooks accessible à d'autres langages de programmation que Python. Bien entendu Jupyter Notebook est toujours utilisable avec le langage Python. [42]

6.2 Les Outils Et Librairies Utilisés :

On présente ci-après les librairies essentielles utilisées ;

6.2.1 Natural Langage Toolkit (NLTK) :

La boîte à outils en langage naturel (NLTK) est une plate-forme utilisée pour créer des programmes Python qui fonctionne avec des données de langage humain pour une application dans le traitement statistique du langage naturel (NLP).

Il contient des bibliothèques de traitement de texte pour la tokenisation, l'analyse, la classification, la racine, le marquage et le raisonnement sémantique. Il comprend également des démonstrations graphiques et des exemples d'ensembles de données, ainsi qu'un livre de recettes et un livre expliquant les principes sous-jacents des tâches de traitement du langage prises en charge par NLTK. [43,42]

6.2.2 OS :

Ce module fournit une manière portable d'utiliser les fonctionnalités dépendantes du système d'exploitation. Si vous voulez uniquement lire ou écrire dans un fichier.

6.2.3 String :

Les chaînes peuvent être créées en insérant des caractères entre guillemets simples ou doubles. Même les guillemets triples peuvent être utilisés en Python mais généralement utilisés pour représenter des chaînes multi-lignes et des docstrings.[42]

6.2.4 Sklearn :

Scikit-learn est une bibliothèque d'apprentissage automatique gratuite pour Python. Il comporte divers algorithmes tels que la machine vectorielle de support, les forêts aléatoires et les k-voisins, et il prend également en charge les bibliothèques numériques et scientifiques Python telles que NumPy et SciPy.[42]

6.2.5 NumPy :

NumPy est une bibliothèque python utilisée pour travailler avec des tableaux. Il a également des fonctions pour travailler dans le domaine de l'algèbre linéaire, de la transformée de Fourier et des matrices. NumPy a été créé en 2005 par Travis Oliphant. C'est un projet open source et vous pouvez l'utiliser librement.[43,42]

6.3 Description de système :

Notre système génère plusieurs sortes de résumé selon la méthode choisie par l'utilisateur et les paramètres personnalisés ;

1ère méthode est basé sur les mots clés automatiques (sac de mots clés) avec classification par arbre binaire.

2ème méthode est basé sur la fréquence TF-IDF avec clustering K-means.

Dans les deux méthodes précédentes il y a la possibilité de combiner d'autres options :

- Prendre le titre d'article en considération ou non, par défaut (oui).
- Introduire d'autres mots clés à prendre en considération.
- Le choix de longueur de résumé (par défaut 3 lignes)

Le texte source à résumer peut être :

- Un fichier au format texte
- Un lien URL, comme les articles dans Wikipedia.

6.4 Démonstration de système :

On présente dans cette section les différentes étapes de déroulement du processus De résumé d'un article (Effet de serre الاحتباس الحراري) sur (wikipedia) dont l'url ci-dessous : [45]

url = "https://ar.wikipedia.org/wiki/%D8%A7%D9%84%D8%A7%D8%AD%D8%AA%D8%A8%D8%A7%D8%B3_%D8%A7%D9%84%D8%AD%D8%B1%D8%A7%D8%B1%D9%8A"

et un autre source sur un fichier data.txt contient un article sur la lune (القمر)

- Exemple chargement de fichier :

le code suivant permet de charger le fichier.

```
with open("data.txt", 'r') as fichierIn:
    lignes = fichierIn.readlines()
    for ligne in lignes:
        Contenu += ligne
        print(ligne)
```

- chargement de texte source :

La capture d'écran suivante présente le chargement de source.

Figure 3.4 chargement du texte source

- Phase prétraitement :

Dans cette phase élimination de ponctuation, normalisation et segmentation
La capture d'écran suivante présente la segmentation.

Figure 3.5 segmentation

- tokenization :

La capture d'écran suivante présente le tokenisation des phrases.

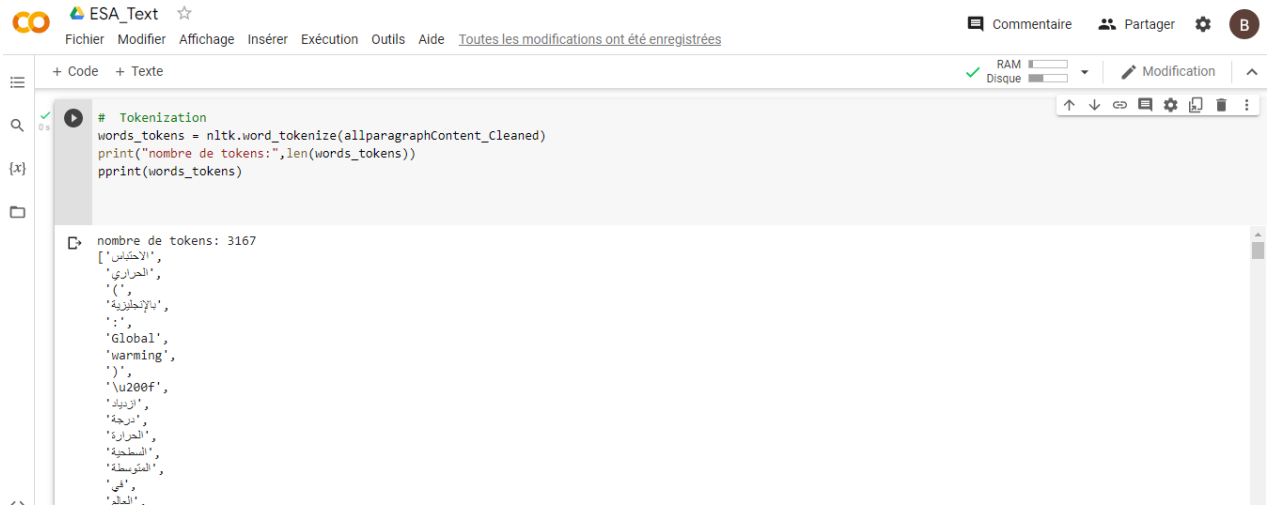

Figure 3.6 Tokenisation

- Phase lemmatisation :

La capture d'écran suivante présente la lemmatisation.

Figure 3.7 lemmatisation

- Phase importation de stopwords :

La capture d'écran suivante présente l'importation des stopwords de langue arabe.

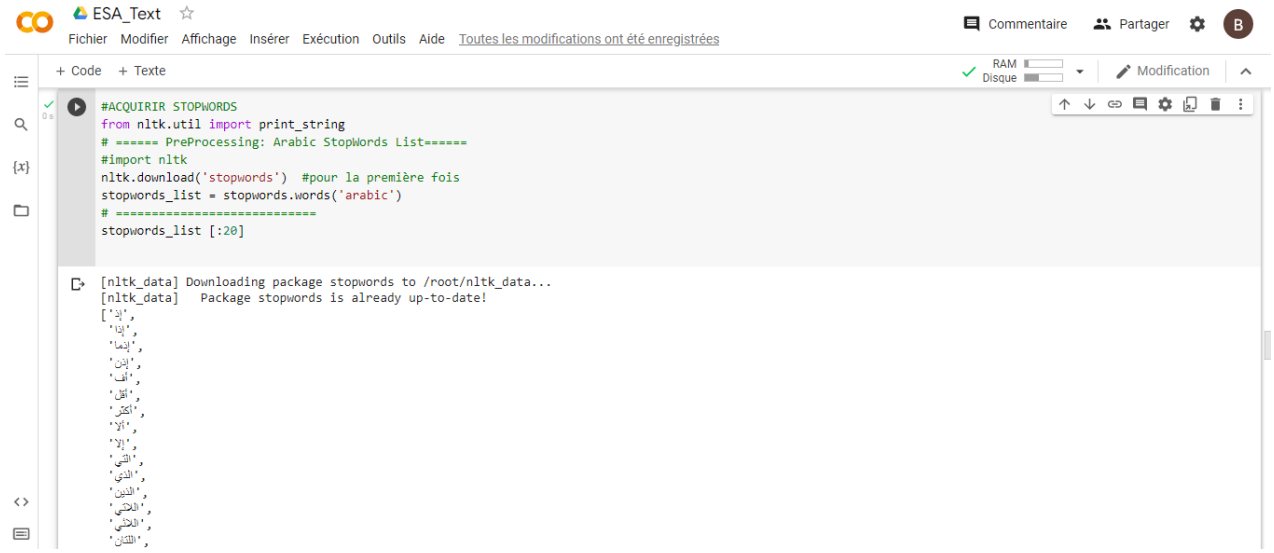

Figure 3.8 importation de stopwords

- Phase calcule de fréquences des termes :

Le code suivant permet de calculer les fréquences des termes

```
word_frequencies = {}
for word in words_stemm:
    if word not in stopwords_list:
        if word not in word_frequencies.keys():
            word_frequencies[word] = 1
        else:
            word_frequencies[word] += 1
# MAX
maximum_frequency_word = max(word_frequencies.values())
#Normalisation
for word in word_frequencies.keys():
    word_frequencies[word] = (word_frequencies[word]/maximum_frequency_word)
```

Ce qui donne le résultat suivant :

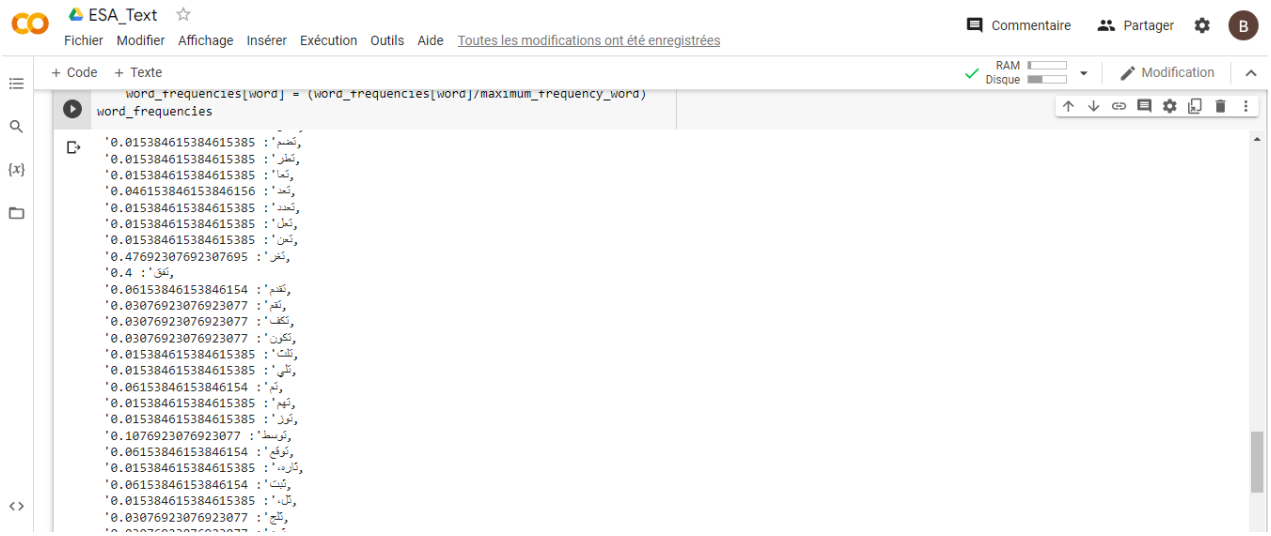

Figure 3.9 termes-fréquences

➢ Phase Calcule de fréquences des phrases :

Figure 3.10 phrases-fréquences

➢ Génération de résumé :

- La capture d'écran suivante présente un résumé d'article (الاحتباس الحراري) de (wikipedia) selon la première méthode.

Figure 3.11 Résumé 1ère Méthode

- La capture d'écran suivante présente un résumé de fichier data.txt selon la deuxième méthode. TF-IDF.

Figure 3.12 Résumé 2ème méthode

7. Conclusion :

Dans ce chapitre, nous avons présenté l'aspect conceptuel de notre approche avec une bref présentation évaluative des algorithmes extractives. Nous avons également détaillé le fonctionnement de tous les processus constituant notre système. Nous avons aussi présenté l'environnement technique de développement qui s'articule autour du langage Python avec ses riches bibliothèques.

CONCLUSION & PERSPECTIVES

Conclusion et perspectives :

Le résumé automatique de texte qui constitue l'une des disciplines les plus importantes du domaine du traitement automatique des langages naturelles ne cesse pas de s'évoluer parallèlement avec la progression des techniques de stockage et par conséquent avec la masse informationnelle très volumineuse disponible sur le Net. Nous pouvons constater que le résumé automatique du texte s'articule autour de deux points. Le premier point concerne les critères utilisés pour choisir le contenu essentiel à extraire. Le deuxième point se focalise sur les moyens qui permettent d'exprimer le contenu essentiel extrait sous la forme d'un texte ciblant les besoins potentiels des utilisateurs.

Pour comprendre tout ce qui est cité au-dessus, nous avons présenté le domaine du traitement automatique des langages naturelle tout en mettant l'accent sur le résumé de texte. Ce dernier a été présenté avec ses approches existantes et ses défis. Nous avons constaté deux approches ; l'approche numérique (dite statistique ou extractive) qui utilise des méthodes de calcul, simplement implantées et rapidement adaptables à d'autres domaines, mais limitée dans le sens où elle n'a pas une vision globale du texte ; l'approche symbolique (dite abstraite) fondée sur des connaissances avancées et qui ne peut être appliquée qu'à des domaines restreints.

Notre choix s'est porté sur les méthodes extractives en raison de leurs avantages, car elles produisent un résumé sûr et répond aux besoins de la majorité des utilisateurs. De plus, elles permettent la création de systèmes non supervisé. L'exploitation de ce type de méthode est faite dans un cadre hybride offrant la possibilité d'introduire des mots clés d'utilisateurs et d'autres options concernant le résumé résultant.

A noter que le résultat de notre système est lié au choix et besoins de l'utilisateur, en plus un autre résumé acceptable et appréciable qu'on peut l'obtenir sans élimination de stopwords.

Cependant, on a rencontré le problème de connaissance approfondie de la langue arabe, à noter que la conception d'un tel système nécessite la collaboration des linguistes.

Nous sommes arrivés à certaine mesure de maitriser les méthodes extractives, et de réaliser un système de résumé automatique de textes en arabes qui peut satisfaire les besoins d'une bonne partie des utilisateurs, pour aller vers un résumé reformulé basé sur le sens, on a la curiosité d'entamer les méthodes abstraites, où il sera utile et faisable de reformuler le résumé résultant de notre système.

Références bibliographiques :

[1] François Yvon, Une petite introduction au Traitement Automatique des Langues Naturelles. In Conférence on Knowledge discovery and data mining (pp 2 -5).
[2] GAHBICHE - BRAHAM, Amélioration des systèmes de traduction par analyse linguistique et thématique : application à la traduction depuis l'arabe (Doctoral dissertation, Université Paris Sud - Paris XI), (2013).
[3] Mourad LOUKAM.
[4] Wikipédia, traitement automatique du langage naturel, consulté le :(05/4/2022).
[5] TORRES - MORENO, Résume automatique de documents, Lavoisier, J. M (2011).
[6] GILLOUX, M. Traitement automatique des langues naturelles. In Annales des télécommunications (Vol. 44, No. 5-6, pp. 301-316). Springer - Verlag. (1989, May).
[7] Gnanasekaran Thangavel, Priya B, J. Nandhini, An Analysis of the Applications of Natural Language Processing in Sectors, ctober 2021 DOI:10.3233/APC210109, pp. 600.
[8] P. Blanca, « La lemmatisation de l'arabe », GIRCSE (Groupe Interdisciplinaire de recherche par Ordinateur sur les Signe's d'Expression), université catholique de Milan, (Italie).
[9] Mohamed Hedi Maaloul. Approche hybride pour le résume automatique de textes. Application _à la langue arabe. Traitement du texte et du document. Thèse de doctorat, Université de Provence - Aix-Marseille I, 2012.
[10] Hammo B., Abu-Salem H., Lytinen S., Evens M., QARAB: A Question Answering System to Support the Arabic Language, Workshop on Computational Approaches to Semitic Languages. ACL 2002, July 2002.
[11] Boubekeur Yassamina, Identification automatique de mots clés dans les textes arabes, Thèse de Master, Université de Djilali BOUNAÂMA Khemis Miliana, 26/05/2016.
[12] Siham Boulaknade « Traitement Automatique des Langues et Recherche d'Information en langue arabe dans un domaine de spécialité : Apport des connaissances morphologiques et syntaxiques pour l'indexation », Thèse de doctorat, October 2008.
[13] Artificial Intelligence Review An International Science and Engineering Journal.2012.
[14] Andrew Turpin, Yohannes Tsegay, David Hawking, and Hugh E Williams." Fast generation of result snippets in web search". In Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. (2007).
[15]Boudraf Khadidja, Les Résumés Automatiques des Documents Textuels, Thèse de Master, UNIVERSITÉ ABDELHAMID IBN BADIS – MOSTAGANEM, (2016).
[16] MaaliMnasri. Résumé automatique multi-document dynamique. Traitement du texte et du document. Université Paris-Saclay, Thèse de doctorat, 2018. Français.
[17] Fouad Soufiane Douzidia, Résumé automatique de texte arabe, thèse de master**,** Université de Montréal**,** 2004.
[18] T.A.S. Pardo, L.H.M. Rino, M.G.V. Nunes, Extractive summarization: how to identify the gist of a text, International Infàrmation TechnoÏogy Symposium - I2TS 2002, fÏorianôpolis-Sc, Brazil, 01-05 October 2002, pp.245-260.
[19]Ishikawa, K., Ando, S., Okumura, A.: Hybrid Texte Summarization Method based on the TF Method and the Lead Method. Proceedings of the Second NTCIR Workshop Meeting on Evaluation of Chinese & Japanese Text Retrieval and Text Summarization. Tokyo. Japan. March 2001. pp.5-219-5-224.

[20] C. Nobata & S. Sekine, Results of CRL/NYU System at DUC-2003 and an Experiment on Division of Document, Proceedings of the Document Understanding Conference (DUC 2003,), Edmonton, Canada. pp. 79-84
[21] M.B. Yllias Chah, C. J. Pinchak: Text Summarization Using Lexical Chains, Proceedings of the Document Understanding Coi/erence (DUC 2001), New Orleans, USA pp135-140.
[22] A. Farzindar, Résumé automatique de textes juridiques, Proposition de projet doctoral, Université de Montréal, Septembre 2003.
[23], J. BERRI: Mise en oeuvre de la méthode d'exploration contextuelle pour le résumé automatique de textes. Implémentation du système SERAPHIN, Actes du colloque de ('L1M'96 MontréaÏ, pp. 128-135.
[24] H. P. Edmundson: New methods in automatic abstracting, Journal ofthe Association for Computing Machineiy (AcM,), vol. 16 N°2 April 1969, pp. 264-285.
[25] T. Strzalkowski, J. Wang and B. Wise, Summarization based Query Expansion in Information Retrieval, Proceedings of 36 Annual Meeting ofthe ACL, Montreal 1998, V. 2, pp. 1258-1264.
[26] Madhuri J. N., Kumar R. G. Extractive text summarization using sentence ranking. Proceedings of the 2019 International Conference on Data Science and Communication (IconDSC); 2019, March; Bangalore, India. IEEE; pp. 1–3.
[27] Khatri C., Singh G., Parikh N. Abstractive and extractive text summarization using document context vector and recurrent neural networks. 2018.
[28] Vimal Kumar K., Yadav D. *Advances in Intelligent Systems and Computing* . Vol. 339. New Delhi: Springer; 2015. An improvised extractive approach to Hindi text summarization; pp. 291–300.
[29] Allahyari M., Pouriyeh S., Assefi M., et al. Text summarization techniques: a brief survey. 2017.
[30] Dutta M., Das A. K., Mallick C., Sarkar A., Das A. K. *Emerging Technologies in Data Mining and Information Security* . Singapore: Springer; 2019. A graph based approach on extractive summarization; pp. 179–187.
[31] Elrefaiy A., Abas A. R., Elhenawy I. Review of recent techniques for extractive text summarization. *Journal of Theoretical and Applied Information Technology* . 2018;**96**(23):7739–7759.
[32] Sanchez-Gomez J. M., Vega-Rodríguez M. A., Pérez C. J. The impact of term-weighting schemes and similarity measures on extractive multi-document text summarization. *Expert Systems with Applications* . 2021;**169** doi: 10.1016/j.eswa.2020.114510.114510
[33] Mihalcea R., Tarau P. Textrank: bringing order into text. Proceedings of the 2004 conference on empirical methods in natural language processing; 2004, July; Barcelona, Spain. pp. 404–411.
[34] Miller D. Leveraging BERT for extractive text summarization on lectures. 2019.
[35] Erkan G., Radev D. R. Lexrank: graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research* . 2004;**22**:457–479. doi: 10.1613/jair.1523.
[36] Christian H., Agus M. P., Suhartono D. Single document automatic text summarization using term frequency-inverse document frequency (TF-IDF) *ComTech: Computer,*

Mathematics and Engineering Applications . 2016;**7**(4):285–294.
doi: 10.21512/comtech.v7i4.3746.
[37] Verma J. P., Patel A. Evaluation of unsupervised learning based extractive text summarization technique for large scale review and feedback data. *Indian Journal of Science and Technology* . 2017;**10**:p. 17. doi: 10.17485/ijst/2017/v10i17/106493.
[38] Verma P., Pal S., Om H. A comparative analysis on Hindi and English extractive text summarization. *ACM Transactions on Asian and Low-Resource Language Information Processing* . 2019;**18**(3):1–39. doi: 10.1145/3308754.
[39] MultiNews dataset reference. https://www.tensorflow.org/datasets/catalog/multi_news . consulté le 20/05/2022.
[40] Divakar Y.et al., Qualitative Analysis of Text Summarization Techniques and Its Applications in Health Domain, 2022, Article ID 3411881, doi : 10.1155/2022/3411881
[41] M. Maaloul, Approche hybride pour le résumé automatique de textes. Application à la langue arabe. Doctorat, Aix-Marseille, 2012.
[42] B.Jean-Daniel et B. Samuel, Installation et utilisation de Python et outils associés, Creative Commons BY-SA,2020, pp. 2-8.
[43] https://docs.python.org, consulté le : 01/05/2022
[44] google, https://colab.research.google.com, consulté le : 01/05/2022
[45] wikipedia, www.wikipedia.com ,consulté le : 04/05/2022

ملخص:

نظرًا للزيادة الهائلة في كمية المحتوى المكتوب باللغة العربية والمتاح عبر الانترنت، أصبح من الضروري توفير أنظمة تلخيص آلية لاستخراج المعلومات الأساسية والمفتاحية من الكمية الكبيرة من النصوص المتاحة. يمكن توفير الكثير من الوقت بمجرد قراءة أو عرض معلومات موجزة مثل الأخبار والمقالات وما إلى ذلك بدلا من قراءة نصوص كاملة. التقنيات الاستخراجية للتلخيص الآلي للنص على الرغم من عيوبها، إلا أنها مستخدمة على نطاق واسع ولها أهمية كبيرة في مجال تلخيص النص. ونظرًا لخصوصية اللغة العربية، يهتم مشروعنا وأبحاث أخرى بتطوير تقنيات استخراجية لتحسين تلخيص النص العربي.

الكلمات المفتاحية: التقنيات الاستخراجية، التلخيص الآلي، اللغة العربية، Python

Abstract:

Due to the exponential increase in the amount of content written in Arabic and available online, it has become necessary to provide automated summarization systems to extract key and key information from the large amount of available text. Much time can be saved by simply reading or viewing brief information such as news, articles, etc. instead of reading full texts. Extractive techniques for automated text summarization, despite their drawbacks, are widely used and are of great importance in the field of text summarization. Given the specificity of the Arabic language, our project and other research are concerned with developing extractive techniques to improve Arabic text summarization.

Keywords: Extractive techniques, automated summarization, Arabic language, Python

Résumé :

En raison de l'augmentation exponentielle de la masse informationnelle disponible en ligne et écrite en arabe, il est devenu nécessaire de fournir des systèmes de résumé automatique pour extraire les informations clés et nécessaires de la grande quantité de texte disponible. Beaucoup de temps peut être gagné en lisant simplement ou en visionnant de brèves informations telles que des nouvelles, des articles, etc. au lieu de lire des textes complets. Les techniques extractives pour le résumé automatique de texte, malgré leurs inconvénients, sont largement utilisées et revêtent une grande importance dans le domaine du résumé de texte. Compte tenu de la spécificité de la langue arabe, notre projet et d'autres recherches portent sur le développement de techniques extractives pour améliorer la synthèse de texte arabe.

Mots clés : Techniques extractives, résumé automatique, langue arabe, Python.