S'informer sur les médias sociaux via l'élaboration d'information synthétique

Alexis Dusart

Université de Toulouse

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par : *l'Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)*

Présentée et soutenue le *16/12/2022* **par :**

Alexis DUSART

S'informer sur les médias sociaux via l'élaboration d'information synthétique

JURY

Catherine BERRUT	Professeure à l'Université Grenoble Alpes	Rapporteure
Mohand BOUGHANEM	Professeur à l'Université Toulouse 3	Président
Gilles HUBERT	MCF-HDR à l'Université Toulouse 3	Co-Directeur de thèse
Karen PINEL-SAUVAGNAT	MCF-HDR à l'Université Toulouse 3	Co-Directrice de thèse
Jacques SAVOY	Professeur ordinaire à l'Université de Neuchâtel, Suisse	Rapporteur
Laure SOULIER	MCF à la Sorbonne Université	Examinatrice

École doctorale et spécialité :
MITT : Image, Information, Hypermédia
Unité de Recherche :
Institut de Recherche en Informatique de Toulouse (UMR 5505)
Directeur(s) de Thèse :
Gilles HUBERT et *Karen PINEL-SAUVAGNAT*
Rapporteurs :
Catherine BERRUT et *Jacques SAVOY*

S’informer sur les médias sociaux via l’élaboration d’information synthétique

Alexis DUSART

7 décembre 2022

REMERCIEMENTS

L'enfer c'est les autres, le paradis c'est vous.

— Alexis Dusartre

Tout d'abord, je tiens à remercier Karen Pinel-Sauvagnat et Gilles Hubert, les meilleurs directeur.rice.s de thèse que j'aurais pu imaginer. Meilleurs par rapport à quelle mesure ? Toutes celles qui suscitent des résultats positifs ! Merci pour la confiance que vous me témoignez, moi le petit béotien qui ne savait même pas écrire une phrase avec un verbe conjugué... Merci d'avoir cru en moi pour décrocher cette bourse de thèse, déjà là ce n'était pas gagné. Merci pour tout ce que vous m'avez appris, de lire à écrire en passant par la rigueur scientifique et tellement plus. Bien que la thèse soit un exercice difficile, merci d'en avoir permis un déroulement agréable, dans la bonne humeur. Merci :)

Je remercie M^me^ Catherine Berrut, Professeure à l'Université Grenoble Alpes, et M. Jacques Savoy, Professeur ordinaire à l'Université de Neuchâtel pour avoir accepté d'être rapporteur.e.s de ma thèse, merci d'avoir passé du temps à lire mes travaux et d'avoir apporté des retours constructifs et détaillés à mon travail. Je remercie également M^me^ Laure Soulier, MCF à la Sorbonne Université, et M. Mohand Boughanem, Professeur à l'Université Paul Sabatier de Toulouse, pour avoir accepté d'être examinateur.rice.s de ma thèse.

Je tiens à remercier l'ensemble de l'équipe enseignante de la formation SID de l'Université Paul Sabatier, et particulièrement Gilles Hubert, Karen Pinel-Sauvagnat, M^me^ Cécile Chouquet, M. Franck Morvan et M. José Moreno. Merci José de m'avoir dit avec la sincérité qui te caractérise que j'avais les capacités pour faire une thèse. Je remercie également l'ensemble des personnes de l'IRIT pour leur accueil, et surtout les membres de l'équipe IRIS. Merci Guillaume pour tes précieux conseils de présentation, tant au niveau de l'image que du discours. Merci Paul pour ton accueil, ta sympathie et tes conseils inestimables à mon début de thèse. Tu m'avais demandé de perpétuer la tradition du bureau, avec Maël et Nihed l'héritage sera transmis. Maël, merci pour ta bienveillance. Merci Luis, tu as été un co-bureau exceptionnel. Malik, merci d'avoir partagé ton bureau avec nous et diverti mes journées. Merci à ceux qui ont partagé leurs repas avec moi pendant ces années de thèse. Je remercie particulièrement les doctorants qui s'engagent pour les autres, notamment ceux qui participent activement à la commission, merci Damien, Morgan, Olivier,...

Je n'oublie pas l'ensemble du personnel de l'IRIT, de l'Université Paul Sabatier, et de l'École Doctorale, pour leur travail qu'on oublie souvent, de l'administratif aux tâches ménagères, qui ont permis au bon déroulement de cette thèse, merci à vous.

Hurt me with the truth, but never comfort me with a lie.

— Erza Scarlet

Parce qu'il m'aura fallu un déclic pour m'épanouir dans mes études, merci Émilie, Marine, et Marine. Merci pour vos regards, vos sourires. Émilie, merci pour la confiance que tu m'as porté, le réconfort que tu m'as apporté, ça part de toi ! Marine, merci pour ton soutien, l'inscription à la danse, quelle bonne idée ! Marine, merci pour ta confiance, tu as été la cerise sur mon gâteau ! Parce que tu m'as écouté, parce que ton regard, ton sourire, ont éclipsé beaucoup de difficultés pendant cette thèse, Alice, merci.

Les amis, la vie d'adulte ne nous permet plus de passer autant de temps ensemble, mais merci à vous d'être vous. Il m'aurait été impossible de réussir cette thèse sans votre soutien, sans les moments passés avec vous. Paul, Kévin, Guillaume, et aussi Aurélien, Tiphaine, Grégoire, MERCI ! Anna (Étude Conduite), ton regard, ton sourire, ton écoute, tes attentions,... donnent à cette fin de thèse une saveur incroyablement agréable, merci !

Maman, Papa, Mélissa, Lucas, Mamie, Monique, Marie-Pierre, Pascal, Nathan, Ghislaine, Philippe, Auriana,... toute la famille qui m'a vu grandir, qui a toujours été là pour moi, MERCI ! Si j'ai pu m'investir autant dans mes études, c'est parce que je sais que vous êtes là pour me soutenir.

Enfin, je termine ces remerciements par la personne la plus importante de cette thèse. Merci à moi d'avoir traversé les difficultés de la thèse en gardant plaisir à me lever le matin. P.S. à moi-même : Ne pas prendre la grosse tête (je l'ai déjà), passer à autre chose la vie n'est pas finie, continuer de se rendre fier.

Assume. Deviens meilleur.

— Moi

RÉSUMÉ

Les médias sociaux regorgent d'informations qui peuvent être utiles et d'intérêt. Cependant, la trop grande quantité de données présentes peut demander au lecteur un effort fastidieux afin d'accéder à l'information pertinente. Une synthèse de cette information apparaît utile voir nécessaire pour l'utilisateur afin de s'informer sans être submergé. Nous nous penchons ainsi dans ces travaux sur l'élaboration de résumés à partir du média social Twitter. L'état de l'art du résumé automatique utilise aujourd'hui des modèles de langue neuronaux, comme beaucoup de tâches du Traitement Automatique des Langues (TAL). Cependant, ces modèles ne sont pas utilisés pour le résumé automatique de flux de tweets. Ceci peut être expliqué par : (i) la difficulté de créer des jeux d'apprentissage de taille suffisante et adéquats pour ces modèles, (ii) la taille du texte à résumer, qui ne permet pas d'appliquer directement les modèles comme dans le cas de résumé classique.

Dans ces travaux, nos contributions sont les suivantes :

— Nous proposons deux corpus de tweets pouvant être utilisés pour l'apprentissage des modèles neuronaux. Le premier, de plus de 80 millions de tweets, est construit avec une méthode faisant également partie de nos contributions, mettant en œuvre peu d'efforts humains. Le second, non soumis au problème de la suppression de tweets, est une adaptation d'un jeu de données utilisé à l'origine pour le filtrage d'information.
— Nous présentons un modèle de résumé automatique de flux de tweets basé sur un modèle de langue neuronal. Nous ajoutons la fréquence des *tokens* du flux pour représenter le contexte flux de tweets.
— Enfin, afin de mieux comprendre les mécanismes mis en place lors du résumé et de permettre le développement de futures méthodes de résumé plus efficaces, nous explorons les représentations pré-établies de l'état de l'art pour le résumé automatique de flux de tweets.

ABSTRACT

Social media are full of information, which can be useful, of interest. However, the large amount of data present can require the reader to make a tedious effort to access the relevant information. A synthesis of this information appears useful or even necessary for the user to be informed. In this work, we focus on the development of summaries from the social media Twitter. The state of the art of automatic summarization today uses neural language models, as do many Natural Language Processing (NLP) tasks. However, these models are not used for automatic tweet stream summarization. This can be explained by : (i) the difficulty to create training sets of sufficient size and adequate for these models, (ii) the size of the text to be summarized, which does not allow the models to be applied directly as in the case of usual summarization.

In this work, our contributions are as follows :

— We propose two collections of tweets that can be used for training neural models. The first one, containing more than 80 million tweets, is built with a method that is also part of our contributions, involving little human effort. The second, not subject to the problem of tweets deletion, is an adaptation of a dataset originally used for information filtering.
— We present a model for automatic tweet stream summarization based on a neural language model. We add the frequency of the stream tokens to represent the stream context of tweets.
— Finally, in order to better understand the mechanisms involved in summarization and to enable the development of future, more efficient summarization methods, we explore pre-established state-of-the-art representations for automatic summarization of tweet streams.

PUBLICATIONS

Article de conférence internationale avec comité de lecture

Alexis Dusart, Karen Pinel-Sauvagnat et Gilles Hubert. *ISSumSet : A Tweet Summarization Dataset Hidden in a TREC Track. (Article long)* Dans : Symposium On Applied Computing (SAC 2021), mars 2021, p. 665-671.

Article de conférence nationale avec comité de lecture

Alexis Dusart, Karen Pinel-Sauvagnat et Gilles Hubert. *Capitalizing on a TREC Track to Build a Tweet Summarization Dataset (Article long)* Dans : Joint Conference of the Information Retreival Communities in Europe (CIRCLE 2020), juillet 2020.

Article de campagne d'évaluation internationale

Alexis Dusart, Karen Pinel-Sauvagnat et Gilles Hubert. *Irit at trec 2019 : Incident streams and complex answer retrieval tracks (Article long)* Dans : Text REtrieval Conference (TREC 2019), novembre 2019.

Prépublication

Alexis Dusart, Karen Pinel-Sauvagnat et Gilles Hubert. *TSSuBERT : Tweet Stream Summarization Using BERT (Article long)* Dans : arXiv, Juin 2021.

En cours de publication

Alexis Dusart, Karen Pinel-Sauvagnat et Gilles Hubert. *TSSuBERT : How to Sum Up Multiple Years of Reading in a Few Tweets (Article long)* ACM Transactions on Information Systems (ACM TOIS) (Minor Revision Novembre 2022, resoumis).

RÉFLEXIONS ÉTHIQUES

Science sans conscience n'est que ruine de l'âme.

— François Rabelais

En prélude de ce manuscrit, j'aimerais partager quelques réflexions éthiques sur ces travaux de recherche, leurs conséquences et l'impact qu'ils ont eu et auront sur la société. Mes travaux ambitionnent d'élaborer une information synthétique à partir de médias sociaux. Pour l'imager assez simplement, nous nous sommes penchés sur la génération automatique de résumé à partir de tweets. Cependant, l'essence même de l'automatisation est de remplacer l'effort humain par une machine. Une question que l'on peut se poser par rapport aux impacts sur la société est de se demander si nos travaux d'automatisation du résumé d'information concurrencent des métiers existants, et si l'automatisation est la seule manière de répondre à ce besoin. Le métier qui se rapproche le plus de nos travaux semble être le journalisme. Au moment où ce manuscrit est écrit, les méthodes de résumé automatique sont encore loin de la qualité d'un résumé humain (El-Kassas *et al.*, 2021). Les résumés générés automatiquement sont, à l'heure actuelle, plus une aide pour un travail journalistique qu'un remplacement. Justement, notre problématique vient de l'impossibilité humaine à traiter un volume aussi important de données que celui que l'on trouve sur les médias sociaux, notamment lors de situations de crise où l'information des médias sociaux peut être utile à des services de secours si elle est traitée rapidement. Néanmoins, il faut faire attention à ce qu'il adviendra par la suite. Il y a effectivement des dérives possibles à la généralisation et à la banalisation d'automatisations de ce type, le travail journalistique, quand il est bien fait, étant (à mon avis et partagé je l'espère) essentiel à toute société.

Aussi, au moment où j'écris ces lignes, l'humanité est dans un contexte de crise écologique. Cependant, les modèles actuels de l'état de l'art et que nous utilisons dans ce manuscrit sont de plus en plus gourmands en ressources et en énergies. Cet article [1] de blog est intéressant pour en savoir un peu plus sur l'empreinte écologique des entraînements de modèles en science des données, et ce site [2] permet de visualiser nos actions du quotidien sur le climat. Notamment, alors que les voyages en avion

1. https://blog.link-value.fr/quelle-est-lempreinte-ecologique-des-entrainements-de-modeles-en-data-science-25aa07beb7a3, dernière consultation le 03/08/2022
2. `https://futur.eco/`, dernière consultation le 03/08/2022

sont actuellement pointés du doigt pour leurs émissions de CO2eq [3], les travaux de Strubell *et al.* (2019) ont montré que l'apprentissage du modèle BERT, mentionné et utilisé dans cette thèse, est équivalent en termes d'émissions de CO2eq à un vol transaméricain ("training BERT on GPU is roughly equivalent to a trans-American flight").

3. `https://bonpote.com/pourquoi-arreter-lavion-ne-devrait-plus-etre-un-debat/`, dernière consultation le 03/08/2022

TABLE DES MATIÈRES

LISTE DES FIGURES

LISTE DES TABLEAUX

1

INTRODUCTION

1 Contexte

Si vous voulez lire tous les tweets générés le 14 Novembre 2015 à propos des attaques terroristes de Paris, soit plus de 7 millions de tweets, à raison d'un tweet lu par seconde (ce qui est très rapide), il vous faudra plus de 81 jours sans aucune pause. Une synthèse de l'information que l'on peut trouver dans l'ensemble de ces tweets ne paraît-elle pas intéressante plutôt que de les lire dans leur totalité ? Cependant, ce problème est plutôt récent à notre échelle et il n'existe pas (encore ?) de méthode efficace pour synthétiser automatiquement un tel flux de données. Revenons synthétiquement sur l'origine de ce problème.

La fin de la préhistoire est indexée chronologiquement sur l'apparition des premiers écrits, car depuis l'Homme peut retranscrire les évènements passés, peut raconter l'Histoire. Cette naissance de l'écriture est estimée aux alentours de 3300 avant J.C. en Mésopotamie et son utilisation fut pendant longtemps réservée à une certaine élite qui pouvait apprendre à écrire (voir figure 1.1). En France, par exemple, la gratuité de l'enseignement primaire public n'apparaît qu'avec la loi du 16 Juin 1881 [1]. La naissance de l'imprimerie vers 1450 par Gutenberg va permettre l'industrialisation de l'écriture, avec une multiplication de la production de livres et la naissance des premiers journaux au début du XVII^{e} siècle [2]. Un peu plus tard, au XX^{e} siècle, la démocratisation de la radio dans les années 1930 [3] puis de la télévision dans les années 1960-1970 [4] vont permettre la diffusion de contenus audio et vidéo. Les contenus, textes, audios, et vidéos deviennent de plus en plus accessibles à tout un chacun, mais leur nombre reste limité de par le fait qu'ils sont créés par un petit nombre de personnes : les journalistes.

1. `https://www.education.gouv.fr/les-grands-principes-du-systeme-educatif-9842`, dernière consultation le 03/08/2022
2. `https://fr.wikipedia.org/wiki/Journal`, dernière consultation le 03/08/2022
3. `https://www.histoire-pour-tous.fr/inventions/746-invention-de-la-radio.html`, dernière consultation le 03/08/2022
4. `https://www.histoire-pour-tous.fr/inventions/744-invention-de-la-television.html`, dernière consultation le 03/08/2022
5. `https://fr.statista.com/infographie/22876/evolution-du-taux-alphabetisation-dans-le-monde/`, dernière consultation le 03/08/2022

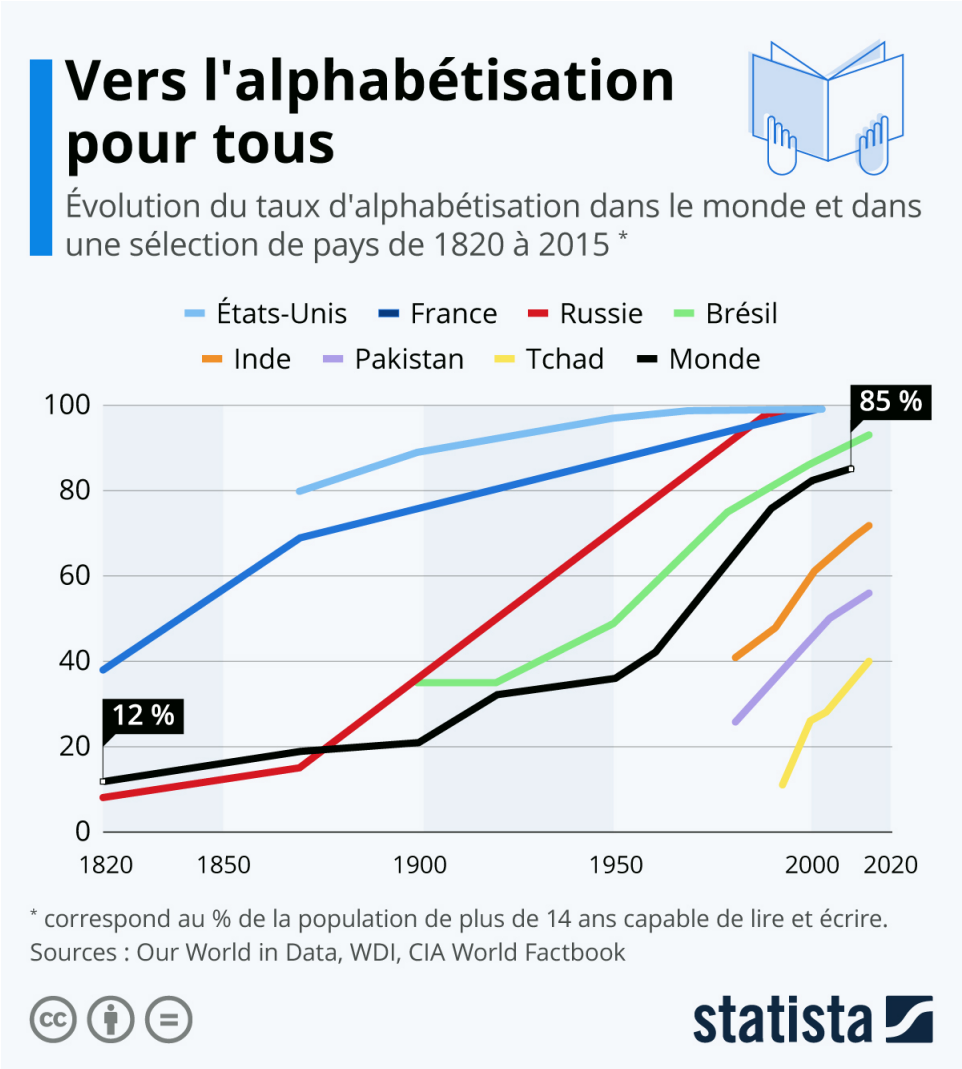


FIGURE 1.1 – Évolution du taux d'alphabétisation dans le monde entre 1820 et 2015 [5].

À partir des années 1960 émerge Internet, qui permet à tous ses utilisateurs de se connecter entre eux. Au début des années 1990, le Web, largement utilisé par les utilisateurs d'Internet fait son apparition (Berners-Lee *et al.*, 1992) et permet à Internet de se développer rapidement, au point d'atteindre 300 millions d'utilisateurs à la fin des années 1990 [6]. En 2021, Internet compte plus de 5 milliards d'utilisateurs dans le monde, soit environ 65 % de la population mondiale. L'architecture/philosophie du Web permet le développement de médias dont le contenu est généré par l'utilisateur (appelé UGC pour User Generated Content dans la littérature) et faciles d'utilisation, caractéristiques qui, entre autres, pousseront certains travaux à

6. `https://www.internetworldstats.com/emarketing.htm`, dernière consultation le 03/08/2022

employer, à partir du début des années 2000, le terme de Web 2.0 pour les définir [7]. Ce terme Web 2.0 met en lumière une possibilité nouvelle, un nouveau type de média, qui se répand de plus en plus à partir du début des années 2000 : grâce à Internet, tout le monde peut partager le contenu qu'il souhaite et le diffuser au monde entier facilement.

Ces médias qui permettent aux utilisateurs d'en être acteur, de partager, réagir, discuter, sont appelés médias sociaux. Les médias sociaux s'étendent des réseaux sociaux au partage de contenus audios, photos, vidéos diffusés en direct ou non, en passant par les blogs. Parmi les principaux médias sociaux nous pouvons citer Facebook [8], Twitter [9], YouTube [10], Wikipedia [11], LinkedIn [12], Twitch [13], TikTok [14] ou encore Instagram [15]. Ces dernières années, les médias sociaux ont pris de plus en plus de place dans le quotidien des utilisateurs d'Internet. Le Digital 2021 [16] rapporte un quasi doublement du nombre d'utilisateurs de médias sociaux entre 2016 et 2021, passant de 2,31 milliards à 4,2 milliards d'utilisateurs (pas forcément uniques). Un utilisateur passe en moyenne 2 heures 25 minutes par jour sur les médias sociaux, possède en moyenne 8,4 comptes différents. 40,4 % des utilisateurs utilisent les médias sociaux pour le travail. Comme le montre la figure 1.2, d'après ce rapport, la principale raison de l'utilisation des médias sociaux chez les 16-64 ans est de rester à jour des nouvelles et des évènements courants (devant : trouver du contenu amusant ou divertissant, occuper le temps libre, et rester au courant de ce que font ses amis).

Sur les médias sociaux le contenu est généré par les utilisateurs. Ce paradigme, associé à la facilité d'accès et d'utilisation des médias sociaux, notamment grâce aux smartphones, apporte des quantités de données très importantes. Par exemple, en 2020, 720 000 heures de contenu vidéo ont été ajoutées chaque jour sur YouTube, soit 30 000 heures de vidéos par heure [17], plus de 500 millions de tweets sont en-

7. le terme Web 2.0 est par ailleurs non reconnu par l'inventeur du Web Tim Berners-Lee pour qui le Web est initialement conçu pour les interactions entre les individus `https://web.archive.org/web/20120821185101/http://www.ibm.com/developerworks/podcast/dwi/cm-int082206txt.html`, dernière consultation le 03/08/2022

8. `https://fr-fr.facebook.com/`, dernière consultation le 03/08/2022

9. `https://twitter.com/`, dernière consultation le 03/08/2022

10. `https://www.youtube.com/`, dernière consultation le 03/08/2022

11. `https://fr.wikipedia.org/`, dernière consultation le 03/08/2022

12. `https://fr.linkedin.com/`, dernière consultation le 03/08/2022

13. `https://www.twitch.tv/`, dernière consultation le 03/08/2022

14. `https://www.tiktok.com/fr/`, dernière consultation le 03/08/2022

15. `https://www.instagram.com/`, dernière consultation le 03/08/2022

16. `https://datareportal.com/reports/digital-2021-global-overview-report`, dernière consultation le 03/08/2022

17. `https://blogfr.influence4you.com/les-chiffres-cles-de-la-celebre-plateforme-youtube-maj-en-2020/`, dernière consultation le 03/08/2022

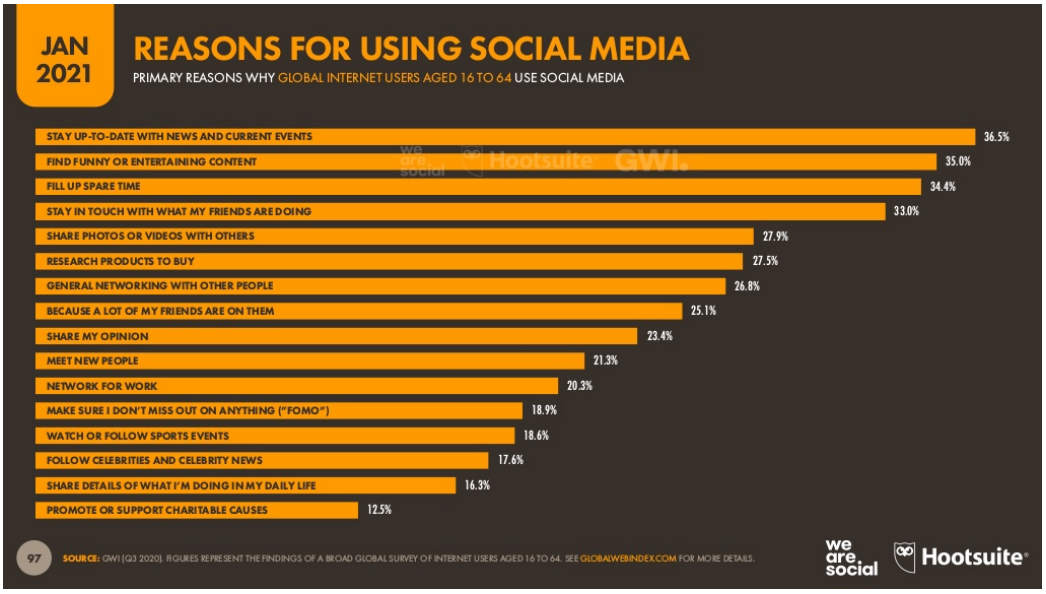


FIGURE 1.2 – Rapport Digital 2021 - raisons d'utilisation des médias sociaux chez les 16-64 ans.

voyés par jour, soit 184 milliards de tweets par an [18]. En conséquence, les médias sociaux regorgent d'informations, avec notamment des informations qui ne sont pas relayées par les médias traditionnels, ou des informations qui sont relayées plus rapidement que dans les médias traditionnels, particulièrement lorsque les évènements sont inattendus comme la rupture d'un barrage par exemple [19].

2 Problématique

Comme nous l'avons vu, les médias sociaux dont l'information est massive, véloce, sont un nouveau vecteur d'information pour les utilisateurs. Cependant, cette masse d'information amène aussi l'utilisateur à y consacrer beaucoup de temps. L'utilisateur tirerait profit d'une information synthétisée pour s'informer. C'est pourquoi, dans le cadre de cette thèse, nous abordons la problématique de la génération automatique d'information synthétique à partir de ces médias sociaux.

Synthétiser l'information soulève différentes problématiques liées à la recherche d'information et à la synthèse d'information, notamment relatives à :

18. `https://www.planetoscope.com/Internet-/1547-.html`, dernière consultation le 03/08/2022
19. `https://www.forbes.com/sites/kalevleetaru/2019/02/26/is-twitter-really-faster-than-the-news/`, dernière consultation le 03/08/2022

— la redondance d'information,
— la multiplicité des types d'information (texte, images, vidéos...),
— la complémentarité des éléments d'information utilisés en termes de thématiques abordées, d'éléments spatiaux et temporels apportés, par exemple,
— la validité, la crédibilité des éléments d'information manipulés,
— la construction et la restitution de l'information synthétisée sous forme structurée.

D'autre part, l'information des médias sociaux arrive de plus en plus souvent sous forme de flux continu.

Les questions soulevées par la synthèse d'information sont nombreuses, notamment lorsqu'on la considère dans un contexte temps-réel (prise en considération d'un délai imparti) en prenant en compte des sources d'information telles que Twitter.

Les récentes avancées en résumé automatique se basent sur des modèles neuronaux. Des modèles de langue neuronaux pré-entraînés (Devlin *et al.*, 2019) permettent de représenter le texte en prenant en compte son contexte. Cependant, ces méthodes sont très coûteuses en temps d'exécution et la taille du texte en entrée est limitée, généralement à moins d'un millier de mots. Or, la quantité de données générées sur les médias sociaux est bien souvent beaucoup plus importante que quelques centaines de mots pour un sujet. Dans cette thèse, nous apportons des réponses aux questions suivantes :

— Comment utiliser ces nouvelles méthodes de représentation textuelle pour le résumé automatique de flux de grandes quantités de données ?
— Comment intégrer le contexte du flux continu à la représentation du texte ?

3 Contributions

Le travail de cette thèse porte sur la génération automatique d'information synthétique. Nous nous focalisons sur le résumé automatique textuel. Nous utilisons des données du média Twitter, largement utilisées dans le domaine de la recherche sur les médias sociaux, de par sa popularité et l'accessibilité à ses données.

— Notre première contribution concerne la définition du cadre d'évaluation du résumé automatique de flux de tweets. Les jeux de données pour évaluer le résumé automatique de flux de tweets sont peu nombreux, et nous avons donc dans un premier temps proposé une méthode de création de jeux de données demandant peu d'efforts humains. Ainsi, nous avons créé un jeu de données de 80 millions de tweets, qui contient les flux de tweets d'évènements entre 2012 et 2016 sans classification en amont. Les vérités terrains de ce jeu de données sont générés à partir d'une ressource externe, le portail des actualités de Wikipedia

(Dusart *et al.*, 2021b). Dans un deuxième temps, nous proposons d'utiliser un jeu de données existant et destiné à une autre tâche afin de l'utiliser pour le résumé automatique de flux de tweets. Nous évaluons les potentielles vérités terrains de ce jeu de données par rapport aux propriétés d'un bon résumé. Ce jeu de données est à notre connaissance le premier jeu de données de tweets accessible facilement, et qui n'est pas soumis à la suppression de tweets. Nous montrons que les annotations de la tâche TREC Incident Streams (McCreadie *et al.*, 2019) sont utilisables sous certaines conditions comme vérité terrain pour le résumé de tweets (Dusart *et al.*, 2021a).

— Notre deuxième contribution est de proposer une approche pour profiter des méthodes de représentation textuelle récentes dans le cadre du résumé automatique de flux de tweets. Nous tirons profit des modèles de langue neuronaux pré-entraînés pour évaluer l'importance d'un tweet auxquels nous combinons la fréquence des mots du flux continu de tweets. Nous testons notre approche sur les jeux de données relatifs à notre première contribution, que nous évaluons par rapport à l'état de l'art (Dusart *et al.*, 2021b).

— Notre troisième contribution est l'exploration de représentations pré-établies pour le résumé automatique de flux de tweets. En effet, des études préliminaires ont montré que les représentations textuelles les plus récentes ne sont pas assurément plus adaptées à la représentation de textes bruités comme les tweets. Nous utilisons également les jeux de données établis dans notre première contribution pour cette exploration.

4 Organisation du manuscrit

Ce manuscrit est structuré en sept chapitres et trois parties. Le premier chapitre est le chapitre introductif (celui-ci). La première partie est composée de trois chapitres et présente une synthèse des travaux de l'état de l'art du sujet abordé. La deuxième partie est constituée de trois chapitres qui décrivent les contributions de cette thèse. Enfin, la dernière partie conclut le manuscrit et expose les pistes de réflexion qui feront l'objet de travaux futurs.

Plus précisément, la première partie, relative aux travaux de l'état de l'art, est composée des chapitres suivants :

— Le chapitre 2 expose une vue d'ensemble des médias sociaux et du filtrage d'information. Dans un premier temps, nous présentons en détail les médias sociaux et plus précisément Twitter. Ensuite, nous discutons de l'accès à l'information dans un contexte de médias sociaux. Enfin, nous détaillons le filtrage d'information et ses différences avec le résumé automatique.

— Le chapitre 3 expose un état de l'art des travaux en résumé automatique d'information textuelle. D'abord, nous définissons la notion de bon résumé, puis nous en présentons les principales classifications. Ensuite, nous détaillons les travaux de l'état de l'art du résumé automatique. Enfin, nous présentons l'état de l'art de l'évaluation du résumé automatique, qu'elle soit automatique ou manuelle.
— Le chapitre 4 expose un état de l'art des méthodes de résumé automatique avec prise en compte de la notion de temporalité. En premier lieu, nous présentons une classification des différentes prises en compte de la temporalité dans les résumés. En second lieu, nous détaillons l'état de l'art du résumé automatique de flux de tweets.

La deuxième partie expose les contributions que nous avons apportées dans le cadre de cette thèse, et est composée des chapitres suivants :

— Le chapitre 5 présente nos travaux pour l'établissement d'un corpus d'évaluation d'approche de résumé automatique de flux de tweets. Dans un premier temps, nous détaillons les motivations à ces travaux, les jeux de données existants et leurs limites. Dans un deuxième temps, nous présentons les jeux de données proposés. Enfin, nous dressons un bilan de ce travail.
— Le chapitre 6 présente notre proposition de modèle pour le résumé automatique de flux de tweets. Nous rappelons le contexte et les motivations à cette contribution et nous définissons le problème. Ensuite, nous présentons le cadre d'apprentissage, puis nous détaillons notre modèle. Enfin, nous évaluons le modèle proposé sur plusieurs jeux de données par rapport aux approches représentatives issues de l'état de l'art, et nous discutons des résultats obtenus.
— Le chapitre 7 présente notre exploration des représentations textuelles récentes pour le résumé automatique de flux de tweets. Nous rappelons le contexte et les motivations à cette contribution et nous présentons le protocole d'évaluation que nous avons mis en place. Enfin, nous présentons le cadre expérimental, et nous discutons des résultats obtenus.

Finalement, en troisième partie nous concluons ce manuscrit, en rappelant les travaux présentés et en envisageant les travaux futurs à mener.

Partie I

SYNTHÈSE DES TRAVAUX DE L'ÉTAT DE L'ART

2

S'INFORMER SUR LES MÉDIAS SOCIAUX

Sommaire

Introduction

Au début du XX^e siècle, une personne souhaitant s'informer aurait probablement demandé des informations à ses connaissances, à des experts, ou serait allée consulter des ouvrages en bibliothèque (Sanderson et Croft, 2012). Elle aurait alors été limitée par les ressources disponibles et par le temps d'accès à l'information recherchée. Aujourd'hui, des outils d'automatisation du processus de recherche existent afin de faciliter cet accès à l'information. Effectivement, depuis le milieu du XX^e siècle, avec notamment l'arrivée des premiers calculateurs automatiques, des travaux s'intéressent à créer des outils d'accès à l'information automatisés (Chiaramella et Mulhem, 2007). Ces travaux se divisent en deux grands domaines qui abordent l'accès à l'information selon deux angles différents. La Recherche d'Information (RI) cherche à retourner les documents pertinents à partir d'une requête utilisateur et le Filtrage d'Information (FI) filtre les informations utiles à un utilisateur selon son profil.

De plus en plus présents, les médias sociaux permettent à leurs utilisateurs de partager des informations. De nouveaux médias sociaux apparaissent régulièrement ces dernières années, Facebook (2004), YouTube (2005), Twitter (2006), Instagram (2010), TikTok (2016),... multipliant les informations disponibles. Les informations générées sur les médias sociaux sont abondantes, par exemple plus de 500 millions de tweets sont envoyés par jour [1]. Cette masse de données, générée par tout type d'utilisateur, à tout moment, engendre un flux de données, et apporte des problématiques supplémentaires à prendre en compte dans l'automatisation de l'accès à l'information. Ces problématiques sont notamment traitées par des approches de filtrage d'information et de résumé automatique.

Dans ce chapitre nous nous focalisons dans un premier temps, en section 1, sur les médias sociaux, nous les définissons, nous listons les principaux, et nous nous attardons sur le média social Twitter. Ensuite, nous listons les principales caractéristiques de l'accès à l'information sur les médias sociaux en section 2. Enfin, nous présentons brièvement les principaux concepts associés au filtrage d'information et mettons en avant les différences avec le résumé automatique en section 3.

1. `https://www.planetoscope.com/Internet-/1547-.html`, dernière consultation le 03/08/2022

1 Médias sociaux

720 000 heures par jour, soit 30 000 heures de vidéos sont publiées par heure sur YouTube [2], plus de 184 milliards de tweets par an, soit plus 500 millions de tweets envoyés par jour en 2020 [3] : les médias sociaux recensent plus 4,2 milliards d'utilisateurs en 2021 [4] et regorgent d'informations. Dans cette section, nous nous intéressons aux médias sociaux, nous les définissons, regardons le paysage actuel, et détaillons les caractéristiques du média social Twitter.

1.1 *Définition*

Pour définir les médias sociaux, reprenons les mots de Thorsen (2013) qui cite (Hermida, 2012) et (Heinonen, 2011). Pour Hermida (2012), les médias sociaux incluent des attributs tels que la participation, un espace ouvert, la conversation, la communauté et la connexion [...], qui sont largement en contradiction avec le modèle de communication unidirectionnel et asymétrique qui caractérisait les médias au 20e siècle :

> *Social media include attributes such as participation, openness, conversation, community and connectivity, which in Hermida's view are largely at odds with the one-way, asymmetric model of communication that characterized media in the 20th century.*

Pour Heinonen (2011), les médias sociaux sont généralement compris comme quelque chose qui permet aux gens d'être plus que de simples membres d'un public :

> *Social media is generally understood as something that enables people to be more than simply members of an audience.*

Aussi, souvent confondus, les réseaux sociaux ne représentent qu'un sous-ensemble des médias sociaux. Les réseaux sociaux sont définis par Boyd et Ellison (2007) et permettent aux utilisateurs de : (i) construire un profil, public ou visible sous conditions, (ii) assembler une liste d'autres utilisateurs du réseau avec lesquels ils partagent une connexion, et (iii) visualiser et parcourir la liste de leurs connexions et celles effec-

2. `https://blogfr.influence4you.com/les-chiffres-cles-de-la-celebre-plateforme-youtube-maj-en-2020/`, dernière consultation le 03/08/2022

3. `https://www.planetoscope.com/Internet-/1547-.html`, dernière consultation le 03/08/2022

4. `https://datareportal.com/reports/digital-2021-global-overview-report`, dernière consultation le 03/08/2022

tuées par d'autres personnes au sein du système. Ces caractéristiques permettent en effet aux utilisateurs d'être plus qu'un membre du public, mais permettre à un utilisateur d'être plus qu'un membre du public ne se limite pas à ces caractéristiques. De plus, les médias sociaux ne sont pas limités à des données textuelles mais à tout type de communication comme le texte, mais aussi l'image, l'audio, et la vidéo.

1.2 *Principaux médias sociaux*

Nous pouvons lister des exemples de médias sociaux existants et largement utilisés au moment où nous écrivons ces lignes. Il y a des réseaux sociaux, Facebook [5], Twitter [6], LinkedIn [7], ... des plateformes de diffusions de contenus vidéos et/ou photos, YouTube [8], TikTok [9], Dailymotion [10], Vimeo [11], Twitch [12], Instagram [13], Pinterest [14], Flickr [15], ... de contenu audio, Deezer [16], SoundCloud [17], Spotify [18], ... des bases de données participatives, Wikipedia [19], IMDb [20], ou encore des forums, Reddit [21], Quora [22], Stack Overflow [23], ...

Une représentation de ces différents médias sociaux est le *conversation prism* [24], lancé en 2008 par Brian Solis et Jesse Thomas. La 5e version du prisme est reportée en figure 2.1. Le prisme catégorise les médias sociaux par leurs caractéristiques intrinsèques (catégories à l'extérieur du cercle, comme réseaux sociaux, blog/microblog, vidéos, wiki ...) et classe également les médias sociaux depuis le prisme de l'utilisateur, ce qu'il peut y faire et y apporter, l'impact d'un investissement dans le média, et la manière dont il s'y investit (catégories selon les 3 anneaux au milieu du cercle).

5. `https://fr-fr.facebook.com/`, dernière consultation le 03/08/2022
6. `https://twitter.com/`, dernière consultation le 03/08/2022
7. `https://fr.linkedin.com/`, dernière consultation le 03/08/2022
8. `https://www.youtube.com/`, dernière consultation le 03/08/2022
9. `https://www.tiktok.com/fr/`, dernière consultation le 03/08/2022
10. `https://www.dailymotion.com/fr`, dernière consultation le 03/08/2022
11. `https://vimeo.com/fr`, dernière consultation le 03/08/2022
12. `https://www.twitch.tv/`, dernière consultation le 03/08/2022
13. `https://www.instagram.com/`, dernière consultation le 03/08/2022
14. `https://www.pinterest.fr/`, dernière consultation le 03/08/2022
15. `https://www.flickr.com/`, dernière consultation le 03/08/2022
16. `https://www.deezer.com/fr/`, dernière consultation le 03/08/2022
17. `https://soundcloud.com/`, dernière consultation le 03/08/2022
18. `https://www.spotify.com/fr/`, dernière consultation le 03/08/2022
19. `https://fr.wikipedia.org/`, dernière consultation le 03/08/2022
20. `https://www.imdb.com/`, dernière consultation le 03/08/2022
21. `https://www.reddit.com/`, dernière consultation le 03/08/2022
22. `https://fr.quora.com/`, dernière consultation le 03/08/2022
23. `https://stackoverflow.com/`, dernière consultation le 03/08/2022
24. `https://conversationprism.com`, dernière consultation le 03/08/2022

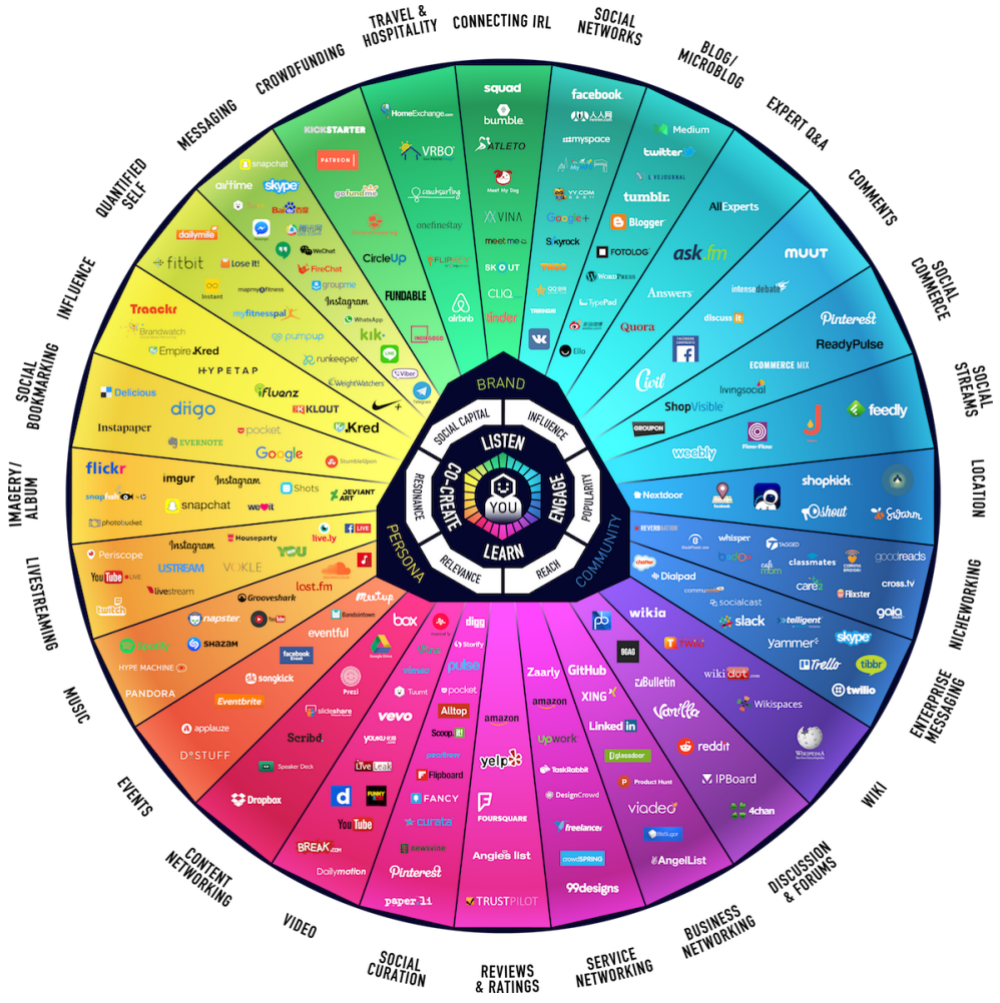


FIGURE 2.1 – The conversation prism - Classification des principaux médias sociaux.

1.3 *Twitter*

Twitter a été fondé en mars 2006 par Jack Dorsey, Evan Williams, Biz Stone et Noah Glass. Twitter compte aujourd'hui près de 350 millions de comptes actifs [25] à travers le globe. On y retrouve les comptes de personnalités politiques, de sportifs, d'artistes, de marques, de chaînes d'information, et même celui de Twitter. En figure 2.2 est reporté le classement des 20 comptes les plus suivis sur Twitter au 1er Janvier 2022 (les cas particuliers des comptes de Donald Trump et Ariana Grande ne sont plus sur Twitter mais reportés dans la liste, le premier supprimé par Twitter, le deuxième supprimé par la détentrice du compte). Si le média est largement

25. `https://datareportal.com/reports/digital-2021-global-overview-report`, dernière consultation le 03/08/2022

utilisé par le grand public, il est également largement utilisé par les chercheurs en informatique ou en sciences sociales notamment, tant pour ses caractéristiques que pour les données qu'il contient.

Rank	Change	Account name	Owner	Followers[1] (millions)	Occupation	Country	Brand
1	–	@BarackObama	Barack Obama	130.8	44th President of the United States	United States	—
2	–	@justinbieber	Justin Bieber	114.2	Musician	Canada	—
3	–	@katyperry	Katy Perry	108.8	Musician	United States	—
4	–	@rihanna	Rihanna	104.7	Musician	Barbados	—
5	–	@cristiano	Cristiano Ronaldo	97.6	Footballer	Portugal	—
6	–	@taylorswift13	Taylor Swift	90.1	Musician	United States	—
—	–	@realDonaldTrump[2]	Donald Trump	88.8	45th President of the United States	United States	—
—	–	@ArianaGrande[3]	Ariana Grande	85.3	Musician and actress	United States	—
7	–	@ladygaga	Lady Gaga	84.2	Musician and actress	United States	—
8	–	@TheEllenShow	Ellen DeGeneres	77.6	Comedian and television host	United States	—
9	▲	@narendramodi	Narendra Modi	76.0	Prime Minister of India	India	—
10	▲	@elonmusk	Elon Musk	74.4	Entrepreneur and business magnate	United States Canada South Africa	—
11	▼	@YouTube	YouTube	74.3	Online video platform	United States	✓
12	–	@KimKardashian	Kim Kardashian	71.2	Television personality	United States	—
13	–	@selenagomez	Selena Gomez	65.5	Musician and actress	United States	—
14	–	@jtimberlake	Justin Timberlake	63.2	Musician and actor	United States	—
15	–	@cnnbrk	CNN Breaking News	62.0	News channel	United States	✓
16	–	@Twitter[a]	Twitter	60.8	Social media platform	United States	✓
17	–	@BillGates	Bill Gates	57.1	Businessman and philanthropist	United States	—
18	–	@cnn	CNN	56.2	News channel	United States	✓
19	▲	@neymarjr	Neymar	56.0	Footballer	Brazil	—
20	▼	@britneyspears	Britney Spears	55.8	Musician	United States	—

FIGURE 2.2 – Utilisateurs de Twitter les plus suivis au 1er Janvier 2022.

1.3.1 *Caractéristiques*

Twitter est un média social de *microblogging*, c'est-à-dire de partage de courtes publications, appelées *tweets* (gazouillis). Ces publications étaient limitées à 140 caractères à la création de Twitter puis étendues à 280 caractères à partir de novembre 2017. Un partage de liens URL dans un tweet est compté à 23 caractères, un maximum de 4 photos peut être publié par tweet, et la taille maximale d'une vidéo est de 2 minutes et 20 secondes [26]. Chaque utilisateur de Twitter doit choisir un *nom d'utilisateur*, modifiable mais unique sur la plateforme, et un *nom d'affichage*, également modifiable, qui peut différer du nom d'utilisateur, pas forcément unique sur la plateforme. Les utilisateurs peuvent se référencer entre eux dans leurs tweets à l'aide du nom d'utilisateur précédé d'un @, on parle alors de *mention*. Pour réagir à un sujet spécifique, l'utilisateur peut se servir des *hashtags*, avec un # précédant

26. `https://help.twitter.com/en/using-twitter#adding-content-to-your-tweet`, dernière consultation le 03/08/2022

chaque mot-clé. Il est ainsi possible de consulter tous les tweets publics contenant un hashtag particulier.

Chaque utilisateur peut suivre un autre utilisateur, de manière unilatérale. Contrairement à d'autres médias sociaux comme Facebook par exemple, où deux utilisateurs A et B sont amis si et seulement si A est ami avec B et B est ami avec A, un utilisateur A de Twitter peut suivre un autre utilisateur B qui ne le suit pas. Si un utilisateur A suit un utilisateur B, on dit alors que A est un *abonné* ou *follower* de B et que B est un *ami* ou *friend* de A.

Un utilisateur peut rejoindre des *listes* afin d'organiser les tweets qu'il voit. Il peut créer ses propres listes et rejoindre des listes créées par d'autres utilisateurs.

Chaque tweet possède un *identifiant* (*id*) unique et est *horodaté* à sa création (*timestamp*). Pour chaque tweet, un utilisateur peut *aimer* (*like*), *répondre/commenter* (*reply*), ou *retweeter*, c'est-à-dire partager le tweet d'origine à ses abonnés. En figure 2.3 est présenté un profil d'un utilisateur de Twitter, et en figure 2.4 un exemple de tweet par l'utilisateur @UT3PaulSabatier retweeté par l'utilisateur @IRIToulouse.

FIGURE 2.3 – Exemple d'un profil d'utilisateur de Twitter.

1.3.2 *Récupération de données*

L'attractivité de Twitter dans le monde de la Recherche vient notamment de l'accessibilité aux données. Twitter met à disposition des développeurs une API permettant l'accès aux données publiques (sous certaines conditions [27]). Cette API permet entre autres de récupérer des tweets à partir d'un hashtag particulier ou encore de récupérer des tweets à partir de leur identifiant. Dans les travaux de ce

27. `https://developer.twitter.com/en/docs/twitter-api/getting-started/about-twitter-api`, dernière consultation le 03/08/2022

FIGURE 2.4 – Exemples de tweet et retweet.

manuscrit, nous utilisons la librairie Python *twarc*[28] qui facilite l'utilisation de l'API Twitter.

2 S'informer sur les médias sociaux

Les médias sociaux font aujourd'hui partie des médias d'information. Cependant, la quantité de données présente complique l'accès à l'information et s'informer efficacement requiert beaucoup de temps. Effectivement, d'après le Digital Report 2021 [29],

28. `https://github.com/DocNow/twarc`, dernière consultation le 03/08/2022
29. `https://datareportal.com/reports/digital-2021-global-overview-report`, dernière consultation le 03/08/2022

la principale motivation d'utilisation des médias sociaux est de rester à jour des nouvelles et des évènements courants, et l'utilisation moyenne de ces médias sociaux est de 2 heures et 25 minutes par jour. De plus, en 2021, une étude conduite auprès de plus de 2 400 journalistes a montré que Twitter est la source d'information principale pour 16 % d'entre eux [30]. Une synthèse de cette masse d'information apparaît intéressante pour aider l'utilisateur qui souhaite s'informer sur les médias sociaux.

De par leur nature, les données issues des médias sociaux, générées par les utilisateurs, présentent des caractéristiques motivantes pour s'informer, limitées néanmoins par des caractéristiques contraignantes. En premier lieu, la *couverture* de l'information est une caractéristique motivante à l'utilisation des médias sociaux pour s'informer. Les travaux de Zhao *et al.* (2011) montrent que les données issues des médias sociaux couvrent un éventail de sujets similaire aux médias traditionnels, mais l'intérêt est porté sur des sujets différents, les sujets des médias sociaux étant davantage axés sur la vie personnelle et la pop culture. Il est à noter que ces résultats datent de 2011 et les médias sociaux n'ont cessé de se développer depuis, leur couverture étant certainement encore plus importante aujourd'hui. Ensuite, la *fraîcheur* de l'information est un point positif en vue de s'informer sur les médias sociaux. Les informations peuvent être disponibles plus rapidement sur les médias sociaux que sur les médias traditionnels, surtout lors d'évènements imprévus. Notamment, Valenzuela *et al.* (2017) montrent que Twitter influence davantage les informations abordées à la télévision que l'inverse, et qu'environ 40 % des nouvelles rapportées par les journalistes (sur Twitter ou à la télévision) proviennent d'informations fournies par le public.

En revanche, sur les médias sociaux, le *volume*, la *redondance*, et la *qualité* de l'information apparaissent comme des contraintes vis-à-vis de l'objectif de s'informer. Dans d'autres contextes, la contrainte du volume de données est qu'il est trop faible, engendrant des carences dans la couverture de l'information. Ici, c'est plutôt l'inverse, le volume est tel que l'utilisateur se retrouve submergé sous cette masse de données. La *redondance* est très présente dans les médias sociaux. En effet, les utilisateurs ne se concertent pas entre eux pour relayer des informations distinctes. Par exemple, sur Twitter, Zanzotto *et al.* (2011) ont trouvé environ 30 % de redondance dans l'extrait considéré, sans compter les retweets. Une paire de tweets est jugée redondante si les deux tweets partagent la même information ou si l'un des deux tweets contient l'information de l'autre. Les médias traditionnels sont rédigés par des professionnels, mais sur les médias sociaux l'information est partagée par tout un chacun, la *qualité* de l'information s'en trouve également impactée. La qualité du contenu, comme la qualité textuelle, n'est pas vérifiée, tant sur la forme (comme l'aspect syntaxique) que sur le fond. Il y a des spams, des contenus non informatifs,

30. `https://muckrack.com/blog/2021/03/15/state-of-journalism-2021`, dernière consultation le 03/08/2022

ainsi que de fausses informations. Vosoughi *et al.* (2018) ont étudié la propagation des fausses informations sur Twitter. Elles se répandent plus profondément, plus rapidement, plus largement que les vraies informations, dans toutes les catégories d'informations étudiées, et les effets sont notamment plus prononcés pour les nouvelles politiques, sur le terrorisme, les catastrophes naturelles, la science, les légendes urbaines et les informations sur la finance.

3 S'informer - filtrage d'information

La Recherche d'Information (RI) est définie dans (Manning *et al.*, 2008) comme l'action de trouver le matériel nécessaire à la satisfaction d'un besoin d'information au sein de grandes collections :

> *Information retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers).*

Les systèmes de Recherche d'Information sont eux définis dans (Belkin et Croft, 1992) comme des guides pour l'utilisateur dans son besoin d'information. Ils ont pour objectif de récupérer l'information qui aide l'utilisateur à partir d'une base de connaissances :

> *Usually, an IR system is considered to have the function of "leading the user to those documents that will best enable him/her to satisfy his/her need for information". Somewhat more generally, "the goal of an information [retrieval] system is for the user to obtain information from the knowledge resource which helps her/him in problem management".*

Dans un cadre de Recherche d'Information traditionnelle, appelée recherche adhoc, l'objectif d'un système est de retourner une liste ordonnée de documents pertinents à une requête utilisateur. Nous nous focalisons ici sur le Filtrage d'Information (FI), vu par Belkin et Croft (1992) comme un type particulier de RI. Pour Hanani *et al.* (2001), le FI hérite des problèmes de RI mais diffère selon plusieurs aspects. Pour nos problématiques, le FI peut s'avérer intéressant pour sa capacité à gérer des flux d'information. Dans cette section, nous explorons les enjeux et intérêts, ainsi que le processus du filtrage d'information.

3.1 *Enjeux et intérêts*

La RI ordinaire se focalise sur la récupération de documents à partir d'une requête exprimée par un utilisateur. Ensuite, l'utilisateur navigue au sein des documents retournés par ordre de pertinence afin de répondre à son besoin d'information. Cependant, d'autres situations requièrent un besoin d'information utilisateur. Notamment lorsque la collection de documents est un flux de données, l'utilisateur peut vouloir s'informer régulièrement sur les informations apportées par le flux. Dans ce cas, l'objectif est de filtrer l'information pour éviter à l'utilisateur de consulter l'entièreté du flux, voire même de ne retourner que l'information qui lui sera utile. Plus précisément, selon Hanani *et al.* (2001), le Filtrage d'Information (FI) :

— est applicable sur des données non structurées et semi-structurées, comme des emails, des documents,
— s'occupe de grandes quantités de données,
— est utilisé principalement pour des données textuelles,
— est basé sur des profils utilisateurs,
— a pour objectif de supprimer les données non pertinentes de flux de données.

Le FI diffère de la RI par 5 aspects d'après Hanani *et al.* (2001), reportés dans le tableau 2.1.

TABLEAU 2.1 – Différences entre la Recherche d'Information (RI) et le Filtrage d'Information (FI) selon Hanani *et al.* (2001).

Paramètre	RI	FI
Fréquence d'utilisation	utilisation ad-hoc, ponctuelle	utilisation répétitive, à long terme
Représentation du besoin d'information	requêtes	profils
Objectif	sélection de données pertinentes pour la requête	filtrage des données non pertinentes ou collecte d'éléments de données
Base de données	relativement statique	très largement dynamique
Type d'utilisateurs	inconnu du système	le système garde les profils utilisateurs
Périmètre du système	concerne uniquement la pertinence des données	concerne aussi les problèmes sociaux comme la vie privée

3.2 *Le processus*

Belkin et Croft (1992) illustrent le processus de FI (voir figure 2.5). Le processus sépare les documents (le flux de données) d'un côté et les besoins de l'utilisateur de l'autre. Pour une même collection de documents, et un même besoin utilisateur, les modèles automatiques proposés diffèrent par rapport à la représentation des documents et des besoins et/ou de la méthode de comparaison ou de filtrage. Il convient

de noter que la représentation des textes peut comporter des pré-traitements particuliers. On retrouve parmi les pré-traitements les plus connus l'élimination des mots vides, la lemmatisation, ou la conversion du texte en minuscule par exemple. Les textes sont pré-traités afin de mieux représenter le texte pour la tâche souhaitée (Uysal et Günal, 2014).

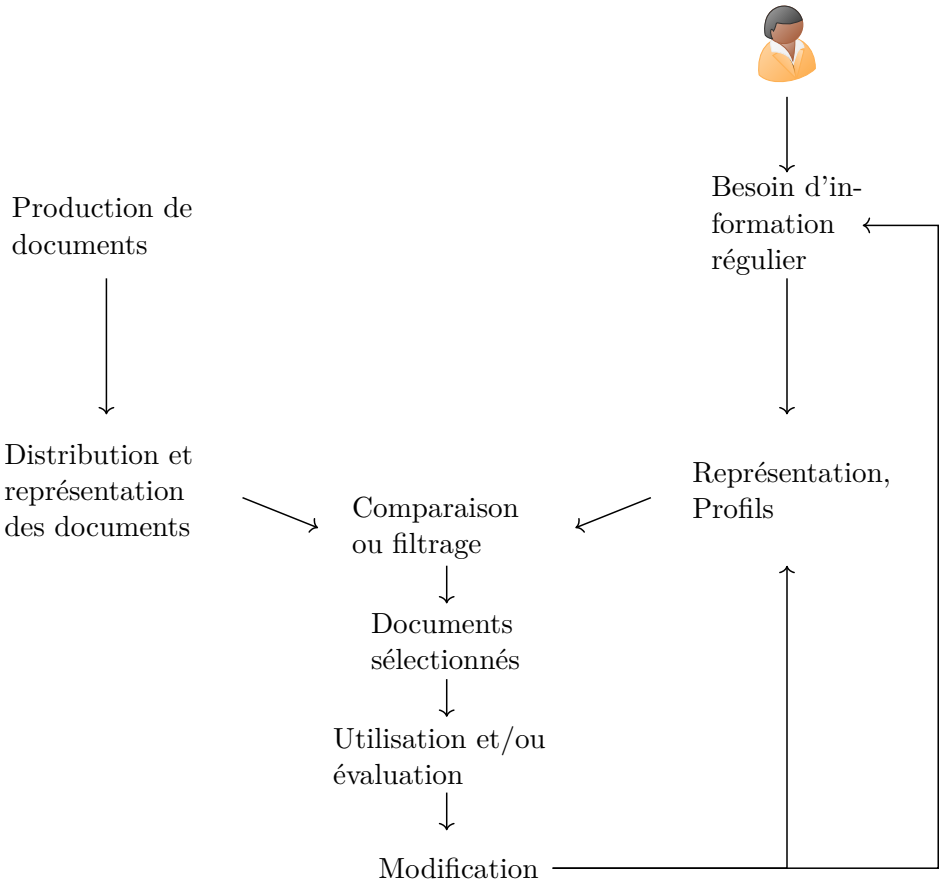


FIGURE 2.5 – Illustration du processus de Filtrage d'Information.

3.3 *Filtrage vs résumé*

Dans cette thèse nous nous focalisons sur la tâche de résumé automatique pour s'informer sur les médias sociaux. Nous avons présenté le filtrage d'information, une tâche qui paraît intéressante également pour s'informer sur les médias sociaux puisqu'elle permet de supprimer les informations non pertinentes d'un flux de données. Parfois confondues, les tâches de résumé et de filtrage d'information sont pourtant différentes :

— La tâche de résumé n'a pas forcément besoin d'un profil ou d'une requête utilisateur,
— le but de la tâche de résumé n'est pas de supprimer les informations non pertinentes mais de retourner les plus pertinentes,
— la tâche de résumé a ses propres propriétés, notamment pour définir ce qu'est un bon résumé.

Aussi, nous ne détaillons pas dans cette section la tâche de résumé automatique, nous nous étendons sur le sujet dans le chapitre 3.

4 Conclusion

Dans ce chapitre, nous avons exploré les médias sociaux, nous les avons définis, recensés les principaux actuellement, puis examiné plus particulièrement le média social Twitter. Enfin, motivés par la difficulté de s'informer sur les médias sociaux, nous avons parcouru les différentes caractéristiques des médias sociaux qui impactent l'accès à l'information. Enfin, nous avons brièvement introduit le filtrage d'information, les enjeux et intérêts, le processus, ainsi que les différences avec le résumé automatique.

Dans le chapitre suivant, et plus globalement dans cette thèse, nous nous focalisons sur le résumé textuel automatique. Nous y dressons un état de l'art, nous apportons les définitions, les méthodes et processus de résumé et d'évaluation des résumés.

3

RÉSUMÉ AUTOMATIQUE D'INFORMATION

Sommaire

Introduction

Dans le but de s'informer à partir de grandes quantités de données, une tâche largement étudiée est le résumé automatique. Le domaine est étudié depuis la fin des années 1950. Les travaux de Hans Peter Luhn (Luhn, 1958), sont reconnus comme étant les premiers publiés en résumé automatique. À l'époque, Luhn conduit ces travaux pour la littérature technique. Il remarque que les résumés (abstract), qui permettent au lecteur un gain de temps et d'efforts, nécessitent du temps, un effort intellectuel considérable pour les élaborer, qui pourrait être utilisé à d'autres fins par la main d'œuvre qualifiée. De plus, il remarque, déjà, que la quantité de données disponibles augmente de plus en plus, et de plus en plus rapidement. Enfin, il pointe une dernière motivation à l'élaboration de résumés de manière automatique, celle de la cohérence et de l'objectivité dans les résumés écrits manuellement, largement biaisés par les antécédents, l'attitude, et la disposition de la personne qui rédige.

Depuis ces travaux, des progrès remarquables ont été faits dans le domaine du résumé automatique. Aujourd'hui il est même possible de trouver des outils de résumé automatique en ligne, comme Resoomer [1], SMMRY [2], Summarizing-tool [3], ou encore SummarizeBot [4] pour Slack [5] et Messenger [6].

Dans ce chapitre, nous reprenons la définition du résumé automatique. Ensuite, nous recensons les différents types de résumés. De plus, nous explorons les méthodes de résumé automatique existantes, puis nous présentons les méthodes d'évaluation de résumés, avant de conclure. Il est à noter que nous nous focalisons ici, et dans l'ensemble du manuscrit, sur le résumé automatique de texte, mais il existe également des travaux en image (Celis et Keswani, 2020), audio (Li *et al.*, 2021), et vidéo (Apostolidis *et al.*, 2021), et même des travaux regroupant plusieurs de ces modalités (Narasimhan *et al.*, 2021).

1 Définition

Pour Maybury (1995), un résumé efficace distille les informations les plus importantes à partir d'une ou plusieurs sources pour produire un version abrégée de l'information originale, pour des utilisateurs et des tâches précis :

1. `https://resoomer.com/`, dernière consultation le 04/08/2022
2. `https://smmry.com/`, dernière consultation le 04/08/2022
3. `https://summarizing-tool.com/fr/`, dernière consultation le 04/08/2022
4. `https://www.summarizebot.com/`, dernière consultation le 04/08/2022
5. `https://slack.com/intl/fr-fr/`, dernière consultation le 04/08/2022
6. `https://www.messenger.com/`, dernière consultation le 04/08/2022

> *An effective summary distills the most important information from a source (or sources) to produce an abridged version of the original information for a particular user(s) and task(s).*

Dans (Radev *et al.*, 2002), un résumé peut être défini comme un texte produit à partir d'un ou plusieurs textes, qui transmet l'information importante des textes originaux, et qui est généralement bien moins long que la moitié de la taille des textes originaux :

> *A summary can be loosely defined as a text that is produced from one or more texts, that conveys important information in the original text(s), and that is no longer than half of the original text(s) and usually significantly less than that. Text here is used rather loosely and can refer to speech, multimedia documents, hypertext, etc.*

Finalement, une formulation concise de l'objectif du résumé automatique est donnée dans (Liu et Lapata, 2019). Le but (du résumé automatique) est de condenser un document dans une version plus courte tout en préservant l'essentiel de son sens :

> *The aim is to condense a document into a shorter version while preserving most of its meaning.*

2 Classification des approches de résumé automatique

Nous venons de le définir dans la section précédente, le résumé automatique a un objectif clair. Cependant, selon les caractéristiques des données et des problématiques attaquées, les approches de résumé automatique sont classées différemment. Dans cette section, nous décrivons les différentiations abstractif versus extractif et mono-document versus multi-document, et nous présentons également quelques autres classifications reconnues.

2.1 *Extractif - abstractif*

Une première classification des différentes approches distingue les modèles de résumés *extractifs* et *abstractifs*. Les approches de résumé extractives, comme (Fabbri *et al.*, 2019; Saini *et al.*, 2019; Zhong *et al.*, 2020), retournent des extraits du texte en entrée, tandis que les approches de résumés abstractives, comme (Qi *et al.*, 2020;

Zhang *et al.*, 2020a; Dou *et al.*, 2021), formulent de nouvelles phrases. Il peut être intéressant de noter que résumé abstractif ne signifie pas interdiction d'utiliser des phrases du texte à résumer, mais la philosophie est plutôt de reformuler les documents originaux s'il y en a l'intérêt. En théorie, les résumés abstractifs sont plus lisibles, mais en pratique il est très compliqué d'obtenir des résumés abstractifs de haute qualité (El-Kassas *et al.*, 2021). Il est à noter que depuis peu, des approches combinent à la fois extraction et abstraction, comme (Liu et Lapata, 2019; Rudra *et al.*, 2019; Song *et al.*, 2020), générant dans un premier temps des résumés extractifs puis les reformulant avec de nouvelles phrases. Ces approches sont appelées *hybrides*.

Pour le document suivant :

> *Le chat marche dans la cuisine. Il est tout noir, avec une petite tâche blanche sur le poitrail. Soudain, il voit la souris. Il entame sa chasse tout doucement, il ne veut pas faire de bruit, il ne veut pas signaler sa présence. Après analyse de la situation, il avance doucement vers sa cible, et, d'un coup, bondit sur elle. La souris esquive le chat de justesse et part se cacher. Le chat la suit, puis recommence une, deux, trois fois de l'attraper. La chasse se poursuit jusqu'au jardin. Arrivée au jardin, le chat perd la trace de la souris. Finalement, la souris s'échappe.*

Un exemple de résumé extractif pourrait être :

> *Le chat marche dans la cuisine. Soudain, il voit la souris. Il entame sa chasse. Le chat la suit, puis recommence une, deux, trois fois de l'attraper. Finalement, la souris s'échappe.*

Un exemple de résumé abstractif pourrait être :

> *Le chat voit la souris. Après de nombreuses tentatives pour l'attraper, le chat n'y arrive pas et la souris s'échappe.*

2.2 *Mono-document - multi-document*

Une deuxième classification des différentes approches distingue les modèles de résumés *mono-document* et *multi-document*. Pour du mono-document, comme (Liu et Lapata, 2019; Qi *et al.*, 2020; Dou *et al.*, 2021), il n'y a qu'un texte à résumer, tandis que pour du multi-document il y a plusieurs documents à résumer.

Goldstein *et al.* (2000) listent 4 différences significatives entre le résumé mono-document et multi-document :

— La redondance d'information est plus fortement présente pour le résumé multi-document,
— le résumé multi-document utilise un groupe d'articles qui peuvent alors contenir une notion de temporalité,
— la compression de l'information est plus élevée pour le résumé multi-document, c'est-à-dire que le ratio de compression est plus faible pour le résumé multi-document
— le problème de coréférence (désignation d'une même entité par un mot différent) est bien plus présent pour le résumé multi-document.

Aussi, appuyés par les travaux de (Goldstein *et al.*, 2000; Huang *et al.*, 2010), les besoins d'un résumé multi-document « idéal » sont :

— la couverture : les principales informations des documents de la collection sont présentes dans le résumé ;
— la non redondance : il n'y a pas de redondance d'information dans le résumé ;
— la cohésion : cela correspond à l'habileté à combiner les différents passages de texte de manière utile pour le lecteur. Les auteurs de (Goldstein *et al.*, 2000) mentionnent quelques directions pour assurer la cohésion, comme l'ordonnancement par document, la cohésion par sujet et l'ordonnancement temporel ;
— la cohérence : le résumé doit être lisible et pertinent pour le lecteur ;
— la contextualisation : le résumé doit inclure suffisamment de contexte pour être compris par un être humain ;
— la concordance : l'information venant de différentes sources, des contradictions peuvent apparaître (par exemple milliards au lieu de millions,...).

2.3 *Autres classifications possibles*

Dans la littérature, d'autres classifications existent pour les résumés, comme par exemple :

— Les résumés *indicatifs* et les résumés *informatifs*, cités dans (El-Kassas *et al.*, 2021). Les résumés indicatifs présentent le document, de quoi il parle (ex : Dans ce document qui parle de la seconde guerre mondiale...) (Yeh *et al.*, 2005). Les résumés informatifs présentent une information sous forme élaborée (ex : Durant la seconde guerre mondiale...) (Ganesan *et al.*, 2010).
— Les résumés *monolingues*, les résumés *multilingues* et les résumés *inter-linguistique*, également cités dans (El-Kassas *et al.*, 2021). Les résumés monolingues utilisent des documents d'une langue et résument dans la même langue

(Genest et Lapalme, 2011). Les résumés multilingues utilisent des documents de plusieurs langues et résument dans ces langues d'entrée (Gupta, 2013). Les résumés inter-linguistique résument dans des langues différentes de celles des documents d'entrée (Yao *et al.*, 2015).

Nous ajoutons une classification supplémentaire qui distingue les résumés à *longueur définie* (constante ou maximale) et les résumés à *longueur variable*. Par exemple, les approches (Fabbri *et al.*, 2019; Rudra *et al.*, 2019; Saini *et al.*, 2019) génèrent des résumés de taille fixe, tandis que les résumés générés par les approches (Liu et Lapata, 2019; Song *et al.*, 2020; Dou *et al.*, 2021) sont de tailles variables, définies à la volée par l'approche.

3 Méthodes de génération de résumé automatique

Dans cette section, nous détaillons les approches de l'état de l'art du résumé automatique. Dans ce manuscrit, et dans nos travaux, nous nous focalisons sur la génération de résumés extractifs. Nous détaillons ainsi dans un premier temps les approches de résumé extractives, puis nous survolons les méthodes de résumé abstractives.

3.1 *Méthodes extractives*

Les approches extractives composent des résumés à partir d'extraits du texte en entrée. Ces textes sont dans un premier temps pré-traités puis représentés selon l'approche utilisée. À partir de ces représentations, un score est attribué aux différentes phrases du texte. Les phrases sont ensuite extraites du texte en fonction de leur score, puis post-traitées, et finalement assemblées pour obtenir le résumé. Nous présentons l'illustration de l'architecture globale d'une méthode de résumé extractive en figure 3.1.

Avant de présenter les différentes approches, nous présentons la MMR (*Maximal Marginal Relevance*) (Carbonell et Goldstein, 1998), commune à beaucoup d'approches, indépendamment de la classification utilisée dans les paragraphes suivants. La MMR est une méthode de classement. Elle est souvent utilisée dans les approches de résumés extractives, pour construire les résumés à partir des scores des différentes phrases. La MMR retourne dans un premier temps les items les plus pertinents, puis dans un second temps privilégie les items les plus dissimilaires. Par exemple, pour un résumé, elle retourne d'abord la phrase la plus pertinente puis cherche la phrase

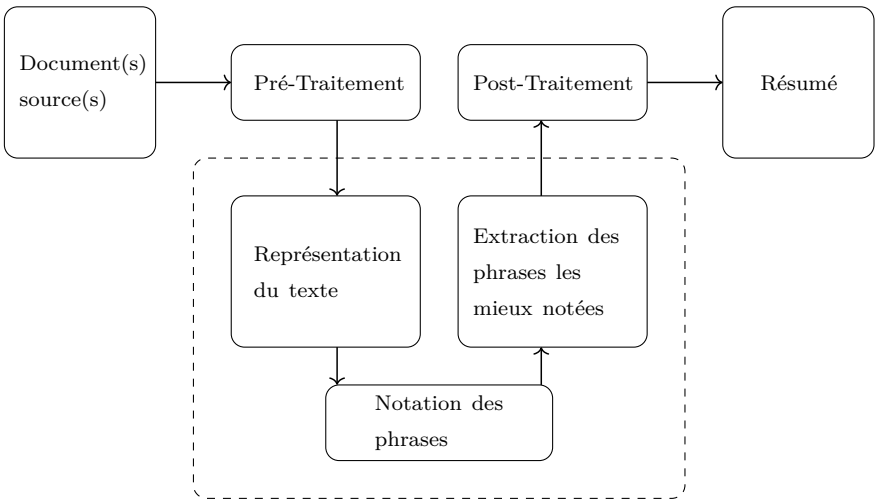


FIGURE 3.1 – Illustration du processus de création de résumés extractifs, repris de (El-Kassas *et al.*, 2021)

suivante la plus pertinente et la plus dissimilaire à celle déjà récupérée, et ainsi de suite jusqu'à la taille souhaitée du résumé.

Dans les paragraphes suivants, nous présentons les méthodes de résumé extractives inspirés par les classifications que nous pouvons retrouver dans (Gambhir et Gupta, 2017; El-Kassas *et al.*, 2021).

3.1.1 *Méthodes basées sur des caractéristiques statistiques*

Les approches basées sur des caractéristiques statistiques, aussi appelées approches statistiques dans (Gambhir et Gupta, 2017; El-Kassas *et al.*, 2021), sont basées sur les caractéristiques extraites à partir de statistiques descriptives des documents. Ces caractéristiques sont indépendantes de la langue et peuvent être utilisées quelle que soit la langue des documents (Gambhir et Gupta, 2017). L'idée de ces méthodes et de découvrir des motifs dans la manière d'écrire ou construire un document. Ko et Seo (2008) utilisent les caractéristiques suivantes : (i) le nombre de termes communs entre une phrase et le titre du document, (ii) la position de la phrase dans le document, les premières phrases étant mieux notées, (iii) la similarité d'une phrase par rapport aux autres phrases du document, (iv) la somme de l'importance des termes d'une phrase calculée par Tf-Idf du document, (v) et le nombre de termes communs entre une phrase et les termes les plus fréquents du documents. Les scores évaluées à partir de ces caractéristiques sont ensuite sommés pour calculer le score d'une phrase. Fattah et Ren (2009) s'appuient également sur les caractéristiques suivantes : (i) le chevauchement de mots entre une phrase et le titre du document, (ii) la position de la phrase dans le document, (iii) le chevauchement de mots entre une phrase et les

autres phrases du document, et utilise d'autres caractéristiques : (iv) l'apparition des mots-clés d'une phrase dans de précédents résumés, (v) la non apparition des mots-clés d'une phrase dans de précédents résumés, (vi) le nombre de noms propres d'une phrase, (vii) le nombre de données numériques d'une phrase, (viii) la taille moyenne des phrases du document, (ix) le nombre de phrases supérieures à un seuil de similarité d'une phrase, (x) et la somme des similarités des phrases supérieures à un seuil de similarité. Ils utilisent ensuite ces caractéristiques pour l'apprentissage de différents modèles.

Ferreira *et al.* (2014) présentent un système de résumé multi-document basé sur des caractéristiques statistiques mais également du traitement linguistique. Les caractéristiques statistiques utilisées sont : (i) la fréquence des termes, (ii) le score Tf-Idf des termes, (iii) la similarité entre les paires de phrases. Les caractéristiques linguistiques sont utilisées avec l'utilisation de relations de synonymie, d'hyponymie, d'hyperonymie, de coréférence, et de relations du discours. Basées sur ces caractéristiques, des groupes (*clusters*) sont construits. Les phrases les plus représentatives de chaque groupe sont retournées à l'utilisateur en tant que résumé. Nous détaillons justement les approches basées sur la création de groupes dans la section suivante.

3.1.2 *Méthodes basées sujets/groupes*

L'idée des méthodes basées sujets ou groupes est de construire des ensembles, de regrouper les phrases similaires. Ces groupes peuvent être créés à partir de sujets définis *a priori* ou identifiés *a posteriori*. Les résumés sont ensuite construits à partir des phrases les plus représentatives des différents groupes. Nous donnons un aperçu des approches basées sur les groupes en figure 3.2.

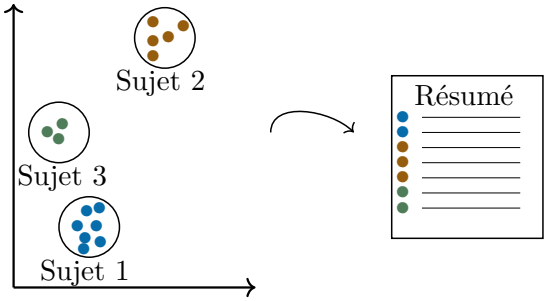


FIGURE 3.2 – Aperçu du résumé par construction de groupes.

Radev *et al.* (2004) introduisent MEAD, une méthode de résumé de documents basée sur les *centroids*. Les *centroids* sont composés de termes. Les termes dont le score calculé à partir du Tf et de l'Idf est supérieur à un certain seuil forment le *centroid* d'un document. Ensuite, pour chaque phrase du document, un score est calculé à partir des termes appartenant au *centroid* : (i) le score du terme à partir

du Tf et de l'Idf, (ii) la position de la phrase dans un document, plus la phrase est proche du début du document, plus le score est élevé, (iii) la similarité d'une phrase à la première phrase du document dans lequel elle apparaît.

Harabagiu et Lacatusu (2005) proposent de diviser les groupes en thèmes. Ainsi, les résumés sont créés à partir des différents thèmes et des phrases qui les composent. Les thèmes sont construits à partir des structures prédicats-arguments des phrases des documents. Les phrases contentant au moins un prédicat commun avec un argument commun sont regroupées selon le même thème. Parmi l'ensemble des thèmes construits, le choix des thèmes sélectionnés pour le résumé est finalement basé sur des caractéristiques telles que le nombre de phrases dans le thème, ou la position de la phrase dans chaque document où le thème est reconnu.

3.1.3 *Méthodes orientées graphes*

Dans un graphe, les éléments du texte (mots ou phrases) sont représentés par des nœuds et les arêtes relient les éléments de texte connexes (sémantiquement liés) entre eux (Gambhir et Gupta, 2017). Par exemple, un nœud représente un mot et deux nœuds sont liés s'ils se succèdent dans une phrase des documents d'entrée. Nous illustrons la représentation sous forme de graphe en figure 3.3.

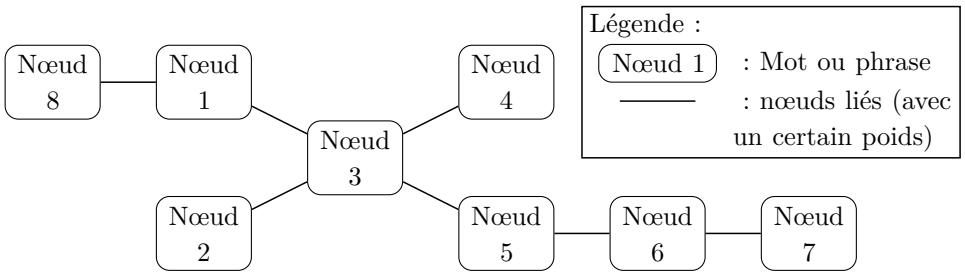


FIGURE 3.3 – Représentation du texte sous forme de graphe.

LexRank (Erkan et Radev, 2004) est basé sur PageRank (Page *et al.*, 1999) dans un cadre de résumé automatique. Un graphe est créé à partir des similarités Cosinus entre les phrases d'un document, selon une représentation Tf-Idf. Les nœuds du graphe sont les phrases, les arêtes sont les similarités entre les deux phrases liées. Contrairement à PageRank, le graphe de LexRank est un graphe non orienté, la similarité Cosinus étant symétrique. TextRank (Mihalcea et Tarau, 2004), similaire à Lexrank, est également basé sur PageRank. Contrairement à LexRank, TextRank calcule les similarités entre phrases par leur nombre de termes communs.

Baralis *et al.* (2013) exploitent les corrélations entre des ensembles de termes. Les nœuds du graphe construit sont des termes ou des ensemble de termes, les arêtes sont les corrélations entre les nœuds. Les ensembles de termes sont créés selon différentes

règles d'associations entre les termes. Le choix des phrases à sélectionner pour le résumé est également basé sur l'algorithme PageRank.

3.1.4 *Discours*

Certaines méthodes de résumé automatique utilisent des règles de théorie du discours. Daumé et Marcu (2002) utilisent la théorie du discours de la structure rhétorique (RST, Rhetorical Structure Theory) (Mann et Thompson, 1988). Cette théorie décrit les relations entre les différentes parties d'un texte. Elle définit notamment la relation noyau-satellite (*nucleus-satellite*) dans laquelle le satellite détaille ce qui est postulé par le noyau. Ainsi, ils procèdent comme suit : (i) analyse du discours d'un document selon la théorie RST, (ii) analyse syntaxique des phrases du document, (iii) suppression des éléments du discours et syntaxiques non essentiels pour la génération du résumé. Contrairement à Daumé et Marcu (2002) qui utilisent la théorie du discours de la structure rhétorique, Clarke et Lapata (2010) utilisent la représentation du discours par chaînes lexicales (*lexical chain*) (Morris et Hirst, 1991). Les chaînes lexicales sont construites selon 5 types de relations à partir d'un thésaurus : (i) deux termes ont une catégorie en commun, (ii) un terme a une catégorie qui pointe vers la catégorie de l'autre terme, (iii) un terme est soit le label de l'autre terme, soit appartient à une catégorie de l'autre terme, (iv) deux mots sont du même groupe et sont donc sémantiquement liés, (v) et les deux mots ont une catégorie qui pointe une catégorie commune. Cette méthode s'appuie également sur la théorie de la centralité (Centering Theory (Grosz *et al.*, 1995)) selon laquelle certaines entités sont plus centrales que d'autres. Des fonctions sont ainsi créées pour intégrer ces théories sous formes de fonctions et contraintes. Les résumés sont ainsi générés par optimisation linéaire en nombres entiers, (ILP, *Integer Linear Programming*). Nous présentons justement dans le paragraphe suivant les méthodes de résumé par optimisation de fonctions.

3.1.5 *Méthodes par optimisation de fonctions*

L'optimisation de fonctions est utilisée pour générer des résumés automatiquement. Les critères qui font un « bon » résumé sont traduits en fonctions avec des objectifs à optimiser, maximiser ou minimiser.

Les critères optimisés par (Alguliev *et al.*, 2013) sont la couverture, la diversité, et la taille du résumé. Les fonctions définies pour les différents critères sont ensuite optimisées sur les données par un algorithme à évolution différentielle (*Differential Evolution*, DE). Sánchez-Gómez *et al.* (2020) évaluent différents critères à optimiser. Comme (Alguliev *et al.*, 2013), la couverture, et la réduction de la redondance sont des critères considérés. En revanche, la pertinence et la cohérence sont également

des critères considérés. Ils définissent des fonctions à optimiser pour chacun de ces critères. L'optimisation est effectuée à l'aide de l'algorithme *Multi-Objective Artificial Bee Colony* (MOABC) (Sánchez-Gómez *et al.*, 2018). Après évaluation des différentes combinaisons des critères, le modèle le plus équilibré est celui qui utilise les critères de couverture, de réduction de la redondance, et de pertinence.

Contrairement à Alguliev *et al.* (2013) et Sánchez-Gómez *et al.* (2020), Peyrard et Gurevych (2018) proposent que la fonction à optimiser ne soit pas définie *a priori*, mais apprise automatiquement à partir de jugements humains. Une fois la fonction à optimiser apprise, les résumés sont générés à l'aide d'un algorithme génétique.

3.1.6 *Méthodes à base de logique floue*

Contrairement à la logique booléenne, où les valeurs possibles d'une variable sont strictement 0 ou 1, la logique floue considère que les valeurs d'une variable sont des réels compris entre 0 et 1. Un système flou comporte 3 parties : (i) la fuzzification qui encode les données pour le moteur d'inférence, entre 0 et 1, (ii) le moteur d'inférence qui suit une succession de règles implémentées à l'aide d'opérateurs de logique floue, (iii) et la défuzzification qui permet de traduire la sortie du moteur d'inférence en données numériques [7].

Dans (Suanmali *et al.*, 2009), 8 caractéristiques de phrases sont considérées : (i) le nombre de termes du titre apparaissant dans une phrase, (ii) la taille d'une phrase, (iii) le poids d'une phrase calculé par Tf-Isf (*Inverse sentence frequency*), (iv) la position d'une phrase dans le document, (v) la similarité d'une phrase aux autres phrases, (vi) le nombre de noms propres, (vii) le nombre de termes thématiques (un terme thématique est un terme dans les 10 termes les plus fréquents du document) dans la phrase, (viii) le nombre de données numériques de la phrase. Ces caractéristiques sont ensuite traitées par un système de logique floue. Le fuzzifieur encode les 8 caractéristiques pour le moteur d'inférence. Le moteur d'inférence applique les règles définies. Le défuzzifieur retourne un score pour chaque phrase. Les premières phrases ordonnées par ordre descendant de score sont retournées pour le résumé. Patel *et al.* (2019) utilisent également 8 caractéristiques avec un modèle de logique floue. Les caractéristiques sont sensiblement les mêmes que dans (Suanmali *et al.*, 2009) mais diffèrent par leur calcul. Par exemple, un terme thématique est un terme dans les 5 termes les plus fréquents et non 10. La génération des résumés est produite à partir des scores retournés par le modèle de logique floue et également par un score de similarité entre les phrases déjà récupérées pour le résumé et les phrases candidates au résumé, à la manière de la MMR (*Maximal Marginal Relevance*) (Carbonell et Goldstein, 1998).

7. `http://www.ferdinandpiette.com/blog/2011/08/les-systemes-flous-le-fonctionnement/`, dernière consultation le 12/08/2022

3.1.7 *Méthodes par apprentissage automatique*

Bien que l'apprentissage automatique ne se limite pas aux approches d'apprentissage profond (*deep learning*), les méthodes de l'état de l'art que nous détaillons dans la section sont essentiellement de cette catégorie car majoritairement présentes aujourd'hui. Avant de détailler les différentes approches de résumé par apprentissage automatique, nous apportons quelques notions utiles à la bonne compréhension des approches. Dans cette section, et dans le reste du manuscrit, nous utilisons la notion de *token*. Un *token* représente l'index d'une unité lexicale, permettant au modèle de l'identifier de manière unique. Globalement, un *token* peut être considéré comme un terme dans ce manuscrit.

3.1.7.1 Notions utiles

Dans cette section, nous présentons rapidement des notions non triviales relatives aux approches neuronales. Nous introduisons différents types de réseaux de neurones, les réseaux de neurones convolutifs (CNN) et récurrents (RNN), et les *Transformers*. Nous introduisons plus globalement les architectures encodeurs-décodeurs dont peuvent faire partie les CNNS, les RNNs et les *Transformers*. Ensuite, nous introduisons les modèles de langue, et nous nous focalisons sur les modèles de langue neuronaux. Les modèles de langue neuronaux sont aujourd'hui essentiellement basés sur les *Transformers*. Enfin, nous présentons l'algorithme *Beam search*. Cet algorithme est utilisé par plusieurs méthodes encodeurs-décodeurs lors de la phase de décodage.

Les réseaux de neurones convolutifs (*Convolutional Neural Networks*) sont des réseaux de neurones pour lesquels des filtres sont appliqués sur les données d'entrée pour extraire des caractéristiques. Les CNNs sont largement utilisés dans le domaine des images, mais ont aussi montré un intérêt pour les données textuelles (Kim, 2014).

Les réseaux de neurones récurrents (*Recurrent Neural Network*) sont des réseaux de neurones utilisés pour traiter des séquences de taille variable, par exemple, pour la génération de texte ou la traduction. Un RNN est un réseau de neurones qui traite l'entrée du réseau séquentiellement et qui, pour chaque itération, prend en compte la prédiction de l'itération précédente. L'idée du réseau de neurones récurrent est de modéliser la mémoire (souvenir) humaine. En effet, en général, nous utilisons nos connaissances pour prendre des décisions. Au sein de cette famille des réseaux de neurones récurrents on retrouve notamment les modèles GRU (*Gated Recurrent Unit*) et LSTM (*Long Short-Term Mermory*). Le LSTM ou en français réseau de neurones récurrent à mémoire court terme et long terme est une variante de RNN. Le LSTM apporte la mémoire à long terme en conservant la mémoire à court terme et apporte la notion d'oubli. Le LSTM contient trois « portes » que sont l'entrée,

l'oubli et la sortie (*input*, *forget* et *output gates*). Le GRU est une autre variante de RNN, il contient deux « portes » qui sont les portes de mise à jour et de remise à zéro (*update* et *reset gates*).

Le *Transformer* est un modèle d'apprentissage profond basé sur le mécanisme d'attention. Le mécanisme d'attention repose sur 3 principaux composants : (i) la requête (*query*), (ii) la clé (*key*), et (iii) la valeur (*value*). Le mécanisme d'attention met en correspondance une requête et un ensemble de paires clé-valeur avec une sortie (Vaswani *et al.*, 2017). En d'autres termes, pour du texte, l'idée du mécanisme d'attention est de pondérer la relation entre un mot m_1 (requête) et un mot m_2 (clé), avec le mot m_2 qui est associé à une certaine valeur. Ainsi, une *attention* plus prononcée est portée sur les relations de mots les plus fortes. Soient Q, K, et V les matrices représentant respectivement les requêtes, les clés, et les valeurs. Soit d_k la dimension des requêtes et des clés. Le score d'attention est calculé comme suit :

$$Attention(Q, K, V) = softmax(\frac{QK^\top}{\sqrt{d_k}})V$$

L'attention du *Transformer* est multi-tête (*Multi-Head Attention*), c'est-à-dire que différentes projections des requêtes, clés, et valeurs sont calculées en parallèles et agrégées ensuite. Soient W^Q, W^K, et W^V les différentes matrices de projections représentant respectivement les différentes représentations des requêtes, des clés, et des valeurs. Soit W^0 la matrice de projection de sortie, permettant l'agrégation des différentes projections. Soit h le nombre de projections parallèles. L'attention multi-tête est calculée comme suit :

$$MultiHead(Q, K, V) = Concat(head_1, \ldots, head_h)W^0$$

avec $head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$.

Contrairement aux réseaux de neurones récurrents, les calculs d'un *Transformer* peuvent être parallélisés. Pour un réseau de neurones récurrent, ce n'est pas possible puisque l'entrée d'une étape correspond à la sortie de l'étape précédente (récurrence).

Une architecture encodeur-décodeur est composée d'une partie encodeur et d'une partie décodeur. La partie encodeur représente l'entrée dans un espace latent, et le décodeur traduit cette représentation dans un espace cible en sortie. Par exemple, pour de la traduction de texte, prenons une phrase en français que nous souhaitons traduire en anglais. La phrase en français est représentée dans un espace latent par l'encodeur. Le décodeur, à partir de cette représentation latente retourne la phrase en anglais. Les architectures encodeur-décodeur sont différenciées selon au moins 3 catégories [8] :

8. `https://huggingface.co/docs/transformers/model_summary`, dernière consultation le 14/08/2022

— les modèles séquence à séquence (*sequence-to-sequence, Seq2Seq*) : ils sont utilisés pour produire une séquence à partir d'une autre séquence.
— Les modèles auto-régressifs (*AutoRegressive*) : ils sont entraînés pour prédire un *token* à partir de ceux qui le précèdent.
— Les modèles auto-encodeurs (*AutoEncoder*) : ils sont entraînés pour reconstruire une séquence originale à partir de la séquence altérée. Contrairement aux modèles auto-régressifs, ils ne doivent pas prédire des *tokens* futurs et ont accès à une séquence complète. Ainsi, popularisé par BERT, les architectures auto-encodeurs sont souvent bidirectionnelles, c'est-à-dire qu'un *token* est considéré par rapport aux *tokens* qui le précèdent et qui le suivent dans une séquence.

Les modèles de langue sont des modèles statistiques qui modélisent la distribution de mots ou de séquences de mots d'une langue. L'état de l'art des modèles de langue est aujourd'hui basé sur les réseaux de neurones [9]. Nous pouvons citer les modèles de langue basés sur des RNN, avec des architectures LSTM ou GRU par exemple (Cheng et Lapata, 2016; Nallapati *et al.*, 2017). Plus récemment, les modèles de langue se basent sur les *Transformers*, comme BERT, GPT, BART, ou encore T5 :

— BERT (Devlin *et al.*, 2019) est un modèle de langue auto-encodeur de 110 millions de paramètres pour la version de base et 340 millions pour la version large. Le modèle est bidirectionnel. L'entraînement de BERT est guidé par les objectifs MLM (*Masked Language Model*) et NSP (*Next Sentence Prediction*). Le MLM consiste à masquer des *tokens* dans une séquence que le modèle doit ainsi prédire par rapport aux autres *tokens* de la séquence. Le NSP demande au modèle de prédire si 2 séquences sont successives ou non dans un texte. La taille maximale du texte encodé par BERT est de 512 *tokens*.
— GPT-3 (Brown *et al.*, 2020) est un modèle de langue auto-régressif qui est composé de 175 milliards paramètres, 10 fois plus que les autres modèles existant au moment de sa sortie. La taille maximale du texte d'entrée et de sortie de GPT-3 est de 2048 *tokens*.
— BART (Lewis *et al.*, 2020) est un modèle de langue séquence à séquence, un auto-encodeur spécifiquement entraîné pour la génération de texte. BART est composé de 140 millions paramètres pour la version de base et 400 millions pour la version large. L'architecture de BART utilise un encodeur bidirectionnel, comme BERT, et un décodeur unidirectionnel tel GPT. Le pré-entraînement de BART utilise des objectifs de masquage de *tokens*, suppression de *tokens*, remplissage de texte, permutation de phrases, et rotation de document. La taille maximale du texte pour BART est de 1024 *tokens*.
— T5 (Raffel *et al.*, 2020) est un modèle de langue séquence à séquence de 11 milliards de paramètres. L'objectif d'apprentissage du modèle est de prédire

9. `https://www.smalsresearch.be/nlp-modeles-de-langue/`, dernière consultation le 14/08/2022

les mots masqués de la séquence d'entrée, inspiré du MLM de BERT. La taille maximale du texte pour T5 est de 512 *tokens*.

Ces modèles de langue sont utilisables pré-entraînés, c'est-à-dire que les modèles sont applicables tel que pour modéliser le langage. Il est aussi possible d'*ajuster* (*fine-tuning*) ces modèles pré-entraînés pour une tâche souhaitée (Devlin *et al.*, 2019). À partir du modèle pré-entraîné, un apprentissage supplémentaire est effectué pour affiner l'entraînement par rapport à la tâche. GPT-3, BART et T5 sont des modèles orientés génération de texte et sont plus adaptés et utilisés pour du résumé abstractif qu'extractif.

L'algorithme *Beam search* ou recherche en faisceau, est un algorithme de recherche dans un graphe. *Beam search* prend en compte un nombre limité de possibilités à chaque étape du parcours du graphe. Nous illustrons l'algorithme par un exemple avec un faisceau de largeur 3, ce qui signifie qu'à chaque étape les 3 possibilités les plus probables sont gardées, pour une traduction d'un texte anglais vers français :

Soit un exemple avec la phrase à traduire suivante : « I am going to play soccer with my friends ».

- Étape 1 : Soit les 3 possibilités les plus probables pour le mot1 : {Je, J', Moi}. Pour la prochaine étape, nous considérons donc les combinaisons Je + mot2, J' + mot2 et Moi + mot2 les plus probables.
- Étape 2 : Soit les 3 combinaisons les plus probables pour mot1 et mot2 : {Je suis, Je vais, J'ai}. Il est à noter que l'on garde les 3 combinaisons les plus probables parmi toutes les combinaisons testées, pas la combinaison la plus probable pour chaque mot. Ici, les combinaisons Moi + mot2 sont toutes moins probables que Je suis, Je vais et J'ai. Pour la prochaine étape, on regarde donc les combinaisons Je suis + mot3, Je vais + mot3 et J'ai + mot3 les plus probables.
- ...
- Étape finale : Soit les 3 combinaisons les plus probables finales :
  - Je vais jouer au football avec mes amis : probabilité 0.8
  - Je vais pratiquer le football avec mes copains : probabilité 0.6
  - J'ai des amis pour jouer au football : probabilité 0.3

  « Je vais jouer au football avec mes amis » est donc la traduction retournée.

3.1.7.2 Approches de résumé automatique

PriorSum (Cao *et al.*, 2015) utilise des caractéristiques dépendantes des documents dont la position de la phrase dans le document et la moyenne du Tf des termes de la phrase, et des caractéristiques extraites à partir d'un réseau de neurones convolutif (CNN). Ces caractéristiques sont ensuite combinées pour obtenir un score par phrase, score qui est ensuite utilisé pour générer les résumés à la manière de la MMR. Cheng

et Lapata (2016) utilisent également un CNN mais pour représenter les phrases. Ils classifient des phrases à 0 ou 1, si elles doivent être gardées pour le résumé ou non. La représentation des phrases est effectuée à partir d'un CNN au niveau des termes. À partir de la représentation des phrases, la représentation du document est obtenue à l'aide d'un réseau de neurones récurrents (RNN) LSTM (*Long Short-Term Memory*). Cette représentation est utilisée par un autre LSTM pour parcourir les phrases du document et les classer pour être gardées (1) ou non (0) dans le résumé. Dans la même idée, Nallapati *et al.* (2017) classifient qu'une phrase doit être gardée pour un résumé ou non à l'aide de 2 couches RNN. La première couche représente les phrases à partir de termes, la seconde couche classifie l'ajout de la phrase au résumé ou non à partir des phrases. Les RNN utilisés sont des GRU (*Gated Recurrent Unit*) bidirectionnels (lecture des termes puis des phrases de gauche à droite et de droite à gauche). Dans les travaux (Ren *et al.*, 2017), les phrases sont également représentées à l'aide de CNN et de LSTM. Cependant, ils ne classifient pas qu'une phrase doit être gardée ou non, mais prédit un score pour chaque phrase. Les résumés sont produits en 2 phases : prédiction de score pour chaque phrase, et sélection des phrases. Les scores des phrases sont prédits en prenant en compte leur contexte, à partir des phrases précédentes et suivantes dans le document. Une fois les scores prédits, les résumés sont générés à la manière de la MMR.

Plus récemment, les approches s'appuient sur les modèles de langue neuronaux, notamment BERT. Le travail de Liu et Lapata (2019), BertSum, utilise BERT pour encoder des documents pour du résumé extractif et abstractif. Plus précisément, un *token* est utilisé pour séparer les phrases, permettant au modèle d'encoder tout le document, tout en différenciant les différentes phrases. La version extractive du modèle prédit un score pour chaque phrase. À partir du score prédit pour chaque phrase, les résumés sont créés à la manière de la MMR (Carbonell et Goldstein, 1998). Les phrases contenant un trigramme déjà présent dans le résumé ne sont pas ajoutées au résumé. Le document à résumer est tronqué si sa taille (en nombre de *tokens*) est supérieure à une taille limite (ils étendent la limite de 512 *tokens* de BERT à une limite de 768 *tokens*). Le modèle proposé par Song *et al.* (2020) pour le résumé abstractif est utilisable pour du résumé extractif. Ce modèle peut être paramétré pour que les résumés générés copient plus ou moins le document d'entrée, qu'ils soient purement extractifs ou hautement abstractifs. Le modèle est basé sur une architecture encodeur-décodeur. L'encodage est réalisé à partir de BERT. Lors du décodage, l'algorithme *Beam search* est notamment utilisé. Ce modèle est évalué sur le résumé de phrase, c'est-à-dire « condenser une longue phrase source en un résumé ressemblant à un titre » (Song *et al.*, 2020). L'approche présentée par Zhong *et al.* (2020) est plutôt un méta-modèle. Elle sélectionne le « meilleur » résumé parmi un ensemble de résumés candidats en sélectionnant le candidat le plus proche du document à résumer. L'ensemble des résumés candidats est généré comme suit : (i) un score est prédit pour chaque phrase à l'aide de BertSum (Liu et Lapata, 2019),

(ii) les phrases les moins bien notées sont enlevées des phrases potentielles au résumé, (iii) parmi les phrases restantes, l'ensemble des combinaisons de phrases de la taille du résumé souhaité (avec les phrases dans l'ordre d'apparition dans le document) sont générées comme résumés candidats. Le modèle est basé sur les représentations sémantiques originales de BERT.

3.2 *Méthodes abstractives*

Comme notre contribution est basée sur des méthodes extractives, nous ne détaillerons pas les méthodes de résumé abstractives dans ce manuscrit. Nous présentons en revanche un aperçu des méthodes les plus récentes. El-Kassas *et al.* (2021) classent les méthodes abstractives comme suit :

— Méthodes basées graphes,
— méthodes basées arbres,
— méthodes basées règles,
— méthodes basées patrons (*template*),
— méthodes basées ontologies,
— méthodes basées sémantiques,
— et méthodes basées apprentissage profond.

Les méthodes basées sur l'apprentissage profond sont aujourd'hui les plus courantes.

PEGASUS (Zhang *et al.*, 2020a), est une approche de résumé abstractive encodeur-décodeur. Cette approche ajoute un nouvel objectif d'apprentissage, le *Gap Sentences Generation*, qui masque des phrases entières (cet objectif diffère du *Masked Language Model* de BERT qui masque des *tokens*). Lors du décodage, PEGASUS utilise également l'algorithme *Beam search*. La taille maximale des documents à résumer de PEGASUS est de 512 ou 1024 *tokens*, suivant que le modèle de base ou le modèle large soit utilisé. Inspirés par PEGASUS (Zhang *et al.*, 2020a), Xiao *et al.* (2022) proposent d'améliorer le *Gap Sentences Generation*. Les stratégies de sélection des phrases masquées (Random : aléatoirement, LEAD : les premières phrases, Principle : score de similarité par rapport aux autres phrases du document à partir de la mesure ROUGE) sont considérées comme sous-optimales. Ainsi, ils proposent d'utiliser la méthode Pyramid (voir section 4.2) pour choisir les phrases à masquer. Les annotations manuelles nécessaires (appelées SCUs) pour la méthode Pyramid sont remplacées par des entités extraites automatiquement. L'encodage des documents utilise LED (*Longformer-Encoder-Decoder* (Beltagy *et al.*, 2020)) et permet d'encoder un maximum de 4096 *tokens*.

L'approche de résumé abstractive ProphetNet (Qi *et al.*, 2020) intègre un objectif de prédictions de n-grammes. Lors du décodage, au lieu de ne prédire qu'un seul *token* à chaque étape, le modèle est entraîné pour prédire *n-tokens*. Ce modèle séquence à séquence pré-entraîné possède également une structure d'encodeur-décodeur *Transformer*.

Le *framework* de résumé abstractif GSum (Dou *et al.*, 2021) s'attaque aux erreurs factuelles dans les résumés générés abstractivement. Pour ce faire, l'entrée du modèle prend le document source ainsi que des indications externes, comme des mots-clés, ou même le résumé Oracle pendant l'apprentissage. Le modèle est évalué selon différentes versions basées sur BERTSum et BART.

4 Évaluation

Quelle que soit la tâche que l'on souhaite automatiser, modéliser, un point essentiel est l'évaluation. En effet, il est impossible de juger de l'efficacité d'un modèle à résoudre notre problème si on ne l'évalue pas. On peut vouloir évaluer la qualité *intrinsèque* ou *extrinsèque* d'un résumé, c'est-à-dire évaluer la qualité d'un résumé en lui-même, comme sa cohérence, son informativité, ou évaluer l'utilité d'un résumé dans un contexte applicatif donné, comme la compréhension de lecture (Lloret *et al.*, 2018). Nous nous focalisons ici sur les mesures d'évaluations intrinsèques, qui sont beaucoup plus présentes dans la littérature. Au sein de ces évaluations intrinsèques, nous retrouvons des mesures d'évaluation *automatique* et *manuelle*. Dans cette section, nous décrivons dans un premier temps l'évaluation automatique des résumés, le processus, puis les différentes mesures existantes. Ensuite, nous exposons les différentes évaluations manuelles proposées dans la littérature.

Avant de poursuivre, il est utile d'indiquer la distinction que nous considérons entre évaluation manuelle et automatique dans ce manuscrit. Cette distinction n'est pas très claire dans la littérature et certaines méthodes d'évaluation sont considérées manuelles, automatiques, ou semi-automatiques dans certains travaux. Par exemple, les mesures ROUGE, que nous considérons ici automatiques et que nous présentons en section 4.1.2, sont considérées automatiques dans la plupart des travaux, comme (Gambhir et Gupta, 2017; El-Kassas *et al.*, 2021), mais certains travaux comme (Mnasri, 2018) les considèrent semi-automatiques. Aussi, la mesure Pyramid que nous considérons manuelle est présentée semi-automatique dans (Gambhir et Gupta, 2017; Mnasri, 2018) mais manuelle dans (El-Kassas *et al.*, 2021).

Ici, nous considérerons les mesures manuelles comme les mesures dépendantes de l'évaluateur. Autrement dit, si l'évaluation est menée par un évaluateur E1 à un instant t, le résultat ne sera pas forcément le même que dans le cas où l'évaluation

est menée par un évaluateur E2, voire même par le même évaluateur E1 mais à un instant t+n. Au contraire, nous considérons les mesures automatiques comme reproductibles, avec (si besoin) ou sans données de référence, que l'évaluation soit menée à un instant t ou t+n, par un évaluateur E1 ou E2.

4.1 *Évaluation automatique*

Généralement, les méthodes d'évaluation automatique ont l'avantage d'être moins coûteuses que des évaluations manuelles, les évaluations manuelles nécessitant un effort humain important. Un autre avantage est la portabilité ou reproductibilité de l'évaluation, puisqu'il suffit d'utiliser la mesure, indépendamment de l'évaluateur ou de l'instant d'évaluation.

4.1.1 *Processus d'évaluation automatique*

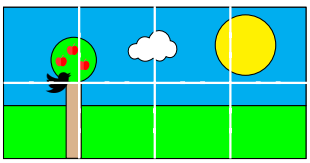
Le processus d'évaluation le plus courant en résumé automatique de texte consiste à évaluer les résumés générés par rapport à des résumés références (*Gold Standard* - G.S. ou *Groud Truth* dans la littérature). De fait, les résumés générés les plus similaires à ces résumés de référence sont considérés comme « meilleurs » selon la mesure de similarité évaluée. Le processus est illustré en figure 3.4 à l'aide d'un puzzle et de plusieurs candidats à la résolution de ce puzzle. L'évaluation des différentes solutions est reportée selon une mesure choisie basiquement, mais avec une autre mesure de similarité, les scores auraient peut-être été différents et la solution considérée comme la « meilleure » différente également. Il faut cependant noter que la création de ces résumés références est souvent manuelle bien que la méthode d'évaluation soit automatique.

4.1.2 *Mesures*

Les principales mesures d'évaluation automatique de résumé utilisées dans la littérature sont les mesures issues de ROUGE (Recall-Oriented Understudy for Gisting Evaluation) (Lin, 2004). ROUGE est un ensemble de mesures basées sur le chevauchement de mots entre deux textes. Dans un premier temps, des éléments de comparaison sont extraits des deux textes. Ces éléments de comparaison sont des mots ou suites de mots. Plusieurs variantes existent :

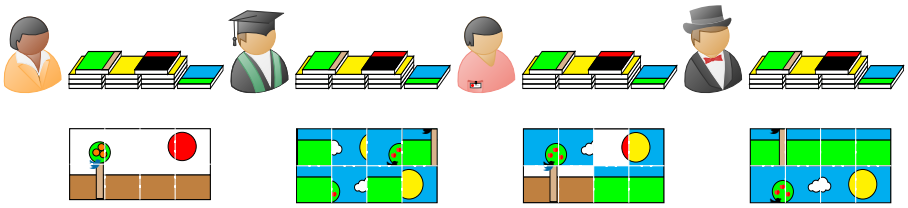
- ROUGE-N : Les éléments sont les suites de N mots du texte, ou n-grammes.
  - ROUGE-1 : Les éléments sont les mots du texte, ou unigrammes.
  - ROUGE-2 : Les éléments sont les suites de mots du texte de taille 2, ou bigrammes.

Ensemble des pièces de puzzle disponibles
↔ Ensemble de documents disponibles.



Résolution du puzzle référence.
↔ Résumé de référence.

Candidats qui essaient de résoudre le puzzle ↔ Méthodes de résumé automatique.

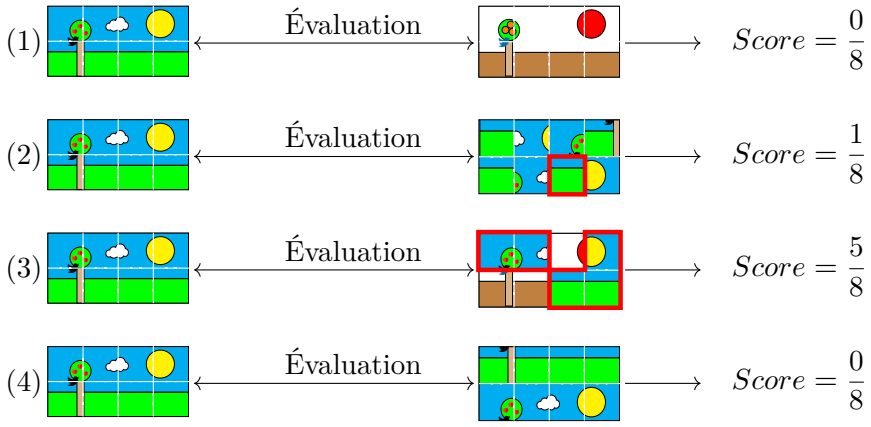


Résolution de puzzle proposée par chaque candidat
↔ Résumé retourné par chaque méthode.

Évaluation de la qualité des puzzles proposés par les différents candidats.
↔ Évaluation de la qualité des résumés générés par les différentes méthodes.

Exemple d'évaluation des puzzles proposés à l'aide d'une mesure naïve :

$$Score = \frac{nombre\ de\ pièces\ identiques\ et\ de\ même\ position\ que\ la\ référence}{nombre\ de\ pièces\ du\ puzzle}$$



Dans les cas (1) et (3) les pièces non identiques sont de couleurs différentes à la référence.
Dans les cas (2) et (4) les pièces non identiques sont de positions différentes à la référence.

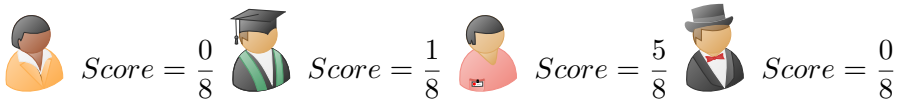


FIGURE 3.4 – Illustration du processus d'évaluation de méthodes de résumé automatique.

— ROUGE-L : Les éléments sont les mots de la plus grande suite de mots (pas forcément consécutifs mais dans le même ordre) commune aux deux textes.
— ROUGE-S* : Les éléments sont les suites de mots de taille 2 avec un intervalle de taille 0 à * entre les deux mots, aussi appelés skip-bigrammes.
— ROUGE-SU* : Les éléments sont les suites de mots de taille 2 avec un intervalle de taille 0 à * entre les deux mots, ainsi que tous les mots du texte.

Ensuite, pour chaque variante, on peut calculer la *Précision*, le *Rappel*, et la *F-mesure* à partir des listes d'éléments construites sur les différents textes. Dans notre cadre d'évaluation d'un résumé généré par un système comparé à un résumé de référence, la précision est le nombre d'éléments communs aux deux résumés divisé par le nombre d'éléments du résumé généré :

$$Précision = \frac{\text{nombre d'éléments communs}}{\text{nombres d'éléments pour le système}}$$

Le rappel est le nombre d'unités communes aux deux résumés divisé par le nombre d'éléments du résumé référence :

$$Rappel = \frac{\text{nombre d'éléments communs}}{\text{nombres d'éléments pour la référence}}$$

La F1-mesure, souvent appelée F-mesure est la moyenne harmonique de rang 1 de la précision et du rappel :

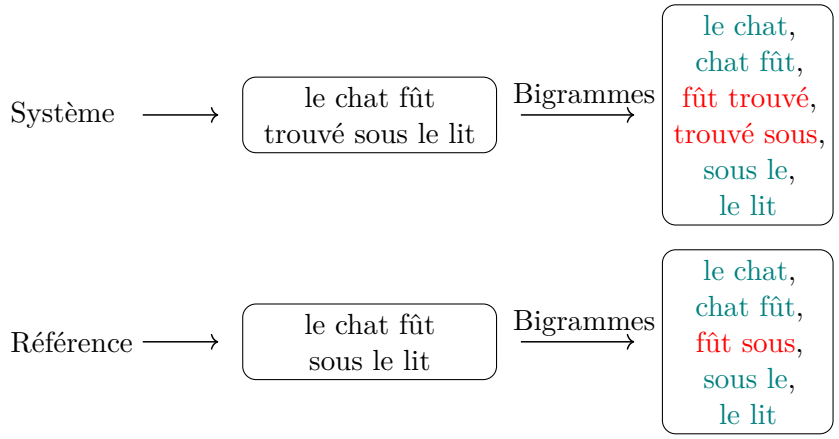
$$F1 - mesure = \frac{2 * (ROUGE - 2_{Précision} * ROUGE - 2_{Rappel})}{ROUGE - 2_{Précision} + ROUGE - 2_{Rappel}}$$

Nous présentons en figure 3.5 un exemple d'évaluation de la similarité entre deux textes selon la mesure ROUGE-2. Lorsque plusieurs références sont disponibles, pour les différentes variantes de ROUGE, Lin (2004) propose le score de similarité maximal entre le résumé généré et chacun des résumés références, évalués deux à deux.

Bien que les métriques issues de ROUGE soient les plus utilisées dans la littérature pour évaluer le résumé automatique (El-Kassas *et al.*, 2021), elles sont tout de même décriées. ROUGE ne correspond qu'à un chevauchement de mots, et se retrouve limité lors de l'utilisation de synonymes ou d'expressions sémantiquement similaires mais syntaxiquement différentes des résumés de référence. Aussi, ROUGE est plus efficace lorsqu'il y a plusieurs résumés de référence, ce qui n'est communément pas le cas du fait de la difficulté à créer de tels résumés (Louis et Nenkova, 2013).

D'autres mesures d'évaluation automatiques existent (Ermakova *et al.*, 2019) mais sont beaucoup moins utilisées que ROUGE :

— Similarité cosinus entre la référence et le résumé évalué, dans un espace commun sous forme vectorielle, par exemple avec une représentation TF-Idf ou de



$$ROUGE-2_{Précision} = \frac{\text{nombre de bigrammes communs}}{\text{nombres de bigrammes pour le système}} = \frac{4}{6} \approx 0{,}67$$

$$ROUGE-2_{Rappel} = \frac{\text{nombre de bigrammes communs}}{\text{nombres de bigrammes pour la référence}} = \frac{4}{5} = 0{,}8$$

$$ROUGE-2_{F} = \frac{2 * (ROUGE-2_{Précision} * ROUGE-2_{Rappel})}{ROUGE-2_{Précision} + ROUGE-2_{Rappel}} = \frac{8}{11} \approx 0{,}73$$

FIGURE 3.5 – Exemple d'évaluation de résumé à l'aide de la mesure ROUGE-2.

plongements lexicaux (*embeddings*). Soit les représentations vectorielles de la référence **A**, et du résumé évalué **B**, le score de similarité Cosinus est calculé :

$$sim_{cos} = \frac{A \cdot B}{||A||\ ||B||}$$

— Les similarités à partir des coefficients de Jaccard et Dice. Comme ROUGE, ces mesures de similarité sont basées sur des représentations de type sac de mots. Soit les représentations sac de mots de la référence **A**, et du résumé évalué **B**, les similarités sont calculées comme suit :

$$sim_{Jaccard} = \frac{|A \cap B|}{|A \cup B|}$$

$$sim_{Dice} = 1 - \frac{2|A \cap B|}{|A| + |B|}$$

— Les divergences de Kullback-Leibler et Jensen-Shannon qui sont à minimiser. Elles calculent la différence des distributions de mots entre les résumés de référence et le résumé à évaluer. Soit les représentations sac de mots de la référence **A**, et du résumé évalué **B**, $p_A(\mathrm{w})$ et $p_B(\mathrm{w})$ les probabilités de voir le

mot w respectivement dans **A** et dans **B**, les divergences sont calculées comme suit :

$$Div_{KL}(B||A) = \sum_{w \in A \cup B} p_B(w) log \frac{p_B(w)}{p_A(w)}$$

$$Div_{JS}(B||A) = \frac{1}{2} Div_{KL}(B||\frac{1}{2}(A+B)) + \frac{1}{2} Div_{KL}(A||\frac{1}{2}(A+B))$$

— BE (Basic Elements) (Hovy *et al.*, 2006) est une mesure de similarité basée sur l'utilisation de petites unités d'information, des BEs. Un BE peut être : (i) une « tête », soit, un nom, un verbe, un adjectif, ou un adverbe, (ii) ou une « relation » entre une « tête » et un « modificateur » (*head - modifier - relation*). Les BEs sont extraits pour le résumé de référence et le résumé évalué, puis un score est calculé par correspondance entre les BEs.

— BLEU (BiLingual Evaluation Understudy) (Papineni *et al.*, 2002) ou METEOR (Metric for Evaluation of Translation with Explicit ORdering) (Banerjee et Lavie, 2005) sont des métriques initialement créées pour l'évaluation de traduction automatique de texte et sont parfois utilisées pour l'évaluation du résumé automatique (Ermakova *et al.*, 2019).

— La métrique GRAD (GRAph Distance) vise à estimer le degré de connexion entre les termes du résumé évalué et ceux du texte à résumer (Ermakova et Firsov, 2018). GRAD est construit selon l'hypothèse qu'un bon résumé est constitué des termes qui sont les plus connectés aux termes du texte à résumer. GRAD utilise un graphe sémantique et calcule les distances entre les termes au sein du graphe.

— La métrique GEM (GEnerosity Measure) (Ermakova *et al.*, 2018) est une mesure pour le résumé d'articles scientifiques. Cette métrique s'appuie sur l'importance des différentes sections d'un article scientifique.

— BERTScore (Zhang *et al.*, 2020b) est une méthode d'évaluation du résumé automatique basée sur BERT. BERTScore calcule un score Cosinus entre les représentations générées par BERT pour la référence et le résumé évalué. BERTScore calcule un score de précision, un score de rappel, et un score F calculé, comme pour ROUGE, par la moyenne harmonique de la précision et du rappel. Basée sur BERT, la taille limite des résumés de référence et des résumés à évaluer par BERTScore est de 512 *tokens*.

— QuestEval (Scialom *et al.*, 2021) est une méthode d'évaluation basée sur le principe de Question-Réponse. QuestEval génère des questions à partir des documents à résumer et des résumés à évaluer. Ensuite, QuestEval répond à ces questions à partir des résumés à évaluer si les questions ont été générées à partir des documents à résumer, et inversement. Un score est enfin calculé à partir des réponses retournées. Comme BERTScore, QuestEval, qui utilise

T5 (Raffel *et al.*, 2020), est limitée par la taille des documents et résumés à évaluer, à 512 *tokens*.

4.2 *Évaluation manuelle*

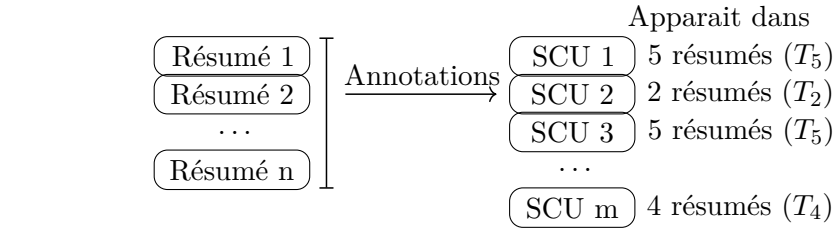
L'évaluation automatique des résumés présente des avantages en termes de temps et d'efforts nécessaires. Cependant, les différentes approches existantes présentent leurs limites. ROUGE, l'ensemble de métriques le plus utilisé, est contesté (Louis et Nenkova, 2013; Paulus *et al.*, 2018). Les autres métriques proposées pour pallier à ces limites, comme BertScore, BLEU, ou METEOR ne montrent pas de meilleures performances que ROUGE par rapport aux annotations humaines (Fabbri *et al.*, 2021). Aussi, bien qu'elle ne soient pas ou difficilement reproductibles, plus coûteuses, les évaluations manuelles demeurent importantes en résumé automatique.

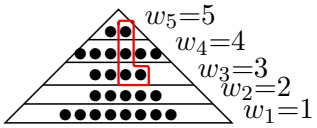
La méthode Pyramid (Nenkova et Passonneau, 2004) permet d'évaluer les résumés manuellement. L'évaluation se déroule en trois temps. Dans un premier temps, des annotateurs doivent construire chacun un résumé à partir des documents (ou du document s'il n'y en a qu'un seul) à résumer. Ensuite, à partir de ces résumés construits, des SCUs (*Summarization Content Unit*), qui représentent des unités d'information sont extraites par annotation humaine. En d'autres termes, des annotateurs identifient les différentes unités d'information importantes des résumés construits. Une même SCU peut apparaître dans plusieurs résumés différents, chaque résumé relatant des mêmes documents. Une *pyramide* est ainsi générée par rapport au nombre de résumés dans lesquels apparaissent les SCUs. Chaque *niveau* de la pyramide comporte les SCUs qui apparaissent dans un même nombre de résumés. Plus le nombre de résumés dans lesquels une SCU apparaît est élevé, plus la SCU est haute dans la pyramide. Intuitivement, plus une SCU apparaît dans un nombre élevé de résumés, plus un grand nombre d'annotateurs ont identifié cette information importante, et plus cette information paraît essentielle. Enfin, pour chaque résumé généré automatiquement, en d'autres termes, pour chaque résumé à évaluer, une annotation est nécessaire pour extraire les SCUs qui le composent. Le score Pyramid, compris entre 0 et 1, maximisé par 1, est ainsi calculé pour un résumé :

$$Pyramid = \frac{\sum_{i=1}^{n} w_i \times D_i}{\sum_{i=j+1}^{n} i \times |T_i| + j \times (X - \sum_{i=j+1}^{n} |T_i|)}, \text{ où j} = \max_i (\sum_{t=i}^{n} |T_t| \geq X)$$

avec n le nombre de niveaux de la pyramide, w_i le poids affecté au niveau i de la pyramide (les auteurs proposent basiquement $w_i = i$ mais suggèrent que d'autres poids peuvent être utilisés, ceux-ci étant les plus intuitifs et naturels (Nenkova et

Passonneau, 2004)), $|T_i|$ le nombre de SCUs au niveau T_i, D_i le nombre de SCUs du résumé au niveau T_i, et X la taille du résumé en nombre de SCUs. Nous présentons un exemple de calcul du score Pyramid en figure 3.6.





Soit un résumé contenant 4 SCUs, 1 au niveau 5, 2 au niveau 4, et 1 au niveau 3, représenté cerclé de rouge dans la pyramide ci-contre. Le score pyramide de ce résumé est :

$$\frac{1\times 0+2\times 0+3\times 1+4\times 2+5\times 1}{5\times 2+4\times(4-2)}=\frac{16}{18}\approx 0{,}89$$

avec j=4 car au niveau 5 : 2 SCUs $<$ 4 SCUs
mais au niveau 4 : 2 SCUs + 6 SCUs $\geq$ 4 SCUs

FIGURE 3.6 – Exemple d'évaluation de résumé avec la méthode Pyramid.

Une autre méthode pour évaluer les résumés manuellement est utilisée dans (Mani *et al.*, 2002; Clarke et Lapata, 2010; Narayan *et al.*, 2018b). Les évaluateurs doivent répondre à des questions après avoir lu les résumés sans avoir lu le texte en entrée. L'évaluation se déroule en deux étapes, dans un premier temps des questions doivent être rédigées à partir des documents d'entrée. Dans un second temps, des évaluateurs qui n'ont pas connaissance des documents d'entrée doivent répondre à ces questions. Les réponses sont évaluées ainsi, 1, 0,5, et 0, respectivement pour une réponse vraie, partiellement vraie, et fausse. L'approche d'évaluation automatique QuestEval (Scialom *et al.*, 2021), que nous avons mentionné en section 4.1.2 est une proposition d'automatisation de ces évaluations questions-réponses.

Aussi, Lloret *et al.* (2011) proposent une évaluation avec une échelle de satisfaction pour le résumé d'article de recherche. Les évaluateurs doivent répondre aux questions suivantes après avoir pris connaissance des résumés :

— Le résumé reflète les points les plus importants du document.
— Le résumé permet au lecteur de savoir de quoi parle l'article.
— Après avoir lu le résumé original (aussi appelé *abstract*) fourni avec l'article, le résumé alternatif (le résumé évalué) est également valable.

Les évaluateurs peuvent répondre à ces questions à l'aide de l'échelle de Likert (Likert, 1932), selon cinq choix :

— Pas du tout d'accord

— Pas d'accord
— Ni en désaccord ni d'accord
— D'accord
— Tout à fait d'accord

5 Conclusion

Nous avons abordé dans ce chapitre les notions élémentaires du résumé automatique. Nous avons présenté différentes classifications très utilisées dans le cadre du résumé automatique, notamment les distinctions de résumé abstractif ou extractif et mono-document ou multi-document. Dans cette thèse nous nous focalisons sur : (i) le résumé extractif, car il est compliqué d'obtenir des résumés abstractifs de haute qualité (El-Kassas *et al.*, 2021)), (ii) multi-document, inhérent aux données manipulées. Nous avons également présenté les différentes mesures d'évaluation des résumés automatiques, qu'elles soient manuelles ou automatiques. Nous avons souligné que, bien que décriée, l'évaluation des résumés par ROUGE est la plus utilisée pour le résumé automatique. Finalement, nous avons exploré les différentes méthodes de résumé automatique de l'état de l'art. Nous remarquons que ces approches utilisent aujourd'hui largement les modèles de langue neuronaux. Nous avons abordés dans ce chapitre les approches qui étudient le résumé de (corpus de) documents « traditionnels », sans notion de temporalité. Cependant, la notion de temporalité est présente lorsqu'on manipule des données issues de médias sociaux. Ainsi, dans le prochain chapitre, nous prêtons attention à la prise en compte de la dimension temporelle dans le résumé automatique.

4

PRISE EN COMPTE DE LA DIMENSION TEMPORELLE POUR LE RÉSUMÉ

Sommaire

Introduction

Ni le résumé mono-document ni le résumé multi-document ne tiennent compte de la dimension temporelle intrinsèque à la nature des documents (par exemple des news ou des tweets), ou à celle des collections (flux de documents), ou encore aux problématiques traitées (création d'un résumé selon un frise chronologique par exemple). Dans ce chapitre, nous discutons de l'impact de cette dimension temporelle sur le résumé automatique. L'intérêt de l'ajout de la dimension temporelle s'accorde avec le résumé d'évènements ou de sujets précis, permettant de suivre leur évolution au cours du temps.

1 Résumés avec notion de temporalité

McCreadie *et al.* (2018) distinguent 4 types de résumé différents :

— Le résumé multi-document (MDS, *Multi Document Summarization*) : les approches de MDS prennent en entrée un ensemble de documents d'un sujet ou évènement spécifique et visent uniquement à les résumer.

— Le résumé chronologique (en tant que spécialisation du MDS, *Timeline Summarization*) : les approches prennent en entrée un ensemble de documents sur un sujet ou évènement spécifique et visent à les résumer en suivant une chronologie.

— Mise à jour de résumé (également une spécialisation de MDS, *Update Summarization*) : les approches prennent en entrée un flux de documents sur un sujet ou évènement spécifique et visent à le résumer de manière incrémentale.

— Le résumé temporel (*Temporal Summarization*) : les approches prennent également en entrée un flux de documents, mais un problème de filtrage est ajouté au résumé, car le flux ne concerne pas uniquement un sujet ou un évènement spécifique.

La figure 4.1 situe cette classification par rapport aux approches de résumés classiques. La figure 4.2 illustre cette classification, et nous discutons des travaux de l'état de l'art dans les paragraphes suivants.

1.1 *Approches de résumés chronologiques*

Étant donné un flux d'entrée, le résumé chronologique vise à présenter les résultats le long d'une frise chronologique. Tran *et al.* (2015) ont souhaité générer des résumés chronologiques d'évènements. Ils ont utilisé uniquement les titres des articles de

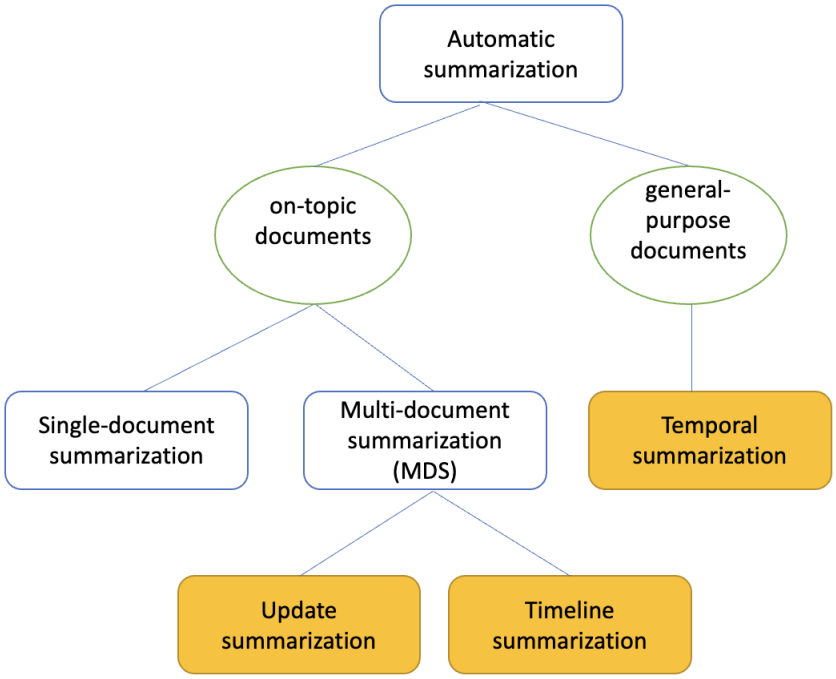


FIGURE 4.1 – Classification générale des approches de résumé automatique. Les approches en jaune prennent en compte la facette temporelle des documents.

presse au lieu du texte entier et ont appliqué un algorithme PageRank personnalisé. Martschat et Markert (2018) ont adapté un modèle d'optimisation multi-documents au résumé chronologique en ajoutant des informations temporelles aux fonctions objectifs.

Zubiaga *et al.* (2012) résument des évènements programmés. Dans ces travaux, les résumés sont générés pour des évènements sportifs à partir de tweets. Dans un premier temps, le modèle détecte l'apparition d'un sous-évènement de l'évènement dans le changement d'activité au sein du flux de tweets. Ensuite, le modèle sélectionne le tweet qui représente le mieux le sous-évènement. Deux versions sont proposées pour choisir le tweet le plus représentatif, basées sur la somme des poids des termes qui composent un tweet : (i) les poids sont les fréquences des termes générés dans la dernière minute, (ii) les poids sont calculés par la divergence de Kullback-Leibler, entre la fréquence des termes dans l'ensemble du flux du sous-évènement et la fréquence des termes dans la dernière minute. Les résultats montrent que la prise en compte de l'information de l'ensemble du flux du sous-évènement impacte positivement la sélection du tweet représentatif.

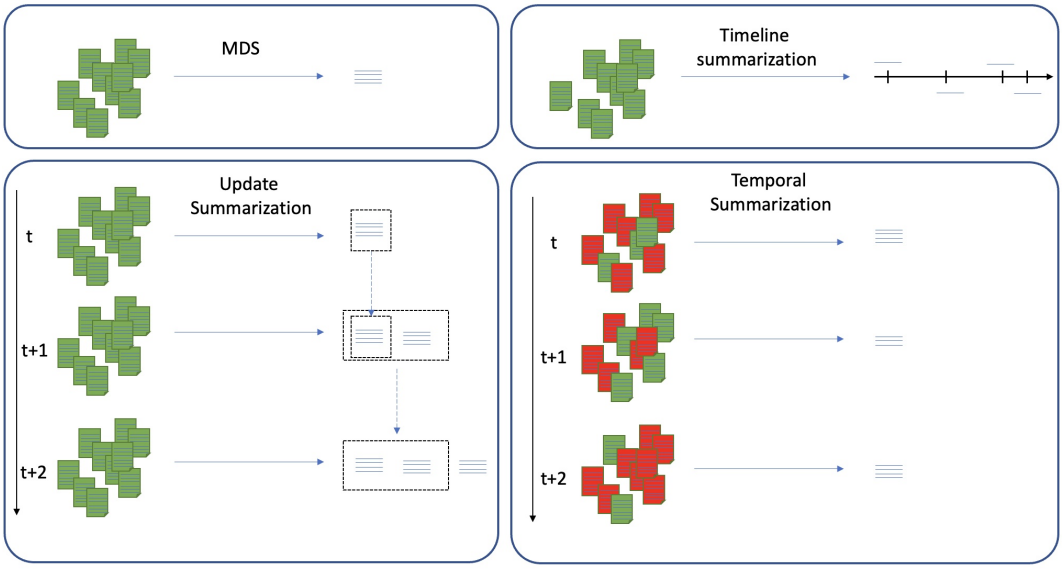


FIGURE 4.2 – Différents groupes d'approches de résumé concernant l'aspect temporel. Les documents d'entrée en vert concernent un sujet ou un évènement spécifique alors que ceux en rouge sont d'ordre général. Les résumés de mise à jour et temporel considèrent des flux de documents, tandis que le résumé multi-document et le résumé chronologique produisent des résumés *a posteriori*.

Une autre approche qui pourrait entrer dans cette catégorie est celle de Ansah *et al.* (2019), qui résument les tweets de manière chronologique en utilisant des graphes pour le traitement et la visualisation.

1.2 *Approches de mises à jour*

L'un des inconvénients des approches multi-document et chronologique est que tout le contenu pertinent en entrée doit être disponible à l'avance, alors que certains cas d'utilisation nécessitent le traitement de flux d'informations.

La tâche de mise à jour de résumé a été introduite pour la première fois par Dang et Owczarzak (2008). Cette tâche vise à produire de manière incrémentale deux résumés courts pour un même sujet. Dans une première phase, un ensemble de 10 documents doit être résumé. Ensuite, 10 autres documents sont ajoutés et doivent être résumés en fonction du premier résumé. Tous les documents de ces deux ensembles sont considérés comme appartenant au même sujet, et chaque résumé a une longueur maximale de 100 mots. Comme expliqué dans (McCreadie *et al.*, 2018), les approches de mise à jour de résumé se concentrent généralement sur le résumé

multi-document et appliquent une étape de suppression de la redondance (Dang et Owczarzak, 2008; Wang et Li, 2010; Yan *et al.*, 2011).

1.3 *Approches de résumés temporels*

Le résumé temporel a été abordé dans le cadre de la campagne d'évaluation TREC Real-Time Summarization (RTS) et vise à renvoyer des tweets nouveaux et pertinents aux utilisateurs, avec en entrée le flux entier de tweets. Dans la partie A de la campagne, les tweets pertinents doivent être renvoyés aussi rapidement que possible après leur publication. La partie B vise à retourner à la fin de chaque journée un résumé composé d'au plus 100 tweets. Le campagne a été arrêtée en 2018 après huit ans mais des améliorations restent possibles et nécessaires (Sequiera *et al.*, 2018). Les meilleurs scores pour la partie B ont été obtenus par Chellal et Boughanem (2018) et Fernández *et al.* (2018). Ils ont utilisé le texte ainsi que des caractéristiques spécifiques au tweet et à l'utilisateur comme les hashtags, les URLs ou les followers, pour apprendre des filtres de pertinence et de redondance. Le récent *survey* (Ma *et al.*, 2022) sur le résumé multi-document utilisant des techniques d'apprentissage profond a mis en évidence les travaux de Tan *et al.* (2017) et Yang *et al.* (2020) pour le résumé de flux. Ces travaux visent à décider si un tweet est pertinent à retourner en fonction d'une requête utilisateur en utilisant l'apprentissage par renforcement. Ils ont également été évalués sur les collections TREC RTS 2016 et 2017. Bien que le résumé temporel soit considéré dans la littérature comme une tâche de résumé, il faut garder à l'esprit que l'un des problèmes les plus importants de cette tâche est le filtrage des flux de tweets.

2 Résumé de flux tweets

Outre les cas très particuliers du résumé chronologique et du résumé temporel qui traitent principalement des tweets, la littérature sur le résumé de tweets traite principalement de résumé multi-document appliqué à la nature des tweets. Même si les flux de tweets sont parfois considérés, les résumés sont majoritairement construits à partir de zéro pour chaque fenêtre de temps considérée.

Sharifi *et al.* (2010) introduisent l'algorithme Phrase Reinforcement. Cet algorithme génère des résumés de tweets d'une seule phrase. Les tweets considérés doivent contenir une locution donnée *a priori*. Un graphe est créé à partir des tweets ainsi considérés. Le graphe est généré de telle sorte qu'il est centré sur un nœud racine commun représentant la locution, et représente les séquences communes de mots qui apparaissent avant et après la locution donnée. Les séquences sont pondérées par leur

nombre d'apparition dans les tweets. Ainsi, le résumé généré est la séquence qui maximise la pondération d'apparition. Nichols *et al.* (2012) utilisent l'algorithme Phrase Reinforcement. Le modèle détecte dans un premier temps les moments importants à résumer à partir du changement du volume de données par minute. L'algorithme de Phrase Reinforcement est ensuite utilisé, mais au lieu de ne retourner qu'une phrase, un score est calculé pour chaque tweet à partir des pondérations du graphe. Les phrases les mieux notées et qui ne partagent aucun terme (mots-vides exclus) sont retournées pour le résumé. Ces méthodes sont limitées à la locution donnée *a priori* et ne prennent pas en compte les tweets qui emploient d'autres termes sémantiquement équivalents comme des synonymes.

Inouye et Kalita (2011) proposent un Tf-Idf hybride pour le résumé de tweets afin de prendre en compte la longueur des documents. Ce Tf-Idf hybride considère chaque phrase comme un document, et, lors du calcul de la fréquence des termes, le document considéré est l'ensemble des tweets. Le Tf-Idf est également utilisé par Alsaedi *et al.* (2016) mais la notion de temporalité est utilisée. Ils proposent trois approches de résumé d'ensembles de tweets : (i) un Tf-Idf temporel calculé sur deux plages de temps au lieu d'une, (ii) une approche de vote basée sur les retweets entre deux plages de temps également, et (iii) une approche par regroupement pour laquelle la notion de temporalité est ajoutée : le tweet le plus représentatif d'un groupe durant la plus longue période de temps est utilisé pour le résumé. Pour les méthodes proposées, les flux de tweets sont manuellement regroupés par sous-évènements (4 par évènements), ce qui limite l'utilisation de la méthode pour des gros volumes de données et pour un résumé généré en « temps-réel ». L'approche Sumblr (Shou *et al.*, 2013) a été proposée pour résumer incrémentalement des flux de tweets. Sumblr regroupe les tweets continuellement et sélectionne les tweets *centroids* des groupes générés pour le résumé. Sumblr peut aussi produire des résumés historiques pour générer des résumés chronologiques. Sumblr a montré qu'il génère des résumés de meilleure qualité que ceux fournis par la méthode proposée par Inouye et Kalita (2011). Parmi les approches les plus récentes pour le résumé de tweet, les approches extractives nommées COWTS (Rudra *et al.*, 2015), SEMCOWTS (Rudra *et al.*, 2018a), et SCC (Rudra *et al.*, 2018b) sont basées sur l'optimisation linéaire en nombres entiers (ILP, *Integer Linear Programming*) pour maximiser la couverture des mots-clés dans le résumé final. Les auteurs de COWTS, SEMCOWTS et SCC ont plus récemment étendu leur approche au résumé abstractif (Rudra *et al.*, 2019). Ces méthodes se concentrent sur des classes de résumé prédéfinies (« Personnes disparues, piégées ou retrouvées », « Infrastructure et services publics », « Services de dons ou de bénévolat », « Refuge et ravitaillement », « Attention et conseils », « Personnes déplacées et évacuations »). Les approches COWTS et SEMCOWTS ont été comparées à Sumblr (Shou *et al.*, 2013) et ont montré des résultats largement supérieurs.

Saini *et al.* (2019) ont proposé une approche multi-objectif utilisant un algorithme génétique et la *Word Mover Distance* (Werner et Laber, 2020) pour générer des résumés extractifs. (Wang et Zhang, 2017) proposent une approche supervisée pour générer des résumés. Le modèle neuronal apprend à détecter, regrouper et résumer conjointement les évènements. Cependant, comme ces approches calculent les similarités entre toutes les paires de tweet du corpus, elles deviennent inutilisables lorsque la collection est trop volumineuse. L'utilisation de modèles de langue pré-entraînés est mentionnée dans (Li et Zhang, 2020) pour le résumé abstractif et récemment dans (Li et Zhang, 2021) pour le résumé extractif. Ce dernier utilise BERT pour encoder les tweets, ainsi qu'un graphe de relations de tweet comme entrée d'un réseau convolutif pour graphes. Le réseau convolutif pour graphes a pour but de générer des caractéristiques cachées de tweet pour l'estimation de la pertinence du tweet. Une collection non accessible au public a été utilisée pour évaluer des résumés répondant à des questions de type qui-que-où-quand (who-what-where-when). L'approche (Li et Zhang, 2021) utilise un graphe de relations de tweets qui calcule la similarité entre toutes les paires de tweet du corpus. De même que (Wang et Zhang, 2017) et (Saini *et al.*, 2019), elle devient inutilisable lorsque la collection est trop grande.

3 Discussion

Le tableau 4.1 résume les différentes approches que nous avons mentionnées dans cette section et la section précédente.

Dans ce manuscrit, nous utiliserons l'expression de « résumé incrémental » pour désigner le résumé de mise à jour. Nous pensons que cette expression traduit mieux l'idée de résumé incrémental, c'est-à-dire de phrases ajoutées à un résumé original, tandis que la première peut être interprétée à tort comme une simple mise à jour d'un contenu déjà existant sans information supplémentaire (par exemple, la mise à jour du nombre de personnes disparues lors d'une catastrophe naturelle). Il est intéressant de noter que les approches classées dans le tableau 4.1 ont étendu la définition de la tâche de résumé de mise à jour proposée dans (McCreadie *et al.*, 2018). Alors que les auteurs indiquent explicitement que le résumé de mise à jour produit des résumés de longueur fixe à chaque fenêtre de temps, nous avons également classé dans cette ligne de recherche certaines approches produisant des résumés de longueur non fixe.

Nous avons vu dans le chapitre précédent que les modèles de langue neuronaux sont aujourd'hui l'état de l'art du résumé automatique. Dans ce chapitre, nous remarquons que cet engouement ne s'est pas appliqué au résumé avec prise en compte de la temporalité, et notamment pour résumer les flux de tweets. Les principales limites que nous avons identifié à la mise en place de ces outils pour ces problèmes

sont liés à : (i) la difficulté de créer des jeux d'apprentissage adéquats pour ces modèles, (ii) la taille de l'entrée à résumer, qui ne permet pas d'appliquer directement ces outils comme dans le cas de résumé multi-document classique. C'est à ces limites que nos contributions, détaillées dans les chapitres suivants, tentent de répondre.

TABLEAU 4.1 – Classification de l'état de l'art des approches selon différentes facettes. Une première ligne d'analyse concerne l'entrée des approches, décrite en termes de : (i) mono-document (S), C (Concaténation, c'est-à-dire la concaténation de multiples documents en un seul), ou M (Multi-document), (ii) nature (Tweets / News / Articles scientifiques / Wikihow / Emails), (iii) qu'elles traitent de flux ou non, (iv) qu'elle soit sur un sujet ou évènement particulier ou non (On-topic). Deuxièmement, les approches sont analysées selon leur sortie : (i) abstractive ou extractive, (ii) avec une taille fixé ou non. Troisièmement, nous indiquons si les approches sont basées sur une architecture neuronale. Finalement, les approches sont comparées en termes de reproductibilité (code et jeu de données). Dans la dernière colonne, les approches : (I., 2011),(A., 2016),(F., 2014),(H., 2005),(R., 2004),(E., 2004),(M., 2004),(B., 2013),(S., 2010),(D., 2002),(C., 2010),(A., 2013),(S-G., 2020),(P., 2018),(S., 2009),(P., 2019) correspondent respectivement à : (Inouye et Kalita, 2011),(Alsaedi *et al.*, 2016),(Ferreira *et al.*, 2014),(Harabagiu et Lacatusu, 2005),(Radev *et al.*, 2004),(Erkan et Radev, 2004),(Mihalcea et Tarau, 2004),(Baralis *et al.*, 2013),(Sharifi *et al.*, 2010),(Daumé et Marcu, 2002),(Clarke et Lapata, 2010),(Alguliev *et al.*, 2013),(Sánchez-Gómez *et al.*, 2020),(Peyrard et Gurevych, 2018),(Suanmali *et al.*, 2009),(Patel *et al.*, 2019)

	Entrée				Sortie		Architecture	Reproductibilité		Approches
	S/C/M	Nature	Flux	On-topic	Type	Taille	neuronale	Code	Jeu de données	
Mono-document	S/C	News		√	Abs. / Ext	Non Fixé	√	√	√	(Liu et al., 2019; Song et al., 2020)
	S	News		√	Abs.	Non Fixé	√	√	√	(Qi et al., 2020)
	S	News / Scien. / Wiki		√	Abs.	Non Fixé	√	√	√	(Dou et al., 2021)
	S/C	News / Scien. / Emails		√	Abs.	Non Fixé	√	√	√	(Zhang et al., 2020)
	S/C	News / Scien. / Wiki		√	Ext.	Non Fixé	√	√	√	(Zhong et al., 2020)
	S	News		√	Ext.	Non Fixé				(Fattah et al., 2009)
	S	News		√	Ext.	Fixé			√	(D., 2002; C., 2010; S., 2009)
	S	News		√	Ext.	Fixé	√		√	(Cheng et al., 2016; Nallapati et al., 2017)
Multi-document	M	News		√	Ext.	Fixé	√	√	√	(Fabbri et al., 2019)
	M	Tweets	√	√	Ext.	Fixé		√	√	(Rudra et al., 2015; 2018a; 2018b)
	M	Tweets	√	√	Abs/ Ext.	Fixé		√	√	(Rudra et al., 2019)
	M	Tweets		√	Ext.	Fixé		√	√	(Saini et al., 2019)
	M	Tweets	√	√	Abs.	Non-Fixé	√			(Li et al., 2020)
	M	Tweets		√	Ext.	Fixé / Non Fixé	√			(Li et al., 2021)
	M	Tweets		√	Ext.	Fixé	√	√		(Wang et al., 2017)
	S/M	News		√	Ext.	Fixé				(Ko et al., 2008)
	M	News		√	Ext.	Fixé	√		√	(Cao et al., 2015; Ren et al., 2017)
	M	News		√	Ext.	Fixé			√	(F., 2014; H., 2005; A., 2013; S-G., 2020; P., 2018)
	M	News		√	Ext.	Fixé		√	√	(R., 2004; E., 2004; M., 2004; B., 2013; P., 2019)
	M	Tweets		√	Ext.	Fixé				(I., 2011; A., 2016; S., 2010)
	M	News / Scien. / Wiki		√	Abs.	Fixé	√	√	√	(Xiao et al., 2022)
Résumé Temporel	M	Tweets	√		Ext.	Non Fixé			√	(Chellal et al., 2018)
	M	Tweets	√		Ext.	Fixé / Non Fixé			√	(Fernandez et al., 2018)
Résumé Chronologique	M	News		√	Ext.	Fixé				(Tran et al., 2015)
	M	News		√	Ext.	Non Fixé		√	√	(Martschat et al., 2018)
	M	Tweets		√	Abs.	Non Fixé				(Ansah et al., 2019)
	M	Tweets	√	√	Ext.	Fixé				(Zubiaga et al., 2012; Nichols et al., 2012)
Mise à jour de résumé	M	News	√	√	Ext.	Non Fixé			√	(Dang et al., 2008)
	M	News	√	√	Ext.	Non Fixé				(Yan et al., 2011)
	M	News	√	√	Ext.	Fixé			√	(Wang et al., 2010)
	M	Tweets	√	√	Ext.	Non Fixé				(Shou et al., 2013)

Partie II

CONTRIBUTIONS

5

CORPUS D'ÉVALUATION

Sommaire

Introduction

Dans les précédents chapitres, nous avons présenté les médias sociaux. Nous en avons donné une définition, exploré les différents médias sociaux existants et les enjeux d'élaborer une information synthétisée de ces sources d'information. Nous expliquons pourquoi nous nous sommes concentrés sur le média social Twitter et les données textuelles en particulier. Nous avons également introduits les différents outils de l'état de l'art du résumé automatique, et notamment du résumé automatique de flux de tweets. Nous allons nous pencher dans ce chapitre sur les premières contributions apportées par cette thèse qui portent sur la proposition de jeux d'évaluation pour le résumé automatique de flux de tweets. Nous verrons le manque de jeu de données de référence pour l'évaluation du résumé automatique de flux de tweets, nous proposerons une méthode de création de jeux de données qui requiert peu d'efforts humains, et nous détaillerons la création d'un jeu de données réutilisable lors de travaux futurs.

1 Contexte et motivations

Dans le chapitre 3, nous avons détaillé les différentes mesures d'évaluation de résumé automatique, qu'elles soient automatiques ou manuelles (section 4.1.2). Afin de permettre la comparaison des modèles, des collections de référence ont été créées et mises à disposition de la communauté. Pour les méthodes de résumé automatique portant sur des documents plus classiques tels que des journaux, on peut citer par exemple les jeux de données Gigaword (Graff *et al.*, 2003; Rush *et al.*, 2015), CNN/DailyMail (Hermann *et al.*, 2015; See *et al.*, 2017), ou (Narayan *et al.*, 2018a) pour du résumé mono-document, ou encore les jeux de données construits lors des conférences dédiées DUC (Document Understanding Conferences) entre 2001 et 2007 (Dang, 2005) [1], et TAC (Text Analysis Conference) entre 2008 et 2015 (Dang et Owczarzak, 2008) [2] pour du résumé multi-document. Il est même possible de récupérer ces jeux de données très facilement en quelques lignes de code, afin d'évaluer de nouvelles méthodes [3], et les comparer à l'état de l'art. Cependant, pour le résumé automatique de flux de tweets, il n'existe pas de collection de référence. D'une part, la diffusion de tweets est restreinte par Twitter [4], et d'autre part, il est fastidieux de créer des résumés de référence. Les rares collections existantes qui proposent de

1. `https://www-nlpir.nist.gov/projects/duc/`, dernière consultation le 04/08/2022
2. `https://tac.nist.gov/`, dernière consultation le 04/08/2022
3. `https://www.tensorflow.org/datasets/catalog/overview#summarization`, dernière consultation le 04/08/2022
4. `https://developer.twitter.com/en/developer-terms/more-on-restricted-use-cases`, dernière consultation le 04/08/2022

diffuser leurs résumés de référence, sont soit aujourd'hui inaccessibles (Nguyen *et al.*, 2018; Shou *et al.*, 2013) [5] [6], soit demandent de contacter les auteurs et obligent donc à être dépendants d'une réponse de leur part (Rudra *et al.*, 2019). Il faut souvent récupérer les tweets à partir d'une liste d'identifiants de tweets, ce qui d'ailleurs entraîne que certains tweets peuvent être irrécupérables car supprimés entre-temps. Actuellement, les méthodes de résumé automatique de tweets établissent un nouveau jeu de données pour chaque nouvelle méthode proposée (Rudra *et al.*, 2019; Olariu, 2014). Pour proposer une nouvelle méthode, il est donc commun d'établir un nouveau jeu de données, puis d'évaluer (ou implémenter si besoin), les méthodes de l'état de l'art.

Parallèlement, depuis 1992, a lieu chaque année la conférence Text REtrieval Conference (TREC) [7]. Elle est co-sponsorisée par le National Institute of Standards and Technology (NIST) et le Département de la Défense des États-Unis. Cette conférence regroupe différentes tâches en Recherche d'Information, faisant l'objet de campagnes d'évaluation. Un sujet est défini, les participants apportent leurs approches au sujet, et les approches sont ensuite évaluées sur un jeu de test défini par les organisateurs de la tâche. Les soumissions des participants sont évaluées, selon des mesures d'évaluation parfois données en amont aux participants, mais pas toujours. Les données proposées pour les tâches TREC sont généralement laissées à disposition pendant des années à la communauté. Par exemple, les données de tâches TREC de 2006 sont toujours disponibles [8]. Justement, depuis 2018, TREC propose la campagne Incident Streams pour laquelle est fourni un jeu de données composé de tweets et d'annotations (McCreadie *et al.*, 2019). Ces tweets et annotations sont stockés sur leurs propres serveurs [9], ce qui permet notamment d'éliminer le problème des suppressions de tweets entre plusieurs travaux. Il est important de noter que cette tâche n'est pas destinée à faire du résumé de tweets au départ.

Dans les prochaines sections, nous présentons dans un premier temps les jeux de données existants pour le résumé automatique de flux de tweets. Dans un deuxième temps, nous présentons une méthodologie de construction de jeu de données dans laquelle les résumés de référence sont extraits du portail d'actualités de Wikipedia, et nous l'appliquons ensuite pour construire la collection TES 2012-2016. Dans un troisième temps, nous étudions la possibilité d'utiliser le jeu de données de la campagne Incident Streams pour le résumé automatique de flux de tweets, dans le but d'amener à la communauté un jeu de données référence pour le résumé automatique de flux de tweets.

5. `https://goo.gl/kXBof9`, inaccessible, dernière consultation le 04/08/2022
6. `http://db.zju.edu.cn/s/sumblr/`, inaccessible, dernière consultation le 04/08/2022
7. `https://trec.nist.gov/`, dernière consultation le 04/08/2022
8. `https://trec.nist.gov/data/blog06.html`, dernière consultation le 04/08/2022
9. `http://dcs.gla.ac.uk/~richardm/TREC_IS/2020/data.html`, dernière consultation le 04/08/2022

1.1 *Jeux de données existants pour le résumé automatique de flux de tweets*

Lorsque l'on considère des jeux de données pour le résumé de flux de tweets, deux considérations principales doivent être prises en compte : (i) les résumés de référence doivent être distribués, et (ii) les tweets doivent être toujours disponibles.

La constitution d'un résumé de référence est particulièrement coûteuse et constitue souvent le nœud du problème dans le processus de constitution des collections. Certains jeux de données publient des résumés de références accessibles au public (Dutta *et al.*, 2018; Rudra *et al.*, 2018b), tandis que d'autres ne le font pas (Duan *et al.*, 2012; Liu *et al.*, 2012; Sharifi *et al.*, 2014). Certaines évaluations n'utilisent pas de résumés de référence et se contentent d'une comparaison manuelle des résumés (Harabagiu et Hickl, 2011; Olariu, 2014).

La disponibilité des tweets est principalement liée à la politique de distribution de Twitter. Twitter impose des règles pour la diffusion des tweets [10] : les tweets entiers ne peuvent pas être distribués, et les collections doivent être diffusées en utilisant uniquement les identifiants des tweets. Les tweets doivent ensuite être récupérés par l'API Twitter à partir de leurs identifiants. Cependant, un inconvénient majeur de ce processus est qu'il est souvent presque impossible de récupérer entièrement les jeux de données tels qu'ils ont été diffusés. En effet, certains tweets peuvent être supprimés et ne sont donc plus disponibles. Un exemple représentatif est le jeu de données présenté dans (Rudra *et al.*, 2018b), pour lequel seuls 63 % des tweets restent actuellement accessibles. Plus précisément, nous avons pu récupérer 79 870 tweets sur les 132 097 identifiants de tweets fournis. De plus, sur les 558 tweets utilisés comme résumés de référence, nous avons récupéré 11 tweets (2 %) en utilisant la comparaison stricte de textes et 115 tweets (21 %) en utilisant le pré-traitement de texte (normalisation de texte, suppression d'url, *stemming*, ...) pour comparer des tweets presque similaires. En d'autres termes, la majorité des tweets des résumés de référence ne sont pas présents dans le flux d'entrée : il n'est donc pas possible de reproduire les résultats publiés par les auteurs, et cela pose la question de l'adéquation des résumés fournis pour le flux d'entrée.

En plus des deux problèmes mentionnés ci-dessus, nous nous focalisons sur deux problèmes spécifiques à nos travaux :

— les résumés de référence doivent être incrémentaux, c'est-à-dire qu'ils doivent être disponibles par fenêtre de temps,
— chaque flux de tweet doit être sur un sujet particulier.

10. `https://developer.twitter.com/en/developer-terms/more-on-restricted-use-cases`, dernière consultation le 04/08/2022

La campagne d'évaluation TREC RTS (Real Time Summarization) peut sembler appropriée à première vue. Cependant, alors que les résumés de référence sont incrémentaux, les flux utilisés dans la campagne d'évaluation ne sont pas sur un sujet particulier, se rapportant ainsi au résumé *temporel* tel que défini dans (McCreadie *et al.*, 2018) (voir section 1.3 du chapitre 4), plutôt qu'au résumé de mise à jour. De plus, en inspectant de plus près les résumés de référence, on constate que seuls 6 tweets par jour sont pertinents en moyenne, ce qui ne mérite pas d'être résumé [11]. TREC RTS est donc plus approprié pour évaluer les approches de filtrage en temps réel que celles de résumé incrémental.

Parmi les autres jeux de données fournissant des résumés de référence pour différentes fenêtres de temps, on peut citer (Rudra *et al.*, 2018b) [12]. Cette méthode souffre toutefois du problème de suppression des tweets évoqué plus haut dans cette section.

1.2 *Discussion*

Parmi les jeux de données existants pour le résumé de flux de tweets, aucun ne sert de référence. Ceci n'est cependant pas surprenant au vu des restrictions sur la distribution de ces jeux de données. Cela a pour effet que, pour chaque nouveau modèle proposé, un nouveau jeu de données est construit. La constitution de tels ensembles de données est toutefois fastidieuse et demande beaucoup d'efforts. Il existe cependant de rares exceptions. Le jeu de données proposé dans (Dutta *et al.*, 2018), et réutilisé dans (Saini *et al.*, 2019) n'a étonnamment (au regard des restrictions de Twitter) pas été diffusé en utilisant les identifiants des tweets mais en listant le texte des tweets dans des fichiers. Ce jeu de données n'est cependant pas adapté à notre problème, à savoir le résumé incrémental, car aucune information temporelle n'est donnée pour chaque tweet.

Le tableau 5.1 résume les jeux de données existants pour le résumé de flux de tweets en tenant compte de plusieurs dimensions : taille du flux de tweets, nombre de sujets par évènement, longueur des résumés de référence, disponibilité, si l'ensemble de données est affecté par le problème de suppression des tweets, s'il est sur un sujet particulier et s'il a des résumés incrémentaux pour référence.

Seule la collection (Rudra *et al.*, 2018a) semble convenir au problème abordé dans le présent travail. Un premier objectif a donc été le suivant :

11. Ni le jeu de données, ni les métriques utilisés dans TREC RTS ne correspondent au problème attaqué dans ce travail. Nous ne reviendrons donc pas sur TREC RTS dans ce manuscrit.

12. Les résumés de référence ont été aimablement fournis par les auteurs lorsque nous les leur avons demandés.

Objectif de recherche : Accroître le cadre d'évaluation des modèles de résumés automatique de flux de tweets.

Pour répondre à cet objectif, nous proposons les contributions suivantes :

Méthodologie pour répondre à l'objectif de recherche :
— une méthodologie pour créer des jeux de données avec peu d'efforts, que nous avons appliqué pour créer le nouveau jeu de données TES 2012-2016,
— un nouveau jeu de données qui n'est pas affecté par la suppression de tweets.

TABLEAU 5.1 – Jeux de données existants pour le résumé de tweets. Les deux dernières lignes concernent les jeux de données que nous proposons dans cette thèse. Pour les jeux de données dont les résumés de référence sont incrémentaux, la taille des résumés de référence reportée est celle à la fin de l'évènement.

Jeux de données	Nombre de tweets	Nombre de sujets	Taille des résumés de référence	Distribué	Impacté par la suppression de tweets	Sur un sujet particulier	Résumé de référence incrémental
Harabagiu et al.	> 2 500/sujet	25	None		√	√	
Duan et al.	900 000	84	10 tweets		√	√	
Liu et al.	1 000 max./sujet	100	5 % de la taille de l'entrée		√	√	
Olariu	3 400 000	64	None	√	√		
Sharifi et al.	2 500	25	4 tweets		√	√	
TREC RTS 2016	49 011 128	56	14 tweets en moyenne	√	√		√
TREC RTS 2017	48 930 825	96	30 tweets en moyenne	√	√		√
Rudra et al.	130 000	3	200 words	√	√	√	√
Dutta et al.	7 000	4	27 tweets en moyenne	√		√	
ISSumSet	35,000	26	45 tweets en moyenne	√		√	√
TES 2012-2016	82 315 567	28	265 mots en moyenne	√	√	√	√

2 TES 2012-2016

Dans cette section, nous proposons une méthodologie pour créer de nouveaux jeux de données pour le résumé incrémental de flux de tweets. Cette méthodologie nécessite peu d'efforts humains et peut être utilisée pour construire de grandes collections de tests. Ensuite, nous présentons le jeu de données TES 2012-2016 construit à l'aide de la méthodologie susmentionnée et utilisée pour évaluer notre approche de résumé incrémental.

2.1 *Méthodologie de construction*

Pour créer un jeu de données pour le résumé de flux de tweets, nous proposons d'utiliser le portail d'actualités de Wikipedia (Wikipedia Current Events Portal -

WCEP) [13], dont les pages associées contiennent des nouvelles répertoriées quotidiennement avec un lien vers des articles de référence, ainsi qu'un ensemble d'évènements et de sous-évènements couverts par le WCEP. Nous proposons les étapes suivantes :

- Tout d'abord, récupérer les tweets concernant un évènement. Cela peut être fait à partir des identifiants des tweets associés à l'évènement apparaissant dans des collections existantes, ou en construisant une nouvelle collection en interrogeant l'API de Twitter avec des mots-clés et des hashtags liés à l'évènement pendant une période prévue couverte par le WCEP.
- Deuxièmement, construire le résumé de référence en utilisant le WCEP, car il rapporte sommairement les évènements quotidiens dans le monde entier. Pour chaque jour de la période couverte par les tweets récupérés à l'étape précédente, mettre en correspondance le résumé de référence avec les sous-évènements associés listés dans le portail. Deux cas particuliers doivent être détaillés :
  - (i) au début de l'évènement, il peut arriver que des tweets soient trouvés alors qu'aucun évènement n'a encore été signalé sur le WCEP. Dans ce cas, le résumé de référence pour la journée est simplement considéré comme vide.
  - (ii) au cours de l'évènement, il peut aussi arriver que des tweets soient trouvés alors qu'aucun sous-évènement n'est signalé dans le WCEP. Dans ce cas, le résumé de référence pour la journée est celui construit pour le sous-évènement précédent.
- Troisièmement, faire correspondre les horodatages de création des tweets avec l'heure locale du lieu de l'évènement. En effet, tous les horodatages des tweets sont dans le fuseau horaire UTC mais le portail Wikipedia Current Events rapporte les évènements en utilisant l'heure locale de l'évènement, qui n'est pas nécessairement dans le fuseau horaire UTC.

Cette méthodologie s'inspire du travail présenté dans (Ghalandari *et al.*, 2020). Dans ce travail, les articles de références liés aux sous-évènements du WCEP sont utilisés comme documents d'entrée à résumer, alors que dans notre cas ce sont les flux de tweets qui sont considérés. Un exemple de création de résumés de référence pour l'évènement *« Sismo Ecuador 2016 »* est donné dans la figure 5.1.

On peut trouver certaines limites à ce processus :

- (i) il ne peut pas être utilisé pour des évènements non présents dans le WCEP. Ceci est contrebalancé par la non-nécessité d'obtenir tous les évènements qui se produisent dans le monde mais juste assez pour évaluer les modèles (ou pour les entraîner).

13. `https://en.wikipedia.org/wiki/Portal:Current_events`, dernière consultation le 04/08/2022

— (ii) il donne une vision journalistique de l'évènement. Dans ce travail, cette limite ne doit pas être considérée comme un biais puisque nous visons à résumer des évènements plutôt qu'à résumer des conversations ou à donner un aperçu des tendances.

2.2 *Collection TES 2012-2016*

Dans cette section nous présentons le jeu de données TES 2012-2016, créé en suivant la méthodologie susmentionnée. Les tweets ont été récupérés à partir de l'ensemble des identifiants fournis dans la collection *Twitter events 2012-2016* (Zubiaga, 2018). Ce jeu de données est composé d'environ 150 millions d'identifiants de tweets explorés pour 30 évènements entre 2012 et 2016. Ces tweets ont été *crawlés* (récupérés) en fonction de différents ensembles de hashtags et de mots-clés liés aux évènements considérés.

Pour chaque évènement, nous avons extrait manuellement les résumés existants associés dans le portail Wikipedia Current Events pour la période couverte par les tweets. Comme les résumés sont présentés quotidiennement dans le WCEP, l'incrément de temps pour chaque résumé est fixé à un jour.

Comme indiqué dans la méthodologie (voir section 2.1) et compte tenu des règles d'édition du WCEP, nous avons dû faire correspondre l'horodatage de création des tweets (fuseau horaire UTC [14]) avec l'heure locale du lieu de l'évènement.

Quelques statistiques sur la collection TES 2012-2016 sont disponibles dans le tableau 5.2. Parmi les 30 évènements, 2 évènements, le *sxsw festival* qui a eu lieu en 2012 et les *st patricks day* 2014 celebrations, ne sont pas rapportés dans le WCEP. Nous les avons donc exclus dans ce travail. Pour les autres évènements, nous avons récupéré entre 1 et 81 sous-évènements par évènement, pour une moyenne de 12 sous-évènements par évènement. Pour certaines fenêtres de temps, plusieurs sous-événements peuvent se produire ou un sous-événement peut être particulièrement riche en termes de faits. Le résumé de référence de ces fenêtres de temps est par conséquent composé de plusieurs phrases. Un tel cas est représenté en figure 5.1 pour les fenêtres de temps correspondant aux dates 19/04/2016 et 20/04/2016. Nous avons pu récupérer un total de 82 315 567 tweets à partir de la collection originale d'évènements Twitter 2012-2016 (seulement 58 % des identifiants publiés, avec un minimum de 49 % et un maximum de 72 % pour un seul évènement). Le nombre minimum de tweets récupérés pour un évènement est de 125 093 (*élection mexicaine 2012*) tandis que le maximum est de 16 617 011 (*attaque parisienne 2015*). En moyenne, le jeu de données TES 2012-2016 est composé de 2 939 842 tweets par évènement.

14. `https://developer.twitter.com/en/docs/twitter-api/v1/data-dictionary/object-model/tweet`, dernière consultation le 04/08/2022

Sous-évènements du WCEP :
2016-04-17 : The death toll from Saturday night's earthquake in Ecuador rises to 262 with more than 2,500 people injured.
2016-04-18 : Aid starts to flow in after an earthquake kills over 270 people in Ecuador.
2016-04-19 : The death toll from Saturday's earthquake has risen to at least 480 with 1,700 missing. Another 2,500 have been injured. President Rafael Correa states it is the worst disaster in Ecuador in seven decades, and the reconstruction will have a 'huge economic impact' on the country.
2016-04-20 : A magnitude-6.1 aftershock has struck off the coast of Ecuador at 3 :33 a.m. local time, the US Geological Survey says, in the same area as the massive earthquake on Saturday.
2016-04-20 : President Rafael Correa announces a sales tax increase, and a one-time levy on millionaires as the country deals with the enormous damage from this disaster.
2016-04-20 : The death toll rises to 570 with 163 people listed as missing. Those made homeless climbs to over 23,500.
2016-04-25 : The United Nations World Food Programme announces it is stepping up assistance to Ecuador's most vulnerable areas following an earthquake that killed over 650 people.

Résumés de référence associés :
2016-04-17 : The death toll from Saturday night's earthquake in Ecuador rises to 262 with more than 2,500 people injured.

2016-04-18 : The death toll from Saturday night's earthquake in Ecuador rises to 262 with more than 2,500 people injured.
Aid starts to flow in after an earthquake kills over 270 people in Ecuador.

2016-04-19 : The death toll from Saturday night's earthquake in Ecuador rises to 262 with more than 2,500 people injured.
Aid starts to flow in after an earthquake kills over 270 people in Ecuador.
The death toll from Saturday's earthquake has risen to at least 480 with 1,700 missing. Another 2,500 have been injured. President Rafael Correa states it is the worst disaster in Ecuador in seven decades, and the reconstruction will have a 'huge economic impact' on the country.

2016-04-20 : The death toll from Saturday night's earthquake in Ecuador rises to 262 with more than 2,500 people injured.
Aid starts to flow in after an earthquake kills over 270 people in Ecuador.
The death toll from Saturday's earthquake has risen to at least 480 with 1,700 missing. Another 2,500 have been injured. President Rafael Correa states it is the worst disaster in Ecuador in seven decades, and the reconstruction will have a 'huge economic impact' on the country.
A magnitude-6.1 aftershock has struck off the coast of Ecuador at 3 :33 a.m. local time, the US Geological Survey says, in the same area as the massive earthquake on Saturday.
President Rafael Correa announces a sales tax increase, and a one-time levy on millionaires as the country deals with the enormous damage from this disaster.
The death toll rises to 570 with 163 people listed as missing. Those made homeless climbs to over 23,500.

2016-04-21 : Identique à **2016-04-20**
2016-04-22 : Identique à **2016-04-21**
2016-04-23 : Identique à **2016-04-22**
2016-04-24 : Identique à **2016-04-23**

2016-04-25 : The death toll from Saturday night's earthquake in Ecuador rises to 262 with more than 2,500 people injured.
Aid starts to flow in after an earthquake kills over 270 people in Ecuador.
The death toll from Saturday's earthquake has risen to at least 480 with 1,700 missing. Another 2,500 have been injured. President Rafael Correa states it is the worst disaster in Ecuador in seven decades, and the reconstruction will have a 'huge economic impact' on the country.
A magnitude-6.1 aftershock has struck off the coast of Ecuador at 3 :33 a.m. local time, the US Geological Survey says, in the same area as the massive earthquake on Saturday.
President Rafael Correa announces a sales tax increase, and a one-time levy on millionaires as the country deals with the enormous damage from this disaster.
The death toll rises to 570 with 163 people listed as missing. Those made homeless climbs to over 23,500.
The United Nations World Food Programme announces it is stepping up assistance to Ecuador's most vulnerable areas following an earthquake that killed over 650 people.

FIGURE 5.1 – Exemple de création d'un gold standard du WCEP concernant l'évènement « Sismo Ecuador 2016 » pour les jours du 2016-04-17 au 2016-04-25 (extrait).

TABLEAU 5.2 – Quelques statistiques sur la collection TES 2012-2016.

Évènement	Nombre d'ids	Nombre/% de tweets récupérés	Nombre de sous-évènements dans le WCEP
euro 2012	8 992 157	5 625 286 / 63	52
hurricane sandy 2012	14 914 566	8 399 189 / 56	9
mexican election 2012	191 788	125 093 / 65	1
obama romney 2012	10 146 517	5 033 962 / 50	8
superbowl 2012	1 659 475	1 027 944 / 62	2
us election 2012	1 740 258	1 001 114 / 58	8
boston marathon bombing 2013	3 430 387	1 738 635 / 51	4
ebola 2014	986 525	663 081 / 67	8
ferguson 2014	8 782 071	5 105 294 / 58	7
gaza under attack 2014	2 886 322	1 407 497 / 49	17
hongkong protests 2014	1 188 372	764 371 / 64	11
indyref 2014	1 524 166	952 495 / 62	5
ottawa shooting 2014	1 075 864	675 022 / 63	1
sydney siege 2014	2 157 879	1 246 560 / 58	3
typhoon hagupit 2014	264 626	161 666 / 61	4
charlie hebdo 2015	18 940 619	11 101 149 / 59	10
germanwings crash 2015	2 648 983	1 583 392 / 60	8
hurricane patricia 2015	1 151 220	578 559 / 50	1
nepal earthquake 2015	12 004 187	7 364 891 / 61	19
paris attacks 2015	29 821 274	16 617 011 / 56	32
refugees welcome 2015	1 743 153	1 064 537 / 61	81
brexit 2016	1 826 290	1 001 836 / 55	3
brussels airport explosion 2016	5 869 990	3 328 377 / 57	10
hijacked plane cyprus 2016	702 586	424 930 / 60	3
irish general election 2016	758 803	541 989 / 71	4
lahore blast 2016	1 149 253	685 922 / 60	2
panama papers 2016	5 044 379	3 368 653 / 67	19
sismo ecuador 2016	1 007 867	727 112 / 72	9
Total	142 609 577	82 315 567 / 58	341

3 ISSumSet

Dans cette section nous proposons un jeu de données, nommé ISSumSet, qui n'est pas soumis à la suppression de tweets, et donc réutilisable pour de futurs travaux. Ce jeu de données est créé à partir de la campagne d'évaluation TREC Incident

Streams. Dans un premier temps, nous présentons la campagne d'évaluation, puis nous détaillons les objectifs et les caractéristiques du jeu de données associés à la campagne d'évaluation. Dans un second temps nous présentons le jeu de données ISSumSet. Nous détaillons les motivations pour utiliser les données de la campagne d'évaluation TREC Incident Streams pour du résumé automatique de flux de tweets. Ensuite, nous analysons les résumés potentiels de ce jeu de données par rapport aux propriétés d'un bon résumé.

3.1 *Description de la campagne TREC Incident Streams*

La campagne TREC Incident Streams a eu lieu chaque année entre 2018 et 2021, soit 4 éditions. Nos travaux portent sur les éditions 2018 et 2019. Nous présentons dans cette section les objectifs de cette campagne d'évaluation, puis nous détaillons les caractéristiques des collections TREC Incident Streams.

3.1.1 *Objectifs*

L'objectif de la campagne est d'utiliser des informations sociales, dans ce cas Twitter, pour aider les services d'urgence lors de l'apparition d'incidents. En effet, des travaux ont montré que des informations utiles et importantes peuvent être récupérées à partir des médias sociaux lors d'incidents (Olteanu *et al.*, 2015). L'objectif de la campagne est double. Tout d'abord, un ensemble de tweets doivent être classés selon des catégories prédéfinies appelées types d'information. Les types d'information sont par exemple « Demande - Information recherchée », « Demande - Recherche et Sauvetage ». Ensuite, un score de priorité doit être associé pour chaque tweet afin de retourner une alerte si nécessaire. Une alerte est remontée si le score de priorité est supérieur à 0,7. Une attention particulière est donnée aux types d'information nommés « Demande - Bien ou Service », « Demande - Recherche et Sauvetage », « Appel à action - Déplacement de personnes », « Rapport - Menaces Émergentes », « Rapport - Nouveau Sous-Évènement », « Rapport - Service Disponible » qui sont considérés comme *actionnables*, ce qui signifie qu'ils doivent conduire à des actions de la part des services d'urgence. L'objectif de la campagne est illustré en figure 5.2.

3.1.2 *Caractéristiques des collections TREC Incident Streams*

Cette section présente les principales caractéristiques des collections des éditions 2018, 2019 partie A et 2019 partie B de la campagne d'évaluation TREC Incident

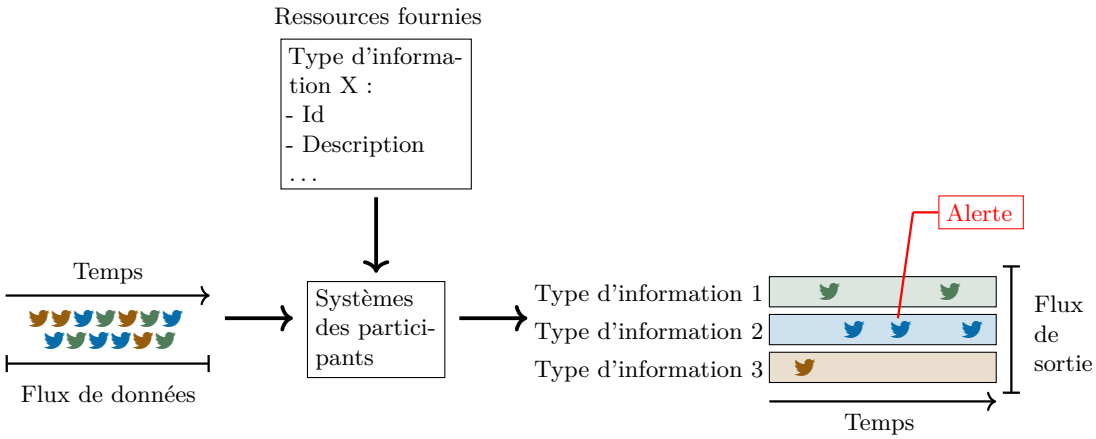


FIGURE 5.2 – Campagne Incident Streams, comme illustrée sur le site de la campagne (`http://dcs.gla.ac.uk/~richardm/TREC_IS/`, dernière consultation le 04/08/2022)

Streams. L'édition 2019 s'est déroulée en deux parties. La partie A a eu lieu entre mars et juin, tandis que la partie B a eu lieu entre juillet et octobre.

3.1.2.1 Tweets collectés

La collection a été construite incrémentalement en trois étapes. D'une part, en 2018, les tweets ont été collectés à partir de la source CrisisLexT26 (Olteanu *et al.*, 2015). D'autre part, plusieurs sources ont été utilisées pour collecter les tweets des parties A et B de l'édition 2019 : CrisisNLP Resource #2 (Imran *et al.*, 2013), CrisisNLP Resource #1 (Imran *et al.*, 2016), des tweets supplémentaires récupérés par les organisateurs de la campagne eux-mêmes via l'API [15] de Twitter, d'autres tweets collectés via le service GNIP [16], et finalement des tweets donnés par des groupes participants. Les évènements relatifs à ces tweets collectés réfèrent aux types d'incidents suivants : incendies, tremblements de terre, inondations, typhons/ouragans, fusillades et attaques à la bombe.

Concernant les sources CrisisLexT26 et CrisisNLP, les organisateurs ont utilisé les tweets labellisés pertinents pour l'évènement. Pour les évènements restants et les tweets correspondants collectés par les organisateurs, des regroupements de tweets ont été réalisés pour sélectionner un tweet par regroupement. Après cela, un filtrage additionnel par mots-clés a été réalisé et conduit à un ensemble réduit, varié, et vraisemblablement pertinent. Finalement, pour tous les évènements, les tweets identifiés

15. Une API (Application Programming Interface) sert à communiquer entre services web de manière normalisée.

16. `https://developer.twitter.com/en/products/twitter-api/enterprise`, dernière consultation le 04/08/2022

par le classifieur de langue de Twitter comme écrits dans un langage différent de l'anglais ont été supprimés.

3.1.2.2 Annotation des tweets

Pour le processus d'évaluation, les organisateurs ont créé une vérité terrain en accord avec l'ensemble des types d'information fournis pour les ensembles d'apprentissage et de test. Pour créer cette vérité terrain, une interface de labellisation a été développée par les organisateurs, illustrée en figure 5.3. Cette interface a pour but d'aider les évaluateurs à annoter les tweets collectés pour chaque évènement (voir section 3.1.2.1) avec :

— Types d'information : « Demande - Bien ou Service », « Demande - Recherche et Sauvetage », « Demande - Information recherchée », « Appel à action - Volontaires », « Appel à action - Donations », « Appel à action - Déplacement de personnes », « Rapport - Premières Observations », « Rapport - Observations Tierces », « Rapport - Météo », « Rapport - Menaces Émergentes », « Rapport - Partage Image ou Vidéo », « Rapport - Service Disponible », « Rapport - Faits », « Rapport - Officiel », « Rapport - Nettoyage », « Rapport - Hashtags », « Rapport - Nouvelles », « Rapport - Nouveau Sous-Évènement », « Rapport - Localisation », « Autre - Conseils », « Autre - Sentiments », « Autre - Discussions », « Autre - Hors Sujet », « Autre - Contexte », « Autre - Information Déjà Connue ».
  Les tweets peuvent être multi-label, c'est-à-dire qu'ils peuvent être labellisés avec zéro, un, ou plusieurs des types d'information ci-dessus. En particulier, le type d'information « Rapport - Nouvelles » est le label correspondant aux tweets qui fournissent une couverture continue de l'évènement. Plusieurs types d'information ont été renommés (« Rapport - Nouvelles », « Rapport - Nouveau Sous-Évènement », « Autre - Contexte », « Autre - Information Déjà Connue »), un ajouté (« Rapport - Localisation ») et un autre supprimé (« Autre - Inconnu ») entre les éditions 2018 et 2019 parties A et B, ce qui explique que les types d'information listés ci-dessus ne sont pas forcément la traduction littérale des types d'information sur la figure 5.3.
— Niveaux de priorité : « Faible », « Moyen », « Haut », « Critique » qui correspondent respectivement aux scores 0,25, 0,5, 0,75 et 1.
  Une alerte est retournée si le score de priorité est supérieur à 0,7, c'est-à-dire si les niveaux de priorité sont « Haut » ou « Critique ».

3.1.2.3 Statistiques sur la collection

Au final, la collection TREC Incident Streams comptant les données des éditions 2018, 2019 partie A et 2019 partie B totalise 35 266 tweets annotés pour 33 évènements distincts. Initialement, les organisateurs ont mentionné 34 évènements mais 33

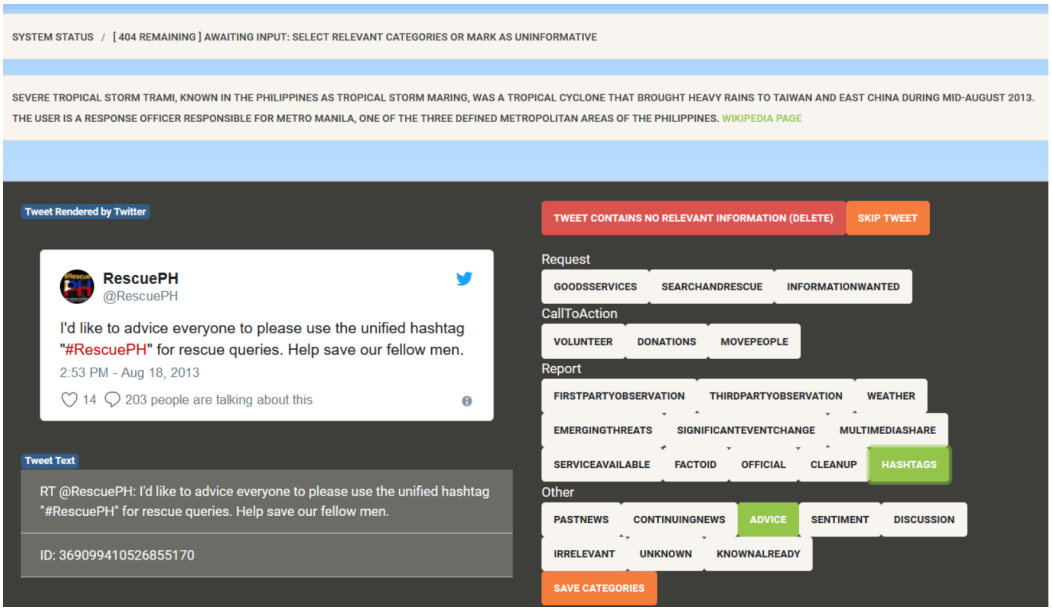


FIGURE 5.3 – Interface d'évaluation de TREC-IS (McCreadie *et al.*, 2019)

étaient au final annotés. Le tableau 5.3 reporte les évènements considérés et quelques statistiques à propos du jeu de données.

Entre 96 et 5 863 tweets ont été labellisés pour un évènement avec une moyenne de 1 068 tweets par évènement. Concernant la taille des tweets (en nombre de caractères), deux groupes distincts peuvent être observés, les tweets publiés avant 2018 et les tweets publiés après 2018. Depuis 2018 les tweets ont une taille maximale de 280 caractères tandis que cette limite est d'environ 140 avant 2018. Cela correspond en fait à l'augmentation de la taille maximale des tweets opérée par Twitter le 7 novembre 2017. Cependant, les évènements flschoolshooting2018 et floodchoco2019 ont une surprenante taille maximale de 140 caractères bien que la limite ait déjà été étendue par Twitter à cette période.

3.2 *Jeu de données proposé à partir de TREC Incident Streams : ISSumSet*

Dans la section précédente nous avons présenté les caractéristiques de la campagne TREC Incident Streams. Dans cette section, nous détaillons notre proposition de jeu de données pour le résumé automatique de flux de tweets à partir de la collection de la campagne TREC Incident Streams. En effet, cette collection n'est pas soumise

17. Pour cet évènement, un tweet était labellisé « Inconnu »

Évènement	# de tweets	# de tweets Nouvelles	# de tweets Faible	# de tweets Moyen	# de tweets Haut	# de tweets Critique	# de tweets dans le résumé CGS
costaRicaEarthquake2012	247	23	205	22	19	1	1 (0,4 %)
fireColorado2012 [17]	263	61	146	82	34	0	0 (0 %)
floodColorado2013	235	41	107	69	56	3	0 (0 %)
typhoonPablo2012	244	39	137	73	34	0	0 (0 %)
laAirportShooting2013	162	60	110	26	24	2	0 (0 %)
westTexasExplosion2013	184	27	110	52	19	3	0 (0 %)
guatemalaEarthquake2012	154	121	46	76	24	8	25 (16,2 %)
bostonBombings2013	535	148	380	86	60	9	58 (10,8 %)
flSchoolShooting2018	1 118	106	774	276	62	6	32 (2,9 %)
chileEarthquake2014	311	202	263	35	12	1	12 (3,9 %)
joplinTornado2011	96	58	39	38	12	7	17 (17,7 %)
typhoonYolanda2013	564	401	298	215	48	3	50 (8,9 %)
queenslandFloods2013	713	364	389	178	131	15	136 (19,1 %)
nepalEarthquake2015	5 863	836	3 950	959	938	16	176 (3,0 %)
australiaBushfire2013	677	435	518	123	34	2	20 (3,0 %)
philipinnesFloods2012	437	84	289	50	96	2	1 (0,2 %)
albertaFloods2013	722	353	651	39	32	0	18 (2,5 %)
typhoonHagupit2014	3 941	1 277	3 086	290	545	20	193 (4,9 %)
manilaFloods2013	411	39	330	38	33	10	6 (1,5 %)
parisAttacks2015	2 066	276	1 760	182	83	41	80 (3,9 %)
italyEarthquakes2012	103	29	79	6	15	3	17 (16,5 %)
floodChoco2019	389	14	385	4	0	0	0 (0 %)
earthquakeCalifornia2014	127	68	127	0	0	0	0 (0 %)
shootingDallas2017	2 000	232	1 945	33	18	4	17 (0,9 %)
earthquakeBohol2013	583	125	578	1	4	0	2 (0,3 %)
fireYMM2016	2 000	688	1 543	218	167	72	134 (6,7 %)
hurricaneFlorence2018	1 999	293	1 703	216	66	14	36 (1,8 %)
philippinesEarthquake2019	1 994	483	1 479	312	197	6	94 (4,7 %)
southAfricaFloods2019	1 347	408	384	716	241	6	80 (5,9 %)
cycloneKenneth2019	1 998	352	1 320	386	275	17	86 (4,3 %)
albertaWildfires2019	2 000	721	1 417	383	152	48	117 (5,9 %)
coloradoStemShooting2019	1 147	333	825	38	227	57	246 (21,4 %)
sandiegoSynagogueShooting2019	636	516	220	361	44	11	39 (6,1 %)

TABLEAU 5.3 – Quelques statistiques descriptives du jeu de données TREC Incident Streams. Pour chaque évènement, le Tableau reporte le nombre de tweets (colonne 2), le nombre de tweets labellisés « Rapport - Nouvelles » comme type d'information (colonne 3), et le nombre de tweets par niveau de priorité (colonne 4 à 7). La dernière colonne indique le nombre de tweets dans les résumés de références candidats (CGS). Le pourcentage entre parenthèses correspond au nombre de tweets gardés par rapport au nombre de tweets de l'évènement (colonne 2).

à la suppression de tweets puisque les organisateurs utilisent leur propre serveur pour les stocker. Dans un premier temps nous exposons notre intuition. Ensuite, nous analysons les résumés induits par ce jeu de données au travers des différentes exigences que doit satisfaire un résumé multi-document « idéal » selon l'état de l'art, que sont la non redondance, la couverture, la cohérence, la cohésion, le contexte, la concordance et la validité des sources.

3.2.1 *Intuition*

En analysant la manière dont la collection a été annotée, nous avons eu les intuitions suivantes, dont nous discuterons plus en détail par la suite :

— les tweets labellisés « Rapport - Nouvelles » apportent une information nouvelle et sont à priori par définition non redondants entre eux,
— ces tweets, associés à un niveau de priorité « Haut » ou « Critique » sont essentiels pour couvrir l'évènement.

Ces deux propriétés rencontrent deux exigences que doit satisfaire un résumé multi-document « idéal », à savoir la non redondance et la couverture. Comme nous l'avons vu en section 2 du chapitre 3, (Goldstein *et al.*, 2000; Huang *et al.*, 2010) définissent les besoins d'un résumé multi-document « idéal », à savoir :

— couverture : les principales informations des documents de la collection sont présentes dans le résumé ;
— non redondance : il n'y a pas de redondance d'information dans le résumé ;
— la cohésion : cela correspond à l'habileté à combiner les différents passages de texte de manière utile pour le lecteur. Les auteurs de (Goldstein *et al.*, 2000) mentionnent quelques directions pour assurer la cohésion, comme l'ordonnancement par document, la cohésion par sujet et l'ordonnancement temporel ;
— cohérence : le résumé doit être lisible et pertinent pour le lecteur ;
— contextualisation : le résumé doit inclure suffisamment de contexte pour être compris par un être humain ;
— concordance : l'information venant de différentes sources, des contradictions peuvent apparaître (par exemple, milliards au lieu de millions,...).

Ces conditions nécessaires sont valides quel que soit le type de document considéré (textes longs ou courts, textes très spécifiques comme les textes médicaux ou légaux,...). Les tweets ne sont, bien sûr, pas une exception, même si d'autres conditions peuvent être vérifiées comme la validité des sources (Vosoughi *et al.*, 2017).

Nous avons donc décidé de considérer les résumés composés des tweets labellisés à la fois « Rapport - Nouvelles » pour le type d'information et « Haut » ou « Critique » pour le niveau de priorité afin de voir s'ils feraient de bons résumés de référence. Nous

appelons ces résumés les résumés références candidats, que nous noterons aussi CGS (pour *Candidate Gold Standard*) dans le reste de ce chapitre. Nous les appelons résumés références candidats puisque les annotations ne sont pas destinées au résumé au départ, et nous souhaitons étudier la pertinence de les utiliser en tant que tels. Aussi, il peut être important de noter que nous considérons les tweets dans l'ordre chronologique dans le reste de ce chapitre.

Nous avons reporté quelques statistiques par rapport à ces CGS dans la dernière colonne du tableau 5.3. Il faut noter que les évènements earthquakeCalifornia2014, fireColorado2012, floodChoco2019, floodColorado2013, laAirportShooting2013, typhoonPablo2012, westTexasExplosion2013 ont des CGS vides. En conséquence, nous les ignorons dans le reste du chapitre.

Sur cette base, nous examinons de manière plus approfondie les possibilités des résumés références candidats de satisfaire toutes les exigences d'un résumé « idéal ».

3.2.2 *Analyse des résumés candidats*

3.2.2.1 Redondance

Comme indiqué par les organisateurs McCreadie *et al.* (2019), un évènement peut avoir été annoté par plusieurs annotateurs sur des ensembles disjoints de tweets. Par exemple, l'évènement nepalEarthquake2015 a été divisé en quatre sous-ensembles et annotés par trois évaluateurs différents, un évaluateur ayant deux de ces sous-ensembles, les autres un chacun. Comme ces ensembles sont disjoints, l'évaluation de la nouveauté est respectée par chaque annotateur mais pas entre les annotateurs. Il peut donc apparaître de la redondance pour les évènements évalués par plusieurs annotateurs. Pour traiter de ces cas, nous avons supprimé les tweets redondants des CGS comme suit. Le coût d'un processus de suppression entièrement manuel étant excessif, nous avons adopté une approche en deux phases. Dans un premier temps, nous avons supprimé automatiquement les tweets similaires par rapport à la mesure ROUGE-2 (pour rappel, nous avons défini et discuté de cette mesure en section 4.1.2 du chapitre 3). Afin de définir le seuil de similarité optimal, nous avons testé différentes valeurs (voir figure 5.4). Le seuil que nous avons choisi permettant de supprimer le plus de tweets sans une perte d'information trop grande (c'est-à-dire sans mauvaise suppression de tweets) est de 0,3. Tous les tweets ayant un score ROUGE-2 supérieur ou égal à 0,3 ont donc été supprimés des résumés références candidats. Au total, 349 tweets ont été supprimés. Nous distinguons plusieurs cas de suppression automatique :

— cas de suppression de redondance correcte
  — exactement le même tweet, dû à un retweet

Tweet 1 : 'RT : Italy earthquake : modern buildings, not ancient ones, pose biggest threat'
Tweet 2 : 'Italy earthquake : modern buildings, not ancient ones, pose biggest threat (video) -'

— même information mais exprimée différemment
Tweet 1 : 'RT : Just released by the FBI : Suspect 1 and Suspect 2 in the Boston Marathon bombing'
Tweet 2 : 'Photo of Suspects 1 and 2 in the Boston Marathon Bombings. #boston #FBI #help'

— cas de suppression de redondance discutable
— cas de mise à jour de l'évènement en utilisant le même motif
Tweet 1 : 'RT : Revised (7.5 -> 7.4) : 7.4 earthquake, 24km S of Champerico, Guatemala. Nov 7 10 :35 at epicenter (20m ago, depth 42 ...'
Tweet 2 : 'RT : Revised (6.6 -> 6.2) : 6.2 earthquake, 24km WSW of Champerico, Guatemala. Nov 11 16 :15 at epicenter (14m ago, depth ...'
— la référence à l'évènement est trop grande par rapport à la taille totale du tweet et surcharge l'évaluation de la redondance (11 tweets seulement dans ce cas)
Tweet 1 : 'Google exec dies in Mt. Everest avalanche after #NepalEarthquake'
Tweet 2 : 'Mt. Everest avalanche survivor speaks.'

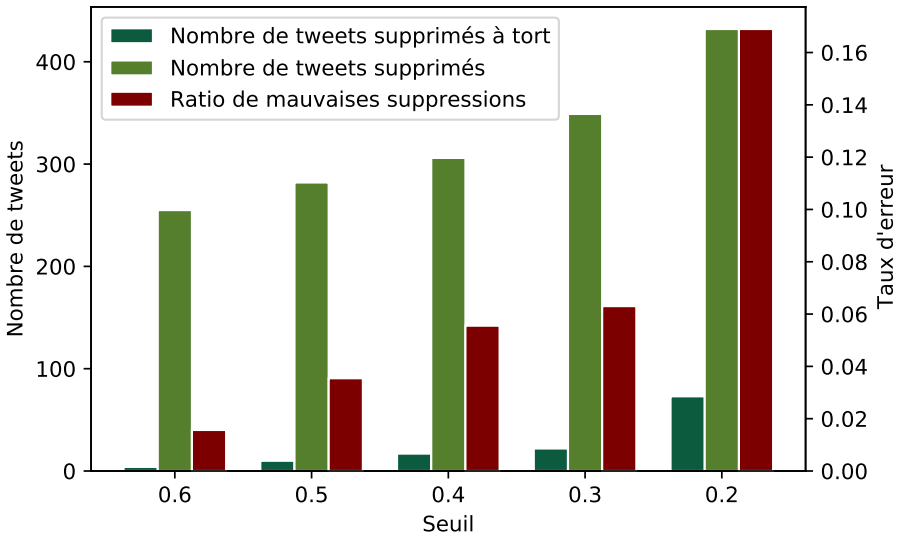


FIGURE 5.4 – Nombre de tweets supprimés par rapport au seuil de similarité ROUGE-2.

Ensuite, nous avons manuellement vérifié les CGS filtrés en utilisant un outil que nous avons spécifiquement développé. Une capture d'écran de l'interface de cet outil

est représentée sur la figure 5.5. Pour un évènement donné, les tweets sont présentés dans l'ordre chronologique dans la zone 1. Pour chaque tweet, l'information contenue est annotée comme *Nouvelle* (zone 2) ou *Déjà connue* (zone 3). Les tweets labellisés « nouveaux » sont ajoutés dans la zone 4 pour être utilisés comme mémoire pour l'annotateur et l'aider à labelliser les prochains tweets. Chaque action de labellisation peut être annulée grâce au bouton dans la zone 5. Après cette phase, 174 tweets ont été ignorés des résumés références candidats (voir le tableau 5.4, troisième colonne).

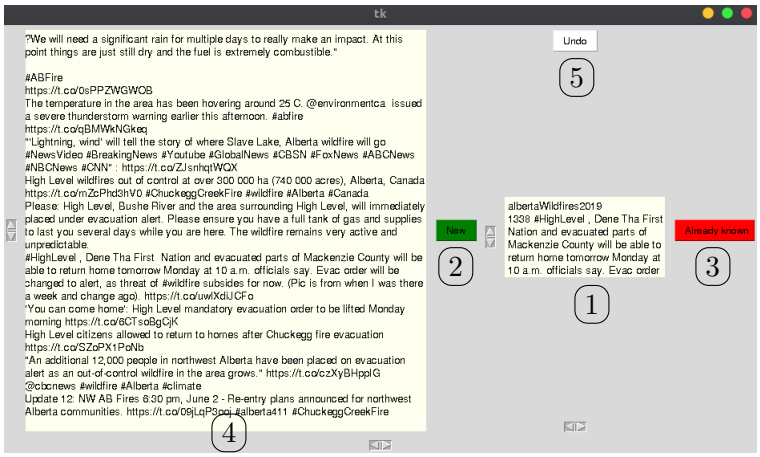


FIGURE 5.5 – Interface de l'outil d'évaluation de la redondance.

3.2.2.2 Couverture

Concernant la couverture, nous faisons l'hypothèse qu'elle est intrinsèque à la construction de la collection TREC Incident Streams. En effet, les nouvelles importantes (c'est-à-dire de niveau de priorité haut ou critique) du jeu de données initial ont été gardées pour construire nos résumés références candidats. Pour supporter cette hypothèse, nous avons analysé les CGS en ce qui concerne les sous-évènements associés à nos évènements reportés dans le portail Wikipedia Current_events [18]. Comme présenté en section 2.1, ce portail reporte quotidiennement les évènements actuels à travers le monde. Pour chaque évènement, nous avons extrait, à partir du portail Wikipedia, les sous-évènements correspondants à la période couverte par la collection TREC Incident Streams initiale. Chaque évènement correspond à au moins un sous-évènement dans le portail Wikipedia (voir le tableau 5.4, colonne 4). Le nombre total de sous-évènements est de 154 avec 6 sous-évènements par évènement en moyenne.

Par exemple, pour l'évènement « albertaFloods2013 », on peut retrouver les sous-évènements suivants :

18. `https://en.wikipedia.org/wiki/Portal:Current_events`, dernière consultation le 04/08/2022

— 20 Juin 2013 (Jeudi) : Extensive flooding begins throughout southern Alberta, Canada, leading to the evacuation of more than 100,000 people, notably in the City of Calgary and Town of High River. It would become the costliest natural disaster in Canadian history.
— 21 Juin 2013 (Vendredi) : 75,000 people are evacuated from their homes during flooding in Calgary, Alberta, Canada. (CNN)
— 22 Juin 2013 (Samedi) : 100,000 residents are displaced on the third day of flooding in Alberta. (CBC)

Ensuite, nous avons comparé les sous-évènements représentés dans les CGS avec les sous-évènements extraits du portail Wikipedia. Pour ce faire, nous avons développé un autre outil d'évaluation, une capture d'écran de l'outil est représentée sur la figure 5.6. Avec cet outil, pour chaque paire (tweet du CGS, sous-évènement) nous avons déterminé si le tweet relate de l'entièreté du sous-évènement, une partie de celui-ci ou pas du tout. Il est à noter qu'un tweet peut être associé à plusieurs sous-évènements. Au total, nous avons annoté 6 560 paires à « pas du tout » (92 %), 543 paires à « en partie » (environ 7 %), et 29 paires à « entièreté » (moins de 1 %). 67 % des tweets des différents CGS ne réfèrent à aucun sous-évènement du portail Wikipedia.

FIGURE 5.6 – Interface de l'outil d'évaluation de la couverture.

Parmi les 154 sous-évènements récupérés du portail Wikipedia, 103 sont représentés dans les CGS (environ 67 %). Les 51 sous-évènements qui ne sont pas représentés

dans les CGS s'étalent sur 10 des 26 évènements, ce qui représente un tiers des sous-évènements et affecte 38 % des évènements. Soit ces sous-évènements ne sont pas représentés par la sélection du type d'information « Rapport - Nouvelles » et le niveau de priorité « Haut » ou « Critique », soit ces sous-évènements ne sont tout simplement pas représentés dans le jeu de données initial.

Pour aller plus loin dans l'analyse de la couverture, nous avons examiné ces sous-évènements qui ne sont pas représentés dans les CGS. Nous avons cherché les mots-clés des sous-évènements dans la collection originale de tweets associés à leur évènement. 36 des 51 sous-évènements n'ont pas été trouvés du tout. Pour les 15 sous-évènements restants, nous avons distingué deux catégories :

— ceux qui ne sont pas directement liés à l'évènement (9 sous-évènements), par exemple, « Organisers and security officials reassess security plans for Sunday's 2013 London Marathon. » pour l'évènement bostonBombings2013, qui exprime que les plans de sécurité pour le marathon de Londres vont être revus, et qui peut être considéré comme une conséquence de l'attaque à la bombe de Boston.
— les autres, qui auraient dû être mentionnés.

Il apparaît donc que 6 sous-évènements sont oubliés en utilisant le filtre du type d'information « Rapport - Nouvelles » et du niveau de priorité « Haut » ou « Critique ». Pour résoudre ce problème, pour chacun de ces 6 sous-évènements, en considérant les tweets de la collection initiale dans l'ordre chronologique, nous avons ajouté manuellement, à partir d'un sous-ensemble de tweets utilisant les mots-clés du sous-évènement, les premiers tweets permettant de couvrir entièrement l'évènement.

À la fin de cette phase d'évaluation de la couverture, 1 177 tweets ont été conservés pour former les nouveaux résumés de références candidats, toujours nommés CGS dans le reste du chapitre. Le plus petit CGS est maintenant composé d'un tweet, tandis que le plus grand en contient 161.

3.2.2.3 Cohésion

Le critère de cohésion est défini par Goldstein *et al.* (2000) comme la façon d'organiser les différents passages du texte (ici les tweets) de manière efficace pour le lecteur. La cohésion peut être exprimée, entre autres, en termes de cohésion par sujet ou par ordonnancement temporel. Un exemple en rapport aux attaques de Paris en 2015 est le suivant :

1. RT @nytimesworld : ISIS reportedly claims responsibility for Paris attacks, referring to them as "miracles." https ://t.co/5cf9F4J0S1 https :/...
2. Young Frenchman Identified As Possible Bomber In Attack On Bataclan Concert Hall : PARIS (AP) — Two French poli... https ://t.co/qPlHNVawa3

3. BBC News VIDEO : Paris attack survivor : 'It was a bloodbath' https ://t.co/eqs6HTrHz4 https ://t.co/601PDL265R
4. RT @BrookingsFP : Paris Attacks : Suicide bomber tried to enter stadium, but was blocked by a French security guard. https ://t.co/7gzJUXRdKm ...
5. RT @PatVPeters : Possible link between Germany arrest and Paris attacks : senior official https ://t.co/h9zflOIAgW

Dans cet exemple, le troisième tweet n'est pas du même sujet que les quatre autres. Le troisième tweet est à propos d'une victime des attaques tandis que les autres sont à propos des assaillants. Il y a aussi une incompatibilité temporelle entre le deuxième et le quatrième tweet, nous savons que le possible poseur de bombe a été identifié avant de savoir qu'un poseur de bombe a essayé d'entrer dans le stade.

Dans un premier temps, nous avons évalué la cohésion pour l'ordonnancement temporel de nos resumés références candidats. Pour ce faire, nous avons comparé les tweets des CGS par ordre chronologique aux sous-évènements du portail Wikipedia également ordonnés chronologiquement. Nous avons créé deux listes. La première liste comprend les tweets des CGS dans l'ordre chronologique en utilisant leur horodatage. La seconde liste contient les tweets dans l'ordre chronologique des sous-évènements qui leur sont associés. Pour cette seconde liste, chaque tweet devrait être associé à un seul sous-évènement, ce qui n'est pas nécessairement le cas dans nos annotations. Pour ce faire, si un tweet porte sur l'entièreté du sous-évènement, il est associé à l'évènement. Sinon, le tweet est assigné au sous-évènement le plus récent qu'il mentionne. Les tweets associés à chaque sous-évènement sont ordonnés par ordre chronologique selon leur horodatage. Dans l'idéal, la liste des tweets ordonnés par les évènements pour lesquels ils sont associés et la liste des tweets sans l'annotation des sous-évènements sont identiques, ce qui nous permettrait de valider le critère de cohésion pour nos résumés par ordonnancement temporel. Nous avons dans un premier temps évalué la différence entre les deux listes par la distance de Manhattan. La distance de Manhattan est la distance associée à la norme 1. Soit une liste X et une liste Y de taille n éléments, la distance de Manhattan est définie par : $\sum_{i=1}^{n} |x_i - y_i|$. La distance de Manhattan idéale est donc de 0. La distance de Manhattan entre nos deux listes donne une différence moyenne de 7 pour la position d'un tweet. Nous avons poussé l'analyse en comparant chaque triplet de tweets dans les listes. Dans 63 % des cas, le tweet avant et/ou après un tweet donné est le même dans les deux listes. Ces résultats ne nous ont pas convaincu à valider une cohésion par ordonnancement temporel.

Dans un deuxième temps, la vérification manuelle pour la couverture permet d'apporter des indications sur la cohésion par sujet. Comme indiqué précédemment, en section 3.2.2.2, 67 % des tweets des CGS ne réfèrent à aucun des sous-évènements reportés dans le portail Wikipedia. En conséquence, nous pouvons garantir la cohé-

sion par sujet pour le sous-ensemble comprenant les 33 % de tweets des CGS qui réfèrent aux sous-évènements du portail Wikipedia, comme illustré en figure 5.7.

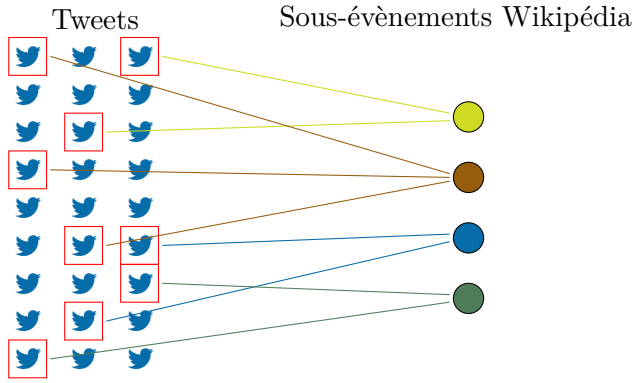


FIGURE 5.7 – Illustration de la cohésion par sujet à partir de nos annotations. Un tiers des tweets font référence à au moins un sous-évènement Wikipedia.

3.2.2.4 Cohérence

Le critère de cohérence est défini par Goldstein *et al.* (2000) comme l'habileté pour un résumé d'être pertinent et lisible pour le lecteur. D'une part, la pertinence est induite par les annotations de la campagne TREC Incident Streams. En effet, les résumés que nous considérons sont annotés comme fournissant une information nouvelle, et suffisamment importante pour retourner une alerte aux services d'urgence. D'autre part, nous avons évalué la lisibilité avec des mesures de l'état de l'art. Ces mesures sont Flesch-Kincaid (Kincaid, 1975), Gunning-Fog (Gunning, 1968), Coleman-Liau (Coleman et Liau, 1975), Dale-Chall (Dale et Chall, 1948), Automatic readability index (Ari) (Klare, 1974), Linsear write (Smith *et al.*, 1967), et Spache (Spache, 1953). Les caractéristiques utilisées par ces mesures sont principalement focalisées sur le nombre de mots, le nombre de caractères, le nombre de syllabes, et sur le nombre de mots dificiles. Nous avons utilisé les niveaux scolaires américains obtenus par ces mesures avec la librairie py-readability-metrics [19] dans notre évaluation de la lisibilité. Les niveaux scolaires Cours moyen, 6ème, 5ème, 4ème-3ème, Lycée, Université ou Grande école, et Diplômé universitaire, sont associés dans le même ordre aux plages de scores suivants 100–90, 90-80, 80-70, 70-60, 60-50, 50-30, 30-0. Nous avons évalué le jeu de données initial entier d'un côté, et les CGS de l'autre pour chaque évènement dont les CGS contiennent au moins 100 mots, ce qui affecte les évènements « earthquakeBohol2013 », « costaRicaEarthquake2012 », « philipinnesFloods2012 », « manilaFloods2013 » pour lesquels la taille en nombre de mots des CGS est inférieure à 100. Le tableau 5.4 présente les résultats pour les mesures Flesch-Kincaid, Coleman-Liau, et Dale-Chall (colonnes 6,8,10 pour le

19. `https://pypi.org/project/py-readability-metrics/`, dernière consultation le 04/08/2022

jeu de données entier, colonnes 7,9,11 pour les CGS). Les résultats pour les mesures Gunning-Fog, Ari, Linsear write et Spache sont fortement corrélés à ceux de la mesure Flesch-Kincaid (corrélation de Pearson, $r > 0{,}9$, $p - value < 0{,}01$).

Évènement	# de tweets après suppression automatique de la redondance	# de tweets après suppression manuelle de la redondance + annotation couverture	Couverture # ss-évnts Portail Wiki	Couverture # ss-évnts CGS + annotation	Lisibilité Flesch Kincaid A	Lisibilité Flesch Kincaid GS	Lisibilité Coleman Liau A	Lisibilité Coleman Liau GS	Lisibilité Dale Chall A	Lisibilité Dale Chall GS	# de termes dans le résumé entier
joplinTornado2011	17 (100 %)	17 (100 %)	20	15	9,5	8,1	9,9	10,3	11,9	11,6	190
costaRicaEarthquake2012	1 (100 %)	1 (100 %)	1	1	-	-	-	-	-	-	12
guatemalaEarthquake2012	19 (76 %)	15 (78,9 %)	1	1	12,9	14,1	12,0	12,7	14,0	13,7	186
philipinnesFloods2012	1 (100 %)	1 (100 %)+3	8	0+3	-	-	-	-	-	-	53
italyEarthquakes2012	16 (94,1 %)	13 (81,3 %)	3	3	14,8	33,4	11,8	13,6	13,8	16,3	151
bostonBombings2013	54 (93,1 %)	51 (94,4 %)+1	24	9+1	11,4	12,8	11,6	12,5	11,5	12,9	578
typhoonYolanda2013	47 (94 %)	43 (91,5 %)	26	17	11,8	16,4	12,7	13,7	12,1	13,2	447
australiaBushfire2013	20 (100 %)	20 (100 %)	4	4	10,4	11,8	11,9	14,2	11,0	12,1	224
queenslandFloods2013	127 (93,4 %)	117 (92,1 %)	4	4	10,7	11,4	11,8	12,9	11,1	12,0	1 278
earthquakeBohol2013	2 (100 %)	2 (100 %)+2	2	0+1	-	-	-	-	-	-	28
albertaFloods2013	18 (100 %)	18 (100 %)	3	3	8,6	8,6	10,9	11,2	10,8	10,7	187
manilaFloods2013	6 (100 %)	6 (100 %)	2	2	-	-	-	-	-	-	71
chileEarthquake2014	12 (100 %)	7 (58,3 %)+1	3	2+1	-	-	-	-	-	-	85
typhoonHagupit2014	168 (87 %)	159 (94,6 %)	4	4	11,5	12,5	14,0	13,1	11,9	13,0	1 493
parisAttacks2015	75(93,8 %)	52 (69,3 %)	10	7	15,5	19,1	13,4	14,1	11,8	13,0	595
nepalEarthquake2015	169 (97,7 %)	161 (95,3 %)	7	7	9,7	13,5	11,6	14,5	10,9	12,3	1 397
fireYMM2016	129 (96,3 %)	111 (86,0 %)	2	2	7,1	8,4	10,6	11,6	9,8	10,7	1 115
shootingDallas2017	17 (100 %)	17 (100 %)	2	1	8,0	10,2	10,2	14,3	9,1	11,3	194
flSchoolShooting2018	29 (90,7 %)	22 (75,9 %)	4	1	14,7	11,4	11,5	12,6	11,3	10,7	289
hurricaneFlorence2018	36 (100 %)	36 (100 %)	12	8	8,1	8,9	10,2	12,8	9,6	10,2	596
philippinesEarthquake2019	87 (92,6 %)	72 (58,6 %)	5	5	9,7	14,0	11,6	14,5	10,8	12,1	1 470
southAfricaFloods2019	60 (75 %)	50 (83,3 %)	1	1	14,2	15,6	12,0	14,6	10,4	11,1	686
cycloneKenneth2019	81 (94,2 %)	62 (74,1 %)	1	1	13,4	15,5	13,7	14,7	10,7	10,7	1 154
albertaWildfires2019	104 (88,9 %)	90 (86,5 %)	2	2	9,4	12,1	11,0	12,1	9,6	10,9	1 826
coloradoStemShooting2019	29 (11,8 %)	15 (51,7 %)	1	1	12,2	14,6	12,2	13,4	9,8	10,0	150
sandiegoSynagogueShooting2019	20 (51,3 %)	12 (60,0 %)	2	2	18,0	16,1	13,7	13,4	11,8	10,7	129
Moyenne	51,69	45,23	5,92	4,19	11,5	13,7	11,8	13,2	11,1	11,9	560,92
Écart-type	49,95	45,94	6,91	4,17	2,8	5,3	1,1	1,2	1,2	1,4	546,75

TABLEAU 5.4 – Quelques statistiques descriptives à propos des CGS. Le pourcentage entre parenthèses dans la seconde (respectivement troisième) colonne correspond au nombre de tweets gardés à partir des CGS (voir le tableau 5.3), (respectivement à partir des résumés de la seconde colonne). Dans les colonnes 4-5, des statistiques pour la couverture pour chaque évènement avec le nombre de sous-évènements dans le portail Wikipedia et le nombre de ces sous-évènements présents dans les CGS. Dans les colonnes 6 à 11, les scores de lisibilité pour les différentes mesures concernant l'entièreté du jeu de données initial (nommé A) et les CGS (nommé GS) pour chaque évènement. Les évènements sans valeur sont ceux dont le résumé pour l'évènement contient moins de 100 mots et pour lequel il n'était pas possible de calculer les scores de lisibilité. En colonne 12, le nombre de termes présents dans le résumé entier de l'évènement.

Nous pouvons observer que les scores de lisibilité sont, exceptés pour l'évènement « italyEarthquake2012 », entre le niveau 5ème et le niveau Université ou Grande école. Nous pouvons aussi noter que les scores de lisibilité pour les CGS sont supérieurs aux scores de lisibilité lorsque l'on considère l'entièreté du jeu de données initial dans la plupart des cas (80 %). Cependant, excepté pour la mesure Coleman-Liau, la différence n'est pas significative (Student t-test, sans la valeur aberrante « italyEarthquake », $p - value < 0{,}05$). La mesure Coleman-Liau prend en compte le nombre de mots dans les phrases et le nombre de caractères dans les mots. En effet, la taille moyenne des mots en nombre de caractères est de 4,6 pour l'entièreté du jeu de données initial et de 4,8 pour les CGS, et la taille moyenne des phrases en nombre de mots est de 13,0 pour l'entièreté du jeu de données initial et de 14,0 pour les CGS. Nous pensons que ces timides résultats sont dûs à l'aspect technique des tweets labellisés nouvelles et de haute priorité. Nous pensons qu'une explication pour cette différence vient de la haute priorité des tweets. Ces tweets sont supposés retourner une alerte et être traités en moins de 30 minutes par les opérateurs de services d'urgence. Cela implique une information détaillée, des mots complexes,... Par conséquent, nous avons évalué la lisibilité pour les tweets de priorité haute ou critique d'un côté, pour des résumés de tweets aléatoires (50 résumés) de même taille, de priorité faible, dans l'ordre chronologique de l'autre côté, par évènement. Les résultats montrent que dans 77 % des cas la lisibilité des tweets avec une priorité haute est moins bonne que la lisibilité des tweets avec une priorité faible. Aussi, pour la mesure Coleman-Liau la différence est toujours significative (Student t-test, sans la valeur aberrante « italyEarthquake », $p - value < 0{,}05$). La différence dans les scores de lisibilité peut donc être expliquée, au moins en partie, par la présence de tweets de priorité élevée dans nos CGS.

3.2.2.5 Contexte, concordance et validité des sources

Nous prenons pour hypothèse que le titre de l'évènement est suffisant pour comprendre le contexte. Nous assumons aussi que la concordance a été vérifiée par les annotateurs TREC Incident Streams et que la validité des sources est inhérente aux annotations TREC. Comme expliqué par McCreadie *et al.* (2019), « les évaluateurs TREC étaient des érudits de l'information, expérimentés, ayant une solide expérience analytique. Pour les tâches d'étiquetage, le NIST forme les évaluateurs à l'aide d'échantillons et travaille en étroite collaboration avec eux pour garantir la qualité et la cohérence des annotations. »

4 Bilan

Dans ce chapitre, nous avons dans un premier temps présenté les différents jeux de données existants pour le résumé automatique de flux de tweets et leurs limites pour nos travaux. Ainsi, dans un deuxième temps, nous avons proposé une méthode pour créer des jeux de données pour la tâche de résumé automatique de flux de tweets nécessitant peu d'efforts humains. Avec cette méthode nous avons créé le jeu de données TES 2012-2016, comprenant plus de 80 millions de tweets pour 28 évènements. Dans un troisième temps, nous avons créé un jeu de données pour le résumé automatique de flux de tweets à partir de la campagne d'évaluation TREC Incident Streams. Nous avons présenté la campagne Incident Streams, dont l'objectif n'est pas le résumé automatique de flux de tweets. Ensuite, nous avons évalué les potentiels résumés de référence de ce jeu de données par rapport aux propriétés d'un résumé « idéal ».

Pour résumer, dans ce chapitre, nous avons présenté deux jeux de données avec des caractéristiques différentes :

— Le jeu de données TES 2012-2016 (Dusart *et al.*, 2021b) avec des résumés orientés journalistiques, dont les résumés références sont issus d'une ressource externe aux tweets, et composé de 82 millions de tweets. C'est à notre connaissance l'une des plus grandes collections utilisée pour évaluer le résumé de tweets.

— Le jeu de données ISSumSet (Dusart *et al.*, 2021a) est plus orienté résumé d'informations vitales dans un flux de tweets, dont les résumés références sont construits à partir des tweets du flux, et composé de 35 000 tweets. Ce dernier jeu n'est pas sensible à la suppression des tweets car il est basé sur une campagne d'évaluation TREC.

Distribution des jeux de données

Pour le jeu de données TES 2012-2016 nous distribuons les résumés de référence créés à partir du portail Wikipedia Current Events sur notre Github [20].

Pour le jeu de données ISSumSet, la collection est facile à reproduire. Premièrement, les annotations et les tweets sont disponibles sur le site de la campagne TREC Incident Streams. Au moment où nous écrivons ces lignes, les identifiants des tweets correspondants au jeu de données utilisé dans ce manuscrit sont « trecis2018-A », « trecis2018-B », « trecis2019-A » and « trecis2019-B » et peuvent être téléchargés ici : `http://dcs.gla.ac.uk/~richardm/TREC_IS/2020/data.html`, dernière consultation le 04/08/2022. Les annotations sont pour leur part disponibles à l'adresse sui-

20. `https://github.com/AlexisDusart/TSSuBERT`

vante http://www.dcs.gla.ac.uk/~richardm/TREC_IS/2020/2020A/TRECIS_2018_2019-labels.json, dernière consultation le 04/08/2022.

L'un des principaux objectifs de TREC étant de créer des collections de tests et de les distribuer à la communauté des chercheurs, il y a fort à parier que ces ensembles de données resteront accessibles.

Deuxièmement, nous fournissons le code pour générer les résumés de référence sur notre Github [21]. Des annotations supplémentaires pour les sous-évènements (annotations manuelles basées sur le portail Wikipedia Current Events), qui peuvent être utilisées pour générer des nouveaux résumés références, sont aussi fournies.

La création de ces jeux de données nous a permis d'avoir un cadre d'évaluation et ainsi de poursuivre les travaux autour de la proposition de modèles de résumé automatique de flux de tweets. C'est pourquoi, dans le chapitre suivant, nous proposons une approche pour utiliser les modèles de langue neuronaux dans le cadre du résumé automatique de flux de tweets.

21. https://github.com/AlexisDusart/ISSumSet

6

UTILISATION DE MODÈLES DE LANGUE PRÉ-ENTRAÎNÉS POUR LE RÉSUMÉ AUTOMATIQUE DE FLUX DE TWEETS

Sommaire

Introduction

Dans le chapitre précédent, nous avons exposé la première contribution de cette thèse, à savoir construire un jeu de données à l'aide d'annotations externes et capitaliser sur un jeu de données existant pour en faire un jeu de données pour le résumé automatique de tweets. Dans ce chapitre, nous présentons la deuxième contribution de cette thèse. Nous avons détaillé dans le chapitre 3 les différents types de résumé automatique, mono ou multi-document, abstractifs ou extractifs. Nous avons évoqué l'utilisation des modèles de langue pré-entraînés, que nous avons définis en section 3.1.7, et les améliorations qu'ils apportent dans la plupart des tâches du TAL (Traitement Automatique des Langues). Nous proposons une approche de génération automatique de résumés extractifs tirant profit de ces modèles de langue pré-entraînés.

La suite de ce chapitre est organisée comme suit. Dans un premier temps nous présentons le contexte et la motivation à ce travail, puis nous définissons les problématiques associées. Ensuite, nous exposons le modèle que nous proposons, l'architecture et les variantes testées. Enfin, nous évaluons notre modèle par rapport à l'état de l'art, et nous discutons des résultats.

1 Contexte et motivations

Bien que de nombreuses approches existent dans la littérature, (Rudra *et al.*, 2018b, 2019; Chellal et Boughanem, 2018; Fernández *et al.*, 2018), le résumé automatique de grands volumes de données est toujours un problème ouvert (El-Kassas *et al.*, 2021). Les approches doivent prendre en compte des questions telles que la redondance ou le traitement en temps réel, en plus de la gestion de grands volumes de données.

Nous avons dressé un état de l'art du résumé automatique textuel dans le chapitre 3, et avons exploré l'état de l'art du résumé automatique textuel lorsque l'on prend en compte la dimension temporelle dans le chapitre 4. Nous remarquons de ces examens que les approches récentes de l'état de l'art du résumé automatique de texte sont principalement basées sur des modèles neuronaux qui utilisent des modèles de langue pré-entraînés (Liu et Lapata, 2019) comme BERT (Devlin *et al.*, 2019), à l'instar de nombreuses tâches du TAL. À notre connaissance, de tels modèles de langue pré-entraînés neuronaux n'ont pas été utilisés pour le résumé automatique de flux, excepté le travail de Li et Zhang (2021) qui se focalise sur la génération de résumés qui répondent aux questions qui-quoi-quand-où (4W who-what-when-where). Cette

rareté pourrait être due à la difficulté de gérer le flux de tweets avec ces modèles ou au manque d'ensembles de données d'entraînement suffisamment grands.

Dans ce chapitre, nous souhaitons tirer profit des récentes avancées de l'état de l'art concernant les modèles de langue pré-entraînés, tout en les adaptant à de très grands volumes de données et au résumé incrémental de flux de tweets. Nous abordons ainsi l'objectif de recherche suivant :

Objectif de recherche : Explorer l'intérêt des modèles de langue pré-entraînés neuronaux ajustables pour le résumé automatique de flux de tweet.

Pour répondre à cet objectif, nous proposons la contribution suivante :

Méthodologie pour répondre à l'objectif de recherche : Nous proposons une approche incrémentale pour le résumé extractif de flux de tweets, capable de résumer de grands flux de tweets sur un sujet particulier. Cette approche présente trois aspects originaux : (i) elle combine un modèle pré-entraîné (BERT) avec le contexte du tweet pour prédire l'importance d'un tweet, (ii) elle estime automatiquement la taille appropriée du résumé à proposer à un instant donné, et (iii) elle ne nécessite pas de ré-évaluer tous les documents à chaque incrément de temps, entraînant ainsi un gain d'efficacité.

2 Définition du problème

De manière générale, le problème abordé est de construire des résumés successifs à partir d'un flux continu de tweets sur un sujet/évènement particulier à différents instants successifs. Un résumé est généré à un instant donné, à partir de tous les tweets composant le flux, depuis le début du flux jusqu'à l'instant donné. Comme le flux de tweets peut représenter un grand volume d'informations (plusieurs millions de tweets), il ne semble pas raisonnable de construire chacun des résumés successifs à partir de zéro, en recommençant toujours depuis le début du flux. Il est préférable de compléter le résumé précédemment généré en prenant en compte les nouveaux tweets arrivés depuis. Le problème se résume donc à un problème de *mise à jour* (voir section 1.2 du chapitre 4), que nous renommons résumé incrémental pour plus lisibilité.

Plus précisément, en utilisant les concepts que nous avons définis dans le tableau 6.1 nous formulons le problème comme suit :

TABLEAU 6.1 – Concepts pour la formalisation de la tâche.

Concept	Notation	Définition
Évènement	E	Évènement ou sujet, c-à-d, l'objet des tweets.
Flux	T	Flux de tweets à propos de l'évènement considéré.
Horodatage	i	Date (heure-minute-secondes) de publication d'un tweet.
Sous-flux	$T_{i,j}$	Flux ne prenant en compte que les tweets publiés entre deux horodatages. Pour $T_{i,j}$, le sous-flux ne considère que les tweets publiés entre les horodatages i et j.
Résumé	$S_{i,j}$	Résumé composé des tweets publiés entre deux horodatages. Pour $S_{i,j}$, le résumé est composé des tweets publiés entre les horodatages i et j.
Incrément de temps		Comme nous considérons le résumé incrémental, un incrément de temps est une durée fixe entre la génération de deux résumés consécutifs.

Considérons un évènement donné E et un flux T de tweets à propos de E. $T_{i,j}$ est défini comme la partie de T entre les deux horodatages i et j, c'est-à-dire le *sous-flux*.

Considérons t_0 l'horodatage correspondant au début du flux et t l'horodatage auquel un résumé doit être construit (avec t au moins un incrément de temps (ou *pas de temps*) supérieur à t_0). L'objectif est d'extraire le résumé $S_{t_0,t}$ du sous-flux $T_{t_0,t}$, c'est-à-dire que $S_{t_0,t}$ est un sous-ensemble des tweets qui composent $T_{t_0,t}$.

Dans le contexte du résumé incrémental, le problème consiste à générer $S_{t_0,t}$ de telle sorte que $S_{t_0,t} = S_{t_0,t-1} \cup S_{t-1,t}$ [1] (avec $S_{t_0,t_0} = \emptyset$).

3 Modèle proposé

Dans la section précédente, nous avons défini le problème étudié. Dans cette section nous présentons le modèle proposé, que nous avons nommé TSSuBERT, pour y répondre.

Les modèles de langue pré-entraînés sont utilisés par les approches de l'état de l'art. Ces approches représentent l'entièreté du (ou des documents concaténés) à résumer

1. Plus la durée entre les deux horodatages $t-1$ et t est faible, plus l'approche peut être considérée comme du « temps réel ».

à l'aide de ces modèles de langue. Cependant, elles s'en retrouvent ainsi inutilisables pour des volumes de données aussi important qu'un flux de tweets. L'idée du modèle proposé est de tirer parti de ces modèles de langue pré-entraînés pour représenter un tweet, mais de représenter le flux de tweets à l'aide de la fréquence des termes du flux. Afin de traiter de grands volumes de données, comme peuvent l'être des flux de tweets, la représentation du contexte, et le modèle pour associer les représentations du tweet et du contexte, sont choisis volontairement pour leur simplicité.

3.1 *Architecture du modèle*

Le modèle est composé de deux grandes parties, une première partie pour prédire l'importance d'un tweet, puis la seconde pour sélectionner les tweets à garder dans le résumé.

3.1.1 *Vue d'ensemble du modèle*

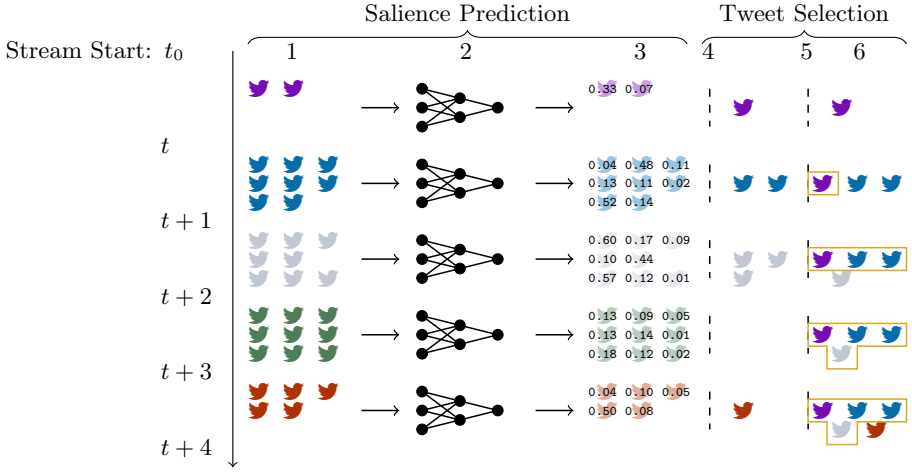


FIGURE 6.1 – Aperçu de l'architecture du modèle pour un évènement avec 5 incréments de temps (t à $t+4$). Chaque colonne correspond à : (1) le flux de tweets pour l'incrément de temps considéré, (2) la prédiction du score d'importance, (3) la sortie du modèle, c'est-à-dire les tweets avec leur score d'importance, (4) le filtrage des tweets en fonction de leur importance, (5) le filtrage des tweets en fonction de leur similarité avec le résumé existant, (6) le résumé incrémental des résultats de l'extraction. Pour chaque incrément de temps, l'encadrement modélise le résumé précédent.

Comme nous l'avons vu dans la section 1.2 du chapitre 4, les techniques de résumé de mise à jour utilisent en général une approche MDS (résumé multi-document) sur les nouveaux documents et suppriment ensuite la redondance. Nous avons adopté la

même stratégie en décomposant le problème en deux phases successives, se déroulant à chaque incrément de temps, c'est-à-dire à la fin de chaque fenêtre de temps (voir figure 6.1). La première phase vise à déterminer un ensemble de tweets candidats à inclure dans le résumé. À cette fin, un score d'importance est prédit pour chaque nouveau tweet du flux en fonction de son contexte. Le contexte comprend tous les tweets du flux obtenus à la fin de la fenêtre de temps depuis le début du flux (étapes 1 à 3 dans la figure 6.1). La deuxième phase considère ensuite les tweets importants un par un et sélectionne de manière incrémentale ceux qui ajoutent des informations aux informations déjà connues (étapes 4 à 6 dans figure 6.1). Pour la première fenêtre de temps considérée, il n'y a aucune information connue, et le résumé sera alors construit de manière incrémentale à partir du premier tweet sélectionné.

3.1.2 *Prédiction d'importance*

L'objectif de cette phase est de décider si un tweet doit être conservé comme candidat dans le résumé de l'évènement, c'est-à-dire si le tweet présente un intérêt pour le résumé. À cette fin, un score d'importance est prédit pour chaque tweet. Ce score d'importance exprime un score de similarité entre le tweet et le résumé de référence. La phase d'apprentissage du modèle consiste donc à apprendre le score de similarité entre chaque tweet et le résumé de référence, et non à prédire directement le contenu textuel du résumé de référence.

Dans l'étape de test, comme en production, pour laquelle il n'existe pas de résumé de référence, le modèle entraîné est appliqué aux nouveaux tweets générés pour estimer leur score. Idéalement, plus le score d'importance estimé est élevé, plus le tweet sera proche d'un résumé de référence potentiel.

La prédiction de l'importance est effectuée à la fin de chaque incrément de temps, c'est-à-dire uniquement pour les tweets du nouveau sous-flux $T_{t-1,t}$.

Pour cette phase, nous tirons parti des représentations textuelles les plus récentes, c'est-à-dire des modèles de langue pré-entraînés tels que BERT (Devlin *et al.*, 2019). De telles représentations ont été appliquées à des tâches spécifiques de compréhension du langage naturel en ajustant une couche de sortie finale pour la tâche. Pour notre tâche de résumé du flux de tweets sur un évènement donné, le modèle de prédiction de l'importance doit fonctionner sur des évènements qui peuvent être très différents. Prédire l'importance d'un tweet sans aucune information contextuelle sur l'évènement ne permettrait pas de capturer les différences entre les évènements. Nous avons donc décidé d'intégrer le contexte des tweets en utilisant l'ensemble du vocabulaire lié à l'évènement, c'est-à-dire les *tokens*, représentés par leurs fréquences.

Formellement, pour un résumé $S_{t_0,t}$, les fréquences sont calculées sur tous les *tokens* de tous les tweets qui composent $T_{t_0,t}$. Pour un *token* v dans l'ensemble du vocabulaire V, la fréquence de v à l'instant t, $F_{v,t}$, est calculée comme le nombre d'occurrences de v dans le sous-flux $T_{t_0,t}$:

$$F_{v,t} = \sum_{tw \in T_{t_0,t}} \sum_{w \in tw} \mathbb{1}_{\{w=v\}}$$

où tw représente chaque tweet du flux $T_{t_0,t}$, et w représente chaque *token* pour chaque tweet tw, en considérant tw comme un ensemble de *tokens*.

Par exemple, considérons T_{t_0,t_1} composé d'un tweet {« This is the first tweet »}. Ici, $F_{v,t} = 1$ pour tous les *tokens* v dans {« this », « is », « the », « first », « tweet » }, c'est-à-dire que la représentation des fréquences du sous-flux T_{t_0,t_1} est un vecteur qui a pour valeur 0 pour tous les *tokens*, exceptés pour les *tokens* dans {« this », « is », « the », « first », « tweet »} qui ont une valeur de 1. Maintenant considérons qu'à la fin d'une seconde fenêtre de temps à l'horodatage t_1, les deux tweets « This is the second tweet » et « This is the third tweet » sont ajoutés. La représentation des fréquences pour le sous-flux T_{t_0,t_2} est un vecteur qui a pour valeur 0 pour tous les *tokens*, excepté pour les *tokens* dans {« first », « second », « third »} qui ont une valeur de 1, et pour les *tokens* dans {« this », « is », « the », « tweet »} qui ont une valeur de 3.

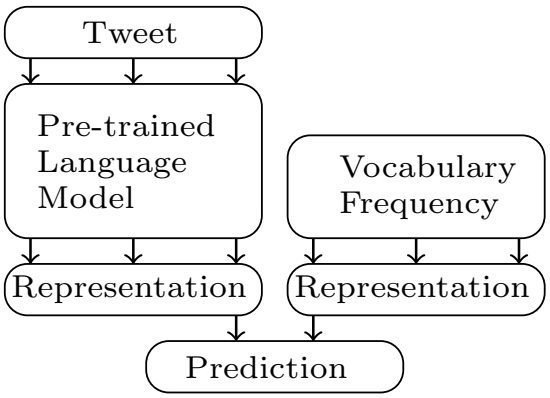


FIGURE 6.2 – Architecture de la partie prédiction de l'importance.

L'architecture de cette partie de prédiction de l'importance de notre modèle est présentée dans la figure 6.2.

Deux contributions peuvent être mises en évidence à partir de cette architecture : (i) l'utilisation combinée d'un modèle de langue pré-entraîné et de représentations de fréquences de vocabulaire pour prédire l'importance ; (ii) l'utilisation d'un modèle de langue pré-entraîné pour le résumé extractif de tweets, ce qui, étonnamment, ne se trouve pas dans la littérature, à l'exception du travail de Li et Zhang (2021).

3.1.3 *Sélection des tweets*

Pour cette étape, nous supposons que les fenêtres de temps précédentes ont déjà été traitées. Nous supposons également qu'un score d'importance est attribué à tous les tweets de la fenêtre de temps (sous-flux), actuellement traités. Les tweets dont le score d'importance est supérieur à un seuil donné $\lambda_{importance}$ sont ensuite considérés un par un et selon leur score d'importance décroissant pour être inclus dans le résumé.

Pour limiter la redondance à la manière de la MMR (Maximal Marginal Relevance) (Carbonell et Goldstein, 1998), nous considérons la similarité entre chaque tweet candidat c et le résumé existant. Plus précisément, un score de similarité est calculé entre chaque tweet candidat et chaque tweet du résumé en construction. Comme illustré en figure 6.3, le résumé en construction est composé de tweets des résumés précédents et des tweets déjà sélectionnés pour la fenêtre de temps actuelle. Le résumé précédent résume tous les horodatages précédents. De la même manière, le résumé précédent contient aussi les résumés qui le précède avec des tweets des fenêtres de temps qui le précède, et ainsi de suite.

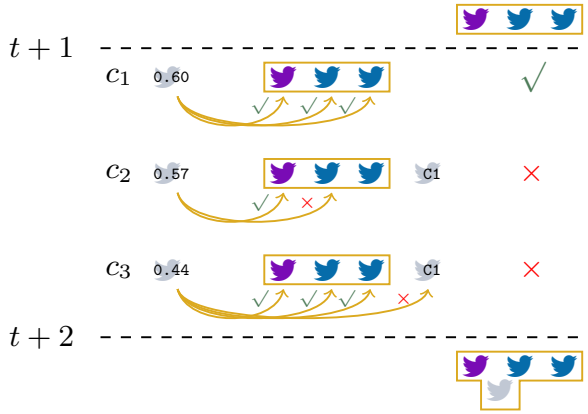


FIGURE 6.3 – Zoom sur la figure 6.1 pour la partie prédiction de la fenêtre de temps entre $t+1$ et $t+2$. Les tweets dont le score de pertinence est supérieur au seuil $\lambda_{importance}$ sont candidats. Le tweet c_1 est ajouté au résumé car la similarité entre le tweet c_1 et chacun des tweets du résumé existant, c'est-à-dire du résumé obtenu pour la fenêtre de temps précédente, est inférieure au seuil $\lambda_{similarite}$. Le tweet candidat c_2 n'est pas ajouté au résumé car sa similarité avec le deuxième tweet du résumé existant est supérieure au seuil $\lambda_{similarite}$. Le tweet candidat c_3 n'est pas ajouté au résumé car sa similarité avec le dernier tweet du résumé existant (ici c_1) est supérieure au seuil $\lambda_{similarite}$.

Chaque tweet ayant un score de similarité inférieur à un seuil $\lambda_{similarite}$ avec tous les tweets déjà présents dans le résumé en construction est sélectionné. Dans le cas particulier d'un résumé existant vide, c'est-à-dire qu'il s'agit de la première fenêtre de temps pour laquelle il existe au moins un tweet avec un score d'impor-

tance supérieur à $\lambda_{importance}$, le tweet avec le score d'importance le plus élevé est automatiquement conservé. Le pseudo-code du processus de sélection de tweet est présenté dans l'algorithme 1.

L'un des points forts de notre modèle réside dans cette sélection de tweets basée sur deux seuils : contrairement à de nombreux modèles de l'état de l'art, nous n'avons pas besoin de donner une taille de résumé fixe en entrée du modèle, puisqu'il adapte automatiquement la taille du résumé de sortie en fonction du flux de tweets d'entrée.

Algorithme 1 : Sélection de tweet pour le sous-flux $T_{t-1,t}$ pour obtenir le résumé $S_{t_0,t}$, en mettant à jour de façon incrémentale le résumé précédent $S_{t_0,t-1}$.

Entrée : C ensemble de tweets candidats issus du sous-flux $T_{t-1,t}$ entre les horodatages $t-1$ et t, c'est-à-dire ayant un score d'importance supérieur à $\lambda_{importance}$, ordonnés par score décroissant, $S_{t_0,t-1}$ résumé existant, c'est-à-dire l'ensemble des tweets sélectionnés à $t-1$.

Sortie : Résumé mis à jour de $S_{t_0,t}$, c'est-à-dire l'ensemble des tweets sélectionnés à t.

```
début
    S_{t0,t} ← S_{t0,t-1}
    pour c ∈ C faire
        si S_{t0,t}.estVide() alors
            S_{t0,t}.ajouter(c)
        sinon
            si (sim(c,s) < λ_similarite pour chaque s ∈ S_{t0,t}) alors
                S_{t0,t}.ajouter(c)
            fin
        fin
    fin
fin
```

4 Expérimentations et résultats

Plusieurs types d'évaluation ont été conduits sur notre modèle. Une première évaluation automatique a utilisé des métriques de similarité syntaxique et sémantique par rapport à un résumé de référence. Comme ces métriques ont tendance à ignorer ce qui rend un résumé utilisable par les humains, nous avons complété l'évaluation automatique par une évaluation manuelle, en nous concentrant davantage sur l'informativité. De plus, nous avons évalué l'efficience de notre modèle par rapport aux

modèles de l'état de l'art. Dans cette section, nous présentons la configuration de ces expériences et les résultats obtenus.

4.1 *Évaluation automatique*

Dans les paragraphes suivants, nous détaillons notre dispositif expérimental pour l'évaluation automatique du modèle proposé.

4.1.1 *Jeux de données*

Les expériences ont été menées sur trois jeux de données différents :

— Nous avons d'abord utilisé un jeu de données de l'état de l'art, à savoir celui de Rudra *et al.* (2018b). Comme présenté en section 1.1 du chapitre 5, ce jeu de données est affecté par la suppression des tweets, et au moment où nous avons réalisé les expériences, seuls 63 % des tweets restaient actuellement accessibles. Les modèles ont donc été réévalués sur notre version du jeu de données pour une comparaison équitable.
— Deuxièmement, nous avons utilisé le jeu de données ISSumSet (Dusart *et al.*, 2021a), présenté en section 3 du chapitre 5.
— Finalement, nous avons également utilisé le jeu de données TES 2012-2016, présenté en section 2 du chapitre 5.

Nous listons quelques statistiques sur ces 3 jeux de données dans le tableau 6.2. Nous pouvons constater que ces jeux de données sont hétérogènes à la fois en termes de nombre de flux, de nombre d'incréments (c'est-à-dire le nombre de jours), de nombre de tweets et de longueur de résumé de référence. La collection TES 2012-2016 est beaucoup plus importante en termes de tweets par évènement (environ 3 millions par évènement). Enfin, il existe une hétérogénéité au niveau des résumés de référence : 2 jeux de données (ISSumSet et celui de Rudra *et al.* (2018b)) considèrent des tweets comme résumés de référence tandis que l'autre (TES 2012-2016) utilise des phrases Wikipedia. Toutes ces hétérogénéités nous permettent d'enrichir les scénarios d'évaluation. Les résumés de référence sont également différents en termes de contenu textuel. Les résumés de référence des jeux de données ISSumSet et de Rudra *et al.* (2018b) sont composés de tweets tandis que ceux de TES 2012-2016 sont dans une forme journalistique, composés de résumés du Wikipedia Current Event Portal, qui ne sont pas des tweets.

TABLEAU 6.2 – Quelques statistiques pour la collection TES 2012-2016, la collection de Rudra *et al.* (2018b) et la collection ISSumSet. Le nombre entre () pour la longueur du résumé de référence représente le nombre moyen de mots ajoutés à chaque incrément de temps, c'est-à-dire chaque jour.

Collection	# flux	# jours (total)	Moyenne # tweets/évènement	Taille moyenne des résumés de référence
TES 2012-2016	28	545	2 939 842	265 (14)
Rudra et al.	11	30	7 261	540 (198)
ISSumSet	26	397	1 295	566 (37)

4.1.2 *Apprentissage*

Tout d'abord, pour choisir *la métrique de similarité à utiliser dans le modèle de prédiction de l'importance* (voir section 3.1.2), nous avons effectué une évaluation manuelle préliminaire sur le jeu de données TES 2012-2016, en comparant les résumés Oracles créés avec les mesures ROUGE-2 F et Cosinus pour chaque évènement. Pour ce faire, pour chaque jour de l'évènement, nous avons utilisé un algorithme *glouton* qui conserve les tweets maximisant la métrique, comme cela est fait dans (Nallapati *et al.*, 2017; Liu et Lapata, 2019; Dou *et al.*, 2021). Les résultats suggèrent que le score Cosinus est plus approprié que le score ROUGE-2 F habituellement utilisé : (Liu et Lapata, 2019; Nallapati *et al.*, 2017; Dou *et al.*, 2021). Par conséquent, les tweets inclus dans les résumés de l'Oracle Cosinus se sont vus attribuer un score parfait (1,0) pendant l'apprentissage.

Deuxièmement, en ce qui concerne le *modèle de langue pré-entraîné* utilisé pour la prédiction d'importance des tweets, nous avons décidé d'utiliser le modèle pré-entraîné DistilBERT (Sanh *et al.*, 2019), une version allégée de BERT (Devlin *et al.*, 2019) [2]. Comme le calcul du modèle BERT est quadratique par rapport à la longueur de l'entrée, nous avons fixé la limite de taille à 50 *tokens* pour profiter de la petite taille d'un tweet (plus de 99,99 % des tweets de la collection ont une longueur < 50 *tokens*). Le *contexte* est évalué pour chaque tweet candidat c comme la fréquence de chaque *token* du vocabulaire de tweet apparaissant dans les tweets de la fenêtre de temps de c et des fenêtres de temps qui la précède. Le vocabulaire de tweet est composé de 30 525 *tokens* uniques. Plus précisément, nous avons utilisé le vocabulaire d'apprentissage de BERT, c'est-à-dire 30 522 *tokens* moins le *token* PAD. Le *token* PAD est le *token* par défaut pour atteindre la taille limite fixée (ici 50) si le tweet est

2. Des modèles plus complexes appliqués au résumé abstractif tels que BART (Lewis *et al.*, 2020) et PEGASUS (Zhang *et al.*, 2020a) pourraient être utilisés pour cette étape. Nous laissons leur intégration pour un travail futur puisque notre objectif ici est de valider notre approche dans son ensemble.

plus petit. En outre, nous avons ajouté 4 *tokens* pour les URLs, hashtags, mentions (@), et retweets. Les tweets ont été pré-processés en conséquence en remplaçant les quatre cas susmentionnés par leur *token* associé.

Notre *architecture neuronale* est détaillée dans figure 6.4. Chaque couche dense utilise une fonction d'activation ReLu. De plus, comme proposé par Srivastava *et al.* (2014), nous avons ajouté sur chacune de ces couches, sauf celle de prédiction, une couche de Dropout avec une probabilité de 0,5.

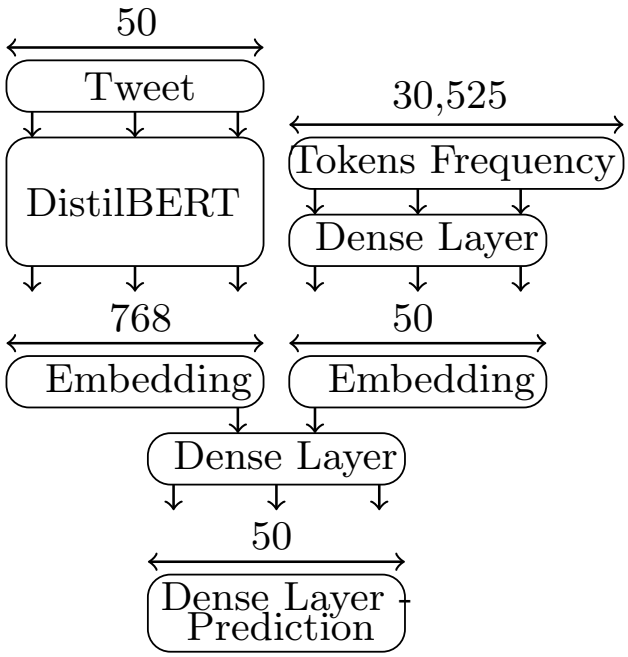


FIGURE 6.4 – Architecture expérimentale de la partie prédiction d'importance.

Parmi les trois ensembles de données susmentionnés, notre modèle a été entraîné sur les ensembles ISSumSet et TES 2012-2016 :

— Lorsque l'apprentissage et le test ont été effectués sur le même jeu de données, nous avons procédé à une validation croisée à trois *folds*. Le jeu de données TES 2012-2016 a été divisé en 2 *folds* contenant 9 évènements, et le dernier contenant 10 évènements. Le jeu de données ISSumSet a été divisé en 2 *folds* contenant 9 évènements, et le dernier contenant 8 évènements. Ces divisions avaient pour but de produire des *folds* comprenant approximativement le même nombre de tweets pour chaque ensemble de données. Il convient de noter que la validation croisée a été effectuée entre les évènements et non au sein de chaque évènement.
— Lorsque l'apprentissage et le test ont été effectués sur des ensembles de données différents, nous avons appris notre modèle sur tous les évènements présents

dans l'ensemble de données d'apprentissage, à l'exception de ceux éventuellement présents dans l'ensemble de données de test.

Il convient de noter que l'ensemble de données de Rudra *et al.* (2018a) a été écarté pour l'apprentissage car il ne contient pas suffisamment d'évènements pour une validation croisée ou même pour un apprentissage « total ». En fait, cet ensemble de données est composé de 11 flux mais de seulement 3 évènements, *Typhoon Hagupit 2014*, *Nepal Earthquake 2015*, et *Pakistan Flood 2014*. Les flux sont liés à différentes classes telles que « Dommages aux infrastructures » ou « Attention et conseils » (4 flux pour *Typhoon Hagupit* et *Nepal Earthquake* et 3 flux pour *Pakistan Flood*). De plus, les évènements *Typhoon Hagupit 2014* et *Nepal Earthquake 2015* sont également inclus dans les collections TES 2012-2016 et ISSumSet, et ils ne peuvent évidemment pas être utilisés à la fois pour l'apprentissage et les tests.

Nous avons entraîné notre modèle en utilisant l'optimiseur Adam avec les paramètres $\beta_1 = 0{,}9$ et $\beta_2 = 0{,}999$, une taille de *batchs* de 128, un taux d'apprentissage décroissant comme dans (Liu et Lapata, 2019; Vaswani *et al.*, 2017) avec une fonction objectif d'erreur quadratique moyenne. Pour chaque *fold*, nous avons entraîné le modèle sur 90 % des 2 autres *folds* avec 10 % pour la validation, pendant 5 *epochs* sur TES 2012-2016 et pendant 1 000 *epochs* sur ISSumSet en raison de la différence de taille des différents jeux de données.

4.1.3 *Test*

Les évaluations ont été réalisées sur 3 jeux de données, à savoir TES 2012-2016, ISSumSet et le jeu de données de Rudra *et al.* (2018a). Pour chaque jeu de données, nous avons expérimenté deux versions du modèle proposé, l'une limitant ses résumés à la longueur des résumés de référence, et l'autre non étant donné que notre modèle ne nécessite pas de fixer une taille limite. De plus, pour chacune de ces deux versions, nous avons comparé un modèle appris sur le jeu de données ISSumSet, et l'autre sur le jeu de données TES 2012-2016.

Le seuil d'importance $\lambda_{importance}$ a été fixé en fonction des valeurs de test des expériences préliminaires par pas de 0,1. Il a été fixée à 0,2 sauf dans le cas où la collection de données TES 2012-2016 a été utilisée pour l'entraînement mais pas pour les tests (les tests sont donc effectués sur les collections Rudra et ISSumSet). En effet, dans ce dernier cas, la valeur 0,2 conduit à des résultats médiocres en termes de nombre de tweets récupérés sur les collections de test, ou même parfois à des résumés vides. Une explication possible est que la taille moyenne de l'incrément dans le jeu de données TES 2012-2016 (14 termes) est beaucoup plus faible que pour les autres collections (198 et 37 respectivement). Le modèle apprend donc à prédire peu d'informations sur le jeu de données TES 2012-2016 alors que plus

d'informations sont nécessaires sur les deux autres collections. Pour contrebalancer cet effet, nous fixons donc le seuil d'importance $\lambda_{importance}$ à 0,0 dans le cas où le jeu de données TES 2012-2016 est utilisé pour l'entraînement mais pas pour les tests. Notez que ce cas n'est pas comparable à une sélection de tweets au hasard car tous les tweets ayant un score de 0,0 ne sont finalement pas conservés dans les résumés.

Enfin, le seuil de similarité $\lambda_{similarite}$ a été fixé expérimentalement comme suit pour toutes les versions, la longueur étant exprimée en nombre de *tokens* :

— $\lambda_{similarite} = 0{,}3$ si $taille(S_{t_0,t}) < 50$

— $\lambda_{similarite} = 0{,}3 * \frac{log(50)}{log(taille(S_{t_0,t}))}$ sinon

Ce seuil adaptatif vise à éviter la redondance dans les résumés et à réduire la taille des résumés prédits.

4.1.4 *Métriques*

Pour chaque modèle, comme cela se fait habituellement pour l'évaluation du résumé, nous avons calculé les scores ROUGE-1 de précision et de F-mesure ainsi que les scores de précision et de F-mesure de ROUGE-2. Comme présenté en section 4.1.2 du chapitre 3, les métriques issues de ROUGE sont les plus utilisées pour l'évaluation du résumé automatique, mais sont limitées, notamment d'un point de vue sémantique. Nous avons ainsi ajouté une mesure de similarité sémantique, appelée Cos Embed, basée sur la métrique Cosinus et des *embeddings* largement utilisés [3] en utilisant Smooth Inverse Frequency (SIF) (Arora *et al.*, 2017) pour représenter les résumés et les résumés de référence. Cette métrique est particulièrement pertinent pour les expérimentations sur le jeu de données TES 2012-2016. En effet, la collection peut être impactée par un problème d'inadéquation du vocabulaire : les résumés générés sont composés de tweets alors que les résumés de référence sont dans une forme journalistique. Nous n'utilisons pas les mesures de similarités QuestEval ou BERTScore, en raison de leur limitation à 512 *tokens*. Les résumés peuvent relater de faits sur plusieurs jours, voire plusieurs semaines, ils deviennent ainsi souvent plus longs que 512 *tokens*. Comme des performances similaires ont été observées sur les scores macro et micro, nous avons décidé de ne rapporter que les scores macro, c'est-à-dire la moyenne des résultats moyens pour chaque flux. Lors de l'évaluation, le pré-traitement est le même pour les tweets et les résumés de référence, que ce soit pour ISSumSet ou TES 2012-2016.

3. Nous avons utilisé le modèle word2vec-google-news-300 fourni par gensim `https://radimrehurek.com/gensim/models/word2vec.html`, dernière consultation le 08/08/2022.

4.1.5 *Baselines*

Nous avons choisi COWTS (Rudra *et al.*, 2015), SEMCOWTS (Rudra *et al.*, 2018a) et SCC (Rudra *et al.*, 2018b) comme *baselines* de l'état de l'art à comparer, car elles sont reproductibles et les plus proches en termes de problèmes abordés. Nous avons utilisé les implémentations fournies par les auteurs (cf. `https://github.com/krudra/`, dernière consultation le 08/08/2022). Il convient de noter que, contrairement à notre approche, ces approches réévaluent l'entièreté des résumés à fournir à chaque fenêtre de temps et ne proposent pas d'approche incrémentale. En outre, nous rapportons une *baseline* aléatoire (Randoms) construite comme suit : nous avons généré 50 résumés aléatoires de la même taille que les résumés de référence et nous avons rapporté la moyenne des 50 scores associés. Enfin, nous avons également rapporté la *borne maximale*, soit les résumés Oracles, construits selon les métriques ROUGE-2 F et Cosinus pour les collections TES 2012-2016 et ISSumSet, comme lors de l'apprentissage (voir en section 4.1.2).

4.2 *Résultats*

Nous présentons, dans le tableau 6.3, les spécifications des différentes versions de TSSuBERT (présenté en section 3) reportées dans le tableau 6.4 qui présente les résultats. Il convient de noter que les résultats donnés dans le tableau 6.4 ne tiennent pas compte des jours sans résumé de référence associé (s'ils existent, ces cas n'apparaissent par définition qu'au début d'un flux). Considérer ces jours n'a aucun sens dans le cas d'une taille a priori fixe pour les résumés de sortie. Nous présentons dans figure 6.5 un exemple de résumé généré par l'approche TSSuBERT pour l'évènement *Sismo Ecuador 2016*. Pour le lecteur intéressé, d'autres exemples sont mis en évidence sur notre Github [4].

TABLEAU 6.3 – Spécification des différentes versions du modèle TSSuBERT par rapport à la taille des résumés et du jeu d'apprentissage utilisé.

Nom du modèle	Jeu d'apprentissage	Taille maximale par incrément de temps
TES TSSuBERT-F	TES 2012-2016	Taille des résumés de référence
TES TSSuBERT	TES 2012-2016	20 tweets
ISSt TSSuBERT-F	ISSumet	Taille des résumés de référence
ISSt TSSuBERT	ISSumet	20 tweets

Nous considère les événements qui évoluent dans le temps. Ainsi, certains chiffres rapportés dans les résumés de référence peuvent évoluer d'un jour à l'autre en fonc-

4. `https://github.com/AlexisDusart/TSSuBERT`

tion des informations disponibles à ce moment-là (par exemple, le nombre de blessures, de décès,...). Quelle que soit la collection considérée, les approches de résumés sont comparées aux valeurs donnés dans les résumés de référence de la fenêtre de temps considérée. Ces valeurs sont particulièrement valables dans les résumés de référence de TES 2012-2016. En effet, les résumés du portail Wikipedia Current Events doivent être liés à des sources fiables [5]. Dans tous les cas, les approches qui ne rapportent pas ces valeurs sont pénalisées.

Il convient également de noter qu'avec l'évaluation et les résumés de référence actuels, les modèles qui ne mettent pas à jour les informations sont pénalisés. Le résumé incrémental peut donc ajouter ce qui pourrait être considéré comme de la redondance. Au contraire, nous considérons que ces informations permettent au lecteur de suivre en détail ce qui s'est réellement passé lors de l'événement. Comme nous pouvons le voir dans l'exemple de la figure 5.1, le nombre de morts entre le jour 17/04/2016 et le jour 18/04/2016 a augmenté de 262 à 270. C'est ainsi que l'information a été publiée à ce moment-là.

4.2.1 *Jeu de données TES 2012-2016*

Une première observation est que le modèle TSSuBERT surpasse significativement les approches de l'état de l'art sur la métrique ROUGE lorsqu'il suit leur exigence de fixer la taille maximale des résumés de sortie à la taille du résumé de référence correspondant, qu'il ait été appris sur TES 2012-2016 ou ISSumSet (TES TSSuBERT-F et ISSt TSSuBERT-F, lignes 7 et 9). Cependant, cela ne reflète pas, à notre avis, un comportement réaliste dans lequel la taille du résumé est inconnue. Les résultats sur un protocole plus réaliste sont donc présentés lignes 8 et 10 (TES TSSuBERT et ISSt TSSuBERT). Dans ce cas, nous indiquons simplement un nombre maximum de tweets qui peuvent être ajoutés à chaque fenêtre de temps. Cette limite est introduite pour des raisons d'efficience et fixée à 20 tweets (voir colonne 8). Dans ce cas, le modèle TSSuBERT est capable d'obtenir une taille de résumé proche de celle du résumé de référence sans la connaître. Une deuxième observation intéressante est donc que le comportement normal de TSSuBERT obtient toujours de meilleurs résultats que les approches de l'état de l'art. De plus, d'un point de vue sémantique (scores COS Embed, colonne 7), les résumés de TSSuBERT sont plus proches du résumé de référence que ceux générés par la *baseline* SEMCOWTS basée sur les relations sémantiques. Cela signifie que notre approche produit des résumés plus informatifs que les approches de l'état de l'art, malgré la différence de vocabulaire entre les résumés générés composés de tweets et les résumés de référence

5. `https://en.wikipedia.org/wiki/Wikipedia:How_the_Current_events_page_works`

6. Étant donné que la distribution n'était pas gaussienne pour tous les modèles utilisant le test de Shapiro-Wilk, nous avons décidé de ne pas présenter le test t de Student et d'utiliser le test de Wilcoxon, comme suggéré par Urbano *et al.* (2013).

TABLEAU 6.4 – Résultats sur les jeux de données TES 2012-2016, ISSumSet, et sur la collection de Rudra *et al.* (2018b). Les résultats avec * sont statistiquement significatifs et différents de l'état de l'art (COWTS-SEMCOWTS-SCC) (test de Wilcoxon[6], * $p-value < 0{,}05$, ** $p-value < 0{,}01$). Les scores en gras mettent en évidence le meilleur modèle pour la métrique considérée (sans tenir compte des résumés Oracle pour les collections TES 2012-2016 et ISSumSet). Les spécifications des différentes versions d'apprentissage du modèle TSSuBERT sont décrites dans le tableau 6.3 (le même schéma de nommage est utilisé pour les versions du modèle MP-HCNN). La colonne de gauche représente le jeu de test utilisé.

	Modèle	ROUGE-1 Macro		ROUGE-2 Macro		COS Embed Macro	Max Taille par incrément de temps (p.t.i.), i.e., par jour
		P	F	P	F		
TES 2012-2016	Oracle R-2	0,566**	0,554**	0,366**	0,357**	0,903**	Taille G.S.
	Oracle COS	0,581**	0,591**	0,241**	0,245**	0,893**	Taille G.S.
	Randoms	0,107**	0,101	0,015**	0,014**	0,608**	Taille G.S.
	COWTS	0,051	0,054	0,002	0,002	0,655	Taille G.S.
	SEMCOWTS	0,064	0,068	0,002	0,002	0,686	Taille G.S.
	SCC	0,046	0,049	0,001	0,001	0,619	Taille G.S.
	TES TSSuBERT-F	**0,134****	0,114**	**0,023****	**0,021****	0,676*	Taille G.S.
	TES TSSuBERT	0,095**	0,111**	0,014**	0,018**	0,741**	20 tweets
	ISSt TSSuBERT-F	0,133**	**0,124****	0,022**	**0,021****	0,656**	Taille G.S.
	ISSt TSSuBERT	0,068**	0,099**	0,009**	0,013**	**0,749****	20 tweets
	TES MP-HCNN-F	0.128**	0.110**	0.012**	0.011**	0.595**	Taille G.S.
	TES MP-HCNN	0.051**	0.083	0.007**	0.011**	0.698*	20 tweets
	ISSt MP-HCNN-F	0.108	0.103**	0.016**	0.015**	0.620**	Taille G.S.
	ISSt MP-HCNN	0.044**	0.074	0.005**	0.009**	0.683*	20 tweets
ISSumSet	Oracle R-2	1,000**	1,000**	1,000**	1,000**	1,000**	Taille G.S.
	Oracle COS	1,000**	1,000**	1,000**	1,000**	1,000**	Taille G.S.
	Randoms	0,331	0,322**	0,122	0,119	0,832**	Taille G.S.
	COWTS	0,345	0,372	0,120	0,130	0,870	Taille G.S.
	SEMCOWTS	0,337	0,363	0,118	0,127	0,869	Taille G.S.
	SCC	0,315	0,339	0,099	0,106	0,852	Taille G.S.
	TES TSSuBERT-F	0,362*	0,357**	0,181**	0,179**	0,849	Taille G.S.
	TES TSSuBERT	0,275**	0,256**	0,110	0,113**	0,854	20 tweets
	ISSt TSSuBERT-F	**0,425****	**0,382**	**0,207****	**0,186****	0,857	Taille G.S.
	ISSt TSSuBERT	0,308**	0,288**	0,129**	0,122	**0,880**	20 tweets
	TES MP-HCNN-F	0.301	0.297**	0.104*	0.103	0.818**	Taille G.S.
	TES MP-HCNN	0.230**	0.261**	0.087	0.103	0.853	20 tweets
	ISSt MP-HCNN-F	0.356**	0.323**	0.144**	0.130	0.848	Taille G.S.
	ISSt MP-HCNN	0.232**	0.277**	0.093	0.110*	0.863	20 tweets
Collection de Rudra *et al.* (2018b)	Randoms	0,349**	0,351**	0,158**	0,161**	0,870**	+ 200 words p.t.i.
	COWTS	0,423	**0,459**	0,221	**0,242**	0,931	+ 200 words p.t.i.
	SEMCOWTS	0,420	0,456	0,214	0,234	0,930	+ 200 words p.t.i.
	SCC	0,390	0,422	0,196	0,215	0,924	+ 200 words p.t.i.
	TES TSSuBERT-F	0,396	0,404**	0,181*	0,188**	0,930	+ 200 words p.t.i.
	TES TSSuBERT	0,387*	0,412*	0,180*	0,192**	**0,935**	20 tweets
	ISSt TSSuBERT-F	**0,478**	0,432	**0,253**	0,227	**0,935**	+ 200 words p.t.i.
	ISSt TSSuBERT	**0,478**	0,434	0,252	0,227	**0,935**	20 tweets
	TES MP-HCNN-F	0.362**	0.362**	0.174**	0.177**	0.888**	+ 200 words p.t.i.
	TES MP-HCNN	0.350**	0.374**	0.175**	0.188**	0.894**	20 tweets
	ISSt MP-HCNN-F	0.427	0.404**	0.201	0.193**	0.918*	+ 200 words p.t.i.
	ISSt MP-HCNN	0.427	0.410**	0.205	0.200*	0.920*	20 tweets
Flux de TES 2012-2016 GS de Rudra *et al.* (2018b)	Randoms	0.226**	0.227**	**0.036**	0.036	0.803**	Taille G.S. (+ 800 words p.t.i.)
	COWTS	**0.270**	**0.297**	0.030	0.033	0.921	Taille G.S. (+ 800 words p.t.i.)
	SEMCOWTS	0.254	0.282	0.034	**0.037**	**0.923**	Taille G.S. (+ 800 words p.t.i.)
	SCC	0.244	0.265	0.025	0.027	0.897	Taille G.S. (+ 800 words p.t.i.)

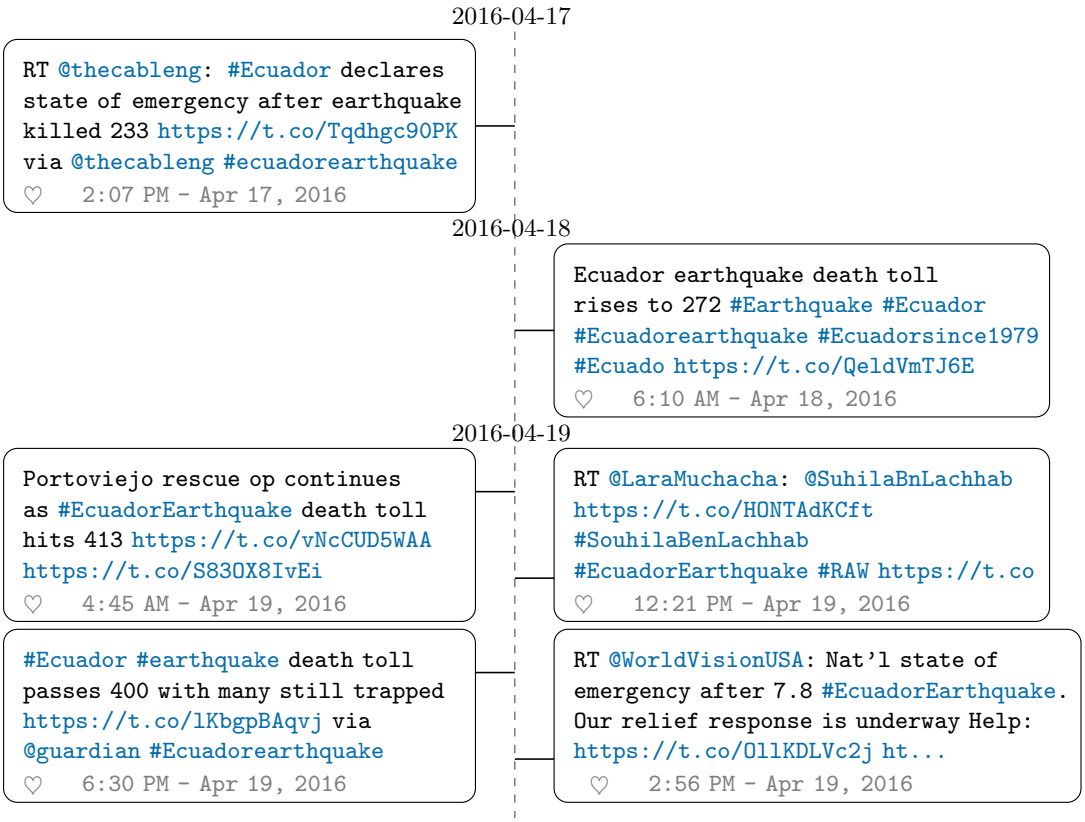


FIGURE 6.5 – Exemple de résumé généré par l'approche TSSuBERT (variante TES TSSuBERT-F) concernant l'évènement « Sismo Ecuador 2016 » (extrait correspondant à l'exemple dans la figure 6.6). Pour un jour donné, les tweets sont classés par score d'importance décroissante.

dans une forme journalistique. L'évaluation manuelle (section 4.4) confirme cette conclusion.

4.2.2 *Jeu de données ISSumSet*

Sur cet ensemble de données, le meilleur score est obtenu par une version de TSSuBERT pour chacune des mesures évaluées. Les versions ISSt TSSuBERT et TSSuBERT-F obtiennent de meilleurs scores que les versions TES TSSuBERT et TSSuBERT-F. Il convient de noter que le score d'importance a été fixé à 0,0 pour les versions TES TSSuBERT et TSSuBERT-F, car un score d'importance de 0,2 a conduit à de mauvais résultats en termes de nombre de tweets récupérés, n'atteignant pas, et de loin, la longueur des résumés de référence. De plus, presque toutes les versions limitées aux longueurs des résumés de référence sont meilleures que les *baselines* de l'état de l'art en ce qui concerne la précision et la F-mesure. En termes

de similarité d'un point de vue sémantique, la version ISSt TSSuBERT obtient le meilleur score, tandis que les autres versions sont proches de l'état de l'art.

4.2.3 *Jeu de données de Rudra* et al. *(2018b)*

D'une part, le modèle TSSuBERT entraîné sur TES 2012-2016 (TES TSSuBERT) obtient des résultats inférieurs à ceux de l'état de l'art pour chaque mesure de ROUGE, mais reste sémantiquement proche d'eux. Il est important de noter que pour ces deux versions, nous avons dû fixer le score d'importance à 0,0, car un score d'importance fixé à 0,2 conduisait à des résumés vides. D'autre part, le modèle TSSuBERT entraîné sur le jeu de données ISSumSet n'est pas significativement plus faible pour les mesures ROUGE ou sémantiques, et obtient un meilleur score en termes de précision.

4.2.4 *Discussion*

Comme le montre le tableau 6.4, le modèle TSSuBERT surpasse l'état de l'art dans presque tous les cas, c'est-à-dire dans presque toutes les configurations, métriques et ensembles de données. Cependant, certains résultats méritent d'être clarifiés. En utilisant la collection de Rudra *et al.* (2018b), les versions TES de notre modèle atteignent les résultats de l'état de l'art, bien que légèrement inférieurs, tandis que les versions ISSt les surpassent. Il en va de même avec le jeu de données ISSumSet, bien que les versions TES de notre modèle obtiennent de meilleurs résultats. Cela peut être dû aux différences entre les résumés de référence constituant les ensembles d'entraînement utilisés. En effet, les versions ISSt entraînées sur ISSumSet pour lequel les résumés de référence sont en moyenne plus de 2 fois plus longs pour un même incrément (37 tokens vs 14 tokens) que TES 2012-2016 sur lequel ont été entraînées les versions TES. Cela signifie que le modèle apprend à identifier environ 2 fois plus d'informations lorsqu'il est entraîné sur ISSt que sur TES 2012-2016. De plus, les informations des résumés de référence de TES 2012-2016 diffèrent de ceux de ISSumSet et du jeu de données de Rudra *et al.* (2018b) puisque les premiers sont construits à partir d'une ressource externe au flux de tweets (en utilisant le WCEP) alors que les autres sont construites à partir du flux de tweets.

Un autre enseignement mis en évidence est que la méthode Randoms représente une *baseline* forte bien qu'*a priori* naïve, qui surpasse étonnamment les *baselines* de l'état de l'art sur le jeu de données TES 2012-2016. Une explication peut être que les termes des résumés de référence apparaissent dans de nombreux tweets. Nous pensons qu'une telle *baseline* devrait alors être systématiquement rapportée dans les expériences futures. En ce qui concerne les résumés Oracles, les résultats montrent qu'il y a encore une grande marge d'amélioration. Enfin, il faut noter que des ex-

périences, non rapportées ici, réalisées en entraînant le modèle sans le contexte des tweets ont conduit à de très mauvais résultats (des résumés vides étaient retournés même si le score d'importance était fixé à 0,0), soutenant ainsi son rôle fondamental.

Comme nous pouvons le voir dans le tableau 6.4, les approches COWTS, SEMCOWTS et SCC ont obtenu des résultats médiocres sur le jeu de données TES 2012-2016 et des résultats inférieurs à ceux de Randoms pour certaines métriques du jeu de données ISSumSet. Nous pouvons distinguer 3 explications différentes :

— Pour la collection ISSumSet, bien que les approches COWTS, SEMCOWTS et SCC obtiennent des résultats décents selon ROUGE-1, elles performent de manière similaire aux Randoms pour ROUGE-2. Cela peut être expliqué par le fait que ces approches ont été optimisées pour ROUGE-1 qui est basé sur les unigrammes. Les auteurs ont en effet considéré que, dû à la nature informelle des tweets, ROUGE-1 seulement était nécessaire (ce qui est questionnable).
— Deuxièmement, toujours pour la collection ISSumSet, une analyse plus approfondie montre que les résultats ne sont pas bons selon la précision, tandis qu'ils sont meilleurs que les Randoms sur la F-mesure. Cela signifie que la rappel est bon mais que les approches manquent de précision. Cela peut être expliqué par le classifieur appris et utilisé par ces approches. Il catégorise les tweets entre *situationnel* et *non situationnel*. Les tweets considérés situationnels sont de deux types : *Status updates* et *Help relief operations*. Voici des exemples :
  — **Status updates** : THagupit : typhoon now making landfall in eastern samar, with winds of 175 to 210 kph, and rainfall up to 30mm per hour ; SHShoot : state police are responding to a report of a shooting at an elementary school in newtown [url]
  — **Help relief operations** : UFlood : call bsnl toll-free numbers 1503, 09412024365 to find out last active location of bsnl mobiles of missing persons in uttarakhand ; HDBlast : blood banks near dilsuknagar slms 040-64579998 kamineni 39879999 hima bindu 9246373536 balaji ; SHShoot : If you want to donate blood, call 1-800-RED CROSS. @CTRedCross @redcrossbloodct

  Les tweets correspondants au type *Help relief operations* ne sont pas utiles pour être inclus dans des résumés comme ils sont définis dans la collection ISSumSet. Les approches COWTS, SEMCOWTS et SCC les considèrent cependant pertinents et les incluent dans les résumés qu'elles génèrent.
— Troisièmement, concernant la collection TES 2012-2016, nous pensons que les approches COWTS, SEMCOWTS et SCC échouent à produire des résumés qui peuvent être comparés à ceux du Wikipedia Current Event Portal. C'est probablement dû à la différence de vocabulaire. En effet, nous avons conduit des expérimentations supplémentaires comme suit. La collection de Rudra *et al.* (2018b) et la collection TES 2012-2016 ont deux évènements en commun (Ty-

phoon Hagupit et Nepal Earthquake). Ainsi, pour ces deux évènements nous pouvons produire des résumés à partir du flux de tweets de TES 2012-2016 et les comparer aux résumés de référence de Rudra *et al.* (2018b) (qui sont composés de tweets et non pas de texte dans une forme journalistique). Les résultats sont reportés dans le tableau 6.4. Nous pouvons voir que les méthodes de l'état de l'art sont significativement meilleures que les Randoms pour mes métriques ROUGE-1 et Cos Embed.

4.3 *Variantes testées*

En plus du modèle présenté dans les paragraphes précédents, nous avons testé plusieurs variantes pour notre modèle. Dans un premier temps, nous avons testé le modèle sans la fréquence des *tokens*, uniquement à partir du modèle de langue neuronal. Dans un second temps nous avons essayé d'ajouter au modèle des caractéristiques spécifiques aux tweets.

4.3.1 *Ablation de la fréquence des tokens*

La prédiction d'importance d'un tweet est composée de deux parties : (i) le tweet, représenté à l'aide d'un modèle de langue pré-entraîné, (ii) le contexte du tweet, représenté par la fréquence des *tokens* du flux jusqu'à la fenêtre de temps du tweet incluse. Pour comprendre l'importance d'utiliser ces deux composantes nous pouvons évaluer les modèles utilisant uniquement l'une des deux. D'un côté, évaluer le modèle utilisant uniquement le contexte n'a pas de sens puisque l'ensemble des tweets d'une même fenêtre de temps partageraient la même représentation, et seraient donc non différenciables. D'un autre côté, nous évaluons le modèle utilisant uniquement le tweet. Intuitivement, utiliser seulement le tweet sera insuffisant car le modèle ne sera pas en mesure de distinguer les mêmes tweets qui apparaissent dans des fenêtres de temps différentes. Lors de l'évaluation, le modèle utilisant uniquement le tweet, sans la fréquence des *tokens* dans le flux de tweets, ne prédit que des scores d'importance de 0,0 pour tous les tweets. Ce comportement nous conforte dans l'utilité de l'ajout d'un contexte des fréquences des *tokens* du flux de tweets pour représenter le contexte du flux.

4.3.2 *Ajout de caractéristiques des tweets*

Nous avons essayé de tirer avantage des caractéristiques spécifiques aux tweets en les ajoutant au modèle. Nous avons ainsi testé les caractéristiques suivantes :

— Utilisateur vérifié ou non (1 ou 0).

— Nombre de followers (d'abonnés) de l'utilisateur.
— Nombre de followees (d'abonnements) de l'utilisateur.
— Nombre de listes dans lesquels l'utilisateur est membre.
— Nombre de favoris de l'utilisateur (nombre de tweets que l'utilisateur a aimé depuis la création du compte).
— Nombres de status émis par l'utilisateur (les retweets sont comptés).
— Nombre de fois que le tweet a été retweeté.
— Nombre de fois que le tweet a été aimé.

Après évaluation du modèle, nous avons observé que les résultats sont bien en deçà des résultats sans l'ajout des caractéristiques. Il semble que l'ajout des caractéristiques perturbe le modèle dans la compréhension du contexte du flux. Nous ne pouvons cependant pas tirer de conclusion forte sur l'efficacité ou non de l'ajout de ces caractéristiques au modèle. En effet, nous avons récupéré ces données au moment de notre étude, pour laquelle des années sont passées depuis la publication du tweet. Les caractéristiques ont ainsi évolué entre-temps et ne sont plus représentatives de l'utilisateur (un utilisateur non vérifié peut être devenu vérifié) et du tweet (un tweet peut voir son nombre de mentions *j'aime* augmenter ou diminuer continuellement).

4.3.3 *Substitution de la prédiction d'importance*

Nous avons évalué séparément la partie de prédiction du score d'importance du modèle TSSuBERT, afin d'évaluer son apport. Nous avons considéré le modèle supervisé MP-HCNN proposé par Rao *et al.* (2019). Ce modèle n'a pas été proposé comme un modèle de résumé mais a été utilisé comme *baseline* pour résumer dans (Yang *et al.*, 2020). Ce modèle retourne un score de pertinence pour un tweet à partir d'une requête et peut ainsi être utilisé pour la prédiction d'importance. Nous avons ensuite créé les résumés en utilisant la partie de sélection des tweets du modèle TSSuBERT. Nous avons utilisé le code fourni par les auteurs [7], avec le titre de l'évènement comme requête. Les Idfs ont été calculés à partir des 80 millions de tweets du jeu de données TES 2012-2016 [8]. Nous avons évalué le modèle automatiquement comme présenté dans les paragraphes suivants (section 4.1) et reportés les résultats dans le tableau 6.4 (les quatre dernières lignes de chaque jeu de données, les modèles dont le nom contient MP-HCNN). Nous pouvons voir que le modèle MP-HCNN performe moins bien que le modèle TSSuBERT pour le résumé en ce qui concerne les métriques évaluées pour toutes les configurations. Cela montre l'utilité de la partie de prédiction d'importance du modèle TSSuBERT.

7. `https://github.com/jinfengr/neural-tweet-search`
8. Les Idfs utilisés par les auteurs pour leur modèle ne sont plus accessibles. Nous les avons contacté mais n'avons pas reçu de réponse.

4.4 *Évaluation manuelle*

Les évaluations automatiques présentent des limites (voir section 4.1.2 du chapitre 3), notamment pour évaluer l'informativité des résumés. Nous avons donc complété les évaluations réalisées par une évaluation plus qualitative basée sur des évaluations manuelles.

4.4.1 *Cadre expérimental*

Inspirés par (Mani *et al.*, 2002; Clarke et Lapata, 2010; Narayan *et al.*, 2018b), nous avons construit des questions dont les réponses apparaissent dans les résumés de référence. Un exemple des questions construites ainsi que leurs réponses associées est donné dans figure 6.6. Deux évaluateurs expérimentés qui ont participé à plusieurs campagnes d'évaluation telles que TREC [9] et CLEF [10], ont évalué manuellement chaque résumé en vérifiant dans quelle mesure ils pouvaient répondre à chaque question. Pour chaque fenêtre de temps, il a été demandé aux annotateurs de répondre aux questions associées ainsi qu'aux questions des fenêtres de temps précédentes si nécessaire (lorsque l'information n'était pas présente dans les résumés précédents ou lorsque l'information était mise à jour dans le résumé actuel, par exemple, le nombre de décès après une catastrophe naturelle). Une réponse est notée comme suit : 1, 0,5 et 0 pour, respectivement, une réponse juste, partiellement juste et fausse. En outre, pour chaque fenêtre de temps, nous avons demandé aux évaluateurs d'écrire des informations qu'ils ont apprises dans les résumés mais qui n'ont pas été abordées dans nos questions. En raison du travail fastidieux de l'évaluation manuelle, nous n'avons pas effectué l'évaluation sur tous les jeux de données et tous les modèles. Nous avons utilisé le jeu de données TES 2012-2016 et nous avons choisi 7 évènements différents de types variés à évaluer (c'est-à-dire, élection, attaque par explosion, prise d'otages, typhon, fusillade, tremblement de terre et sport). Ce nombre d'évènements est similaire à celui utilisé dans (Clarke et Lapata, 2010). Nous avons demandé d'évaluer les Oracles pour visualiser les bornes maximales, les modèles de l'état de l'art et les versions TES TSSuBERT, car ils ont obtenu de meilleurs scores de précision que les versions ISSt TSSuBERT sur cet ensemble de données. Les approches COWTS, SEMCOWTS et SCC produisent des résumés qui ne contiennent pas certains *tokens* spécifiques à Twitter tels que les mentions, les hashtags et les URLs. Afin de comparer équitablement nos approches avec les trois susmentionnées, les annotateurs ont évalué deux versions différentes de nos résumés : avec et sans ces *tokens*, respectivement désignés comme résumés bruts et filtrés dans ce qui suit.

9. `https://trec.nist.gov/`
10. `https://www.clef-initiative.eu/`

Questions :
2016-04-17 :
Question 1 : How many dead ?
Question 2 : How many injured ?

2016-04-18 :
Question 1 : How many dead ?

2016-04-19 :
Question 1 : How many dead ?
Question 2 : How many injured ?
Question 3 : How many people are missing ?
Question 4 : What did President Rafael Correa state about the reconstruction of the country after this disaster ?

gold standard summary and associated answers (Q1 to Q4 tags) :
2016-04-17 : <Q2><Q1>The death toll from Saturday night's earthquake in Ecuador rises to 262</Q1> with more than 2,500 people injured</Q2>.
2016-04-18 : Aid starts to flow in after <Q1>an earthquake kills over 270 people in Ecuador</Q1>.
2016-04-19 : <Q3><Q1>The death toll from Saturday's earthquake has risen to at least 480</Q1> with 1,700 missing</Q3>. <Q2>Another 2,500 have been injured</Q2>. <Q4>President Rafael Correa states it is the worst disaster in Ecuador in seven decades, and the reconstruction will have a 'huge economic impact' on the country</Q4>.

FIGURE 6.6 – Exemple de questions-réponses pour l'évaluation manuelle concernant l'évènement « Sismo Ecuador 2016 » pour la période du 17-04-2016 au 19-04-2016.

4.4.2 *Résultats et discussion*

Les résultats sont présentés dans la tableau 6.5. L'évaluation manuelle confirme la grande qualité des résumés Oracles (ils permettent de répondre correctement à près de trois quarts des questions). En considérant les résumés filtrés (lignes 3 à 7), on peut voir que les approches TSSuBERT surpassent toujours les *baselines* de l'état de l'art. Cette conclusion est valable que l'on considère les collections dont les résumés de référence sont composées de tweets ou la collection TES 2012-2016 pour laquelle les résumés de référence sont sous une forme journalistique (ce qui entraîne une inadéquation du vocabulaire). Enfin, le fait de conserver les hashtags, les mentions et les retweets (résumés bruts, lignes 10-11) semble apporter des informations supplémentaires utiles. En outre, il convient de noter qu'en moyenne, les évaluateurs ont appris plus d'informations supplémentaires (c'est-à-dire des informations non abordées dans les résumés de référence) avec les résumés TSSuBERT (1,1

à 1,9 information supplémentaire par évènement) qu'avec les résumés de l'état de l'art (0,4 à 1,6).

Toutes ces évaluations qualitatives ainsi que les résultats détaillés sont disponibles sur notre page Github [11].

TABLEAU 6.5 – Résultats de l'évaluation qualitative sur le jeu de données TES 2012-2016. Les colonnes nommées A, B, C, D, E, F et G représentent respectivement les évènements suivants : « Charlie Hebdo 2015 », « Superbowl 2012 », « Sydney Siege 2014 », « Boston Marathon Bombing 2013 », « Indyref 2014 », « Typhoon Hagupit 2014 » et « Sismo Ecuador 2016 ». Les scores sont exprimés en termes de ratio de bonnes réponses. Les modèles désignés par « brut » renvoient des tweets entiers tandis que les modèles désignés par « filtré » suppriment les tokens tels que @, #, URLs (d'une manière similaire à COWTS, SEMCOWTS, et SCC). Les meilleurs scores par évènement sont indiqués en gras (sans tenir compte des Oracles).

Model	A	B	C	D	E	F	G	Mean
Oracle Rouge-2 (filtré)	0,474	1,000	0,500	0,667	0,909	0,818	0,629	0,714
Oracle Cosine (filtré)	0,707	0,750	0,600	0,500	1,000	0,818	0,652	0,718
COWTS	0,009	0,000	0,000	0,000	0,091	0,000	0,197	0,042
SEMCOWTS	0,207	0,000	0,000	0,000	0,182	0,000	0,144	0,076
SCC	0,052	0,000	0,000	0,000	0,091	0,000	0,106	0,036
TES TSSuBERT (filtré)	**0,276**	0,125	0,200	**0,111**	**0,636**	0,000	**0,394**	0,249
TES TSSuBERT-F (filtré)	0,172	0,000	0,000	**0,111**	0,273	0,000	0,371	0,132
Oracle Rouge-2 (brut)	0,474	1,000	0,500	0,667	0,909	0,818	0,629	0,714
Oracle Cosine (brut)	0,707	0,750	0,600	0,500	1,000	0,818	0,652	0,718
TES TSSuBERT (brut)	**0,276**	**0,375**	**0,350**	**0,111**	**0,636**	**0,545**	**0,394**	**0,384**
TES TSSuBERT-F (brut)	0,172	0,083	0,150	**0,111**	0,455	0,364	0,371	0,244

4.5 *Évaluation de l'efficience*

Nous avons évalué le gain d'efficience du modèle TSSuBERT par rapport aux autres modèles [12]. Nous montrons le temps d'exécution des différents modèles sur TES 2012-2016, ISSumSet et le jeu de données de Rudra *et al.* (2018b), respective-

11. `https://github.com/AlexisDusart/TSSuBERT/`

12. Tous les modèles ont été évalués avec la même configuration (CPU Xeon 2640 V4 et GPU GTX 1080 TI). Les expériences présentées dans ce mémoire ont été réalisées en utilisant la plateforme OSIRIM qui est administrée par l'IRIT et soutenue par le CNRS, la région Midi-Pyrénées, le gouvernement français, et le FEDER (voir `http://osirim.irit.fr`).

ment dans les figures 6.7, 6.8, et 6.9. Comme l'affirment les auteurs, l'approche SCC a un temps d'exécution plus élevé que COWTS (Rudra *et al.*, 2018b). Les temps d'exécution de SCC ne sont donc pas reportés. L'approche MP-HCNN nécessite les URLs originales (et non les URLs raccourcies). Comme il faut du temps pour attendre la réponse du serveur externe afin d'obtenir le lien original d'une URL, et comme cette méthode n'est pas adaptée aux 80 millions de tweets du jeu de données TES 2012-2016, nous les avons calculés en utilisant du *multi-threading*. Comme nous ne savons pas comment ils ont été calculés par les auteurs, nous n'avons pas ajouté ce temps d'exécution au temps d'exécution total de l'approche. Nous pouvons constater que l'approche TSSuBERT est plus rapide que les autres approches sur les trois jeux de données. Plus précisément, sur le plus grand jeu de données (TES 2012-2016), TSSuBERT et MP-HCNN, qui génèrent des résumés de manière incrémentale et n'utilisent pas le flux entier comme entrée pour une fenêtre de temps, ont un temps d'exécution inférieur aux approches de l'état de l'art. Sur les petits jeux de données (ISSumSet et celui de Rudra *et al.* (2018b)), l'approche MP-HCNN a cependant les temps d'exécution les plus élevés. Cela peut s'expliquer par sa complexité à calculer le score d'un tweet.

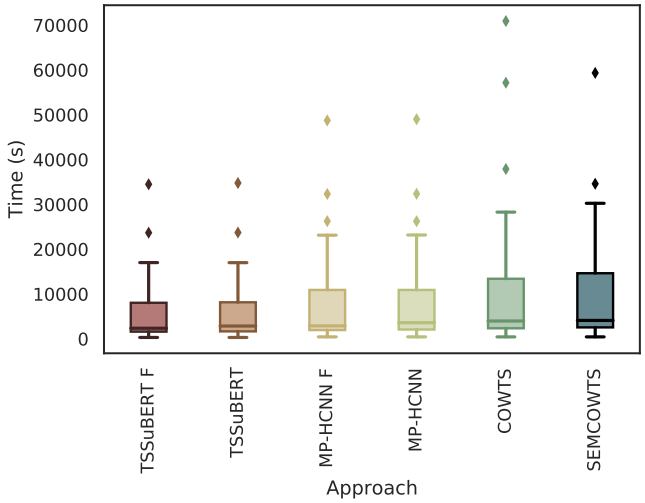


FIGURE 6.7 – Temps d'exécution en secondes des différentes approches évalués sur les évènements du jeu de données TES 2012-2016.

5 Bilan

Dans ce chapitre, nous avons proposé un modèle neuronal pour le résumé extractif incrémental de flux de tweets, nommé TSSuBERT. TSSuBERT surpasse l'état de l'art en termes d'efficacité et d'efficience lors des expérimentations menées. Plusieurs

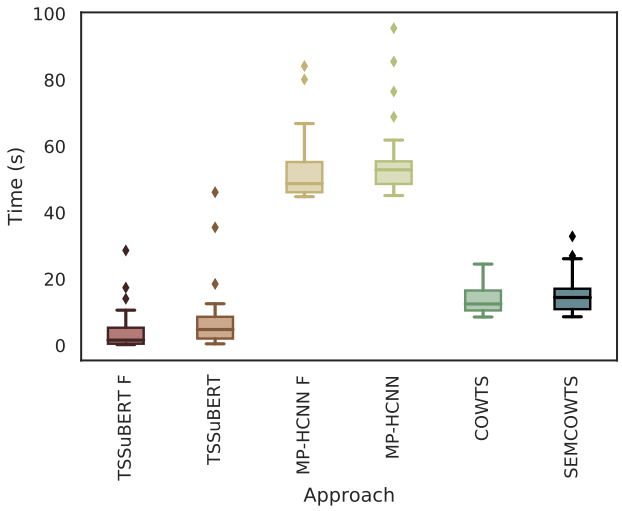


FIGURE 6.8 – Temps d'exécution en secondes des différentes approches évalués sur les évènements du jeu de données ISSumSet.

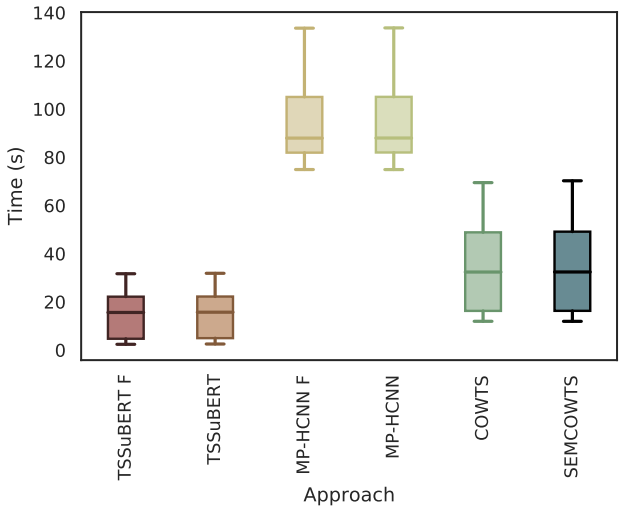


FIGURE 6.9 – Temps d'exécution en secondes des différentes approches évalués sur les évènements du jeu de données de Rudra *et al.* (2018b).

caractéristiques originales de ce modèle peuvent être soulignées : (i) des modèles de langue neuronaux pré-entraînés ainsi que le contexte des tweets sous la forme de fréquences du vocabulaire sont utilisés dans le modèle, (ii) le modèle fixe automatiquement la taille des résumés de sortie ; cela correspond à un cas d'utilisation plus réaliste que celui des modèles de l'état de l'art : ces derniers produisent toujours la même taille de résumés, que ce soit utile ou non, et (iii) TSSuBERT est une méthode incrémentale qui ne nécessite pas de ré-évaluer tous les documents à chaque

incrément de temps, ce qui entraîne un gain d'efficience. Le modèle est capable de traiter des flux de tweets particulièrement volumineux. Nous distribuons sur notre Github [13] :

— le code pour reproduire le modèle, la prédiction d'importance et la sélection de tweets, ainsi que les poids des modèles entraînés dans les expérimentations présentées,
— les résumés générés par les méthodes de l'état de l'art présentées, ainsi que ceux générés par les différentes versions du modèle TSSuBERT, et les évaluations manuelles qualitatives (questions et réponses, ainsi que les résultats pour les méthodes précédemment mentionnées).

Comme évoqué dans l'état de l'art, nous avons identifié que les modèles de langue neuronaux sont aujourd'hui largement utilisés dans les approches de résumé automatique « traditionnelles », tandis qu'ils sont quasiment inexistants dans les approches de résumé automatique de flux de tweets. Une des raisons que nous avons identifié à cela est la taille de l'entrée des documents à résumer. En effet, les approches basées sur les modèles de langue neuronaux sont limitées à résumer des documents de quelques centaines ou milliers de mots, ce qui n'est pas suffisant pour résumer un flux de tweets. Dans ce chapitre nous avons présenté une approche pour profiter de ces modèles de langue neuronaux pour résumer un flux de tweets. Cette approche représente chaque tweet à l'aide des modèles de langue neuronaux, et représente le contexte d'un tweet à l'aide de la fréquence des *tokens* qui composent le flux. Dans le chapitre suivant, nous souhaitons profiter des modèles de langue neuronaux pour également représenter le contexte d'un tweet. Pour ce faire, nous explorons les représentations pré-établies par les modèles de langue neuronaux.

13. `https://github.com/AlexisDusart/TSSuBERT`

7

EXPLORATION DES REPRÉSENTATIONS TEXTUELLES PRÉ-ÉTABLIES DANS LE CONTEXTE DU RÉSUMÉ AUTOMATIQUE DE FLUX DE TWEETS

Sommaire

Introduction

Dans le chapitre précédent, nous avons proposé une approche pour tirer profit des modèles de langue neuronaux pré-entraînés. Nous avons ajusté ces modèles de langue pré-entraînés sur notre tâche de résumé automatique de flux de tweets. Nous les avons combiné à la fréquence d'apparition des *tokens* dans le flux de tweets afin de modéliser le contexte d'un tweet.

Dans ce chapitre, nous étudions une manière différente d'utiliser les modèles de langue neuronaux pour le résumé de flux de tweets. Notre objectif est de pouvoir profiter des modèles de langue neuronaux pour représenter un tweet, mais également son contexte, le flux de tweets. Cependant, les modèles de langue neuronaux ne sont pas trivialement utilisables pour représenter un volume de données aussi large qu'un flux de tweets. Dans ce chapitre, nous explorons différentes pistes d'utilisation de représentations textuelles pré-établies dans le contexte du résumé automatique de flux de tweets. L'apprentissage *a posteriori* des modèles neuronaux pré-entraînés avec une couche de sortie adaptée à la tâche visée a montré ses performances sur de nombreuses tâches du Traitement Automatique du Langage naturel (TAL). Cependant, les performances des représentations générées par ces modèles, sans apprentissage supplémentaire, ne sont pas encore bien comprises, particulièrement pour des données bruitées comme des tweets (Wang *et al.*, 2020a).

Il est à noter que dans ce chapitre les représentations que nous manipulons sont des représentations denses, et, pour une même représentation, un tweet est projeté selon un vecteur de taille spécifique au modèle utilisé, entre 384 et 768 dans nos expériences. Aussi, afin d'alléger la lecture, lorsque nous parlons d'opérations (comme une moyenne) sur les mots ou les tweets, nous sous-entendons par rapport à la représentation dense associée au mot ou au tweet. De même, pour alléger la lecture, lorsque nous évoquons la similarité entre deux représentations, nous faisons référence à la similarité Cosinus.

La suite de ce chapitre est organisée comme suit. Dans un premier temps, nous développons le contexte, les motivations, et les objectifs à ce travail. Dans un deuxième temps, nous présentons les réponses que nous apportons à ces objectifs. Finalement, nous concluons le chapitre.

1 Contexte et motivations

Bien que les modèles de langue neuronaux pré-entraînés s'inscrivent aujourd'hui dans l'état de l'art de nombreuses tâches du TAL, les représentations qu'ils génèrent

ne sont pas totalement comprises. Wang *et al.* (2020a) s'intéressent à ces représentations pour des tweets par l'évaluation de la qualité des *clusters* formés par chacune des représentations évaluées. Ils indiquent également que certaines représentations, comme celles générées par XLNet (Yang *et al.*, 2019) et USE (Cer *et al.*, 2018) sont plus performantes. Cependant, ils concluent que d'autres expérimentations sont à mener pour comprendre ces nouvelles représentations sur des données bruitées, générées par les utilisateurs. Ainsi, les expériences n'étant pas produites sur du résumé automatique, nous souhaitons évaluer ces représentations pour notre problématique. C'est pourquoi nous visons un premier objectif :

Objectif de recherche : Parmi les représentations textuelles pré-établies de l'état de l'art, laquelle est la plus appropriée pour le résumé automatique de flux de tweets ?

Comme mentionné par Kenter *et al.* (2016), obtenir une représentation d'une phrase par la simple moyenne des mots qui la composent s'est avéré être une solide *baseline* pour une multitude de tâches (Faruqui *et al.*, 2015; Gershman et Tenenbaum, 2015; Kenter et de Rijke, 2015). Aujourd'hui, les représentations fournies par les modèles de langue neuronaux pré-entraînés permettent d'obtenir des représentations de phrases et de documents. Une phrase est composée de mots. Un tweet est un document très court, souvent composé d'une seule phrase. Un flux de tweets est composé de tweets. Notre idée est donc d'explorer la possibilité de représenter un flux de tweets à partir de la moyenne de chaque tweet qui le compose. Nous traduisons cette problématique au travers de l'objectif suivant :

Objectif de recherche : La représentation d'un flux de tweets à partir de la moyenne des tweets qui composent le flux est-elle utilisable pour le résumé automatique de flux de tweets ?

Enfin, Iyyer *et al.* (2015) introduisent le modèle *Deep Averaging Network* (DAN). Ce modèle est utilisé pour la classification de phrases. La phrase présentée en entrée du modèle est un vecteur, qui est le vecteur moyen (sans pondération) de chacun des mots de la phrase. DAN est d'ailleurs utilisé pour générer une des deux versions de représentation du modèle USE (Cer *et al.*, 2018). Ces travaux justifient l'intérêt d'utiliser un modèle à partir d'une représentation moyenne. Ainsi, nous proposons de répondre à l'objectif suivant :

Objectif de recherche : Peut-on améliorer la représentation d'un flux de tweets créée à partir de la moyenne des tweets qui composent le flux à l'aide d'un modèle d'apprentissage automatique ?

Dans la suite du chapitre, nous présentons les expérimentations et résultats obtenus pour permettre de répondre à ces différents objectifs.

2 Évaluation des représentations textuelles pré-établies de l'état de l'art pour le résumé automatique de flux de tweets

Dans cette section, nous proposons de répondre au premier objectif de recherche énoncé dans la section précédente, à savoir :

Objectif de recherche : Parmi les représentations textuelles pré-établies de l'état de l'art, laquelle est la plus appropriée pour le résumé automatique de flux de tweets ?

Pour répondre à cet objectif, nous proposons la contribution suivante :

Méthodologie pour répondre à l'objectif de recherche : Nous proposons d'évaluer les représentations textuelles pré-établies de l'état de l'art par la construction de résumés Oracles pour chacune d'entre elles. Ainsi, nous obtenons une borne maximale atteignable pour chacune des représentations.

2.1 *Protocole d'évaluation*

Nous créons les résumés Oracles incrémentalement, guidés par la taille des résumés de référence. Inspirés par les travaux (Nallapati *et al.*, 2017; Liu et Lapata, 2019; Dou *et al.*, 2021), nous avons utilisé un algorithme *glouton* qui conserve les tweets maximisant la métrique évaluée. Ainsi, ces résumés Oracles sont construits comme suit, illustré en figure 7.1 :

1. Au départ le résumé Oracle est vide. Les tweets et le résumé de référence sont projetés selon le même espace de représentation. À partir de ces représentations, un score de similarité est calculé pour chaque tweet par rapport au résumé de référence.
2. Ensuite, le tweet le plus similaire au résumé de référence est ajouté au résumé Oracle. Ainsi, le résumé Oracle n'est plus vide.
3. Le résumé Oracle n'étant plus vide, la similarité est calculée entre le résumé de référence, et, non pas uniquement chaque tweet, mais chaque tweet *ajouté* au résumé Oracle. Ainsi, tant que la *taille* du résumé Oracle n'est pas *suffisante* par rapport à la *taille* du résumé de référence **et** qu'au moins un tweet améliore la similarité entre le résumé de référence et le résumé Oracle, alors le tweet qui maximise la similarité est *ajouté*.

Plusieurs points sont à éclaircir lors de la construction des résumés Oracles. Premièrement, nous utilisons la taille moyenne d'un tweet pour établir si la *taille* du résumé Oracle est *suffisante* par rapport à la *taille* du résumé de référence. Effectivement, il est important de réguler la taille des résumés pour les comparer. Sinon, un résumé plus long peut être avantagé pour apporter plus d'informations. Les résumés de référence étant utilisés pour construire les résumés Oracles, nous utilisons leur taille pour réguler celle des résumés générés. Nous stoppons l'incrémentation du résumé Oracle lorsque sa taille est supérieure à la taille du résumé de référence moins la taille moyenne d'un tweet, en nombre de mots. Ainsi, les résumés sont approximativement de la même taille en nombre de mots, à, au pire, un tweet près.

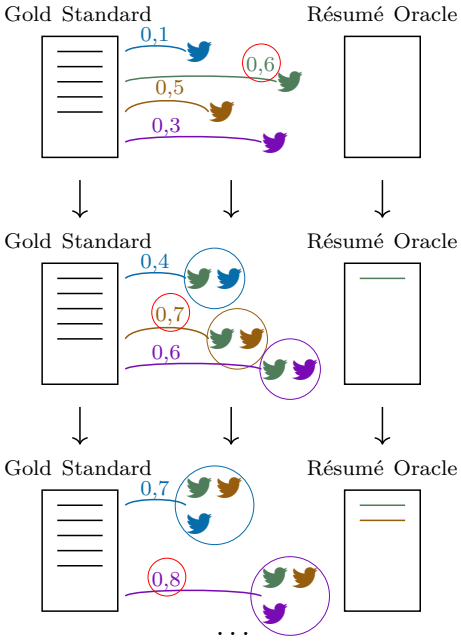


FIGURE 7.1 – Illustration de la création incrémentale du résumé Oracle. Lorsque le résumé Oracle n'est plus vide, la similarité est calculée entre le résumé de référence et chaque tweet ajouté au résumé Oracle.

Deuxièmement, nous évaluons deux méthodes pour *ajouter* les tweets au résumé Oracle. Nous illustrons ces deux méthodes en figure 7.2. D'un côté le résumé de référence ne varie pas, la représentation est toujours la même tout au long de la construction du résumé Oracle correspondant. D'un autre côté, le résumé Oracle évolue au fur et à mesure qu'il est construit. La première méthode évaluée est de concaténer le texte du tweet au texte du résumé Oracle puis de le représenter (méthode Concat). Il est à noter que cela signifie qu'à chaque incrément, tous les tweets candidats sont concaténés au résumé Oracle existant, puis projetés à nouveau dans l'espace de représentation pour être comparés au résumé de référence. La deuxième méthode est d'utiliser la moyenne des tweets qui composent le résumé Oracle (mé-

thode Mean). Ainsi, il n'est pas nécessaire d'évaluer les représentations à chaque incrément, mais il suffit simplement d'effectuer une moyenne entre les tweets qui composent le résumé Oracle et chaque tweet candidat.

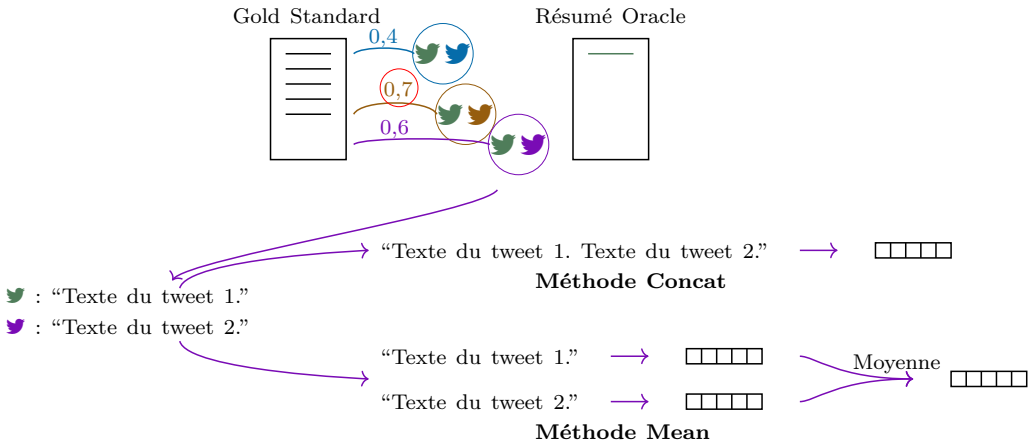


FIGURE 7.2 – Illustration des deux méthodes utilisées pour ajouter des tweets au résumé Oracle.

Ensuite, nous évaluons les résumés Oracles générés à l'aide de chaque représentation par rapport aux résumés de référence selon le même protocole que nous avons utilisé en section 4.1 du chapitre 6. Cela nous permet de comparer les représentations entre elles, et également par rapport à l'état de l'art.

2.2 *Cadre expérimental*

Dans cette section nous détaillons le cadre expérimental mis en place. Nous présentons les différentes représentations évaluées et l'évaluation conduite.

2.2.1 *Présentation des différentes représentations évaluées*

Nous évaluons cinq représentations différentes de l'état de l'art pour cette étude. Nous avons choisi les représentations générées par les modèles XLNet et USE d'après les résultats des travaux menés par Wang *et al.* (2020a).

Ensuite, nous avons choisi trois représentations supplémentaires. Nous avons choisi des représentations générales, DistilBERT (Sanh *et al.*, 2019) et Sentence-BERT (Reimers et Gurevych, 2019). DistilBERT atteint des résultats similaires à BERT avec une architecture réduite de 40 % et une rapidité augmentée de 60 %. Sentence-BERT se focalise sur les représentations générées, pour que des phrases ou documents sémantiquement similaires soient proches dans l'espace de projection. Nous avons

également choisi une représentation spécialisée pour les tweets, BERTweet (Nguyen *et al.*, 2020). BERTweet est basé sur BERT mais spécifiquement entraînée pour les tweets. Nous détaillons les différents modèles utilisés dans les paragraphes suivants, et nous les récapitulons dans le tableau 7.1.

2.2.1.1 DistilBERT

DistilBERT (Sanh *et al.*, 2019) est une version allégée de BERT. L'architecture générale de DistilBERT est la même que BERT avec un nombre de couches divisé par 2. Le modèle est appris par distillation de connaissances du modèle BERT. Nous utilisons le modèle de base de DistilBERT, qui comprend 66 362 880 paramètres. La taille des vecteurs de représentation générés est de 768.

2.2.1.2 BERTweet

BERTweet (Nguyen *et al.*, 2020) est une version de BERT spécialement entraînée pour les tweets. Le vocabulaire est enrichi au vocabulaire des tweets (64 000 *tokens* contre 30 522 *tokens* pour BERT). Contrairement à BERT, le corpus d'entraînement du modèle est un ensemble de tweets. Nous utilisons le modèle de base de BERTweet, qui comprend 134 899 968 paramètres. La taille des vecteurs de représentation générés est de 768.

2.2.1.3 Sentence-BERT

Sentence-BERT (Reimers et Gurevych, 2019) ou SBERT est une représentation de BERT pour laquelle les phrases ou documents proches sémantiquement sont proches dans l'espace de projection généré. Sentence-BERT ajuste les représentations de BERT. En fait, il est possible d'utiliser l'apprentissage de Sentence-BERT pour d'autres modèles de langue. Par abus de langage nous utilisons le nom de Sentence-BERT ou SBERT quel que soit le modèle utilisé au départ. Dans ce travail, nous utilisons la version Sentence-BERT du modèle MiniLM (Wang *et al.*, 2020b). Le dépôt des différents modèles pré-entraînés avec Sentence-BERT fournit un comparatif des différents modèles [1]. Ainsi, nous utilisons le modèle « all-MiniLM-L6-v2 » pour le compromis entre la taille du modèle, la rapidité de génération des représentations et les performances obtenues. Ce modèle comprend 22 713 216 paramètres. La taille des vecteurs de représentation générés est de 384.

1. `https://www.sbert.net/docs/pretrained_models.html`, dernière consultation le 08/08/2022

TABLEAU 7.1 – Récapitulatif des différentes représentations évaluées.

Représentation	Modèle pré-entraîné utilisé	Nb paramètres	Taille représentation	Fonction/code dédié à calculer la représentation à partir du texte	Impacté par batch size/padding	Orienté tweet
XLNet	xlnet-base-cased	116 718 336	768		✓	
DistilBERT	distilbert-base-uncased	66 362 880	768			
Bertweet	vinai/bertweet-base	134 899 968	768		✓	✓
SentenceBERT	all-MiniLM-L6-v2	22 713 216	384	✓		
USE	Modèle DAN de base, dernière version	256 797 824	512	✓		

2.2.1.4 XLNet

Contrairement à BERT qui a une architecture auto-encodeur, et peut ainsi prendre en compte le contexte d'un *token* de façon bidirectionnelle, l'architecture de XLNet (Yang *et al.*, 2019) est auto-régressive, elle ne peut prendre en compte le contexte d'un *token* que dans une direction, de gauche à droite ou de droite à gauche. En d'autres termes, un modèle dont l'architecture est auto-régressive ne peut « lire » que de gauche à droite ou de droite à gauche, tandis que BERT qui n'est pas limité par une telle architecture peut « lire » tous les mots en même temps. Cependant, pour prendre le contexte de façon bidirectionnelle, XLNet utilise l'objectif PLM (*Permutation Language Modeling*). L'idée derrière cet objectif est de prendre en compte l'ensemble des relations entre les *tokens*, en permutant les *tokens*, et ainsi permettre au modèle d'accéder au contexte de gauche à droite et de droite à gauche, non pas au niveau de l'architecture mais au niveau de l'apprentissage. Dans ce travail nous utilisons la version de base du modèle qui comprend 116 718 336 paramètres. La taille des vecteurs de représentation générés est de 768.

2.2.1.5 USE

La représentation USE (*Universal Sentence Encoder*) (Cer *et al.*, 2018) existe selon deux variantes, Transformer et DAN (*Deep Averaging Network*). Dans ce travail nous utilisons la version DAN pour le temps d'exécution linéaire à la taille de la phrase représentée, contrairement à la représentation Transformer pour laquelle le temps d'exécution est quadratique. La représentation USE version DAN est générée à partir de la moyenne des représentations des mots et des bigrammes (représentation *word2vec skip-gram* (Mikolov *et al.*, 2013)), passés dans une architecture neuronale basique (*feedforward deep neural network*). Dans ce travail nous utilisons la dernière version du modèle de base version DAN [2] qui comprend 256 797 824 paramètres. La taille des vecteurs de représentation générés est de 512.

2. `https://tfhub.dev/google/universal-sentence-encoder/4`, dernière consultation le 08/08/2022

TABLEAU 7.2 – Composants des flux et des GS des jeux de données (Rudra *et al.*, 2018b), TES 2012-2016, et ISSumSet.

Collection	Flux	Gold Standard
TES 2012-2016	Tweets	Texte « classique » : référence externe Wikipedia Current Event Portal
Rudra et al.	Tweets	Tweets : impacté par la perte de tweets, la majorité des tweets du GS ne sont pas dans le flux
ISSumSet	Tweets	Tweets : tous les tweets du GS sont dans le flux

2.2.2 *Jeux de données*

Comme pour les évaluations en section 4.1 du chapitre 6, les expériences ont été menées sur trois jeux de données différents, celui présenté dans (Rudra *et al.*, 2018b), et sur les deux jeux de données ISSumSet (Dusart *et al.*, 2021a) et TES 2012-2016 (Dusart *et al.*, 2021b) que nous avons construits et détaillés en section 2 et section 3 du chapitre 5 :

— Nous avons d'abord utilisé le jeu de données créé dans (Rudra *et al.*, 2018b). Comme présenté en section 1.1 du chapitre 5, ce jeu de données est affecté par la suppression des tweets, et au moment où nous avons réalisé les expériences, seuls 63 % des tweets restaient actuellement accessibles. Pour ce jeu de données, les GS sont composés de tweets. Cependant, une majorité des tweets des GS ne sont plus récupérables, et ne sont plus présents dans les flux à résumer.

— Deuxièmement, nous avons utilisé le jeu de données ISSumSet (Dusart *et al.*, 2021a), présenté en section 3 du chapitre 5. Pour ce jeu de données, les GS sont composés de tweets et sont tous présents dans les flux à résumer.

— Finalement, nous avons également utilisé le jeu de données TES 2012-2016, présenté en section 2 du chapitre 5. Pour ce jeu de données, les GS sont des résumés du Wikipedia Current Event Portal, et ne sont donc pas des tweets.

Quelques statistiques de ces 3 jeux de données sont listées dans le tableau 6.2 du chapitre précédent. Dans un souci de lisibilité et de compréhension nous rappelons en tableau 7.2 les composants des flux et des GS des différents jeux de données.

2.2.3 *Détails d'implémentation*

Pour les méthodes USE et Sentence-BERT, le code est fourni pour obtenir la représentation à partir du texte. Pour les autres, DistilBERT, BERTweet et XLNet, nous récupérons la représentation en calculant la moyenne des représentations de tous les *tokens* sur la dernière couche, comme dans (Wang *et al.*, 2020a).

Pour chaque méthode de représentation nous avons évalué 3 versions différentes sur chaque jeu de données.

Jeux de données ISSumSet et celui de Rudra et al. *(2018b)*

Pour les jeux de données ISSumSet et celui de Rudra *et al.* (2018b), nous avons évalué des versions où les résumés Oracles sont construits par :

— concaténation des tweets gardés pour l'Oracle puis encodé, nommée *Concat*,
— la moyenne de chacun des tweets gardés pour l'Oracle, nommée *Mean*,
— la moyenne de chacun des tweets gardés pour l'Oracle, en encodant les tweets par *batchs* de taille 64 afin d'exécuter les opérations en parallèle. Cette version est dénotée *Mean Batch 64*.

Comme ces jeux contiennent moins de 100 000 tweets, nous avons utilisé l'ensemble des évènements pour évaluer la version *Concat*.

Jeu de données TES 2012-2016

Le temps d'exécution de la version *Concat* sur le jeu de données TES 2012-2016 ne nous a pas permis de l'évaluer sur l'ensemble des données. Également, le temps d'exécution ne nous a pas permis d'encoder les tweets un à un pour les méthodes XLNet, BERTweet et DistilBERT : nous avons dû les encoder par *batchs* (de taille 64) pour les versions *Concat* et *Mean*. Nous avons pour cela évalué les 2 versions *Concat* et *Mean* sur un sous-ensemble de 7 évènements représentant chacun un « type d'évènement » afin de rencontrer le maximum de vocabulaire présent dans l'ensemble du jeu de données (attentat : Boston marathon bombing 2013, prise d'otages : Hijacked plane 2016, manifestation : Hongkong protests 2014, catastrophe naturelle : Hurricane Patricia 2015, fusillade : Ottawa shooting 2014, sport : Superbowl 2012, élection : US election 2012), avec des *batchs* de taille 64 pour encoder les tweets. Nous nommons donc les versions *Concat* et *Mean* respectivement *Partial Concat Batch 64* et *Partial Mean Batch 64*. Le temps d'exécution de la méthode *Mean*, avec une exécution par *batchs* (de taille 64) sur l'ensemble des données restant acceptable, nous avons également évalué cette version, nommée comme précédemment *Mean Batch 64*.

Smart batching

Aussi, pour les 3 méthodes XLNet, DisitilBERT et BERTweet, nous avons utilisé le *smart batching*, comme indiqué dans (Reimers et Gurevych, 2019), qui accélère l'encodage. Lorsque l'encodage de textes est effectué par *batchs*, tous les éléments d'un même *batch* doivent être de même taille (en termes de *tokens*). Lorsque les éléments ne sont pas de même taille, un *padding* est effectué. Des *tokens* « mots vides » sont ajoutés à la fin des éléments afin qu'ils atteignent la même taille. Par

TABLEAU 7.3 – Temps en secondes de génération (chargement des tweets, encodage, création de l'Oracle, sauvegarde de l'Oracle) des Oracles par méthode de représentation pour le jeu de données TES 2012-2016. *Partial Concat Batch 64* et *Partial Mean Batch 64* sont évalués sur 7 évènements, *Mean Batch 64* est évalué sur tous les évènements.

Représentation	Partial Concat Batch 64	Partial Mean Batch 64	Mean Batch 64
XLNet	50 124	20 623	303 113
DistilBERT	30 999	13 596	198 640
Bertweet	29 174	17 819	232 411
USE	7 229	5 760	77 868
Sentence-BERT	9 150	6 035	79 847

TABLEAU 7.4 – Temps en secondes de génération (chargement des tweets, encodage, création de l'Oracle, sauvegarde de l'Oracle) des Oracles par méthode de représentation pour le jeu de données ISSumSet.

Représentation	Concat	Mean	Mean Batch 64
XLNet	3 036	3 256	408
DistilBERT	2 036	2 060	270
Bertweet	3 497	3 496	369
USE	75	76	91
Sentence-BERT	88	79	123

exemple, soit deux phrases p1 = « La phrase un » et p2 = « Ceci est la phrase deux ». Considérons un *token* strictement équivalent à un mot, la taille de p1 est de 3 *tokens*, tandis que la phrase p2 est de 5 *tokens*. Pour être encodées par *batch*, les deux phrases doivent être de la même taille, la phrase p1 serait donc *padded* jusqu'à 5 *tokens*. La phrase p1 deviendrait « La phrase un PAD PAD ». Dans le cas du *smart batching*, les *batchs* sont créés en regroupant les phrases dont les tailles sont similaires (en termes de *tokens*).

2.2.4 *Évaluation de l'efficience*

Nous reportons dans les tableaux 7.3, 7.4, et 7.5 les temps d'exécution des différentes méthodes évaluées respectivement sur les jeux de données TES 2012-2016, ISSumSet et celui de Rudra *et al.* (2018b).

Nous pouvons voir que les temps d'encodage avec XLNet, DistilBERT et BERTweet sont plus longs qu'avec USE et Sentence-BERT. De plus, XLNet et BERTweet, avec plus de paramètres, ont un temps d'encodage plus long que DistilBERT dans la quasi-totalité des configurations sauf une. En effet, pour le jeu de données TES 2012-2016, version *Partial Concat Batch 64*, le temps d'encodage de DistilBERT

TABLEAU 7.5 – Temps en secondes de génération (chargement des tweets, encodage, création de l'Oracle, sauvegarde de l'Oracle) des Oracles par méthode de représentation pour le jeu de données de Rudra *et al.* (2018b).

Représentation	Concat	Mean	Mean Batch 64
XLNet	8 125	6 810	507
DistilBERT	4 488	4 148	379
Bertweet	8 164	7 516	414
USE	94	102	122
Sentence-BERT	137	117	183

est plus long que Bertweet. Pour comprendre ce comportement, il faut revenir à la composition des différents GS. Les GS de ISSumSet et de Rudra *et al.* (2018b) sont créés à partir de tweets, un sous-évènement est donc un tweet, et donc de la taille d'un tweet. Ainsi, lors de la création des Oracles, guidés par la taille des GS, un tweet est dans la majorité des cas suffisant pour atteindre la taille des GS, même si le tweet choisi pour l'Oracle n'est pas le même suivant la méthode d'encodage utilisée. Pour les versions *Mean*, *Partial Mean Batch 64*, et *Mean Batch 64*, un seul encodage est effectué par sous-évènement, quel que soit le nombre de tweets ajoutés à l'Oracle pour le sous-évènement. En revanche, pour les versions *Concat* et *Partial Concat Batch 64* le nombre d'encodages du texte effectués dépend du nombre de tweets ajoutés à l'Oracle. Pour les jeux de données ISSumSet et celui de Rudra *et al.* (2018b), le nombre d'encodages est donc sensiblement le même pour chaque méthode. D'ailleurs, le temps d'exécution pour la version *Concat* et la version *Mean* sont sensiblement les mêmes sur ces deux jeux de données puisque le nombre d'encodages est sensiblement le même. En revanche, pour le jeu de données TES 2012-2016, les GS sont construits à partir de résumés du portail Current Events de Wikipedia, et sont souvent composés de plusieurs phrases. Ainsi, lorsqu'on divise le temps par le nombre de tweets ajoutés à l'Oracle, l'encodage par la méthode DistilBERT est plus rapide que pour les versions BERTweet et XLNet. Un autre comportement à noter est que l'encodage des tweets par *batchs* est moins efficace pour la méthode XLNet que pour la méthode BERTweet. En effet, lorsque les tweets sont encodés un à un, le temps d'exécution est plus rapide pour la méthode XLNet que la méthode BERTweet, tandis que lorsque les tweets sont encodés par *batchs*, le temps d'exécution est plus rapide pour la méthode BERTweet que la méthode XLNet.

Pour les méthodes USE et Sentence-BERT, le temps d'exécution est sensiblement plus rapide pour USE que Sentence-BERT. Un comportement à noter est que ces 2 méthodes sont plus rapides sans regrouper les tweets par *batchs*. En fait, pour ces méthodes, des fonctions sont fournies pour encoder le texte, et un encodage par *batch* est notamment déjà utilisé pour encoder le texte avec la méthode Sentence-BERT.

C'est pour cela que nous n'avons pas utilisé l'encodage par *batchs* pour le jeu de données TES 2012-2016 avec ces 2 méthodes.

2.2.5 *Évaluation de l'efficacité*

Dans la section précédente, nous avons évalué l'efficience des différentes représentations étudiées. Dans cette section nous évaluons l'efficacité des différentes représentations étudiées pour le résumé automatique de flux de tweets. Dans un premier temps nous présentons les métriques utilisées. Ensuite, nous présentons les résultats obtenus.

2.2.5.1 Métriques

Comme pour les évaluations en section 4.1 du chapitre 6, nous avons calculé les scores ROUGE-1 de précision et de F-mesure, les scores de précision et de F-mesure de ROUGE-2, ainsi que Cos Embed, basée sur la métrique Cosinus et des *embeddings* largement utilisés [3].

2.2.5.2 Résultats

Nous analysons les performances des méthodes XLNet, DistilBERT, BERTweet, USE, et Sentence-BERT par rapport aux mesures d'évaluation du résumé reportées dans le tableau 7.6. Tout d'abord, pour le jeu de données ISSumSet, les tweets du flux sont également présents dans les GS. Nous pouvons voir que lorsque les tweets sont encodés un à un (versions *Concat* et *Mean*), la similarité entre un tweet et lui-même est bien de 1 pour chaque méthode. Nous discutons du cas *Mean Batch 64* par la suite. Ensuite, pour le jeu de données de Rudra *et al.* (2018b), les meilleures performances sont obtenues pour les méthodes USE, Sentence-BERT et BERTweet. Enfin, les meilleures performances sont également obtenues pour les méthodes USE et Sentence-BERT mais pas BERTweet sur le jeu de données TES 2012-2016. Le modèle BERTweet a été appris spécifiquement sur des tweets, et la représentation encodée par le modèle paraît performante lorsque le flux et le résumé sont composés de tweets. En revanche, lorsque le résumé n'est pas composé de tweets mais de texte plus classique, les Oracles générés avec BERTweet n'ont pas les mêmes performances. Également, lorsque l'encodage est effectué par *batchs*, même si le résumé est composé de tweets, les performances chutent pour BERTweet.

Malgré le temps d'exécution supérieur des versions *Concat* (dont *Partial Concat Batch 64*) par rapport aux versions *Mean* (dont *Partial Mean Batch 64*), les versions

3. Nous avons utilisé le modèle word2vec-google-news-300 fourni par gensim `https://radimrehurek.com/gensim/models/word2vec.html`, dernière consultation le 08/08/2022.

Mean obtiennent des résultats presque similaires, parfois même meilleurs que la version *Concat*. En revanche, pour les jeux de données ISSumSet et celui de Rudra *et al.* (2018b), pour les méthodes XLNet et BERTweet, la version *Mean Batch 64* perd en performances par rapport à la version *Mean* où l'encodage des tweets est effectué un à un. Lors de l'ajout de *tokens* mots vides en fin de tweet, la représentation du tweet est légèrement modifiée. Le tweet n'est ainsi plus représenté exactement de la même manière qu'il ne l'est lors de l'encodage tweet par tweet. En fait, comme les travaux de Reimers et Gurevych (2019) le montrent pour BERT, les espaces générés par les méthodes XLNet et BERTweet ne semblent pas avoir la propriété que deux textes similaires sémantiquement sont proches dans ces espaces. Pour les méthodes USE et Sentence-BERT, il n'y a pas de perte lors de l'encodage par *batchs*, ce qui n'est pas étonnant puisque ces méthodes de représentation sont spécifiquement conçues pour créer un espace dans lequel deux textes similaires sémantiquement sont proches dans l'espace, et ainsi pouvoir utiliser des mesures de similarités telles que Cosinus (Reimers et Gurevych, 2019). Pour la méthode de représentation DistilBERT c'est en revanche surprenant que l'encodage par *batchs* n'affecte quasiment pas les résultats de l'évaluation, cette méthode étant au départ conçue pour reproduire les résultats de BERT avec une architecture plus légère.

2.3 *Discussion*

L'évaluation de l'efficience des différentes méthodes de représentations montre que USE et Sentence-BERT offrent un gain de vitesse d'exécution par rapport aux autres méthodes évaluées. Il convient tout de même de noter que, contrairement aux autres méthodes évaluées, les méthodes USE et Sentence-BERT se focalisent sur les représentations générées et, pour ces méthodes, les fonctions pour encoder du texte sont fournies par les auteurs. L'évaluation de l'efficacité des méthodes de représentations par rapport aux mesures de résumé montre également que USE et Sentence-BERT sont les plus adaptées.

Ainsi, les performances des différentes méthodes par rapport à l'évaluation des résumés Oracles générés et par rapport au temps d'exécution, nous ont conduits à utiliser les méthodes de représentation USE et Sentence-BERT uniquement pour la suite des expérimentations menées dans la suite de ce chapitre. Nous utiliserons également les versions *Mean* (pour ISSumSet et le jeu de données de Rudra *et al.* (2018b)) et *Mean Batch 64* (pour TES 2012-2016), plus rapides, pour représenter le résumé construit. Dans la section suivante, nous évaluons la représentation d'un flux de tweets à partir de la moyenne des tweets qui le composent.

TABLEAU 7.6 – Résultats sur les jeux de données TES 2012-2016, TES 2012-2016 réduite à 7 évènements, ISSumSet, et sur la collection de Rudra *et al.* (2018b). Les résultats avec * sont statistiquement significatifs et différents du meilleur résultat pour chaque métrique évaluée (test de Student, * $p-value < 0.05$, ** $p-value < 0.01$). Les scores en gras mettent en évidence le meilleur modèle pour la métrique considérée.

	Modèle	ROUGE-1 Macro		ROUGE-2 Macro		COS Embed Macro
		P	F	P	F	
TES 2012-2016	XLNet Mean Batch 64	0,309**	0,270**	0,081**	0,069**	0,774**
	USE Mean Batch 64	**0,499**	0,440	**0,187**	0,163	**0,858**
	BERTweet Mean Batch 64	0,248**	0,231**	0,054**	0,053**	0,752**
	Sentence-BERT Mean Batch 64	0,494	**0,448**	0,184	**0,168**	0,857
	DistilBERT Mean Batch 64	0,364**	0,347**	0,115**	0,110**	0,834**
TES 2012-2016 7 évènements	XLNet Partial Mean Batch 64	0,366**	0,291**	0,122*	0,091*	0,817**
	XLNet Partial Concat Batch 64	0,346**	0,276**	0,122*	0,091*	0,802**
	USE Partial Mean Batch 64	0,518	0,443	0,189	0,156	0,875
	USE Partial Concat Batch 64	**0,533**	**0,468**	**0,204**	**0,173**	**0,898**
	BERTweet Partial Mean Batch 64	0,258**	0,253**	0,063**	0,068**	0,777**
	BERTweet Partial Concat Batch 64	0,242**	0,241**	0,064**	0,069**	0,792**
	Sentence-BERT Partial Mean Batch 64	0,477	0,429	0,169	0,155	0,856*
	Sentence-BERT Partial Concat Batch 64	0,464	0,426	0,172	0,159	0,865
	DistilBERT Partial Mean Batch 64	0,381**	0,366**	0,134*	0,132*	0,866**
	DistilBERT Partial Concat Batch 64	0,394**	0,372**	0,141*	0,137	0,866**
ISSumSet	XLNet Mean Batch 64	0,761**	0,756**	0,689**	0,684**	0,956**
	XLNet Mean	**1,000**	**1,000**	**1,000**	**1,000**	**1,000**
	XLNet Concat	**1,000**	**1,000**	**1,000**	**1,000**	**1,000**
	USE Mean Batch 64	**1,000**	**1,000**	**1,000**	**1,000**	**1,000**
	USE Mean	**1,000**	**1,000**	**1,000**	**1,000**	**1,000**
	USE Concat	**1,000**	**1,000**	**1,000**	**1,000**	**1,000**
	BERTweet Mean Batch 64	0,607**	0,642**	0,488**	0,515**	0,937**
	BERTweet Mean	**1,000**	**1,000**	**1,000**	**1,000**	**1,000**
	BERTweet Concat	**1,000**	**1,000**	**1,000**	**1,000**	**1,000**
	Sentence-BERT Mean Batch 64	**1,000**	**1,000**	**1,000**	**1,000**	**1,000**
	Sentence-BERT Mean	**1,000**	**1,000**	**1,000**	**1,000**	**1,000**
	Sentence-BERT Concat	**1,000**	**1,000**	**1,000**	**1,000**	**1,000**
	DistilBERT Mean Batch 64	0,984**	0,985**	0,979**	0,980**	0,999
	DistilBERT Mean	**1,000**	**1,000**	**1,000**	**1,000**	**1,000**
	DistilBERT Concat	**1,000**	**1,000**	**1,000**	**1,000**	**1,000**
Collection de Rudra *et al.* (2018b)	XLNet Mean Batch 64	0,516**	0,530**	0,343**	0,353**	0,927**
	XLNet Mean	0,612**	0,653**	0,465**	0,496**	0,961**
	XLNet Concat	0,612**	0,653**	0,465**	0,496**	0,962**
	USE Mean Batch 64	**0,757**	**0,820**	**0,656**	**0,710**	**0,987**
	USE Mean	**0,757**	**0,820**	**0,656**	**0,710**	**0,987**
	USE Concat	**0,757**	**0,820**	**0,656**	**0,710**	**0,987**
	BERTweet Mean Batch 64	0,600**	0,675**	0,468**	0,526**	0,970**
	BERTweet Mean	0,732	0,793	0,628	0,680	0,983
	BERTweet Concat	0,733	0,794	0,629	0,680	0,984
	Sentence-BERT Mean Batch 64	0,748	0,814	0,644	0,700	**0,987**
	Sentence-BERT Mean	0,748	0,814	0,644	0,700	**0,987**
	Sentence-BERT Concat	0,748	0,814	0,644	0,700	**0,987**
	DistilBERT Mean Batch 64	0,581**	0,659**	0,445**	0,504**	0,976**
	DistilBERT Mean	0,600**	0,675**	0,466**	0,523**	0,977**
	DistilBERT Concat	0,600**	0,674**	0,466**	0,523**	0,977**

3 Évaluation de la représentation d'un flux de tweets à partir de la moyenne des tweets qui le composent pour le résumé automatique

Dans cette section, nous proposons de répondre au deuxième objectif de recherche énoncé dans la section précédente, à savoir :

Objectif de recherche : La représentation d'un flux de tweets à partir de la moyenne des tweets qui composent le flux est-elle utilisable pour le résumé automatique de flux de tweets ?

Pour répondre à cet objectif, nous proposons la méthodologie suivante :

Méthodologie pour répondre à l'objectif de recherche : Nous nous intéressons à la représentation moyenne brute, sans modèle. Nous proposons de créer des résumés en considérant la moyenne des tweets qui composent le flux comme résumés de référence. Ainsi, nous pouvons comparer les résumés générés par cette méthode aux résumés Oracles et aux résumés de l'état de l'art.

3.1 *Protocole et cadre d'évaluation*

Dans ces expérimentations, nous construisons les résumés en utilisant le même algorithme glouton que celui décrit en section 2.1. Les résumés construits de façon incrémentale sont ici comparés non pas au GS mais à la moyenne des tweets qui composent le flux.

Le cadre d'évaluation est également le même que décrit en section 2. Les jeux de données utilisés sont les jeux de données TES 2012-2016, ISSumSet, et celui de Rudra *et al.* (2018b). Les métriques utilisées sont ROUGE-1, ROUGE-2, et Cos Embed.

3.2 *Résultats et discussion*

Suite aux résultats obtenus lors de l'évaluation des Oracles générés par les différentes méthodes de représentation textuelle, voir section 2.2, nous conduisons nos expérimentations avec les représentations USE et Sentence-BERT. Guidés par les GS, nous avons construit des résumés Oracles lors de l'évaluation des différentes méthodes de représentation. L'évaluation suit le même procédé mais considère quant

à elle la moyenne de l'ensemble des tweets du flux à résumé comme équivalent GS. Nous reportons en tableau 7.7 les résultats des évaluations conduites sur les jeux de données TES 2012-2016, ISSumSet, et celui de Rudra *et al.* (2018b), pour les méthodes de représentation textuelle USE et Sentence-BERT, selon la version *Mean* pour la construction du résumé. Pour chaque jeu de données, ces résultats sont reportés sur les lignes *USE* et *Sentence-BERT*. L'évaluation étant la même que celle menée en section 4.1 du chapitre 6, nous reportons également les résultats de l'état de l'art ainsi que ceux des versions de TSSuBERT guidées par la taille des GS.

Nous pouvons voir que l'utilisation de la moyenne de l'ensemble des tweets du flux à résumer permet d'obtenir de meilleurs résultats que l'état de l'art pour au moins une métrique parmi celles évaluées sur chaque jeu de données. Pour l'ensemble des mesures de précision ROUGE-1 et ROUGE-2 les résultats surpassent ceux de l'état de l'art pour les versions USE et Sentence-BERT. Les résultats sont également meilleurs que ceux de TSSuBERT sur les jeux de données TES 2012-2016 et celui de Rudra *et al.* (2018b).

En revanche, cette méthode de résumé atteint des limites en termes de rappel. Effectivement, les résumés sont construits à partir d'une moyenne d'information. Des informations sont donc perdues au profit de la compression des données. Ceci impacte également la mesure de similarité sémantique *Cos Embed* puisque l'information perdue lors de la compression n'est pas traitée dans les résumés.

Nous pouvons également noter que la représentation USE se montre plus adaptée que la représentation Sentence-BERT pour notre évaluation.

L'utilisation de la moyenne de l'ensemble des tweets du flux à résumer se montre prometteuse pour la génération de résumé automatique de flux de tweets. Aussi, nous nous intéressons dans la suite à l'utilisation d'un modèle d'apprentissage pour améliorer cette méthode.

4 Évaluation d'un modèle d'apprentissage automatique pour améliorer la représentation d'un flux de tweets à partir des tweets qui le composent

Dans cette section, nous proposons de répondre au troisième objectif de recherche énoncé dans la section précédente, à savoir :

TABLEAU 7.7 – Résultats sur les jeux de données TES 2012-2016, ISSumSet, et sur la collection de Rudra *et al.* (2018b). Les résultats avec * sont statistiquement significatifs et différents de l'état de l'art (COWTS-SEMCOWTS-SCC) (test de Student, * $p-value < 0.05$, ** $p-value < 0.01$). Les scores en gras mettent en évidence le meilleur modèle pour la métrique considérée.

	Modèle	ROUGE-1 Macro		ROUGE-2 Macro		COS Embed Macro
		P	F	P	F	
TES 2012-2016	COWTS	0,051	0,054	0,002	0,002	0,655
	SEMCOWTS	0,064	0,068	0,002	0,002	0,686
	SCC	0,046	0,049	0,001	0,001	0,619
	TES TSSuBERT-F	0,134**	0,114**	0,023**	0,021**	0,676*
	ISSt TSSuBERT-F	0,133**	0,124**	0,022**	0,021**	0,656**
	USE	**0,180****	**0,133****	**0,037****	**0,028****	**0,626****
	Sentence-BERT	0,134**	0,116**	0,027**	0,024**	0,612**
ISSumSet	COWTS	0,345	0,372	0,120	0,130	**0,870**
	SEMCOWTS	0,337	0,363	0,118	0,127	0,869
	SCC	0,315	0,339	0,099	0,106	0,852
	TES TSSuBERT-F	0,362*	0,357**	0,181**	0,179**	0,849
	ISSt TSSuBERT-F	**0,425****	**0,382**	**0,207****	**0,186****	0,857
	USE	0,407**	0,223**	0,154**	0,097**	0,801
	Sentence-BERT	0,400**	0,198**	0,164**	0,099**	0,778**
Collection de Rudra *et al.* (2018b)	COWTS	0,423	**0,459**	0,221	**0,242**	0,931
	SEMCOWTS	0,420	0,456	0.214	0,234	0,930
	SCC	0,390	0.422	0,196	0,215	0,924
	TES TSSuBERT-F	0,396	0,404**	0,181*	0,188**	0,930
	ISSt TSSuBERT-F	0,478	0,432	0,253	0,227	**0,935**
	USE	**0,587****	0,192**	**0,270**	0,089**	0,814**
	Sentence-BERT	0,581**	0,171**	0,262	0,082**	0,801**

Objectif de recherche : Peut-on améliorer la représentation d'un flux de tweets créée à partir de la moyenne des tweets qui composent le flux à l'aide d'un modèle d'apprentissage automatique ?

Pour répondre à cet objectif, nous proposons la méthodologie suivante :

Méthodologie pour répondre à l'objectif de recherche : Pour répondre à cette question, nous construisons un modèle dans le but de prédire la représentation du résumé de référence à partir de la moyenne des tweets qui composent le flux. Ainsi, une fois la représentation prédite, les résumés sont construits et évaluables de la même manière que pour les deux autres objectifs.

4.1 *Protocole et cadre d'évaluation*

Le protocole d'évaluation mis en place est le même que décrit en section 2 et en section 3, les résumés sont construits avec un algorithme glouton. En revanche, les résumés de référence sont remplacés par la représentation prédite par le modèle d'apprentissage à partir de la moyenne des tweets qui composent le flux. Le cadre d'évaluation est également le même que décrit en section 2 et section 3. Les jeux de données utilisés sont les jeux de données TES 2012-2016, ISSumSet, et celui de Rudra *et al.* (2018b). Les métriques utilisées sont ROUGE-1, ROUGE-2, et Cos Embed.

Contrairement aux section 2 et section 3, nous utilisons un modèle d'apprentissage. Nous présentons dans les paragraphes suivants le modèle, ainsi que la création des jeux de données pour l'apprentissage du modèle.

4.1.1 *Modèle d'apprentissage*

Nous avons choisi le modèle DAN (Deep Averaging Network), présenté dans (Iyyer *et al.*, 2015). En effet, ce modèle utilise la moyenne des mots d'une phrase pour la représenter. Les différentes étapes du modèle sont :

— calculer la moyenne vectorielle des représentations associées à une séquence d'entrée de *tokens*,
— passer cette représentation moyenne par une ou plusieurs couches de propagation avant (*feedforward layers*),
— effectuer une classification (linéaire) sur la représentation de la couche finale.

L'architecture du modèle est représentée en figure 7.3

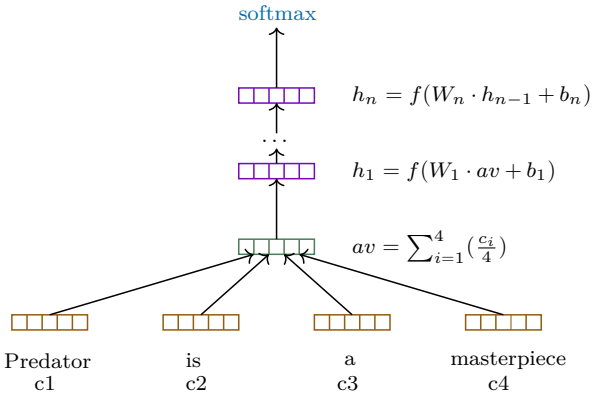


FIGURE 7.3 – Architecture du modèle DAN (Iyyer *et al.*, 2015).

Le modèle DAN est destiné initialement pour la tâche de classification de texte. Cependant, nous ne souhaitons pas utiliser le modèle pour une tâche de classification de texte, mais pour améliorer la représentation du flux dans le même espace. Aussi, au lieu d'utiliser une fonction *softmax* en dernière couche du modèle, nous utilisons une fonction tangente hyperbolique, définie sur $\mathbb{R}$ à valeurs dans $]-1,1[$, comme les espaces de représentations des modèles étudiés, en dernière couche du modèle. Cette dernière couche est de la taille de l'espace de représentation, c'est-à-dire de la même taille que les vecteurs en entrée du modèle. L'idée est que le modèle trouve des motifs à partir de la représentation du flux pour modifier sa représentation vers la représentation du GS (Gold Standard, résumé de référence) associé pendant l'apprentissage, et vers la représentation d'un potentiel GS pendant les tests. Suivant le nombre par défaut utilisé sur leur dépôt Github [4], nous utilisons 3 couches de propagation avant, chaque couche utilise une fonction d'activation ReLu. De plus, comme proposé par Srivastava *et al.* (2014), nous avons ajouté sur chacune de ces couches, sauf celle de prédiction, une couche de Dropout avec une probabilité de 0.5. Nous avons entraîné notre modèle en utilisant l'optimiseur Adam avec les paramètres $\beta_1 = 0.9$ et $\beta_2 = 0.999$, une taille de *batchs* de 128, un taux d'apprentissage décroissant comme dans (Liu et Lapata, 2019; Vaswani *et al.*, 2017) avec une fonction objectif d'erreur quadratique moyenne. Pour chaque *fold*, nous avons entraîné le modèle sur 90 % des 2 autres *folds* avec 10 % pour la validation, pendant 1 000 *epochs*. Pour plus de lisibilité, nous nommons notre version du modèle DAN, DAN-TSS (pour Tweets Stream Summarization), et nous présentons l'architecture en figure 7.4.

Pour résumer, le modèle DAN-TSS prend en entrée la moyenne du flux de tweets selon une représentation dense (selon un vecteur), pour prédire le GS (selon l'espace de représentation de l'entrée) associé à ce flux de tweets. Si aucun GS n'est associé

4. `https://github.com/miyyer/dan`, dernière consultation le 08/08/2022

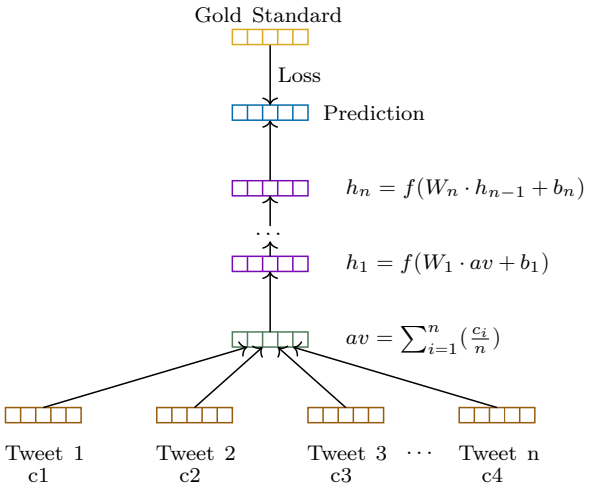


FIGURE 7.4 – Architecture du modèle DAN-TSS.

à ce flux, le modèle doit retourner une phrase vide (selon l'espace de représentation de l'entrée également).

4.1.2 *Jeux de données*

Pour l'évaluation du modèle, nous avons, comme en section 4.1 du chapitre 6, différentes configurations des jeux de données :

— La *cross-validation*, c'est-à-dire que l'apprentissage et le test sont effectués sur le même jeu de données. Ici, comme en section 4.1 du chapitre 6, nous divisons nos jeux de données en 3 *folds*, 2 *folds* pour l'apprentissage et un pour le test. 10 % du *fold* d'apprentissage est utilisé pour validation.

— L'apprentissage est effectué sur un jeu de données et l'évaluation du modèle sur un autre. Dans cette configuration, tout le jeu de données d'apprentissage est utilisé, à l'exception des évènements similaires au jeu de données d'évaluation. 10 % de ces données d'apprentissage sont utilisés pour validation.

Comme en section 4.1 du chapitre 6, le modèle est évalué en *cross-validation* sur ISSumSet et TES 2012-2016 mais pas sur le jeu de données de Rudra *et al.* (2018b) car il contient trop peu d'évènements. À l'image de la section 4.1 du chapitre précédent, le modèle est appris sur ISSumSet (évalué sur TES 2012-2016 et la collection de Rudra *et al.* (2018b)) et TES 2012-2016 (évalué sur ISSumSet et la collection de Rudra *et al.* (2018b)). La collection de Rudra *et al.* (2018b) ne contient qu'un seul évènement qui n'apparaît pas dans ISSumSet et TES 2012-2016, nous n'apprenons donc pas le modèle à partir de ce jeu de données.

En revanche, contrairement au travail présenté en section 4.1 du chapitre 6, l'entrée du modèle n'est pas un tweet mais le flux entier à résumer. Ce flux est la moyenne

de tous les tweets, selon la représentation choisie. En sortie, le modèle doit prédire le GS associé à ce flux. Ainsi, le nombre d'échantillons différents est celui du nombre de fenêtres de temps, soit seulement 545 pour TES 2012-2016, et seulement 397 pour ISSumSet. Afin d'augmenter le nombre d'échantillons pour l'apprentissage du modèle, nous n'utilisons pas la moyenne du flux complet de la fenêtre de temps. Pour une fenêtre de temps associée à un GS, nous générons 500 échantillons comprenant aléatoirement 80 % de l'ensemble des tweets du flux. Ainsi, la représentation moyenne n'est pas la même entre les échantillons mais fait référence au même GS. Pour une fenêtre de temps non associée à un GS (ou GS vide), nous ne générons pas 500 mais 25 échantillons, afin de ne pas déséquilibrer le jeu de données et que le modèle ne se contente pas d'apprendre à prédire un GS vide quel que soit l'entrée. Effectivement, pour le jeu de données TES 2012-2016 le cas de figure d'un GS vide apparaît pour près de 65 % des fenêtres de temps.

Nous avons également expérimenté les différents seuils de proportion du flux total à utiliser pour générer les échantillons sur le jeu de données TES 2012-2016 à l'aide de la représentation Sentence-BERT. Nous avons évalué la similarité entre la moyenne des tweets du flux total et la moyenne des tweets des échantillons de taille 5 % à 95 % des tweets du flux total, avec un pas de 5 %. Entre 95 % et 80 %, la similarité minimale entre un échantillon et la moyenne du flux total est la même, mais diminue en-dessous de 80 %, voir figure 7.5. C'est pourquoi nous avons choisi le seuil de 80 %.

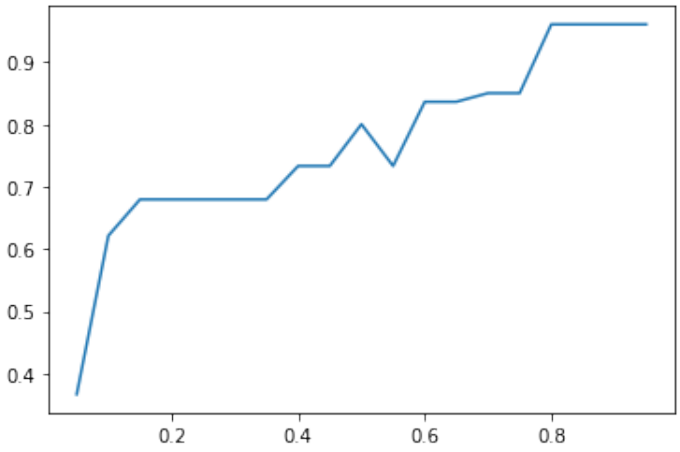


FIGURE 7.5 – Similarité Cosinus minimale (en ordonnée) pour l'ensemble des échantillons suivant la proportion du flux gardée par les échantillons (en abscisse) pour le jeu données TES 2012-2016, selon la représentation Sentence-BERT.

4.2 *Résultats et discussion*

Nous avons évalué le modèle décrit en section 4.1.1 sur les jeux de données TES 2012-2016, ISSumSet, et celui de Rudra *et al.* (2018b). Les différentes versions évaluées, spécifiées dans le tableau 7.8, sont reportées dans le tableau 7.9 sont :

TABLEAU 7.8 – Spécification des différentes versions du modèle évalué par rapport au jeu d'apprentissage et à la représentation utilisés.

Nom du modèle	Jeu d'apprentissage	Représentation utilisée
TES Model USE	TES 2012-2016	USE
TES Model Sentence-BERT	TES 2012-2016	Sentence-BERT
ISSt Model USE	ISSumet	USE
ISSt Model Sentence-BERT	ISSumet	Sentence-BERT

Pour le jeu de données TES 2012-2016, nous pouvons voir que la version *TES Model USE* perd en précision mais améliore le rappel, la F-mesure de ROUGE-1 et ROUGE-2, et la mesure de similarité sémantique COS Embed de la représentation USE. Pour ces métriques, ROUGE-1 F, ROUGE-2 F et COS Embed, *TES model USE* obtient même les meilleurs résultats évalués tous modèles confondus. Pour les autres versions, les scores obtenus par la mesure de similarité sémantique sont améliorés par l'ajout du modèle d'apprentissage. Pour le reste des mesures évaluées, les scores diminuent par rapport à la représentation sans modèle, mais ils restent tout de même supérieurs à l'état de l'art.

Pour le jeu de données ISSumSet, quasiment toutes les mesures évaluées sont améliorées par l'utilisation du modèle. L'amélioration s'accentue lorsque le modèle est appris et testé sur le même jeu de données. La version *ISSt Model USE* obtient les meilleurs résultats tous modèles confondus pour les mesures ROUGE-1 P et ROUGE-2 P.

Pour le jeu de données de Rudra *et al.* (2018b), l'utilisation du modèle a tendance à diminuer les résultats des mesures évaluées. Hormis pour la mesure de similarité sémantique COS Embed dont les scores des versions *TES Model USE* et*ISSt Model USE* augmentent, les autres scores diminuent, ou au mieux atteignent les scores sans l'utilisation du modèle.

Lorsque le modèle est appris et testé sur un même jeu de données, l'utilisation du modèle semble intéressante pour améliorer le rappel. Cependant, il semble limité par rapport aux mesures ROUGE lorsque le jeu de données d'apprentissage et de test ne sont pas les mêmes. En revanche, que les jeux d'apprentissage et de test soient les mêmes ou non, le modèle semble intéressant pour améliorer la similarité sémantique.

TABLEAU 7.9 – Résultats sur les jeux de données TES 2012-2016, ISSumSet, et sur la collection de Rudra *et al.* (2018b). Les résultats du tableau 7.7 sont reportés pour faciliter la comparaison. Les résultats avec * sont statistiquement significatifs et différents de l'état de l'art (COWTS-SEMCOWTS-SCC) (test de Student, * $p-value < 0.05$, ** $p-value < 0.01$). Les scores en gras mettent en évidence le meilleur modèle pour la métrique considérée. Les spécifications des différentes versions d'apprentissage du modèle sont décrites dans le tableau 7.8. Le jeu de test utilisé est indiqué dans la colonne de gauche.

	Modèle	ROUGE-1 Macro		ROUGE-2 Macro		COS Embed Macro
		P	F	P	F	
TES 2012-2016	COWTS	0,051	0,054	0,002	0,002	0,655
	SEMCOWTS	0,064	0,068	0,002	0,002	0,686
	SCC	0,046	0,049	0,001	0,001	0,619
	TES TSSuBERT-F	0,134**	0,114**	0,023**	0,021**	0,676*
	ISSt TSSuBERT-F	0,133**	0,124**	0,022**	0,021**	0,656**
	USE	**0,180****	0,133**	**0,037****	0,028**	0,626**
	Sentence-BERT	0,134**	0,116**	0,027**	0,024**	0,612**
	TES Model USE	0,177**	**0,156****	0,033**	**0,029****	**0,723****
	TES Model Sentence-BERT	0,113**	0,089*	0,011**	0,009**	0,646**
	ISSt Model USE	0,131**	0,114**	0,019**	0,017**	0,666
	ISSt Model Sentence-BERT	0,094**	0,074	0,007**	0,006**	0,616**
ISSumSet	COWTS	0,345	0,372	0,120	0,130	**0,870**
	SEMCOWTS	0,337	0,363	0,118	0,127	0,869
	SCC	0,315	0,339	0,099	0,106	0,852
	TES TSSuBERT-F	0,362*	0,357**	0,181**	0,179**	0,849
	ISSt TSSuBERT-F	0,425**	**0,382**	0,207**	**0,186****	0,857
	USE	0,407**	0,223**	0,154**	0,097**	0,801
	Sentence-BERT	0,400**	0,198**	0,164**	0,099**	0,778**
	TES Model USE	0,419**	0,266**	0,174**	0,119	0,828
	TES Model Sentence-BERT	0,388**	0,199**	0,162**	0,093**	0,781**
	ISSt Model USE	**0,448****	0,276**	**0,214****	0,137	0,824
	ISSt Model Sentence-BERT	0,426**	0,238**	0,193**	0,114	0,811
Collection de Rudra *et al.* (2018b)	COWTS	0,423	**0,459**	0,221	**0,242**	0,931
	SEMCOWTS	0,420	0,456	0.214	0,234	0,930
	SCC	0,390	0.422	0,196	0,215	0,924
	TES TSSuBERT-F	0,396	0,404**	0,181*	0,188**	0,930
	ISSt TSSuBERT-F	0,478	0,432	0,253	0,227	**0,935**
	USE	**0,587****	0,192**	**0,270**	0,089**	0,814**
	Sentence-BERT	0,581**	0,171**	0,262	0,082**	0,801**
	TES Model USE	0,457	0,194**	0,198	0,085**	0,859**
	TES Model Sentence-BERT	0,447	0,134**	0,162*	0,052**	0,807**
	ISSt Model USE	0,459	0,191**	0,192	0,083**	0,830**
	ISSt Model Sentence-BERT	0,455	0,148**	0,201	0,070**	0,794**

Discussion par rapport à TSSuBERT

Nous pouvons comparer TSSuBERT à l'utilisation de la moyenne des tweets du flux dans le cadre du résumé automatique à partir du tableau 7.9. D'un coup d'œil, l'utilisation de la moyenne des tweets du flux, pourtant simple, paraît obtenir les meilleurs résultats et nous pouvons penser que TSSuBERT n'a plus d'intérêt. Cependant, il est à noter que :

— la similarité sémantique, par rapport aux GS, des résumés créés à partir de la moyenne des tweets du flux souffre dans la majorité des cas quand on la compare à TSSuBERT,
— les résumés créés à partir de la moyenne des tweets du flux évalués nécessitent la taille des GS, contrairement à TSSuBERT qui peut être utilisé sans cette information,
— TSSuBERT utilise le modèle DistilBERT pour sa légèreté, mais pourrait être utilisé avec un autre modèle de langue, comme USE et Sentence-BERT.

Ainsi, ces résultats soulèvent la nécessité de poursuivre l'étude, à la fois du modèle TSSuBERT, et de l'utilisation de la moyenne des tweets du flux. D'un côté, le modèle TSSuBERT pourrait potentiellement gagner en efficacité avec l'utilisation de représentations comme USE et Sentence-BERT. D'un autre côté, l'utilisation de la moyenne des tweets du flux pourrait être développée afin de se soustraire de la taille du GS.

5 Conclusion

Dans ce chapitre, nous avons exploré les représentations textuelles pré-établies pour le résumé automatique de flux de tweets. Dans un premier temps, nous avons évalué différentes représentations existantes, générées par les modèles XLNet, Sentence-BERT, DistilBERT, BERTweet et USE. Nous les avons évaluées pour le résumé automatique de flux de tweets par la création de résumés Oracles. Les résumés Oracles sont des résumés optimaux construits à partir des GS (résumés références). Les résumés générés à l'aide de USE et Sentence-BERT ont montré des résultats significativement meilleurs sur les mesures de résumés évaluées. Aussi, l'utilisation des représentations USE et Sentence-BERT a montré un temps d'exécution plus rapide que XLNet, DistilBERT et BERTweet. Nous avons également remarqué que les représentations générées par XLNet et BERTweet sont impactées par le *padding*. Enfin, les représentations générées par BERTweet, spécifiquement appris sur corpus de tweets, souffrent lorsque le texte n'est pas composé de tweets.

Dans un deuxième temps, nous avons généré des résumés, non pas en considérant les GS mais la moyenne de l'ensemble des tweets. Nous avons montré que les résumés

ainsi générés obtiennent une bonne précision pour les mesures ROUGE-1 et ROUGE-2, surpassant l'état de l'art. Néanmoins, ces résumés souffrent au niveau du rappel pour ces mesures.

Dans un troisième temps, nous avons généré des résumés à l'aide de la moyenne de l'ensemble des tweets avec l'ajout d'un modèle d'apprentissage. Ce modèle montre des limites mais paraît prometteur lorsque l'apprentissage et le test sont effectués sur le même jeu de données pour améliorer le rappel. Le modèle permet également d'améliorer l'aspect sémantique des résumés.

Finalement, nous pensons qu'il est intéressant de creuser l'utilisation de la moyenne des tweets du flux pour le résumé automatique. En effet, à l'aide des représentations USE et Sentence-BERT, cette méthode surpasse l'état de l'art sur de nombreuses métriques évaluées. L'ajout d'un modèle d'apprentissage nous a également permis d'améliorer le rappel et l'aspect sémantique des résumés. Comme travaux futurs, il serait intéressant de développer le modèle afin de ne plus utiliser la taille des GS. Il serait également intéressant d'ajouter à cette méthode une détection de sous-évènements ou un regroupement de tweets en sous-flux par sous-évènements (*clustering*) afin de calculer la moyenne des tweets des sous-flux. En effet, lors de la création des résumés Oracles, nous disposons d'un vecteur de représentation par sous-évènement. De ce fait, s'il y a plusieurs sous-évènements lors d'une même fenêtre de temps, il y a plusieurs représentations pour la fenêtre de temps. Or, lorsqu'on utilise la moyenne des tweets du flux, s'il y a plusieurs sous-évènements dans ce flux, nous ne pouvons nous appuyer que sur un vecteur de représentation pour plusieurs sous-évènements. Ainsi, nous pourrions limiter la perte d'information lors de la compression des tweets en une moyenne, générer des résumés plus diversifiés, et améliorer le rappel lors de l'évaluation.

Partie III

CONCLUSION

CONCLUSION

Synthèse des contributions

Dans cette thèse, nous nous sommes intéressés à l'élaboration d'information synthétisée à partir de flux de données issus de médias sociaux. Les médias sociaux regorgent d'informations, qui peuvent être utiles, d'intérêt. Cependant, la trop grande quantité de données présente peut demander au lecteur un effort fastidieux afin d'accéder à l'information pertinente. Ainsi, nous avons axés nos travaux sur l'élaboration automatique de résumés de flux de tweets, données du média social Twitter.

Tout d'abord, nous nous sommes intéressés à l'accès à l'information sur les médias sociaux dans le chapitre 2. Ensuite, nous nous sommes intéressés à l'élaboration d'information synthétique sous forme de résumé, et plus particulièrement sur les méthodes de résumé automatique dans le chapitre 3. Enfin, nous avons examinés les méthodes de résumés automatique avec prise en compte de la dimension temporelle dans le chapitre 4. Nous avons remarqué que les approches de résumé automatique de flux de tweets n'utilisent pas les modèles de langue neuronaux, pourtant largement utilisés dans l'état de l'art du résumé automatique. Nous avons identifié deux principales explications à cela : (i) la difficulté à construire des jeux de données de taille suffisante pour entraîner de tels modèles, et (ii) la taille des flux de tweets, trop importante pour utiliser trivialement les modèles de langue neuronaux. Ainsi, nous avons proposé plusieurs contributions pour le résumé automatique de flux de tweets.

Dans un premier temps, nous avons construits 2 jeux de données, TES 2012-2016 et ISSumSet. Effectivement, il n'existe pas de jeu de données de référence pour le résumé automatique de flux de tweets. Cependant, construire un jeu de données pour le résumé automatique est une tâche fastidieuse. Nous avons donc proposé une méthode pour créer des collections pour le résumé automatique de flux de tweets demandant un minimum d'efforts humains. Nous avons, par le biais de cette méthode, créé le jeu de données TES 2012-2016, contenant plus de 80 000 000 de tweets. Nous avons également proposé un jeu de données qui pourrait servir de référence pour le résumé automatique de flux de tweets, que nous avons nommé ISSumSet. En effet, la distribution d'un jeu de données contenant des tweets est soumise à des règles strictes, notamment au regard de la diffusion de données personnelles. Ainsi, pour

diffuser un jeu de données contenant des tweets, les tweets sont diffusés via leurs identifiants. Néanmoins, les tweets peuvent être supprimés entre la diffusion du jeu de données et la récupération des tweets par un utilisateur tiers. Aussi, il devient impossible de se comparer aux travaux précédents sur un même jeu de données. Pour répondre à ce problème, nous proposons d'utiliser la collection de la tâche TREC Incident Streams pour le résumé automatique de flux de tweets. Nous étudions la possibilité et les limites d'utiliser cette collection qui n'a pas été construite pour du résumé automatique de flux de tweets. La particularité de cette collection est que les tweets sont hébergés par les organisateurs de la tâche et elle n'est donc pas soumise à la suppression des tweets.

Dans un second temps, nous avons proposé un modèle de résumé automatique de flux de tweets, nommé TSSuBERT. Ce modèle est composé d'un modèle de langue neuronal pré-entraîné. Nous utilisons ici une version allégée de BERT, DistilBERT. Afin que le modèle prenne en compte le contexte d'un tweet au sein du flux, nous ajoutons au modèle la fréquence d'apparition des mots dans le flux. Nous montrons que l'ajout de la fréquence d'apparition des mots permet d'améliorer le modèle. Nous montrons également que TSSuBERT surpasse l'état de l'art sur de nombreuses métriques de résumé automatique et jeux de données.

Finalement, nous avons étudié une autre manière de prendre en compte le contexte d'un tweet, qui utilise les modèles de langue pré-entraînés. Pour cela, nous avons exploré les représentations textuelles pré-établies de l'état de l'art pour le résumé automatique de flux de tweets. D'abord, nous avons évalué la capacité de chaque représentation à générer des résumés Oracles. Ensuite, nous avons exploré la possibilité de générer des résumés à partir de la moyenne de l'ensemble des tweets d'un flux à résumer, selon les meilleures représentations évaluées précédemment. Enfin, nous avons utilisé un modèle d'apprentissage afin d'améliorer les représentations de la moyenne de l'ensemble des tweets d'un flux à résumer. Nous montrons que l'utilisation de de la moyenne de l'ensemble des tweets d'un flux à résumer est intéressante pour le résumé automatique de flux de tweets.

Perspectives

Les travaux de cette thèse amènent à différentes perspectives, nous en citons quelques unes :

- À court terme :
  - Nous avons exploré quelques variantes du modèle TSSuBERT, mais d'autres sont encore à étudier. Nous pensons que l'utilisation d'autres modèles de langue neuronaux, peuvent améliorer son efficacité, comme par

exemple USE et Sentence-BERT présentés dans le chapitre 7. De plus, nous avons essayé d'évaluer l'apport de l'ajout de caractéristiques propres aux tweets dans le modèle TSSuBERT. Cependant, les caractéristiques des données que nous utilisons ne sont plus dans le même état qu'au moment de la publication du tweet. Il serait intéressant d'utiliser des données dont l'état des caractéristiques est identique à celui de la fenêtre de temps étudiée. Par exemple, nous pourrions travailler avec les tweets du jour.

- Nous avons exploré l'utilisation d'un modèle d'apprentissage automatique pour améliorer les représentations de la moyenne de l'ensemble des tweets d'un flux à résumer. Cependant, nous pensons qu'il faudrait explorer davantage d'améliorations de cette représentation. En effet, nous pensons que classer les flux en sous-flux en amont de la représentation moyenne pourrait être bénéfique. Par exemple, nous pourrions regrouper les tweets qui relatent d'un même sujet ensemble et considérer la représentation moyenne de chaque groupe afin de les résumer séparément. Ainsi, les pertes d'informations liées à l'utilisation de la moyenne uniquement seraient moindres. Nous pensons également à ajouter des informations contextuelles supplémentaires au modèle. En effet, il ne prend en entrée que la représentation du flux de la fenêtre de temps considérée. Il pourrait par exemple être bénéfique d'ajouter également la représentation du flux de la fenêtre de temps précédente, pour enrichir le contexte et potentiellement améliorer l'apprentissage du modèle. Nous pensons finalement qu'il serait intéressant d'améliorer le modèle afin de se soustraire de la taille des GS.

- À long terme :
  - Lors de la construction du jeu de données ISSumSet, qui n'est pas initialement conçu pour l'évaluation du résumé automatique de flux de tweets, nous avons évalué le jeu de données selon les différentes propriétés d'un bon résumé. Cependant, parmi les propriétés d'un bon résumé, toutes ne sont pas évaluées. Il est commun dans l'état de l'art de n'évaluer que l'informativité et la non redondance des résumés, et de n'utiliser que la mesure ROUGE pour évaluer les résumés générés, que ce soit pour du résumé de flux de tweets ou pour des textes plus classiques. Dans nos travaux, nous n'avons pas non plus évalué toutes les propriétés d'un bon résumé comme la cohérence ou la cohésion des résumés générés par exemple. C'est une perspective à considérer pour la suite de ces travaux, ou même en général dans le cadre du résumé automatique.
  - Nous avons énoncé les problématiques liées à la synthèse de l'information, mais nous ne les avons pas toutes abordées dans ce manuscrit. Aussi, dans des travaux futurs il serait intéressant de considérer la multiplicité des types d'information et de considérer les images et les documents audios et vidéos. La validité des éléments d'information utilisées est également une piste intéressante, les médias sociaux étant fortement impactés par les fausses

informations, aussi appelées *fake news*. La restitution de l'information sous forme structurée amènerait également une plus-value à ces travaux, nous ne considérons ici qu'une restitution par ordre temporel ou de score de pertinence.

BIBLIOGRAPHIE

Rasim M. Alguliev, Ramiz M. Aliguliyev et Nijat R. Isazade : Multiple documents summarization based on evolutionary optimization algorithm. *Expert Syst. Appl.*, 40(5):1675–1689, 2013. URL https://doi.org/10.1016/j.eswa.2012.09.014.

Nasser Alsaedi, Pete Burnap et Omer F. Rana : Automatic summarization of real world events using twitter. *In Proceedings of the Tenth International Conference on Web and Social Media, Cologne, Germany, May 17-20, 2016*, pages 511–514. AAAI Press, 2016. URL http://www.aaai.org/ocs/index.php/ICWSM/ICWSM16/paper/view/13017.

Jeffery Ansah, Lin Liu, Wei Kang, Selasi Kwashie, Jixue Li et Jiuyong Li : A graph is worth a thousand words : Telling event stories using timeline summarization graphs. *In The World Wide Web Conference, WWW 2019*, pages 2565–2571. ACM, 2019. URL https://doi.org/10.1145/3308558.3313396.

Evlampios E. Apostolidis, Eleni Adamantidou, Alexandros I. Metsai, Vasileios Mezaris et Ioannis Patras : Video summarization using deep neural networks : A survey. *Proc. IEEE*, 109(11):1838–1863, 2021. URL https://doi.org/10.1109/JPROC.2021.3117472.

Sanjeev Arora, Yingyu Liang et Tengyu Ma : A simple but tough-to-beat baseline for sentence embeddings. *In 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=SyK00v5xx.

Satanjeev Banerjee et Alon Lavie : METEOR : an automatic metric for MT evaluation with improved correlation with human judgments. *In* Jade Goldstein, Alon Lavie, Chin-Yew Lin et Clare R. Voss, éditeurs : *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization@ACL 2005, Ann Arbor, Michigan, USA, June 29, 2005*, pages 65–72. Association for Computational Linguistics, 2005. URL https://aclanthology.org/W05-0909/.

Elena Baralis, Luca Cagliero, Naeem A. Mahoto et Alessandro Fiori : Graphsum : Discovering correlations among multiple terms for graph-based summariza-

tion. *Inf. Sci.*, 249:96–109, 2013. URL https://doi.org/10.1016/j.ins.2013.06.046.

Nicholas J. Belkin et W. Bruce Croft : Information filtering and information retrieval : Two sides of the same coin ? *Commun. ACM*, 35(12):29–38, 1992. URL https://doi.org/10.1145/138859.138861.

Iz Beltagy, Matthew E. Peters et Arman Cohan : Longformer : The long-document transformer. *CoRR*, abs/2004.05150, 2020. URL https://arxiv.org/abs/2004.05150.

Tim Berners-Lee, Robert Cailliau, Jean-François Groff et Bernd Pollermann : World-wide web : The information universe. *Electron. Netw. Res. Appl. Policy*, 2(1):74–82, 1992.

Danah Michele Boyd et Nicole B. Ellison : Social network sites : Definition, history, and scholarship. *J. Comput. Mediat. Commun.*, 13(1):210–230, 2007. URL https://doi.org/10.1111/j.1083-6101.2007.00393.x.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever et Dario Amodei : Language models are few-shot learners. *In* Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan et Hsuan-Tien Lin, éditeurs : *Advances in Neural Information Processing Systems 33 : Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

Ziqiang Cao, Furu Wei, Sujian Li, Wenjie Li, Ming Zhou et Houfeng Wang : Learning summary prior representation for extractive summarization. *In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 2 : Short Papers*, pages 829–833. The Association for Computer Linguistics, 2015. URL https://doi.org/10.3115/v1/p15-2136.

Jaime G. Carbonell et Jade Goldstein : The use of mmr, diversity-based reranking for reordering documents and producing summaries. *In SIGIR '98 : Proceedings of the 21st Annual International ACM SIGIR Conference on Research*

and Development in Information Retrieval, pages 335–336. ACM, 1998. URL https://doi.org/10.1145/290941.291025.

L. Elisa Celis et Vijay Keswani : Implicit diversity in image summarization. *Proc. ACM Hum. Comput. Interact.*, 4(CSCW2):139 :1–139 :28, 2020. URL https://doi.org/10.1145/3415210.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope et Ray Kurzweil : Universal sentence encoder for english. *In* Eduardo Blanco et Wei Lu, éditeurs : *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018 : System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018*, pages 169–174. Association for Computational Linguistics, 2018. URL https://doi.org/10.18653/v1/d18-2029.

Abdelhamid Chellal et Mohand Boughanem : IRIT at TREC real-time summarization 2018. *In Proceedings of the Twenty-Seventh Text REtrieval Conference, TREC 2018*. National Institute of Standards and Technology (NIST), 2018. URL https://trec.nist.gov/pubs/trec27/papers/IRIT-RT.pdf.

Jianpeng Cheng et Mirella Lapata : Neural summarization by extracting sentences and words. *In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1 : Long Papers*. The Association for Computer Linguistics, 2016. URL https://doi.org/10.18653/v1/p16-1046.

Yves Chiaramella et Philippe Mulhem : La recherche d'information. de la documentation automatique à la recherche d'information en contexte. *Document Numérique*, 10(1):11–38, 2007. URL https://doi.org/10.3166/dn.10.11-38.

James Clarke et Mirella Lapata : Discourse constraints for document compression. *Comput. Linguistics*, 36(3):411–441, 2010. URL https://doi.org/10.1162/coli_a_00004.

Meri Coleman et T. L. Liau : A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60:283–284, 1975.

Edgar Dale et Jeanne S. Chall : A formula for predicting readability. *Educational Research Bulletin*, 27(1):11–28, 1948.

Hoa Trang Dang : Overview of duc 2005. 2005.

Hoa Trang Dang et Karolina Owczarzak : Overview of the TAC 2008 update summarization task. *In Proceedings of the First Text Analysis Conference, TAC 2008, Gaithersburg, Maryland, USA, November 17-19, 2008*. NIST, 2008.

Hal Daumé et Daniel Marcu : A noisy-channel model for document compression. *In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 449–456. ACL, 2002. URL https://aclanthology.org/P02-1057/.

Jacob Devlin, Ming-Wei Chang, Kenton Lee et Kristina Toutanova : BERT : pre-training of deep bidirectional transformers for language understanding. *In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, NAACL-HLT 2019*, pages 4171–4186. Association for Computational Linguistics, 2019. URL https://aclanthology.org/N19-1423.pdf.

Zi-Yi Dou, Pengfei Liu, Hiroaki Hayashi, Zhengbao Jiang et Graham Neubig : Gsum : A general framework for guided neural abstractive summarization. *In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, NAACL-HLT 2021*, pages 4830–4842. Association for Computational Linguistics, 2021. URL https://doi.org/10.18653/v1/2021.naacl-main.384.

Yajuan Duan, Zhimin Chen, Furu Wei, Ming Zhou et Heung-Yeung Shum : Twitter topic summarization by ranking tweets using social influence and content quality. *In 24th Int. Conf. COLING, Technical Papers*, pages 763–780. The Association for Computer Linguistics, 2012. URL https://www.aclweb.org/anthology/C12-1047/.

Alexis Dusart, Karen Pinel-Sauvagnat et Gilles Hubert : Issumset : a tweet summarization dataset hidden in a TREC track. *In SAC '21 : The 36th ACM/-SIGAPP Symposium on Applied Computing, Virtual Event, Republic of Korea, March 22-26, 2021*, pages 665–671. ACM, 2021a. URL https://doi.org/10.1145/3412841.3441946.

Alexis Dusart, Karen Pinel-Sauvagnat et Gilles Hubert : Tssubert : Tweet stream summarization using BERT. *CoRR*, abs/2106.08770, 2021b. URL https://arxiv.org/abs/2106.08770.

Soumi Dutta, Vibhash Chandra, Kanav Mehra, Asit Kumar Das, Tanmoy Chakraborty et Saptarshi Ghosh : Ensemble algorithms for microblog summarization. *IEEE Intell. Syst.*, 33(3):4–14, 2018. URL https://doi.org/10.1109/MIS.2018.033001411.

Wafaa S. El-Kassas, Cherif R. Salama, Ahmed A. Rafea et Hoda K. Mohamed : Automatic text summarization : A comprehensive survey. *Expert Systems with Applications*, 165:113679, 2021. URL http://www.sciencedirect.com/science/article/pii/S0957417420305030.

Günes Erkan et Dragomir R. Radev : Lexpagerank : Prestige in multi-document text summarization. *In Conf. EMNLP 2004*, pages 365–371, 2004.

Liana Ermakova, Frederique Bordignon, Nicolas Turenne et Marianne Noel : Is the abstract a mere teaser ? evaluating generosity of article abstracts in the environmental sciences. *Frontiers Res. Metrics Anal.*, 3:16, 2018. URL https://doi.org/10.3389/frma.2018.00016.

Liana Ermakova, Jean-Valère Cossu et Josiane Mothe : A survey on evaluation of summarization methods. *Inf. Process. Manag.*, 56(5):1794–1814, 2019. URL https://doi.org/10.1016/j.ipm.2019.04.001.

Liana Ermakova et Anton Firsov : GRAD : A metric for evaluating summaries. *In CORIA 2018, 15th French Information Retrieval Conference*, 2018.

Alexander Fabbri, Irene Li, Tianwei She, Suyi Li et Dragomir Radev : Multi-news : A large-scale multi-document summarization dataset and abstractive hierarchical model. *In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084. Association for Computational Linguistics, 2019. URL https://aclanthology.org/P19-1102.

Alexander R. Fabbri, Wojciech Kryscinski, Bryan McCann, Caiming Xiong, Richard Socher et Dragomir R. Radev : Summeval : Re-evaluating summarization evaluation. *Trans. Assoc. Comput. Linguistics*, 9:391–409, 2021. URL https://doi.org/10.1162/tacl_a_00373.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard H. Hovy et Noah A. Smith : Retrofitting word vectors to semantic lexicons. *In* Rada Mihalcea, Joyce Yue Chai et Anoop Sarkar, éditeurs : *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 1606–1615. The Association for Computational Linguistics, 2015. URL https://doi.org/10.3115/v1/n15-1184.

Mohamed Abdel Fattah et Fuji Ren : Ga, mr, ffnn, PNN and GMM based models for automatic text summarization. *Comput. Speech Lang.*, 23(1):126–144, 2009. URL https://doi.org/10.1016/j.csl.2008.04.002.

Javi Fernández, Fernando Llopis, Yoan Gutiérrez, Patricio Martínez-Barco, José M. Gómez et Rafael Muñoz : GPLSI at TREC 2018 RTS track. *In Proceedings of the Twenty-Seventh Text REtrieval Conference, TREC 2018*. National Institute of Standards and Technology (NIST), 2018. URL https://trec.nist.gov/pubs/trec27/papers/UA-GPLSI-RT.pdf.

Rafael Ferreira, Luciano de Souza Cabral, Frederico Luiz Gonçalves de Freitas, Rafael Dueire Lins, Gabriel de França Pereira e Silva, Steven J. Simske

et Luciano Favaro : A multi-document summarization system based on statistics and linguistic treatment. *Expert Syst. Appl.*, 41(13):5780–5787, 2014. URL https://doi.org/10.1016/j.eswa.2014.03.023.

Mahak Gambhir et Vishal Gupta : Recent automatic text summarization techniques : a survey. *Artif. Intell. Rev.*, 47(1):1–66, 2017. URL https://doi.org/10.1007/s10462-016-9475-9.

Kavita Ganesan, ChengXiang Zhai et Jiawei Han : Opinosis : A graph based approach to abstractive summarization of highly redundant opinions. *In* Chu-Ren Huang et Dan Jurafsky, éditeurs : *COLING 2010, 23rd International Conference on Computational Linguistics, Proceedings of the Conference, 23-27 August 2010, Beijing, China*, pages 340–348. Tsinghua University Press, 2010. URL https://aclanthology.org/C10-1039/.

Pierre-Etienne Genest et Guy Lapalme : Framework for abstractive summarization using text-to-text generation. *In Proceedings of the Workshop on Monolingual Text-To-Text Generation*, MTTG '11, page 64–73, USA, 2011. Association for Computational Linguistics. ISBN 9781937284053. URL https://dl.acm.org/doi/10.5555/2107679.2107687.

Samuel Gershman et Joshua B. Tenenbaum : Phrase similarity in humans and machines. *In* David C. Noelle, Rick Dale, Anne S. Warlaumont, Jeff Yoshimi, Teenie Matlock, Carolyn D. Jennings et Paul P. Maglio, éditeurs : *Proceedings of the 37th Annual Meeting of the Cognitive Science Society, CogSci 2015, Pasadena, California, USA, July 22-25, 2015*. cognitivesciencesociety.org, 2015. URL https://mindmodeling.org/cogsci2015/papers/0141/index.html.

Demian Gholipour Ghalandari, Chris Hokamp, Nghia The Pham, John Glover et Georgiana Ifrim : A large-scale multi-document summarization dataset from the wikipedia current events portal. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 1302–1308. Association for Computational Linguistics, 2020. URL https://aclanthology.org/2020.acl-main.120.pdf.

Jade Goldstein, Vibhu Mittal, Jaime Carbonell et Mark Kantrowitz : Multi-document summarization by sentence extraction. *In Proceedings of the 2000 NAACL-ANLPWorkshop on Automatic Summarization - Volume 4*, NAACL-ANLP-AutoSum '00, page 40–48, USA, 2000. Association for Computational Linguistics.

David Graff, Junbo Kong, Ke Chen et Kazuaki Maeda : English gigaword. *Linguistic Data Consortium, Philadelphia*, 4(1):34, 2003.

Barbara J. Grosz, Aravind K. Joshi et Scott Weinstein : Centering : A framework for modeling the local coherence of discourse. *Comput. Linguistics*, 21(2):203–225,

1995. URL https://repository.upenn.edu/cgi/viewcontent.cgi?article=1118&context=ircs_reports.

R. Gunning : *The technique of clear writing*. McGraw-Hill, 1968.

Vishal Gupta : Hybrid algorithm for multilingual summarization of hindi and punjabi documents. *In* Rajendra Prasath et T. Kathirvalavakumar, éditeurs : *Mining Intelligence and Knowledge Exploration*, pages 717–727, Cham, 2013. Springer International Publishing. ISBN 978-3-319-03844-5.

Uri Hanani, Bracha Shapira et Peretz Shoval : Information filtering : Overview of issues, research and systems. *User Model. User Adapt. Interact.*, 11(3):203–259, 2001. URL https://doi.org/10.1023/A:1011196000674.

Sanda M. Harabagiu et Andrew Hickl : Relevance modeling for microblog summarization. *In* Lada A. Adamic, Ricardo Baeza-Yates et Scott Counts, éditeurs : *Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011*. The AAAI Press, 2011. URL http://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/view/2863.

Sanda M. Harabagiu et V. Finley Lacatusu : Topic themes for multi-document summarization. *In* Ricardo A. Baeza-Yates, Nivio Ziviani, Gary Marchionini, Alistair Moffat et John Tait, éditeurs : *SIGIR 2005 : Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Salvador, Brazil, August 15-19, 2005*, pages 202–209. ACM, 2005. URL https://doi.org/10.1145/1076034.1076071.

A Heinonen : The journalist s relationship with users. new dimensions to conventional roles, jane b. singer et al.(eds), participatory journalism. guarding open gates at online newspapers, 2011.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman et Phil Blunsom : Teaching machines to read and comprehend. *In Advances in neural information processing systems*, pages 1693–1701, 2015.

Alfred Hermida : Tweets and truth : Journalism as a discipline of collaborative verification. *Journalism Practice*, 6(5-6):659–668, 2012.

Eduard H. Hovy, Chin-Yew Lin, Liang Zhou et Junichi Fukumoto : Automated summarization evaluation with basic elements. *In* Nicoletta Calzolari, Khalid Choukri, Aldo Gangemi, Bente Maegaard, Joseph Mariani, Jan Odijk et Daniel Tapias, éditeurs : *Proceedings of the Fifth International Conference on Language Resources and Evaluation, LREC 2006, Genoa, Italy, May 22-28, 2006*, pages 899–902. European Language Resources Association (ELRA), 2006. URL http://www.lrec-conf.org/proceedings/lrec2006/summaries/438.html.

Lei Huang, Yanxiang He, Furu Wei et Wenjie Li : Modeling document summarization as multi-objective optimization. *In Third International Symposium on Intelligent Information Technology and Security Informatics, IITSI 2010, Jinggangshan, China, April 2-4, 2010*, pages 382–386. IEEE Computer Society, 2010.

Muhammad Imran, Shady Elbassuoni, Carlos Castillo, Fernando Diaz et Patrick Meier : Extracting information nuggets from disaster- related messages in social media. *In* Tina Comes, Frank Fiedrich, Simon Fortier, Jutta Geldermann et Tim Müller, éditeurs : *10th Proceedings of the International Conference on Information Systems for Crisis Response and Management, Baden-Baden, Germany, May 12-15, 2013*. ISCRAM Association, 2013.

Muhammad Imran, Prasenjit Mitra et Carlos Castillo : Twitter as a lifeline : Human-annotated twitter corpora for NLP of crisis-related messages. *In* Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk et Stelios Piperidis, éditeurs : *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016*. European Language Resources Association (ELRA), 2016.

David I. Inouye et Jugal K. Kalita : Comparing twitter summarization algorithms for multiple post summaries. *In PASSAT/SocialCom 2011, Privacy, Security, Risk and Trust (PASSAT), 2011 IEEE Third International Conference on and 2011 IEEE Third International Conference on Social Computing (SocialCom), Boston, MA, USA, 9-11 Oct., 2011*, pages 298–306. IEEE Computer Society, 2011. URL `https://doi.org/10.1109/PASSAT/SocialCom.2011.31`.

Mohit Iyyer, Varun Manjunatha, Jordan L. Boyd-Graber et Hal Daumé III : Deep unordered composition rivals syntactic methods for text classification. *In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1 : Long Papers*, pages 1681–1691. The Association for Computer Linguistics, 2015. URL `https://doi.org/10.3115/v1/p15-1162`.

Tom Kenter, Alexey Borisov et Maarten de Rijke : Siamese CBOW : optimizing word embeddings for sentence representations. *In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1 : Long Papers*. The Association for Computer Linguistics, 2016. URL `https://doi.org/10.18653/v1/p16-1089`.

Tom Kenter et Maarten de Rijke : Short text similarity with word embeddings. *In* James Bailey, Alistair Moffat, Charu C. Aggarwal, Maarten de Rijke, Ravi

KUMAR, Vanessa MURDOCK, Timos K. SELLIS et Jeffrey Xu YU, éditeurs : *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015*, pages 1411–1420. ACM, 2015. URL https://dl.acm.org/citation.cfm?id=2806475.

Yoon KIM : Convolutional neural networks for sentence classification. *In* Alessandro MOSCHITTI, Bo PANG et Walter DAELEMANS, éditeurs : *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1746–1751. ACL, 2014. URL https://doi.org/10.3115/v1/d14-1181.

J.P. KINCAID : *Derivation of New Readability Formulas : (automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel.* Research Branch report. Chief of Naval Technical Training, Naval Air Station Memphis, 1975.

George R. KLARE : Assessing readability. *Reading Research Quarterly*, 10(1):62–102, 1974.

Youngjoong KO et Jungyun SEO : An effective sentence-extraction technique using contextual information and statistical approaches for text summarization. *Pattern Recognit. Lett.*, 29(9):1366–1371, 2008. URL https://doi.org/10.1016/j.patrec.2008.02.008.

Mike LEWIS, Yinhan LIU, Naman GOYAL, Marjan GHAZVININEJAD, Abdelrahman MOHAMED, Omer LEVY, Veselin STOYANOV et Luke ZETTLEMOYER : BART : denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 7871–7880. Association for Computational Linguistics, 2020. URL https://aclanthology.org/2020.acl-main.703/.

Daniel LI, Thomas CHEN, Albert TUNG et Lydia B. CHILTON : Hierarchical summarization for longform spoken dialog. *In* Jeffrey NICHOLS, Ranjitha KUMAR et Michael NEBELING, éditeurs : *UIST '21 : The 34th Annual ACM Symposium on User Interface Software and Technology, Virtual Event, USA, October 10-14, 2021*, pages 582–597. ACM, 2021. URL https://doi.org/10.1145/3472749.3474771.

Quanzhi LI et Qiong ZHANG : Abstractive event summarization on twitter. *In Companion of The 2020 Web Conference 2020*, pages 22–23. ACM / IW3C2, 2020. URL https://doi.org/10.1145/3366424.3382678.

Quanzhi LI et Qiong ZHANG : Twitter event summarization by exploiting semantic terms and graph network. *In Proceedings of the The Thirty-Third Annual*

Conference on Innovative Applications of Artificial Intelligence (IAAI-21), pages 15347–15354. AAAI Press, 2021. URL https://ojs.aaai.org/index.php/AAAI/article/view/17802.

Rensis Likert : A technique for the measurement of attitudes. *Archives of psychology*, 1932.

Chin-Yew Lin : ROUGE : A package for automatic evaluation of summaries. *Text Summarization Branches Out*, pages 74–81, juillet 2004. URL https://www.aclweb.org/anthology/W04-1013.

Jimmy Lin, Salman Mohammed, Royal Sequiera, Luchen Tan, Nimesh Ghelani, Mustafa Abualsaud, Richard McCreadie, Dmitrijs Milajevs et Ellen M. Voorhees : Overview of the TREC 2017 real-time summarization track. *In Proceedings of The Twenty-Sixth Text REtrieval Conference, TREC 2017*. National Institute of Standards and Technology (NIST), 2017. URL https://trec.nist.gov/pubs/trec26/papers/Overview-RT.pdf.

Jimmy Lin, Adam Roegiest, Luchen Tan, Richard McCreadie, Ellen M. Voorhees et Fernando Diaz : Overview of the TREC 2016 real-time summarization track. *In Proceedings of The Twenty-Fifth Text REtrieval Conference, TREC 2016*. National Institute of Standards and Technology (NIST), 2016. URL https://trec.nist.gov/pubs/trec25/papers/Overview-RT.pdf.

Xiaohua Liu, Yitong Li, Furu Wei et Ming Zhou : Graph-based multi-tweet summarization using social signals. *In 24th Int. Conf. COLING, Technical Papers*, pages 1699–1714. The Association for Computer Linguistics, 2012. URL https://www.aclweb.org/anthology/C12-1104/.

Yang Liu et Mirella Lapata : Text summarization with pretrained encoders. *In* Kentaro Inui, Jing Jiang, Vincent Ng et Xiaojun Wan, éditeurs : *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3728–3738. Association for Computational Linguistics, 2019.

Elena Lloret, Laura Plaza et Ahmet Aker : The challenging task of summary evaluation : an overview. *Lang. Resour. Evaluation*, 52(1):101–148, 2018. URL https://doi.org/10.1007/s10579-017-9399-2.

Elena Lloret, María Teresa Romá-Ferri et Manuel Palomar : COMPENDIUM : A text summarization system for generating abstracts of research papers. *In* Rafael Muñoz, Andrés Montoyo et Elisabeth Métais, éditeurs : *Natural Language Processing and Information Systems - 16th International Conference on Applications of Natural Language to Information Systems, NLDB 2011, Alicante, Spain,*

June 28-30, 2011. Proceedings, volume 6716 de *Lecture Notes in Computer Science*, pages 3–14. Springer, 2011. URL https://doi.org/10.1007/978-3-642-22327-3_2.

Annie Louis et Ani Nenkova : Automatically assessing machine summary content without a gold standard. *Computational Linguistics*, 39(2):267–300, juin 2013. URL https://aclanthology.org/J13-2002.

Hans Peter Luhn : The automatic creation of literature abstracts. *IBM J. Res. Dev.*, 2(2):159–165, 1958. URL https://doi.org/10.1147/rd.22.0159.

Congbo Ma, Wei Emma Zhang, Mingyu Guo, Hu Wang et QUAN Z. Sheng : Multi-document summarization via deep learning techniques : A survey. *ACM Comput. Surv.*, mar 2022. ISSN 0360-0300. URL https://doi.org/10.1145/3529754. Just Accepted.

Inderjeet Mani, Gary Klein, David House, Lynette Hirschman, Therese Firmin et Beth Sundheim : SUMMAC : a text summarization evaluation. *Nat. Lang. Eng.*, 8(1):43–68, 2002. URL https://doi.org/10.1017/S1351324901002741.

William Mann et Sandra Thompson : Rethorical structure theory : Toward a functional theory of text organization. *Text*, 8:243–281, 01 1988.

Christopher D. Manning, Prabhakar Raghavan et Hinrich Schütze : *Introduction to information retrieval*. Cambridge University Press, 2008.

Sebastian Martschat et Katja Markert : A temporally sensitive submodularity framework for timeline summarization. *In Proceedings of the 22nd Conference on Computational Natural Language Learning, CoNLL 2018*, pages 230–240. Association for Computational Linguistics, 2018. URL https://doi.org/10.18653/v1/k18-1023.

Mark T. Maybury : Generating summaries from event data. *Inf. Process. Manag.*, 31(5):735–751, 1995. URL https://doi.org/10.1016/0306-4573(95)00025-C.

Richard McCreadie, Cody Buntain et Ian Soboroff : TREC incident streams : Finding actionable information on social media. *In* Zeno Franco, José J. González et José H. Canós, éditeurs : *Proceedings of the 16th International Conference on Information Systems for Crisis Response and Management, València, Spain, May 19-22, 2019*. ISCRAM Association, 2019.

Richard McCreadie, Rodrygo L. T. Santos, Craig Macdonald et Iadh Ounis : Explicit diversification of event aspects for temporal summarization. *ACM Trans. Inf. Syst.*, 36(3):25 :1–25 :31, 2018. URL https://doi.org/10.1145/3158671.

Rada Mihalcea et Paul Tarau : Textrank : Bringing order into text. *In Conference on Empirical Methods in Natural Language Processing , EMNLP*, pages 404–411, 2004.

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado et Jeffrey Dean : Distributed representations of words and phrases and their compositionality. *In* Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani et Kilian Q. Weinberger, éditeurs : *Advances in Neural Information Processing Systems 26 : 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119, 2013. URL https://proceedings.neurips.cc/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html.

Maali Mnasri : *Résumé automatique multi-document dynamique*. Theses, Université Paris-Saclay, septembre 2018. URL https://tel.archives-ouvertes.fr/tel-01902781.

Jane Morris et Graeme Hirst : Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Comput. Linguistics*, 17(1):21–48, 1991. URL https://aclanthology.org/J91-1002.pdf.

Ramesh Nallapati, Feifei Zhai et Bowen Zhou : Summarunner : A recurrent neural network based sequence model for extractive summarization of documents. *In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 3075–3081. AAAI Press, 2017. URL http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14636.

Medhini Narasimhan, Anna Rohrbach et Trevor Darrell : Clip-it ! language-guided video summarization. *In* Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang et Jennifer Wortman Vaughan, éditeurs : *Advances in Neural Information Processing Systems 34 : Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 13988–14000, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/7503cfacd12053d309b6bed5c89de212-Abstract.html.

Shashi Narayan, Shay B. Cohen et Mirella Lapata : Don't give me the details, just the summary ! topic-aware convolutional neural networks for extreme summarization. *In* Ellen Riloff, David Chiang, Julia Hockenmaier et Jun'ichi Tsujii, éditeurs : *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1797–1807. Association for Computational Linguistics, 2018a.

Shashi Narayan, Shay B. Cohen et Mirella Lapata : Ranking sentences for extractive summarization with reinforcement learning. *In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational*

Linguistics : Human Language Technologies, NAACL-HLT 2018, pages 1747–1759. Association for Computational Linguistics, 2018b. URL https://aclanthology.org/N18-1158/.

Ani Nenkova et Rebecca Passonneau : Evaluating content selection in summarization : The pyramid method. *In Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics : HLT-NAACL 2004*, pages 145–152, Boston, Massachusetts, USA, mai 2 - mai 7 2004. Association for Computational Linguistics. URL https://aclanthology.org/N04-1019.

Dat Quoc Nguyen, Thanh Vu et Anh Tuan Nguyen : Bertweet : A pre-trained language model for english tweets. *In* Qun Liu et David Schlangen, éditeurs : *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing : System Demonstrations, EMNLP 2020 - Demos, Online, November 16-20, 2020*, pages 9–14. Association for Computational Linguistics, 2020. URL https://doi.org/10.18653/v1/2020.emnlp-demos.2.

Minh-Tien Nguyen, Lai Dac Viet, Huy-Tien Nguyen et Minh-Le Nguyen : TSix : A Human-involved-creation Dataset for Tweet Summarization. *In 11th Int. Conf. on Language Resources and Evaluation, LREC*, pages 3204–3208, 2018.

Jeffrey Nichols, Jalal Mahmud et Clemens Drews : Summarizing sporting events using twitter. *In* Carlos Duarte, Luís Carriço, Joaquim A. Jorge, Sharon L. Oviatt et Daniel Gonçalves, éditeurs : *17th International Conference on Intelligent User Interfaces, IUI 2012, Lisbon, Portugal, February 14-17, 2012*, pages 189–198. ACM, 2012. URL https://doi.org/10.1145/2166966.2166999.

Andrei Olariu : Efficient online summarization of microblogging streams. *In* Gosse Bouma et Yannick Parmentier, éditeurs : *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2014, April 26-30, 2014, Gothenburg, Sweden*, pages 236–240. The Association for Computer Linguistics, 2014.

Alexandra Olteanu, Sarah Vieweg et Carlos Castillo : What to expect when the unexpected happens : Social media communications across crises. *In* Dan Cosley, Andrea Forte, Luigina Ciolfi et David McDonald, éditeurs : *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing, CSCW 2015, Vancouver, BC, Canada, March 14 - 18, 2015*, pages 994–1009. ACM, 2015.

Lawrence Page, Sergey Brin, Rajeev Motwani et Terry Winograd : The pagerank citation ranking : Bringing order to the web. Rapport technique, Stanford InfoLab, 1999.

Kishore Papineni, Salim Roukos, Todd Ward et Wei-Jing Zhu : Bleu : a method for automatic evaluation of machine translation. *In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. ACL, 2002. URL https://aclanthology.org/P02-1040/.

Darshna Patel, Saurabh Shah et Hitesh Chhinkaniwala : Fuzzy logic based multi document summarization with improved sentence scoring and redundancy removal technique. *Expert Syst. Appl.*, 134:167–177, 2019. URL https://doi.org/10.1016/j.eswa.2019.05.045.

Romain Paulus, Caiming Xiong et Richard Socher : A deep reinforced model for abstractive summarization. *In International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=HkAClQgA-.

Maxime Peyrard et Iryna Gurevych : Objective function learning to match human judgements for optimization-based summarization. *In* Marilyn A. Walker, Heng Ji et Amanda Stent, éditeurs : *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 654–660. Association for Computational Linguistics, 2018. URL https://doi.org/10.18653/v1/n18-2103.

Weizhen Qi, Yu Yan, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang et Ming Zhou : Prophetnet : Predicting future n-gram for sequence-to-sequence pre-training. *In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing : Findings, EMNLP 2020*, pages 2401–2410. Association for Computational Linguistics, 2020. URL https://aclanthology.org/2020.findings-emnlp.217.pdf.

Dragomir R. Radev, Eduard H. Hovy et Kathleen R. McKeown : Introduction to the special issue on summarization. *Comput. Linguistics*, 28(4):399–408, 2002. URL https://doi.org/10.1162/089120102762671927.

Dragomir R. Radev, Hongyan Jing, Magorzata Sty et Daniel Tam : Centroid-based summarization of multiple documents. *Inf. Process. Manag.*, 40(6):919–938, 2004.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li et Peter J. Liu : Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140 :1–140 :67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

Jinfeng Rao, Wei Yang, Yuhao Zhang, Ferhan Türe et Jimmy Lin : Multi-perspective relevance matching with hierarchical convnets for social media search.

In The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019, pages 232–240. AAAI Press, 2019. URL https://doi.org/10.1609/aaai.v33i01.3301232.

Nils Reimers et Iryna Gurevych : Sentence-bert : Sentence embeddings using siamese bert-networks. *In* Kentaro Inui, Jing Jiang, Vincent Ng et Xiaojun Wan, éditeurs : *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics, 2019.

Pengjie Ren, Zhumin Chen, Zhaochun Ren, Furu Wei, Jun Ma et Maarten de Rijke : Leveraging contextual sentence relations for extractive summarization using a neural attention model. *In* Noriko Kando, Tetsuya Sakai, Hideo Joho, Hang Li, Arjen P. de Vries et Ryen W. White, éditeurs : *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*, pages 95–104. ACM, 2017. URL https://doi.org/10.1145/3077136.3080792.

Koustav Rudra, Niloy Ganguly, Pawan Goyal et Saptarshi Ghosh : Extracting and summarizing situational information from the twitter social media during disasters. *TWEB*, 12(3):17 :1–17 :35, 2018a.

Koustav Rudra, Subham Ghosh, Niloy Ganguly, Pawan Goyal et Saptarshi Ghosh : Extracting situational information from microblogs during disaster events : a classification-summarization approach. *In 24th ACM Int. Conf. CIKM*, pages 583–592, 2015.

Koustav Rudra, Pawan Goyal, Niloy Ganguly, Muhammad Imran et Prasenjit Mitra : Summarizing situational tweets in crisis scenarios : An extractive-abstractive approach. *IEEE Trans. Comput. Soc. Syst.*, 6(5):981–993, 2019.

Koustav Rudra, Pawan Goyal, Niloy Ganguly, Prasenjit Mitra et Muhammad Imran : Identifying sub-events and summarizing disaster-related information from microblogs. *In The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018*, pages 265–274. ACM, 2018b. URL https://dl.acm.org/doi/10.1145/3209978.3210030.

Alexander M. Rush, Sumit Chopra et Jason Weston : A neural attention model for abstractive sentence summarization. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, 2015. URL http://dx.doi.org/10.18653/v1/D15-1044.

Naveen Saini, Sriparna Saha et Pushpak Bhattacharyya : Multiobjective-based approach for microblog summarization. *IEEE Trans. Comput. Soc. Syst.*, 6(6): 1219–1231, 2019. URL https://doi.org/10.1109/TCSS.2019.2945172.

Jesús M. Sánchez-Gómez, Miguel A. Vega-Rodríguez et Carlos J. Pérez : Extractive multi-document text summarization using a multi-objective artificial bee colony optimization approach. *Knowl. Based Syst.*, 159:1–8, 2018. URL https://doi.org/10.1016/j.knosys.2017.11.029.

Jesús M. Sánchez-Gómez, Miguel A. Vega-Rodríguez et Carlos J. Pérez : Experimental analysis of multiple criteria for extractive multi-document text summarization. *Expert Syst. Appl.*, 140, 2020. URL https://doi.org/10.1016/j.eswa.2019.112904.

Mark Sanderson et W. Bruce Croft : The history of information retrieval research. *Proc. IEEE*, 100(Centennial-Issue):1444–1451, 2012. URL https://doi.org/10.1109/JPROC.2012.2189916.

Victor Sanh, Lysandre Debut, Julien Chaumond et Thomas Wolf : Distilbert, a distilled version of BERT : smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019. URL http://arxiv.org/abs/1910.01108.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, Jacopo Staiano, Alex Wang et Patrick Gallinari : Questeval : Summarization asks for fact-based evaluation. *In* Marie-Francine Moens, Xuanjing Huang, Lucia Specia et Scott Wen-tau Yih, éditeurs : *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 6594–6604. Association for Computational Linguistics, 2021. URL https://doi.org/10.18653/v1/2021.emnlp-main.529.

Abigail See, Peter J. Liu et Christopher D. Manning : Get to the point : Summarization with pointer-generator networks. *In* Regina Barzilay et Min-Yen Kan, éditeurs : *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1 : Long Papers*, pages 1073–1083. Association for Computational Linguistics, 2017.

Royal Sequiera, Luchen Tan et Jimmy Lin : Overview of the TREC 2018 real-time summarization track. *In Proceedings of the Twenty-Seventh Text REtrieval Conference, TREC 2018*. National Institute of Standards and Technology (NIST), 2018. URL https://trec.nist.gov/pubs/trec27/papers/Overview-RTS.pdf.

Beaux Sharifi, Mark-Anthony Hutton et Jugal K. Kalita : Summarizing microblogs automatically. *In Human Language Technologies : Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings,*

June 2-4, 2010, Los Angeles, California, USA, pages 685–688. The Association for Computational Linguistics, 2010. URL https://aclanthology.org/N10-1100/.

Beaux SHARIFI, David I. INOUYE et Jugal K. KALITA : Summarization of twitter microblogs. *Comput. J.*, 57(3):378–402, 2014. URL https://doi.org/10.1093/comjnl/bxt109.

Lidan SHOU, Zhenhua WANG, Ke CHEN et Gang CHEN : Sumblr : continuous summarization of evolving tweet streams. *In* Gareth J. F. JONES, Paraic SHERIDAN, Diane KELLY, Maarten de RIJKE et Tetsuya SAKAI, éditeurs : *The 36th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR '13, Dublin, Ireland - July 28 - August 01, 2013*, pages 533–542. ACM, 2013.

E.A. SMITH, R.J. SENTER et Air Force Aerospace Medical Research Laboratory (U.S.) : *Automated Readability Index*. AMRL-TR. Aerospace Medical Research Laboratories, 1967.

Kaiqiang SONG, Bingqing WANG, Zhe FENG, Ren LIU et Fei LIU : Controlling the amount of verbatim copying in abstractive summarization. *In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8902–8909. AAAI Press, 2020. URL https://par.nsf.gov/servlets/purl/10191761.

George SPACHE : A new readability formula for primary-grade reading materials. *The Elementary School Journal*, 53(7):410–413, 1953.

Nitish SRIVASTAVA, Geoffrey E. HINTON, Alex KRIZHEVSKY, Ilya SUTSKEVER et Ruslan SALAKHUTDINOV : Dropout : a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, 2014. URL https://jmlr.org/papers/volume15/srivastava14a/srivastava14a.pdf.

Emma STRUBELL, Ananya GANESH et Andrew MCCALLUM : Energy and policy considerations for deep learning in NLP. *In* Anna KORHONEN, David R. TRAUM et Lluís MÀRQUEZ, éditeurs : *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1 : Long Papers*, pages 3645–3650. Association for Computational Linguistics, 2019.

Ladda SUANMALI, Naomie SALIM et Mohammed Salem BINWAHLAN : Fuzzy logic based method for improving text summarization. *CoRR*, abs/0906.4690, 2009. URL http://arxiv.org/abs/0906.4690.

Haihui TAN, Ziyu LU et Wenjie LI : Neural network based reinforcement learning for real-time pushing on text stream. *In* Noriko KANDO, Tetsuya SAKAI, Hideo JOHO, Hang LI, Arjen P. de VRIES et Ryen W. WHITE, éditeurs : *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*, pages 913–916. ACM, 2017. URL https://doi.org/10.1145/3077136.3080677.

E. THORSEN : *Live Blogging and Social Media Curation : Challenges and Opportunities for Journalism*. CJCR : Centre for Journalism & Communication Research, Bournemouth University, 2013.

Giang Binh TRAN, Mohammad ALRIFAI et Eelco HERDER : Timeline summarization from relevant headlines. *In Advances in Information Retrieval - 37th European Conference on IR Research, ECIR 2015*, pages 245–256. Springer International Publishing, 2015. URL https://doi.org/10.1007/978-3-319-16354-3_26.

Julián URBANO, Mónica MARRERO et Diego MARTÍN : A comparison of the optimality of statistical significance tests for information retrieval evaluation. *In The 36th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR '13*, pages 925–928. ACM, 2013. URL https://doi.org/10.1145/2484028.2484163.

Alper Kursat UYSAL et Serkan GÜNAL : The impact of preprocessing on text classification. *Inf. Process. Manag.*, 50(1):104–112, 2014. URL https://doi.org/10.1016/j.ipm.2013.08.006.

Sebastián VALENZUELA, Soledad PUENTE et Pablo M FLORES : Comparing disaster news on twitter and television : An intermedia agenda setting perspective. *Journal of Broadcasting & Electronic Media*, 61(4):615–637, 2017. URL https://www.tandfonline.com/doi/abs/10.1080/08838151.2017.1344673.

Ashish VASWANI, Noam SHAZEER, Niki PARMAR, Jakob USZKOREIT, Llion JONES, Aidan N. GOMEZ, Lukasz KAISER et Illia POLOSUKHIN : Attention is all you need. *In Advances in Neural Information Processing Systems 30 : Annual Conference on Neural Information Processing Systems 2017*, pages 5998–6008. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Soroush VOSOUGHI, Mostafa 'Neo' MOHSENVAND et Deb ROY : Rumor gauge : Predicting the veracity of rumors on twitter. *ACM transactions on knowledge discovery from data (TKDD)*, 11(4):1–36, 2017.

Soroush VOSOUGHI, Deb ROY et Sinan ARAL : The spread of true and false news online. *Science*, 359(6380):1146–1151, 2018. URL https://www.science.org/doi/abs/10.1126/science.aap9559.

Dingding WANG et Tao LI : Document update summarization using incremental hierarchical clustering. *In Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010*, pages 279–288. ACM, 2010. URL https://doi.org/10.1145/1871437.1871476.

Lili WANG, Chongyang GAO, Jason WEI, Weicheng MA, Ruibo LIU et Soroush VOSOUGHI : An empirical survey of unsupervised text representation methods on twitter data. *In* Wei XU, Alan RITTER, Tim BALDWIN et Afshin RAHIMI, éditeurs : *Proceedings of the Sixth Workshop on Noisy User-generated Text, W-NUT@EMNLP 2020 Online, November 19, 2020*, pages 209–214. Association for Computational Linguistics, 2020a. URL https://doi.org/10.18653/v1/2020.wnut-1.27.

Wenhui WANG, Furu WEI, Li DONG, Hangbo BAO, Nan YANG et Ming ZHOU : Minilm : Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *In* Hugo LAROCHELLE, Marc'Aurelio RANZATO, Raia HADSELL, Maria-Florina BALCAN et Hsuan-Tien LIN, éditeurs : *Advances in Neural Information Processing Systems 33 : Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020b. URL https://proceedings.neurips.cc/paper/2020/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Zhongqing WANG et Yue ZHANG : A neural model for joint event detection and summarization. *In* Carles SIERRA, éditeur : *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 4158–4164. ijcai.org, 2017. URL https://doi.org/10.24963/ijcai.2017/581.

Matheus WERNER et Eduardo LABER : Speeding up word mover's distance and its variants via properties of distances between embeddings. *In ECAI 2020 - 24th European Conference on Artificial Intelligence*, pages 2204–2211. IOS Press, 2020. URL https://ecai2020.eu/papers/888_paper.pdf.

Wen XIAO, Iz BELTAGY, Giuseppe CARENINI et Arman COHAN : PRIMERA : pyramid-based masked sentence pre-training for multi-document summarization. *In* Smaranda MURESAN, Preslav NAKOV et Aline VILLAVICENCIO, éditeurs : *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 5245–5263. Association for Computational Linguistics, 2022. URL https://aclanthology.org/2022.acl-long.360.

Rui YAN, Xiaojun WAN, Jahna OTTERBACHER, Liang KONG, Xiaoming LI et Yan ZHANG : Evolutionary timeline summarization : a balanced optimization framework via iterative substitution. *In Proceeding of the 34th International ACM*

SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2011, pages 745–754. ACM, 2011. URL https://doi.org/10.1145/2009916.2010016.

Min Yang, Chengming Li, Fei Sun, Zhou Zhao, Ying Shen et Chenglin Wu : Be relevant, non-redundant, and timely : Deep reinforcement learning for real-time event summarization. *In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9410–9417. AAAI Press, 2020. URL https://ojs.aaai.org/index.php/AAAI/article/view/6483.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov et Quoc V. Le : Xlnet : Generalized autoregressive pretraining for language understanding. *In* Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox et Roman Garnett, éditeurs : *Advances in Neural Information Processing Systems 32 : Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/dc6a7e655d7e5840e66733e9ee67cc69-Abstract.html.

Jin-ge Yao, Xiaojun Wan et Jianguo Xiao : Compressive document summarization via sparse optimization. *In Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, page 1376–1382. AAAI Press, 2015. ISBN 9781577357384. URL https://dl.acm.org/doi/10.5555/2832415.2832441.

Jen-Yuan Yeh, Hao-Ren Ke, Wei-Pang Yang et I-Heng Meng : Text summarization using a trainable summarizer and latent semantic analysis. *Inf. Process. Manag.*, 41(1):75–95, 2005. URL https://doi.org/10.1016/j.ipm.2004.04.003.

Fabio Massimo Zanzotto, Marco Pennacchiotti et Kostas Tsioutsiouliklis : Linguistic redundancy in twitter. *In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27-31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 659–669. ACL, 2011. URL https://aclanthology.org/D11-1061/.

Jingqing Zhang, Yao Zhao, Mohammad Saleh et Peter J. Liu : PEGASUS : pre-training with extracted gap-sentences for abstractive summarization. *In Proceedings of the 37th International Conference on Machine Learning, ICML 2020*, pages 11328–11339. PMLR, 2020a. URL http://proceedings.mlr.press/v119/zhang20ae/zhang20ae.pdf.

Tianyi ZHANG, Varsha KISHORE, Felix WU, Kilian Q. WEINBERGER et Yoav ARTZI : Bertscore : Evaluating text generation with BERT. *In 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020b. URL https://openreview.net/forum?id=SkeHuCVFDr.

Wayne Xin ZHAO, Jing JIANG, Jianshu WENG, Jing HE, Ee-Peng LIM, Hongfei YAN et Xiaoming LI : Comparing twitter and traditional media using topic models. *In* Paul D. CLOUGH, Colum FOLEY, Cathal GURRIN, Gareth J. F. JONES, Wessel KRAAIJ, Hyowon LEE et Vanessa MURDOCK, éditeurs : *Advances in Information Retrieval - 33rd European Conference on IR Research, ECIR 2011, Dublin, Ireland, April 18-21, 2011. Proceedings*, volume 6611 de *Lecture Notes in Computer Science*, pages 338–349. Springer, 2011. URL https://doi.org/10.1007/978-3-642-20161-5_34.

Ming ZHONG, Pengfei LIU, Yiran CHEN, Danqing WANG, Xipeng QIU et Xuanjing HUANG : Extractive summarization as text matching. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 6197–6208. Association for Computational Linguistics, 2020. URL https://aclanthology.org/2020.acl-main.552/.

Arkaitz ZUBIAGA : A longitudinal assessment of the persistence of twitter datasets. *J. Assoc. Inf. Sci. Technol.*, 69(8):974–984, 2018. URL https://doi.org/10.1002/asi.24026.

Arkaitz ZUBIAGA, Damiano SPINA, Enrique AMIGÓ et Julio GONZALO : Towards real-time summarization of scheduled events from twitter streams. *In* Ethan V. MUNSON et Markus STROHMAIER, éditeurs : *23rd ACM Conference on Hypertext and Social Media, HT '12, Milwaukee, WI, USA, June 25-28, 2012*, pages 319–320. ACM, 2012. URL https://doi.org/10.1145/2309996.2310053.