République Algérienne démocratique et Populaire
Université Abbes Laghrour khenchela
Faculté des Sciences Et De La Technologie
Département De Mathématique d'Informatique

Mémoire De Fin d'étude

Spécialité : Génie De Logicielle Système Distribué

Thème

Résumé automatique d'un document (article scientifique) par l'élection des phrases pertinentes

***Réalisé** par :*

Benaroua Anouar

Sahraoui Abderrahmane

***Encadré** par :*

Mr Boussalem Mohamed

2020/2021

Dédicaces

À nos très chers parents

Nous vous devons ce que nous sommes aujourd'hui, grâce à votre amour, votre patience et vos innombrables sacrifices.
Que ce modeste travail, soit pour vous une petite compensation et reconnaissance pour tout ce que vous avez fait.
Que Dieu, vous préserve et vous procure santé et longue vie afin que nous puissions à notre tour vous combler.

À nos très chers frères et sœurs

Aucune dédicace ne pourrait exprimer assez profondément ce que nous ressentons envers vous.
Nous vous dirons tout simplement, un grand merci, nous vous aimons.

À nos très chers ami(e)s

En témoignage de l'amitié sincère qui nous lie et les bons moments passés ensemble.
Nous vous dédions ce travail en vous souhaitons un avenir radieux et plein de réussites

Remerciement

En premier lieu nous remercions DIEU tout puissant de nous avoir donné la patience, la santé et la volonté pour achever ce travail.

Et Nos remerciements vont tout particulièrement à nos parents, pour leur soutien et leur patience

Nous aimerons adresser plus qu'un merci pour notre encadreur monsieur **Boussalem Mohamed**. Qui a su partager son savoir faire, ses connaissances et son temps pour nous porter aide pendant et hors de ses heures de travail.

Enfin, nous adressons nos plus sincères remerciements à tous nos proches et amis, ils nous ont toujours soutenus et encouragés pendant la réalisation de ce mémoire et tous ceux qui de loin ou de près ont procuré leurs apports, nous exprimons ici grandement notre reconnaissance.

Merci à vous tous.

Abstract

The large increase of text available in digital format has highlighted the need to design and develop effective summary tools in order to locate and extract relevant information in an abbreviated form.

This paper proposes a method of producing abstracts for textual documents. Our methodological approach consisted of studying: the characteristics of the classical abstract, the techniques used in the automatic abstract, the presentation of the development environment, detailing the different tools used and explaining our proposed approach, and the presentation of the architecture Of our application, the explanation of the implementation of the application, and then the evaluation of our system with other reference systems. The objective of this study was to produce the right and best summarized automatically.

Keywords: Summary, Automatic abstract, Summary mono-document, Summary multi-document, evaluation of the automatic abstract

Résumé

La forte augmentation du texte disponible au format numérique a mis en évidence la nécessité de concevoir et de développer des outils de synthèse efficaces afin de localiser et d'extraire les informations pertinentes sous une forme abrégée.

Cet article propose une méthode de production de résumés pour des documents textuels. Notre approche méthodologique a consisté à étudier : les caractéristiques du résumé classique, les techniques utilisées dans le résumé automatique, la présentation de l'environnement de développement, détaillant les différents outils utilisés et expliquant notre démarche proposée, et la présentation de l'architecture de notre application, l'explication de la mise en œuvre de l'application, puis l'évaluation de notre système avec d'autres référentiels. L'objectif de cette étude était de produire le bon et le mieux résumé automatiquement .

Mots clés : Résumé, Résumé automatique, Résumé mono_document , Résumé multi document, évaluation du résumé automatique

ملخص

أبرزت الزيادة الكبيرة في النص المتاح في شكل رقمي الحاجة إلى تصميم وتطوير أدوات تلخيص فعالة من أجل تحديد واستخراج المعلومات ذات الصلة في شكل مختصر

تقترح هذه الورقة طريقة لإنتاج الملخصات للوثائق النصية. يتكون منهجنا المنهجي من دراسة: خصائص الملخص الكلاسيكي ، والتقنيات المستخدمة في الملخص التلقائي ، وعرض بيئة التطوير ، وتفصيل الأدوات المختلفة المستخدمة وشرح نهجنا المقترح ، وعرض بنية تطبيقنا ، شرح تطبيق التطبيق ومن ثم تقييم نظامنا بأنظمة مرجعية أخرى. كان الهدف من هذه الدراسة هو إنتاج ما هو صحيح وأفضل تلخيص تلقائيًا

Table des matières

Table des figures

Introduction générale

La forte augmentation de documents disponibles en format numérique a fait ressortir la nécessité de concevoir des outils spécifiques pour accéder à l'information pertinente. Parmi ces outils on trouve les systèmes de résumé automatique.

Le but du résumé automatique est de produire une version condensée du document source à l'aide de techniques informatiques. Ceci afin d'aider le lecteur à décider si le document en question contient l'information recherchée ou non. La plupart des travaux dans le domaine du résumé automatique sont basés sur l'approche par extraction, ceci consiste à extraire des phrases complètes censées être les plus pertinentes du texte et à les concaténer de façon à produire un extrait. Il s'agit d'appliquer les méthodes statistiques pour attribuer des scores à chaque phrase reflétant son importance dans le texte. Le résumé final ne gardera que les phrases avec un score élevé. Le score attribué à chaque phrase est calculé en combinant les scores des critères utilisés (la fréquence des mots, la position de phrases, etc.). En effet, cette combinaison est plus qu'une simple somme, puisque certains critères sont plus importants que d'autres. Ainsi, il faut attribuer un poids à chaque critère soit manuellement, soit par apprentissage.

Dans ce mémoire de master, nous nous somme fixé l'objectif de proposer une méthode de résumé automatique d'un article scientifique (Résumer la contribution de travail présentée dans l'article). D'un coté, la méthode proposée utilise la structure particulière de l'article scientifique pour minimiser le volume de text à utilisé dans le processus de traitement. D'un autre coté, cette méthode utilise les technique d'extraction utilisée dans le résumé automatique d'un text générale.

Notre système de résumé automatique d'un article scientifique repose principalement sur deux modules qui peuvent communiquer entre eux afin de permettre la génération de résumé. L'entrée de notre système est un article scientifique, pour pouvoir résumer son contribution on doit passer par le module préparation où le système sépare ses sections en utilisant l'architecture particuliière de l'article scientifique.

Pour atteindre cet objectif nous avons structuré notre mémoire comme suit :

Notre mémoireest composé de quatre chapitres auxquels s'ajoutent une introduction générale et une conclusion générale et perspectives.

Le premier chapitre présente les concepts de text mining, où sont décrits les notions de base de ce domaine , Les techniques de prétraitement de text pour les exploiter par les techniques de text mining tel que le résumé automatique de texte..

Nous présentons dans le second chapiter, La résumé automatique de text qui fait le cœur de notre travail, où on présente les déférentes méthodes de résumé automatique de text en évoquant les avantages et les inconvénients de chaque méthode.

Le troisième chapitre a pour objectif de fixer le cadre applicatif que nous avons envisagé. Dans une première partie, nous décrivons brièvement la structure générale d'un article scientifique, la seconde partie est consacrée aux méthodes de résumé automatique d'un article scientifique.

Le chapitre 4 est réservé à notre contribution. Nous présentions à travers ce chapitre la plate-forme de réalisation, et la présentation de notre approche et la présentation de certains fonctionnalités de notre système.

La conclusion et les perspectives de ce travail seront présentées à la fin de ce mémoire.

Chapitre 01 : Fouille de textes

1. Introduction

Dans le monde contemporain, le texte est le moyen le plus commun pour échanger des informations. Les données stockées dans l'ordinateur peuvent être dans l'une des formes structurée, semi-structurée et non structuré. Les outils traditionnels d'exploration de données sont incapables de gérer les textes donnés car cela demande du temps et des efforts pour extraire des informations.

La fouille de textes (Text Mining) est un domaine multidisciplinaire basé sur la recherche d'informations, l'exploration de données, l'apprentissage automatique, statistiques, et la linguistique computationnelle. Dans ce chapitre, nous présenterons des notions de base de texte mining et les techniques de prétraitement de texte pour l'exploiter par les techniques de text mining tel que le résumé automatique de texte.

2. Text Mainnig, également appelée exploration de données textuelles, est le processus de transformation de texte non structuré en un format structuré pour identifier des modèles significatifs et de nouvelles informations. En appliquant des techniques analytiques avancées, telles que Naïve Bayes, Support Vector Machines (SVM) et d'autres algorithmes d'apprentissage en profondeur, les entreprises sont en mesure d'explorer et de découvrir des relations cachées au sein de leurs données non structurées. **[1]**

3. Format des données textuel : le texte est l'un des types de données les plus courants dans les bases de données. Selon la base de données, ces données peuvent être organisées comme:

a. Données structurées: ces données sont standardisées dans un format tabulaire avec de nombreuses lignes et colonnes, ce qui facilite le stockage et le traitement des algorithmes d'analyse et d'apprentissage automatique. Les données structurées peuvent inclure des entrées telles que des noms, des adresses et des numéros de téléphone. **[1]**

b. Données non structurées: ces données n'ont pas de format de données prédéfini. Il peut inclure du texte provenant de sources, comme des médias sociaux ou des critiques de produits, ou des formats multimédias riches tels que des fichiers vidéo et audio. **[1]**

c.Données semi-structurées: comme son nom l'indique, ces données sont un mélange entre des formats de données structurés et non structurés. Bien qu'elle ait une certaine organisation, elle n'a pas une structure suffisante pour répondre aux exigences d'une base de données relationnelle. Des exemples de données semi-structurées incluent les fichiers XML, JSON et HTML. **[1]**

4 Le prétraitement du texte

Le processus d'exploration de texte comprend plusieurs activités qui nous permettent de déduire des informations à partir de données textuelles non structurées. Avant de pouvoir appliquer différentes techniques d'exploration de texte, nous devons commencer par le prétraitement de texte, qui consiste à nettoyer et à transformer des données de texte dans un format utilisable. Cette pratique est un aspect central du traitement du langage naturel (PNL) et elle implique généralement l'utilisation de techniques telles que l'identification de la langue, la tokenisation, le marquage d'une partie du discours, la segmentation et l'analyse syntaxique pour formater les données de manière appropriée pour l'analyse.

4.1 Tokenisation:

Le texte brut contient beaucoup d'aléatoire qui nuit à l'estimation des modèles : les accents, les minuscules, les majuscules, les signes de ponctuation… On peut les garder mais plus de variabilité implique plus de données pour les apprendre. On préfère alors le nettoyer avant de le découper en mot (ou caractères ou syllabe). C'est la seule partie qui est spécifique au langage. Même si langue latine partage les mêmes caractères, elles n'ont pas les mêmes accents, la même façon de composer les mots, les mêmes stopwords ou mots sans importance. [2]

4.2 Stemming

La tige est le processus de réduction de l'inflexion des mots (par exemple trouble, troubles) à leur forme racine (par exemple trouble). La «racine» dans ce cas peut ne pas être un vrai mot racine, mais juste une forme canonique du mot original.

La racine utilise un processus heuristique grossier qui coupe les extrémités des mots dans l'espoir de les transformer correctement en leur forme racine. Ainsi, les mots «trouble», «troublé» et «problèmes» pourraient en fait être convertis en trouble au lieu

de problème parce que les extrémités étaient juste coupées (ughh, comme c'est grossier!).[2]

Il existe différents algorithmes de dérivation. L'algorithme le plus courant, qui est également connu pour être empiriquement efficace pour l'anglais, est Algorithme des porteurs . Voici un exemple de tige en action avec Porter Stemmer:

	Original-words	Stemmed-words
0	**Connect**	Connect
1	**Connected**	Connect
2	**Connection**	Connect
3	**Connects**	Connect
	Original-words	Stemmed-words
trouble		trouble
1	**troubled**	trouble
2	**troubles**	trouble
3	**troublemsome**	trouble

4.3 la lemmatisation

D'après mon expérience, la lemmatisation n'apporte aucun avantage significatif par rapport à la racine à des fins de recherche et de classification de texte. En fait, selon l'algorithme que vous choisissez, cela pourrait être beaucoup plus lent par rapport à l'utilisation d'un stemmer très basique et vous devrez peut-être connaître la partie de discours du mot en question afin d'obtenir un lemme correct. Cet article constate que la lemmatisation n'a pas d'importance impact sur la précision de la classification de texte avec des architectures neuronales. [2]

Personnellement, j'utiliserais la lemmatisation avec parcimonie. Les frais généraux supplémentaires peuvent en valoir la peine ou non. Mais vous pouvez toujours l'essayer pour voir l'impact qu'il a sur votre métrique de performances.

4.4 Suppression des mots vides

Les mots vides sont un ensemble de mots couramment utilisés dans une langue. Des exemples de mots vides en anglais sont «a», «the», «is», «are», etc. L'intuition derrière l'utilisation de mots vides est qu'en supprimant les mots à faible information du texte, nous pouvons nous concentrer sur les mots importants à la place . Par exemple, dans le contexte d'un système de recherche, si votre requête de recherche est "qu'est-ce que le prétraitement de texte?" , vous voulez que le système de recherche se concentre sur les

documents qui parlent de prétraitement de texte par rapport aux documents qui parlent sur ce que c'est. Cela peut être fait en empêchant tous les mots de votre liste de mots vides d'être analysés. Les mots vides sont couramment appliqués dans les systèmes de recherche, les applications de classification de texte, la modélisation de sujets, l'extraction de sujets et autres. [2]

La suppression des mots vides est efficace dans les systèmes de recherche et d'extraction et de recherche d'informations , s'est révélé non critique dans les systèmes de classification. Cependant, cela aide à réduire le nombre de fonctionnalités prises en compte, ce qui permet de garder vos modèles de taille décente.

Voici un exemple de suppression de mots vides en action. Tous les mots vides sont remplacés par un caractère factice, W :

Originale sentence = this is a text full for contenet and we need to clean it up

Sentece with stop words removed = w w w text full w contenet w w w w clean w w

Les listes de mots vides peuvent provenir d'ensembles préétablis ou vous pouvez créer un personnalisé pour votre domaine . Certaines bibliothèques (par exemple sklearn) vous permettent de supprimer les mots apparaissant dans X% de vos [2]documents, ce qui peut également vous donner un effet de suppression de mots vides.

4.5 Suppression du bruit

Bruit la suppression consiste à supprimer les caractères, les chiffres et les morceaux de texte qui peuvent interférer avec votre analyse de texte. La suppression du bruit est l'une des étapes les plus essentielles du prétraitement de texte. Il dépend également fortement du domaine. Par exemple, dans les Tweets, le bruit peut être tous les caractères spéciaux à l'exception des hashtags, car il signifie des concepts qui peuvent caractériser un Tweet. Le problème avec le bruit est qu'il peut produire des résultats incohérents dans vos tâches en aval. Prenons l'exemple ci-dessous:[2]

Raw-words		**Stemmed-words**
0	**..trouble..**	..trouble..
1	**troublr<**	troublr<

2	**trouble !**	trouble !
3	**<a>trouble</>**	<a>trouble</>
4	**1.trouble**	1.trouble

Notez que tous les mots bruts ci-dessus contiennent du bruit ambiant. Si vous arrêtez ces mots, vous pouvez voir que le résultat obtenu n'a pas l'air très joli. Aucun d'entre eux n'a de tige correcte. Cependant, avec un certain nettoyage tel qu'appliqué dans ce notebook , les résultats sont désormais bien meilleurs: [2]

	Raw-words	Cleaned-words	Stemmed-words
0	**..trouble..**	Trouble	troubl
1	**troublr<**	Trouble	troubl
2	**trouble !**	trouble	troubl
3	**<a>trouble</>**	Trouble	troubl
4	**1.trouble**	Trouble	troubl

La suppression du bruit est l'une des premières choses que nous devons examiner en matière d'exploration de texte et de PNL. différentes façons de supprimer le bruit. Cela inclut la suppression de la ponctuation , la suppression des caractères spéciaux , la suppression des nombres, la suppression du format HTML, la suppression des mots clés spécifiques au domaine (par exemple 'RT' pour retweet), suppression du code source, suppression de l'en-tête et plus. Tout dépend du domaine dans lequel vous travaillez et de ce qui entraîne du bruit pour votre tâche. Le extrait de code dans mon cahier montre comment faire quelques nois de base Suppression. [2]

5. Représentation de texte

Ce titre montre comment les textes sont transformés en vecteur de nombres pour être utilisés par les approches mettant en œuvre des apprentissages numériques. En général, les représentations n'utilisent pas d'information grammaticale ni d'analyse syntaxique des mots :seule la présence ou l'absence de certains mots est porteuse d'informations.

Nous présentons dans ce chapitre une méthode originale de sélection de descripteurs en deux étapes, qui présente plusieurs avantages. Elle est entièrement automatique et ne nécessite pas de ressources externes (comme une liste de mots les plus fréquents dans une langue donnée) et elle est couplée avec un critère d'arrêt pour trouver le "bon" nombre de descripteurs.[3]

5.1 La représentation en sac de mots

La représentation des textes la plus simple a été introduite dans le cadre du modèle vectoriel, et porte le nom de "sac de mots". Les textes sont transformés simplement en vecteurs dont chaque composante représente un terme. Dans un premier temps, les termes sont les mots qui constituent un texte. Dans les langues comme le français ou l'anglais, les mots sont séparés par des espaces ou des signes de ponctuations ; ces derniers,tout comme les chiffres, sont supprimés de la représentation. On peut choisir de conserver les majuscules pour aider, par exemple, à la reconnaissance de noms propres, mais il faut alors résoudre le problème des débuts de phrase. Les composantes du vecteur sont une fonction de l'occurrence des mots dans le texte.[3]

À titre d'exemple, nous présentons, sur la Figure 5.1, une dépêche de l'Agence France Presse qui fournit des informations sur des prises de participations entre des entreprises. La transformation de ce texte en vecteur est présentée sous le texte. À partir de ces informations,un filtre doit détecter que cette dépêche est pertinente pour le thème des participations.Cette représentation des textes exclut toute analyse grammaticale et toute notion de distanceentre les mots : c'est pourquoi cette représentation est appelée "sac de mots".[3]

5.1.1 A one-hotencoding

Cet algorithme est utilisé pour générer un vecteur dont la longueur est égale au nombre de catégories dans votre ensemble de données, une catégorie étant un seul mot distinct. Disons par exemple que nous voulons une représentation d'encodage à chaud pour chacun des trois documents suivants correspondant aux avis sur un restaurant.[3]

Figure 1

La représentation d'encodage à chaud de chaque document se fait en suivant ces étapes :

- Étape 1 : Créer un ensemble de tous les mots du corpus

Set of all the words in the corpus
great
restaurant
and
service
they
can
do
better
to
provide
only
Two
thumbs
up
worst
ever

- Étape 2 : Déterminer la présence ou l'absence d'un mot donné dans une revue particulière. La présence est représentée par 1 et l'absence représentée par 0. Chaque avis sera alors représenté comme un tuple de 0, 1 éléments.[3]

Set of all the words in the corpus		**R1:** Great Restaurant and great service !	**R2:** They can do better to provide better service	**R3:** Only two thumbs up, worst service ever
great		1	0	0
restaurant		1	0	0
and		1	0	0
service		1	1	0
they		0	1	0
can		0	1	0
do		0	1	0
better		0	1	0
to		0	1	0
provide		0	1	0
only		0	0	1
Two		0	0	1
thumbs		0	0	1
up		0	0	1
worst		0	0	1
ever		0	0	1

Figure 2

À la fin des deux étapes, nous pouvons enfin obtenir la représentation d'encodage à chaud de nos trois revues (R1 à R3). Cette technique semble simple, mais présente les inconvénients suivants :

- Le vocabulaire du monde réel a tendance à être énorme, de sorte que la taille du vecteur représentant chaque document sera également énorme, quel que soit le nombre de mots dans un document donné**.**
- On perd complètement l'ordre dans lequel les mots apparaissent dans la revue/document, ce qui conduit malheureusement à une perte de contexte.
- L'information de fréquence des mots est perdue en raison des représentations binaires. Par exemple, les mots super apparaissent deux fois dans la première revue et le mot mieux apparaît deux fois dans la deuxième revue, mais il n'y a aucun moyen de l'exprimer, tout ce que nous savons, c'est leur existence.[3]

5.2 Tf-Idf

Cet algorithme est une amélioration des vecteurs de comptage et est largement utilisé dans les technologies de recherche. Tf-Idf signifie Terme de fréquence-Fréquence de document inverse. Il a tendance à capter :

- La fréquence à laquelle un mot/terme Wi apparaît dans un document dj. Cette expression peut être représentée mathématiquement par Tf(Wi, dj)
- La fréquence à laquelle le même mot/terme apparaît dans l'ensemble du corpus D. Cette expression peut être mathématiquement représentée par df(Wi, D).
- Idf mesure la rareté du mot Wi dans le corpus D.

Avec ces informations supplémentaires, nous pouvons calculer le Tf-Idf en utilisant le produit des valeurs tf et idf en utilisant la formule suivante :

$$\textbf{tf-idf}_{X_{i,j}} = \textbf{tf}(W_i , d_j).\textbf{idf}\ (W_i ,D)$$

Chaque document dj sera alors représenté par le score Tf-Idf de chaque mot dans ce document comme ceci :

$$d_j = [x_{1j} , x_{2j} , \ldots . , x_{nj}]$$

Tf donne plus d'importance (poids) aux mots apparaissant plus fréquemment dans un même document. D'autre part, Idf essaiera de sous-pondérer les mots qui apparaissent plusieurs fois dans l'ensemble du corpus, pour cela nous pouvons penser à des mots tels que « le », « ceci », « un », « un », etc. Puis les assembler **(Tf-Idf)** permet de capturer les mots rares qui n'apparaissent pas fréquemment dans le document.[3]

D.1) Avantages

- Capture à la fois la pertinence et la fréquence d'un mot dans un document.

D.2) Inconvénient

- Chaque mot est toujours capturé de manière autonome, donc le contexte dans lequel il apparaît n'est pas capturé.

5.3Word Embedding

Par analogie, le wordembedding est capable de saisir la dimension de capturer le contexte, la similarité sémantique et syntaxique (genre, synonymes, …) d'un mot. Par exemple, on pourrait s'attendre à ce que les mots « chien » et « chat » soient représentés par des vecteurs relativement peu distants dans l'espace vectoriel où sont définis ces vecteurs.[3]

Comme pour les images, nous pensons que ça soit le modèle qui choisit les caractéristiques les plus pertinentes représentant le mot. Par exemple, la caractéristique « être vivant » pourrait être intéressante pour différencier « chien » et « ordinateur », et rapprocher « chien » et « chat ».[3]

6. Mesure de similarité entre les documents

Une mesure de similarité est, en général, une fonction qui quantifie le rapport entre deux objets, comparés en fonction de leurs points de ressemblance et de dissemblance. Les deux objets comparés sont, bien entendus de même type[4]

6.1 Mésure de cosinus

La similarité cosinus est fréquemment utilisée en tant que mesure de ressemblance entre deux documents d1 et d2. Il s'agit de calculer le cosinus de l'angle entre les représentations vectorielles des documents à comparer. La similarité obtenue Simcosinus(d1; d2) ϵ [0; 1]. [4]

$$\textbf{Sim cos(d1,d2)}= \frac{\mathbf{d1}^{\rightarrow}.\mathbf{d2}^{\rightarrow}}{||\boldsymbol{d2}^{\rightarrow}||||\boldsymbol{d2}^{\rightarrow}||}$$

6.2 Mésure de Jaccard

L'indice de Jaccard ou coefficient de Jaccard est le rapport entre la cardinalité (la taille) de l'intersection des ensembles considérés et la cardinalité de l'union des ensembles. Il permet d'évaluer la similarité entre les ensembles. Les documents d1 et d2 sont donc représentés, non pas comme des vecteurs, mais comme des ensembles de termes. La similarité obtenue Simjaccard(d1; d2) ϵ [0; 1]. [4]

$$\mathbf{Simjaccard(d1,d2)=\frac{||d1\cap d2||}{||d1\cup d2||}}$$

Il est aussi possible d'utiliser la représentation vectorielle.

$$\text{Simjaccard(d1,d2)=}\frac{\mathbf{d1^{\rightarrow}.d2^{\rightarrow}}}{\mathbf{||d1^{\rightarrow}.|||.d2^{\rightarrow}||-d1^{\rightarrow}.d2^{\rightarrow}}}$$

6.3 Mésure dedice:

L'indice de Dice mesure la similarité entre deux documents d1 et d2 en se basant sur le nombre de termescommuns à d1 et d2.

$$\mathbf{Simdice(d1,d2)=\frac{2N}{N1+N2}}$$

Où Nc est le nombre de termes communs à d1 et d2, et N1 (resp. N2) est le nombre de termes de d1 (resp. d2).[4]

7 Conclusion :

Dans ce chapitre, nous avons présenté des notions de base de texte mining et les techniques de prétraitement de texte pour les exploiter par les techniques de text mining tel que le résumé automatique de texte.

Chapitre 2 : Résumé automatique de texte

1. Introduction

Le résumé automatique se propose de faire une extraction de l'information jugée importante d'un texte d'entrée pour construire, à partir de cette information, un nouveau texte de sortie,condensé. Ce nouveau texte permet d'éviter la lecture en entier du document source.Nous présentons dans ce chapitre l'état de l'art du résumé automatique ainsi une présentationde leurs différentes étapes et techniques.

2. Le Résumé Automatique

Le résumé automatique c'est de faire par une machine la tache faite par un humain résumer, et aussi c'est un résumé qui est généré par un logiciel ou un système d'information, à partir d'un texte source on produit un texte court. Le résumé automatique de document est un processus de compréhension avec perte d'informations, à la différence des méthodes et logiciels de compression du texte.

3. Pourquoi le résumé automatique ?

Le but d'un résumé automatique de texte est de produire une représentation abrégée d'un ou de plusieurs documents. Il peut aider à traiter de façon efficace cette masse grandissante d'informations que les personnes s'avèrent tout simplement incapables d'absorber.

4. Approches du résumé automatique :

Il existe deux approches en matière de résumé automatique : l'approche par compréhension et l'approche par extraction : **[11]**

4.1 Approche par compréhension :

Elle repose sur des modèles fondés sur des concepts de psychologie cognitive et sur le paradigme de l'intelligence artificielle. À partir d'un texte source, elle permet de générer un nouveau texte, avec de nouvelles phrases et de nouvelles constructions syntaxiques. Pour obtenir un résumé pertinent, il faut coder un grand nombre de connaissances qui ne figurent pas toujours explicitement dans le texte originel.

4.2 Approche par extraction :

Elle est utilisée dans des produits commerciaux et dans certains laboratoires, est inspirée du postulat : « Pour résumer, il suffit d'extraire ». Elle repose sur des algorithmes de repérage d'unités textuelles pertinentes. Le résumé respecte la linéarité et la structure du texte source, dans la suite de ce chapitre on va détailler cette approche.

5. Les étapes du résumé automatique :

Dans le résumé automatique de documents, on peut identifier trois différentes étapes. Ces étapes sont : l'identification du thème, l'interprétation, et la génération du résumé. La plupart des systèmes aujourd'hui utilisent la première étape seulement. L'identification des thèmes produit un résumé simple ; une fois le système repère les unités importantes, il les présente comme un extrait. Ensuite, l'interprétation qui comporte lafusion des concepts, l'évaluation, ou autres procédures qui utilisent une connaissance autre que le (les) document(s) d'entrée. Le résultat de l'interprétation est un abstrait non lisible, ou un extrait incohérent. Donc, l'étape de génération sert à produire un texte (document) lisible par l'humain, et dans le cas de l'extrait cette étape peut être considérée comme étape de "lissage" pour rendre le résumé plus cohérent.**[12]**

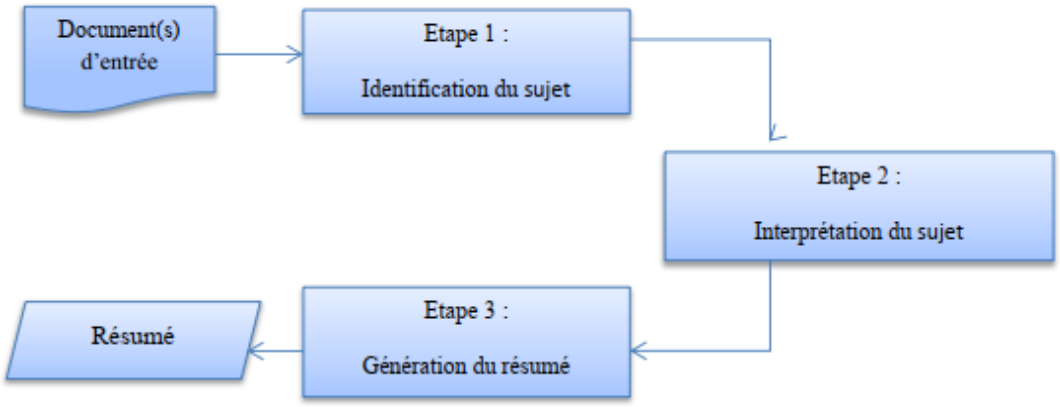

Figure 3 : Les étapes du résumé automatique [12]

Etape 1 : Identification des thèmes :

Elle sert à produire un résumé simple (extrait) en détectant les unités importantes dans le document (mot, phrase, paragraphes, etc.). Les systèmes de résumé qui utilisent seulement l'étape d'identification du thème, produisent un résumé extractif. Ceci se fait par filtrage du fichier d'entrée pour obtenir seulement les thèmes les plus importants. Une fois ces thèmes identifiés, ils sont présentés sous forme d'un extrait. Pour effectuer cette étape, presque tous les systèmes utilisent plusieurs modules indépendants. Chaque module attribue un score aux unités d'entrée (mot, phrase ou passage plus long), puis un module de combinaison, combine les scores pour chaque unité afin d'attribuer un score unique. Enfin, le système renvoie les unités du plus haut en score, en fonction de la longueur du résumé, demandé par l'utilisateur ou fixé préalablement par le système.

Etape 2 : Interprétation des thèmes :

Dans l'interprétation, le but est de faire un compactage en réinterprétant et en fusionnant les thèmes extraits pour avoir des thèmes plus brefs. Ceci est indispensable du moment que les abstraits sont généralement plus courts que les extraits équivalents. Cette deuxième phase de résumé automatique (passage de l'extrait vers l'abstrait) est naturellement plus complexe que la première. Pour compléter cette phase, le système a besoin de connaissances sur le monde (par exemple, les anthologies), puisque sans connaissance aucun système ne peut fusionner les sujets extraits pour produire des sujets moins nombreux afin de former une abstraction.

Lors de l'interprétation, les thèmes identifiés comme importants sont fusionnés, représentée en des termes nouveaux, et exprimé en utilisant une nouvelle formulation, en utilisant des concepts ou des mots qui n'existent pas dans le document original.

Etape 3 : Génération du résumé :

Le résultat de l'interprétation est un ensemble de représentations souvent non lisibles, c'est le cas du résumé par abstraction. Pour le résumé extractif, le résultat est un extrait rarement cohérent, à cause des références coupées, la négligence des liens entre les phrases, et la redondance ou la négligence de quelques matériels. De ce fait, les systèmes incluent une étape de génération du résumé afin de produire un texte cohérent et lisible par l'humain

6. Les méthodes du résumé par 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 cohérencen'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 **[21]**.

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.

6.1 Les critères de sélection des phrases du résumé

Dans cette partie nous détaillons les critères de sélection des unités textuelles utilisés par les systèmes de résumé. Ces unités peuvent être des phrases, des N-grammes ou n'importe quel segment du texte. Ces critères ne sont pas spécifiques d'une méthode bien déterminée mais sont applicables à tous les types de résumés extractifs qu'ils soient mono-document, multi-document ou dynamiques.

6.1.1 Critères liés au contenu du texte

Cet ensemble de critères s'intéresse au contenu du texte et aux informations qu'il apporte. Le contenu est analysé soit par des approches de surface, comme le calcul des fréquences d'occurrence des mots, soit par des approches sémantiques qui exploitent le sens des mots et leurs relations sémantiques, comme avec l'annotation en rôles sémantiques. Nous citons, dans ce qui suit, les critères les plus utilisés. Fréquence d'occurrence des mots. Ce critère a été introduit initialement par Luhn[22]. L'idée est que les mots les plus fréquents sont les plus liés au sujet du texte.

6.1.2 La fréquence d'occurrence des mots

Est largement exploitée, même dans des systèmes récents où elle est combinée à d'autres critères. Même les méthodes reposant sur l'analyse sémantique des mots utilisent la fréquence d'occurrence comme première étape pour déterminer les thèmes principaux abordés par le texte. Le point fort de ce critère est qu'il est totalement indépendant de la langue.

6.1.3 Similarité entre les phrases

La similarité textuelle est une notion très importante en TAL comme en témoignent les évaluations SemEval par exemple. De nombreuses mesures de similarité textuelle ont ainsi été établies [23]. Dans le domaine du résumé automatique, cette similarité est d'abord exploitée pour l'élimination de la redondance mais aussi plus indirectement pour la sélection de phrases pertinentes, sans oublier la comparaison avec des résumés modèles lors de l'évaluation. Certaines méthodes de résumé s'appuient uniquement sur ce critère. Tel est le cas de l'algorithme de résumé mono-document TextRank[24]. Ce critère est par ailleurs particulièrement important dans le cas multi-document. Dans ce contexte, les documents sont généralement représentés par des vecteurs de mots pondérés avec une mesure comme TF*IDF (Term Frequency * Inverse Document Frequency) **[25]** et regroupés selon la similarité de leurs vecteurs. Plus une phrase est similaire au barycentre du regroupement, plus elle décrit les informations caractéristiques du groupe de documents considéré **[26]** et peut être alors considérée comme représentative de ce groupe, ce qui est un critère de sélection important.

6.1.4 Reconnaissance d'entités nommées / Annotation en rôles sémantiques.

La reconnaissance des entités nommées dans un texte améliore le filtrage des informations pertinentes [**27**]. Elle permet aussi de répondre à des requêtes factuelles (OÙ, QUI, QUAND, etc.) dans le résumé guidé [**28**]. Certains vont au-delà de cette étape et déterminent les rôles sémantiques des entités reconnues [**20**]. L'entité la plus fréquente est identifiée et considérée comme entité principale. Par la suite, les phrases contenant cette entité sont sélectionnées. Enfin, seules les phrases où l'entité principale possède un rôle sémantique fondamental (non auxiliaire) sont gardées pour le résumé. Les rôles sémantiques peuvent aussi être utilisés pour simplifier les phrases complexes, c'est-à-dire les phrases contenant deux prédicats ou plus. Le prédicat est généralement un verbe. Dans ce cas, les prédicats pour lesquels l'entité principale a un rôle auxiliaire sont éliminés.

Ces critères mettent l'accent sur le contenu du texte et le message qu'il communique. Il existe d'autres critères qui ne s'intéressent pas au contenu du texte, mais qui renferment des informations très importantes et décisives dans l'étape de sélection. Elles font l'objet du paragraphe suivant.

6.2 Critères liés à la forme et à la structure du texte

La structure du texte est très importante dans le jugement de la pertinence d'une phrase. En effet, lors de la rédaction d'un texte, l'ordre des phrases n'est pas arbitraire. De plus, les styles de rédaction diffèrent d'un domaine à l'autre. Par exemple, dans le domaine journalistique, les informations les plus importantes sont souvent mentionnées au début du texte. Ceci n'est pas toujours le cas dans un article scientifique ou un roman. Ce facteur a été exploité par les chercheurs en TAL pour déterminer l'importance des segments textuels. Nous expliquons dans cette partie les critères les plus importants.

6.2.1 Position de la phrase

Ce critère dépend de la nature du document et de son genre. Les phrases se trouvant au début sont généralement plus informatives et décrivent le sujet principal du document. De plus, les

phrases situées au début de chaque paragraphe tendent à apporter plus d'informations pertinentes [**29**]. Dans le résumé des articles scientifiques, certains travaux se sont appuyés principalement sur la structure des articles [**30**] pour générer des revues scientifiques. Les revues descriptives (résumé informatif) sont formées par les phrases des parties Résumé et Introduction. En revanche, dans le cas des revues intégratives (critique et comparaison des études), les phrases les mieux notées sont celles des parties Résultats et discussion et Conclusion. Cette approche est déduite de l'analyse d'un corpus de 20 revues scientifiques et de 349 références pointées par ces revues. Il a été constaté que plus que 25% des informations contenues dans les revues ont été extraites de la partie Résumé des articles source.

6.2.2 Similarité avec le titre

Plus une phrase est similaire avec le titre, plus elle est liée au sujet principal du texte [**31**] étant donné que dans la majorité des cas le titre informe de façon très brève sur le contenu principal du texte. La similarité avec les sous-titres est aussi considérée comme indicateur de pertinence.

6.2.3 Longueur de la phrase

La longueur moyenne d'une phrase dans un texte dépend de son genre. Généralement, les phrases très courtes sont considérées comme peu informatives alors que les phrases très longues sont présumées détailler des informations déjà exprimées dans l'ensemble des documents par des phrases plus courtes et donc favoriser la redondance. Cette caractéristique est exploitée en fixant un intervalle de longueur (entre 15 et 30 mots). Une phrase ayant une longueur en dehors de cet intervalle est pénalisée [**32**].

6.2.4 Les mots indices(cueword).

Ce critère prend la forme d'une liste de mots activant ou inhibant la sélection d'une phrase, généralement en fonction du rôle qu'ils permettent d'attribuer à la phrase dans laquelle ils apparaissent (exemple, conclusion, etc.) [**33**]. Ces listes sont constituées manuellement ou définies par apprentissage à partir d'un corpus de documents représentatifs [**34**].

6.2.5 Analyse du discours.

L'analyse du discours permet ainsi de contextualiser les énoncés et de leur donner un rôle par rapport à l'ensemble du texte, rôle qui peut être exploité pour leur sélection dans le cadre du résumé. Parmi les méthodes d'analyse du discours qui [**37**] ont été largement appliquées pour le résumé, on peut citer la Rhetorical Structure Theory (RST) [**38**].

La RST s'appuie sur une segmentation des textes en unités discursives élémentaires classées selon leur importance en noyaux (information essentielle) et satellites (information marginale). Elle représente la cohérence et la structure du texte par un ensemble de relations rhétoriques entre les noyaux et les satellites : exemplification, preuve, justification, etc.

6.3 Exploitation et intégration des critères

Il est très rare qu'un système de résumé automatique utilise un seul critère pour sélectionner les phrases du texte source. Plusieurs critères sont combinés. Les méthodes d'intégration sont assez nombreuses. Nous décrivons dans cette partie les différentes méthodes pour combiner les critères et les utiliser pour sélectionner les phrases du résumé.

6.3.1 Méthodes par apprentissage automatique

La plupart des travaux sur le résumé automatique ont considéré ce dernier aussi bien comme un problème de classification que comme un problème de régression. Étant donné un ensemble de textes source et leurs résumés, les méthodes par apprentissage visent à apprendre un modèle de choix des phrases du résumé. Les phrases des textes source sont caractérisées par divers critères de sélection.

- **Un problème de classification**. Dans l'approche par classification, le modèle choisi distingue les phrases du texte à inclure dans le résumé et celles à ne pas inclure dans le résumé. Le modèle bayésien naïf donne généralement les meilleurs résultats [**35**].
- **Un problème de régression.** Dans l'approche par régression, le modèle prédit les scores des phrases [**42**]. La décision est alors quantifiée. L'ordonnancement des phrases reste à la charge du système de résumé.

6.3.2 Méthodes fondées sur les graphes

En représentant un texte sous la forme d'un graphe de phrases, il devient possible d'appliquer un certain nombre d'algorithmes génériques, comme l'algorithme **PageRank** [**51**], pour déterminer l'importance relative de celles-ci.
PageRank est un algorithme de classement utilisé par le moteur de recherche de Google. Il représente les pages Web par les sommets d'un graphe et les hyperliens entre ces pages par des arcs entre ces sommets. Il attribue récursivement à chaque nœud un score dépendant à la fois de ses arcs entrants et du score des nœuds source de ces arcs.
TextRank est un algorithme pour le résumé automatique mono-document fondé sur les graphes [**52**]. Le texte est représenté par un graphe où les sommets sont tout simplement les phrases du texte. Alors que les arcs des arbres rhétoriques représentent des relations rhétoriques entre les phrases, les arcs dans TextRank représentent leurs similarités. Pour ne pas favoriser les phrases longues au détriment des phrases courtes, la valeur de la similarité entre deux unités textuelles est divisée par la somme de leurs longueurs. Initialement, à chaque sommet est attribué un score aléatoire. Par la suite, à chaque itération de l'algorithme TextRank, le score de chaque nœud est calculé récursivement en fonction de sa similarité avec ses voisins et des scores de ces derniers.

6.3.3 Méthodes fondées sur l'ILP

À l'origine des approches fondées sur l'ILP (Interger Linear Programming 2), [**54**] ont proposé d'exprimer le problème du RA sous la forme d'un problème ILP dont la fonction objectif cherche à maximiser le poids des phrases sélectionnées. Ce poids est pénalisé par la redondance avec les phrases déjà incluses dans le résumé. Le modèle intègre en outre la contrainte de la taille maximale du résumé. Ce problème a ensuite été reformulé par [**55**] en définissant une fonction objective se focalisant sur la

maximisation du poids des bigrammes de mots sélectionnés, toujours sous la contrainte de la longueur maximale du résumé.

7. Conclusion :

Dans ce chapitre nous avons abordé les notions du résumé de texte et les étapes de ce dernier, nous avons vu aussi les méthodes de résumé automatique de texte, et comme dernier point nous avons détaillé les méthodes de résumer automatique de texte par extraction, afin de l'utiliser dans notre travail.

Chapitre 03 : état de l'art : Résumé automatique d'un article scientifique

1. Introduction

Rédiger un article scientifique peut s'avérer un exercice plus complexe qu'il n'y paraît au premier abord. L'écriture scientifique possède, en effet, son propre code qui diffère de celui qui s'applique à l'écriture utilitaire ou à l'écriture créative. L'écriture scientifique requiert des phrases courtes, concises et directes, alors que plus de liberté est permise dans la création littéraire. L'écriture scientifique est également régie par des règles précises concernant la présentation et les contenus à aborder.

2. Structure d'un article scientifique

Toute publication scientifique possède une présentation normée. Souvent présente en ligne, elle permet de donner un aperçu de l'article grâce au résumé et aux mots-clés,

Titre : chaque article scientifique possède un titre

Le résumé

La grande majorité des articles possède un résumé, souvent présenté en première page. Il permet aux lecteurs de comprendre l'objectif de l'article, de savoir quel sera le sujet traité.

Les mots-clés

Le résumé s'accompagne souvent d'une série de mots-clés. Ces derniers permettent que l'article soit plus facilement référencé et trouvable en ligne.

Introduction

L'introduction permet de donner au lecteur les bases du sujet, le "pourquoi" de la recherche. Son rôle de cadrage de l'information définit l'axe de recherche, et mène à une problématisation.

Méthodologie

La méthodologie répond au “comment” de la question de recherche scientifique. Cette section constitue le noyau central de l’article. Elle permet d’expliquer en détails les principaux éléments de la recherche, les étapes de sa réalisation, ainsi que l’approche expérimentale utilisée pour valider les hypothèses.

Résultats

Cette section présente les résultats. Ceux-ci sont parfois présentés sous forme de tableaux, schémas ou graphiques afin de mieux les analyser.

Discussion

Une discussion doit suivre les résultats, afin de les analyser et de leur donner une signification scientifique. Dans certaines revues, il est possible de trouver la discussion dans la section des résultats.

Conclusion

La conclusion permet de dresser un bilan, établir un résumé des résultats et des principales interprétations de la recherche. C’est un lieu où le contexte (ou cadre théorique) peut être rappelé et comparé aux résultats obtenus.

Remerciements

La section dédiée aux remerciements est facultative.

Bibliographie

La bibliographie présente une liste de sources (articles, thèses et autres publications). Cette espace reprend chacune des références citées dans l’article qui ont un lien direct avec le sujet.

3. Résumé automatique d’un article scientifique

La diffusion des résultats de recherche originaux par la publication d’articles scientifiques est essentielle pour permettre le développement des connaissances, l’amélioration des pratiques et l’émergence de débats. La lecture de plusieurs articles scientifiques de A à Z est une tâche très

couteuse en matière de temps et d'effort , surtout avec la grande masse de documents sur internet, pour cette raison la communauté scientifique a pensé d'aider les chercheurs par la proposition des système de résumé automatique de texte, afin de ne pas lire tout le document et d'avoir une idée claire sur ce dernier sans le lire entièrement. Dans la littérature les travaux de résumé automatique d'articles scientifiques se regroupe en trois ensembles d'approche à savoir : le résumé par extraction, le résumé par citation et les approches hybride, dans la section suivante nous allons détailler chaque approche.

3.1 Résumé par extraction

Les premiers travaux portant sur le résumé automatique de textes datent de la fin des années 50 **[33]**. Luhn décrit une technique simple, spécifique aux articles scientifiques qui utilise la distribution des fréquences de mots dans le document pour pondérer les phrases. Luhn était déjà motivé par la problématique de surcharge d'information, face à des quantités qui peuvent paraître dérisoires presque 50 ans plus tard. Il décrit quelques-uns des avantages que présentent les résumés produits de manière automatique par rapport aux résumés manuels : coût de production très réduit, non assujetti aux problèmes de subjectivité et de variabilité observés sur les résumer professionnels. De plus, Luhn avait déjà pensé au problème de la normalisation des motsen proposant une version primitive de stemmer 2 regroupant les mots similaires du point de vue de l'orthographe. La normalisation a pour but premier de s'affranchir des variations orthographiques des mots en regroupant les mots porteurs du même sens. La conséquence directe de la normalisation est la diminution de la complexité des traitements numériques (i.e. moins de termes à considérer lors les calculs).

L'idée de Luhn d'utiliser des techniques statistiques pour la production automatique de résumés a eu un impact considérable, la grande majorité des systèmes d'aujourd'hui étant basés sur ces mêmes idées. Par la suite, **[34]**a étendu les travaux de Luhn en tenant compte de la position des phrases, de la présence des mots provenant de la structure du document (i.e. titres, sous-titres, etc.) et de la présence de mots indices (cuewords, e.g. « significant », « impossible », « hardly », etc.). L'évaluation de son

approche a été faite en comparant manuellement les résumés produits par son système avec des résumés de référence (phrases extraites manuellement). Edmundson a pu montrer que la combinaison –position, mots des titres, mots indices– était plus performante que la distribution des fréquences de mots. Il a également trouvé que la position de la phrase dans le document était le paramètre le plus important.

Les recherches menées par **[35]** au sein du Chemical Abstracts Service (CAS) dans la production de résumés à partir d'articles scientifiques de Chimie ont permis de valider la viabilité des approches d'extraction automatique de phrases.Un nettoyage des phrases reposant sur des opérations d'élimination fut pour la première fois introduit. Les phrases commençant par exemple par « in » (e.g. « in conclusion ») ou finissant par « that » sont éliminées du résumé. Afin que les résumés satisfassent les standards imposés par le CAS, une normalisation du vocabulaire est effectuée, elle inclut le remplacement des mots/phrases par leurs abréviations, une standardisation des variantes orthographiques (e.g. conversion de l'anglais UK en anglais US) et le remplacement des noms de substances chimiques par leurs formules.

Ces travaux ont posé les fondements du résumé automatique de textes. De leur analyse émerge une méthodologie de production des résumés en deux étapes (figure 2.1) : i. identification/sélection des unités (généralement les phrases) importantes dans le document source et ii. Génération du résumé par assemblage des unités les plus importantes.

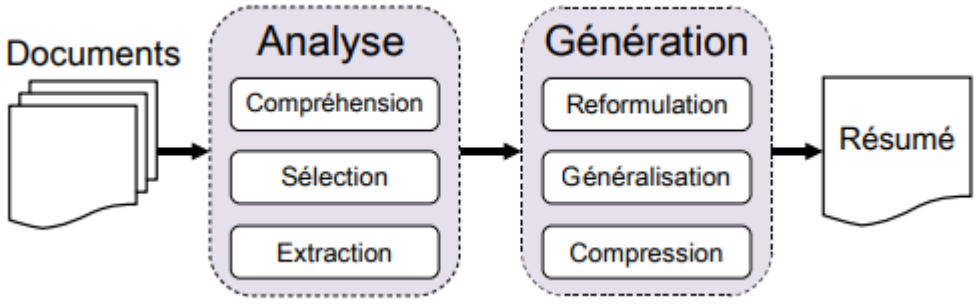

FIGURE 5 : Méthodologie de production d'un résumé par extraction : une première étape d'analyse du document source, suivie d'une étape de génération.

3.2 Résumé par citation

VahedQazvinian, et all. **[47]** proposent une technique utilisant les citations et le résumé signalétique d'un article pour construire un résumé. Ils modélisent l'ensemble des phrases citant un article choisi comme le graphe des citations de l'article (Citation Summary Network). Les arcs de ce graphe vont être décorés par une valeur de similarité des citations. Ils utilisent quatre méthodes pour choisir les phrases pour les résumés à partir de ce graphe : C-LexRank, C-RR, LexRank et MASCS. Ensuite, ils vont comparer les résultats avec un résumé par un humain et un autre composé de phrases choisies aléatoirement. Ils utilisent trois sources d'informations différentes par article :

- Un premier résumé à partir de l'article complet.
- Un autre utilisant le résumé signalétique.
- Un dernier à partir des citations.

Ils en concluent que les citations et les résumés signalétiques contiennent plus d'information unique et utile que le corps de l'article.

La construction de résumé que nous proposons poursuit ces travaux qui ont montré que les citations d'un article contiennent de l'information pertinente pour construire un résumé de l'article cité. Cette information peut être obtenue assez simplement, car plusieurs sites internet contiennent des articles pour lesquels les références ont déjà été extraites (Citeseer, Google scholar et Microsoft Academic Search). Nous allons donc utiliser les citations venant de plusieurs articles pour construire le résumé de l'article cité. Pour éviter de traiter des phrases complexes (contenant plus d'une idée) certains auteurs réduisent la taille des phrases, soit en cherchant les événements atomiques ou en retrouvant la section centrale de la phrase.

Il faut ensuite construire le résumé à l'aide de l'information trouvée. Nous devrons choisir certaines phrases parmi les citations en évitant la redondance. Pour cela, plusieurs auteurs utilisentdes métriques de similarité : MMR, utilisation de SimFinder, et mesure de centralité. Il est aussi proposé d'utiliser des règles afin d'améliorer un résumé.

3.3 Méthodes hybrides

Étant donné que les résumés produits sont généralement peu cohérents à cause de l'extraction de phrases déconnectées de leur contexte, les approches hybrides essayent de combler cette lacune en proposant des méthodes numériques qui tiennent en compte les traits du discours qui assurent sa compréhension. La majorité des travaux sur l'extraction sont fondé sur l'extraction basée sur les connaissances hybrides provenant de différentes sources symboliques et numériques ou même à base d'attribution de poids/score à des informations extraites d'une façon symbolique **[48], [49].** Dans ce contexte d'idée, Ono et Marcu font précéder l'étape d'extraction par une présentation du texte source sous forme d'un arbre RST tout en l'assignant un poids a ses différents nœuds en fonction de leur position. En effet, c'est le score final qui juge la pertinence des nœuds d'un arbre. Ainsi, la sélection de phrases pour le résumé est effectuée en fonction de la longueur désirée du résumé, et le choix est plus ou moins d'éléments, sera fixé selon dans un ordre déterminé par l'algorithme de sélection **[50].**

Après étude des différentes approches pour le domaine du résumé automatique, ainsi que les différentes méthodes utilisées, a notre connaissance, il n'existe pas des travaux de recherche qui ont résolu le problème d'extraction à base d'un contrôle terminant la part de chaque technique dans le résultat final. Il faut, cependant, noter que le traitement du problème de résumé en combinant les méthodes numériques et symboliques, pourrait permettre de franchir un palier et de s'approcher un peu plus de ce que peut faire les résumeras humains. Le paradigme d'extraction des phrases en se basant sur une approche hybride qui privilégiera l'utilisation des techniques numériques et symboliques en fonction des données peut servir, à notre point de vue, à être un pas en avant vers la génération d'extrait d'une meilleure qualité.

4. Conclusion

Le résumé automatique des articles scientifique fait partie de domaine de résumé automatique de texte, par contre il y a des particularités, que ce soit de la structure le l'article scientifique ou de la longue utilisée (Lalongueur des phrases, les termes scientifique, les tableaux …..), dans ce chapitre nous avons étudies les approches existantes du résumé automatique des articles scientifiques.

*Chapitre 04 :*Résumé automatique d'un article scientifique

1Introduction :

Dans ce chapitre nous allons décrire notre système de résumé automatique d'un article scientifique. Résumer un texte consiste à réduire ce texte en un nombre limité de mots. Le texte ainsi réduit doit rester fidèle aux informations et idées du texte original, et dans la mesure du possible rendre compte du style et de l'intention de l'auteur. Cette discipline, quoique très ancienne, est mal formalisée. Le processus de résumé est en effet dépendant à la fois du type de texte à résumer et de l'utilisation qui en sera faite**.**

2 Objectif :

L'objectif de la majorité des systèmes de résumé automatique de texte (mono document) est de résumer un texte complet (fichier, article,), par contre, dans notre travail, l'objectif est de résumer la contribution d'un article scientifique, et pour pouvoir le faire, nous allons présenter la structure particulière de l'article scientifique dans la section suivante.

3Environnement de travail et outils utilisés

L'environnement de travail est constitué par deux parties nommées environnement matériel et environnement logiciel.

3.1 -Environnement matériel

L'environnement matériel utilisé pour accomplir ce travail est caractérisé par :

a. Système d'exploitation : Windows 10 professionnel 64-bit

b. CPU : Intel(R) Core (TM) i7-8700k CPU 3.70GHz 3.70 GHz

c. Mémoire : 32 go ddr4

d. carte graphique : NVidiaGtx 1080 8 go gdd5

3.2 -Environnement de logiciel

a-python

Python est un langage de programmation populaire. Il a été créé par Guido van Rossum

Et sorti en 1991. (w3schools, s.d.)

Il est utilisé pour :

- Développement web (côté serveur),
- Développement de logiciels,
- Mathématiques,
- Scripts système. (w3schools, s.d.)

Que peut faire Python ?

- Python peut être utilisé sur un serveur pour créer des applications Web.
- Python peut être utilisé avec un logiciel pour créer des flux de travail.
- Python peut se connecter aux systèmes de base de données. Il peut également lire et modifier des fichiers.
- Python peut être utilisé pour gérer le Big Data et effectuer des mathématiques complexes.
- Python peut être utilisé pour le prototypage rapide ou pour le développement de logiciels prêts pour la production. (w3schools, s.d.)

Pourquoi Python ?

- Python fonctionne sur différentes plateformes (Windows, Mac, Linux, Raspberry Pi, etc.).
- Python a une syntaxe simple similaire à la langue anglaise.
- Python a une syntaxe qui permet aux développeurs d'écrire des programmes avec moins de lignes que certains autres langages de programmation.
- Python s'exécute sur un système d'interprétation, ce qui signifie que le code peut être exécuté dès qu'il est écrit. Cela signifie que le prototypage peut être très rapide. (w3schools, s.d.)
- Python peut être traité de manière procédurale, orientée objet ou fonctionnelle

b- Les bibliothèques utilisées :

nltk : est une bibliothèque permettant la création de programmes pour l'analyse de texte. Cet ensemble a été créé à l'origine par Steven Bird et Edward Loper, en relation avec des cours

de linguistique informatique à l'Université de Pennsylvanie en 2001. Il existe un manuel d'apprentissage pour cet ensemble titré Natural Language Processing with Python (en anglais).

numby : est une bibliothèque pour langage de programmation Python, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux.
Plus précisément, cette bibliothèque logicielle libre et open source fournit de multiples fonctions permettant notamment de créer directement un tableau depuis un fichier ou au contraire de sauvegarder un tableau dans un fichier, et manipuler des vecteurs, matrices et polynômes.

networkX : est un progiciel en langage Python pour la création, la manipulation et l'étude de la structure, de la dynamique et de la fonction de réseaux complexes. Il est utilisé pour étudier de grands réseaux complexes représentés sous forme de graphes avec des nœuds et des arêtes. En utilisant NetworkX, nous pouvons charger et stocker des réseaux complexes. Nous pouvons générer de nombreux types de réseaux aléatoires et classiques, analyser la structure du réseau, construire des modèles de réseau, concevoir de nouveaux algorithmes de réseau et dessiner des réseaux.

4. Notre approche :

Notre système est un système de résumé automatique d'un article scientifique écrit en anglais basé principalement sur des techniques d'extraction. La mise en œuvre fonctionnelle de notre système est représentée à la figure suivante. Elle repose principalement sur deux modules qui peuvent communiquer entre eux afin de permettre la génération de résumé.

L'entrée de notre système est un article scientifique, pour pouvoir résumer son contribution on doit passer par le module préparation où le système sépare ses sections en utilisant l'architecture particulière de l'article scientifique (section présidente).

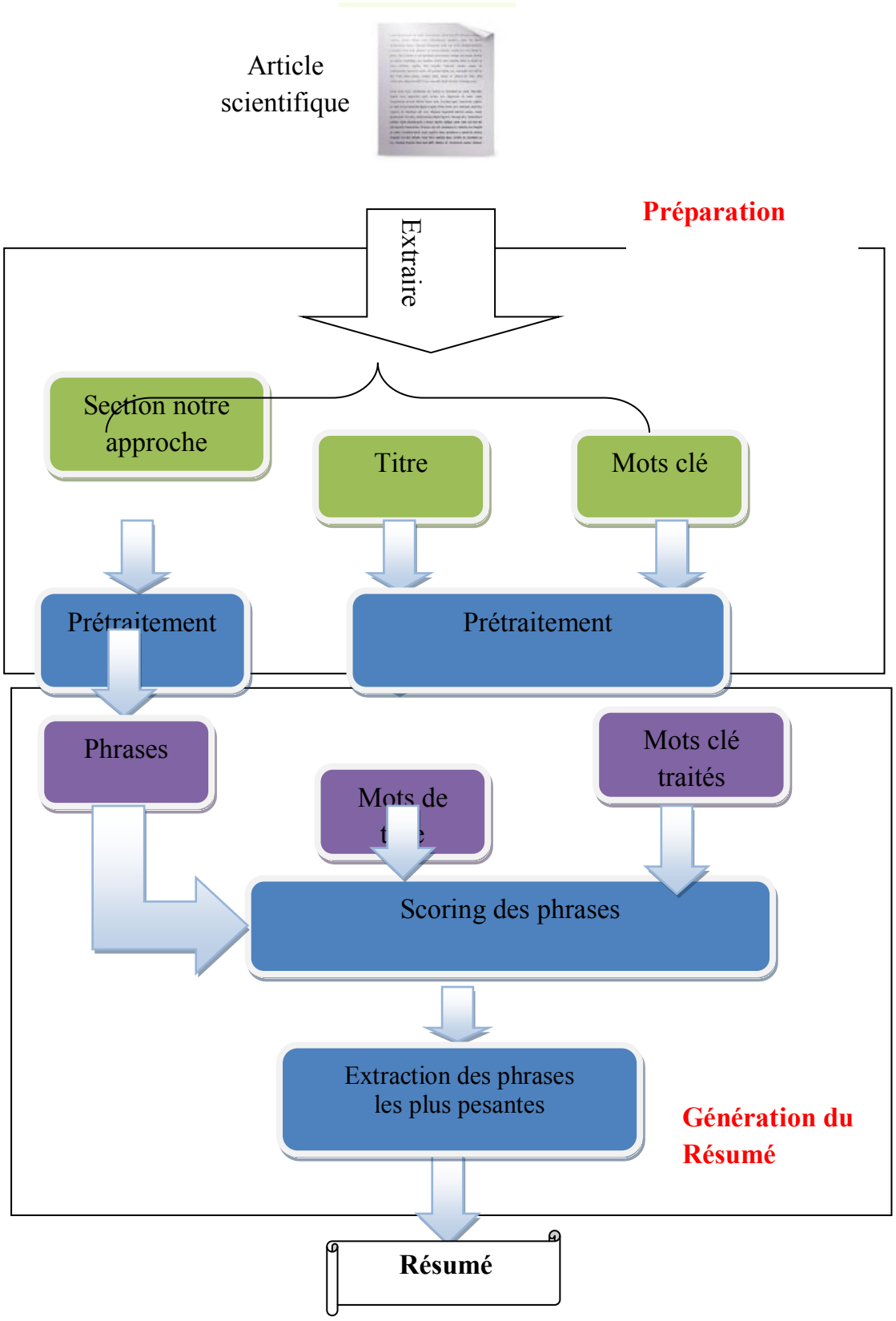

Figure 6 : Architecture globale de notre system

5 Description détaillée de l'architecture globale de l'application :

Notre système est composé en deux modules, dans cette section nous allons les détailler :

5.1Module de préparation

L'entrée de ce module est l'article scientifique qu'on va résumer, comme nous l'avons mentionné précédemment l'objectif de notre travail est de résumer la contribution d'un article scientifique, donc, nous n'avons pas besoin de traiter tout le texte,

a. Extraction des sections nécessaires pour le résumé

- ✓ Extraction de la section notre approche
- ✓ Extraction de titre
- ✓ Extraction des mots clé

b. Prétraitement :

Prétraitement des mots clé et titre

Prétraitement de la section notre approche

Segmentation : l'objectif de la segmentation est d'extraire des phrases séparées

6Génération de résumé :

Les entrées de ce module sont les mots qui compose le titre après le prétraitement, les mots clé et les phrases de la section notre approche, l'objectif de ce module est de sélectionner les phrases qui compose le résumé, ces derniers doivent avoir les scores les plus élevés parmi les phrases candidates, dans la section suivante nous allons expliquer comment calculer le score d'une phrase :

a. Scoring des phrases : ce module assigne pour chaque phrase un score, le score d'une phrase est calculé comme suit :

$$S = s1 + s2 + s3 + s4$$

S1 : la similarité avec le titre :

Étant 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. Dans ce cas on considère les mots du titre du texte comme des mots-clés et on produit le résumé en sélectionnant les phrases qui couvrent certains mots apparaissant dans un titreScore titre(S) = b (t) * F (t)F(t) est la fréquence du terme t dans la phrase

Sb (t) = { A si t∈ liste de mots du titre (A>1)
1 sinon

S2 : l'existence des mots clé :

On extraire du document les mots les plus fréquents c'est-à-dire les mots les plus répétés.

Cette méthode attribue un poids à une phrase selon les mots-clés qu'elle contient, on peut Calculer le score de chaque phrase comme suit :

Score (phrase) = ∑ (A (moti) * F (moti)), i = 1..n

n : nombre total des mots dans la phrase

F (mot) : La fréquence de ce mot dans la phrase.

A (mot) = a > 1 si ce mot appartient à la liste des mots clés. (On a choisi a=3)

= 1 sinon

S3 : l'existence des cue words :

Cette méthode choisit des unités de texte avec des indications spécifiques ou des expressions spécifiques. Par exemple, pour les textes scientifiques, on a comme expressions le but de cetravail …, ce papier présente …, les résultats et des conclusions sont de bons candidats pour indiquer les phrases à inclure dans un résumé. Des textes de types différents peuvent avoir desexpressions indicatives différentes. On peut déduire un score pour une phrase d'un texte quelconque à analyser en fonction de la ressemblance qu'elle présente, pour le trait donné.On pourrait définir le score d'une phrase S correspondant à un certain motif comme : Score-cue

(S) = { 1 si S correspond à un motif
0 Sinon

S4 : la position de la phrase :

Cette méthode attribue un poids à une phrase en fonction de la position de cette phrase dans le document. Elle suppose que la position d'une phrase dans un texte indique son importance

dans le contexte. Les premières et les dernières phrases d'un paragraphe, par exemple, peuvent transmettre l'idée principale et devraient donc faire partie du résumé.

Score (phrase) = βi

βi = B > 0 si i < N

0 si i >= N

N : le nombre total des phrases.

i : la position de la phrase dans le document.

b. Sélectiondes phrase de résumé

Cette partie permet de retourner le résultat final suivant le choix du nombre de phrases extraites par rapport au nombre de phrases contenues dans le document.

Après avoir calculé le score de chaque phrase candidate au résumé, dans cette partie nous allons trier les phrases candidates selon leurs scores.

Les phrases qui possèdent les scores les plus élevés sont les phrases sélectionner par ce processus, le nombre de phrases est déterminer par un pourcentage de résumé.

7. Code source

7.1 Importer des bibliothèques

```
import nltk
from nltk.corpus import stopwords
from nltk.cluster.util import cosine_distance
import numpy as np
import networkx as nx
```

Figure 7 : les bibliothèques

7.2 Fonction de la lecture de l'article

```
def read_article(file_name):
    file = open(file_name, "r")
    filedata = file.readlines()
    article = filedata[0].split(",")
    sentences = []
    for sentence in article:
        sentences.append(sentence.replace("[^a-zA-Z]", " ").split(" "))
    sentences.pop()
    return sentences
```

Figure 8 : la lecture de l'article

7.3 Fonction qui calcule la similarité entre deux sentences

```
def sentence_similary(sent1, sent2, stopwords=None):
    if stopwords is None:
        stopwords = []
    sent1 = [w.lower() for w in sent1]
    sent2 = [w.lower() for w in sent2]
    all_words = list(set(sent1 + sent2))

    vector1 = [0] * len(all_words)
    vector2 = [0] * len(all_words)
    for w in sent1:
        if w in stopwords:
            continue
        vector1[all_words.index(w)] += 1
    for w in sent2:
        if w in stopwords:
            continue
        vector2[all_words.index(w)] += 1
    return 1 - cosine_distance(vector1, vector2)
```

Figure 9 :la similarités entre les phrases

7.4 Fonction de similarité matrix

```
def gen_sim_matrix(sentences, stop_words):
    similarity_matrix = np.zeros((len(sentences), len(sentences)))
    for idx1 in range(len(sentences)):
        for idx2 in range(len(sentences)):
            if idx1 == idx2:
                continue
            similarity_matrix[idx1][idx2] = sentence_similary (sentences[idx1], sentences[idx2], stop_words)
    return similarity_matrix
```

Figure 10 : fonction de similarité matrix

7.5 Fonction qui Faire le Résumé d'article

```
def generate_summary(file_name, top_n=5):
    stop_words = stopwords.words('english')
    summarize_text = []
    sentences = read_article(file_name)
    sentence_similarity_matrix = gen_sim_matrix(sentences, stop_words)
    sentence_similarity_graph = nx.from_numpy_array(sentence_similarity_matrix)
    scores = nx.pagerank(sentence_similarity_graph)
    ranked_sentence = sorted(((scores[i], s) for i, s in enumerate(sentences)), reverse=True)
    for i in range(top_n):
        summarize_text.append(" ".join(ranked_sentence[i][1]))
    print("Summary \n", ". ".join(summarize_text))
```

Figure 11 : Fonction de résumé d'article

7.6 Application de fonction de résumé sur l'article

```
generate_summary("anouar.txt", 3)
```

Figure 12 : Fonction de résumé sur l'article

8. Exécution de code

8.1 Input

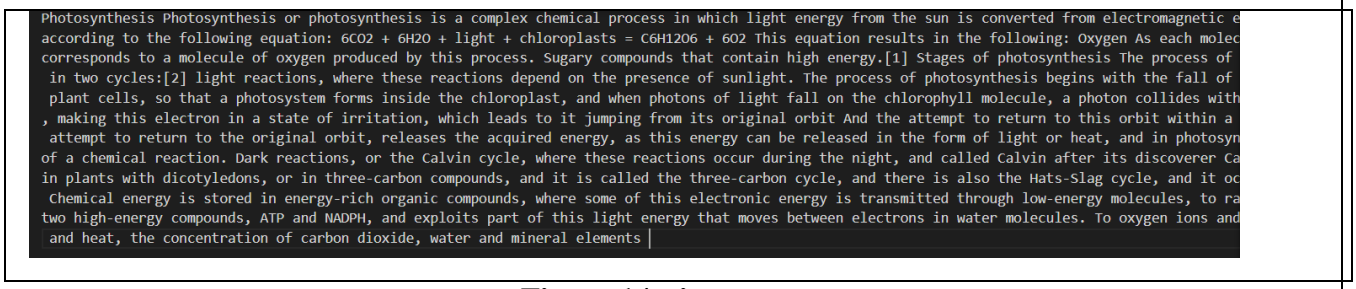

Figure 14 : input

8.2 Output

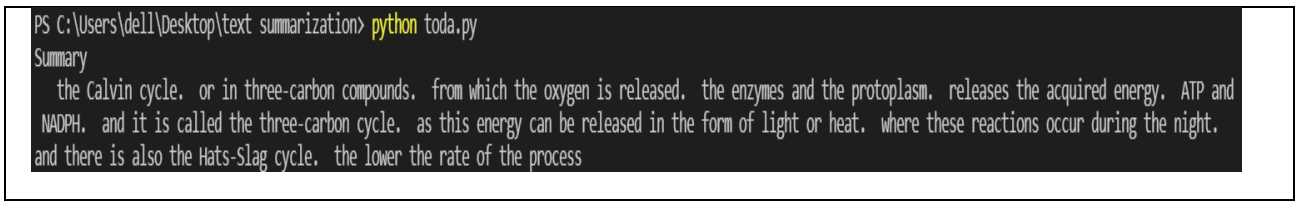

Figure 15 : output

9Conclusion

Dans ce chapitre nous avons commencé par la présentation de la plate-forme de développement de notre système de résumé d'un article scientifique, en l'occurrence le **Python**. Nous avons en deuxième lieu présenté l'architecture globale de notre système et par la suite nous avons détaillé chaque module de système.

Finalement nous avons présenté quelques Captures d'écran de notre code source qui présentent les principales fonctionnalités de notre système.

Conclusion générale et perspectives

La notion de résumé automatique devient un des grands thèmes du Traitement Automatique des Langues naturelles (TALN). Plutôt que de diffuser les documents entiers, n'est-il pas préférable de diffuser seulement les résumés qui contiendraient les informations vraiment pertinentes ? En effet, il est plus facile de lire quelques lignes ou quelques pages pour s'apercevoir qu'aucune information nouvelle ne s'y trouve. Un document textuel devra donc être maintenant géré en même temps que son résumé qui sera, par ailleurs, un des moyens d'accès au contenu du document.

Notre travail s'inscrit dans le cadre de résumé automatique d'un article scientifique, où nous avons appliqué les techniques de résumé automatique de texte par extraction avec l'utilisation de l'architecture particulière d'un article scientifique afin de pouvoir résumer la contribution des auteurs de cet article automatiquement.

Vue la contrainte du temps quelques taches de ce travail sont réalisées manuellement (extraction de titre, mots clé, ….), et nous n'avons pas évalué notre système pour juger que notre travail est performent au pas, donc notre travail reste loin d'être validé actuellement.

En perspective à ce travail, nous projetons d'automatiser tous les taches de notre système, d'un part, et d'une autre part valider le système par des outils d'évaluation de résumé automatique de texte comme par exemple la mesure ROUGE ou même l'évaluation de résumé par des experts.

Bibliographies:

[0] Data Analytics: Practical Guide to Leveraging the Power of Algorithms, Data Science, Data Mining, Statistics, Big Data, and Predictive Analysis to Improve Business, Work, and Life Arthur Zhang

[1] Nicolas TURENNE« Apprentissage statistique pour l‟extraction de concepts à partir de textes. Application au filtrage d‟informations textuelles. » 2004. https://tel.archives-ouvertes.fr/tel-00006210/document

[2] Thomas Heitz. Modélisation du prétraitement des textes. JADT'06 (International Conference on Statistical Analysis of Textual Data), 2006, Besançon, France, pp.499-506. ffinria-00119608

[3] Universidad de la República – Facultad de Ingeniería INCO J. Herrera y Reissig 565, Montevideo, Uruguay jcouto@fing.edu.uy ** MoDyCo, UMR7114, CNRS Université Paris X 200, avenue de la République, France Jean-Luc.Minel@u-paris10.fr

[4] Haïfa Zargayouna1 et Sylvie Salotti2 1 LIMSI/CNRS, Université Paris 11 haifa.zargayouna@limsi.fr 2 LIPN - CNRS UMR 7030, Université Paris 13 sylvie.salotti@lipn.univ-paris13.fr

[11] Khadija EL GAJOUI, Fadoua ATAA ALLAH,«Vers un système de reconnaissance optique descaractères dans des documents multilingues : Français-Amazighe» 2013.

[12] ARIES Abdelkrime, Mémoire pour l'obtention du Magister de l' Ecole Nationale Supérieure d'Informatique (ESI), Thème : Résumé automatique des textes, le 26/06/2013.

[13] Maâli Mnasri, (1) CEA, LIST, Laboratoire Vision et Ingénierie des Contenus, Gif-surYvette, F-91191 France. (2) Univ. Paris Sud, Orsay, France.

[14] Role of Biodiversity Conservation in the Transition to Rural Sustainability (Science and Technology Policy)

Stephen S. Light, Poland) NATO Advanced Research Workshop on the Role of Biodiversity Conservation i

[15] Joris-karl Huysmans Auteurs francais 1848 – 1907

[16] Peul Verlaine Auteurs francais 1844 – 1896

[17] Manual for the Design of Reinforced Concrete Building Structure 1948

[18] Histories of the Immediate Present: Inventing Architectural Modernism

Anthony Vidler, Peter Eisenman 2008 **.**

[19] The Secret Places of the Heart.txt

[20][Pincemin,2001] Pincemin B., « Résoudre la surcharge informationnelle sans décontextualiser », in Filtrage et résumé informatique de l'information sur les réseaux, 3e Congrès du Chapitre français de l'ISKO, 5-6 juillet 2001, sous la dir. de S. Chaudiron et C. Fluhr, Université de Paris X, 2001, pp. 149-158.

[21] Bossard A., Rodrigues C. (2011). Combining a multi-document update summarization system – CBSEAS – with a genetic algorithm. In I. Hatzilygeroudis, J. Prentzas (Eds.), Combinations of intelligent methods and applications. Springer.

[22] Erkan G., Radev D. R. (2004). Lexrank: Graph-based centrality as salience in text summarization. Journal of Artificial Intelligence Research (JAIR), vol. 22.

[23]Edmundson H. P., Wyllys R. E. (1961). Automatic abstracting and indexing-survey and recommendations. Commun. ACM, vol. 4, n° 5, p. 226–234.

[24] Flesch R. (1948). A new readability yardstick. Journal of applied psychology, vol. 32, n° 3, p. 221–233.

[26] (Brandow et al., 1995) R. Brandow, K. Mitze, et L. F. Rau, 1995. Automatic condensation of electronic publications by sentence selection. Information Processing and Management : an International Journal 31(5), 675–685.

[27] (Mann et Thompson, 1988) W. C. Mann et S. A. Thompson, 1988. Rhetorical Structure Theory : A Theory of Text Organization. Text 8(3), 243–281.

[28] (Ono et al., 1994) K. Ono, K. Sumita, et S. Miike, 1994. Abstract generation based on rhetorical structure extraction. Dans les actes de 15th conference on Computational linguistics, Volume 1, 344–348. Association for Computational Linguistics Morristown, NJ, USA.

[29] (Marcu, 1997) D. Marcu, 1997. From discourse structures to text summaries. Dans les actes de ACL Workshop on Intelligent Scalable Text Summarization, 82–88.

[30] (Kleinberg, 1999) J. M. Kleinberg, 1999. Authoritative sources in a hyperlinked environment. [

[31] (Fernández et al., 2008a) S. Fernández, E. SanJuan, et J.-M. Torres-Moreno, 2008a. Enertex : un système basé sur l'énergie textuelle. Dans les actes de Traitement Automatique des Langues Naturelles (TALN), Avignon, France, 10 pages

[32] Publié le 26 février 2020 par Chloé Leterme. Mis à jour le 28 avril 2020.

[33](Luhn, 1958) H. P. Luhn, 1958. The Automatic Creation of Literature Abstracts. IBM Journal of Research and Development 2(2), 159–165.

[34] (Edmundson, 1969) H. P. Edmundson, 1969. New Methods in Automatic Extracting. Journal of the ACM (JACM) 16(2), 264–285.

[35] (Pollock et Zamora, 1975) J. J. Pollock et A. Zamora, 1975. Automatic Abstracting Research at Chemical Abstracts Service. Journal of Chemical Information and Computer Sciences 15(4), 226–232

[36] (Brandow et al., 1995) R. Brandow, K. Mitze, et L. F. Rau, 1995. Automatic condensation of electronic publications by sentence selection. Information Processing and Management : an International Journal 31(5), 675–685.

[37] (Spärck Jones, 1972) K. Spärck Jones, 1972. A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation 28(1), 11–21.

[38] (Mann et Thompson, 1988) W. C. Mann et S. A. Thompson, 1988. Rhetorical Structure Theory : A Theory of Text Organization. Text 8(3), 243–281.

[39](Ono et al., 1994) K. Ono, K. Sumita, et S. Miike, 1994. Abstract generation based on rhetorical structure extraction. Dans les actes de 15th conference on Computational linguistics, Volume 1, 344–348. Association for Computational Linguistics Morristown, NJ, USA.

[40] (Marcu, 1997) D. Marcu, 1997. From discourse structures to text summaries. Dans les actes de ACL Workshop on Intelligent Scalable Text Summarization, 82–88.

[41](Ono et al., 1994) K. Ono, K. Sumita, et S. Miike, 1994. Abstract generation based on rhetorical structure extraction. Dans les actes de 15th conference on Computational linguistics, Volume 1, 344–348. Association for Computational Linguistics Morristown, NJ, USA.

[42] (Kleinberg, 1999) J. M. Kleinberg, 1999. Authoritative sources in a hyperlinked environment. Journal of the ACM (JACM) 46(5), 604–632.

[43](Brin et Page, 1998) S. Brin et L. Page, 1998. The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems 30(1-7), 107–117.

[44] (Erkan et Radev, 2004a) G. Erkan et D. R. Radev, 2004a. LexRank : Graph-based Lexical Centrality as Salience in Text Summarization. Journal of Artificial Intelligence Research 22(2004), 457–479.

[45] (Mihalcea, 2004) R. Mihalcea, 2004. Graph-based ranking algorithms for sentence extraction, applied to text summarization. Dans les actes de ACL 2004 on Interactive poster and demonstration sessions, 181–184. Association for Computational Linguistics Morristown, NJ, USA.

[46](Fernández et al., 2008a) S. Fernández, E. SanJuan, et J.-M. Torres-Moreno, 2008a. Enertex : un système basé sur l'énergie textuelle. Dans les actes de Traitement Automatique des Langues Naturelles (TALN), Avignon, France, 10 pages.

[47] Vahed Qazvinian, Dragomir R. Radev, Saif Mohammad, Bonnie J. Dorr, David M. Zajic, M. Whidby, and T. Moon. Generating extractive summaries of scientific paradigms. JAIR, 46 :165–201, 2013.

[48] Kenjl Ono, Kazuo Sumlta et Seijl Miike. Abstract Generation based on Rhetorical Structure Extraction. In Proceedings of the International Conference on Computational Linguistics (COLING'94), pages 344–348, Kyoto, Japon, 1994.

[49] Daniel Marcu. The theory and practice of discourse parsing and summarization. MIT Press, Cambridge, MA, USA, 2000

[50]Juan-Manuel Torres-Moreno. Résumé automatique de documents : une approche statistique. hermes science, 2011.

Webographie :

[8]http://ldelafosse.pagesperso-orange.fr/Glossaire/Tal.htm visité le 01/12/2016

[9] http://fis.ucalgary.ca/Brian/ecrire/e-resume.htm visité le 19/11/2016.

[10] https://www.cairn.info/revue-sante-publique-2006-4-page-533.htm?contenu=plan

[21]http://www.ims.uni-stuttgart.de/~schmid/french-tagset.html.

[25]Les campagnes TAC, pour Text Analysis Conference, sont organisées annuellement par le National Institute of Science and Technology. : http://www.nist.gov/tac**.**