Université de Lorraine

Une Approche Statistique de Maximisation des Traits pour le Résumé de Documents Scientifiques

Rapport

présenté et soutenu publiquement le 31 août 2016

pour l'obtention du

Master de l'Université de Lorraine

(Traitement Automatique des Langues)

par

Hazem AL SAIED

Composition du jury

Rapporteurs :	Yves LAPRIE Maxime AMBLARD
Encadrants :	Jean-Charles LAMIREL Nicolas DUGUÉ

Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503

Mis en page avec la classe thesul.

Abstract

The rapid growth of the amount of information has prompted the emergence of many IT domains including automatic summarization (AS). These systems are used to extract the most relevant information from huge amounts of data.

In this report, we are working on an AS system that focuses on multiple scientific articles summarization. These summaries could help in organizing the huge amount of scientific production and could provide scientific digests attached to the same subject.

In our system, we experiment the features maximization (FM) in this context. this statistical method initially designed for machine learning provides a language-agnostic and non-parametric approach of AS. We integrate this method with traditional structures of AS systems and with a graph-based model exploiting the spread activation algorithm.

In sum, this thesis introduce a new approach for AS based on FM and it aims to evaluate the performance of this method in the task of producing extractive summaries, either generic summaries or query-focused ones.

Keywords: Automatic Summarization, Retrieval Systems, Features Maximization, Cosine Similarity, Graph-Based Model, Spreading Activation

Résumé

La croissance rapide de la quantité d'information a motivé l'émergence de nombreux domaines informatiques, dont les systèmes de résumé automatique (RA). Ces systèmes sont utilisés pour extraire les informations les plus pertinentes de l'énorme quantité de données.

Dans ce mémoire, nous travaillons sur un système de RA qui se concentre sur les résumés des multiples articles scientifiques. Ces résumés permettent de surmonter la redondance des articles de la communauté scientifique et de fournir des synthèses de documents attachées au même sujet.

Nous utilisons dans notre système la maximisation des traits (MT) qui incarne une nouvelle approche langage-agnostique et non paramétrique de RA. Elle nous permet de générer des résumés de communauté de documents multiples en utilisant les structures traditionnelles du RA et un modèle à base de graphes en utilisant l'algorithme de la propagation d'activation (PA).

En somme, ce mémoire a pour objectif de combiner l'approche de la MT avec les systèmes de RA et d'évaluer les résumés extractifs produits, que ce soient des résumés génériques ou des résumés se basant sur une requête.

Mots-clés: Résumé Automatique, Systèmes d'Information, Maximisation des Traits, Similarité Cosinus, Modèle à Base de Graphe, Propagation d'Activation.

Remerciements

Je tiens à remercier toutes les personnes qui ont contribué au succès de mon stage et qui m'ont aidé lors de la rédaction de ce rapport.

Tout d'abord, j'adresse mes remerciements à mon professeur, M. Maxime Amblard de l'Université de Lorraine qui m'a beaucoup aidé dans ma recherche de stage.

Je tiens à remercier vivement mes maîtres de stage, M. Jean-Charles Lamirel et M. Nicolas Dugué pour leur accueil, le temps passé ensemble et le partage de leur expertise au quotidien. Grâce à leur confiance, j'ai pu m'accomplir totalement dans mes missions. Ils furent d'une aide précieuse dans les moments les plus délicats.

Je remercie également toute l'équipe du Synalp pour leur accueil, et mon ami, Michel Daoud, qui m'a aidé dans la rédaction de ce rapport.

Enfin, je tiens à remercier toutes les personnes qui m'ont conseillées et relues lors de la rédaction de ce rapport de stage : mon amour, Yafa, ma famille et mes amis.

Tables des matières

Liste des acronymes

LC Linguistique Computationnelle
MT Maximisation des Traits
MVS Machine à Vecteurs Supports
PA Propagation d'Activation
RA Résumé Automatique
ROUGE Analyse basée sur le rappel pour l'évaluation fine
TAL Traitement Automatique des Langues
FT*FID Fréquence du Terme * Fréquence Inverse des Documents
UCR Unités de Contenu de Résumé

Liste des tableaux

Chapitre 1

Introduction

«Ils me racontent que nous vivons dans l'ère de l'information. Pourtant, rien n'appartient aux informations dont j'ai besoin ou ne s'approche de ce que je voudrais savoir !» [1]

1.1 Motivations

Depuis plusieurs décennies, nous ajoutons chaque jour des millions de pages au web. La croissance rapide de ce dernier conduit à une autre, celle de l'information en ligne, notamment textuelle. Ce fait motive l'émergence de nombreux domaines informatiques, dont les systèmes d'information et plusieurs sous-domaines du traitement automatique des langues (TAL).

Ces solutions informatiques sont utilisées pour gérer et traiter l'énorme quantité de données. Une de ces solutions est les systèmes de résumé automatique (RA) qui permettent d'extraire les informations les plus pertinentes. De plus, ils facilitent la compréhension des données textuelles et réduisent leur nombre de dimensions. L'objectif de ces systèmes est donc de produire des fragments de texte contenant les informations les plus importantes afin de gérer les documents de ressources plus facilement [Lloret, 2015].

Dans ce mémoire, nous travaillons sur un système de RA qui se concentre sur les résumés de multiples articles scientifiques. Ces résumés permettent de surmonter la redondance des articles de la même communauté scientifique et de fournir des synthèses de documents attachées au même sujet. Ce système peut également être combiné avec les systèmes d'information et de questions-réponses afin d'augmenter leur efficacité et la pertinence de leurs résultats.

Nous introduisons un nouveau système basé sur la maximisation des traits (MT), une méthode statistique d'apprentissage automatique qui sert à choisir les traits les plus importants dans les données hétérogènes.

Pour l'adapter dans le cadre du problème du RA, nous considérons que les articles scientifiques sont des données textuelles semi-structurées. Chaque article est composé de plusieurs sections que nous considérons comme des classes. Les dizaines de mots qui composent ces sections jouent ainsi le rôle des traits. La MT se sert, par conséquent, de cette représentation pour calculer la F-mesure des traits qui détermine leur importance dans leur classe. Les équations pour calculer ces mesures sont les suivantes :

$$FR_c(f) = \frac{W_c^f}{\sum_{c' \epsilon C} W_{c'}^f}$$

$$FP_c(f) = \frac{W_c^f}{\sum_{f' \epsilon F_c} W_c^{f'}}$$

1. Citation traduite de The Kingdom of Ohio, Matthew Flaming, December 2009 by Penguin Adult HC/TR.

$$FF_c(f) = 2\left(\frac{FR_c(f) \cdot FP_c(f)}{FR_c(f) + FP_c(f)}\right)$$

Où W_c^f est l'occurrence du mot f dans la classe (section) c, C est l'ensemble des classes (sections), et F_c vaut pour l'ensemble des traits dans la classe c. Si un trait possède un rappel élevé, alors il est saillant, typique de cette classe. Si il a un score de précision élevé, il est apte à décrire ou à représenter sa classe.

Outre sa performance compétitive pour l'apprentissage sur les données textuelles (supervisé [Lamirel et al., 2015a] et non supervisé [Lamirel et al., 2015b]), la définition purement statistique de la MT rend cette procédure **langage-agnostique**, ce qui signifie qu'elle est totalement indépendante du langage des documents d'entrée. Elle nous permet également d'avoir un système de résumé **non paramétrique** comme nous le verrons Section 3.1. Par ailleurs, la MT a prouvé sa remarquable performance dans les procédures de sélection de variables : cette métrique permet donc de synthétiser l'information de façon efficace. La MT n'a également pas besoin de corpus supplémentaires à celui étudié pour fonctionner contrairement aux approches usuelles basées sur le FT*FID comme nous le verrons.

Enfin, sa flexibilité nous promet de nombreuses possibilités de combinaisons, notamment dans les systèmes de RA et les systèmes d'information.

1.2 À propos du projet

Ce mémoire a pour objectif de combiner l'approche de la MT avec les systèmes de RA afin de produire des résumés extractifs**,** que ce soient des résumés génériques ou des résumés se basant sur une requête [voir sec. 2.1].

Le faible nombre des ressources linguistiques annotées est l'une des difficultés principales dans l'évolution du domaine du RA. Ainsi, la publication du corpus CL-SciSumm [2] [SCI-Summ, 2016] (proposé dans le cadre d'un challenge sur le résumé automatique des articles scientifiques dans le domaine de la linguistique computationnelle) a motivé ce travail de mémoire. En effet, ce corpus est composé de publications scientifiques semi-structurées, particulièrement adaptées pour l'utilisation de la MT.

1.3 Problématiques

Nous nous intéressons à la seconde tâche du challenge indiqué ci-dessus, celle de générer un résumé de communauté qui contient les phrases citées par des articles scientifiques donnés. Il s'agit donc de générer un résumé se basant sur une requête, un des problèmes classiques du domaine du RA.

Ce corpus nous permet de bénéficier de documents semi-structurés, de résultats de références associés à des approches classiques ou innovantes, ce qui nous permet d'établir un «état de l'art» et de proposer une comparaison avec notre méthode. La mise en œuvre de cette expérimentation consiste à combiner la MT avec la métrique de similarité cosinus. En effet, l'application de la similarité cosinus avec la méthode FT*FID nous encourage à exploiter cette approche en raison de sa similarité avec notre méthode [Tata and Patel, 2007].

En somme, nous évaluons l'importance des mots du document et extrayons les scores associés en utilisant la MT. Puis, nous générons les modèles vectoriels des phrases de l'article en utilisant les résultats de la MT. Nous exploitons ensuite cette représentation pour proposer une méthode de sélection des phrases exploitant la mesure de similarité cosinus. Cela nous permet ainsi de générer le résumé final, celui de communauté.

Avant de commencer cette expérimentation, nous voulons expérimenter l'utilisation de la MT dans le contexte du résumé d'un seul document. L'objectif principal de cette expérience est de créer un système simplifié, de l'évaluer sur une tâche standard et de comparer sa performance avec d'autres systèmes

2. Computational Linguistics Scientific Document Summarization

avant de le combiner avec des systèmes plus complexes. Dans cette première expérimentation, les mots seront classés selon les poids produits par la MT en supposant que l'article se compose de plusieurs sections, appelées classes. Chaque mot est un trait associé à une classe. Selon la MT ils entrent dans une compétition se basant sur leurs capacités de discrimination et de généralisation pour les classes, dans le but de prouver leur importance. Ensuite, le poids des phrases est produit en agrégeant les poids des mots. Enfin, nous sélectionnons les phrases qui font partie du résumé selon le classement produit à l'étape précédente.

Pour évaluer les résultats de notre méthode en l'adaptant aux tâches proposées par les corpus de référence, nous avons également travaillé sur une troisième expérimentation destinée à combiner l'approche de la MT avec un nouveau système de RA. La nouvelle structure du système se rapproche de la tâche de génération de résumés de communauté en utilisant les concepts de systèmes d'information. De plus, nous utilisons les modèles à base de graphes afin de représenter les mots des documents. La nouveauté de cette méthode est sa combinaison avec l'algorithme de propagation d'activation à partir des scores obtenus via la MT afin de favoriser une partie du graphe des mots, partie qui contient les mots attachés aux requêtes lors de la recherche des phrases proches d'elles.

1.4 Organisation du rapport

Dans la suite du document, nous allons présenter l'état de l'art des systèmes de RA et l'évaluation de leur performance au chapitre 2. Le chapitre 3 décrit l'approche de la MT que nous proposons et son combanaison avec un système classique de RA pour un seul document. Le chapitre 4 présente une autre application de notre approche dans un système classique de RA sur les documents multiples. Le chapitre 5 propose une troisième application de la MT mais dans un nouveau système de RA se basant sur les concepts de la théorie des graphes et sur un algorithme de propagation d'activation. Enfin, la conclusion et les recommandations concernant les travaux futurs sont présentés dans le chapitre 6.

Chapitre 2

Revue de la Littérature

Nous allons tout d'abord présenter les différents types de résumés existants et leurs objectifs. Ensuite, nous décrirons les approches actives introduites au préalable. Puis, nous détaillerons les méthodes de résumés extractives de façon plus exhaustive. En effet, pour situer ce travail, nous proposons une méthode de résumé extractif.

2.1 Types de résumés

Les résumés sont catégorisés en de nombreux types. Ils sont classifiés selon leur nature textuelle, le nombre de documents à résumer, le contexte dans lequel le résumé se produit et leur utilisation applicative.

Les Résumés Extractifs se composent des phrases importantes sélectionnées par le système de RA sans aucune modification sur leur contenu. Ce type de résumé est très commun dans le domaine du RA [Nenkova et al., 2011].

En revanche, quand le système ou l'homme se servent de leurs connaissances linguistiques et des relations sémantiques du texte, reformulant les phrases, les termes et les mots, il s'agit des **Résumés Abstractifs ou Abstraits** du même type que ceux que l'on trouve en tête de chaque article scientifique**.**

Les systèmes de RA ont tout d'abord commencé à s'appliquer aux articles scientifiques, aux rapports d'actualités, etc., afin de produire **le Résumé d'un Seul Document.** Le développement du web et le besoin de traiter son vaste contenu de textes ont cependant motivé l'émergence **du Résumé de Documents Multiples** [voir figure.2.1]. Ces résumés permettent de surmonter la redondance et la quantité de données issues du web en fournissant des synthèses de documents attachés au même sujet. Les systèmes d'information et les systèmes de questions-réponses se servent de ces résumés pour augmenter leur efficacité et la pertinence de leurs résultats [Nenkova et al., 2011].

Les Résumés Indicatifs essaient de mettre en avant les points les plus importants du texte, le style d'écriture, la longueur du texte. Tandis que **les Résumés Informatifs** ont pour objectif de remplacer le texte par le résumé en se concentrant sur les informations importantes du texte d'entrée.

Il existe trois types de résumés selon leur format. **Le Résumé de Mots-Clés** ne contient que les mots et les termes les plus pertinents du texte d'entrée. Par contre, **le Résumé de Manchettes** essaie de faire la synthèse du document d'entrée en une ou plusieurs manchettes. Enfin, supposé compréhensible par tout le monde, **le Résumé Générique** essaie de créer un résumé bien structuré et lisible.

Les systèmes de **Résumé se Basant sur une Requête** classifient les phrases et leur importance en considérant le contenu d'une requête d'entrée. Ce type de résumés est couramment combiné avec les systèmes d'information. De plus, il utilise les concepts de ces systèmes dans l'objectif de trouver la meilleure correspondance entre la requête et le contenu des documents d'entrée [Varadarajan and Hristidis, 2006].

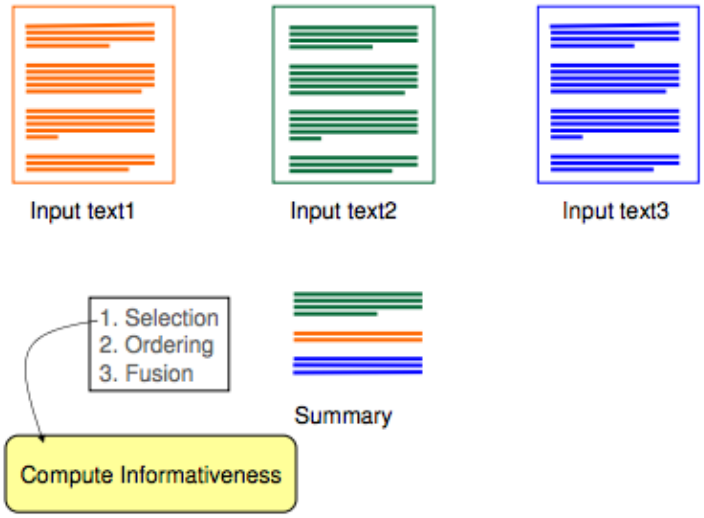

FIGURE 2.1 – Le résumé automatique de documents multiple
Source : Ref. [Nenkova et al., 2011]

Le Résumé de Type Mise à Jour[3] est une sorte de résumé de documents multiples, considérant ceux qui évoluent au fil du temps. Dans ce type, les documents sont classifiés chronologiquement dans plusieurs groupes, et le résumé de chaque groupe de documents est harmonisé avec les résumés des groupes précédents. Par ailleurs, les groupes permettent d'améliorer la cohérence de la collection des résumés et d'augmenter leur informativité et leur consistance.

2.2 Approches pour résumer

Cette section passe en revue les méthodes les plus connues du RA. Nous allons discuter des expérimentations et des méthodes fondamentales, notamment celles qui se concentrent sur les résumés extractifs, à base d'apprentissage automatique, et pour la production de résumés non extractifs.

2.2.1 Émergence du résumé extractif

Afin d'écrire un résumé extractif concis et aisé, les systèmes de RA doivent pouvoir répondre à cette question : quelles sont les phrases les plus pertinentes dans le texte d'entrée ?

Dans son ouvrage, Luhn [Luhn, 1958] a donné naissance à plusieurs idées qui forment aujourd'hui les concepts de base des systèmes de RA. Selon lui, certains mots peuvent décrire un document et leur nature sémantique permet de localiser le contenu important du document.

Dans cet ouvrage, Luhn s'est servi de la notion de fréquence pour sélectionner ces mots et les classer. Intuitivement, les phrases contenant ces mots sont les plus pertinentes et les candidates les plus éligibles à la construction du résumé.

Dans ses expérimentations, Luhn faisait face au fait que les articles, les prépositions et les conjonctions sont les mots les plus fréquents dans n'importe quel document. Il a donc défini une liste de mots à éviter

3. Update Summary

selon la mesure de la fréquence. Cette liste s'appelle liste des «mots vides» ou «stop words» et constitue aujourd'hui une notion incontournable dans ce domaine.

D'autres classes de mots qui limitent l'efficacité de sa méthode sont les mots qui se trouvent souvent dans tous les documents d'un domaine précis. Par exemple, le mot «verbe» dans le domaine de la linguistique ne peut pas être considéré comme un mot-clé.

Dans l'objectif de surmonter ce problème, Luhn proposa de construire un corpus de documents du même domaine que le document d'entrée. Ce corpus est ensuite utilisé pour comparer la fréquence de chaque mot dans le corpus avec sa fréquence dans le document d'entrée.

Toutefois, nous pouvons exprimer un concept en utilisant de nombreux mots. Par exemple, les mots «result, resultant, outcome, termination» se trouvent dans le même *synset* de WordNet [4]. De plus, le même concept peut apparaître sous plusieurs formes selon sa position lexicale ou ses extensions morphologiques. Afin de réduire l'effet de ces désavantages, Luhn développa une version primitive de l'analyse morphologique.

2.2.2 Approches non extractives

Le domaine du RA se sert de ces approches pour améliorer la qualité des données d'entrée et celle du résumé lui-même en utilisant les technologies de la génération de texte. Le travail de Paice [Paice, 1980] forme la première tentative destinée à améliorer la lisibilité du résumé. Paice a utilisé les techniques de la génération de texte afin de reformuler les phrases sélectionnées. De plus, il a proposé de mettre en place un système pour résoudre les coréférences des phrases [5].

Les techniques les plus connues de ce domaine ont un des trois objectifs suivants :

1. **Donner un Ordre aux Phrases :** ces approches sont combinées avec les systèmes de RA de documents multiples par exemple. Elles mettent en ordre les phrases obtenues par la procédure de sélection afin de fournir le résumé le plus cohérent ;
2. **Reformuler les Phrases :** ces approches proposent de modifier les phrases sélectionnées en utilisant des parties de la phrase selon le contexte [Mani et al., 1999] ;
3. **Fusionner les Phrases** qui contiennent des parties d'information partagées et d'autres parties indépendantes dans l'objectif de garder les parties partagées et de diminuer la redondance du résumé.

2.3 Extraction des phrases

Dans cette section, nous décrivons en détail la partie la plus importante de l'algorithme de résumé extractif, l'extraction des phrases. Nous discutons ainsi plusieurs méthodes issues de différentes approches : guidées par les données, basées sur l'apprentissage automatique ou celles qui s'adaptent à une requête donnée.

4. WordNet est une base de données lexicale développée par l'université de Princeton depuis une vingtaine d'années. Son but est de répertorier, classifier et mettre en relation de diverses manières le contenu sémantique et lexical de la langue. voir <*https://wordnet.princeton.edu*>. [page consultée le 29 juillet 2016]

5. La résolution des coréférences est un problème bien connu en extraction d'information. L'objectif consiste à déterminer si deux expressions dans un texte réfèrent ou pas à la même entité du domaine de discours. Par exemple, le pronom il désigne Hazem : Hazem est compétant en linguistique computationnelle et il est à l'aise avec ce domaine [Zweigenbaum et al., 2012]
.

2.3.1 Méthodes non supervisées et guidées par les données

Ce genre d'extraction n'a pas besoin de données annotées par l'homme. Ces méthodes extraient les données de façon statistique sans aucun besoin de modèle, de ressource de connaissance, de connaissance linguistique ou d'interprétation de données [Nenkova et al., 2011].

La notion de **Probabilité des Mots**, par exemple, est une application du travail de Luhn. Elle considère que la fréquence d'un mot reflète son importance dans le document.

Autrement dit, la probabilité $P(m)$ d'un mot m est, tout simplement, sa fréquence $F(m)$ dans le document divisée par N, la somme des fréquences des mots de ce document.

$$P(m) = F(m)/N$$

Alors, la probabilité du résumé R peut être calculée par cette équation :

$$P[R] = \frac{N!}{n_1!...n_r!} P(w_1)^{n_1}.....P(w_r)^{n_r}$$

Où $N = n_1 + ... + n_r$ tel que n_i est la fréquence du mot w_i dans le résumé et $P(w_i)$ est la probabilité du mot w_i.

Le système **SUMBasic** applique cette notion pour sa mise en œuvre [Vanderwende et al., 2007]. L'importance de chaque phrase Ph_j est, par conséquent, la moyenne de la probabilité de ses mots :

$$Poids(Ph_j) = \frac{\sum_{w_i \epsilon Ph_j} P(w_i)}{|\{w_i \mid w_i \epsilon Ph_j\}|}$$

Le résumé est produit par une sélection gloutonne des phrases ayant les meilleurs scores. Après la sélection d'une phrase, le système minimise la probabilité de ses mots les plus importants et recalcule la probabilité des autres phrases. L'objectif de cette procédure est de diminuer la probabilité des phrases similaires et d'augmenter l'informativité du résumé.

Un des problèmes principaux de cette méthode est qu'elle donne des poids importants aux mots fréquents dans le document, tandis qu'une bonne partie de ces mots est fréquente dans n'importe quel document du fait de sa nature morphologique ou sa généralité au niveau sémantique. Afin de surmonter ce problème, l'approche **Fréquence du Terme * Fréquence Inverse des Documents** (FT*FID) propose de marginaliser le poids de ce genre de mots en calculant le poids à partir d'un corpus externe comme suit :

$$FT * FID = P(w) * log \frac{N}{F(w)}$$

Où $F(w)$ est la fréquence du mot w dans un corpus composé de N documents.

Contrairement à la FT*FID, la méthode **Test de Seuil de Vraisemblance pour les Signatures de Sujet** [6] propose un seuil pour classifier les mots dans deux classes : pertinents et descriptifs, «signatures de sujet», et non pertinents [Lin and Hovy, 2000].

L'approche se sert d'un corpus externe comme la FT*FID. Elle calcule le poids de chaque mot dans le document d'entrée et dans le corpus. Si le poids dans le document est supérieur à celui du corpus, le mot est annoté comme signature de sujet.

6. Log-likelihood ratio test for topic signature.

$$f(w) = \begin{cases} P(w \mid D) = P(w \mid C) & \text{alors } w \text{ n'est pas une signature de sujet} \\ P(w|D) = pd \text{ and } P(w|C) = pc \text{ and } pd > pc & \text{alors } w \text{ est une signature de sujet} \end{cases}$$

où $P(w \mid D)$ est la distribution du mot w dans le document D et $P(w \mid C)$ est la distribution du w dans le corpus C.

Ici, le nombre des signatures de sujet détermine l'importance de la phrase au lieu des valeurs FT*FIDs.

La **Classification des Phrases** est une méthode pour le résumé de documents multiples à base de phrases. Chaque classe contient les phrases similaires et lors de la génération du résumé, une phrase de chaque classe est sélectionnée. Diminuer la redondance est l'avantage principal de cette méthode.

Étant donné que chaque phrase peut exprimer plusieurs aspects, c'est-à-dire, peut être classifiée dans plusieurs classes, **les Modèles à Base de Graphe** proposent une approche assez similaire à la méthode précédente, mais plus flexible [Erkan and Radev, 2004].

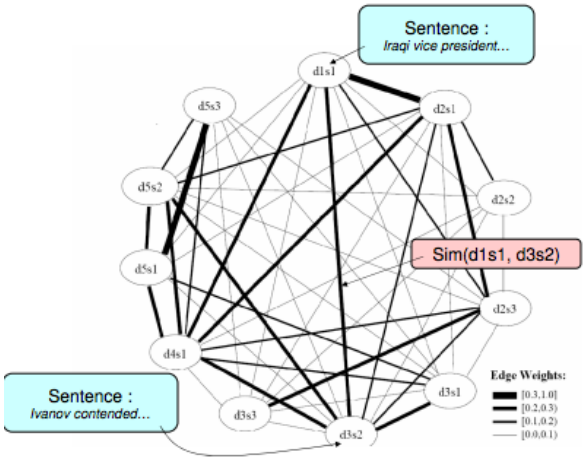

FIGURE 2.2 – Un modèle à base de graphe issu de [Nenkova et al., 2011]

Dans ces modèles, les phrases sont les nœuds d'un graphe non orienté et l'arête entre deux phrases est la valeur de similarité de ces phrases [Figure. 2.2]. Erkan et Radev [Erkan and Radev, 2004] proposent d'utiliser la similarité cosinus pour calculer cette valeur. Par ailleurs, ils utilisent les poids générés par FT*FID lors de la vectorisation des phrases.

D'autres modèles proposent de supprimer l'arête entre deux phrases si le score de similarité ne surmonte pas un seuil prédéfini. Avec ce genre de représentation flexible, l'importance d'une phrase peut être mesurée en utilisant des algorithmes génériques de la théorie des graphes comme l'algorithme PageRank.

Ces modèles combinent la méthode précédente avec celles à base de fréquence [Qazvinian and Radev, 2008]. De plus, ces méthodes obtiennent d'excellents résultats pour le résumé d'un seul document et celui de documents multiples [Erkan and Radev, 2004].

2.3.2 Méthodes à base d'apprentissage automatique

L'apprentissage automatique est une composition de la conception, de l'analyse, du développement et de l'implémentation de méthodes permettant à une machine d'évoluer par un processus systématique. Il a deux catégories principales, l'apprentissage statistique et celui symbolique. Les algorithmes utilisés permettent à un système piloté par ordinateur d'adapter ses analyses et ses comportements en réponse,

en se fondant sur l'analyse de données empiriques provenant d'une ou plusieurs sources de données [MParUuez and Salgado, 2000].

Les systèmes de RA se basant sur l'apprentissage supervisé voient le jour avec le travail d'Edmundson [Edmundson, 1969]. Cette approche se base sur la structure proposée par Luhn. En revanche, ses expérimentations se servent des traits des phrases et des mots pour déterminer leur importance dans le document d'entrée.

Dans ses expérimentations, Edmundson utilise des traits comme la longueur de la phrase, sa position, le nombre de ses mots, la fréquence de chaque mot, ou encore sa fréquence dans les manchettes et les titres du document. Les résultats de ces expérimentations ont montré que la fréquence des mots est le trait le moins important pour cette approche.

En TAL, les corpus forment les sources principales des données. Chaque corpus est divisé en trois parties, une pour l'étape d'entraînement, une autre pour le développement et la dernière pour le test. Le modèle est adaptatif, de façon à prendre en compte l'évolution du corpus pour lequel les comportements en réponse ont été validés. Ceci permet d'autoaméliorer le système d'analyse ou de réponse [Roberts, 2003].

Dans le domaine de résumé automatique, deux technologies de l'apprentissage automatique ont été principalement expérimentées : le **Classificateur Naïf Bayésien** et le **Modèle de Markov Caché**.

En ce qui concerne la première méthode, Kupiec et Pedersen [Kupiec et al., 1995] ont proposé plusieurs traits comme le nombre des groupes nominaux fixés, la position de la phrase dans les paragraphes, la position du paragraphe lui-même, et la fréquence des mots.

De plus, ils ont ajouté d'autres traits comme les mots commençant par une majuscule pour déterminer les noms propres et les acronymes, ou encore la longueur de la phrase.

À la suite de la construction d'un corpus d'articles techniques annotés par ces traits, Kupiec et Pedersen ont appliqué le **Classificateur Naïf Bayésien** sur de nombreuses variations de traits. La meilleure variation était celle qui se compose des trois premiers traits indiqués ci-dessus.

Conroy et O'Leary utilisent le **Modèle de Markov Caché** pour le RA [Conroy and O'leary, 2001] selon trois traits : la position de la phrase, le nombre de ses termes et leurs probabilités. D'autres versions proposent des traits différents, mais les résultats ne sont pas satisfaisants en comparaison de ceux obtenus avec les méthodes à base de graphe ou à base de fréquence. D'autre part, le manque de corpus bien annotés et destinés à la tâche de résumé automatique à base d'apprentissage automatique est un vrai problème pour l'évolution de ces méthodes.

2.4 Évaluation de résumé

Les méthodes utilisées visant à évaluer les systèmes de RA se concentrent sur l'informativité et la qualité des résumés. Ces deux critères sont les plus importants, mais au fil du temps, d'autres ont été ajoutés comme celui de la fidélité à la source qui compare l'informativité du résumé avec celle de la source.

L'informativité du résumé consiste à comparer le résumé automatique avec un autre, le résumé de référence, qui est écrit par l'homme. Nous utiliserons dans ce mémoire des méthodes basées sur ce critère.

2.4.1 Évaluer l'informativité

Un résumé acceptable doit être plus court que le document source (C'est la notion de compression), et doit retenir les informations les plus importantes du document (Ce sont les ratios de rétention).

Le rappel, la précision et la F-mesure sont les mesures de performances les plus connues pour évaluer l'informativité d'un résumé [Rijsbergen, 1981]. Le rappel est défini par le nombre de phrases correctement sélectionnés par le système de RA au regard du nombre de mêmes unités sélectionnées par l'homme, alors

que la précision reflète le même nombre de phrases correctement sélectionnées par le système mais au regard du nombre total de phrases sélectionnées par le système. La F-mesure combine la précision et le rappel et calcule leur moyenne harmonique.

Évaluer la performance d'un système de RA à base de phrases ne se montre pas nécessairement très efficace. En effet, deux résumés très proches peuvent être jugés très différents à cause de l'existence d'une mauvaise correspondance entre les phrases.

Dans le but de surmonter ce problème, **l'Utilité de Relativité** a été proposée [Radev and Tam, 2003]. Dans cette méthode, le score de chaque phrase, de 0 à 10, détermine son importance dans le résumé.

D'autres méthodes évaluent le résumé en utilisant des fragments de texte, **le Score de Factoids** extrait les unités atomiques sémantique, appelés factoids [7], en commun de plusieurs résumés de référence et les cherche dans le résumé automatique pour calculer son score [Teufel and Van Halteren, 2004].

La Méthode des Pyramides se sert de la même logique, mais au lieu des factoids elle utilise les unités de contenu de résumé [8] (UCR). Chaque UCR a un poids calculé selon le nombre d'évaluateurs humains assurant son importance [Nenkova et al., 2007].

Ces trois méthodes dépendent d'annotateurs humains pour déterminer l'importance des phrases, ce qui prend beaucoup de temps.

2.4.1.1 La méthode ROUGE

ROUGE est l'acronyme de *l'Analyse Basée sur le Rappel pour l'Évaluation Fine* [9] destinée à évaluer les résumés de systèmes de RA. Ce package comprend plusieurs mesures d'évaluation : la mesure de N-gramme, celle des séquences de mots et celle des paires de mots. Ces mesures sont appliquées pour «déterminer la qualité du résumé automatique en comparaison avec des résumés de référence faits par l'homme» [Lin, 2004].

ROUGE-N ROUGE-N est un rappel de N-gramme entre les résumés automatiques et ceux de référence. Il est calculé comme suit :

$$\frac{\sum_{S\epsilon \text{ resumes referentiels}} \sum_{gramme_n \epsilon S} Count_{match}(gramme_n)}{\sum_{S\epsilon \text{ resumes referentiels}} \sum_{gramme_n \epsilon S} Count(gramme_n)}$$

Où n est la longueur du $N-gramme$, et $Count_{match}(gramme_n)$ est le nombre maximal de $n-grammes$ qui se trouvent dans les deux sortes de résumés.

Il convient de noter qu'en utilisant plusieurs résumés de référence, nous pouvons construire une procédure d'évaluation qui se concentre sur plusieurs aspects du résumé. De plus, cette méthode favorise les résumés qui contiennent des n-grammes existants dans plusieurs résumés de référence. Autrement dit, elle favorise les résumés qui ressemblent le plus à tous les résumés de référence.

7. La phrase "La police a arrêté un homme allemand." contient les factfoids suivants :
   1. un suspect a été arrêté
   2. La police a fait une arrestation.
   3. Le suspect est un homme.
   4. Le suspect est allemand.
8. Summary Content Units (SCU).
9. Recall-Oriented Understudy for Gisting Evaluation.

ROUGE-L Cette mesure utilise la séquence partagée la plus longue [10]. Elle est calculée comme suit :

$$R_{sp} = \frac{SP(X,Y)}{m}$$

$$P_{sp} = \frac{SP(X,Y)}{n}$$

$$F_{sp} = \frac{(1+\beta^2)R_{sp}P_{sp}}{R_{sp}+\beta^2 P_{sp}}$$

Où $SP(X,Y)$ est la séquence partagée la plus longue entre X et Y, m est la longueur de X et n est celle de Y, $ß$ est un paramètre pour régler le pourcentage entre le rappel et la précision lors du calcul de la F-mesure.

Cette méthode ne requiert pas une correspondance consécutive de mots. Par ailleurs, il n'est pas nécessaire de fixer un n comme pour les N-grammes, celui-ci est déterminé automatiquement via la plus longue séquence. Prenons par exemple les trois phrases suivantes :

$S1$: police killed the gunman

$S2$: police kill the gunman

$S3$: the gunman kill police

Considérons la première phrase comme un résumé de référence et les autres comme des résumés automatiques, ROUGE-2 S2 et S3 auront le même score grâce au 2-gramme "the gunman". En revanche, ROUGE-L S2 aura un score de 3/4 et S3 aura un score de 2/4 !

Il existe toutefois un désavantage à cette méthode. Elle ne compte que les séquences principales, c'est-à-dire que les séquences similaires en longueur et les plus courtes séquences ne comptent pas. Prenons par exemple S4. "the gunman police killed" et comparons-la avec S1, ROUGE-L comptera soit "the gunman" soit "police killed".

ROUGE-W

$$X : [ABCDEFG]$$

$$Y1 : [ABCDHIK]$$

$$Y2 : [AHBKCID]$$

Prenons les résumés ci-dessus où X est le résumé de référence et les autres sont des résumés automatiques. Le score de *Y1* et celui de *Y2* sont identiques selon la méthode précédente bien que *Y1* est clairement plus proche de X que *Y2*. Dans le but de surmonter ce défaut, ROUGE-W a été proposé. W est un acronyme de Weighted ROUGE-L. Cette mesure se sert de la longueur de correspondance consécutive pour favoriser les séquences les plus courtes, mais s'applique en utilisant la programmation dynamique.

10. The Longest Common Subsequence LCS

ROUGE-S Cette méthode exploite le concept *Skip-Bigramme* qui représente une paire de mots respectant leur ordre dans la phrase et permettant en même temps d'avoir des espaces entre eux. Par exemple, "Police gunman" est un *Skip-Bigramme* de la phrase "Police killed the gunman". le score de ROUGE-S se calcule comme suit :

$$R_{skip_2} = \frac{SKIP2(X,Y)}{c(m,2)}$$

$$P_{skip_2} = \frac{SKIP2(X,Y)}{c(n,2)}$$

$$F_{skip_2} = \frac{(1+\beta^2)R_{skip_2}P_{skip_2}}{R_{skip_2}+\beta^2 P_{skip_2}}$$

Où *SKIP2(X, Y)* est le nombre des Skip-Bigrammes partagés entre X et Y. $ß$ est un paramètre pour contrôler le pourcentage de rappel et précision lors du calcul de la F-mesure. C'est une fonction de combinaison [11].

2.4.1.2 Les méthodes de l'analyse de dépendance

Dans cette méthode, une analyse de dépendance s'applique aux résumés de référence et automatique pour mieux répondre à la tâche de trouver les informations en commun entre ces deux résumés.

La Méthode Éléments de Base est un exemple de ces méthodes. Chaque phrase est divisée en unités élémentaires de contenu afin de faciliter la tâche de trouver la correspondance entre les résumés. Plus précisément, pour chaque phrase, nous essayons d'extraire des unités composées de trois éléments, la tête, le modificateur et la relation.

Cette méthode utilise une liste fixée des motifs pour extraire les unités, mais une autre version a été proposée pour surmonter ce problème. **La méthode Éléments de Base avec Transformations pour l'Évaluation** [12] informatise la génération des règles de la liste précédente pour identifier les acronymes, les groupes propositionnels, les synonymes, les nominalisations, etc [Tratz and Hovy, 2008].

La dépendance de cette méthode des contraintes linguistiques de la langue d'entrée au niveau de l'analyse syntaxique est l'un de ces désavantages principaux.

2.4.1.3 La modélisation générative pour évaluer

Cette méthode cherche les termes de signature dans le résumé automatique. Ces termes sont des vecteurs de mots dédiés à un sujet particulier. Ils sont générés en se servant des POSs [13] du résumé de référence. Puis, la distribution de ces termes dans le document de ressource est calculée pour déterminer le poids de chaque terme. À la suite, la même procédure d'identification est appliquée sur le résumé automatique pour évaluer son informativité [Katragadda, 2010].

2.4.1.4 L'évaluation de Résumé Flou

Cette méthode ressemble aux autres méthodes en comparant les résumés de référence avec les automatiques, mais le texte du résumé est modélisé par des groupes flous.

11. la valeur de cette fonction pour un résumé d'une seule phrase composée de quatre mots est $C(4,2) = 4!/(2! * 2!) = 6$
12. Basic Elements with Transformations for Evaluation
13. Part of Speech

La similarité de chaque phrase du résumé automatique avec celle du résumé de référence est calculée en utilisant la Distance de Hamming. *L'Évaluation de Résumé Flou Améliorée par Dictionnaire*[14] est une amélioration de cette méthode. Dans cette version, les relations de WordNet sont utilisées pour améliorer le calcul de similarité entre les phrases [Ravindra et al., 2006] .

Se basant sur les théories de Dijkstra à propos de l'analyse discursive, d'autres méthodes sont proposées. Branny [Branny, 2007] se sert de *la Grammaire de Texte* dans l'objectif de développer sa méthode. La grammaire de texte est une façon formelle pour décrire la structure d'un texte. Elle décrit sa structure superficielle et profonde. La méthode l'utilise afin de créer une liste des propositions. Puis, l'évaluateur humain doit déterminer si les propositions sont pertinentes ou non. Pour chaque proposition, il y a trois scores à calculer, l'informativité, la grammaticalité et la mésinformation. Comme l'effort humain est requis, cette méthode est considérée comme coûteuse.

2.4.2 L'évaluation de la qualité

De nombreux chercheurs du domaine du RA s'intéressent à ce genre d'évaluation depuis l'émergence de ce domaine. La cohérence du résumé, sa grammaticalité et sa redondance sont les issues les plus importantes au point de vue de ce genre d'évaluation.

Le protocole FAN est parmi les premières tentatives à établir une base théorique de cette évaluation. Il [Minel et al., 1997] propose quatre critères à suivre pour évaluer la qualité d'un résumé :

1. Le nombre d'anaphores dépossédées de leurs référents[15] ;
2. La rupture des segments textuels[16] ;
3. La présence de tautologies[17] ;
4. La lisibilité du résumé.

D'autres critères sont considérés, comme l'évaluation de l'indicativité du résumé qui détermine si le résumé appartient au même sujet que ressource. L'acceptabilité de ses phrases est un autre critère qui détermine si la phrase sélectionnée par le résumé est adéquate à celle sélectionnée par l'homme.

14. Dictionary-Enhanced Fuzzy Summary Evaluator
15. En linguistique, une anaphore est un mot ou un syntagme qui, dans un énoncé, assure une reprise sémantique d'un précédent segment appelé antécédent.
16. Lorsque l'organisation syntaxique se modifie au cours d'une phrase, on a alors affaire à une véritable rupture syntaxique. par exemple, *"Mon voisin, son fils, il a eu un accident"*
17. La tautologie est une phrase ou un effet de style ainsi tourné que sa formulation ne puisse être que vraie. Exemple : *J'ai de la force parce que j'ai de la force.*

Chapitre 3

Résumé d'un seul document et maximisation des traits

La première expérimentation combine la maximisation des traits (MT) avec un système de résumé automatique d'un seul document. L'objectif principal de cette expérience est dcombiner notre approche avec un système simple, évaluer les résultats sur une tâche standard et comparer sa performance avec d'autres systèmes avant de le combiner avec des systèmes plus complexes.

Tout d'abord, nous allons expliquer la méthode de la MT, ses notions fondamentales et ses avantages. Ensuite, nous allons présenter la structure de notre système de RA d'un seul document et la manière dans laquelle nous y intégrons la MT. Nous allons également introduire la procédure suivie pour réduire la redondance des résumés produits. La section suivante décrit le corpus des expérimentations. Enfin, nous introduisons la plate forme d'évaluation utilisé et discutons les résultats des expérimentations .

3.1 Maximisation des traits

La MT est une méthode statistique basée sur la sélection de variables. Cette approche sert à choisir les traits les plus importants pour caractériser des classes issues de données hétérogènes. Elle s'applique ainsi particulièrement bien sur des données textuelles.

3.1.1 Sélection de variables

La sélection de variables est une méthode statistique de l'apprentissage automatique. Elle sert à la sélection des caractéristiques les plus importantes pour la construction des modèles dans une cadre supervisé.

Les autres avantages de ce type de méthode selon [Guyon and Elisseeff, 2003] sont de faciliter la visualisation et la compréhension des données, de diminuer le volume requis pour le stockage de données, de diminuer le temps d'entraînement et de réduire le bruit présent dans les données. En somme, la sélection de variables est une approche qui a pour objectif de diminuer la redondance des données.

Les méthodes de sélection de variables sont classées en trois catégories : les approches intégrées, celles basées sur l'enveloppement et celles basées sur le filtrage [Ladha and Deepa, 2011].

Les approches intégrées appliquent la partie de sélection à l'étape d'apprentissage. Les méthodes se basant sur les classifieurs de type MVS [18] et les réseaux de neurones sont les applications les plus connues. Par

18. Machine à Vecteurs Supports.

exemple, la méthode ERC-MVS [19] est un procédé intégré qui sélectionne les variables de manière itérative pour supprimer les marginales en se servant du classifieur MVS [Guyon et al., 2002].

Alternativement, la méthode *WrapperSubsetEval*, qui est une des applications les plus connues du procédé d'enveloppement, commence avec un ensemble vide. Pendant l'itération, elle ajoute les variables qui répondent à des critères précis comme la performance en apprentissage automatique en exploitant des technologies de validation croisée. Ces critères sont déterminés explicitement [Witten and Frank, 2005].

Dans les méthodes de filtrage, la sélection survient avant l'étape d'apprentissage d'une manière indépendante et statistique. Ces méthodes sont, par conséquent, plus favorables du point de vue de la simplicité et de la rapidité.

3.1.2 Maximisation des traits

Lamirel [Lamirel et al., 2015a] propose une nouvelle approche de filtrage pour la sélection de variables et pour la mesure de leur contraste. Elle se base sur la métrique de la MT. D'ailleurs, cette méthode présente des performances supérieures aux méthodes précédentes dans le contexte de la classification de données textuelles, voire, des données multidimensionnelles et hétérogènes.

Cette méthode exploite les traits des données associés à chaque classe sans aucune considération de son profil. Son indépendance de l'approche de classification et de son mode de fonctionnement est son avantage principal.

Elle peut être utilisée lors de la classification pour calculer les distances des classes [Lamirel et al., 2011]. Elle peut être également appliquée après l'étape d'apprentissage dans le cadre de l'étiquetage des classes [Lamirel, 2008].

Nous allons décrire la méthode et ses applications dans le domaine du RA en détail en nous servant d'un exemple proche de nos expérimentations.

Les articles scientifiques peuvent être considérés comme des données textuelles semi-structurées. Ils ont normalement une structure fixée ; chaque article se compose de plusieurs sections. Chaque section contient des dizaines de phrases. De plus, les titres de sections se ressemblent d'un article à l'autre.

C'est sur cette base que nous allons considérer que chaque article n'est qu'une liste de classes de mots. Chaque section est donc une classe et les mots sont les traits associés à cette classe.

Les rangs du tableau 3.1 représentent les sections, les classes, et les colonnes sont les traits, les mots. La valeur numérique dans la cellule (X, Y) représente donc le nombre d'occurrences du mot Y dans la classe X.

La MT va se servir de cette représentation dans l'objectif de trouver les mots les plus importants de chaque section. Comme pour la sélection de variables, la MT utilise la *F-mesure de trait* $FF(f)$ pour déterminer si le trait f associé à la classe c doit être retenu ou non [Lamirel et al., 2015b].

La *Feature F-mesure* ou *F-mesure de trait* $FF(f)$ représente la moyenne harmonique du Rappel $FR(f)$ et de la Précision $FP(f)$ de trait f. Les équations pour calculer ces mesures sont les suivantes :

$$FR_c(f) = \frac{W_c^f}{\sum_{c' \epsilon C} W_{c'}^f}$$

$$FP_c(f) = \frac{W_c^f}{\sum_{f' \epsilon F_c} W_c^{f'}}$$

$$FF_c(f) = 2\left(\frac{FR_c(f) \cdot FP_c(f)}{FR_c(f) + FP_c(f)}\right)$$

19. Recursive Feature Elimination for Support Vector Machines.

Où W_c^f est l'occurrence du mot f en classe c, C l'ensemble des classes, F_c est l'ensemble de traits dans la classe c. Si un trait possède un rappel élevé, alors il est saillant, typique de cette classe. Si il a un score de précision élevé, il est apte à décrire ou à représenter sa classe.

Pour formaliser la sélection des traits, nous utilisons les formules suivantes :

$$S_c = \left\{ f \epsilon F_c \mid FF_c(f) > \overline{FF}(f) \text{ and } FF_c(f) > \overline{FF_D} \right\}$$

$$\overline{FF}(f) = \sum_{c' \epsilon C} \frac{FF_{c'}(f)}{\left|C_{/f}\right|}$$

$$\overline{FF_D} = \sum_{f \epsilon F} \frac{\overline{FF}(f)}{|F|}$$

où S_c est le groupe des traits retenus dans la classe c, $\left|C_{/f}\right|$ est le nombre des classes dans lesquelles le trait f se retrouve, F est l'ensemble des traits. Par conséquent, l'ensemble de toutes les traits retenus est :

$$S_C = \bigcup_{c \epsilon C} S_C$$

Pour déterminer si le trait est *actif* (sur-représenté et saillant) ou *passif* (sous-représenté) dans sa classe, il faut calculer le contraste $G_c(f)$ qui représente le score de F-mesure du trait f divisé par la F-mesure de trait moyenne de tous les traits. Ce dernier est labellisé comme actif lorsque son contraste dépasse 1, il est passif sinon.

$$G_c(f) = FF_c(f)/\overline{FF}(f)$$

TABLE 3.1 – Les occurrences des mots les plus fréquents dans un document du corpus AQUINT

classe	"currency"	"EUR"	"Europe"	"Single"
1	4	5	3	2
2	2	5	3	2
3	10	6	5	8
4	3	10	5	1
5	6	10	3	4

Dans notre exemple, les colonnes du tableau 3.1 contiennent un ensemble de mots des cinq premiers articles du premier sujet du DUC 2007, alors que les rangs listent les indices de ces documents. L'occurrence de chaque mot Y dans le document X se localise dans la cellule (X, Y).

En considérant que nous n'avons que ces documents, le rappel, la précision et la F-mesure de trait pour la classe 1 sont calculés comme suit : $FF_1(currency) = \frac{2 \cdot FR_1(currency) \cdot FP_1(currency)}{FR_1(currency) + FP(currency,1)}$.

Puisque $FR_1(currency) = 4/25 = 0.16$ et $FP_1(currency) = 4/97 = 0.0412371134$, on a :

$$FF_1(currency) = 2 \cdot (0.16 \cdot 0.0412371134)/(0.16 + 0.0412371134) = 0.06557377048$$

À la suite de ces calculs pour chaque mot sur toutes les classes, les seuils à surmonter sont calculés et la comparaison détermine si le mot est saillant et descriptif, ou non.

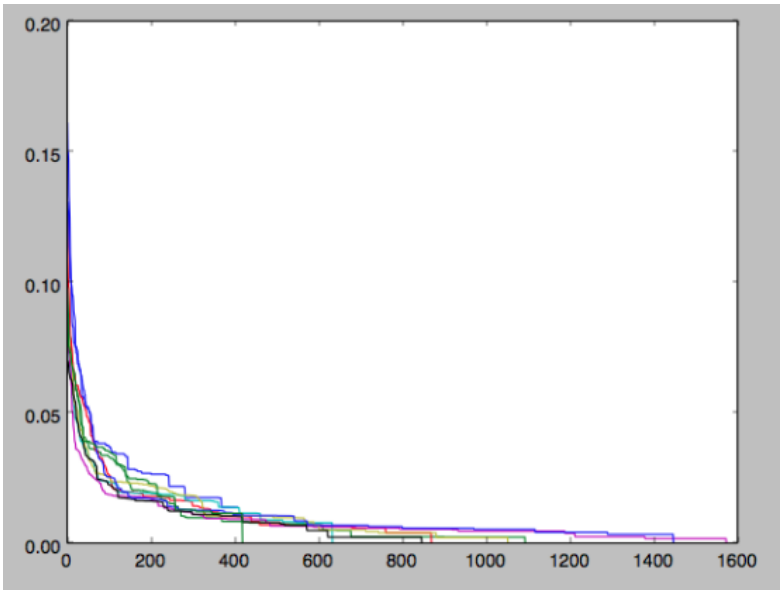

FIGURE 3.1 – La distribution des valeurs maximales de F-mesure de trait dans un document du corpus AQAINT.

3.1.3 Avantages et motivations

La MT présente une performance compétitive par rapport aux autres méthodes dans le domaine de la sélection de variables pour la classification de texte. Elle a par exemple montré des capacités de discrimination égales à celles de la métrique de CHi-2 [Lamirel et al., 2015a] tout en ayant des capacités de généralisation très supérieures. Par ailleurs, cette méthode posséde un faible complexité de calcul, notamment en comparaison avec les méthodes intégrées et avec les méthodes basées sur l'enveloppement.

Au niveau du résumé automatique, la méthode de MT est **langage-agnostique** ce qui signifie qu'elle est totalement indépendante du langage des documents d'entrée. Elle n'exige aucune connaissance linguistique et contrairement au FT*FID, aucun corpus n'est requis.

Cette approche nous permet également d'avoir un système de résumé **non paramétrique**. Le nombre de classes se calcule en bénéficiant de la structure de données d'entrée. De plus, la longueur du résumé se détermine de manière automatique.

Chaque couleur du graphique 3.1 montre la distribution des mots selon leurs valeurs maximales de F-mesure de trait dans un document du corpus AQUAINT. Nous pouvons constater que pour tous les documents, les valeurs de F-mesure de trait suivent une loi de puissance.

D'ailleurs, Clauset [Clauset et al., 2009] montre que nous pouvons déterminer une valeur X_{min} pour séparer les valeurs importantes de F-mesure de trait de celles qui suivent une loi de puissance et forment une longue queue dans le graphique 3.1.

De plus, le classement des phrases selon leur poids a la même distribution. Ce fait nous motive donc à exploiter ces distributions pour déterminer le seuil d'importance automatiquement dans le système de résumé automatique afin de découvrir la longueur optimale du résumé ou même pour combiner cette approche avec les systèmes de l'extraction des termes.

D'autre part, lancer une compétition entre les mots des sections de l'article garantit une meilleure représentation des résumés générés et réduit automatiquement la redondance.

3.2 Système de résumé automatique

Notre système génère un résumé extractif d'un seul document. Les mots sont tout d'abord classés selon les scores produits par la MT. Ces scores se calculent en considérant que le document semi-structuré se compose de plusieurs sections, classes, et que chaque mot est un trait de sa classe. Les mots entrent dans une compétition se basant sur leur fréquence prouver leur importance.

À la suite du calcul des scores de chaque mot, le poids des phrases est obtenu en agrégeant les poids des mots. Afin d'améliorer la redondance du résumé, nous appliquons une procédure gloutonne qui choisit les phrases les plus élevées en termes de poids et les plus variées en termes de similarité. Enfin, nous sélectionnons les phrases selon le classement produit de l'étape précédente. L'itération sur les documents s'applique comme suit :

1. Calculer la F-mesure de trait de chaque mot ;
   — Nous ne considérons que les mots importants qui ne sont ni des mots vides (déterminés à partir d'une liste ad-hoc), ni de la ponctuation, ni des nombres, ni un mélange de lettres et de chiffres.
   — L'occurrence d'un mot est celle de son lemme dans une séquence de lemmes des mots importants de l'article.
   — Nous établissons une pondération supplémentaire pour les mots-clés qui se trouvent dans les titres de l'article et des sections ou du résumé humain.
2. Calculer les poids des phrases ;
   — Le poids d'une phrase est la moyenne des poids de ses mots :

$$Score(S) = \frac{1}{|S|} \sum_{w \epsilon S} p(w)$$

3. Déterminer le seuil de coupure par rapport aux poids des mots afin de déterminer le seuil d'importance des mots ;
   — La distribution de la fréquence des mots suit une loi de puissance. Nous en bénéficions dans le but de déterminer ce seuil.
4. Réduire la redondance ;
5. Générer le résumé.

3.3 Réduction de redondance

La redondance est un des problèmes principaux de domaine du RA. Ce problème survient à la suite de la production de scores des mots où les phrases similaires ont des scores proches. C'est-à-dire que les phrases sont organisées dans plusieurs classes, chacune contenant des phrases similaires en termes de longueur, de mots en commun et de position des phrases.

Afin de générer un résumé non redondant, il faut représenter chaque classe et éviter de choisir plusieurs phrases de la même classe. Pour vérifier les capacités de notre méthode à éliminer la redondance, nous

l'avons couplée avec un des algorithmes les plus connus [Haghighi and Vanderwende, 2009] :

while *La longueur du résumé n'est pas atteinte* **do**
  Classer les phrases selon leur poids ;
  Choisir la phrase ayant le poids le plus élevé ;
  for *Chaque mot ayant une des F-mesures de trait les plus élevées de la phrase sélectionnée* **do**
    Réduire les F-mesures de trait des mots :

$$P^{nouveau}(w) = P^{ancien}(w).P^{ancien}(w)$$

  end
  Reproduire les poids du reste des phrases en utilisant les nouvelles F-mesures de trait des mots ;
end

Algorithm 1: L'algorithme simplifié de la diminution de redondance.

3.4 Problème de départ

Le mémoire a pour but de répondre à la tâche proposée sur le corpus *CL-SciSumm 2016*. Avant cela, nous devons évaluer notre méthode sur une tâche plus primitive, celle du résumé d'un seul document.

Dans le domaine du RA, il existe de nombreux corpus dont les spécialistes se servent pour évaluer leurs méthodes. Notre méthode considère que chaque document se compose de plusieurs clusters de phrases. Plus précisément, elle ne s'applique que sur les documents longs et semi-structurés. Nous devons donc trouver un corpus alternatif avec les traits suivants :

1. Un corpus standard, utilisé précédemment dans le même contexte.
2. Un corpus de documents assez longs à cause de la limite de notre méthode qui s'intéresse aux documents semi-structurés, comme les articles scientifiques.
3. Un corpus utilisé par les experts du RA pour bénéficier des résultats acquis dans la comparaison de performance de notre méthode.

Après la consultation des corpus les plus cités dans plusieurs revues de littérature du domaine, aucun corpus ne répond à nos conditions [Lloret, 2015].

La plupart de ces corpus se composent de documents courts, liés aux médias comme les rapports des agences de presse (brèves). En somme, la seule solution qui nous reste est de normaliser un des corpus les plus adaptés à notre approche.

3.5 Corpus AQUAINT

Ce corpus [Nist, 2007] a été proposé par la conférence DUC 2007 [20], organisée par le NIST [21]. L'objectif principal de cette conférence est d'encourager les chercheurs à établir un réel progrès sur le plan du RA et son évaluation.

Le challenge de DUC 2007 se composait de deux tâches, la «Tâche Principale» qui s'intéresse aux liens entre le RA et les systèmes de questions-réponses et la «Tâche Pilote» qui se base sur la production d'un résumé automatique de multiples documents et de sa mise à jour au fil de temps.

20. The Document Understanding Conference.
21. l'Institut National des standards de technologies

Les documents de ce corpus sont des articles des agences de presse composés de quelques lignes, classifiés selon leur sujet et leur date. Chaque sujet contient trois groupes chronologiques de documents, A, B et C, dont il faut générer le résumé de chacun d'entre eux, tout en considérant le groupe précédent.

Autrement dit, la tâche liée à ce corpus a pour objectif de générer un résumé de cent mots de chaque classe comme suit :

1. Un résumé de la classe A.
2. Une mise à jour du résumé précédent en bénéficiant des documents de la classe B.
3. Une autre mise à jour des deux résumés précédents en bénéficiant des documents de la classe C.

Le corpus comprend dix sujets qui se trouvent dans des fichiers XMLs structurés comme suit :

```
<collection name= 'D0703A-A'>
        <document name= 'XIE19960129.0071'>
                <line> ..... </line>

                <line> The European Commission (EC) spokesman today formally denied
                newspaper reports that the EC is secretly drawing up plans to delay
                the launch of the  single currency, the Euro.

                <annotation scu-count= '1' sum-count= '3' sums= '38,44,49'>
                <scu uid= 41' label= 'Eoro predicted to be
                                introduced on schedule' weight= '2' />
                </annotation>
                </line>
                .....
        </document>
</collection>
```

La racine de ce fichier est le nœud «collection». Ce nœud se compose de plusieurs nœuds «document». Chaque document contient plusieurs nœuds «line». La ligne peut être annotée par des «unités de contenu de résumé» (UCR) utilisées lors de l'évaluation par la méthode des pyramides [Hennig et al., 2010] . Les résumés de référence sont divisés dans plusieurs dossiers selon la méthode d'évaluation utilisée et les critères considérés.

Pour la méthode ROUGE, chaque sujet est associé aux quatre résumés de référence. De plus, il existe les résumés produits par les systèmes participants avec les résultats d'évaluation.

Ce qui nous intéresse dans cette tâche est la structure des documents où nous pouvons considérer que la classe A de chaque sujet est un seul document composé de plusieurs sections. Nous nous servirons des résultats de la première phase de la tâche pilote en considérant qu'elle peut rprésenter une tâche de résumé automatique d'un seul document.

3.6 ROUGE

Dans l'objectif d'évaluer les résumés générés par tous les systèmes participant au challenge DUC 2007, nous avons utilisé le package ROUGE 1.5.5.

Les résumés générés ont été limités à 100 mots. Puis, ROUGE 1.5.5 a été appliqué pour mesurer les scores ROUGE-N, où le gramme maximum à considérer ne contient que n mots, ROUGE-W, ROUGE-L, ROUGE-S, ROUGE-SU-n, où la distance entre les composants des grammes peut arriver à n mots.

L'évaluation s'est passée à la suite de la stemmatisation en gardant les mots vides. D'autres méthodes d'évaluations ont été appliquées mais nous ne les avons pas utilisées à cause de leur nature manuelle et leur exigence d'effort humain.

3.6.1 Fichier d'entrée de ROUGE

Le fichier d'entrée a le format XML et se compose du nœud *ROUGE-EVAL*. Ce nœud peut comprendre plusieurs tâches d'évaluation *EVAL*. Le chemin d'accès des résumés automatiques doit être indiqué dans le nœud *PEER-ROOT* et celui des résumés de référence se trouve dans le nœud *MODEL-ROOT*.

Les résumés automatiques sont indiqués dans le nœud *PEERS* et ceux de référence sont dans le nœud *MODELS*. D'ailleurs, le nœud *INPUT-FORMAT* peut contenir une liste des fichiers de configuration pour les utiliser lors de l'évaluation.

3.6.2 Paramètres d'exécution

Les paramètres d'exécution de ROUGE sont les suivants :

ROUGE-1.5.5.pl -n 4 -w 1.2 -m -2 4 -u -c 95 -r 1000 -f A -p 0,5 -t 0 -a -d ROUGEjk.in

ROUGE-1.5.5 : La version 1.5.5 de ROUGE.

-n 4 : ROUGE-4 pour les grammes de longueur 4.

-w 1.2 : Pour calculer ROUGE-W (Voir sec. 2.4.1.1) où le poids se multiplie par 1,2 pour favoriser la correspondance consécutive dans la plus longue subséquence en commun.

-m : Pour appliquer la stemmatisation.

-2 : Pour compter la cooccurrence des skips bigrammes (ROUGE-S) et pour déterminer la longueur maximale d'espaces dans les grammes.

-u : Comme avant mais il ajoute également les unigrammes.

-c 95 : La valeur défaut de l'intervalle de confiance.

-r 1000 : Pour déterminer les points d'échantillonnage.

-f A : Sélectionner le modèle d'affichage. **A :** La moyenne. **B :** Le meilleur résultat.

-p 0,5 : Favoriser le rappel quand > 1 ou la précision quand < 1.

-t 0 : Compter la moyenne de ROUGE sur tout le corpus de test au lieu des phrases. 0 pour utiliser les phrases, 1 pour utiliser les mots.

-a : Évaluer tous les résumés indiqués dans le fichier d'entrée.

-d : Imprimer le score de moyenne, évaluation de chaque système.

ROUGEjk.in : Le fichier d'entrée qui contient tous les détails des résumés de référence et automatiques.

3.7 Résultats et discussion

Les résultats de notre approche sur les données du corpus AQUAINT sont indiqués dans les tableaux 3.2, 3.3, 3.4, 3.5. Ils montrent les résultats de plusieurs expérimentations, avec et sans l'application de la diminution de redondance, en favorisant les phrases longues ou non, et en se servant de la structure ordinaire des documents ou en combinat les sections par groupe de deux.

Il convient de noter qu'ils ne sont pas significativement différents et que l'expérimentation produisant les scores les plus élevés ne contient ni l'application de la diminution de redondance, ni la valorisation des phrases longues. Ces résultats permettent donc, entre autre choses, de confirmer que notre méthode possède bien des capacités intrinséques de gestion de la redondance.

TABLE 3.2 – Les résultats de notre méthode sur DUC 2007 avec l'application de la diminution de redondance sur la structure ordinaire des documents, en favorisant les phrases longues.

	Rappel	Précision	F-mesure
ROUGE-1	0.23192	0.2743	0.24539
ROUGE-2	0.02826	0.0333	0.0298
ROUGE-3	0.00423	0.00432	0.00422
ROUGE-4	0.00145	0.00145	0.00143
ROUGE-L	0.20383	0.23945	0.215
ROUGE-SU*	0.05476	0.0747	0.05886
ROUGE-S*	0.05122	0.06946	0.05493
ROUGE-W-1.2	0.07076	0.15295	0.09475

TABLE 3.3 – Les résultats de notre méthode sur DUC 2007 avec l'application de la diminution de redondance sur une structure spécialisée des documents, en favorisant les phrases longues.

	Rappel	Précision	F-mesure
ROUGE-1	0.22499	0.27164	0.24184
ROUGE-2	0.02513	0.03003	0.02692
ROUGE-3	0.00334	0.00338	0.00331
ROUGE-4	0.00077	0.00074	0.00073
ROUGE-L	0.20242	0.24352	0.21717
ROUGE-SU*	0.04876	0.07137	0.05486
ROUGE-S*	0.0453	0.06633	0.05093
ROUGE-W-1.2	0.07082	0.15546	0.09572

TABLE 3.4 – Les résultats de notre méthode sur DUC 2007 avec l'application de la diminution de redondance sur la structure ordinaire des documents, sans favoriser les phrases longues.

	Rappel	Précision	F-mesure
ROUGE-1	0.2363	0.26976	0.24613
ROUGE-2	0.02761	0.03163	0.02873
ROUGE-3	0.0038	0.00372	0.00371
ROUGE-4	0.00123	0.0012	0.0012
ROUGE-L	0.20907	0.23779	0.21738
ROUGE-SU*	0.05692	0.07167	0.0593
ROUGE-S*	0.05334	0.06664	0.05541
ROUGE-W-1.2	0.07233	0.15123	0.09587

TABLE 3.5 – Les résultats de notre méthode sur DUC 2007 sans l'application de la diminution de redondance sur la structure ordinaire des documents, sans favoriser les phrases longues.

	Rappel	Précision	F-mesure
ROUGE-1	0.26066	0.26949	0.26336
ROUGE-2	0.03634	0.03811	0.03695
ROUGE-3	0.00541	0.00579	0.00554
ROUGE-4	0.00171	0.0018	0.00174
ROUGE-L	0.22918	0.23717	0.23164
ROUGE-SU*	0.06245	0.068	0.06376
ROUGE-S*	0.0791	0.15001	0.10309
ROUGE-W-1.2	0.06628	0.07213	0.06768

TABLE 3.6 – Les meilleurs résultats, le système 40, sur DUC 2007.

	Rappel	Précision	F-mesure
ROUGE-1	0,3675	0,3688	0,36793
ROUGE-2	0,12095	0,12072	0,12076
ROUGE-3	0,05946	0,05911	0,05924
ROUGE-4	0,0364	0,03606	0,0362
ROUGE-L	0,3334	0,33446	0,33373
ROUGE-SU*	0,12045	0,1212	0,12059
ROUGE-S*	0,11555	0,11625	0,11567
ROUGE-W-1.2	0,12118	0,21873	0,15586

TABLE 3.7 – Les pires résultats, le système 57, sur DUC 2007.

	Rappel	Précision	F-mesure
ROUGE-1	0,23655	0,23861	0,23744
ROUGE-2	0,03318	0,0334	0,03326
ROUGE-3	0,00376	0,00376	0,00376
ROUGE-4	0,00046	0,00045	0,00046
ROUGE-L	0,21349	0,21522	0,21424
ROUGE-SU*	0,05387	0,05498	0,05429
ROUGE-S*	0,0502	0,05125	0,05059
ROUGE-W-1.2	0,07549	0,13709	0,09731

D'autre part, la même procédure d'évaluation a été appliquée sur les résumés de 24 systèmes, qui ont participé au challenge de DUC 2007. Ces systèmes ont été numérotés de 35 à 58, et les scores les plus élevés, les plus bas et leur moyenne sont indiqués dans les tableaux 3.6, 3.7, 3.8.

Parmi les 24 systèmes, notre système a été classé 17ème au niveau de ROUGE-W, 18ème par rapport aux ROUGE-1...3, ROUGE-S et ROUGE-SU et 20ème par rapport à ROUGE-4.

Bien que le système ait atteint le score du système de base, les scores obtenus ne reflètent pas ceux attendus de l'approche. Une des raisons principales de ces scores est la nature du corpus qui s'intéresse au résumé de documents multiples. Notre méthode est mise en œuvre dans l'objectif de mesurer la performance de la MT sur la tâche du résumé automatique d'un seul document semi-structuré, mais l'absence d'un corpus destiné à ce genre d'expériences nous a conduit à nous adapter à celui-ci.

Dans les tableaux 3.10, 3.9 nous comparons la distributions des mots dans deux documents, le premier appartient au corpus DUC 2007 et le deuxième appartient au corpus CL-SciSumm, que nous utilisons dans le chapitre suivant et pour lequel nous obtenons de meilleures performances. Les mots de forte fréquence étant considérés comme non informatifs, donc éliminés, ceux qui sont pris en compte dans ces tableaux sont ceux dont les rangs de classement en fréquence varient entre 10 et 20. La différence du

TABLE 3.8 – La moyenne des résultats sur DUC 2007.

	Rappel	Précision	F-mesure
ROUGE-1	0,3122984	0,3184032	0,3147808
ROUGE-2	0,0778952	0,0790492	0,0783448
ROUGE-3	0,0282812	0,0285344	0,0283696
ROUGE-4	0,0143408	0,014432	0,0143712
ROUGE-L	0,277538	0,2827008	0,2796316
ROUGE-SU*	0,0929892	0,0961824	0,093992
ROUGE-S*	0,0886288	0,091636	0,0895612
ROUGE-W-1.2	0,0990956	0,1818752	0,128096

TABLE 3.9 – la distribution de dix mots classés par leur occurrence dans un document du corpus CL-SciSumm composé de six sections.

mot	Classement	Fréquence	Fréquence dans les sections
arc	10	95	3- 1- 46- 41- 4- 0
Unification	11	88	26- 7- 12- 7- 24- 12
Copy	12	75	18- 0- 15- 38- 0- 4
hod	13	72	34- 0- 7- 11- 10- 10
methods	14	71	33- 0- 7- 11- 10- 10
ares	15	62	11- 7- 24- 13- 7- 0
structures	16	57	25- 3- 13- 10- 0- 6
feature	17	50	5- 16- 6- 2- 19- 2
values	18	43	0- 9- 20- 10- 4- 0
Cons	19	41	19- 4- 6- 8- 3- 1

TABLE 3.10 – la distribution de dix mots classés par leur occurrence dans un document du corpus AQUAINT composé de dix sections.

mot	Classement	Fréquence	Fréquence dans les sections
countries	10	8	1- 0- 3- 1- 1- 0- 0- 1- 1- 0
January	11	7	0- 1- 1- 1- 0- 0- 1- 1- 2- 0
low	12	7	0- 0- 1- 0- 1- 0- 2- 3- 0- 0
plans	13	7	0- 1- 2- 0- 3- 0- 0- 1- 0- 0
Zambia	14	7	0- 0- 0- 0- 0- 7- 0- 0- 0- 0
rates	15	6	0- 0- 3- 0- 1- 0- 0- 2- 0- 0
market	16	6	0- 0- 0- 0- 1- 3- 0- 0- 0- 2
nations	17	6	0- 1- 2- 0- 0- 0- 1- 0- 1- 1
financial	18	6	0- 0- 1- 0- 1- 2- 1- 0- 0- 1
launch	19	6	0- 2- 0- 0- 2- 0- 1- 0- 1- 0

niveau de fréquence des mots considérés entre les deux corpus ainsi que le fait que la contribution en fréquence de ces mots varie de manière plus significative entre les sections dans le corpus CL-SciSumm, aident à expliquer les scores assez faibles lors de l'application de notre méthode sur le corpus DUC 2007. En effet, dans ce dernier cas, les caractéristiques spécifiques du corpus DUC 2007 semblent clairement handicaper les capacités de discrimination de notre approche.

De plus, le grand nombre de classes, au moins dix pour chaque sujet du corpus, et le faible nombre de phrases dans les classes limitent également les capacités de l'approche, là où la compétition entre les mots s'enflamme et les poids des mots se rapprochent.

Chapitre 4

Résumé de Communauté : MT avec Similarité Cosinus

4.1 Corpus CL-SciSumm 2016

CL-SciSumm [22] est un challenge sur le résumé automatique des articles scientifiques dans le domaine de la linguistique computationnelle [SCI-Summ, 2016].

Les résumés à générer doivent être de deux types : le résumé de facette de l'abstrait des articles scientifiques et le résumé de communauté qui contient les phrases citées de l'article par d'autres articles donnés.

Il y a également une autre tâche : la classification des citations selon les facettes auxquelles elles sont associées. Cette tâche suit « La Tâche Pilote de Linguistique Computationelle (LC) », une des tâches de TAC 2014 [23] [Tratz and Hovy, 2008] dans laquelle le dataset SciSumm14 a été publié. Le dataset se compose de dix articles traitant de LC et leurs résumés avec d'autres annotations comme les citations de chaque article et les liens de chaque citations avec l'article de référence et les facettes que la citation représente.

Le corpus se compose de trois ensembles de documents, un pour l'entraînement, le deuxième pour le développement et le dernier pour le test. Les trois ensembles ont une structure identique, mais ils se différencient au niveau du nombre de documents.

L'ensemble d'entraînement, par exemple, comprend dix articles de référence. Chaque article est associé à plusieurs articles citants.

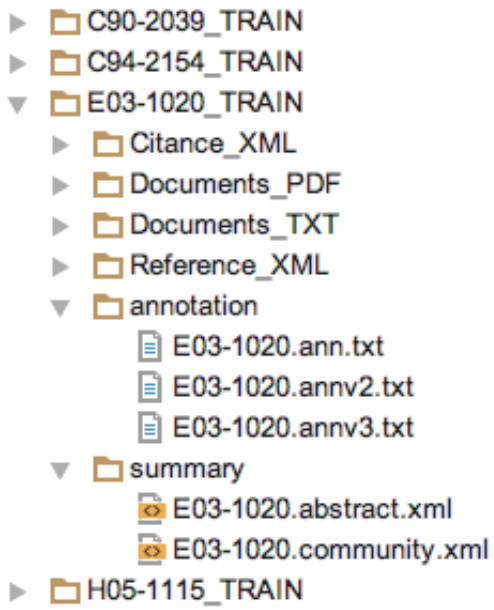

FIGURE 4.1 – La structure du corpus SCi-Summ.

22. Computational Linguistics Scientific Document Summarization.
23. BiomedSumm Track at the Text Analysis Conference 2014.

Sur l'image 4.1 ci-contre, le document ***E03-1020-TRAIN*** a, comme les autres, 6 dossiers. Le dossier ***Citance_XML*** contient les articles citants en format XML. Celui de ***Documents_PDFs*** ou celui de ***Documents_TXT*** contient l'article de référence avec les citants au format PDF et TXT. Le dossier ***Reference_XML*** comprend quant à lui un seul fichier, celui de l'article de référence au format XML.

Le format XML des articles, que ce soit d'article citant ou de référence, est simple et efficace. Le dossier ***Annotation*** contient trois fichiers d'annotation de trois annotateurs, mais le troisième fichier est le fichier approuvé pour cette tâche.

Chaque ligne du fichier contient toutes les informations concernant l'article citant. Elle est numérisée avec le nom de l'article de référence et celui de l'article citant, les phrases de citations et celles de la référence. Les phrases citantes ou celles de référence sont copiées de leur source en gardant le format XML.

Enfin, le dossier ***summary*** contient le résumé direct de l'article de référence et le résumé de communauté.

En somme, ce chapitre de mémoire mémoire a été attaché au challenge CL-SciSumm [24] dans l'objectif de bénéficier du corpus fourni, des résultats obtenus, mais également de pouvoir comparer notre méthode avec les méthodes les plus modernes et leurs mises à jour.

La tâche à laquelle nous nous intéressons est la suivante :

Tâche 1 (Résumé de communauté CL-SciSumm) *Nous disposons d'un document référent D_r à résumer et d'un ensemble de documents D=(D_1, $\cdots$, D_n) dans lesquels on trouve une citation au moins vers D_r. Pour résumer D_r, le challenge nous impose d'utiliser les citations contenues dans D. Ainsi, il s'agit pour chaque citation d'un document de D vers D_r, de trouver la ou les phrases de D_r auxquelles la citation fait référence. L'hypothèse est que ces phrases de $D_r$sont particulièrement pertinentes, puisque leur contenu et leur sens sont cités par d'autres documents. Elles peuvent ainsi être combinées pour former un résumé de communauté (à l'aide des documents de D) de D_r.*

4.2 Normalisation du corpus

Les fichiers TXT et XML des articles du corpus contiennent de nombreuses erreurs de reconnaissance des mots parce qu'ils sont extraits par un système OCR de leurs versions PDFs.

Dans l'objectif de réduire l'effet de ce problème, nous devons normaliser les textes des articles avant de les passer au système de RA.

La phrase suivante [25] est un exemple des erreurs qui se trouvaient dans le corpus :

U__ Bottom Figure 1 : Exainple of a type symbol lattice --2-- pe Symbol10 eaturel TypeSymboll]]] I feature2 TypeSymbol2 I feature3 ?Tag T ypeSymbol3]]feature4 TypeSymbol4 L [.feature5 TypeSymbol5 TIeature3 7Tag (a) feature-value matrix notation &quot ; ?&quot ; i~ the prefix for a tag and TFSs with the same tag are token-identical.

Afin d'améliorer la qualité du texte d'entrée, un module de correction a été mis en œuvre. Le module traite les situations suivantes :

— Les situations comme « Feat- ure » : Ces situations sont très nombreuses et faciles à régler. Elles se produisent quand un mot est divisé entre deux lignes du fichier PDF. Toutefois, cette hypothèse nous interdit de détecter des cas comme celui du mot « 2-gramme ».

— Les situations comme « Symbol10 : Dans les articles scientifiques, il y a souvent des exemples avec des paramètres. Une grande partie des noms de ces paramètres est exprimée comme dans notre

24. The 2nd Computational Linguistics Scientific Document Summarization Shared Task.
25. La ligne numéro 83 du fichier *Reference_XML* du document *C90-2039_TRAIN,* Corpus *CL-SciSumm.*

exemple « Symbol10 ». Nous avons considéré qu'en séparant ces noms en deux parties [« Symbol », « 10 »] nous pouvions améliorer la performance de notre méthode grâce à l'augmentation de l'occurrence des mots.

— Les situations de typographie : Nous appliquons une itération sur tous les mots de chaque article du corpus. Si le mot ne se trouve pas dans le dictionnaire, nous essayons de suggérer un autre mot. Quand le dictionnaire ne trouve pas un mot, il propose de nombreuses suggestions classées selon leurs similarités au mot. Par exemple, les suggestions du mot « fornalism » sont [« journalism », « formalism », « paternalism », « formalist », « factionalism »].

Dans le cas d'un article de LC comme l'article dans lequel le mot se trouve, il est clair que la deuxième suggestion est la plus pertinente. La suggestion « journalismes » est toutefois la plus proche selon le dictionnaire. Afin de régler ce problème et d'augmenter la performance de notre correcteur, nous nous servons du texte de l'article pour associer les suggestions aux scores de leur occurrence dans l'article.

D'autre part, les suggestions du mot « ubstructures » sont [« substructures », « substructure », « understructures », « strucures », « restructure »]. La suggestion requise est la première, mais le mot « structure » est un mot très fréquent en comparaison avec le mot « substructure ». Afin de surmonter ce scénario, nous devons modifier notre procédure précédente.En conséquence, nous mesurons la distance entre le mot d'entrée et la suggestion en utilisant l'algorithme de Levenshtein [26]. Ensuite, nous n'utilisons la mesure d'occurrence de mot dans l'article qu'en cas d'une distance dépassant les trois lettres entre ce mot et ses suggestions.

Le texte suivant est un exemple de la performance de cette procédure :
« In such unification-based formalisms, feature **structure** [~trueture] FS unification is the most fundamental and **significant** [~ignifieant] operation. »

Toutefois, les noms propres et les entités nommées sont des exemples des problèmes de notre procédure parce que le dictionnaire ne les trouve pas et que la procédure essaie de trouver les suggestions les plus proches. Pourtant, dans le contexte de notre système, ce genre de problème n'a pas de conséquences en raison de sa généralisation sur toutes les occurrences !

4.3 Similarité cosinus avec maximisation des traits

Pour rappel (Voir Tâche 1 décrite Page 29 : Résumé de communauté CL-SciSumm), chaque article de D contient une ou plusieurs citations faisant référence à une ou plusieurs phrases de l'article de référence D_r. L'objectif est de trouver dans D_r les phrases auxquelles font références les articles de D. Ces phrases sont ensuite utilisées pour former le résumé de communauté. Nous pouvons ainsi nous considérer face à un problème de génération de résumés se basant sur une requête [27].

Ce genre de problèmes est normalement approché par une combinaison de techniques de TAL et de celles exploitéees dans les systèmes d'information. Nous allons, par conséquent, proposer une méthode composée de la combinaison de la maximisation des traits avec la similarité cosinus. L'application de la similarité cosinus avec la méthode TF*IDF nous encourage à appliquer cette méthode en raison de la similarité générale avec la nôtre. Rappelons que ces méthodes se basent toute deux sur le concept de fréquence des mots [Tata and Patel, 2007].

De manière générale, nous allons évaluer l'importance des mots de l'article de référence et extraire leur F-mesure de trait. Puis, nous allons générer les modèles vectoriels des phrases de l'article et des requêtes

26. Un algorithme couramment utilisée dans le but de calculer la distance entre les mots. [en ligne] Une mise en oeuvre de l'algorithme en Python. <*https://pypi.python.org/pypi/python-Levenshtein/0.12.0*>. [Consultée le 1 août 2016]
27. Query-focused summerization.

selon les valeurs de F-mesure de trait. Enfin, la métrique de similarité cosinus sera appliquée afin d'avoir le résumé final, celui de communauté.

4.3.1 Modèle spatial de vecteurs

Les modèles spatiaux de vecteurs sont des modèles algébriques qui représentent les données textuelles en tant que vecteurs. Ces vecteurs doivent contenir des valeurs numériques représentant des aspects précis des données textuelles.

Par exemple, dans les systèmes d'information, la fréquence des mots ou des termes représente ces valeurs dans l'objectif de mesurer la similarité d'une requête des documents d'un corpus donné.

Il est important de souligner que ces modèles peuvent utiliser des paramètres locaux et globaux. Autrement dit, si nous voulions mesurer la similarité d'une requête d'une phrase, nous pourrions bénéficier de la fréquence des mots de la phrase dans un corpus afin de formuler nos vecteurs. Cet avantage accorde la flexibilité requise pour approcher plusieurs problèmes.

La première étape de la procédure de vectorisation consiste à établir un sac de mots avec les mots se trouvent dans la requête et ceux qui se trouvent dans la phrase à comparer. Le sac de mots est une technique couramment utilisée dans le domaine du TAL et des systèmes d'information. Elle vise à produire une représentation condensée des données de l'entrée. Soient les deux phrases suivantes en exemple :

$Ph1$: Hazem déteste le jeu Pokémon-Go.

$Ph2$: Hazem déteste la chanteuse Rihanna.

Le sac de mots de ces deux phrases sera :

S : {« Hazem », « déteste », « le », « jeu », « Pokémon-Go », « . », « la », « chanteuse », « Rihanna »}

Les mots vides, les articles et les ponctuations sont normalement supprimés du sac à cause de leurs faibles valeurs au niveau sémantique, ce qui donne S_{Stop}.

S_{Stop} : {« Hazem », « déteste », « jeu », « Pokémon-Go », « chanteuse », « Rihanna »}

Dans notre méthode, nous appliquons une autre simplification. Chaque mot a sa propre F-mesure de trait et les scores de F-mesure de trait des mots suivent une distribution en loi de puissance qui nous permet d'extraire un seuil pour l'importance des mots. Nous nous servons de ce seuil pour déterminer si le mot doit être ajouté au sac ou non (Voir Section 3.1.3).

Considérons que les mots « jeu », « Pokémon-Go » font partie des mots inférieurs au seuil d'importance, le sac aura donc les éléments suivants :

S_{mt} : {« Hazem », « déteste », « chanteuse », « Rihanna »}

C'est ce type de sac de mots S_{mt} que nous utilisons dans la suite de ce mémoire. Ainsi, nous générons les vecteurs en utilisant les valeurs de F-mesure de trait des mots de ce sac. Une itération sur les mots

de chaque phrase construit le vecteur comme suit :

$V = []$;
for *w in Ph* **do**
  if *w in S* **then**
    V.Ajoute ($F(w)$) ;
  end
end
return V

Algorithm 2: L'algorithme simplifié de la construction des vecteurs.

Nous devons souligner à nouveau que F-mesure de trait d'un mot est dépendante de la section de l'article à laquelle la phrase du mot appartient.

Supposons que l'article se compose de trois sections, on a alors un sac de mot, où, pour chaque mot, la valeur associé varie en fonction de la section :

S_{mt}^{sec} : {
« Hazem » : {sec1= 0.1, sec2 = 0.3, sec3 = 0.01},
« déteste » : {sec1= 0.4, sec2 = 0.02, sec3 = 0.11},
« chanteuse » : {sec1= 0.1, sec2 = 0.0, sec3 = 0.6},
« Rihanna » : {sec1= 0.0, sec2 = 0.3, sec3 = 0.4}
}

On suppose que Ph1 est une phrase de la section 1 et on lui affecte donc la représentation suivante :

$$V1 = [0.1, 0.4, 0, 0]$$

Si l'on considère que Ph2 est une requête à laquelle il nous faut répondre, nous proposons en revanche plusieurs représentations . Nous pouvons tout d'abord utiliser les valeurs des mots de la section à laquelle appartient la phrase Ph1 et présents dans Ph2 comme suit :

Représentation 1 (Valeurs de F-Mesure de trait de la section)

$$V2_1 = [0.1, 0.4, 0.1, 0]$$

Une autre représentation peut être appliquée si nous prenons les valeurs maximales de F-mesure de trait de chaque mot de la requête toutes sections confondues. Cette représentation serait :

Représentation 2 (Valeurs maximales de F-mesure de trait)

$$V2_2 = [0.3, 0.4, 0.6, 0.4]$$

Nous proposons enfin une version binaire ce qui donne les représentations suivantes dans le cas précédent (nous adaptons également la représentation de Ph1 dans ce cas) :

Représentation 3 (Binaire)

$$V2_2^b = [1, 1, 1, 1]$$
$$V1^b = [1, 1, 0, 0]$$

4.3.2 Similarité cosinus

À la suite de la vectorisation de deux phrases à comparer, nous appliquons la mesure similarité cosinus afin de calculer la distance entre ces deux vecteurs, la similarité de deux phrases données. Cette méthode

est très courante dans le domaine de traitement de données. Elle permet de calculer le cosinus de l'angle entre les deux vecteurs qui doivent être de même dimension.

Cette technique se base sur le concept de produit scalaire. Soient deux vecteurs $\overrightarrow{a} = (a_1, a_2, ..., a_n)$ et $\overrightarrow{b} = (b_1, b_2, ..., b_n)$

le produit scalaire de ces vecteurs est :

$$\overrightarrow{a}.\overrightarrow{b} = \sum_{i=1}^{n} a_1 b_1 = a_1 b_1 + a_2 b_2 + ... + a_n b_n$$

Appliqué à notre précédent exemple, le produit scalaire de $V1 = [0.1, 0.4, 0, 0]$ et $V2_1 = [0.1, 0.4, 0.1, 0]$ est :

$$V1.V2_1 = 0,1 * 0,1 + 0,4 * 0,4 + 0 * 0,1 + 0 * 0 = 0,17$$

D'autre part, la définition géométrique de ce produit est :

$$\overrightarrow{a}.\overrightarrow{b} = \|\overrightarrow{a}\| \left\|\overrightarrow{b}\right\| cos\theta$$

La norme d'un vecteur ||a|| est évaluée par la norme Euclidienne. Soit le vecteur

$\overrightarrow{x} = (x_1, x_2, ..., x_n)$, sa norme est donc $||x|| = \sqrt{\sum_{k=1}^{n} |x_k|^2}$

Par exemple, les normes de $V1$ et de $V2_1$ sont :

$$||V1|| = \sqrt{(0,1 * 0,1 + 0,4 * 0,4)} = 0,41231056256$$

$$||V2_1|| = \sqrt{(0,1 * 0,1 + 0,4 * 0,4 + 0,1 * 0,1)} = 0,42426406871$$

Le cosinus entre les deux vecteurs est calculé comme suit :

$$cos\theta = \frac{\overrightarrow{a}.\overrightarrow{b}}{\|\overrightarrow{a}\| \left\|\overrightarrow{b}\right\|}$$

$$Cos(\varnothing) = (0,17)/(0,41231056256 * 0,42426406871) = 0,97182531581$$

La valeur du cosinus détermine la similarité entre la phrase et la requête. Si l'angle entre les deux vecteurs est important, le cosinus est proche de zéro, sinon le cosinus est proche de 1. Ainsi, dans notre exemple, la différence entre les deux vecteurs est faible (forte valeur de cosinus) et nous pouvons dire que les deux phrases sont des phrases similaires.

Ci-dessous, les trois situations des valeurs extrêmes de la métrique de similarité cosinus. Quand les vecteurs sont proches l'un de l'autre, l'angle est proche de 0 et le cosinus est proche de 1. Dans les deux autres situations, l'angle droit et l'angle plat, le cosinus est compté soit comme 0 (angle droit), soit comme -1 (angle plat) [Image 4.2].

Toutes nos expérimentations précédentes sont faites en utilisant les F-mesures de trait de mots. La vectorisation se sert des valeurs de F-mesures de trait afin d'établir les vecteurs. Nous allons, toutefois, expérimenter la similarité cosinus en se basant sur les F-mesures de trait des termes et mots.

Les termes sont les expressions composées de deux et de trois mots. Par exemple, la phrase « Hazem déteste la chanteuse américaine Rihanna » contient les termes suivants :

« Hazem déteste », « chanteuse américaine », « américaine Rihanna », « chanteuse américaine Rihanna »

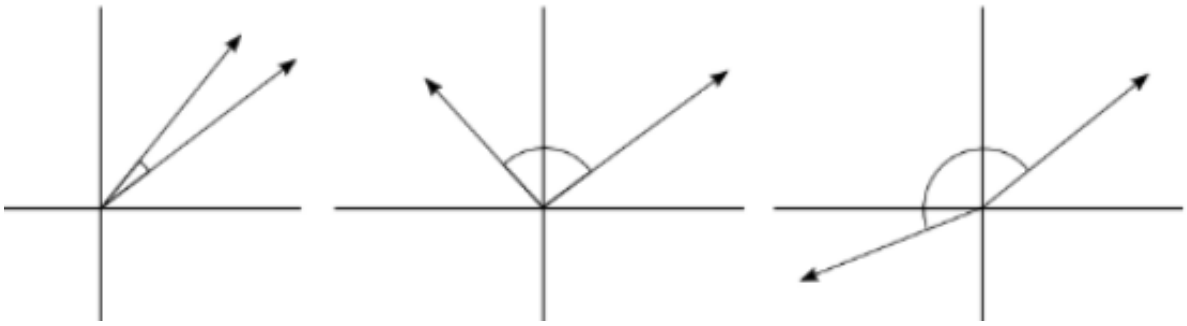

FIGURE 4.2 – différentes situations de cosinus entre deux vecteurs.
Source : <*http://goo.gl/Om1Lyt*> [Cconsultée 01 août 2016].

Notons bien que des termes comme « déteste la » ne se comptent pas parce qu'ils contiennent des mots vides. À la suite de la génération de ces termes, nous leur appliquons la maximisation des traits afin d'avoir leur score de F-mesure de trait. Lors de la vectorisation, nous intégrons les scores de F-mesures de trait des mots et des termes de chaque phrase en supprimant les valeurs de F-mesures de trait des mots qui se trouvent dans les termes de la phrase marquées comme termes importants.

Dans notre exemple, nous considérons que le terme « chanteuse américaine » : 0,35 est un terme important dans l'article de la phrase précédente, et que le vecteur à base de mots est V1 = [« Hazem » : 0,6, « déteste » : 0,2, « chanteuse » : 0,4, « américaine » : 0,41, « Rihanna » : 0,6]. Ainsi, le nouveau vecteur est :

V1 = [« Hazem » : 0,6, « déteste » : 0,2, « chanteuse américaine » : 0,35, « Rihanna » : 0,6].

4.4 Algorithme

En somme, l'algorithme simplifié de notre méthode se traduit comme suit :

```
Lire l'article ;
Normaliser l'article avec la correction automatique des erreurs système OCR ;
Calculer la F-mesure de trait de chaque mot et de chaque terme ;
Calculer les poids des phrases ;
for chaque citation de l'annotation do
    for chaque phrase de l'article do
        Générer le modèle spatial de vecteur de la phrase et de la requête ;
        Mesurer la distance entre le vecteur de la citation et celle de la phrase ;
    end
    Trier les phrases selon leur distance ;
    Ajouter la phrase ayant la distance maximale au résumé ;
end
Générer le résumé.
```

Algorithm 3: Algorithme simplifié de notre méthode de résumé de communauté en utilisant la métrique de la similarité cosinus

4.5 Évaluation et discussion

Afin d'évaluer notre méthode, nous allons l'appliquer au corpus CL-SciSumm en bénéficiant de la structure de ses articles, des annotations et des résumés annotés par l'homme.

TABLE 4.1 – Les résultats de notre méthode **sans** l'application de la diminution de redondance, en utilisant la **Représentation 3** (Binaire).

	Rappel	Précision	F-mesure
ROUGE-1	0,551859	0,426142	0,45335
ROUGE-2	0,262861	0,193951	0,209538
ROUGE-3	0,192741	0,139041	0,15149
ROUGE-4	0,17187	0,122583	0,134116
ROUGE-L	0,519583	0,400512	0,426116
ROUGE-S*	0,306957	0,181125	0,187657
ROUGE-W-1.2	0,150405	0,22588	0,169172

Dans ce corpus, il y a, pour chaque article de référence D_r, un résumé extractif de communauté composé des phrases de D_r qui sont citées par les articles de D (Voir Tâche 1). Nous allons nous en servir dans la procédure d'évaluation automatique. Pour cette évaluation, nous utilisons la famille de métriques ROUGE qui quantifient l'intersection entre le résumé de référence extrait par l'homme et celui automatique généré par notre système.

Grâce à la nature des résumés générés par ce système, nous n'avons pas besoin d'autres mesures d'évaluation. ROUGE est d'une très grande efficacité sur le plan de la mesure de l'informativité des résumés extractifs (Voir section 2.4.1.1). D'autre part, l'évaluation de la quantité n'est pas requise à cause de la structure de ces résumés qui ressemble aux rapports techniques et de leurs composants qui ne requièrent aucun lien entre eux.

Lors de l'application de ROUGE sur ces résumés, contrairement à son application sur le corpus AQUAINT, nous devons considérer tous les scores, le rappel, la précision et la F-mesure parce que les longueurs des résumés de référence et automatiques sont différentes et parce qu'il n'y a pas d'autres résumés à comparer.

Les tableaux de cette section montrent les résultats de plusieurs expérimentations. En général, l'évaluation de ROUGE sur les résumés montre que l'approche a une remarquable performance pour cette tâche en comparaison des résultats obtenus sur le corpus AQUAINT. Cela est dû à la nature des documents du corpus SCi-Summ qui convient à la méthode de la MT en ce qui concerne la longueur des documents et leur structure.

Nous avons fait varier les expérimentations en utilisant plusieurs configurations. L'application de la procédure de diminution de redondance, la nature des valeurs utilisées lors de la vectorisation des phrases et des requêtes et la forme des composants utilisés dans les vecteurs sont les paramètres principaux des profils appliqués.

En somme, les expérimentations qui ont donné les meilleurs résultats sont celles qui proposent des versions simplifiées de vecteurs (i.e vecteurs binaires, Représentation 3). Dans le cas de la comparaison entre la phrase et la requête dans la version binaire qui a réalisé les meilleurs résultats, nous avons extrait le seuil d'importance à la suite de la génération des scores de la MT en n'ajoutant au sac de mots de comparaison que les mots ayant des scores qui dépassent ce seuil.

L'expérimentation qui se sert des termes aux lieu des mots a également réalisé un des meilleurs scores grâce à sa capacité à fournir des vecteurs courts. La plupart des requêtes et des phrases mises en correspondance n'ont en commun que deux ou trois mots, ce qui explique pourquoi la vectorisation la plus simple est la plus efficace.

Notons que les scores obtenus dans les tableaux 4.2 et 4.3, sont très proches et que la seule différence entre eux est l'application de la procédure de diminution de la redondance. Cela nous permet de remarquer une nouvelle fois la remarquable performance de la méthode de la MT pour générer un résumé non redondant.

TABLE 4.2 – Les résultats de notre méthode **avec** l'application de la diminution de redondance en utilisant la **Représentation 1** (Valeurs de F-Mesure de trait de la section).

	Rappel	Précision	F-mesure
ROUGE-1	0,489387	0,428087	0,427332
ROUGE-2	0,181257	0,146823	0,152146
ROUGE-3	0,110385	0,081786	0,088691
ROUGE-4	0,090158	0,063947	0,070674
ROUGE-L	0,455041	0,397189	0,39667
ROUGE-S*	0,239932	0,175776	0,160392
ROUGE-W-1.2	0,120873	0,19966	0,140738

TABLE 4.3 – Les résultats de notre méthode **sans** l'application de la diminution de redondance, en utilisant la **Représentation 1** (Valeurs de F-Mesure de trait de la section).

	Rappel	Précision	F-mesure
ROUGE-1	0,482337	0,441186	0,430337
ROUGE-2	0,202945	0,1647	0,169299
ROUGE-3	0,137149	0,1011	0,108852
ROUGE-4	0,11752	0,083411	0,091246
ROUGE-L	0,45173	0,410144	0,401069
ROUGE-S*	0,244899	0,185832	0,165638
ROUGE-W-1.2	0,123589	0,2112	0,144905

TABLE 4.4 – Les résultats de notre méthode **sans** l'application de la diminution de redondance, en utilisant la **Représentation 2** (Valeurs maximales de F-mesure de trait).

	Rappel	Précision	F-mesure
ROUGE-1	0,469711	0,441958	0,427079
ROUGE-2	0,177244	0,152965	0,153839
ROUGE-3	0,108025	0,086128	0,090166
ROUGE-4	0,090752	0,070646	0,074673
ROUGE-L	0,442192	0,414388	0,401049
ROUGE-S*	0,229933	0,187493	0,165037
ROUGE-W-1.2	0,123428	0,221741	0,148487

TABLE 4.5 – Les résultats de notre méthode **avec** l'application de la diminution de redondance, en utilisant la **Représentation 2** (Valeurs maximales de F-mesure de trait) avec les **termes (n-grams)**.

	Rappel	Précision	F-mesure
ROUGE-1	0,530658	0,444783	0,448867
ROUGE-2	0,267323	0,216231	0,221787
ROUGE-3	0,20393	0,163283	0,168538
ROUGE-4	0,182847	0,14561	0,150541
ROUGE-L	0,500638	0,418554	0,422587
ROUGE-S*	0,288997	0,192504	0,181576
ROUGE-W-1.2	0,14672	0,241613	0,168754

TABLE 4.6 – Les résultats de notre méthode **sans** l'application de la diminution de redondance, en utilisant la **Représentation 2** (Valeurs maximales de F-mesure de trait) avec les **termes et les mots**.

	Rappel	Précision	F-mesure
ROUGE-1	0,511981	0,422193	0,434146
ROUGE-2	0,213993	0,161666	0,171087
ROUGE-3	0,143964	0,100296	0,109144
ROUGE-4	0,122566	0,082603	0,09077
ROUGE-L	0,47661	0,391267	0,402704
ROUGE-S*	0,26785	0,172567	0,168379
ROUGE-W-1.2	0,129109	0,201744	0,146779

Chapitre 5

Résumé de communauté : MT à base de graphe avec la propagation d'activation

Dans ce chapitre, nous allons ajouter à notre système un algorithme inspiré de la théorie des graphes. Le problème est le même que pour l'expérimentation précédente [voir section 4] et le corpus d'expérimentation est donc identique également. Il s'agit de générer des résumés extractifs de communauté. Pour rappel, nous nous intéressons à la Tâche 1 décrite Page 29 : Résumé de communauté CL-SciSumm.

Pour la plupart des citations issues des documents de D, on retrouve deux ou trois mots communs avec les phrases de D_r qui sont citées. Toutefois, la formulation de la citation ne ressemble à aucune phrase de D_r. Par exemple, la citation suivante est attachée à la phrase 11 du document C90-2039.

La citation : *"That is, unless some new scheme for reducing excessive copying is introduced such as scucture-sharing of an unchanged shared-forest ([Kogure, 1990])."*

Les phrases : *"For example, a spoken Present, Japanese analysis system based on llPSG [Kogure, 891] uses 90% - 98% of the elapsed time in FS unification."*

Cette différence entre la citation et la phrase de D_r est l'une des limitations principales de notre expérimentation précédente, et nous oblige à approcher le problème différemment, en utilisant les mots des citations, mais également ceux qui en sont *proches*.

Nous abordons ce problème comme un problème de résumé se basant sur une requête [voir section 2.3]. D'autre part, nous allons représenter les composants des articles (les mots ou les termes), comme un graphe de nœuds et d'arêtes. Nous définissons donc un Graphe $G = (V, E, W)$ où V représente l'ensemble des nœuds. Les nœuds représentent des composants (mots ou termes) et ont pour attribut les scores de la MT pour ces composants. Par ailleurs, pour toute paire $(u, v) \in V \times V$, on a $(u, v) \in E$ s'il existe une arête entre les nœuds u et v, i.e. si les composants apparaissent au moins dans une phrase ensemble. Enfin, W est une fonction de E vers l'ensemble des réels qui est définie en utilisant le nombre de phrases en commun des deux composants représentés par les noeuds à l'extrémité de l'arête et la moyenne de la distance entre ces composants dans ces phrases.

L'algorithme de la propagation d'activation a pour but de modifier les attributs sur les nœuds (leur valeur de F-Mesure de trait) des composants proches de ceux qui forment la requête, les nœuds excitants. L'algorithme agit comme un PageRank enraciné où l'activation des nœuds de la requête est propagée de façon à mettre en lumière les nœuds du réseau qui y sont fortement connectés. Une fois ces nœuds découverts, nous modifions leurs valeurs de F-mesure de trait afin de favoriser les phrases les plus proches lors de l'application de la métrique de similarité cosinus.

5.1 La propagation d'activation

La propagation d'activation (PA) est un algorithme de recherche dans les réseaux, qu'ils soient sociaux, lexicaux, sémantiques, ou biologiques. Cet algorithme a été inspiré par un mécanisme supposé de la mémoire humaine, et il suppose, dans notre cas, qu'il y a une relation sémantique derrière la distribution statistique des mots, des documents, etc [Crestani, 1997].

Cet algorithme a été adopté dans de nombreux domaines comme l'intelligence artificielle, les sciences cognitives, la psychologie, la biologie et les systèmes d'information, sa structure étant modifiée au besoin du domaine.

5.1.1 Modèle pur de la PA

Le modèle pur permet de représenter les données sous forme de réseau et d'exploiter cette structure. Les nœuds font référence aux objets du monde réel et stockent une partie de ses informations. La connectivité des nœuds reflète les relations de ces objets. Les arêtes peuvent avoir des directions et des poids. En général, cette structure est donc un modèle à base de graphe [Crestani, 1997].

L'algorithme qui agit sur cette structure est défini comme une série d'itérations. Chaque itération vient à la suite de la précédente jusqu'au moment où la condition de terminaison est établie.

Chaque itération est composée de trois étapes principales :

1. Le préajustement ;
2. La propagation ;
3. Le post-ajustement.

La première et la dernière étapes sont optionnelles et consistent à préparer la structure des nœuds pour la deuxième étape. La propagation applique une ou plusieurs vagues d'activation aux nœuds connectés aux nœuds d'activation.

L'équation suivante met en avant la procédure d'activation et la valeur d'entrée de chaque nœud :

$$I_j = O_i \cdot W(i,j) \tag{5.1}$$

Où I_j est l'entrée du nœud $\boldsymbol{j}$, O_i est la sortie du nœud $\boldsymbol{i}$ connecté au nœud $\boldsymbol{j}$ et $W(i,j)$ est le poids de l'arête entre $\boldsymbol{i}$ et $\boldsymbol{j}$.

Les valeurs sont des nombres réels, des valeurs binaires ou toute sorte de valeurs numériques adaptées à la mise en œuvre de l'algorithme.

Ensuite, les valeurs de sortie sont calculées comme une fonction des valeurs d'entrée :

$$O_j = \gamma(I_j) \tag{5.2}$$

Une des fonctions les plus connues est celle du seuil, où le seuil k_j est à surmonter :

$$O_j = \begin{cases} 0 & I_j < k_j \\ 1 & I_j > k_j \end{cases} \tag{5.3}$$

Puis, le nœud propage sa valeur de sortie à tous ses voisins. L'itération de cette procédure de propagation active la plupart des nœuds du réseau jusqu'au moment de la terminaison. Le résultat aboutit à un même réseau de nœuds, mais avec de nouvelles valeurs qui déterminent la *proximité* des noeuds avec ceux qui ont été activés.

5.1.2 Les modèles de rétroaction

De nombreuses modifications ont été proposées pour améliorer le modèle précédent. Par exemple, le contrôle de la distance de la propagation est appliqué en utilisant une classification des relations parmi les nœuds.

Les parcours composés d'une seule arête sont labellisés comme des relations de premier degré, ceux qui en contiennent deux sont labellisés comme des relations de deuxième degré, etc. D'autres modifications ont été adoptées pour former de nouveaux modèles, ceux de rétroaction. Les informations en retour de ces modèles ont été utilisées dans l'objectif de modifier la force de l'activation, la nature de la fonction de seuil et les poids des nœuds et des arêtes, que ce soit à l'étape du préajustement ou à celle du post-ajustement.

5.2 Système de RA à base d'un Modèle de la PA

Dans l'intention d'exploiter notre modèle spécialisé à base de la PA, nous devons construire le graphe qui représente l'article. Nous allons décrire ce modèle et notre algorithme de PA.

5.2.1 Modèles à base de graphe

Les modèles à base de graphe sont composés de nœuds et d'arêtes qui représentent les liens entre ces nœuds. Les arêtes sont non orientées dans notre cas et ont des valeurs numériques qui reflètent la similarité de chaque paire de nœuds. La flexibilité de cette représentation offre le bénéfice de pouvoir utiliser des algorithmes génériques et efficaces bien connus dans le domaine de la théorie des graphes, comme celui de la PA [voir sec. 2.2].

Nous allons proposer deux versions de notre modèle selon les nœuds choisis. Et nous allons l'exploiter en utilisant deux manières pour calculer les valeurs de F-mesure de trait.

Pour ce qui est des poids des arêtes, nous utilisons l'équation suivante :

$$W((u,v)) = \ (|N|+1)\cdot 100 - \frac{\sum_{i\in 1,\cdots,p} Di}{|N|} \tag{5.4}$$

Où $N = (N_1, \cdots, N_p)$ est l'ensemble des phrases communes entre les composants représentés par les nœuds u et v, Di est la distance entre u et v dans chaque phrase N_i de N. Par exemple, si les deux mots « unification » et « process » se trouvent dans cinq phrases ensemble et si la distance entre eux dans chaque phrase est [2, 4, 2, 9, 6], alors la valeur de l'arête est :

$$W(\text{"}unification\text{"}, \text{"}process\text{"}) = (5+1)\cdot 100 - (2+4+2+9+6)/5 = 596 \tag{5.5}$$

Nous avons utilisé le nombre de phrases en commun $|N|$ de chaque paire de nœuds et la moyenne de la distance entre eux parce que ces paramètres ont été utilisés dans de nombreux systèmes de RA, notamment ceux à base d'apprentissage automatique [voir sec. 2.3]. D'ailleurs, nous avons amplifié l'effet du premier attribut $|N|$ parce qu'il reflète les relations des mots par rapport au résumé entier, tandis que le second, la distance entre les mots, les reflète par rapport à une phrase en particulier. Le second attribut a cependant été ajouté dans l'objectif de donner un ordre plus précis des arêtes entre nœuds.

Voici l'algorithme de la construction du graphe :

for *chaque composant de l'article c* **do**
  for *chaque nœud des phrases dans lesquelles est c* **do**
    if *n'y a pas d'arête entre les deux nœuds* **then**
      la créer ;
    end
    Else modifier les informations sur l'arête.
  end
end

Algorithm 4: l'algorithme simplifié de la construction de notre graphe.

5.2.2 Notre Modèle de PA

La procédure d'activation de notre modèle est composée d'une étape de préajustement et d'une autre pour l'activation. Dans la première, nous déterminons les nœuds du graphe à partir desquels l'activation commence. Nous générons un sac de composants de la citation donnée : un sac des nœuds excitants. Il détermine les parties du graphe à activer.

L'extraction de ce sac dépend de la nature des composants du modèle à base de graphe. Par exemple, avec un graphe à base de mots, ['scheme', 'excessive', 'copying', 'scucture-sharing', 'Kogure'] seront les composants excitants de la citation indiquée ci-dessus [voir sec. 5.1] en supposant que les mots « reducing », « introduced » ne sont pas importants selon la MT.

D'autre part, nous proposons deux procédure d'activation.

La première est semblable à la propagation de la chaleur dans tout matériau conducteur. Elle consiste à activer le voisinage des nœuds du sac précédent en **maximisant** la valeur de leur attribut F initialisé par leur valeur de F-mesure de trait.

Procédure 1 (maximisation) *La mise à jour de la valeur de l'attribut F pour tout nœud v connecté à un nœud u activé est faite comme suit :*

$$F(v)_{nouveau} = F(v)_{ancien} * (k_1 + W(u,v)/k_2) \tag{5.6}$$

Où $F(v)_{nouveau}$, $F(v)_{ancien}$ sont les nouvelle et ancienne valeurs de F-mesure de trait, $W(u,v)$ est le poids de l'arête entre les deux composants, k_1, k_2 sont des valeurs expérimentales.

La seconde sert à maximiser les attributs des nœuds de la requête, tandis qu'elle **minimise** les valeurs des nœuds attachés aux nœuds des requêtes, mais seulement ceux qui ne se trouvent pas dans les requêtes. De cette manière, les nœuds qui se trouvent dans les requêtes jouent un rôle principal pour déterminer les poids des phrases.

Procédure 2 (minimisation) *La valeur mise à jour de l'attribut F d'un nœud v connecté à un autre noeud u activé est calculé comme suit :*

$$F(v)_{nouveau} = \begin{cases} F(v)_{ancien}{}^{k_1+W(u,v)/k_2} & \text{où } v \in C_{req} \\ F(v)_{ancien} * k_3 & \text{où } v \notin C_{req} \end{cases} \tag{5.7}$$

Où C_{req} est l'ensemble des nœuds de la requête, k_1 ,k_2, k_3 sont des valeurs expérimentales.

Le choix du type de composant et de la procédure d'activation permettent de déterminer plusieurs versions de notre méthode :

Modèle 1 (mots et maximisation) *La première version se sert de tous les mots non vides de l'article. D'ailleurs, les mots vides, les articles, les nombres et les ponctuations ne font partie d'aucune version à cause de leur faible valeur au niveau sémantique. Les valeurs de F-mesure de trait sont produites en utilisant la Procédure 1 (maximisation).*

Modèle 2 (mots et minimisation) *La deuxième version utilise le même graphe à base de mots mais la Procédure 2 (minimisation) d'activation est appliquée.*

Modèle 3 (termes et minimisation) *La troisième version remplace les mots par les termes. Nous extrayons donc les 2-grammes de l'article pour construire les nœuds. Ensuite, la la Procédure 2 (minimisation) d'activation est appliquée.*

Ces différents modèles capturent de nombreux aspects de notre problème. Le premier exploite la plupart des mots de la requête afin de comparer deux phrases entières, tandis que le deuxième modifie profondément l'ordre et les poids des phrases. Le dernier exploite les termes seulement afin de réduire au mieux la dimension des vecteurs, ainsi que la citation à une série de termes. Notons qu'en réduisant le nombre de composants nous augmentons la rapidité de production du graphe.

5.3 Évaluation et discussion

L'évaluation de cette expérimentation a les mêmes étapes que celle de la précédente (voir sec. 4.4). En somme, nous utilisons la métrique ROUGE pour évaluer les résumés produits.

Les tableaux de cette section montrent les résultats des trois modèles mentionnés dans la section précédente. En général, l'évaluation automatique de ROUGE sur les résumés montre que la combinaison de l'algorithme de la propagation d'activation avec les modèles à base de graphe ont augmenté la performance de notre système en comparaison des résultats de l'expérimentation précédente.

Le Modèle 1 donne des scores acceptables de ROUGE, mais il ne dépasse pas les scores obtenus lors de l'expérimentation de la méthode du chapitre précédent (Voir tableau 5.1). Nous avons défini $k_1 = 4.2$, $k_2 = 10000$ à la suite de plusieurs expérimentations.

Le Modèle 2 réalise les meilleurs résultats (Voir tableau 5.2). La marginalisation des mots non connectés change l'ordre des phrases et favorise les phrases contenant un petit nombre de mots non connectés. Autrement dit, les plus proches de la requête. D'autre part, Nous avons défini $k_1 = 2$, $k_2 = 10000$, $k_3 = 5$ à la suite de plusieurs expérimentations.

Le Modèle 3, qui se sert des termes aux lieu des mots a également utilisé les mêmes valeurs de i k_1 , k_2 , k_3 et réalisé des scores remarquables grâce à sa capacité à produire des vecteurs courts [Voir tableau 5.3]. La plupart des requêtes s'attachent aux phrases qui n'ont en commun que deux ou trois mots, ce fait explique pourquoi la vectorisation simple produit ces résultats. De plus, cette expérimentation a montré une rapidité remarquable en comparaison d'autres expérimentations.

Table 5.1 – Résultats de ROUGE avec le Modèle 1 (mots et maximisation).

	Rappel	Précision	F-mesure
ROUGE-1	0,452918	0,536778	0,445446
ROUGE-2	0,248473	0,265707	0,232325
ROUGE-3	0,195819	0,19948	0,178013
ROUGE-4	0,177627	0,176695	0,159201
ROUGE-L	0,43109	0,506292	0,422395
ROUGE-S*	0,228108	0,266644	0,173306
ROUGE-W-1.2	0,128505	0,286701	0,163985

TABLE 5.2 – Résultats de ROUGE avec le Modèle 2 (mots et minimisation).

	Rappel	Précision	F-mesure
ROUGE-1	0,760024	0,294699	0,405691
ROUGE-2	0,41459	0,130564	0,188957
ROUGE-3	0,192741	0,139041	0,15149
ROUGE-4	0,17187	0,122583	0,134116
ROUGE-L	0,738114	0,283424	0,39102
ROUGE-S*	0,561456	0,090149	0,143792
ROUGE-W-1.2	0,229007	0,166463	0,181085

TABLE 5.3 – Résultats de ROUGE avec le Modèle 3 (termes et minimisation).

	Rappel	Précision	F-mesure
ROUGE-1	0,684212	0,350334	0,444289
ROUGE-2	0,362201	0,170364	0,220145
ROUGE-3	0,278965	0,127601	0,165935
ROUGE-4	0,249769	0,112614	0,146891
ROUGE-L	0,647165	0,330955	0,419435
ROUGE-S*	0,449065	0,121854	0,175001
ROUGE-W-1.2	0,187827	0,184229	0,176124

Chapitre 6

Conclusion

6.1 Discussion

La MT est une méthode de sélection de variables qui sert à choisir les traits les plus importants dans des données hétérogènes. Elle a été validée dans le cadre de l'apprentissage supervisé, mais également pour l'étiquetage de *clusters*, le clustering proprement dit, ou pour la mesure de qualité du *clustering* dans un contexte non supervisé. Cette méthode est particulièrement utilisée et efficace sur les données textuelles. Par ailleurs, elle est langage-agnostique, sans paramètres, rapide à calculer, et ne nécessite pas de corpus externe. Nous considérons donc dans ce mémoire son utilisation dans le cadre du RA.

Notre travail a été divisé en trois expérimentations. La première était de générer un résumé extractif d'un seul document. Nous avons été obligés d'adapter notre méthode à un corpus destiné au résumé de documents multiples à cause du manque de ressources linguistiques adaptées à notre approche du résumé de documents semi-structurés. Le système a dépassé le score du système de base, et les résultats sont encourageants même si des progrès restent à faire dans ce contexte. La nature du corpus utilisé, la brièveté de ses documents et l'absence d'un corpus destiné à ce genre d'expériences défavorisent en effet notre méthode qui bénéficie néanmoins des atouts précédemment listés. Ces expérimentations mettent également en valeur la performance de la MT pour générer naturellement des résumés non redondants.

Les résultats de la deuxième expérimentation sur un corpus plus adapté ont mieux permis de prouver la performance de notre approche. Le corpus CL-SciSumm propose en effet de travailler sur des publications scientifiques semi-structurées, particulièrement adaptées à l'utilisation de la MT. Pour cet expérimentation imposée par le challenge CL-SciSumm duquel provient le corpus, nous avons travaillé sur la génération de résumés se basant sur une requête. Nous avons généré des résumés de communauté d'un article de référence en utilisant les phrases de cet article auxquelles faisaient référence une communauté d'autres articles. Pour mener à bien cette tâche, nous avons combiné notre approche avec la métrique de similarité cosinus.

Dans la troisième expérimentation dont l'objectif était le même que la seconde, nous avons introduit une méthode inspirée de la théorie des graphes pour améliorer nos résultats. Ce système consiste à modéliser les articles sous forme de graphes et d'appliquer un algorithme de propagation d'activation afin de mettre en lumière itérativement les nœuds proches de la requête.

Ces expérimentations ont toutes montré l'efficacité de la MT pour générer des résumés non redondants. Par ailleurs, ils ont montré sa flexibilité : la métrique a été utilisée pour résumer un seul document mais également pour les résumés de communauté. Les résultats sur des corpus adaptés composés de documents semi-structurés sont particulièrement encourageants, et l'approche à base de graphe montrent que des modèles plus complexes dans lesquels la MT est exploitée peuvent encore contribuer à l'amélioration de ces résultats.

6.2 Améliorations futures

Les résultats de la combinaison de notre approche avec des systèmes de RA sont très encourageants. Nous devons néanmoins poursuivre nos travaux afin d'explorer de façon plus exhaustive l'étendue des capacités, les limitations, et les performances de notre approche au regard de l'état de l'art du domaine du RA.

À court terme, nous envisageons d'améliorer la qualité des ressources linguistiques utilisées dans les expérimentations. Nous aimerions travailler avec un autre corpus dédié à la tâche du RA d'un seul document semi-structuré au lieu du corpus AQUAINT. Nous pensons qu'il est l'une des raisons principales des résultats assez modestes obtenus lors de notre première expérimentation.

Par ailleurs, nous souhaitons améliorer la normalisation du texte appliquée au corpus CL-SciSumm.

Nous pensons également à combiner nos systèmes avec des algorithmes de résolution de coréférence (Voir sec. 2.2.2) pour améliorer la performance des mesures de fréquence. D'autres techniques de désambiguïsation peuvent également être appliquées comme la compression des phrases.

Nous bénéficierons également sous peu des publications et des données issues du challenge CL-SCiSumm. Ces résultats nous permerront de comparer la performance de notre système avec celle des systèmes développées par les autres équipes ayant participé au challenge.

Enfin, nous considerons à long terme, la possibilité d'adapter notre modèle à base de graphe,utilisant l'algorithme de propagation d'activation, au domaine du RA d'un seul document et de documents multiples.

Bibliographie

[Branny, 2007] Branny, E. (2007). Automatic summary evaluation based on text grammars. *Journal of Digital Information*, 8(3).

[Clauset et al., 2009] Clauset, A., Shalizi, C. R., and Newman, M. E. (2009). Power-law distributions in empirical data. *SIAM review*, 51(4) :661–703.

[Conroy and O'leary, 2001] Conroy, J. M. and O'leary, D. P. (2001). Text summarization via hidden markov models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 406–407. ACM.

[Crestani, 1997] Crestani, F. (1997). Application of spreading activation techniques in information retrieval. *Artificial Intelligence Review*, 11(6) :453–482.

[Edmundson, 1969] Edmundson, H. P. (1969). New methods in automatic extracting. *Journal of the ACM (JACM)*, 16(2) :264–285.

[Erkan and Radev, 2004] Erkan, G. and Radev, D. R. (2004). Lexrank : Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22 :457–479.

[Guyon and Elisseeff, 2003] Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar) :1157–1182.

[Guyon et al., 2002] Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1-3) :389–422.

[Haghighi and Vanderwende, 2009] Haghighi, A. and Vanderwende, L. (2009). Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies : The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 362–370. Association for Computational Linguistics.

[Hennig et al., 2010] Hennig, L., De Luca, E. W., and Albayrak, S. (2010). Learning summary content units with topic modeling. In *Proceedings of the 23rd International Conference on Computational Linguistics : Posters*, pages 391–399. Association for Computational Linguistics.

[Katragadda, 2010] Katragadda, R. (2010). Gems : generative modeling for evaluation of summaries. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 724–735. Springer.

[Kupiec et al., 1995] Kupiec, J., Pedersen, J., and Chen, F. (1995). A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 68–73. ACM.

[Ladha and Deepa, 2011] Ladha, L. and Deepa, T. (2011). Feature selection methods and algorithms. *International journal on computer science and engineering*, 1(3) :1787–1797.

[Lamirel, 2008] Lamirel, J.-C. (2008). Combination of hyperbolic visualization and graph-based approach for organizing data analysis results : an application to social network analysis. In *4th International Conference on Webometrics, Informetrics and Scientometrics and 9th Collnet Meeting*.

[Lamirel et al., 2015a] Lamirel, J.-C., Cuxac, P., Chivukula, A. S., and Hajlaoui, K. (2015a). Optimizing text classification through efficient feature selection based on quality metric. *Journal of Intelligent Information Systems*, 45(3) :379–396.

[Lamirel et al., 2015b] Lamirel, J.-C., Dugué, N., and Cuxac, P. (2015b). Performing and visualizing temporal analysis of large text data issued for open sources : Past and future methods. In *International Conference : Beyond Databases, Architectures and Structures*, pages 56–76. Springer.

[Lamirel et al., 2011] Lamirel, J.-C., Mall, R., Cuxac, P., and Safi, G. (2011). Variations to incremental growing neural gas algorithm based on label maximization. In *Neural Networks (IJCNN), The 2011 International Joint Conference on*, pages 956–965. IEEE.

[Lin, 2004] Lin, C.-Y. (2004). Rouge : A package for automatic evaluation of summaries. In *Text summarization branches out : Proceedings of the ACL-04 workshop*, volume 8. Barcelona, Spain.

[Lin and Hovy, 2000] Lin, C.-Y. and Hovy, E. (2000). The automated acquisition of topic signatures for text summarization. In *Proceedings of the 18th conference on Computational linguistics-Volume 1*, pages 495–501. Association for Computational Linguistics.

[Lloret, 2015] Lloret, E. (2015). *Text summarisation based on human language technologies and its applications*. PhD thesis, Universidad de Alicante.

[Luhn, 1958] Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of research and development*, 2(2) :159–165.

[Mani et al., 1999] Mani, I., Gates, B., and Bloedorn, E. (1999). Improving summaries by revising them. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 558–565. Association for Computational Linguistics.

[Minel et al., 1997] Minel, J.-L., Nugier, S., and Piat, G. (1997). How to appreciate the quality of automatic text summarization ? examples of fan and mluce protocols and their results on seraphin. In *Proc. of the ACL/EACL'97 Workshop on Intelligent Scalable Text Summarization*, pages 25–30. Citeseer.

[MParUuez and Salgado, 2000] MParUuez, L. and Salgado, J. G. (2000). Machine learning and natural language processing.

[Nenkova et al., 2011] Nenkova, A., Maskey, S., and Liu, Y. (2011). Automatic summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics : Tutorial Abstracts of ACL 2011*, page 3. Association for Computational Linguistics.

[Nenkova et al., 2007] Nenkova, A., Passonneau, R., and McKeown, K. (2007). The pyramid method : Incorporating human content selection variation in summarization evaluation. *ACM Transactions on Speech and Language Processing (TSLP)*, 4(2) :4.

[Nist, 2007] Nist, D. (2007). Duc 2007 : Task, documents, and measures.

[Paice, 1980] Paice, C. D. (1980). The automatic generation of literature abstracts : an approach based on the identification of self-indicating phrases. In *Proceedings of the 3rd annual ACM conference on Research and development in information retrieval*, pages 172–191. Butterworth & Co.

[Qazvinian and Radev, 2008] Qazvinian, V. and Radev, D. R. (2008). Scientific paper summarization using citation summary networks. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 689–696. Association for Computational Linguistics.

[Radev and Tam, 2003] Radev, D. R. and Tam, D. (2003). Summarization evaluation using relative utility. In *Proceedings of the twelfth international conference on Information and knowledge management*, pages 508–511. ACM.

[Ravindra et al., 2006] Ravindra, G., Balakrishnan, N., and Ramakrishnan, K. (2006). Methods for automatic evaluation of sentence extract summaries. In *Proceedings of the 2nd International Conference on Universal Digital Library, Alexandria, Egypt*.

[Rijsbergen, 1981] Rijsbergen, V. (1981). *information retrieval, elsevier science and technology*.

[Roberts, 2003] Roberts, A. (2003). Machine learning in natural language processing.

[SCI-Summ, 2016] SCI-Summ, N. U. o. S. W. (2016). The 2nd computational linguistics scientific document summarization shared task (cl-scisumm 2016).

[Tata and Patel, 2007] Tata, S. and Patel, J. M. (2007). Estimating the selectivity of tf-idf based cosine similarity predicates. *ACM Sigmod Record*, 36(2) :7–12.

[Teufel and Van Halteren, 2004] Teufel, S. and Van Halteren, H. (2004). Evaluating information content by factoid analysis : Human annotation and stability. In *EMNLP*, pages 419–426.

[Tratz and Hovy, 2008] Tratz, S. and Hovy, E. (2008). Bewte : basic elements with transformations for evaluation. In *TAC 2008 Workshop*.

[Vanderwende et al., 2007] Vanderwende, L., Suzuki, H., Brockett, C., and Nenkova, A. (2007). Beyond sumbasic : Task-focused summarization with sentence simplification and lexical expansion. *Information Processing & Management*, 43(6) :1606–1618.

[Varadarajan and Hristidis, 2006] Varadarajan, R. and Hristidis, V. (2006). A system for query-specific document summarization. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 622–631. ACM.

[Witten and Frank, 2005] Witten, I. H. and Frank, E. (2005). *Data Mining : Practical machine learning tools and techniques*. Morgan Kaufmann.

[Zweigenbaum et al., 2012] Zweigenbaum, P., Wisniewski, G., Marco, D., Grouin, C., and Rosset, S. (2012). Résolution des coréférences dans des comptes rendus cliniques. une expérimentation issue du défi i2b2/va 2011. In *RFIA 2012 (Reconnaissance des Formes et Intelligence Artificielle)*, pages 978–2.