UNIVERSITE BRETAGNE LOIRE

université angers

Thèse de Doctorat

Imane EL ALAOUI

Mémoire présenté en vue de l'obtention du
grade de Docteur de l'Université d'Angers
sous le sceau de l'Université Bretagne Loire

École doctorale : Ecole doctorale Sciences pour l'Ingénieur (SPI)

Discipline : Sciences de l'ingénieur
Spécialité : Informatique et applications
Unité de recherche :
Laboratoire des Systèmes de Télécommunications et Ingénierie de la Décision
Laboratoire Angevin de Recherche en Ingénierie des Systèmes

Soutenue le *04.07.2018*
Thèse N° : *112336*

Transformer les big social data en prévisions - méthodes et technologies

Application à l'analyse de sentiments

JURY

Rapporteurs :	**M. Youssef El Merabet**, Professeur, Université Ibn Tofail **M. Fabrice Bouquet**, Professeur, Université de Bourgogne Franche-Comté **Mme Raja Touahni,** Professeur, Université Ibn Tofail
Examinateur:	**M. Youssef Gahi**, Professeur, École nationale des sciences appliquées de Kénitra
Directeur de Thèse :	**M. Rochdi Messoussi**, Professeur, Université Ibn Tofail **M. Abdessamad Kobi**, Professeur, Université d'Angers
Co-encadrant de Thèse :	**M. Alexis Todoskoff,** Docteur, Université d'Angers

L'auteur du présent document vous autorise à le partager, reproduire, distribuer et communiquer selon les conditions suivantes :

- Vous devez le citer en l'attribuant de la manière indiquée par l'auteur (mais pas d'une manière qui suggérerait qu'il approuve votre utilisation de l'œuvre).
- Vous n'avez pas le droit d'utiliser ce document à des fins commerciales.
- Vous n'avez pas le droit de le modifier, de le transformer ou de l'adapter.

Consulter la licence creative commons complète en français :
http://creativecommons.org/licences/by-nc-nd/2.0/fr/

Ces conditions d'utilisation (attribution, pas d'utilisation commerciale, pas de modification) sont symbolisées par les icônes positionnées en pied de page.

REMERCIEMENTS

Tout d'abord, je remercie mes excellents superviseurs Pr. Rochdi Messoussi, Dr. Alexis Todoskoff et Pr. Abdessamad Kobi. Je suis très reconnaissante pour votre soutien et pour m'avoir offert cette expérience formidable et enrichissante. Ils ont su me laisser libre de mes orientations tout en étant présent chaque fois que j'en ai eu besoin malgré leurs nombreuses charges. Je les remercie également pour m'avoir accueilli dans l'unité de recherche et de m'avoir permis de travailler dans d'aussi bonnes conditions. Ils ont été et resteront des moteurs de mon travail de chercheur. Merci pour tout !

Egalement, je suis profondément reconnaissante à Pr. Youssef Gahi, un conseiller très consciencieux. Il est assez patient pour prendre le temps d'écouter mes idées et de partager ses pensées. Sa grandeur scientifique, sa gentillesse et son ouverture d'esprit, ont été pour moi de véritables atouts.

Je tiens à remercier Pr. El Merabet Youssef de l'Université Ibn Tofail et Pr. Fabrice Bouquet de l'Université de Bourgogne Franche-Comté pour l'intérêt qu'ils ont porté à ce travail en acceptant d'en être rapporteurs. Je tiens également à remercier Pr. Raja Touahni de l'Université Ibn Tofail d'avoir accepté de participer au jury de ma thèse.

Un grand merci à l'Université Ibn Tofail et l'Université d'Angers, qui m'ont fourni un environnement de travail très agréable dans lequel ils m'ont permis de travailler. Mes remerciements s'adressent également à tous les enseignants à qui je dois chaque bribe de ma connaissance et qui ont si bien mené leur noble quête d'enseignement.

La réalisation de cette thèse n'aurait pas été possible sans le soutien moral et affectif de ma famille bien aimée : Fouzia Lhaiti, Fouad El Alaoui, Amine El Alaoui, Fatima Raïssouli qui n'ont jamais hésité à m'offrir le meilleur qu'ils pouvaient. Je remercie spécialement maman et papa de m'avoir donné un environnement idéal : tendresse, amour, matériel et j'en passe. Vous avez su me comprendre et me soutenir pendant les moments les plus difficiles de doute. Sans vous, je ne serais pas ce que je suis devenue aujourd'hui. Je vous aime.

Aussi, tous mes remerciements à mon mari bien aimé : Youssef Sibari, grâce à qui cette « grande aventure » a été des plus agréables à vivre. J'ai même envie de dire qu'il a « co-encadré » cette thèse tellement il m'a entourée, soutenue, et m'a prodigué des encouragements constants.

Enfin, mais non des moindres, je remercie toutes les autres personnes ayant contribué de près ou de loin dans le bon déroulement de cette thèse et qui n'ont pas été mentionnées ci-dessus. Votre aide et votre soutien ont été appréciés.

Acronymes

SGBD	Système de Gestion des Bases de Données
XML	Extensible Markup Language
HTML	HyperText Markup Language
URL	Uniform Resource Locator
HTTP	Hypertext Transfer Protocol
CAP	Consistency Availability Partition Tolerance
ETL	Extract-Load-Transform
HDFS	Haddop Distributed File System
SGBD	Systèmes de Gestion de Base de Données
SGBDR	Systèmes de Gestion de Base de Données Relationnelle
ODBC	Open Database Connectivity
JDBC	Java Database Connectivity
BI	Business Intelligent
AM	Apprentissage Automatique
SQL	Structured Query Language
NoSQL	Not Only Structured Query Language
ACID	Atomicity, Consistency, Isolation et Durability
K-NN	k-Nearest Neighbors
LOF	Local Outlier Factor)
DJIA	Dow Jones Industrial Average
GTFS	General Transit Feed Specification
REN	Reconnaissance d'Entités Nommées
POS	Part of speech
TF	Term Frequency
NB	Naïve Bayes
SVM	Les machines à vecteurs de support ou séparateurs à vaste marge
AD	Arbres de decision
API	Application Programming Interface

Table des matières

INTRODUCTION 1

1. Contexte 1
2. Motivation 1
3. Objectifs 2
4. Contributions 2
5. Schéma de la thèse 3
6. Liste des publications 3

CHAPITRE 1 BIG DATA : ETAT DE L'ART 4

1. Introduction 4
2. Big data : outils et méthodes 4
2.1. Outils d'analyse big data 4
2.2. L'apprentissage automatique et les techniques de classification conçues pour le big data 9
3. Analyse de sentiments pour les big social data 12
3.1. Les approches basées sur l'apprentissage automatique 12
3.2. Les approches basées sur le lexique 15
3.3. Les approches hybrides 16
3.4. L'analyse de sentiment des big social data dans la politique 18
4. Conclusion 23

CHAPITRE 2 LA CHAINE DE VALEUR DU BIG DATA : METHODES, TECHNIQUES ET OUTILS 24

1. Introduction 24
2. Big data : chronologie et définitions 24
3. La chaine de valeur big data 28
3.1. Ingestion 29
3.1.1. Collecte 29
3.1.2. Prétraitement 31
3.1.3. Stockage 32
3.2. Analyse 33
3.3. Visualisation 36
4. Les outils d'analyse big data 37
4.1. L'écosystème Hadoop 37
4.1.1. La philosophie d'Hadoop 37
4.1.2. Le système de stockage d'Hadoop 39
4.1.3. Les techniques de traitement d'Hadoop 40
a) MapReduce 40
b) Yet Another Resource Negotiator as a platform 41
c) Yet Anbother Resource Negotiator as a platform V2 42
4.1.4. L'architecture d'Hadoop 42
4.1.5. Les distributions d'Hadoop 43
4.2. Accompagner la chaine de valeur big data 44
4.2.1. La phase d'ingestion 44
a) La collecte 44
Mode de chargement par lot 44
Mode stream et microbatch 45
b) Le prétraitement 45
c) Le stockage 46
4.2.2. La phase d'analyse 48
a) Batch processing 48
b) Stream Processing 49
c) L'apprentissage automatique 52
d) La recherche d'information 53
4.2.3. Visualisation 54
4.3. Applications 55
4.3.1. La santé 55

a) Pronostics et des diagnostiques pour les patients : 55
b) Modèle moderne et architecture système pour gérer de grandes quantités d'informations : 56
c) Réduction des coûts de traitement et amélioration de la santé de la population 57
4.3.2. Affaires et marketing 57
a) Finance 57
b) Publicité 57
c) Marketing 58
4.3.3. Ville intelligente 59
a) Conditions de vie des citoyens 59
b) L'administration publique et les services 60
c) La gestion des ressources 60
4.4. Outils couramment utilisés dans les applications 61
5. Conclusion 63

CHAPITRE 3 ANALYSE DES BIG SOCIAL DATA 64

1. Introduction 64
2. Texte mining 64
2.1. La phase d'analyse 65
2.2. La phase d'interprétation 66
3. L'analyse de sentiments 67
3.1. Les approches basées sur l'analyse lexicale 68
3.2. Les approches basées sur l'apprentissage automatique 70
3.2.1. Les desctipteurs 71
a) Représentation par stemmes 72
b) Représentation par lemme 72
c) Représentation par Sac de mots 72
d) Représentation par n-grammes 73
3.2.2. Codage des termes 73
a) Le codage booléen 74
b) Le codage Fréquence du terme (Term Frequency TF) 74
c) Le codage TF-IDF 75
3.2.3. Algorithmes de classification 75
a) Les réseaux de neurones artificiels 75
b) Les machines à vecteurs de support 77
c) K plus proches voisins – k-NN 78
d) Arbres de décision 79
e) Naïve bayes 80
3.2.4. L'évaluation du « classifieur » 81
a) Validation croisée 82
b) La précision et le rappel 82
c) F-score 83
d) La fonction d'efficacité du récepteur 83
4. Application 84
a) Amélioration des conditions de vie des citoyens 84
b) L'administration publique et services 85
c) La gestion et la surveillance des ressources 87
5. Conclusion 87

CHAPITRE 4 NOUVELLE METHODOLOGIE D'ANALYSE BIG SOCIAL DATA 89

1. Introduction 89
2. Problématique 89
3. L'approche proposée 91
3.1. Première étape : construction des dictionnaires annotés 92
3.2. Deuxième étape : Classification des sentiments 95
3.3. Troisième étape : la prédiction 98
4. Expérience et résultats 100
4.1. Implémentation 101
4.2. La collecte des données de l'expérience 102
4.3. Traitement des données 103
4.4. Performances 104
4.4.1. Évaluation de la précision de la classification 105
4.4.2. Comparaison avec d'autres méthodes existantes dans la littérature 109
4.5. Comparaison avec d'autres outils de l'analyse de sentiments 110
4.6. Évaluation de l'exactitude de la prédiction 112

5. Conclusion 112
CONCLUSIONS ET PERSPECTIVES 114
1. Conclusions 114
2. Perspectives 116
BIBLIOGRAPHIE 118
TABLE DES ILLUSTRATIONS 136
TABLE DES TABLEAUX 137

Introduction

1. Contexte

Avec l'avènement de la technologie numérique et des appareils intelligents, une grande quantité de données numériques est générée chaque jour. Cette forte augmentation des données, tant en taille qu'en forme, est principalement due aux réseaux sociaux qui permettent à des millions d'utilisateurs de partager des informations, d'exprimer et de diffuser leurs idées et leurs opinions sur un sujet, et montrer leurs attitudes envers un contenu. Toutes ces actions stockées sur les médias sociaux génèrent un ensemble massif d'opinions qui offre une opportunité pour les systèmes automatiques de fouille et d'analyse de données pour déterminer les tendances des internautes. Plusieurs chercheurs ont montré un vif intérêt pour l'exploitation de ces informations afin de prédire les comportements humains des domaines aussi variés que la médecine, la politique, le marketing, etc. Cette exploitation est principalement basée sur l'analyse d'opinion.
L'analyse d'opinion vise à déterminer le sentiment des gens en analysant leurs messages et différentes actions sur les médias sociaux. Elle consiste à classer la polarité des messages en différents sentiments opposés tels que positif et négatif. Elle pourrait être divisée en deux catégories principales :

• Analyse lexicale : vise à calculer la polarité d'un document à partir de l'orientation sémantique des mots ou des phrases qu'il contient. Cependant, les applications basées sur l'analyse du lexique utilisent des référentiels qui ne tiennent pas compte du contexte étudié.

• Apprentissage automatique (AM) : vise à construire des modèles à partir des données d'apprentissage étiquetées (instances de textes ou de phrases) afin de déterminer l'orientation d'un document. Néanmoins, les études utilisant ce type de méthodes ont été réalisées sur un sujet spécifique.

2. Motivation

L'analyse des réseaux sociaux a suscité un vif intérêt dans plusieurs domaines. Cependant, les résultats montrent des degrés de précision très variables. En outre, la plupart des méthodes

d'analyse des « big social data»[1] proposées dans la littérature souffrent d'un manque de flexibilité et d'adaptabilité au contexte d'analyse. Ce manque est principalement dû aux challenges des méthodes d'analyse d'opinion. En utilisant des référentiels dans l'analyse lexicale par exemple, nous ne prenons pas en compte le contexte de l'orientation sémantique des mots qui pourrait fortement changer selon le contexte. En utilisant l'apprentissage automatique, nous étiquetons des données (training data) qui sont limitées à un contexte spécifique. Par conséquent, nous ne pouvons pas les réutiliser dans l'analyse dans un contexte diffèrent.

Nous sommes motivés par le fait que les lacunes de l'analyse d'opinion font l'objet de vastes efforts de recherche afin d'améliorer la performance des systèmes automatiques de fouille de données et d'analyse des réseaux sociaux.

3. Objectifs

Les objectifs de cette thèse sont d'aborder les défis de l'analyse des « big social data » :

- Proposer une méthode d'analyse adaptable qui peut traiter efficacement les données issues de contextes différents dans les réseaux sociaux.
- Fournir une étude comparative actualisée des différents outils utilisés pour extraire l'information stratégique du Big data et les mapper aux différents besoins de traitement.

4. Contributions

Les principales contributions à la recherche de cette thèse sont les suivantes :

1. Dans une première phase de l'étude, nous nous sommes intéressés aux outils utilisés pour le traitement des données massives. Ces derniers montrent une grande diversité à la fois dans les points de vue et dans les cas d'utilisation. Pour cela, nous avons étudié les différentes nouvelles techniques et outils dans ce domaine et les avons mappés aux différents besoins de traitement. Cette étape nous a permis de choisir les outils adéquats pour mettre en œuvre notre système d'analyse de sentiment des « big social data ».
2. Dans une deuxième phase de l'étude, nous avons proposé un système d'analyse qui vise à contribuer à apporter des propositions aux contraintes de l'analyse de sentiments, en termes d'orientation sémantique contextuelle, en introduisant une nouvelle approche adaptable qui s'appuie sur les « big social data » pour analyser les comportements et les tendances. Cette

[1] Big social datareprésente les données issues des réseaux sociaux

approche permet d'attribuer une orientation sémantique aux mots en fonction du contexte étudié.

5. Schéma de la thèse

La présentation du manuscrit s'articule autour de 4 chapitres.

Le chapitre 1 est consacré à un état de l'art sur les méthodes, techniques et outils utilisés dans le traitement du big data et l'analyse des réseaux sociaux en particulier. Le chapitre 2 présente le contexte général de cette thèse en expliquant le big data, son histoire et sa chaine de valeur (génération, collecte, stockage et analyse). Il propose également un tour d'horizon des outils de traitement du big data en comparant leurs caractéristiques et leurs performances. En outre, nous soulignons l'importance de choisir l'outil approprié pour différents types d'applications.

Le chapitre 3 expose les différentes méthodes d'analyse des « big social data », données issues des réseaux sociaux, ainsi que leurs applications dans plusieurs domaines tels que la santé, l'éducation, le tourisme et la politique. Dans le chapitre 4, nous présentons notre approche pour le traitement des « big social data » ainsi que les expérimentations que nous avons menées afin de mesurer la robustesse et la performance de la méthode de classification proposée. Ce mémoire s'achève par une conclusion générale et quelques perspectives à ce travail.

6. Liste des publications

Ce travail a été présenté dans une conférence internationale et a fait l'objet de plusieurs publications dans des revues internationales :

1. I. El Alaoui, Y. Gahi, R. Messoussi, A. Todoskoff, A. Kobi, Big data analytics: A comparison of tools and application, in: The Mediterranean Symposium on Smart City Applications (2017), Springer.
2. I. El Alaoui, Y. Gahi, R. Messoussi, Y. Chaabi, A. Todoskoff, A. Kobi, A novel adaptable approach for sentiment analysis on big social data. The journal of big data 2018 5:12
3. I. El Alaoui, Y. Gahi, R. Messoussi, A. Todoskoff, A. Kobi, A survey of big data value chain - methods, techniques, tools and applications (under review).
4. I. El Alaoui, Y. Gahi, R. Messoussi, A. Todoskoff, A. Kobi, Big data and Smart cities: A state of the art of analytics tools and sentiment analysis based applications. (under review)

Chapitre 1 Big data : Etat de l'art

1. Introduction

Nous vivons dans un monde numérique, où nos actions sur Internet génèrent des traces numériques étroitement liées à nos vies personnelles. Le volume de ces traces générées quotidiennement augmente exponentiellement, créant des charges massives d'informations, appelées big data. Un tel volume important d'informations ne peut pas être stocké ni traité à l'aide des outils de Systèmes de Gestion de Base de Données (SGBD) classiques, et de nouveaux outils ont donc émergé pour nous aider à faire face aux défis des données volumineuses. Dans la première partie de ce chapitre, nous présentons une synthèse de la littérature existante relative au Big data.

Les utilisateurs d'internet utilisent de plus en plus les réseaux sociaux pour exprimer leurs avis sur des produits ou des services utilisés ou même leurs opinions politiques ou religieuses. Ces données représentent des informations importantes et massives, appelé « Big social data ». C'est donc dans ce contexte que nous nous intéressons particulièrement aux travaux relatifs à l'analyse des données issues des réseaux sociaux dans la dernière partie de ce chapitre.

2. Big data : outils et méthodes

2.1. Outils d'analyse big data

Le big data est un sujet important qui aide à identifier et extraire des connaissances précieuses et pertinentes. Ainsi de nombreux chercheurs ont travaillé sur le big data notamment sur sa chaîne de valeur et sur les outils de traitement du big data. En effet, Mohammed, Humbe, and Chowhan (2016) visent à fournir une vue d'ensemble du contexte big data. Par ailleurs, ils ont défini la chaîne de valeur des big data en 5 V :

- Volume : la quantité de données.
- Vélocité : le taux de génération et de transmission de données.
- Variété : les types de données structurées, semi-structurées et non structurées.
- Valeur : les résultats importants des données filtrées.
- Véracité : la confiance et l'intégrité des données.

Ces auteurs se sont concentrés sur une classification basée sur cinq catégories : les magasins de données, le format de contenu, les sources de données, le traitement des données et la mise en scène des données. Siddiqa et al. (2016), quant à eux, ont mis l'accent sur la gestion des données volumineuses en étudiant les techniques de gestion des big data en stockage, prétraitement, traitement et sécurité. Ils ont également présenté les orientations futures de la recherche pour ces derniers telles que l'intégration et la gouvernance des données. Quant à Philip Chen and Zhang (2014), ils ont donné un bref aperçu des problèmes du big data tel que : l'exploration des données, la gestion du volume, de la variété et de la vélocité dans différents domaines notamment le commerce, les administrations, et la recherche scientifique. En outre, ils se sont intéressés aux opportunités et aux défis de stockage, de transmission, d'analyse et de visualisation liés au big data au travers des techniques et technologies actuelles. Katal, Wazid, and Goudar (2013) ont modifié la définition des 5 V en proposant six propriétés avec quelques variantes : Variété, Volume, Vélocité, Variabilité, Valeur et Complexité (la nécessité d'établir des liaisons, des correspondances et le nettoyage de données entre des systèmes provenant de diverses sources). Enfin, ils ont mis en avant les défis du big data tels que la confidentialité, la sécurité, l'accès aux données, le partage des informations, le stockage, le traitement et les problèmes techniques tels que la tolérance aux pannes, l'évolutivité et la qualité des données. Ils ont également mis en évidence les bonnes pratiques du big data : l'intégration de plusieurs types de données (données structurées, semi-structurées et non-structurées), la généralité de la technologie pour gérer les différents types de données, et l'investissement dans la qualité des données.

Le big data et ses outils de traitement se développent rapidement, atteignant et affectant de plus en plus de domaines. Skourletopoulos et al. (2017) et Hashem et al. (2015) ont exploré les opportunités, les défis et les techniques du big data et leurs relations avec les technologies du Cloud, telles que le Big data-As-A-Service[2]. En outre, ils ont identifié quelques défis de l'analyse des données tels que l'évolutivité, la disponibilité, l'intégrité des données et la transformation des données. Li et al. (2014) ont donné un aperçu des techniques et systèmes utilisés pour le stockage du big data et de la gestion des ressources tels que le système de fichier distribué, les bases de données distribuées, l'interface d'accès et le langage de requête. Ils ont également présenté brièvement trois problèmes principaux de ce domaine :

- Le stockage à grande échelle,

[2] Le big data-As-A-Service (BDaaS) est la fourniture d'outils d'analyse statistique ou d'informations par un fournisseur externe qui aide les organisations à comprendre et à utiliser les connaissances acquises à partir de grands ensembles d'informations afin d'obtenir un avantage concurrentiel

- La complexité de gestion causée par l'hétérogénéité des données,
- L'exigence en termes de performance et de fiabilité du stockage.

Finalement, ils ont proposé des solutions aux problèmes cités plus haut, telles que :

- L'amélioration de l'index des requêtes dans les systèmes distribués : Pour résoudre le problème de stockage en termes de gestion et de fiabilité. En effet, l'index distribué consiste à diviser toutes les données d'index sur plusieurs nœuds. Lorsque la quantité de données augmente et que le système de stockage est surchargé, il suffit d'ajouter plus de nœuds dans le système. Egalement, pour un nœud unique, s'il est en panne, les données peuvent être perdues. Ce problème est résolu dans le système distribué où plusieurs copies sont sauvegardées sur un autre nœud, donc si un nœud est en panne, le système peut migrer la tâche du nœud vers un autre nœud.
- Le stockage et le traitement en temps réel et en streaming des big data : Les données sont généralement sous la forme de flux de données lorsqu'elles arrivent dans le système. La réduction du délai de chargement permet donc un chargement plus efficace et plus fiable dans le système de stockage efficacement et fiable est le fondement de
- Traitement d'événements complexes : Les sources et les types de données sont de toutes sortes, par conséquent, le filtrage et la logique de traitement sont très différents. Un moteur de processus complexe et flexible est donc nécessaire pour s'adapter à toutes sortes de données.

Afin de mettre en évidence les défis du stockage, Begoli (2012) a publié une revue sur l'état de l'art dans les architectures et les plates-formes pour l'analyse de données à grande échelle et la découverte des connaissances à partir des données. Mazumder (2016) a exposé les concepts, les techniques communes et les modèles dans le big data telles que le stockage et la gestion des services ainsi que l'écosystème Hadoop. Ils ont également proposé des implémentations, des cas d'utilisation (Ingestion de données, exploration de données, création d'information, consommation de données, archivage et purge de données) et de leur mise en correspondance avec divers outils et plates-formes big data.

Le processus d'extraction des données a autant d'importance que les données elles-mêmes. Ainsi, le besoin d'analyser et de qualifier les données collectées à partir de sources variées est l'un des principaux moteurs des outils d'analyse de big data. Pour cela, plusieurs contributions ont été réalisées dans le but de discuter et de proposer des améliorations des techniques d'analyse actuelles.

Comme l'un des principaux facteurs de succès du big data est la capacité à gérer les contraintes en temps réel, Liu, Iftikhar and Xie (2014) , Zheng et al. (2015) et Mohamed and Al-

Jaroodi (2014) ont abordé ce sujet en fournissant un aperçu de l'état actuel des outils qui permettent de traiter le big data. Liu, Iftikhar, and Xie (2014) ont présenté une analyse des technologies de traitement en temps réel open source tout en mettant l'accent sur les architectures en temps réel. Zheng et al. (2015) ont discuté des défis du big data et en particulier ceux du traitement en temps réel. Ils ont également présenté un modèle de stockage à plusieurs niveaux et quelques méthodes de déploiement pour répondre aux exigences du big data en temps réel et en hétérogénéité. Mohamed and Al-Jaroodi (2014) ont présenté quelques défis techniques face aux applications en temps réel dans le big data. En outre, ils ont fourni une analyse de performance et de certaines exigences du big data en temps réel.

D'autres chercheurs se sont focalisés sur la comparaison des outils de traitement du big data. En effet, Lopez, Lobato, and Duarte (2016) ont décrit, analysé et comparé trois principales plateformes de traitement de flux distribuées open source telles que Spark, Flink et Storm. Ils ont fourni des résultats expérimentaux de performance axés sur le débit et le parallélisme dans une application de détection de menaces dans le trafic réseau. Dans une étude menée par Lu et al. (2014), un référentiel de cadres de calcul de flux distribués modernes est proposé. Ce benchmark couvre les performances, la capacité de récupération des pannes ainsi que la durabilité des frameworks big data. Néanmoins, cette étude fournit un résultat uniquement pour Storm et Spark. Hess et al. (Hesse and Lorenz 2015) ont proposé une vue d'ensemble et une comparaison de quatre plates-formes de traitement de flux soit une plateforme de plus (Samza) que l'étude menée par Lopez, Lobato, and Duarte (2016). Cependant, la comparaison n'est basée que sur l'architecture et les responsabilités des composants des systèmes et ne permet pas de conclure à une supériorité en termes d'efficacité.

Dans une étude de Lu et al.(2014), un benchmark basé sur la capacité de récupération d'erreur ainsi que sur la durabilité pour les outils de calcul de flux distribués modernes est présentée. Le papier fournit un ensemble de résultats de comparaison entre Storm et Spark. Liu, Iftikhar, and Xie (2014) ont présenté des technologies de traitement Open Source en temps réel / en temps quasi réel tout en se concentrant sur leurs architectures et leurs plates-formes. Yadranjiaghdam, Pool, and Tabrizi (2016) ont présenté des outils d'analyse en temps réel du big data et ont classé différentes études en fonction des outils utilisés et du type d'application. Ils se sont concentrés sur les applications liées à la surveillance, à l'environnement, aux médias sociaux et aux soins de santé. Tsai et al. (2015) ont discuté des solutions d'analyse de données volumineuses. En outre, ils ont exposé certaines directions de recherche et des questions sur les techniques et les plate-formes ouvertes dans ce domaine de recherche. Gong, Morandini, and Sinnott (2017) ont proposé un benchmarking et l'implémentation de SMASH, une solution Cloud générique et

hautement évolutive, pour traiter les données de trafic à grande échelle. Katal, Wazid, and Goudar (2013) ont comparé trois frameworks (Storm, Hadoop, Drill) au travers des caractéristiques suivantes : propriétaire, charge de travail, code source, faible latence et complexité. En outre, les problèmes et les défis de la gestion des big data ont été discutés. Dans l'étude de Almeida and Bernardino (2015), les auteurs ont comparé six plateformes Open Source big data: Apache Mahout, MOA, Projet R, Vowpal Wabbit, PEGASUS et GraphLab, en fonction du paradigme de programmation et le langage de programmation, l'interface, type des données et les algorithmes supportés. Urmila (2016) a introduit et comparé Hive, Pig et MapReduce pour l'analyse du big data. La comparaison est basée sur le type du langage, l'interface d'utilisateur, les algorithmes disponibles et l'échelle des données supportées dans chaque outil.

Certaines recherches se sont concentrées sur NoSQL (Not only Structured Query Language). En effet, Corbellini et al. (2017) ont comparé les systèmes de stockage NoSQL en fonction de leurs types (les bases de données clé-valeur, colonne, document et ligne). La comparaison est basée sur l'API, le langage et la persistance. Néanmoins, elle ne nous permet pas de choisir la base de données adaptée au besoin applicatif. Cette étude couvre plus de bases de données NoSQL que les autres études telles que celle menée par Dede et al. (2013). Cependant Moniruzzaman and Hossain (2013) ont couvert moins de bases de données NoSQL que Corbellini et al. (2017) mais ont cité plus d'attributs de comparaison tels que l'intégrité (atomicité, la consistance, l'isolation, la durabilité…), l'indexation (index géo spatial, indexs secondaires) et la distribution (réplication, évolutivité horizontale…).

Il est important de souligner qu'Hadoop n'est pas la seule plate-forme attrayante pour le big data, il y a aussi Apache Spark. Alors que les deux outils sont parfois considérés comme des concurrents, il est souvent admis qu'ils fonctionnent encore mieux lorsqu'ils sont combinés. Apache Spark est devenu de plus en plus adapté en tant que projet de haut niveau pour l'analyse de big data, ainsi, de nombreuses recherches ont tendance à se concentrer sur son amélioration. Par exemple, Gulzar et al. (2016) ont créé un outil appelé BigDebug qui fournit des primitives de débogage interactives en temps réel pour le traitement des données massives dans Spark. NetSpark, un framework Spark amélioré est présenté par Li, Chen, and Xu (2016). Ce Framework réduit la durée d'exécution d'une tâche Spark en combinant la gestion de mémoire tampon réseau, la technologie RDMA (hardware-supported Remote Direct Memory Access : accès direct à la mémoire à distance assisté par le matériel) et l'optimisation de la sérialisation des données. Une stratégie, appelée MPTE (Multiple Phases Time Estimation), a été présentée

par Yang et al. (2016) afin de réduire l'impact des « straggler machines[3] ». En outre, la planification des tâches de sauvegarde a été améliorée en concevant un nouveau planificateur de tâches.

Spark utilise un calcul rapide en mémoire pour traiter les données. Néanmoins, le traitement en mémoire peut causer un problème de volatilité, d'échec ou de manque d'un RDD[4] (Resilient Distributed Datasets) qui amènera Spark à recalculer tous les RDD manquants sur la lignée. Une longue lignée augmentera également le coût en temps et l'utilisation de la mémoire du pilote analysant la lignée. Pour cela, Zhu, and Hu (2016) ont présenté un algorithme de point de contrôle automatique pour résoudre le problème du « long lignage » (longue lignée) de Spark avec peu d'impact sur la performance globale.

D'autres études ont conçu de nouveaux cadres basés sur Spark afin de rendre l'analyse du big data plus puissante. Par exemple, Yan et al. (2016) ont conçu TR-Spark pour faire face à des problèmes de ressources transitoires. Ce framework peut s'exécuter en tâche de fond secondaire sur les ressources transitoires et rend les applications basées sur Spark plus efficace. La conception de ce nouveau framework repose sur deux principes : la stabilité des ressources et la planification de la réduction de la taille des données. La combinaison de ces principes permet à TR-Spark de s'adapter à la stabilité de l'infrastructure. Afin de mieux prendre des décisions commerciales, Park et al. (2017) se sont appuyés sur Spark pour proposer un Framework d'analyse big data orienté sur les objectifs. Ce dernier a été expérimenté sur la décision d'expédition.

2.2. L'apprentissage automatique et les techniques de classification conçues pour le big data

L'apprentissage automatique (AM) est un domaine de l'informatique qui utilise des techniques statistiques pour donner aux systèmes informatiques la capacité d'« apprendre », sans être explicitement programmés. Toutefois, les machines ont besoin de données à analyser et sur lesquelles s'entraîner pour « apprendre » et construire des modèles. De fait, le big data est le carburant de l'apprentissage automatique, et l'apprentissage automatique est la technologie qui permet d'exploiter pleinement le potentiel du big data. Pour cela, plusieurs chercheurs se sont focalisés à introduire des méthodes mathématiques pour accompagner la croissance du big data en termes de besoins de l'analyse. En effet, Landset et al. (2015) ont exploré les différents outils

[3]. Les straggler machines sont responsables d'affecter les machines sur lesquelles les tâches MapReduce s'exécutent lentement.
[4] Spark utilise RDD pour l'abstraction de ses données, ciblant à plusieurs itérations le traitement de données à grande échelle avec réutilisation des données

open source qui se basent sur l'écosystème Hadoop pour l'analyse des données massives. Les chercheurs ont comparé l'extensibilité, la vitesse, la couverture et la facilité d'utilisation de tous ces outils. En outre, ils ont fourni une vue d'ensemble des algorithmes d'apprentissage automatique supportés par Hadoop. Dans le même sens, Richter et al. (2015) ont également abordé en profondeur les performances des outils big data au travers des techniques d'apprentissage automatique telles que la classification, la régression et le deep learning. Les outils comparés sont Mahout, MLlib, H2O et SAMOA, ainsi que les grands moteurs de traitement de données qu'ils utilisent, notamment Hadoop MapReduce, Apache Spark et Apache Storm. Les auteurs ont conclu qu'il n'y a pas encore un framework qui « fait tout », mais l'étude fournit un aperçu des forces et des faiblesses de chaque outil, ainsi que des conseils sur le choix des outils pour des besoins spécifiques.

Comme la performance est très importante dans un contexte tel que le big data, Zheng and Dagnino (2014) ont évalué les performances de quatre systèmes informatiques distribués à usage général. Les auteurs ont présenté les limites des approches actuelles, des modèles analytiques et des outils utilisés (Mahout, MLlib, R, Hama, ORAAH, Oryx Graph Lab et Misc) pour effectuer des analyses prédictives d'apprentissage automatique pour de très grands volumes de données où le traitement des données entraîne une défaillance du processeur. De plus, ils ont proposé une approche fondée sur le big data pour traiter les données historiques des applications de systèmes d'alimentation. Les résultats ont indiqué la faisabilité de prévoir les événements de panne de sous-stations et la charge d'électricité dans un système d'alimentation électrique, en utilisant un algorithme d'apprentissage automatique écrit dans le paradigme MapReduce ou des outils d'apprentissage automatique spécifiques au big data. Ces prévisions peuvent réduire le temps de réponse et améliorer la préparation pour réparer un arrêt, ce qui entraîne des économies significatives sur les coût d'exploitation et de maintenance. De leur côté, Qiu et al. (2016) ont passé en revue les algorithmes d'apprentissage automatique traditionnels et avancés tels que :

- L'apprentissage par représentation : une solution qui permet d'apprendre les représentations significatives et utiles des données qui facilitent l'extraction d'informations utiles lors de la construction des classificateurs ou de prédicteurs.
- L'apprentissage profond (deep learning) : différent à la plupart des techniques d'apprentissage traditionnelles, qui utilisent des architectures d'apprentissage à structure superficielle, l'apprentissage en profondeur utilise principalement des stratégies supervisées et / ou non supervisées dans des architectures profondes pour apprendre automatiquement des représentations hiérarchiques.

- L'apprentissage distribué et parallèle : contrairement à l'apprentissage classique, dans lequel la collecte des données dans une base de données pour le traitement central est éxigée, dans le cadre de l'apprentissage distribué, l'apprentissage est effectué de manière distribuée
- L'apprentissage par transfert : L'hypothèse majeure dans les algorithmes traditionnels d'apprentissage est que les données d'apprentissage ont les mêmes caractéristiques et distributions. Cependant, la grande hétérogénéité du big data détruit l'hypothèse. Pour résoudre ce problème, l'apprentissage par transfert a été proposé afin de permettre aux domaines, tâches et distributions d'être différents.
- L'apprentissage actif : Les données peuvent être abondantes mais les étiquettes sont rares ou coûteuses à obtenir. L'apprentissage actif tente de résoudre ce problème en sélectionnant un sous-ensemble des instances les plus critiques pour l'étiquetage.
- L'apprentissage basé sur le noyau : est une technique puissante qui permet d'augmenter la capacité de calcul en se basant sur la conception d'algorithmes d'apprentissage non linéaires efficaces.

Ensuite, ils ont présenté des solutions possibles telles que l'apprentissage en ligne, la découverte des connaissances dans les bases de données (KDD) et les technologies data mining, pour relever les défis de l'apprentissage automatique dans le domaine du big data (la vitesse, la variété, …).

Certaines recherches se sont intéressées à l'amélioration des méthodes de l'apprentissage automatique pour le big data. Ene, Im, and Moseley (2011) ont présenté une nouvelle méthode afin d'optimiser les algorithmes de clustering K-means en utilisant MapReduce alors que Qiu et al. (2016) se sont focalisés sur le clustering K-median et K-center. Ces études ont amélioré les méthodes d'apprentissage automatique traditionnelles de manière significative.

D'autres recherches se sont focalisées sur l'application de méthodes d'apprentissage automatique dans les données massives pour résoudre des problèmes spécifiques tels que la sécurité des réseaux informatique. En effet, Suthaharan (2014) a associé l'apprentissage automatique à d'autres technologies et notamment la mise en réseau moderne pour résoudre le problème de classification des données volumineuses dans le domaine de la sécurité des réseaux informatiques. Il a également souligné les problèmes et les défis que posent cette classification tels que le manque d'adaptabilité, la visualisation et l'incertitude des données.

Nous avons remarqué que les articles précédents ne couvrent pas tous les frameworks et toutes les plateformes big data. A ce jour, les études n'ont pas encore mis en évidence toutes les caractéristique et spécificités des outils big data ; ainsi, une étude complète de ce domaine est toujours attendue. Par ailleurs, ces technologies se développent tellement vite que leurs

documentations deviennent rapidement obsolètes. Toutes ces raisons nous ont poussé à combler ces lacunes. En effet, nous proposons dans ce travail de thèse une revue complète de la chaine de valeur du big data intégrant l'acquisition le stockage et l'analyse. En outre, nous présentons une étude comparative des outils big data pour chaque phase de la chaine de valeur en mettant en évidence leur caractéristiques, forces et limites.

Après avoir présenté de façon générique les études existantes dans la littérature au sujet du big data, regardons plus précisément les spécificités du big social data avec notamment l'analyse de sentiment.

3. Analyse de sentiments pour les big social data

Les réseaux sociaux, tels que Twitter, Facebook, etc, sont des plateformes où des millions de personnes interagissent en permanence sur des sujets très variés. Ces réflexions reflètent forcément leur opinions et sentiments par rapports aux sujets traités. Ces données subjectives en constante croissance constituent, sans aucun doute, une source d'information extrêmement riche et pertinente. Ainsi, un domaine nouveau a fait son apparition : l'extraction de sentiments à partir de l'analyse automatique des big social data. Cette approche, détaillée dans le chapitre 3, vise à classifier les sentiments en fonction de leur polarité : positive ou négative par rapport au sujet d'intérêt. Dans cette section, nous décrivons une synthèse de travaux relatifs à l'analyse de sentiments pour les données issues des réseaux sociaux.

Dans la littérature, trois approches ont été adoptées en se basant sur : l'apprentissage automatique (supervisé et non supervisé), l'analyse lexicale ou en combinant les deux (approches hybrides).

3.1. Les approches basées sur l'apprentissage automatique

Il existe principalement deux types de techniques d'apprentissage automatique : l'apprentissage supervisé et l'apprentissage non supervisé. L'apprentissage non supervisé consiste à apprendre sans superviseur[5] à partir de données non étiquetées et ne fournit donc pas des classes connues à priori. Il s'agit d'extraire des classes présentant des caractéristiques communes. Ainsi, le nombre et la définition des classes n'étant pas donnés à priori, rendant la tâche de classification ou d'étiquetage de sentiments difficile. Par conséquent, la grande majorité des recherches sur la classification des sentiments fait appel aux techniques d'apprentissage supervisées. Ces méthodes

[5] un superviseur est celui qui fournit la bonne « étiquette » à chaque observation.

utilisent des algorithmes d'apprentissage automatique où l'on cherche à produire des classes de sentiments à partir d'une base de données d'apprentissage contenant des « exemples » étiquetées par sentiments. Un exemple étiqueté est sous la forme (phrase,sentiment) ou (mot,sentiment) ou (descripteur,sentiment). Le sentiment peut être négatif, positif ou neutre : (aimer,positif) à titre d'exemple.

Le travail de Pang, Lee, and Vaithyanathan (2002) est l'une des premières approches qui s 'est intéressée à l'analyse de sentiments sur les réseaux sociaux en utilisant l'apprentissage automatique. Les auteurs ont utilisé un seul classificateur Naïve Bayes (NB) sur un corpus de critiques de films. Plusieurs modèles de Naïve Bayes ont été formés à l'aide de différentes fonctionnalités, telles que l'étiquetage morpho-syntaxique, les unigrammes et les bigrammes avec lesquels ils ont atteint une précision de classification de 77,3%. Ce taux est considéré comme une performance élevée du classificateur Naïve Bayes dans ce domaine.

Les chercheurs Ritterman, Osborne, and Klein (2009) ont utilisé les données de Twitter pour évaluer le sentiment du public et informer les marchés du modèle de prédiction. Leur approche utilise le classificateur Machine à vecteurs de support pour prévoir le sentiment public sur un sujet particulier, à travers l'analyse des messages échangés dans les microblogs. La méthode a été appliquée à des messages de twitter relatifs à une pandémie de grippe de 2009. Les résultats ont été comparés avec des données de marché de prédiction provenant d'une source indépendante. Les auteurs suggèrent que l'information présente dans les médias sociaux peut être utilisée comme indicateur pour les opinions publiques.

Toujours dans le contexte de l'analyse de sentiments, Go, Bhayani, and Huang (2009) ont comparé trois classificateurs : Naïve Bayes, l'entropie maximum, et Machine à vecteurs de support. Les résultats ont montré que ces algorithmes d'apprentissage automatique peuvent atteindre une précision supérieure à 80% lorsqu'ils sont formés avec des émoticônes. La contribution principale de cet article réside dans l'utilisation des tweets avec des émoticônes pour l'apprentissage supervisé. Les auteurs utilisent uniquement deux classes : positive et négative pour la classification des sentiments sans considérer la classe neutre.

Bifet, Holmes, and Pfahringer (2011) ont présenté un système, MOATweetReader, qui permet l'analyse de sentiments en temps réel. L'article montre une corrélation directe entre les sentiments de Twitter et la crise de Toyota et affirme que l'outil MOA-TweetReader aurait pu prédire la crise. Pour construire leurs classificateurs, ils se sont basés sur l'hypothèse suivante : lorsque l'auteur d'un tweet utilise un émoticône, il annote son propre texte avec un état émotionnel. Les Tweets annotés ont été ensuite utilisés pour former le classificateur. Cependant,

un ensemble restreint d'émoticônes (5 positives et 3 négatives) est utilisé, ce qui peut constituer une limitation pour couvrir tous les émoticônes disponibles sur twitter.

Duwairi et al. (2014) ont proposé un framework pour l'analyse de sentiments de tweets qui permet de les classer selon trois catégories : positive, négative et neutre. Ce framework, qui permet de traiter les dialectes arabes, Arabizi en plus des émoticônes, fait appel à trois classificateurs intégrés dans Rapidminer, à savoir Naïve Bayes, k-NN et Machine à vecteurs de support. Le système utilise ensuite la technique du vote pour affecter l'étiquette finale à chaque tweet. Le résultat expérimental a montré que le framework peut atteindre une précision de 76.78%.

Une autre méthodologie d'analyse de sentiments basée sur les médias sociaux arabes est proposée par Ahmed, Pasquier, and Qadah (2013). Les auteurs ont présenté une taxonomie de l'analyse des sentiments du texte arabe en général et dans les réseaux sociaux en particulier. En outre, ils ont proposé une phase de prétraitement de l'analyse des sentiments qui permet d'améliorer les résultats de l'extraction du sentiment. Dans toutes les expériences de test, une validation croisée 10 fois a été utilisée. Le classifieur Naïve Bayes est utilisé avec un vecteur de présence où les caractéristiques sont les mots N-grammes. Les résultats ont montré que l'uni-gramme donne la meilleure précision.

Nodarakis et al. (2016) proposent un algorithme qui utilise comme étiquette de sentiment, les hashtags et les émoticônes qui apparaissent dans un tweet. Ensuite, la procédure de classification de divers types de sentiments est faite de manière parallèle et distribuée avec l'algorithme k-NN. En outre, ils ont utilisé des filtres Bloom (Bloom 1970) pour compacter la taille de stockage des données intermédiaires et améliorer les performances de l'algorithme. La performance du système proposé peut aller de 37% à 78% (précision).

L'étude de Altawaier and Tiun (2016) vise à étudier des techniques d'apprentissage automatique en termes d'analyse des sentiments en langue arabe sur Twitter. Trois techniques ont été utilisées, à savoir Naïve Bayes, Arbre de Décision (AD) et Machine à vecteurs de support. Les résultats expérimentaux concluent que l'AD a surpassé les autres techniques en obtenant 78% de F-score (cf. chapitre trois).

Sodanil (2016) a proposé une approche d'analyse de sentiment pour l'évaluation des hôtels. Pour cela, il a appliqué trois méthodes de classification : Naïve Bayes, Arbre de Décision, et Machine à vecteurs de support, pour classer les critiques d'hôtels en polarités positives et négatives. Les expériences ont été menées en utilisant des critiques d'hôtels récoltées sur les portails de TripAdvisor et d'Agoda. Les résultats ont révélé que l'algorithme machine à vecteurs de support est l'algorithme le plus précis.

Mubarok, Adiwijaya, and Aldhi (2017) ont mené une recherche en trois phases, à savoir le prétraitement des données en se basant sur l'étiquetage morpho-syntaxique, la sélection des caractéristiques (les termes pertinents à utiliser dans la construction du modèle) en utilisant Chi Square et la classification de la polarité en utilisant Naïve Bayes. Les résultats d'évaluation montrent que le système peut atteindre jusqu'à 75% de F-score (avec une valeur moyenne de 53,96%).

Nirmal and Amalarethinam (2017) ont présenté une approche d'analyse de sentiment basée sur Naïve Bayes et implémentée avec Spark afin de fournir des résultats rapides et précis. L'étude comparative a montré que la technique de Naïve Bayes parallélisée proposée fournit des résultats à la fois plus rapides et plus précis par rapport aux algorithmes séquentiels Naïve Bayes et Machine à vecteurs de support.

3.2. Les approches basées sur le lexique

Les méthodes d'apprentissage automatique ont été largement utilisées dans l'analyse de sentiment dans les réseaux sociaux. Cependant, l'analyse de sentiments basée sur le lexique semble attirer plus d'attention.

Nagy and Stamberger (2012) ont proposé une technique basée sur SentiWordNet 3.0, et une liste d'émoticônes pour détecter les sentiments lors de catastrophes et de crises. Comparée aux réseaux bayésiens, la technique proposée a permis d'obtenir de meilleur résultats (27% de plus). Cependant, la combinaison des réseaux bayésiens avec SentiWordNet, et les émoticônes a donné la meilleure précision et le meilleur rappel.

Montejo-Ráez et al. (2012) ont proposé une nouvelle approche pour détecter la polarité dans les messages twitter ; basée sur la combinaison de la pondération des synsets du texte par un algorithme de promenade aléatoire (random walk en anglais) et les scores de polarité fournis par SentiWordNet. Les tests ont montré la possibilité de construire un système comparable à une approche supervisée basée sur l'algorithme Machine à vecteurs de support en termes de performance (62,85% F-score). Cependant, les chercheurs n'ont pas considéré des éléments qui pourraient biaiser l'analyse tels que la négation et l'élimination des mots vides.

Il est évident que même le marché boursier devient sensible à l'humeur des réseaux sociaux. En effet, les chercheurs Bollen et al. (Bollen, Mao, et Zeng 2011) ont obtenu des résultats assez satisfaisants (précision de 87,6%) en utilisant Twitter pour déduire l'évolution des marchés DJIA (Dow Jones Industrial Average) sur la période de février à décembre 2008. Dans ce travail, OpinionFinder (un système qui permet d'affecter à un texte un nombre : +1 si l'opinion est positive ou -1 si elle est négative) et Google-Profile of Mood States (un système qui permet

d'identifier l'humeur d'une phrase en six dimensions : (1) calme (2) alerte (3) sûr (4) vital (5) et (6) content) ont été utilisés pour classifier les données issues de Twitter. Le modèle de réseau neuronal flou a été appliqué sur les tweets et les variations quotidiennes du Dow Jones (indice boursier américain) pour vérifier la corrélation.

Chiavetta, Lo Bosco, and Pilato (2016) ont présenté un système qui classifie automatiquement l'orientation de sentiments exprimée dans les critiques de livres. Le système est basé sur une approche lexicale et utilise les techniques de traitement automatique du langage naturel (TALN) pour prendre en compte la relation linguistique entre les termes. La classification de la polarité globale du texte est basée sur la force de sentiment moyenne de ses phrases, tandis que la classification de chaque phrase est obtenue par un processus d'analyse inspectant, pour chaque terme, une fenêtre d'items précédents pour détecter des combinaisons particulières d'éléments donnant des inversions ou variations de polarité. Le système proposé est capable de classer automatiquement les révisions positives et négatives, avec une précision moyenne de 82%.

Dans l'étude de Lailiyah, Sumpeno, and Purnama (2017), Sentiwordnet a été appliqué pour classer le sentiment sur les plaintes publiques indonésiennes afin d'améliorer la satisfaction du public. Ensuite, ils l'ont comparé avec une méthode basée sur un lexique de sentiment indonésien. Cette dernière a obtenu une précision de 65,4% sur les données issues de Twitter et 81,4% sur les données du site officiel du gouvernement contre une précision de 47% sur Twitter et 56,85% sur les données gouvernementales pour Sentiwordnet.

3.3. Les approches hybrides

Afin de réunir les avantages des approches basées sur le lexique et l'apprentissage automatique, des chercheurs ont proposé des modèles hybrides combinant les deux approches.

Pour des analyses marketing, Zirn et al. (2011) ont proposé un framework pour l'analyse de sentiments en combinant plusieurs lexiques sentimentaux, le voisinage des mots, ainsi que des relations de discours. La logique de Markov a été utilisée pour intégrer les scores de polarité de différents lexiques de sentiment à partir des informations sur les relations entre les segments voisins. Pour valider leur méthode, ils ont utilisé les revues de produits et obtiennent une précision pouvant aller jusqu'à 69%.

De leur côté, Elmasry, Soliman, and Hedar (2014) ont proposé un nouveau système pour classifier la polarité des commentaires postés sur Facebook dans le e-commerce. Ce système associe les référentiels Sentiment Words et Idioms Lexicon (SSWIL) au classifieur Machine à vecteurs de support. L'ensemble des données testé a été collecté manuellement à partir de

microblogs. Le résultat expérimental a montré un taux de précision de 86,86% avec une précision de 88,63% et un rappel de 78%.

Gamallo and Garcia (2014) ont proposé une famille de classificateurs basée sur Naïve Bayes pour détecter la polarité des tweets. Les expériences ont montré que la meilleure performance est obtenue en utilisant un classificateur binaire conçu pour détecter seulement deux catégories : positive et négative. Afin de détecter les tweets avec et sans polarité, le système utilise une règle très basique qui recherche les mots de sentiment dans les tweets à analyser. Les auteurs ont construit un lexique de polarité avec des entrées positives et négatives provenant de différentes sources telles que : AFINN-11, Hedonometer et Sentiwordnet 3.0. Le modèle hybride obtient un F-score de 63%.

Khan, Qamar, and Bashir (2016) ont présenté une approche hybride nommée SWIMS qui intègre l'apprentissage automatique à une approche basée sur le lexique SentiWordNet. L'approche se base sur le lexique pour déterminer le poids des caractéristiques et les Machines à vecteurs de support pour apprendre leurs poids. Une approche de sélection de modèle intelligente est utilisée afin d'améliorer les performances de classification. Sept ensembles de données de référence disponibles en ligne ont été utilisés dans cette recherche, y compris un ensemble de données de critiques de films connus et un ensemble de données de critiques de films de Cornell. L'étude expérimentale montre une précision allant de à 68.17% à 85.78% en utilisant une validation croisée de 10.

Mertiya and Singh (2016) ont utilisé des critiques de films comme ensemble de données pour la formation et les tests. Ils ont levé l'ambiguïté de classification de l'algorithme Naïve Bayes en intégrant l'analyse des adjectifs. Premièrement, le modèle Naïve Bayes est appliqué sur les tweets collectés, ce qui donne des ensembles de tweets correctement et faussement étiquetés. Le faux ensemble étiqueté est ensuite traité avec une analyse d'adjectif pour déterminer sa polarité. Les résultats expérimentaux montrent que la précision globale du processus est améliorée par rapport à l'algorithme classique Naïve Bayes.

Dans le but d'identifier la compagnie de voyage la plus fiable dans le monde, Piyush et al. (2017) ont proposé une approche d'analyse de sentiment basée sur l'analyse lexicale et l'algorithme de Naïve Bayes. En effet, l'analyse lexicale repose sur des sources externes pour calculer la polarité d'une phrase alors que l'algorithme Naïve Bayes est utilisé pour la classification. Afin de valider l'approche proposée, les auteurs ont utilisé Twitter comme source de données. Les résultats ont montré que l'approche donne une précision de 83% et que Airbnb est le site de voyage préféré entre tous les leaders du marché.

Bhargava and Katarya (2017) ont proposé un framework dans le domaine de la bourse afin de prédire le coût réel des stocks. Ce système attribue des scores aux éléments du lexique en utilisant la notation relative basée sur la régression logistique. Il attribue une large gamme de scores aux éléments du lexique en se basant sur leur historique d'utilisation dans Web et les (les conséquences de leur utilisation selon le contexte) effets correspondants. Par cette méthode, la pertinence du lexique a été améliorée de manière significative dans la bourse. En changeant les informations de préparation, le framework proposé pourrait être utilisé pour toutes les bourses de différentes nations.

Dans la sous-section suivante, nous avons regroupé l'ensemble des recherches qui font appel à l'analyse de sentiments pour l'analyse des données issues des réseaux sociaux dans le domaine de la politique.

3.4. L'analyse de sentiment des big social data dans la politique

Dans ce domaine, plusieurs études se sont concentrées sur l'analyse des médias sociaux, en particulier lorsqu'il s'agit de grands événements qui attirent une grande attention comme les élections présidentielles. En effet, les médias sociaux sont devenus une plate-forme importante pour les candidats (qui partagent leurs programmes et établissent en contact direct avec les gens), et les électeurs (qui s'expriment et partagent leur opinion sur les candidats). Cette utilisation intensive des médias sociaux a attiré une grande attention dans la recherche universitaire et de nombreuses contributions ont été réalisées pour suivre et analyser ce genre d'événements. En effet, un rapport sur les médias sociaux dans le monde arabe reconnaît que « Yet no one was denying the pivotal role of the micro-blogging site either, or the role that social media will continue to play in Tunisia, Egypt, and the rest of the Arab world. »(Ghannam, Jeffrey 2011) (Pourtant, personne ne nia le rôle central des site de micro-blogging, ni le rôle que les médias sociaux continueront de jouer en Tunisie, en Egypte et dans le reste du monde arabe). De leur côté, DiGrazia et al. (2013) ont démontré la corrélation entre le pourcentage de noms de candidats mentionnés dans les tweets et la marge de vote aux élections du Congrès américain de 2010 et 2012. Dans le même contexte, Burnap et al. (2016) ont défini un modèle basé sur l'analyse de sentiment pour prédire les élections présidentielles. Malgré l'écart entre les résultats, leur méthode préserve correctement l'ordre des trois principaux partis en termes de vote pour l'élection générale britannique de 2015.

Les contributions dans ce domaine pourraient être classées en quatre catégories :

- **Approche fondée sur l'opinion** : Dans laquelle les auteurs ont basé leurs modèles sur les méthodes d'analyse de sentiments, afin de classifier les messages selon leur polarité. Il existe deux catégories de classes :
    - *Classes de sentiment* : les chercheurs O'Connor et al. (2010), Ramteke et al. (2016) et Tunggawan and Soelistio (2016), ont classé les sentiments sous deux classes (positive et négative), alors que d'autres auteurs, Jahanbakhsh and Moon (2014), Jose and Chooralil (2016), Razzaq, Qamar, and Bilal (2014), Smailović et al. (2015), Tumitan and Becker (2014) et Wicaksonoet al. (2016) ont défini une troisième classe : la classe neutre, ou même une quatrième (Wang et al. 2012) : la classe incertaine.
    - *Classes de contexte* : D'autres chercheurs ont utilisé des classes personnalisées en fonction du contexte. En effet, Conover et al. (2011) ont identifié trois catégories de postes : Left, Ambiguous et Right tandis que Mahmood et al. (2013) ont basé leur modèle de prédiction sur deux classes (Pros et Anti) pour chaque partie.
- **Approche basée sur le volume** : Ici, les chercheurs visent à prédire le candidat élu en se basant sur le nombre de tweets qui le mentionnent (% mention) ou sur le volume de retweet. En effet, DiGrazia et al. (2013), Livne et al. (2011), Shi et al. (2012), Soler, Cuartero, and Roblizo (2012) et Xie et al. (2016) ont tous mis en évidence une corrélation intéressante entre le candidat, le pourcentage ou le volume de retweet et le vote. De leur côté Finn, Mustafaraj, and Metaxas (2014) ont proposé une approche pour mesurer la polarisation politique sans utiliser de texte. Ils ont utilisé un réseau co-retweeté, ainsi que le comportement de retweeting des utilisateurs. Afin de construire le réseau co-retweeté, ils ont créé une matrice de relations de retweet, où chaque ligne représente les utilisateurs, et les colonnes représentent les utilisateurs qui ont été retweeté. A partir de cette matrice, ils ont obtenu la matrice co-retweetée, ne contenant que les utilisateurs qui ont été retweeté.
- **Approche basée sur l'opinion et le volume (OV) :** dans laquelle les approches basées sur l'analyse de sentiments et sur le volume sont combinées. Par exemple, Jahanbakhsh and Moon (2014) ont combiné les classes de sentiments (positive, négative, mixte, neutre), trois autres classes (unannotable, non-relevant, unclear) et les mesures basées sur le volume. Wong et al. (2016) ont utilisé des informations positives, négatives, neutres, des retweets et des tweeters. De leur côté, Conover et al. (2011) ont utilisé l'analyse de sentiments en se basant sur des classes de contexte ainsi que le nombre de

tweets mentionnant chaque partie et le volume de retweet. Khatua et al. (2015) ont utilisé le volume des tweets associé aux deux méthodes d'analyse de sentiments suivantes : la méthode du lexique d'opinion de Hu and Liu (2004), qui permet de classifier les mots en catégories positives et négatives, et la liste de AFINN-111 (Hansen et al. 2011) qui permet d'attribuer un score à un mot de fortement négatif (-5) à fortement positif (+5).

- **Approche basée sur les émoticônes** : Ici, la classification des messages est basée sur les émoticônes. Delenn et al. (2016) ont sélectionné des emoji pertinents et les ont catégorisés en classes (happy, sad, fear, laughter, et angry), puis ont dépouillé le sentiment à partir du premier emoji dans le post.

Tableau 1: Approches couramment utilisées dans l'état de l'art

Apprentissage automatique			Statistiques		
K-means	**NB / KNN**	**Autres**	**LR**	**OLS**	**MA**
(Jahanbakhsh and Moon 2014) (Ramteke et al. 2016) (Razzaq, Qamar, and Bilal 2014) (Tunggawan and Soelistio 2016) (Wang et al. 2012)2) Wicaksono, Suyoto, and Pranowo 2016)		(Jose and Chooralil 2016) (Razzaq, Qamar, and Bilal 2014) (Tumitan and Becker 2014)			(O'Connor et al. 2010) (Jahanbakhsh and Moon 2014)
(Mahmood et al. 2013)		(Mahmood et al. 2013)			
(Finn, Mustafaraj, and Metaxas 2014)	(Finn, Mustafaraj, and Metaxas 2014)	(Soler, Cuartero, and Roblizo 2012)	(Livne et al. 2011),	(DiGrazia et al. 2013)	(Xie et al. 2016)
				(Khatua et al. 2015)5)	
	(Delenn Chin, Jessica Zhao and Anna Zappone 2016)	(Delenn Chin, Jessica Zhao and Anna Zappone 2016)			

Approches			Lexique		SVM
			Manuel	Dictionnaire	
Basée sur l'opinion	Classes de Sentiment		(Razzaq, Qamar, and Bilal 2014) (Ramteke et al. 2016) (Tunggawan and Soelistio 2016) (Jahanbakhsh and Moon 2014) (Wang et al. 2012)2) (Smailović et al. 2015) (Tumitan and Becker 2014)	(O'Connor et al. 2010) (Jose and Chooralil 2016) (Wicaksono, Suyoto, and Pranowo 2016) (Tumitan and Becker 2014)	(Tumitan and Becker 2014) (Ramteke et al. 2016) (Razzaq, Qamar, and Bilal 2014) (Smailović et al. 2015)
	Classes de Contexte		(Mahmood et al. 2013)		(Mahmood et al. 2013)
Basées sur le volume					
Basées sur l'opinion et le volume			(Conover et al. 2011)	(Wong et al. 2016) (Khatua et al. 2015)5)	(Conover et al. 2011)
Basées sur les Emojis					(Delenn Chin, Jessica Zhao and Anna Zappone 2016)

Il existe des opinions divergentes sur la fiabilité de l'analyse des médias sociaux, certaines études ont montré la corrélation entre les résultats électoraux et leurs publications connexes, tandis que d'autres, comme Tumasjan (2010), Gayo-Avello (2012) et Delenn Chin, Jessica Zhao and Anna Zappone (2016) ont montré l'inverse. Bien que leurs recherches aient été basées sur des méthodes statistiques efficaces et / ou sur des dictionnaires connus, comme indiqué dans le

tableau 1, ils ont échoué à prédire les résultats des élections présidentielles. En général, ce problème est dû à des défis d'analyse de sentiments tels que la définition de l'orientation sémantique des mots qui pourrait fortement changer en fonction du contexte. Par exemple dans le contexte des dernières élections présidentielles américaines, utiliser le terme « sexuel » serait considéré comme un mot négatif dans un tweet lié à Trump, qui a été accusé d'agression sexuelle par quinze femmes, et par opposition, il sera considéré comme un mot positif pour Hillary. Un autre exemple, l'orientation sémantique du mot « email », qui est généralement un mot neutre, mais pourrait changer à négatif dans le contexte de Hillary, qui a contrevient les lois fédérales en utilisant un compte de messagerie électronique personnel pour les affaires du gouvernement.

4. Conclusion

Dans ce chapitre, nous avons introduit un certain nombre de techniques visant à analyser le sentiment public à partir des big social data. La classification des sentiments peut être obtenue par apprentissage automatique ou par des méthodologies lexicales ou une combinaison des deux. Cependant, nous avons vu que la plupart de ces techniques souffrent encore de quelques lacunes. En effet, les contributions précédentes ont utilisé des dictionnaires génériques qui ne tiennent pas compte du contexte étudié, ou ont fait appel aux méthodes d'apprentissage automatique qui utilisent des données d'apprentissage pour un domaine spécifique. Cependant, dans les deux cas, les méthodes énoncées sont peu efficaces. Dans les chapitres suivant, nous chercherons à combler les lacunes de l'opinion en termes d'orientation sémantique du contexte en proposant une nouvelle approche lexicale adaptable permettant d'attribuer automatiquement un score positif ou négatif aux mots en fonction du contexte. Contrairement aux approches basées sur le lexique, qui offrent une polarité fixe et statique des mots indépendamment de leur contexte, notre approche prend en compte l'occurrence des mots dans différents contextes dans les tweets pour assigner une polarité.

Chapitre 2 La chaine de valeur du big data : méthodes, techniques et outils

1. Introduction

Au cours des 20 dernières années, le monde numérique a connu une explosion en termes de données générées. En effet, nous générons quelques 2,5 quintillions d'octets par jour, sachant que plus de 90% des données existantes ont été créées récemment (Reinsel and Gantz 2011). Cependant, l'histoire du big data est souvent négligée car elle est généralement présentée comme déconnectée du passé. En fait, cela a débuté il y a soixante-dix ans, comme expliqué dans un article publié sur Forbes.com (Press 2013). Les premières tentatives de quantifier le taux de croissance du volume de données eurent lieu dans les années 1940. Le terme utilisé fut "explosion de l'information", apparu pour la première fois selon Oxford English Dictionary en 1941. Au début des années 1960, Price (1961) conclut que le nombre de nouvelles revues scientifiques a augmenté de façon exponentielle et a doublé tous les 15 ans. Il appela cette loi « la loi d'augmentation exponentielle ». Plus tard, en 1967, les scientifiques commencèrent à s'intéresser aux techniques de compression afin de s'attaquer aux problèmes de rétention de données et d'améliorer les capacités de stockage (Marron and de Maine 1967). Cependant, les limites de stockage furent atteintes dans les années 80, le volume de données continuant de croître plus rapidement encore. Dans les années quatre-vingt-dix, les scientifiques ont développé un vif intérêt à la rétention des données et aux défis de traitement, tels que le stockage et les capacités de traitement, et c'est la première fois où le terme big data apparût (Marron and de Maine 1967)(Cox and Ellsworth 1997)(Coffman and Odlyzko 2002)("MINTS - Minnesota Internet Traffic Studies" n.d.)(K. G. and Andrew 1999)(Bryson et al. 1999).

2. Big data : chronologie et définitions

En 1989, Tim Berners-Lee, un informaticien britannique, a inventé le World Wide Web. Son intention était de permettre le partage d'informations à travers un système hypertexte. Il n'avait aucune idée du genre d'impact que son invention aurait sur le monde. Tim avait écrit les trois technologies fondamentales qui restent la base du web d'aujourd'hui.

- *HTML (HyperText Markup Language)* : langage de balisage hypertexte.

- *URI (Uniform Resource Locator)* : Identificateur de ressource uniforme. Une sorte d '"adresse" unique utilisée pour identifier chaque ressource sur le web. Il est également communément appelé URL.
- *HTTP (HyperText Transfer Protocol)* : protocole de transfert hypertexte. Il permet la récupération de ressources liées à travers le Web.

À l'aube des années 1990, la première page Web fut diffusée sur Internet. Au fil des ans, le web n'eut de cesse de se développer et eut permis de nous connecter avec le monde entier : En 1994, les premiers blogs personnels furent créés. En 2000, Wikipédia, la plus collaborative des encyclopédies qui a révolutionné notre accès à la connaissance et à l'information, fut mise en ligne. Ce site a publié 20 000 articles dès la première année.

Un peu plus tard dans les année 2000, le web a évolué vers l'interactivité en créant le « Web 2.0 ». Ce dernier concerne plus particulièrement les interfaces et plateformes permettant aux internautes d'échanger les informations et d'interagir de façon simpliste, l'exemple les plus connu est le Web social. En effet, les réseaux sociaux notamment LinkedIn, Facebook et Twitter, furent leur apparition et ont révolutionné notre façon d'utiliser le web, que ce soit dans la sphère privée ou professionnelle. Ces organisations sont maintenant considérés comme les géants du web et ont su se construire les données (les bases utilisateurs et les informations) les plus colossales du monde. L'émergence du web 2.0 a fait naître plusieurs métiers notamment les hébergeurs web. Un service d'hébergement Web est un type de service d'hébergement Internet qui permet aux particuliers et aux organisations de rendre leur site Web accessible aux internautes. Comme le nombre d'utilisateurs sur Web augmentait, la pression pour les entreprises, grandes et petites, d'avoir une présence en ligne augmentait. Cette augmentation extrêmement rapide a accéléré le développement des services d'hébergement et a donné naissance à l'hébergement cloud. Le cloud est un nouveau type de plateforme d'hébergement qui permet un hébergement décentralisé, puissant, évolutif et fiable basé sur plusieurs serveurs.

À l'ère du cloud et des géants du web, toutes nos actions sur internet génèrent des données ; comme les achats en ligne, les contributions aux réseaux sociaux, la création de blogues, le partage de vidéos. Les organisations changent leur manière de traiter les données et créent ce que l'on appelle Open data (données ouvertes). Ce dernier offre la possibilité d'accéder et d'exploiter des informations précieuses, afin d'améliorer les services dans notre vie quotidienne.

Le concept de l'open data est récent mais similaire au mouvement open source. Ces données peuvent provenir de n'importe quelle source et tournent autour de la publication de données numériques en ligne. Selon la définition ouverte ("The Open Definition - Open Definition - Defining Open in Open Data, Open Content and Open Knowledge" 2013) « Open data and content can be freely used, modified, and shared by anyone for any purpose (les données ouvertes et le contenu peuvent être librement utilisées, modifiées et partagées par n'importe qui dans n'importe quel but) ». Ainsi, les données ouvertes pourraient être considérées comme l'essence du big data.

La figure 1 illustre la chronologie du big data :

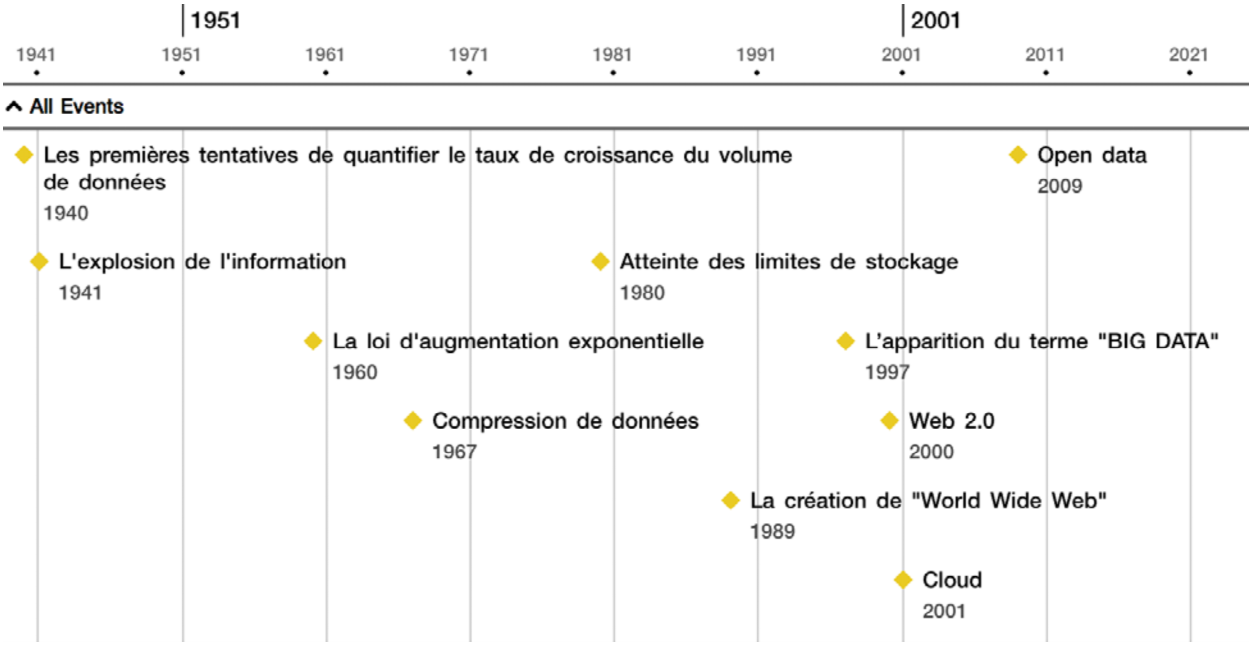


Figure 1: la chronologie du big data

A l'ère de l'open data, d'Internet, des géants du web, du cloud et des réseaux sociaux, le big data est devenu essentiel dans le monde d'aujourd'hui. Voici quelques « big chiffres » qui montrent l'incroyable masse de données qui nous entourent quotidiennement ; en une seconde, 6 801 vidéos sont visionnées sur YouTube, 2 561 661 emails sont envoyés, Google reçoit 58 588 requêtes et 2450 appels sont effectués depuis Skype ("Internet Live Stats" 2018). Suite à cette explosion de données, les données massives ne sont plus formées uniquement par des données structurées, mais d'autres types de données ont émergé :

- **Données structurées** : ce type a un format défini et une longueur fixe. Il est généralement représenté par des types primitifs tels que les entiers, les dates et les

chaînes. Les données structurées sont généralement stockées dans une base de données pré formatée.

- **Données semi-structurées** : ont une forme intermédiaire entre les données structurées et non structurées. Elles ne suivent pas un format prédéfini mais elles sont stockées avec des métadonnées associées. Un exemple de données semi-structurées est le langage de balisage extensible (XML).

- **Données non structurées** : Contrairement aux données structurées, les données non structurées sont stockées sans aucun format prédéfini. Elles sont généralement constituées de fichiers contextuels et multimédias tels que les e-mails.

Malgré ce déluge de données, tant en taille qu'en forme, le big data a continué à susciter plus d'attention et ses avantages sont devenus plus concrets dans plusieurs domaines tels que la physique, la biologie et les sciences sociales. Alors qu'un engouement croissant s'opère, personne ne semble pourtant s'accorder sur une définition bien arrêtée du big data. Néanmoins, afin d'éviter les confusions potentielles et de mieux caractériser le concept du big data, un ensemble d'attributs a été identifié comme caractéristique déterminante. Cet ensemble d'attributs est appelé « 7 V's », sept principales caractéristiques qui ont été initialement inspirés de la première définition 3V décrite par Gartner (Laney 2001) :

- **Volume** : Il représente l'une des caractéristiques les plus importantes du Big que l'on associe au big data. Cette association avec l'ampleur de l'ensemble de données se produit naturellement car tous les domaines tendent actuellement à collecter et stocker des quantités massives de données. Ce comportement est favorisé à la fois par les faibles coûts de stockage des données et des résultats plus précis, du point de vue de l'analyse des données.

- **Variété** : La collecte de données provenant de diverses sources conduit à une forte hétérogénéité. Le traitement du big data implique le plus souvent de manipuler des données sans structure relationnelle prédéfinie. Par conséquent, organiser les données avant de les stocker et de les traiter devient une tâche critique. La puissance du big data provient de la capacité à traiter et à extraire des informations de toutes sortes de données.

- **Vélocité** : C'est la vitesse à laquelle les données sont générées, collectées, stockées et traitées. La terminologie commune utilisée pour les données à déplacement rapide est la

"transmission de données en continu". Initialement, les défis liés à la vélocité étaient limités à des segments spécifiques de l'industrie, mais il devient un problème d'un cadre beaucoup plus large avec l'Internet des objets.

Plusieurs caractéristiques s'ajoutent aux trois originales pour mieux couvrir la nature complexe du big data :

- **Valeur** : il s'agit des informations fournies par les données. Ces informations sont l'objectif ultime et le facteur moteur du big data. Reposer sur une analyse rigoureuse du big data est une tâche complexe et implique des coûts importants. Par conséquent, les bénéfices obtenus, qu'ils soient financiers ou scientifiques, doivent compenser les ressources et les efforts investis.

- **Véracité** : La diversité des sources et des formes que prend le Big data, offre moins de contrôle sur son exactitude. La fiabilité des données a une grande incidence sur leur valeur. Par conséquent, l'un des nouveaux défis identifiés en matière de big data est la véracité, qui implique un processus d'élimination des mauvaises données avant le traitement (épuration) et de production d'un ensemble de données exact.

- **Variabilité** : est différente de la variété. La variabilité fait référence à des données dont la signification change continuellement. C'est particulièrement le cas lorsque la collecte de données repose sur le traitement du langage. En effet, les mots n'ont pas de définitions statiques, et leur signification peut varier énormément selon le contexte.

- **Visualisation** : Une fois les données traitées, le besoin de présenter les données de manière lisible et accessible devient primordial. L'utilisation de diagrammes et de graphiques pour visualiser de grandes quantités de données complexes est plus efficace pour transmettre des informations que des rapports remplis de nombres.

Le but ultime de tout projet big data devrait être de considérer ces « Vs ». En outre, tout système big data doit suivre une série d'étapes que l'on appelle « la chaîne de valeur du big data » pour extraire les informations les plus utiles et pertinentes à partir des données.

3. La chaine de valeur big data

À l'ère des géants du web, toutes nos actions génèrent des traces numériques. Nous trouvons une énorme quantité de données générées quotidiennement qui est énormément liée à vie personnelle

et que l'on peut exploiter dans différents domaines. Il existe également les données générées internes, appelées données hors ligne, qui sont créées par les opérations des organisations.

Le traitement de ces données massives joue un rôle clé dans la prise de décision. Le big data est devenu essentiel dans le monde d'aujourd'hui. Par conséquent, il est important de mettre en évidence les différentes phases de la chaîne de valeur du big data. Nous proposons la chaine de valeur dans la figure 3, qui peut être divisée en quatre phases principales : ingestion de données, analyse de données et visualisation de données.

3.1. Ingestion

La phase d'ingestion de données regroupe elle-même trois sous-phases : la collecte, le prétraitement, et le stockage dans une infrastructure adéquate.

3.1.1. Collecte

i) ***Batch-Loading*** (chargement des données par lots) : est un mécanisme efficace pour traiter d'énormes données sur une période de temps. Ce mode est recommandé lorsque les données sont déjà stockées dans un autre système traditionnel, lorsque l'application ne nécessite pas de traitement en temps réel ou si le temps de traitement n'a aucune influence sur le résultat. Dans le cas inverse, le stream loading est largement recommandé.

ii) ***Stream-Loading*** (chargement des flux de données) : est la deuxième méthode utilisée pour alimenter le système de stockage. Il permet de collecter et d'agréger des données provenant de différentes sources en temps réel.

iii) ***Micro-Batch Loading*** : Cette technique est l'intersection du stream et du batch loading comme indiqué dans la figure 2. Elle divise les flux en micro-lots. Par conséquent, les données sont obtenues en un temps quasi réel.

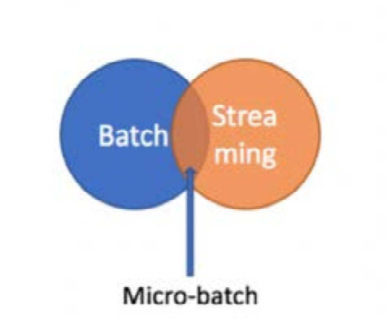


Figure 2: Batch vs Micro-batch vs Streaming

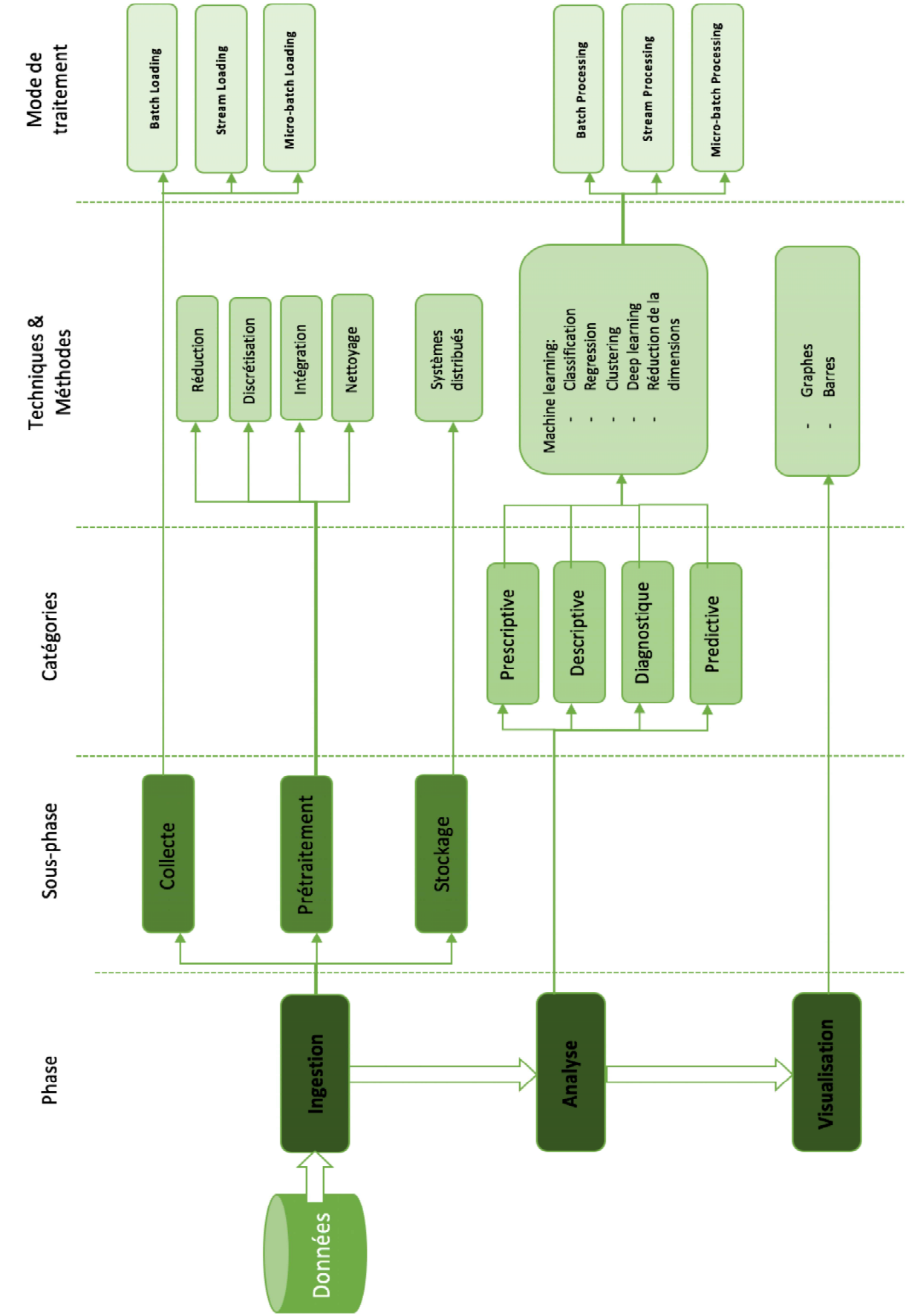


Figure 3: la chaine de valeur big data

3.1.2. Prétraitement

En règle générale, les données brutes collectées à partir de différentes sources sont énormément redondantes et consomment une capacité de stockage en conservant des données insignifiantes. En outre, certaines méthodes d'analyse requièrent un certain niveau de qualité des données. En effet, l'étude menée par William D. ("Newspapers.com - Historical Newspapers from 1700s-2000s" n.d.) "Work With New Electronic 'Brains' Opens Field For Army Math Experts" confirme que la construction de bons modèles se fonde sur la qualité des données à analyser. Par conséquent, la sous-phase de prétraitement semble être une étape de la plus haute importance.

Egalement, elle est la sous-phase la plus laborieuse. Ceci est dû à l'absence de structuration et la multiplication des sources d'information. Effectivement, les données réelles sont souvent incomplètes : valeurs manquantes, informations importantes manquantes. Certains attributs n'ont pas de valeur et cela peut être causé par un mauvais fonctionnement d'un équipement, ou considérés comme peu importants et donc ignorés lors de la saisie. A titre d'exemple, les profils des utilisateurs des sites internet sont souvent incomplets. Et/ou bruitées : présence d'erreur dont la cause peut être un problème de saisie, de transmission, d'enregistrement dupliqué, d'appareil de mesure défectueux. Et/ou incohérente : divergence entre les attributs.

Il existe plusieurs méthodes pour améliorer la qualité des données ; effectuer le nettoyage, la qualification et l'amélioration de la fiabilité des données (Han, Kamber, and Pei 2012). Différentes études ont été menées afin de résumer les techniques de prétraitement telles que (García et al. 2016) qui a mis en évidence les méthodes les plus utilisées pour le prétraitement des données. Ils ont classé les contributions en fonction de la catégorie de prétraitement, du nombre d'entités, du nombre d'instances, de la taille maximale supportée par chacune d'entre elles. Les principales techniques du prétraitement sont :

i) ***Nettoyage des données*** *:* le nettoyage peut inclure les processus suivants : 1) La suppression des tuples dont les valeurs sont manquantes mais cette méthode s'avère peu efficace quand le pourcentage des valeurs manquantes est important. 2) La complétion manuelle (infaisable, surtout pour le big data).3) L'utilisation d'une constante globale, de la moyenne de l'attribut, de la valeur la plus probable (arbre de décision), de la moyenne de l'attribut pour la même classe.

ii) ***Intégration des données*** : combinaison de différentes sources de données en une seule. Elle permet notamment de résoudre les conflits de valeur et de gérer le bruit.

iii) ***Transformation*** : elle inclut plusieurs processus ; lissage en réduisant le bruit dans les données, agrégation en simplifiant et construisant les cubes de données,

généralisation en remplaçant les données finies par des données de plus haut niveau, normalisation de certains attributs numériques afin qu'ils varient entre 0 et 1.

iv) ***Réduction*** (échantillonnage, equi-fre, equi-width,) : Nous supposons que les données suivent un modèle et nous estimons les paramètres du modèle en ne stockant que ces derniers. Généralement, une représentation réduite des données permet d'obtenir à peu près les mêmes résultats.

v) ***Discrétisation*** : En général, elle consiste à la fois au découpage de données en classes homogènes et en la justification mathématique de cette classification. Dans le prétraitement, nous divisons l'intervalle de valeurs possibles en sous intervalles.

Ces étapes permettent de réduire significativement la quantité du bruit, l'incohérence, dans les données. Les données obtenues au final peuvent être considérées comme fiables si l'application des étapes est réussie.

3.1.3. Stockage

Cette phase consiste à stocker une énorme quantité de données collectées à partir de plusieurs sources. Cette phase doit fournir un espace de stockage fiable et permettre un accès puissant pour l'analyse des données. Généralement, les systèmes qui fournissent un stockage de données volumineux s'appuient sur une architecture distribuée

En informatique, les systèmes distribués ("Full Text of 'DISTRIBUTED SYSTEMS Concepts And Design 5th Edition 3'" n.d.) consistent à partager les données et les tâches (calcul et traitement) entre plusieurs machines interconnectées d'une manière contrôlée et efficace. L'une de leurs caractéristiques importantes est que la façon dont les machines communiquent est pour la plupart cachée aux utilisateurs. Ainsi, l'utilisateur perçoit le système comme une unité unique.

Un système distribué sera normalement disponible en permanence, même si certaines parties peuvent être temporairement hors service. Les utilisateurs et les applications ne doivent pas remarquer le replacement, la réparation et l'ajout des parties pour servir plus d'utilisateurs ou d'applications.

Ce concept a été créé dans les années 1970 lorsque le réseau français Cyclades a voulu mettre en commun toutes les ressources informatiques des grandes entreprises et des centres universitaires. Depuis, le système distribué n'a cessé d'évoluer et a donné naissance à plusieurs plateformes telles que Hadoop.

3.2. Analyse

L'analyse big data consiste à manipuler des quantités énormes et des variétés de données réparties sur un cluster afin d'en extraire des informations. En utilisant les bonnes méthodes d'analyse, les données massives peuvent fournir des informations précieuses, car elles proviennent d'un large éventail de sources pour découvrir des modèles cachés.

Les méthodes d'analyse du big data sont divisées en quatre catégories majeures :

i) ***Prédictives*** : ces méthodes visent à extrapoler des nouvelles informations à partir des informations présentes. Elles fournissent des estimations sur la probabilité d'un résultat futur. L'aperçu extrait est généralement une prédiction des tendances futures et des motifs de comportement. Les méthodes prédictives se caractérisent par l'explication des données avec l'obligation d'existence d'au moins une variable à expliquer (Variable mesurant le résultat d'un phénomène ou d'une expérience.).

L'analyse prédictive peut être utilisée dans le cas d'une activité commerciale ; elle peut être utilisée, depuis la prévision du comportement des clients d'achat jusqu'à l'identification des tendances dans les activités de vente. Elle permet également de prévoir la demande d'intrants provenant de la chaîne d'approvisionnement, des opérations et des stocks.

ii) ***Descriptives*** : comme leur nom l'indique, ces méthodes décrivent ou résument les données brutes en identifiant les relations entre ces dernières. Ce type d'analyse permet d'identifier des comportements dans le passé et de comprendre comment ils pourraient influencer les résultats dans le futur. Généralement, un tableau de bord en temps réel est utilisé pour extraire les analyses.

Les méthodes descriptives se caractérisent par l'absence d'une variable à expliquer. Les exemples les plus courants de l'analyse descriptive sont des rapports qui fournissent des informations historiques sur la production, les opérations, les ventes les stocks et les clients de l'entreprise.

iii) ***Prescriptives*** : ce type d'analyse « prescrit » quelles décisions doivent être prises en considération. Elle permet de prédire ce qui va se passer, pourquoi il se produira en fournissant des recommandations concernant les actions possibles. C'est le type d'analyse le plus important et aboutit généralement à des règles et des recommandations pour les étapes suivantes.

L'analyse prescriptive est relativement complexe à administrer et par conséquent, elle est peu utilisée. Lorsqu'elle est mise en œuvre correctement, elle peut avoir un impact

important sur la façon dont les entreprises prennent des décisions. Par exemple, elle peut être utilisée pour optimiser la production, la planification dans la chaîne d'approvisionnement afin de garantir la livraison des bons produits au bon moment.

iv) ***Diagnostique*** : ce type d'analyse vise à analyser les performances passées pour déterminer ce qui s'est passé et pourquoi. Bien que l'analyse diagnostique comporte souvent une description et une caractérisation substantielles des éléments de données, elle se concentre davantage sur la détermination des facteurs et des événements qui ont contribué au résultat. Le résultat de l'analyse est souvent un tableau de bord analytique.

*P*ar exemple, pour une campagne de marketing sur les médias sociaux, nous pouvons utiliser ce type d'analyse pour évaluer le nombre de messages, de partages, de mentions et d'interactions connexes pour savoir ce qui n'a pas fonctionné dans la compagne.

Alors que différentes formes d'analyse, prédictive, descriptive, prescriptive, diagnostique, peuvent fournir différentes quantités de valeur, elles ont toutes leur place. Il existe différentes méthodes pour traiter et extraire ces informations telles que l'apprentissage automatique (AM). L'apprentissage automatique selon J. S. Brownstein et al. (1998) aide à créer des modèles pour construire des applications intelligentes telles que la reconnaissance de la parole et de l'image, la recherche web à haute performance ou l'imagerie médicale (Seixas, Barbon, and Mantovani 2015). Il est divisé en deux phases : la phase d'apprentissage dans laquelle un modèle est construit (basé sur une partie des données) et la phase de test dans laquelle le modèle est testé (sur l'autre partie des données).

L'apprentissage automatique peut être supervisé et non supervisé :

i) ***Apprentissage supervisé*** : dans lequel l'enseignant (le système responsable de l'apprentissage) connaît la bonne réponse et donne les entrées ainsi que les résultats souhaités. L'algorithme fait itérativement des prédictions sur les données d'apprentissage. L'apprentissage s'arrête lorsque l'algorithme atteint un certain niveau de performance.

ii) ***Apprentissage non supervisé*** : Il ne s'appuie que sur les données d'entrée et aucune variable de sortie correspondante. Néanmoins, l'algorithme découvre et présente par lui-même la structure intéressante à adopter pour les données.

Les tâches de l'apprentissage automatique regroupent plusieurs types d'algorithmes pouvant être classés en plusieurs familles :

i) ***La classification automatique*** : est une tâche d'apprentissage supervisée qui consiste à attribuer une classe à une nouvelle observation à classer. La prédiction se base sur un ensemble de données d'apprentissage contenant des observations dont l'appartenance à une classe est connue. Elle comprend de nombreux algorithmes tels que Naïve Bayes, Random Forest, Support Vector Machines, etc.

ii) ***La régression*** : est également une tâche d'apprentissage supervisé qui fait correspondre un modèle à un groupe de données. La régression est utilisée pour analyser la relation d'une variable avec une ou plusieurs autres. Les algorithmes de régression couramment utilisés sont la régression linéaire, la régression linéaire en continu, la forêt aléatoire, etc.

iii) ***Le clustering*** : est une tâche d'apprentissage non supervisée qui regroupe les éléments similaires dans le même groupe (appelé cluster ou grappe). Autrement dit, les données regroupées dans le même cluster partagent des caractéristiques communes qui correspondent à des critères de proximité. Il peut être réalisé par différents algorithmes, tels que le clustering k-Means, qui diffèrent de manière significative dans leur notion de constitution des clusters et comment les determiner efficacement. Les notions populaires de clusters incluent des groupes avec de petites distances entre les membres du même cluster, la densité de l'espace de données, des intervalles ou des distributions statistiques particulières. Le clustering peut donc être formulé comme un problème d'optimisation multi-objectif. Il existe plusieurs algorithmes de ce type, le clustering en fait partie.

iv) ***La réduction de la dimension*** : est une technique qui minimise les dimensions d'un ensemble de données en combinant, en transformant et en supprimant leurs caractéristiques à condition de garantir des informations similaires. La réduction de la dimension peut également être vue comme le processus de dérivation d'un ensemble relativement petit qui peut être utilisé pour reproduire la plus grande partie de la variabilité d'un ensemble de données massive.

v) ***Le deep Learning*** (Apprentissage profond) : les méthodes du « deep learning » fonctionnent avec un apprentissage à plusieurs niveaux de détail ou de représentation des données à travers différentes couches ; les algorithmes peuvent être supervisés ou non supervisés. Elles visent à modéliser les relations mathématiques entre les points de

données. En utilisant le « deep learning », une observation peut être représentée de différentes façons par un vecteur de données. Certaines représentations et une bonne aptitude d'analyse automatique des différenciations rendent le processus d'apprentissage plus efficace. Le deep learning utilise des réseaux neuronaux artificiels et essaie de modéliser la façon dont le cerveau humain traite la lumière et le son en vision et en audition respectivement.

En outre, les catégories, techniques et méthodes d'analyse de données volumineuses sont principalement utilisées de trois manières : en mode batch (hors ligne), en mode Stream (en temps réel) et en mode Micro-batch (en temps quasi réel).

3.3. Visualisation

La visualisation de données permet d'illustrer les relations au sein des données et de transmettre des informations de manière universelle avec une représentation visuelle artistique telle que des graphiques, des barres et des tableaux de bord.

Lorsque les résultats d'une analyse sophistiquée sont présentés de manière lisible et facile à comprendre, la prise de décision devient plus efficace et plus facile. En effet, la combinaison de puissantes fonctionnalités d'analyse et de visualisation des données permet d'identifier les domaines nécessitant une attention ou une amélioration, les meilleures opportunités, de repérer les tendances, de comprendre les risques et de déterminer les facteurs influençant une tendance donnée.

D'énormes quantités de données opérationnelles et financières sont stockées dans le monde des affaires. Ainsi, l'analyse et la visualisation de big data fournissent plus de transparence et de précision dans les organisations. En effet, ils permettent de prendre des décisions commerciales plus précises telles que l'amélioration des stratégies de prix, les chaînes d'approvisionnement et les campagnes publicitaires (Deng, Gao, and Vuppalapati 2015).

Un autre exemple est celui des données de localisation, qui ont longtemps été collectées et utilisées à des fins de cartographie dans presque toutes les industries. La visualisation de ce type de données ne consiste pas seulement à mettre des points sur une carte, mais aussi à les combiner avec d'autres données afin de produire des informations plus approfondies. Par exemple, une publication ou un message dans les médias sociaux pourrait contenir un emplacement GPS et des personnes marquées. Autrement, dans le secteur de la santé, il est possible de localiser une personne atteinte d'une maladie et les personnes susceptibles d'être contaminées. En effet, une étude menée dans (Dredze 2012) a démontré une corrélation entre les tweets et les données publiées par CDC (Centers for Disease Control and Prevention) sur une maladie grippale. Cette

étude permet de suivre et estimer la propagation d'une maladie plus rapidement que le CDC et l'OMS (Organisation mondiale de la santé). Ainsi, l'intelligence épidémique est largement utilisée et joue un rôle essentiel dans le secteur de la santé. Elle pourrait également être utile pour les campagnes de vaccination, la communication au public et la mise en œuvre de stratégies pour faire face à la propagation d'une maladie.

Suivre un tel rythme, en acquérant, stockant, analysant les données, représente un défi critique pour les outils traditionnels de traitement des données. Pour cela, plusieurs outils et solutions ont été proposés pour accompagner la chaîne de valeur du big data. Cependant, il est généralement difficile de trouver quels sont les outils appropriés pour un contexte spécifique. Dans les sections suivantes, nous nous concentrerons sur les différents outils et plateformes proposés et fournirons des comparaisons de performance pour les trois premières phases du big data.

4. Les outils d'analyse big data

Le big data regroupe plusieurs disciplines notamment le traitement parallélisé, le data mining, et l'apprentissage automatique. Dans cette partie nous allons présenter et comparer les performances des outils existants pour chaque phase de la chaine de valeur du big data. Tout d'abord, nous présenterons l'écosystème Hadoop qui est considéré comme une plateforme de référence pour le traitement du big data.

4.1. L'écosystème Hadoop

Hadoop est un framework libre et open source géré par Apache Software Foundation. Il est le plus largement utilisé et reconnu pour analyser, stocker et manipuler de grandes quantités de données tout en s'appuyant sur une architecture distribuée.

4.1.1. La philosophie d'Hadoop

En 2004, Google a inventé un modèle de programmation appelé MapReduce (Dean and Ghemawat 2004) (Jeffrey, Dean and Sanjay, Ghemawat 2004) dans lequel des calculs parallèles (souvent distribués) sont effectués et exécutés sur un grand cluster, qui représente un ensemble de machines interconnectées. Google a également inventé son système de fichiers appelé GoogleFS (Ghemawat, Gobioff, and Leung 2003), pour permettre le traitement d'énormes quantités de données. Doug Cutting s'est appuyé à la fois sur MapReduce et sur GoogleFS et a introduit Hadoop (T. White 2012).

Hadoop est un framework open source écrit en Java et administré par Apache Software Foundation. Il permet le stockage distribué et le traitement par lots pour les grands ensembles de

données sur un cluster allant d'une à plusieurs centaines de machines appelées nœuds. De plus, il garantit un stockage et un traitement des données tolérant aux pannes en fournissant une évolutivité linéaire sur le matériel standard. En outre, plusieurs sociétés comme Amazon, AOL, eBay, Google et Facebook, l'ont adopté.

La première version de Hadoop (0.14.x) a été publiée en 2007. Plusieurs versions ont ensuite été développées pour apporter des améliorations, des optimisations et des corrections de bogues, y compris les corrections dans la gestion des erreurs, les messages de journal et la sécurité. Cependant, nous notons que les changements majeurs ont été apportés par la version 2.0, qui a intégré non seulement un système de gestion des ressources appelé YARN (voir Figure 4 ("Hortonworks" 2011)) mais également une haute disponibilité pour HDFS (détaillé ci-dessous). Cette version a donné naissance à d'autres versions permettant de corriger des bogues, d'apporter des améliorations significatives telles que le passage à un niveau de stockage SSD et la prise en charge du cryptage avancé des fichiers (AES) pour un cryptage plus rapide des câbles. ("Apache Hadoop Releases" 2012) donne plus de détails sur les améliorations apportées par toutes les versions. La version 3.0, qui est encore en phase alpha, améliorera les versions Hadoop 2.x sur plusieurs champs. Les changements attendus sont :

- **Erasure Encoding** : permet de réduire le temps de stockage tout en offrant le même niveau de tolérance aux pannes que précédemment.

- **YARN Timeline Service v.2** : cette version de YARN améliorera l'évolutivité et la fiabilité du service Timeline. En outre, il améliorera la facilité d'utilisation en introduisant les flux et les agrégations.

- **Opportunistic containers** : un nouveau type d'exécution, que l'on peut programmer pour être exécuté dans un NodeManager même sans ressources disponibles au moment de la planification.

- **Optimisation native MapReduce au niveau des** tâches : améliore les performances de 30% ou plus en cas de tâches fastidieuses.

- **Prise en charge de plus de deux NameNodes** : afin de répondre aux besoins des déploiements critiques en termes de degré de tolérance aux pannes, Hadoop 3.0 permet d'exécuter plusieurs NameNodes de secours. Par exemple, en configurant trois NameNodes (un actif et deux passifs), le cluster peut tolérer l'échec de deux nœuds.

Toutes les versions de Hadoop (Figure 4 (“Welcome to ApacheTM Hadoop®!” 2011)) offrent une combinaison du système HDFS pour la partie stockage et de l'algorithme MapReduce pour la partie traitement. Cependant, les versions 2.x permettent d'utiliser d'autres algorithmes que MapReduce.

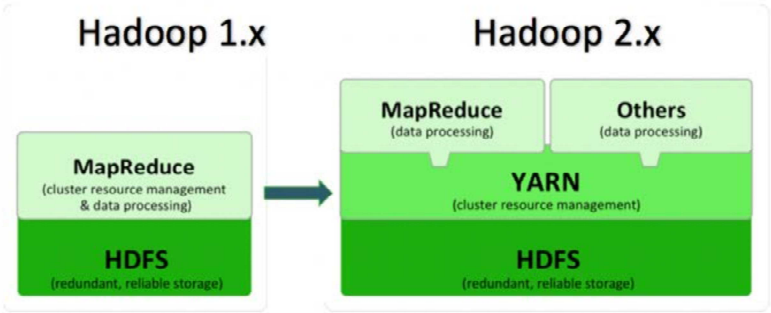


Figure 4: Hadoop 1.x vs Hadoop 2.x

4.1.2. Le système de stockage d’Hadoop

HDFS est un système de fichiers distribué qui permet de stocker d'énormes quantités de données sur un grand nombre de machines. Il a été conçu pour être déployé sur du matériel de base. Il est utilisé pour gérer un système de fichiers distribué comme s'il s'agissait d'un seul disque dur. HDFS (“Guide D’architecture HDFS” 2011) peut stocker n'importe quel type de données sous forme de fichiers HDFS qui sont divisés en 64 Mo (ou plus selon la configuration). En outre, il fournit un système de réplication de bloc configuré à 3 par défaut (modifiable). Chaque bloc est répliqué sur plusieurs nœuds lors de l'écriture. Ainsi, si un bloc est indisponible sur un nœud pour la lecture, les autres seront disponibles.

Hadoop suit une architecture maître/esclave. Un cluster Hadoop se compose d'un seul **NameNode**, un serveur maître qui gère l'espace de noms du système de fichiers et régule l'accès client aux fichiers et de plusieurs **DataNodes** où sont stockées les données, chaque bloc réside dans différents DataNodes si possible. Une discussion plus détaillée sur des composantes est fournie dans la sous-section architecture d’Hadoop.

Dans un contexte opérationnel, la requête du client n'atteint pas immédiatement le DataNode. En effet, les données sont d'abord cachées par le client dans un fichier local temporaire jusqu'à ce qu’elles atteignent une taille plus grande que le bloc HDFS. Ensuite, le client contacte le NameNode pour stocker le bloc et vide le bloc de données du fichier temporaire local dans le DataNode spécifié.

4.1.3. Les techniques de traitement d'Hadoop

a) MapReduce

La technique de traitement Hadoop est appelée MapReduce. Bien que HDFS distribue des données sur plusieurs nœuds, MapReduce convient au traitement des ensembles de données à grande échelle ("Welcome to ApacheTM Hadoop" 2011)(D. Schneider 2012). Son principe consiste à décomposer une tâche en plusieurs tâches identiques pouvant être exécutées sur le DataNode. Comme son nom l'indique, MapReduce est composé de deux principales phases : la phase Map, où chaque tâche est parallélisée, et la phase Reduce, où tous les résultats intermédiaires sont combinés en un résultat final.

MapReduce est divisé en 5 phases comme le montre la figure 5 : La première consiste à identifier les nœuds qui contiennent les données à traiter. La deuxième est la phase Map, à travers laquelle nous appliquons le traitement pour chaque ensemble. La tâche de Map est réalisée par un nœud, qui distribue les données à d'autres nœuds. Chaque nœud récepteur est responsable du traitement des données reçues. Dans cette phase, les sorties sont une collection de paires <valeur, clé>, et forment les entrées pour la phase Reduce, comme illustré dans la figure 6. Le contenu des paires dépend du traitement. La troisième phase, appelée Shuffle, consiste à trier les données puis à regrouper les données liées pour être traiter dans le même nœud. Dans la quatrième étape, appelée Reduce, les données sont agrégées. La clé est utilisée lors de la fusion pour regrouper les valeurs.

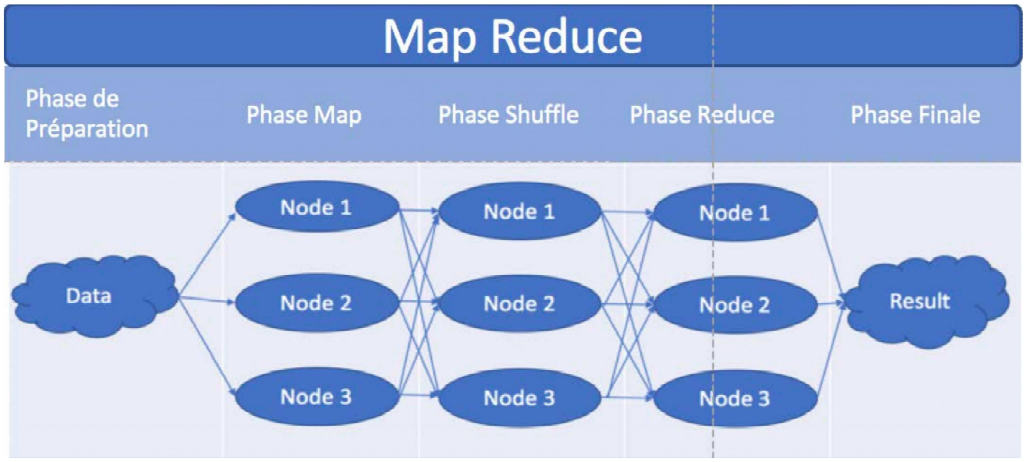


Figure 5: Le concept de MapReduce

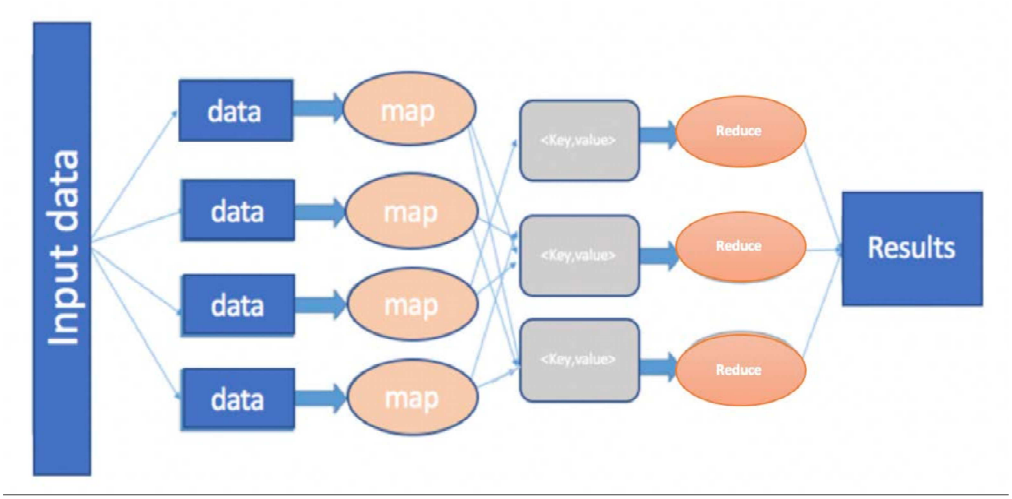


Figure 6: La phase MapReduce

b) Yet Another Resource Negotiator as a platform

Hadoop a introduit YARN (Yet Another Resource Negotiator) dans les versions 2.x.

Yarn permet à n'importe quelle application de s'exécuter sur de gros volumes de données à travers les clusters. Il déploie plusieurs applications sur les mêmes plates-formes et résout les limites de MapReduce, y compris en temps réel.

YARN ("Apache Hadoop 2.7.1 – Apache Hadoop NextGen MapReduce (YARN)" n.d.) divise les fonctionnalités JobTracker en deux « daemons » distincts : **Resource Manager** et **Application Master**.

- **ResourceManager** : arbitre les ressources dans toutes les applications (jobs) du système et comporte deux composants principaux : Scheduler et Application Manager.
    - **Le planificateur (Scheduler)** : alloue les ressources aux applications en fonction de leurs besoins. Il est basé sur la notion abstraite du "Resource Container" qui intègre des éléments tels que la mémoire, le CPU, etc.
    - **Le gestionnaire d'application (Application Manager)** : permet d'accepter des "soumissions de travaux" (job-submissions), de négocier l'exécution des applications et de suivre leur statut. Il permet également de redémarrer un conteneur en cas d'échec.

- **L'Application Master :** est une bibliothèque qui négocie les ressources avec le ResourceManager et se base sur le(s) NodeManager(s) pour exécuter et surveiller les tâches.

    - **Le NodeManager** : surveille l'utilisation des ressources du conteneur (CPU, mémoire, disque ...) et le communique au RessourceManager / Scheduler.

c) Yet Anbother Resource Negotiator as a platform V2

Dans cette version YARN (Hadoop 3.0), un ensemble de collecteurs distribués (writers) est utilisé pour écrire des données dans le stockage back-end. Les collecteurs sont co-localisés avec les maîtres d'application auxquels ils sont dédiés. À l'exception du collecteur de chronologie du Resource Manager (resource manager timeline collector), toutes les données appartenant à l'application sont envoyées aux « level timeline collectors ».

- Les **NodeManagers** exécutent les conteneurs pour l'application. En outre, ils sont dédiés à l'écriture de données dans le collecteur de timeline sur le nœud, sur lequel le maître d'application s'exécute.

- Le **ResourceManager** permet de gérer son propre collecteur de chronologie.

- Les **Timeline Readers** permettent de servir les requêtes via l'API REST.

Pour mieux comprendre la différence entre YARN et MapReduce, il serait judicieux que nous présentions l'architecture de Hadoop.

4.1.4. L'architecture d'Hadoop

L'architecture Hadoop repose sur la configuration maître / esclave. Il existe également un maître secondaire qui permet d'effectuer des sauvegardes du nœud maître en cas d'erreur.

Il existe deux composants dans la partie MapReduce : JobTracker et plusieurs TaskTrackers ("MapReduce Tutorial" n.d.), tandis qu'un cluster HDFS comprend un NameNode et plusieurs DataNodes.

- **NameNode** : gère l'espace de noms et l'arborescence du système de fichiers. Il permet aussi de reconstruire les fichiers à partir des blocs.
- **DataNodes** : stockent et restaurent les données qu'ils contiennent. Egalement, ils communiquent la liste des blocs au NameNode.

Contrairement aux architectures traditionnelles, Hadoop déplace le traitement vers la couche de stockage (c-à-d vers les données). Cela permet d'éviter la consommation de bande passante. Il existe plusieurs façons de configurer une plateforme Hadoop. Pour cela, nous présentons les différentes alternatives basées sur hadoop, dites distributions.

4.1.5. Les distributions d'Hadoop

Une distribution Hadoop s'appuie sur Hadoop et propose des outils et des packages qui facilitent l'administration et la surveillance des clusters Hadoop. Dans la littérature, on distingue trois éditeurs principaux pour les plateformes Hadoop (Fléchaux 2015):

- **Cloudera** : a été la première société à développer et distribuer des applications basées sur Apache Hadoop. Elle s'appuie principalement sur Hadoop à l'exception des outils d'administration.
- **MapR** : contribue à de nombreux projets Apache Hadoop tels que HBase, Pig, Hive. Néanmoins, cette distribution s'appuie principalement sur sa propre vision de MapReduce et HDFS qui est MapR, MapR-FS et MapR-DB. Elle est également enrichie de nombreuses solutions propriétaires
- **Hortonworks** : est la seule plateforme Hadoop entièrement open source. Il fournit de nombreux outils de stockage et d'analyse qui aident à gérer les big data. Hortonworks a introduit Yarn, qui est considéré comme une révolution pour les frameworks de traitement.

Il est parfois difficile de choisir la distribution adéquate à un besoin spécifique. Pour cela, nous présentons dans la table suivante les différentes caractéristiques, les avantages et les inconvénients de chaque distribution.

Il est important de souligner qu'un facteur important qui fait la différence entre les distributions Hadoop est le nombre d'outils impliqués. Plus il y a d'outils pour accompagner la chaîne de valeur du big data, plus la plateforme devient attrayante.

Tableau 2: Hortonworks vs Cloudera vs MapR

	Hortonworks	**Cloudera**	**MapR**
Frameworks	Environ 24	Environ 13	Environ 12
File distribution	HDFS	HDFS	MapR File system

Outils de Management	Oui	Oui	Oui
Haute disponibilité	Single failure recovery	Single failure recovery	Self-healing across multiple failures
Performance globale du cluster lecture/écriture	La même que cloudera	La même que Hortonworks	Plus rapide que cloudera et hortonworks
Avantages	Supporte la platforme Windows	une interface conviviale avec de nombreuses fonctionnalités	la distribution Hadoop la plus rapide
Inconvénients	L'interface de gestion sur HDP n'a pas beaucoup de fonctionnalités	Comparativement plus lent que MapR	n'a pas une bonne console d'interface comme Cloudera

4.2. Accompagner la chaine de valeur big data

4.2.1. La phase d'ingestion

Comme expliqué précédemment, la phase d'ingestion consiste à collecter des données provenant de différentes sources avec tous les modes possibles, en appliquant un processus de filtrage et en les stockant. Dans ce qui suit, nous fournissons un benchmark des outils les plus connus utilisés dans cette phase.

a) La collecte

Mode de chargement par lot

Dans le contexte du big data, le chargement par lots peut être utilisé de trois manières différentes :

La première est de continuer à collecter les données localement puis de les importer par vacation dans le système de stockage.

La seconde consiste à utiliser la technologie d'extraction ETL (Extract-Transform-Load). Elle crée un réseau d'échange pour synchroniser des informations massives. Cette option est plus efficace pour extraire que pour importer des données.

La troisième est **Sqoop** ("Sqoop -" 2012), qui permet l'extraction ou l'importation de données depuis un SGBDR (Système de Gestion de Base de Données Relationnelle) traditionnel vers Hadoop HDFS en utilisant la technologie Map/Reduce. Il est également possible d'exporter le résultat d'un traitement dans le cluster vers une base de données traditionnelle.

Mode stream et microbatch

Il existe des outils et des Application Programming Interface (API) qui collectent des données et sont considérés comme des outils de chargement de flux. Cependant, ils le font en micro-lot (micro-batch). Les solutions sont donc divisées en deux catégories : les outils en temps réel (Flume et Chukwa) et les outils micro-batch (Spark, Kafka). Voici une brève description de ces solutions :

- **Flume** ("Welcome to Apache Flume — Apache Flume" 2012) : peut collecter, agréger et déplacer de grandes quantités efficacement des données du journal. Il est robuste et tolère les pannes.

- **Chukwa** ("Chukwa - Welcome to Apache Chukwa" 2013): permet de collecter des grumes à grande échelle. Il s'appuie sur HDFS et MapReduce et hérite de l'évolutivité de Hadoop.

- **Spark** ("Spark Streaming | Apache Spark" 2014)("Apache SparkTM - Lightning-Fast Cluster Computing" 2014): est un Framework conçu spécialement pour le traitement de données, mais il possède plusieurs API qui permettent également le streaming, à savoir SparkStreaming. Ce dernier permet le streaming micro-batch en transmettant et enregistrant des données dans n'importe quel système de stockage supporté par Hadoop tout en s'appuyant sur RDD (Resilient Distributed Datasets) (Zaharia et al. 2012).

- **Kafka** ("Apache Kafka" 2012): est une plateforme de streaming distribuée. Il offre 3 API pour le streaming :

  Streams API : permet de faire le streaming et de stocker des données dans des catégories appelées Topics.

  Producer API : permet à une application de publier les streaming data pour un ou plusieurs sujets.

  Connector API : est responsable de la connexion des sujets et de l'application.

b) Le prétraitement

Les mêmes frameworks sont utilisés pour le prétraitement et l'analyse (cf. sous-section 4.2.2)

c) Le stockage

La persistance et la gestion des données d'une manière évolutive qui répond aux besoins des applications nécessitent un accès rapide aux données. Les SGBDR ont été la solution principale, et presque unique, au paradigme du stockage depuis près de 40 ans. Cependant, les propriétés ACID (Atomicity, Consistency, Isolation et Durability) qui garantissent les transactions de base de données manquent de souplesse en termes de changement de schéma, de tolérance aux pannes et aux performances lorsque les volumes de données et la complexité augmentent. Le stockage distribué et les systèmes tels que HDFS et NoSQL ont été conçues en tenant compte de l'objectif d'évolutivité et présentent une large gamme de solutions basées sur des modèles de données alternatifs pour manipuler des bases de données géantes persistantes et hétérogènes dans un environnement distribué.

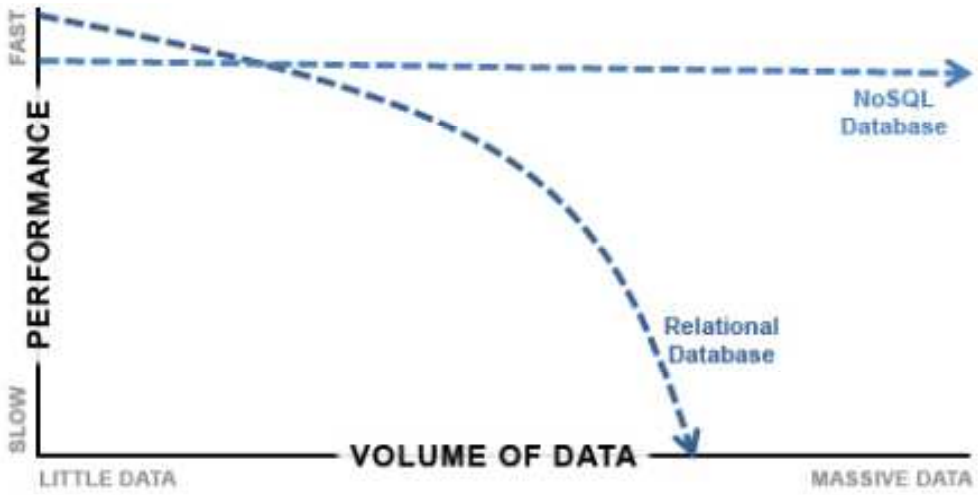


Figure 7: SQL vs NoSQL

La figure 7 (“What Is Hadoop and NoSQL?” 2012) illustre la variation de performance NoSQL par rapport au SQL. Évidemment, plus le nombre de données est important, moins SQL devient performant, ce qui n'est pas le cas avec NoSQL.

On peut distinguer différents types de BDD NoSQL qu’on peut regrouper en 4 compartiments essentiels :

- **Clé/ valeur** : simple à mettre en œuvre, le stockage de l’information est fait à l’aide des tableaux associatifs. Les données sont donc représentées par un couple Clé/Valeur où la valeur peut être des chiffres ou un string.
- **Orienté colonne :** ce modèle a le même principe qu’un SGBDR traditionnel sauf que le nombre de colonnes est dynamique et peut varier en fonction de l’entrée afin d’éviter les valeurs NULL. Ce principe évite de renvoyer des valeurs NULL.

- **Orienté document** : il consiste à stocker, gérer et récupérer des informations orientées document. Ce modèle se base sur le même principe d'une BDD orientée clé/valeur et dans ce cas la valeur est un document de type JSON ou XML.
- **Orienté graphe** : est un concept NoSQL basé sur la théorie des graphes et qui consiste à utiliser des structures de graphe pour des requêtes sémantiques avec des nœuds, des arêtes et des propriétés pour représenter et stocker des données. Cette relation graphique permet de relier les données entre elles et facilite la récupération de l'information.

Les bases de données NoSQL ont également des corollaires avec le théorème CDP (Cohérence, Disponibilité, Tolérance au partitionnement). Ce théorème stipule que tout système distribué ne peut garantir que deux contraintes mais pas les trois à un instant T.

- **Cohérence** : tous les nœuds du système voient exactement les mêmes données en même temps.
- **Disponibilité** : Les données sont accessibles même en cas de panne.
- **Tolérance de partition** : Le système continue à fonctionner malgré un nombre arbitraire de messages supprimés par le réseau entre les nœuds.

Une comparaison de certaines bases de données NoSQL est présentée dans le tableau 3.

HDFS et NoSQL peuvent être combinés pour gérer de gros volumes de données en augmentation rapide. Néanmoins, HDFS est conçu pour gérer plus de volume de données que NoSQL et n'a pas de limite contrairement à ce dernier (contrainte du CDP théorème).

Une fois les données stockées, l'étape suivante sera d'analyser et d'extraire des informations précieuses. Dans la section suivante, nous présentons et comparons les différentes méthodes utilisées dans la phase d'analyse.

Tableau 3 : CouchBase vs MongoDB vs Hbase vs Cassandra

		CouchBase	MongoDB	Hbase	Cassandra
Modèle de stockage		Doc	Doc	colonne	Column
Opération/sec (sur 8 nœuds)	**Processus de chargement**	86,924.94	34,305.30	74,405.64	86,924.94
	Lire-modifier-écrire	4,576.17	2,028.06	4,582.67	65,822.21
Latence	**Lecture**	98,514.50	1,126,244.08	350,383.54	26,973.70

moyenne par charge de travail (us) (sur 8 nœuds)					
	Lire-modifier-écrire	476,732.96	1,032,139.72	467,970.50	31,200.83
Réplication	**Lecture**	+	+	-	+
	Écrire dans la réplique	+	-	-	-
	Cohérence	-	+/-	+	+/-

Dans la sous-section suivante, nous présentons quelques outils d'analyse utilisés dans les deux modes hors-ligne (batch processing) et en temps réel (stream processing). Nous présentons également les outils de recherche d'information.

4.2.2. La phase d'analyse

a) Batch processing

MapReduce : Comme détaillé ci-dessus, c'est la technique de traitement utilisée par Hadoop pour effectuer des calculs distribués. Malgré sa popularité, elle éprouve des difficultés pour plusieurs opérations de traitement comme la jointure qui n'est pas facile à implémenter en MapReduce. En outre, elle exige l'écriture des programmes personnalisés qui sont difficiles à maintenir et à réutiliser. Une autre limitation est lorsque le système de stockage est une base de données, MapReduce devient moins performant.

DSL : (Digital Subscriber Line) permet de transmettre des données numériques sur liens spécialisés. Deux langages DSL sont conçus pour fonctionner sur Hadoop :

- ***Pig*** ("Welcome to Apache Pig!" 2008) : une plate-forme d'analyse de grands ensembles de données. il fournit un langage de traitement simple, qui est compilé en tant que Jobs MapReduce.
- ***Hive*** ("Apache Hive TM" 2011) : est le "data warehouse" de Hadoop, il fournit un langage qui extrait la structure "relationnel-db like" à partir des données non-relationnelles ou non-structurées (fichiers plats, JSON, logs web, Hbase, Cassandra ...). Il pourrait interagir avec les outils BI via les connecteurs ODBC / JDBC (("Open Database Connectivity" 2017) fournit plus détails sur ces connecteurs).

Le tableau 4 ci-dessous résume une comparaison entre Hive et Pig.

Tableau 4 Hive vs Pig

		Hive	Pig
Langage		HiveQL (similar to SQL)	PigLatin
Partitions		Oui	No
Schémas des données		Oui (données structurées)	Non (données non-structurées)
Standalone		Non	Oui
Performance moyenne pour un fichier de 6 G	**Arithmétique**	2633.72 sec	423.63 sec
	Groupe	141.36 sec	497.61 sec
	Jointure	4388.34 sec	1045.34 sec

En moyenne, Pig est 33% plus rapide que Hive. Pig offre également plus d'optimisation et de contrôle sur le flux de données (Urmila 2016).

b) Stream Processing

Les outils les plus utilisés dans le stream processing sont :

- **Spark** (“Apache SparkTM - Lightning-Fast Cluster Computing” 2014) est conçu pour traiter des données en micro batch. Ces données pourraient être structurées, semi-structurées ou non structurées et provenir de différentes sources (batch ou stream ou micro-batch). De plus, Spark possède des librairies telles que Spark SQL, Spark MLlib, SparkR qui simplifient l'analyse et le traitement des données massives. En outre, Spark s'appuie sur les mêmes principes Hadoop (calcul distribué, MapReduce), sauf que les données sont directement transmises à la mémoire pour traitement. Ainsi, il évite au système de lire à partir du disque chaque fois que des données sont nécessaires pour une opération. Il dispose de plusieurs modes de déploiement ; en mode **Standalone** ou dans un **Cluster Hadoop YARN**. La différence entre les deux modes :

  En mode **Standalone**, une version compilée de spark doit être placée sur tous les nœuds du cluster. Pour exécuter un traitement sur un cluster Spark, il est nécessaire de soumettre une application dont le traitement sera contrôlé par un pilote (driver). Dans ce

mode, spark prend en charge deux modes de déploiement : D'une part le mode **client**, où le pilote est lancé dans la même machine que le client qui soumet l'application. D'autre part, le mode **cluster**, où le pilote est créé à l'intérieur du cluster. Un cluster Spark se compose d'un maître et d'un ou plusieurs travailleurs (workers). Le maître a la responsabilité de gérer le cluster. Les travailleurs sont les exécuteurs et apportent des ressources au cluster (mémoire et cœurs de traitement).

Dans un cluster **Hadoop-Yarn**, il existe deux modes de déploiement pour lancer des applications **Spark** : en mode cluster, le pilote spark ne s'exécute pas sur la même machine que le client mais dans un processus maître d'application géré par YARN dans le cluster. En mode client, le pilote s'exécute dans le processus client tandis que le maître d'application est utilisé uniquement pour demander des ressources à YARN.

Spark a plus d'avantages que MapReduce, il est facile à utiliser et fonctionne plus rapidement. En outre, il permet aux applications dans Hadoop de fonctionner jusqu'à 100 fois plus vite en mémoire et 10 fois plus vite sur le disque (“Apache SparkTM - Lightning-Fast Cluster Computing” n.d.). En mode Standalone et en utilisant 10 fois moins de machines que MapReduce (210 contre 2100), Spark a trié 100 To de données (1 trillion d'enregistrement) en 23 minutes, contre 72 minutes (“Apache Spark Officially Sets a New Record in Large-Scale Sorting” 2014). Une étude comparative entre MapReduce et Spark est détaillée dans (“Apache Spark Officially Sets a New Record in Large-Scale Sorting” 2014)(Neumann n.d.).

- **Storm** (“Apache Storm” 2015) : a été conçu pour être distribué et tolérant aux pannes, il permet le traitement ou le calcul en temps réel. Il manipule les fichiers d'une manière similaire à OS-Unix, ce qui le rend relativement simple à utiliser.

- **Flink** (“Apache Flink: Scalable Stream and Batch Data Processing” 2014) : est un framework de traitement distribué, qui est plus relié à Hadoop (YARN) que Spark. Il fonctionne à grande échelle et fournit des résultats précis même en cas de données retardées (late-arriving) ou "hors service". En outre, il fournit un modèle d'exécution supplémentaire qui est le traitement par lot.

- **Samza** (“Samza - Documentation” 2014) : est un framework de traitement de flux distribué. Il est basé sur Apache Kafka pour la messagerie et Apache Hadoop YARN pour fournir la tolérance aux pannes et la gestion de ressources.

La table 5 montre une comparaison de ces solutions et la figure 8 montre les tendances de recherche pour ces framework au cours des cinq dernières années selon les statistiques de Google (“Google Trends” 2018). Ces frameworks ont deux tendances : ils peuvent stagner (et être remplacés par d'autres) ou évoluer.

Tableau 5: Spark vs Storm vs Flink vs Samza

	Spark	**Storm**	**Flink**	**Samza**
Modèle de streaming	Micro-Batching	Native	Native Hybrid	Native Streaming
Latence	Moyen (en fonction de la taille du lot)	Très bas	Bas	Bas
Maturité	Élevé	Élevé	Bas	Moyen
Auto-évolutif	Oui	Non	No	No
Tolérance aux pannes	RDD based check pointing	Record ACKs	Check pointing	Log-based
Outils d’apprentissage automatique associés	MLib, Mahout, H2O	Samoa	Flink-ML, Samoa, Mahout	Samoa

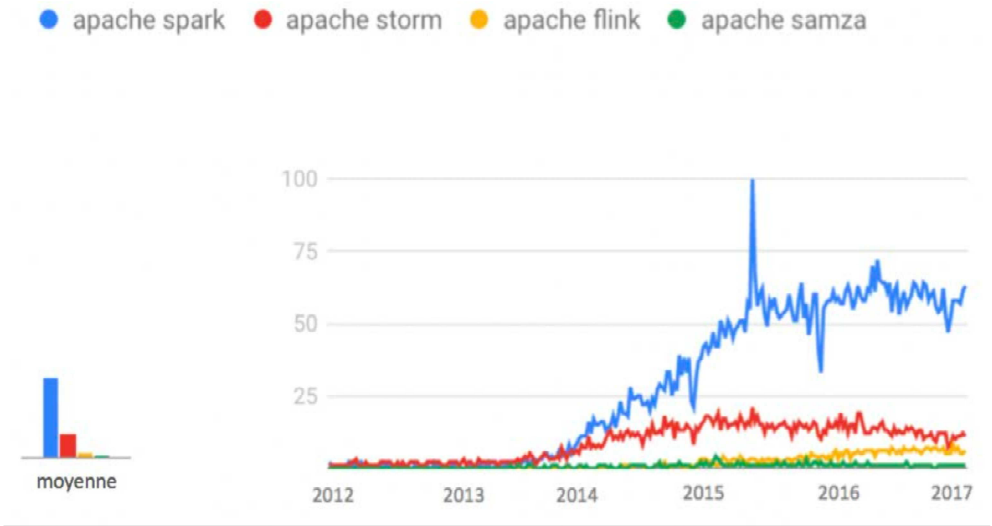


Figure 8: Tendances de recherche de Spark, Storm, Flink et Samza

Dans la section suivante, nous montrons comment l'apprentissage automatique peut être utilisé pour répondre aux besoins de données volumineuses.

c) L'apprentissage automatique

Dans le big data, les données augmentent rapidement et les outils traditionnels d'apprentissage automatique deviennent insuffisants pour le traitement en temps réel. Cette problématique a donné naissance à d'autres frameworks sophistiqués basés sur l'apprentissage automatique pour répondre à de nouveaux besoins :

- **Mahout** ("Apache Mahout: Scalable Machine Learning and Data Mining" 2014): est un projet Apache dont le but est de construire des bibliothèques d'apprentissage évolutives. Il fournit par exemple des algorithmes de partitionnement des données ou de classification automatique. Il peut être intégré à plusieurs frameworks dont Flink ("Apache Mahout: Scalable Machine Learning and Data Mining" n.d.).

- **MLib** ("MLlib | Apache Spark" 2014) : est une bibliothèque d'apprentissage automatique qui s'intègre dans les APIs sparks, son but est de rendre les algorithmes d'apprentissage automatique évolutifs et faciles.

- **Samoa** ("Apache SAMOA" 2014): est un framework d'apprentissage automatique distribué et en temps réel. Il contient une abstraction de programmation pour les algorithmes de streaming distribués. Samoa offre la possibilité de construire en ligne, en temps réel et d'évaluer des modèles d'apprentissage automatique à partir des flux de données. De plus, il permet de développer de nouveaux algorithmes d'apprentissage machine sans traiter la complexité des moteurs de traitement de flux distribués sous-jacents.

- **H2o**("H2O.ai" 2016): est une plate-forme d'apprentissage open source qui fonctionne comme un produit plutôt qu'un projet. H2o est utilisé pour le training des modèles d'apprentissage automatique par lots et fournit plus de fonctionnalités telles que la validation et le scoring des modèles. Cependant, il n'a pas été beaucoup utilisé dans la recherche académique

La création de valeur passe par l'exploitation des données, il est donc important, à l'ère du big data, d'adopter une vision centrée sur les données. Ensuite, il est possible de tirer de la valeur de toutes les données, qu'elles soient internes, externes ou des données présentes en masse mais non exploitées.

d) La recherche d'information

La recherche d'information permet d'effectuer des recherches sophistiquées dans un corpus. Les outils les plus connus pour le big data sont :

- **ElasticSearch** ("Elastic - Home" 2014): réalise et combine différentes requêtes sur des données structurées et non structurées. Il dispose également d'un connecteur ES-Hadoop qui permet d'utiliser Hadoop et offre une fonctionnalité spécifique appelée "Percolator" qui enregistre les requêtes et notifie en cas de modification du résultat.

- **Solr** ("Apache Solr -" 2015): est un peu plus rigoureux et contraignant comparé à ElasticSearch, car il consiste à définir le méta-modèle. Il prend en charge l'écriture et la lecture sur HDFS, mais n'utilise pas MapReduce pour la recherche. Au contraire, il a un outil appelé MapReduceIndexerTool qui permet le traitement des données Solr. Dans le tableau 6, nous fournissons une comparaison entre Solar et ElasticSearch.

Tableau 6 : Solr vs ElasticSearch

	Solr	**ElasticSearch**
Outils de synchronisation	Non	Oui (River)
Déduplication	Oui	Non
Output	JSON, XML, PHP, Python, Ruby, CSV, Velocity, XSLT, native Java	JSON, XML/HTML
Multiple document types par schéma	Un ensemble de champs par schéma, un schéma par cœur	Oui
Inter-index joins	Oui	Non
Mécanisme de synchronisation pour les base de données	A définir	Oui (River)

Les moteurs de recherche sont essentiellement conçus pour faire de la recherche. Ils sont plus efficaces pour la lecture que pour la mise à jour des données. Par conséquent, il est préférable de les combiner avec d'autres méthodes analytiques.

Dans la partie finale de ce chapitre, nous présentons un ensemble d'applications qui s'appuient sur ce type d'outils d'analyse pour extraire des informations pertinentes dans différents domaines tels que la santé, les affaires et la ville intelligente. Ensuite, nous soulignons les outils couramment utilisés dans ces secteurs.

4.2.3. Visualisation

Certains des outils de visualisation de big data populaires sont :

- **Tableau** ("Business Intelligence and Analytics | Tableau Software" 2017): permet de créer des présentations interactives et de gérer des jeux de données volumineux et en évolution rapide. il peut être intégré avec plusieurs solutions big data, telles que Hadoop,et permet de se connecter avec des données provenant de différentes sources.
- **Qlikview** ("Analyses Guidées | Logiciel de Business Intelligence | QlikView" 2016):est un outil de Business Intelligence (BI) et de visualisation big data. Il combine des données provenant de diverses sources. De plus, son package Qliksense permet de gérer l'exploration de données et de découvrir des patterns.
- **Spotfire** ("Data Visualization & Analytics Software | TIBCO Spotfire" 2016): il permet d'analyser visuellement les big data et les données textuelles en 27 langues. Les connecteurs big data de Spotfire supporte trois types d'accès de données ; in-datasource, in-memory.
- **Microsoft Power BI (MS BI)** ("Power BI | Outils Décisionnels de Visualisation Interactive Des Données" 2017): est une suite d'outils d'analyse métier de Microsoft qui permet d'analyser, de partager et de visualiser des données volumineuses via plusieurs conseils, notamment les ordinateurs de bureau, les tablettes et les smartphones. Il permet de se connecter à différentes sources telles que les feuilles de calcul Excel, les données en continu et les données sur les services cloud.

Le tableau 7 présente brièvement une comparaison entre les outils présentés.

Tableau 7: Qlikview vsTableau vs Spotfire vs Microsoft Power BI

	Tableau	**Qlikview**	**Spotfire**	**MS BI**
Vitesse d'implémentation	Bien	Elevé	Bien	Moyen
Évolutivité	Bien	RAM limité	Matériel limité	Bien
Intégration de données	Excellent	Très bien	Très bien	Très bien

Visualisation interactive	Très bien	Excellent	Très bien	Aussi bien qu'Excel
Drill-Down visuel	Bien	Excellent	Très bien	Moyen
Multidimensionnel	Très bien	Aucun	Aucun	Excellent
Modélisation et analyse	Sous la moyenne	Sous la moyenne	Excellent	R&D

4.3. Applications

4.3.1. La santé

En santé, le big data correspond à l'ensemble des données sociodémographiques et sanitaires disponibles auprès de sources internes et publiques. Les données dans le secteur de la santé augmentent à un rythme impressionnant, elles devraient être multipliées par 50 d'ici 2020, notamment grâce à l'évolution de la génomique, du matériel médical connecté, de la numérisation des dossiers médicaux et de l'utilisation d'applications de santé mobiles et de capteurs. L'utilisation de ces données présente de nombreux intérêts tels que l'identification des facteurs de risque de la maladie, l'aide au diagnostic, la sélection et le suivi de l'efficacité des traitements.

a) Pronostics et des diagnostiques pour les patients :

Thommandram et al. (2013) ont développé un système qui permet de détecter si un patient est en train de vivre une crise cardiorespiratoire. On dira d'une crise qu'elle est cardiorespiratoire lorsqu'un patient présente une certaine combinaison d'insuffisances physiologiques : une pause dans la respiration, une baisse de la saturation en oxygène du sang, et une diminution de la fréquence cardiaque. Leur système, appelé Artémis, est conçu pour recevoir des flux de données physiologiques du patient à travers différentes machines (la fréquence cardiaque, l'onde d'impédance respiratoire et la saturation en oxygène du sang) et diagnostiquer en temps réel le type de crise cardiorespiratoire que le patient présente. Ce qui permettrait aux médecins d'agir plus vite.

Nair, Shetty, and Shetty (2018) ont développé un système basé sur Spark qui prédit l'état de santé en temps réel. Dans ce système, l'utilisateur tweete ses attributs de santé. Ensuite, l'application exécute le modèle d'apprentissage automatique sur les attributs extraits des tweets pour prédire l'état de santé de l'utilisateur qui lui est envoyé en temps réel pour prendre les mesures appropriées.

Yan et al. (2016) ont proposé une nouvelle méthode basée sur l'algorithme KNN (k-Nearest Neighbors) pour détecter efficacement les valeurs aberrantes dans les données de santé à grande échelle. Cette méthode a été implémentée sur la plateforme Hadoop. Elle a surpassé les approches traditionnelles telles que le KNN et le LOF (Local Outlier Factor) en termes de précision et d'efficacité de traitement.

Afin de mesurer les impacts de la pollution de l'air sur la santé, Chen et al. (2017) ont utilisé un modèle spatial de Durbin et les données d'enquête sur la mortalité par cancer du poumon et par les maladies respiratoires dans 116 villes chinoises sur la période 2006-2012. En outre, ils ont estimé les taux de mortalité annuels et les frais médicaux induits par la pollution de l'air.

b) Modèles modernes et architectures système pour gérer de grandes quantités d'informations :

Goli-Malekabadi, Sargolzaei-Javan, and Akbari (2016) ont proposé un modèle basé sur le NoSQL et appuyé sur un environnement cloud pour gérer les big data émanant des informations de santé. L'efficacité du modèle a été mesurée en le comparant avec un modèle de données basé sur le modèle relationnel. Cette comparaison est basée sur le temps d'exécution des requêtes, la préparation des données, la flexibilité et les paramètres d'extensibilité.

Rasyid et al. (2016) ont construit une plate-forme d'application appelée EepisCure, basée sur Hadoop et Hive, afin d'intégrer et de visualiser les données collectées à partir de capteurs bruts. Ces capteurs forment un réseau de surface corporelle sans fil (Wireless Body Area Network) qui sont des dispositifs informatiques portables permettant de surveiller l'état d'un corps humain.

Ma et al. (2016) ont élaboré une architecture, appelée « Big Health ». Le système proposé est basé sur le big data, le cloud, et l'Internet des objets (les smartphones en particulier) pour surmonter les problèmes de santé mondiaux tels que la répartition inégale des ressources médicales et l'augmentation des dépenses médicales.

Ho et al. (2015) ont mis l'accent sur la connexion de la surveillance de la qualité de l'air dans un environnement domestique à un système de rapport de santé personnel afin de mettre en place un système d'alerte. Ce système pourrait identifier les zones hautement polluées et rappeler aux individus de prendre des précautions personnelles pour éviter d'inhaler des polluants.

Ta, Liu, and Nkabinde (2016) ont proposé une architecture générique, basée sur storm, Kafka et NoSQL Cassandr, pour l'analyse des soins de santé. Cette architecture peut prendre en charge l'analyse des soins de santé en fournissant un traitement par lots et par flux. Elle pourrait être utilisée pour les découvertes pharmaceutiques, les systèmes d'aide à la décision clinique et le diagnostic assisté par ordinateur.

c) Réduction des coûts de traitement et amélioration de la santé de la population

Gupta and Tripathi (2016) ont discuté de l'impact de l'analyse du big data et ses nouvelles technologies dans le secteur de l'assurance et de la façon dont il pourrait transformer les soins de santé en milieu rurale et conduire éventuellement au progrès économique en réduisant les coût de soins.

En utilisant les nouvelles technologies du big data, Kumar, S, and Swarnalatha (2016) ont analysé les dossiers, les rapports, les symptômes et les réactions des patients concernant l'assurance maladie électronique afin d'améliorer la qualité du service. Permet une classification des centres de santé en fonction de la qualité de service par type de maladie et de coût.

4.3.2. Affaires et marketing

L'analyse du big data n'est pas seulement une opportunité mais devient une nécessité pour améliorer la qualité des organisations modernes (prise de décision et expérience client). De nombreuses études ont confirmé que s'appuyer sur les big data, pour les domaines commerciaux et marketing, est un facteur clé de succès pour un développement continu. Par exemple, Nous pouvons évaluer le succès d'une campagne publicitaire ou d'un lancement de nouveaux produits, déterminer les versions d'un produit ou d'un service les plus populaires en fonction des caractéristiques démographiques.

a) Finance

Dans le domaine de la finance, Dong, Yang, and Tian (2015) ont évalué divers modèles mathématiques en se focalisant sur une forme limitée des big data issues de la finance, à savoir les données financières à haute fréquence. En outre, ils ont introduit une méthode basée sur l'analyse de données financières à haute fréquence de l'indice composite de Shanghai pour prédire l'évolution du marché financier.

De leur côté, Zamani-Dehkordi et al. (2016)ont proposé une approche pour analyser l'effet de la production éolienne sur le prix de gros de l'électricité. La méthode a été appliquée au marché de l'Alberta comme cas d'étude. Les résultats montrent que l'augmentation de la production d'énergie éolienne réduit les prix du marché de gros.

b) Publicité

En utilisant le big data dans le domaine de la publicité, Aivalis, Gatziolis, and Boucouvalas (2016) ont développé une application d'analyse qui évalue le facteur d'impact de deux paramètres (trafic dans le cite e-commerce et revenus des publicité) afin de fournir une publicité ciblée dans

les médias sociaux. La publicité contient des liens directs vers des listes de produits spécifiques présélectionnés, des groupes de produits ou des offres spéciales.

Pour le même sujet, Deng, Gao, and Vuppalapati (2015) ont proposé un framework pour l''analyse des données massives issues des appareils mobiles pour fournir des publicités et des recommandations basées sur divers critères tels que le profilage, l'historique de navigation, la localisation et les comportements d'accès.

Suguna, Vithya, and Eunaicy (2016) ont traité l'importance de l'analyse des fichiers journaux (logs) dans le monde du commerce électronique. Ils ont proposé un système, basé sur le prétraitement des logs web en utilisant Hadoop MapReduce, qui permet de prédire des pages préférées de l'utilisateur pour les activités commerciales e-commerce.

c) Marketing

Dans le domaine de la commercialisation, un nouveau modèle a été proposé par H. Zhang et al. (2016) pour prédire les acheteurs potentiels de voitures de luxe haut de gamme en utilisant le big data mining (réseaux de neurones et HDFS). Pour cela, ils ont utilisé plusieurs algorithmes, tels que la régression logistique et les réseaux de neurones, et se sont basés sur plusieurs attributs tels que les caractéristiques de l'utilisateur, le comportement de communication, l'attribut terminal (telephone, ordinateur…) et le cercle social. La contribution a également reposé sur les médias sociaux pour prouver l'efficacité du modèle.

Bollen, Mao, and Zeng (2011) ont examiné la corrélation entre l'humeur publique et les indicateurs économiques. Pour cela, ils ont utilisé les flux de données de Twitter pour prédire l'évolution des marchés DJIA (Dow Jones Industrial Average). Les résultats ont indiqué que la précision des prédictions de DJIA peut être significativement améliorée par l'inclusion de dimensions spécifiques de l'humeur publique.

Toujours en s'appuyant sur les réseaux sociaux, Zhang and Pennacchiotti (2016) ont présenté un système de recommandation avec un démarrage à froid. Ce système permet de prédire le comportement d'achat des utilisateurs dans les sites web e-commerce à partir de leurs profils Facebook. Cette étude fournit une analyse sur la corrélation des ensembles de profils Facebook et le comportement d'achat sur eBay en se basant sur Pearson's chi-square test.

Attigeri et al. (2015) ont développé une méthode qui consiste à recueillir des informations et des données sur les médias sociaux et à extraire les sentiments exprimés par les individus en utilisant Hadoop et la régression logistique. Le modèle a montré une étroite corrélation entre l'analyse des sentiments des données sociales et la performance boursière.

4.3.3. Ville intelligente

Nous avons passé en revue plusieurs exemples d'applications big data qui pourraient être utilisées pour améliorer la qualité de vie des citadins dans différents domaines. Par conséquent, ces applications pourraient être considérées comme guides de diverses applications de villes intelligentes. Par exemple, afin d'améliorer la santé des résidents ou de changer la façon dont les soins de santé sont dispensés dans une ville, les applications précédentes (dans la sous-section 3.3.1 et 3.3.2) pourraient être un élément crucial dans le développement des systèmes de soins de santé des villes intelligente. En général, la ville intelligente implique l'amélioration de plusieurs composantes urbaines telles que l'énergie, les transports, les administrations, la sécurité publique, comme discuté dans Nuaimi et al. (2015). En conséquence, de nombreux gouvernements ont encouragé le développement de villes intelligentes à travers le monde.

Les exemples suivants montrent à quel point le big data est important pour le développement d'une ville intelligente dans principalement 3 sous-domaines :

a) Conditions de vie des citoyens

Ce sous domaine comprend la maison, la communauté, les soins de santé et l'éducation. Pour améliorer ces secteurs, plusieurs travaux ont été proposés.

Wich and Kramer (2016) ont examiné les possibilités de recherche (technologies, analyse, et comportement) pour les maisons intelligentes. Egalement, ils ont exploré l'amélioration des services de maison intelligente tels que la sécurité, la santé et le confort en intégrant à la fois l'analyse du big data et les services de réseaux sociaux.

Xu et al. (2016) ont construit un modèle basé sur Hadoop et les réseaux de neurones afin de prédire le comportement des utilisateurs dans un système de maison intelligente traditionnel. L'analyse est effectuée sur une combinaison de nombreuses données comportementales générées par l'utilisateur et les paramètres environnementaux (capteurs et équipements tels que le climatiseur). Le but de cette étude est de fournir des services intelligents qui facilitent les activités quotidiennes tels que le contrôle quotidien des équipements (température, lumière).

Pour l'apprentissage intelligent, Udupi, Malali, and Noronha (2016) ont exploré davantage les recherches possibles dans un système intégré et ont proposé de nouveaux modèles pour l'intégration des frameworks big data et de la technologie intelligente dans le paradigme de l'apprentissage en ligne, qui mène à un apprentissage intelligent.

Jagtap et al. (2016) ont intégré les réseaux sociaux et les outils d'analyse du big data afin de fournir un système de recommandations dans un contexte d'éducation intelligente. Ce système pourrait, à partir des profils qu'il définit, proposer des contenus appropriés par rapport aux

domaines d'intérêt, les amis et des professeurs intéressés par le même domaine, des vidéos et des livres électroniques.

b) L'administration publique et les services

Cet axe regroupe lui-même plusieurs domaines tels que la supervision de la sécurité publique, le trafic intelligent et le tourisme intelligent.

Dans le trafic intelligent, les technologies de géolocalisation peuvent fournir des informations utiles sur un certain nombre de problèmes différents tels que la congestion, les retards, la fiabilité du transport public. En effet, Raghothama, Shreenath, and Meijer (2016) ont utilisé des données GTFS (General Transit Feed Specification) et des données provenant d'autres sources afin de comprendre les facteurs influençant les retards de transport public dans les villes.

Rathore et al. (2015) ont proposé un mécanisme basé sur Spark et d'autres technologies afin de mettre en place un système de transport intelligent. Ce système aide à maîtriser le vaste volume de données collectées en continu par les capteurs routiers tel que l'information globale sur le trafic, le réseau et la vitesse des véhicules afin de fournir des services intelligents aux citoyens et aux autorités (économiser le carburant en gérant efficacement la route pour atteindre la destination, ainsi que pour se protéger contre la pollution de l'environnement en réduisant le trafic de la zone congestionnée).

Dans le contexte du tourisme intelligent, Chua et al. (2016) ont introduit une nouvelle approche basée sur les données issues des réseaux qui contiennent une géolocalisation. Cette approche permet de fournir des informations sur les flux touristiques telles que les informations spatiales, temporelles et démographiques plus détaillées sur les mouvements touristiques, en comparaison avec la compréhension actuelle des flux touristiques.

Pour la sécurité intelligente, une approche novatrice a été proposée par Hochstetler, and Fu (2016) qui vise à minimiser le temps d'arrivée des policiers et le nombre total de policiers en patrouille en utilisant l'analyse du trafic en temps réel. Les chercheurs se sont basés sur l'entropie informationnelle pour étudier le problème de la planification des patrouilles de police et l'emplacement des crimes.

c) La gestion des ressources

Comme le gaspillage de l'énergie a d'énormes conséquences économiques et environnementales, il a suscité un vif intérêt dans le but d'optimiser la gestion et la consommation des ressources, telles que l'eau, l'électricité et l'agriculture. Pour aborder ces sujets intéressants, Vajjala (2016) a proposé un framework basé sur Spart et Cassandra pour analyser

les données de consommation d'énergie et de température intérieure d'un groupe de maisons d'habitation. Le but de cette étude est de calculer les temps de réponse à la demande pour répondre aux niveaux de confort et aux normes de conformité définis par l'utilisateur. Dans le même contexte, un Framework basé sur Hadoop et d'autres technologies a été conçu par Vaidya and Deshpande (2015) pour gérer les données dans le secteur de l'énergie afin de réduire le gaspillage d'énergie. Dans le domaine de l'agriculture, l'étude vise à mieux comprendre l'état de l'art des applications big data dans l'agriculture intelligente et identifier les défis socio-économiques à résoudre. La revue montre que la portée des applications big data dans l'agriculture intelligente va au-delà de la production primaire ; elles influencent toute la chaîne d'approvisionnement alimentaire. En effet, elles permettent de fournir des informations prédictives sur les opérations agricoles et de piloter les décisions opérationnelles en temps réel.

4.4. Outils couramment utilisés dans les applications

Afin de comprendre l'utilisation des plateformes au travers des exemples proposés précédemment, nous avons classifié (table ci-dessous) ces dernières par domaines / sous-domaines et par rapport aux outils d'analyse big data utilisés.

Nous avons remarqué que Hadoop, MapReduce et spark sont les frameworks les plus largement utilisés pour le traitement du big data dans les recherches académiques. Bien que Strom, Flink et Samza ont des capacités de traitement en temps réel, ils n'ont pas été particulièrement utilisés dans ce domaine. Nous avons également noté que la plupart des contributions ne soulignaient pas pourquoi elles favorisaient certains frameworks par rapport à d'autres. Pour cela, nous avons fourni une étude comparative des outils existant pour accompagner la chaine de valeur du big data.

Tableau 8: Outils big data couramment utilisés dans les applications

	Autres
Samza	(Thommandram et al. 2013) (Chen et al. 2017)
	(Goli-Malekabadi, Sargolzaei-Javan, and Akbari 2016) (Rasyid et al. 2016)
	(Gupta and Tripathi 2016)
	(Xu et al. 2016)
	(Rathore et al. 2015)
	(Dong, Yang, and Tian 2015)] (Zamani-Dehkordi et al. 2016) (Deng, Gao, and Vuppalapati 2015) (Attigeri et al. 2015)

		Hadoop	MapReduce	Spark	Storm	Flink
Santé	**Pronostics et diagnostiques**	(Yan et al. 2016)		(Nair, Shetty, and Shetty 2018)		
	Modèles et architectures	(Rasyid et al. 2016) (Ma et al. 2016) (Ta, Liu, and Nkabinde 2016)		(Ma et al. 2016) (Ta, Liu, and Nkabinde 2016)	(Ma et al. 2016)	
	Réduction des coûts de traitements	(Gupta and Tripathi 2016)]	(Gupta and Tripathi 2016)0] (Kumar, S, and Swarnalatha 2016)	(Gupta and Tripathi 2016)		
Ville intelligente	**Conditions de vie des citoyens**	(Xu et al. 2016) 4](Miguel, Caballé, and Xhafa 2015)	(Miguel, Caballé, and Xhafa 2015)			
	Administration publique et services	.(Rathore et al. 2015)		(Rathore et al. 2015)		
	Gestion des ressources	. (Vaidya and Deshpande 2015)	(Vaidya and Deshpande 2015)	Vajjala 2016		
	Affaires et marketing	(Suguna, Vithya, and Eunaicy 2016) (Deng, Gao, and Vuppalapati 2015)	(Suguna, Vithya, and Eunaicy 2016)	(Deng, Gao, and Vuppalapati 2015)6]		

Il est important de mentionner que les outils de traitement big data sont en constante évolution, de nouveaux outils sont présentés chaque jour. Ces derniers offrent des fonctionnalités plus au moins adaptées aux objectifs d'études et à la nature des données. Par conséquent, les outils utilisés pour construire des environnements basés sur le traitement doivent être bien sélectionnés.

5. Conclusion

L'histoire du big data étant souvent présentée comme déconnecté du passé, nous avons consacré la première partie de ce chapitre à l'histoire de la façon dont le volume de données est devenu important. Pour cela, nous sommes remontés aux années 40, où les premières tentatives de quantifier le taux de croissance du volume de données ont eu lieu. Ensuite, nous avons présenté la chronologie des inventions majeures qui ont causé l'explosion des données notamment l'émergence du Web 2.0.

Suite à cette explosion de données, le big data n'est plus constitué uniquement par des données structurées. Nous avons donc présenté d'autres types de données tels que les données semi-structurées et non-structurées qui ont émergé dans les applications. Ensuite, nous avons rappelé les différentes définitions du big data, qui, continuent à être mises à jour en fonction des évolutions du domaine. Nous avons également présenté les 4 phases majeures de la chaîne de valeurs du big data qui sont : l'ingestion, l'analyse et la visualisation des données. Toutefois, les éditeurs proposent des solutions et des outils permettant d'accompagner la chaine de valeur du big data de façon plus au moins adaptée aux objectifs d'études et à la nature des données, d'où notre étude comparative des outils et méthodes existants.

Dans la deuxième partie de ce chapitre, nous nous sommes intéressés aux outils accompagnant la chaine de valeur du big data. Pour cela, nous avons présenté son système de référence, à savoir Hadoop et ses outils associés. Nous avons ensuite examiné et comparé les dernières technologies et frameworks pour chaque phase de la chaîne de valeurs du big data. Nous avons également mis l'accent sur les spécificités de ces outils pour tirer pleinement parti des possibilités offertes par le big data.

Finalement, nous avons apporté une vue assez large de cas d'utilisation de ces outils dans différents domaines (santé, finance et marketing, et ville intelligente), et nous mis en avant les outils les plus couramment utilisés.

Dans le chapitre suivant, nous exposons les différentes méthodes d'analyse des « big social data », données issues des réseaux sociaux, ainsi que leurs applications dans plusieurs domaines tels que la santé, l'éducation, le tourisme et la politique.

Chapitre 3 Analyse des big social data

1. Introduction

Au fil des années, l'objectif principal de l'utilisation d'Internet s'est étendu pour permettre à des milliards d'utilisateurs de se connecter les uns aux autres, de partager des informations, d'exprimer et diffuser leurs idées et leurs opinions sur un sujet, et montrer leurs attitudes envers un contenu particulièrement via les réseaux sociaux. Toutes ces actions, qui s'accumulent constamment dans les médias sociaux, génèrent des données à grand volume, à haute vélocité, de grande variété, à haute valeur et à haute variabilité, appelées big social data. En général, ce type de données se réfère à un ensemble massif d'opinions que l'on peut traiter pour déterminer les tendances des internautes ou évaluer leur degré de satisfaction sur un sujet (services, produits, événements, sujets ou personnes). L'analyse des big social data a suscité un vif intérêt dans plusieurs domaines tels que la politique, le marketing, ou la santé.

Comme les quantités importantes de données issues des réseaux sociaux sont des textes non structurés (près de 80% ; Ramanathan and Meyyappan (2013), l'exploitation de ces données fait de l'analyse de texte un facteur important pour l'extraction de connaissances et la fouille de données. C'est donc dans ce contexte que nous nous intéressons particulièrement aux méthodes d'analyse des données textuelles notamment la polarité des opinions (positive, négative ou neutre) et l'apprentissage automatique.

2. Texte mining

Le texte mining ou l'analyse textuelle désigne un ensemble de traitements informatiques qui permet d'extraire des informations utiles à partir de données textuelles non structurées. Son but est de décrire le contenu, la structure et les fonctions des messages contenus dans les textes afin d'en faire émerger des patterns[6]. Ce domaine d'analyse comprend des techniques émergentes et des algorithmes tels que le Traitement Automatique du Langage Naturel (TALN) et l'exploration de données.

Le texte mining regroupe deux phases principales : l'analyse et l'interprétation.

[6] Le mot anglais « pattern » est souvent utilisé pour désigner un modèle, une structure, un motif, etc.

2.1. La phase d'analyse

La phase d'analyse consiste à reconnaître les mots, les phrases, leurs relations et leurs rôles grammaticaux. Cette phase produit une forme de normalisation dans le but de standardiser et uniformiser le texte à travers plusieurs méthodes ou de déterminer automatiquement la langue d'un contenu donné. Elle consiste à résoudre des problèmes de catégorisation de texte en utilisant des méta-informations, la reconnaissance de caractères, etc.

Les processus communément utilisés dans la phase d'analyse sont :

- *La Tokenisation* : est le processus de conversion des chaîne de caractères en une liste de symboles (tokens en anglais). Les tokens sont des chaînes avec une signification assignée et identifiée. La tokenisation consiste à éliminer les « bruits » du texte source notamment les commentaires, les espaces blancs et la ponctuation.
- *La lemmatisation* : désigne l'analyse lexicale permettant d'associer la forme de base, appelée lemme, à chaque mot fléchi dans le texte. Cette opération permet donc de regrouper les variations flexionnelles (pluriel, conjugaisons…) ou dérivationnelles (verbes, adjectifs…) sous le même terme. Par exemple, le mot « faire », verbe à l'infinitif, est un lemme des différentes flexions sont « faisons », « fera », « faisait ». Ou encore, la lemmatisation de « jouets » est « jouet ».
- *Le Stemming* : est un processus de transformation des flexions (ou parfois des dérivés) en leur forme de base ou de racine. La racine d'un mot correspond à la partie du mot restante une fois que l'on a supprimé les préfixes et suffixes, à savoir son radical. Il existe des procédures spécialisées pour chaque langue. En anglais, le "Potter stemming" est le plus connu. Le stemming offre deux avantages : la rapidité (il se base sur des algorithmes simples ne faisant pas référence aux dictionnaires et règles de dérivation) et la possibilité de traiter les mots inconnus sans traitement spécifique. Néanmoins, il est moins approfondi que la lemmatisation et il peut conduire à des erreurs. Ceci est dû au « sous-stemming », qui supprime un petit suffixe, et le « sur-stemming », qui supprime un grand suffixe. Par exemple, « croûtons » est le pluriel de « croûton », donc enlever le « ons » serait un sur-stemming, tandis que roulons est une forme verbale de rouler, donc supprimer le « s » serait un sous- stemming.
- *Le Filtrage* : consiste à appliquer des filtres qui suppriment des mots vides ...
- *La Reconnaissance d'Entités Nommées (REN)* : est une sous-tâche de l'extraction de l'information dans des documents textuels. Elle consiste à étiqueter un texte avec des

balises et à rechercher les objets textuels catégorisables dans des classes d'étiquettes racines telles que personne, lieux, organisation, date et auxquelles s'ajoutent des classes selon le besoin applicatif (Jacquemin and Bush 2000). La REN s'appuie sur les systèmes statistiques qui utilisent pour leur part une grande quantité de données pré-annotées pour apprendre les formes possibles des entités nommées. Il n'est plus nécessaire donc de rédiger de nombreuses règles à la main, mais d'étiqueter un corpus qui servira d'outil d'apprentissage. Ces systèmes sont très coûteux en temps humain. Pour résoudre ce problème, récemment, des initiatives telles que DBpedia ou Yago cherchent à fournir des corpus sémantiques susceptibles d'aider à concevoir des outils d'étiquetage. Dans le même esprit, certaines ontologies sémantiques telles que NLGbase se sont largement orientées vers l'étiquetage. Une étude comparant les différentes méthodes d'extraction des entités nommées est présentée par Abdallah, Carman, and Haffari (2017).

2.2. La phase d'interprétation

Bien que l'analyse des données puisse aider à résumer et à identifier les principales constatations, il est toujours nécessaire d'interpréter les résultats et de tirer des conclusions. Cette phase d'interprétation est basée sur des méthodes de data mining (Han, Kamber, and Pei 2012a), dont le but est d'établir des modèles fiables de prédiction (Hung and Zhang 2008) (He, Zha, and Li 2013) (Han, Kamber, and Pei 2012c) (B. Liu, Cao, and He 2011).

Un des exemples d'application est la classification de courriers en spam. Cette sélection est une classification basée sur une propriété lexicale, la présence ou l'absence de mots-clés. Le critère de sélection peut être d'au moins deux types : la nouveauté ou la similarité. Celui de la nouveauté d'une connaissance consiste à essayer de découvrir des relations qui n'étaient pas explicites car indirectes entre deux éléments éloignés dans le texte. Celui de la similarité consiste à découvrir des textes qui correspondent le plus à un ensemble de descripteurs dans la requête initiale. Les descripteurs sont par exemple les noms et verbes les plus fréquents d'un texte (voir sous-section 3.2.1 pour plus de détails sur les descripteurs).

Comme évoqué plus haut, le processus d'analyse textuelle fait référence à différentes approches, permettant de filtrer, d'inspecter, de nettoyer, de transformer et de modéliser des données, que nous allons exposer dans les sous-sections suivantes.

3. L'analyse de sentiments

L'analyse de sentiments est apparu au début des années 2000, elle consiste à classifier la polarité en deux sentiments opposés tels que (aimer, détester) ou (positif, négatif) (P. Kumar, Manocha, and Gupta (2016) ; Boullier and Lohard (2012) ;He et al. (2015)). L'analyse d'opinion est utilisée à différentes fins et sur une variété de corpus. Elle est devenue essentielle dans plusieurs domaines tels que le marketing (en particulier, pour évaluer et suivre l'e-réputation d'une marque, pour traiter les avis des consommateurs, pour maintenir la relation avec le client et détecter l'état émotionnel), la santé (pour mesurer l'état de santé des patients et la qualité de soins délivrés), ou la politique (pour prédire les élections présidentielles),

La première étape de cette classification consiste à extraire des fonctionnalités textuelles telles que :

- *Présence vs Fréquence du terme* : Selon une étude réalisée par Pang, Lee, and Vaithyanathan (2002), la représentation d'un texte par un vecteur, dans lequel les éléments indiquent l'existence d'un terme (1) ou non (0), donne un meilleur résultat par rapport à la méthode de fréquence (fréquence d'occurrence du terme) pour la classification de polarité.
- *Fonctionnalités basées sur les termes* : Dans un texte, la position d'un mot (par exemple, au milieu ou près de la fin d'un document) peut affecter le sentiment global ou la subjectivité. Ainsi, les informations de position sont parfois codées dans les vecteurs caractéristiques utilisés (S.-M. Kim and Hovy 2006).
- *Analyse morphosyntaxique (Part of speech en anglais)* : permet d'expliquer comment un mot est utilisé dans une phrase. Il y a plusieurs parties principales du discours (aussi appelées classes de mots) notamment : les noms, les pronoms, les adjectifs, les verbes, les adverbes, les prépositions, les conjonctions et les interjections. L'analyse de sentiment par apprentissage automatique comme par lexique a plutôt un attrait vers les adjectifs comme démontré par (Hatzivassiloglou and Wiebe 2000).
- *Syntaxe* : l'intégration des relations syntaxiques dans l'analyse semble particulièrement pertinente avec les textes courts.
- *Négation* : Dans la méthode sac de mots (bag of words) à titre d'exemple, la négation n'est pas prise en compte. Les phrases "Je n'aime pas cet appareil" et "J'aime cet appareil" sont considérées comme similaires. D'où l'intérêt du traitement de la négation.
- *Fonctionnalités axées sur le sujet (Topic-oriented features)* : l'interaction entre le sujet et l'opinion. Par exemple, les phrases « L'entreprise A rapporte que les profits ont augmenté

» et « L'entreprise B rapporte que les profits ont augmenté » pourraient indiquer des types de nouvelles complètement différentes (bons ou mauvais) concernant le sujet du document.

On peut distinguer trois principales catégories d'analyse de sentiments : une approche basée sur l'analyse lexicale, une approche fondée sur l'apprentissage automatique et une approche hybride qui combine les deux premières. La recherche présentée dans cette thèse se concentre sur l'utilisation de méthodes basées sur l'approche lexicale. Néanmoins, nous discutons des trois approches en justifiant nos préférences. Nous concluons sur les lacunes dans l'état actuel de la recherche dans l'analyse de sentiments pour les big social data et pour lesquels cette thèse tente d'apporter des solutions.

3.1. Les approches basées sur l'analyse lexicale

Les approches d'extraction d'opinion fondées sur l'analyse lexicale consistent à extraire la polarité d'une phrase en utilisant une analyse sémantique des mots : cela signifie qu'une phrase est classée par ses instances (mots d'opinion) pour lesquelles des émotions sont déjà attribuées. Dans la littérature, les mots d'opinion sont également connus sous le nom de mots polaires ou mots porteurs d'opinion. Les mots d'opinion positifs sont utilisés pour exprimer certains états désirés tandis que les mots d'opinion négatifs sont utilisés pour exprimer les états non désirés. Des exemples de mots d'opinion positifs sont : épatant, fantastique, formidable. Les exemples de mots d'opinion négatifs sont : infect, horrible, abominable, épouvantable, ignoble.

Dans l'analyse lexicale, le texte d'entrée est converti en tokens. Si le token a une correspondance positive, son score est ajouté au total du score du texte d'entrée. Par exemple, si le mot « dure » a une correspondance positive dans la liste des mots d'opinion, le score total du texte est incrémenté du poids associé. Sinon, le score est décrémenté d'autant quand le mot est étiqueté comme négatif.

Pour générer la liste des mots d'opinion, il existe trois approches principales : une approche manuelle, une approche basée sur les dictionnaires, et une approche basée sur le corpus.

- *Approche manuelle* : cette méthode est précise mais elle est chronophage. Elle n'est donc généralement pas utilisée seule, mais utilisée lorsqu'elle est combinée avec des approches automatisées en tant que vérification finale.
- *Approche basée sur un dictionnaire* : approche dans laquelle un petit ensemble de mots d'opinion annotés est collecté manuellement et ensuite développé en recherchant leurs synonymes et leurs antonymes dans un dictionnaire. Les mots nouvellement trouvés sont ajoutés à la liste de départ. Ce processus est itératif et s'arrête quand aucun nouveau mot

n'est trouvé. Une fois le processus terminé, une inspection manuelle peut être effectuée pour corriger les éventuelles erreurs. Plusieurs chercheurs (Esuli and Sebastiani 2006), (Hu and Liu 2004) ont utilisé cette approche et ont généré des listes de mots d'opinion.

Cependant, les approches basées sur les dictionnaires présentent une lacune majeure. Elles ne sont pas toujours adaptées pour prendre en compte le changement de vocabulaire induit par le changement de domaine. Par exemple, pour la batterie d'un ordinateur portable, si elle est « large », c'est généralement négatif mais pour un écran, s'il est « large », c'est positif.

D'autres mots semblent en revanche neutres, comme « Email », mais pourraient changer selon le contexte. En effet, dans le cadre des élections présidentielles américaines, le terme Email associé à Hillary, qui a contrevenu aux lois fédérales en utilisant un compte de courriel personnel pour les affaires du gouvernement, prend une connotation négative.

- *Approche basée sur le corpus* : approche dans laquelle les dictionnaires sont utilisés pour annoter les mots ainsi que le contexte pour lequel la polarité est valide. Cette approche commence par une liste de mots d'opinion qu'elle élargit ensuite en se basant sur un large corpus.

  Hatzivassiloglou and McKeown (1997) ont proposé une technique « cohérence du sentiment » qui commence par une liste d'adjectifs d'opinion, et identifie des mots d'opinion adjectifs supplémentaires et leurs orientations en utilisant un ensemble de contraintes linguistiques ou de connexions de phrases (OU, MAIS, SOIT, NI NI …). L'une des contraintes concerne la conjonction « ET », qui indique que les adjectifs associés ont généralement la même orientation. Par exemple, dans la phrase « Cet homme est brave et gentil », si « brave » est connue pour être positif, on peut en déduire que « gentil » l'est également. C'est parce que les gens expriment généralement la même opinion des deux côtés d'une conjoncture. La phrase suivante est plutôt contre nature : « Cet homme est brave et autoritaire » Si elle est changée en « Cet homme est brave mais autoritaire », elle devient acceptable. L'apprentissage est appliqué à un grand corpus pour déterminer si deux adjectifs dans une même phrase (« conjoined adjectives »), ont des orientations identiques ou différentes.

  En pratique, ce n'est pas toujours cohérent. En effet, Ding, Liu, and Yu (2008) ont montré que le même mot pouvait indiquer des orientations différentes dans des contextes différents, même dans le même domaine. Par exemple, dans le domaine de l'ordinateur portable, le mot « long » exprime des opinions opposées dans les deux phrases : « La

durée de vie de la batterie est longue » (positive) et « Le temps de démarrage est long » (négatif). Par conséquent, la génération des mots d'opinion en fonction du domaine devient insuffisante. Pour cela, ils ont proposé de considérer à la fois les mots d'opinion possibles et les aspects : utiliser le couple (aspect, opinion_word) comme contexte d'opinion, par exemple, le couple (« démarrage », « long »). Leur méthode détermine ainsi les mots d'opinion et leurs orientations ainsi que les aspects pour lesquels l'orientation est valide.

Les approches d'analyse lexicale sont généralement utiles si nous connaissons le sujet. Elles doivent donc être combinées avec d'autres techniques telles que la reconnaissance d'entités nommées pour établir des relations entre la polarité et le sujet. Néanmoins, les problèmes de syntaxe et de grammaire rendent souvent cette tâche difficile à traiter.

3.2. Les approches basées sur l'apprentissage automatique

Dans ces approches, la machine est entraînée à détecter des modèles dans un corpus en la faisant apprendre sur un premier corpus test. Ce type d'apprentissage est similaire à l'apprentissage humain des expériences passées pour acquérir de nouvelles connaissances afin d'améliorer sa capacité d'effectuer des tâches dans le monde réel. Dans l'apprentissage automatique, la machine apprend à partir de données collectées dans le passé, qui représentent des expériences passées dans certaines applications du monde réel.

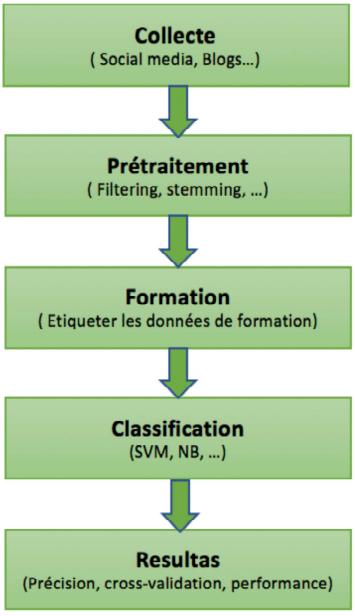


Figure 9: Les différentes étapes de l'apprentissage automatique

Comme illustré dans la figure 9, l'apprentissage automatique implique cinq étapes : la collecte de données, le prétraitement, la formation, la classification et les résultats

- *Collecte de données* : Les données à analyser sont collectées à partir de diverses sources telles que les blogs, les réseaux sociaux (Twitter, Facebook, LinkedIn, etc.) en fonction du besoin de l'application, du domaine et du contexte étudiés.
- *Prétraitement* : où les données collectées sont nettoyées et préparées pour être introduites dans le « classifieur". Le prétraitement est une étape de la plus haute importance et a un impact direct sur la qualité de l'opération de classification. Le nettoyage des données textuelles est effectué en plusieurs étapes et comprend la tokenization, le stemming, le filtrage (sous-section 2.1).
- *Formation* : Elle consiste à étiqueter une collection de données à la main pour générer les données de formation. La méthode la plus couramment utilisée est la production participative (le crowd-sourcing). Ces données sont introduites à l'algorithme choisi à des fins d'apprentissage.
- *Classification* : Le « classifieur » est entraîné à détecter des modèles ou des motifs dans le corpus en se basant sur des descripteurs (sous-section 3.2.1). Après avoir terminé la formation et construit le modèle de prévision, le « classifieur » est déployé sur les nouvelles données pour extraire les sentiments. Les méthodes de classification spécifiques à l'analyse des sentiments sont discutées dans la sous-section 3.2.3.
- *Résultats* : Les résultats sont tracés en fonction du type de représentation sélectionnée. Ensuite la performance du classificateur est mesurée en fonction de plusieurs méthodes notamment la précision, le rappel, F-score, la validation croisée (sous-section 3.2.4).

3.2.1. Les desctipteurs

Les systèmes de classification sont incapables de traiter directement des textes bruts. En effet, ils se basent sur la représentation des textes, appelée descripteur, dictionnaire, ou encore histogramme, qui permet d'extraire les caractéristiques en conservant l'information contenue dans le texte. La construction des descripteurs est donc une étape critique pour les performances des systèmes de classification.

Généralement, la construction des descripteurs consiste à représenter chaque document ou message par un vecteur, dont les composantes sont les termes rencontrés dans le texte et auxquelles il est possible d'associer un poids notamment le nombre d'occurrence et cooccurrence. La majorité des systèmes de classification représente les données textuelles par un vecteur, dans lequel les éléments indiquent la présence d'un terme (1) ou son absence (0).

Dans la littérature, plusieurs méthodes sont proposées pour le choix des termes et les poids qui leur sont attribués. Dans ce qui suit, nous allons exposer les différentes façons de construire les descripteurs pour représenter les données textuelles.

a) Représentation par stemmes

Comme expliquée précédemment, cette représentation consiste à ramener un mot à sa forme canonique (le stemme). Ainsi, le stemming permet de rassembler dans le même groupe, des mots qui font partie du même lexique et de significations très analogues. Par exemple, le stemme de « optimal » est « OPTIM » et regroupe également : Optimiser, optimisation, optimalisation. Cependant, cette opération n'étant pas basée sur des contraintes linguistiques, peut générer, des erreurs en regroupant des mots de différentes significations. Par exemple, en anglais les mots « univers » et « university » ont la même forme canonique (« univers ») alors qu'ils n'ont pas forcement la même signification.

b) Représentation par lemme

Le but de la lemmatisation est de réduire les variantes morphologiques à une forme commune en fonction des règles grammaticales. Elle permet de réduire l'espace des descripteurs en représentant des termes de la même famille par un même descripteur.

Cette technique est plus difficile à mettre en œuvre que le stemming, puisqu'elle nécessite une analyse des règles grammaticales. Elle risque également de générer des erreurs et une perte d'information. Par exemple, le mot « froid » n'a pas le même sens dans les phrases suivantes : « il fait froid » et « cet homme est froid ». Les mots « analogue » et « similaire » font référence au même concept mais seront considérés comme deux racines distinctes.

c) Représentation par Sac de mots

Le modèle sac de mots, « Bag-of-words », est une représentation très utilisée dans la recherche d'information et le TALN (traitement automatique de la langue naturelle). Dans ce modèle, le texte est représenté comme le sac « bag » de ses mots, indépendamment de leur ordre d'apparition dans le texte et de leur rôle grammatical, mais en gardant la fréquence de chaque occurrence. Cependant, les fréquences de termes ne sont pas nécessairement la meilleure représentation du texte. Les mots vides comme « le », « à », « de » sont les termes les plus fréquents dans le texte. Ainsi, avoir une fréquence élevée ne signifie pas forcément que le mot correspondant est pertinent. Pour résoudre ce problème, il est assez courant d'utiliser une liste de rejet « anti-dictionnaires » (les mot vides) de mots à discréditer (les pronoms) étant trop

nombreux dans les corpus textuels pour être informatifs afin d'établir une distinction entre les documents.

Il est généralement nécessaire de prétraiter les mots du vecteur afin de les standardiser (un prétraitement semblable est également appliqué sur les documents à classifier). Les normalisations classiques les plus utilisées sont la lemmatisation et le stemming.

d) Représentation par n-grammes

Un modèle n-grammes est un type de modèle de langage probabiliste qui permet de prédire l'élément suivant dans une séquence sous la forme d'un modèle de Markov d'ordre n. Dans ce modèle, seules les dernières observations sont utilisées pour la prédiction de l'élément suivant. Dans le traitement du langage naturel, les modèles n-grammes désignent généralement des séquences de mots et non de lettres. Ce modèle offre deux avantages qui sont la simplicité et l'évolutivité : avec n plus grand, un modèle peut stocker plus de contexte avec un compromis spatio-temporel, permettant ainsi aux petites expériences d'évoluer efficacement. Cependant, le problème qui se pose et que certaines séquences peuvent ne pas apparaitre dans le corpus d'apprentissage, mais risquent d'apparaître à l'utilisation.

3.2.2. Codage des termes

L'étape qui suit la détermination des attributs est la pondération des descripteurs. Généralement, les vecteurs sont composés par le nombre d'occurrences du terme dans le corpus. Cette information basique doit être pondérée en fonction de plusieurs paramètres liés aux messages, notamment, le nombre de termes dans le message ou dans l'intégralité du corpus.

Autrement dit, étant donné une collection de documents D, soit $E = \{t_1, t_2, \ldots, t_{|E|}\}$ l'ensemble des termes distinctifs de la collection, où t_i est un terme. L'ensemble E est généralement appelé le vocabulaire de la collection, et $|E|$ le nombre de termes dans E (la taille de E). Un document d est décrit par un ensemble de termes distinctifs où chaque terme est associé à un poids (pondération). Un poids p_{ij} est associé à chaque terme t_i d'un document $d_j \in D$. Pour un terme qui n'apparaît pas dans le document $p_{ij} = 0$. Ainsi, chaque document d_j est représenté avec un vecteur de termes, $d_j = (p_{1j}, p_{2j}, \ldots, p_{|E|j})$. Avec cette représentation vectorielle, les documents sont simplement représentés comme une matrice où chaque terme est un attribut et chaque poids est une valeur d'attribut.

Les systèmes utilisant une telle approche calculent p_{ij} en se basant sur l'une des deux hypothèses suivantes :

- Plus le nombre d'occurrences d'un terme dans le corpus est important, moins il est discriminant pour le texte.
- Plus le nombre d'occurrence d'un terme est important, plus il est discriminant pour la classe associée.

a) Le codage booléen

Le codage booléen est l'un des codages les plus anciens et les plus simples. Il utilise la notion de correspondance exacte pour faire correspondre les documents à la requête de l'utilisateur (classification de l'opinion dans notre cas). La requête et la récupération sont toutes deux basées sur l'algèbre de Boole.

Dans le codage booléen, les documents sont représentés comme des ensembles de termes où chacun ne peut posséder qu'un seul statut : présent ou absent dans un document. En utilisant la représentation vectorielle du document ci-dessus, le poids $p_{ij} \in \{0, 1\}$ du terme t_i dans le document d_j est 1 si t_i apparaît dans le document d_j, ou 0 dans le cas inverse.

$$p_{ij} = \begin{cases} 1 & \text{si } t_i \text{ apparaît dans } d_j \\ 0 & \text{sinon} \end{cases} \quad (1)$$

Pour réponde à la requête de l'utilisateur, le système identifie tous les documents qui rendent la requête logiquement vraie. Ainsi, l'identification est basée sur le critère de décision binaire, donc, un document est soit pertinent soit pas. C'est l'un des inconvénients majeurs du modèle booléen, qui conduit souvent à des résultats d'identification faibles. Par conséquence, le codage booléen est rarement utilisé seul dans l'opinion mining.

b) Le codage Fréquence du terme (Term Frequency TF)

Un document dans le codage TF est représenté comme un vecteur de pondération, dans lequel chaque poids de composant est calculé sur la base du schéma Fréquence du terme (TF) tf_{ij}. Le poids p_{ij} du terme t_i dans le document d_j est le nombre d'occurrence de t_i dans d_j, noté f_{ij}. La normalisation pourrait également être appliquée par l'équation suivante, où le maximum est calculé sur tous les termes qui apparaissent dans le document dj :

$$tf_{ij} = \frac{f_{ij}}{\text{Max}\{f_{1j}, f_{2j}, \ldots, f_{|E|j}\}} \quad (2)$$

L'inconvénient des systèmes utilisant le codage de TF est qu'ils ne considèrent pas la situation où un terme apparaît dans de nombreux documents. Un tel terme pourrait ne pas être discriminant.

c) Le codage TF-IDF

TF-IDF : de l'anglais « Term Frequency Inverse Document Frequency » permet de montrer la façon dont les mots sont distribués dans un corpus. Il existe plusieurs variantes de ce schéma. Ici, nous nous contentons de donner le plus basique.

Soit L le nombre total de documents dans le système et df_i le nombre de documents dans lesquels le terme t_i apparaît au moins une fois. Soit f_{ij} le nombre d'occurrence du terme t_i dans le document d_j. La fréquence normalisée (tf_{ij}) du terme t_i dans un document d_j est calculé par l'équation 2 (ci-dessus).

La fréquence du document inverse (idf_i) du terme t_i est calculée par :

$$idf_i = \log \frac{L}{df_i} \qquad (3)$$

Dans le codage TF-IDF, le terme qui apparaît dans un grand nombre de documents dans la collection n'est probablement pas discriminatoire. Le poids final du terme TF-IDF est calculé par :

$$p_{ij} = tfij * idfi \qquad (4)$$

En plus de sa simplicité, la fonction TF-IDF a prouvé son efficacité dans des tâches de classification de textes (Saldarriaga 2010).

Pour classifier les opinions, les algorithmes de classification de l'apprentissage automatique se basent sur les descripteurs et un codage des termes. Nous allons présenter les différents algorithmes d'apprentissage automatique dans la sous-section suivante.

3.2.3. Algorithmes de classification

Il existe de nombreuses méthodes dans l'apprentissage automatique pouvant être explorées. Nous nous limiterons dans ce document aux principales méthodes de classification de sentiments qui sont basées sur les algorithmes suivants :

a) Les réseaux de neurones artificiels

Cet algorithme fut très utilisé dans les années 80-90. Il est inspiré à l'origine du fonctionnement du cerveau humain. En effet. Il est fondé sur l'utilisation de neurones « artificiels » qui effectuent la tâche d'apprentissage. Un neurone artificiel est défini comme étant une fonction algébrique non linéaire, paramétrée, à valeur bornées (Dreyfusetall, 2002).

En général, un réseau neuronal artificiel peut être divisé en trois parties, appelées couches :

- Couche d'entrée : Cette couche est chargée de recevoir les données provenant de l'environnement externe. Ces entrées (échantillons) sont généralement normalisées par les fonctions d'activation. Cette normalisation entraîne une meilleure précision numérique pour les opérations mathématiques effectuées par l'algorithme comme évoqué dans les deux sous-sections précédentes.
- Couches cachés : Ces couches sont composées de neurones qui sont responsables de l'extraction de motifs associés au système analysé. Ces couches exécutent la majeure partie du traitement interne.
- Couche de sortie : Cette couche est également composée de neurones. Elle est responsable de la production et de la présentation des sorties du réseau final (classification de l'opinion dans notre cas) qui résultent du traitement effectué par les neurones dans les couches précédentes.

Les architectures principales des réseaux de neurones artificiels furent dans un premier temps réseau de neurones « monocouche », et devinrent ensuite, « multicouches ». Comme illustré dans la figure 10, le réseau neuronal artificiel monocouche n'a qu'une seule couche d'entrée et une seule couche neurale, qui est également la couche de sortie. Il est composé de n entrées et de m sorties. L'information circule toujours dans une seule direction (donc unidirectionnelle), qui va de la couche d'entrée à la couche de sortie.

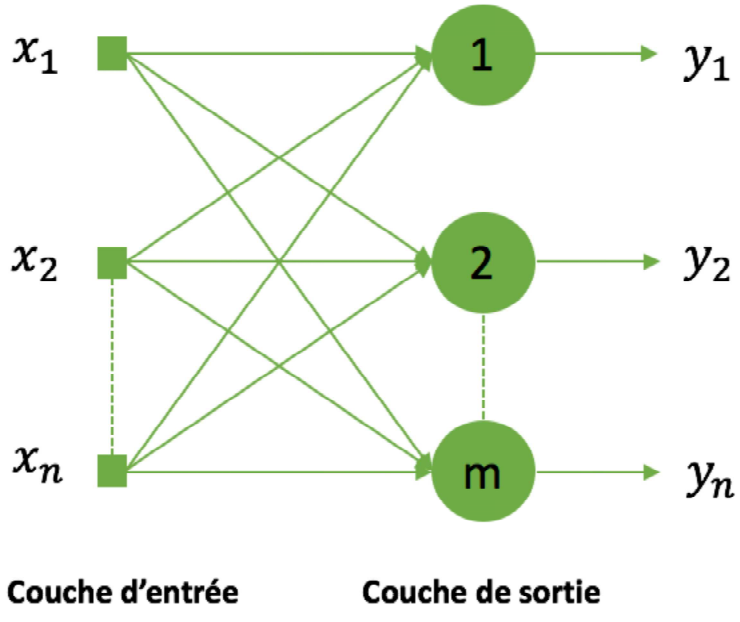


Figure 10: Exemple de Naïve Bayes monocouche

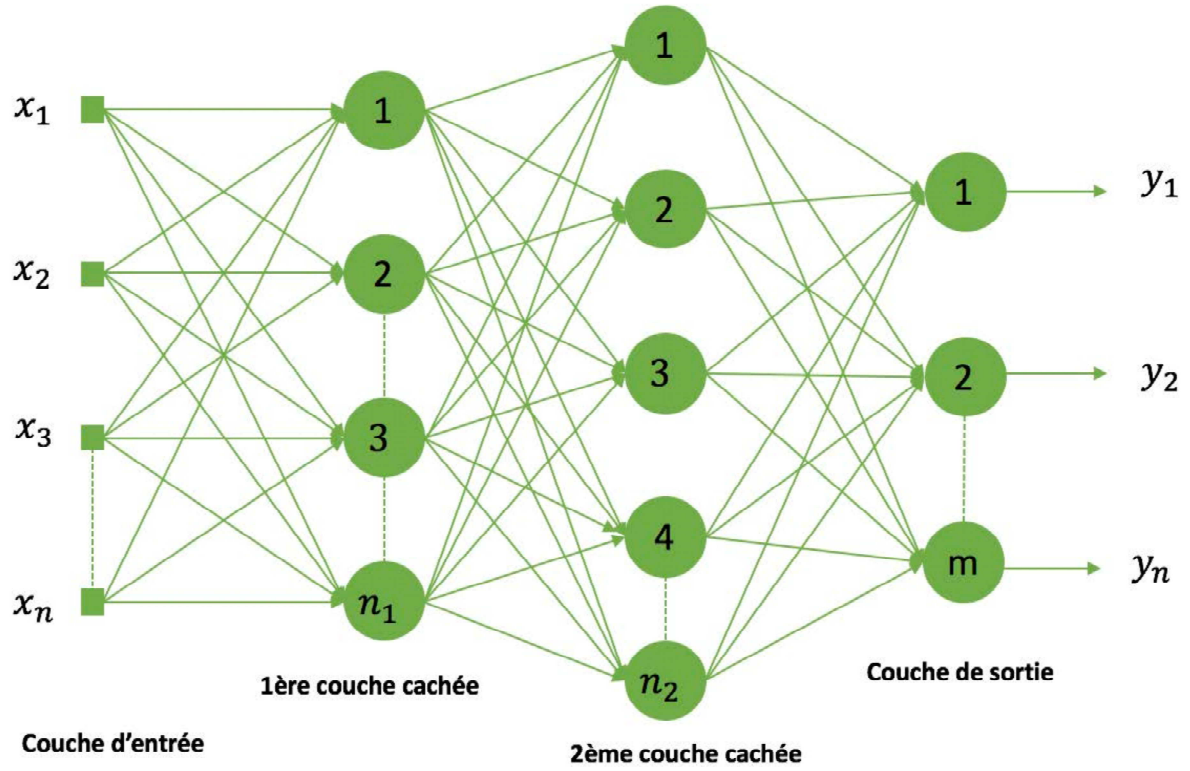


Figure 11: Exemple de Naïve Bayes multicouches

Comme illustré dans la figure 11, la quantité de neurones composant la première couche cachée est différente du nombre de signaux composant la couche d'entrée du réseau. En effet, le nombre de couches cachées et leur quantité respective de neurones dépendent de la nature et de la complexité du problème posé par le réseau, ainsi que de la quantité et de la qualité des données disponibles sur le problème.

Une couche n est composée d'un certain nombre de neurones artificiels dont les entrées sont les sorties des neurones de la couche n-1. La connexion neuronale est effectuée par le biais de liaisons pondérées unidirectionnelles. Ainsi, nous pouvons interpréter un réseau de neurones comme un graphe orienté dont les nœuds sont les neurones artificiels.

b) Les machines à vecteurs de support

Les machines à vecteurs de support ou séparateurs à vaste marge (en anglais support vector machine, SVM) sont des modèles d'apprentissage supervisé avec des algorithmes d'apprentissage associés pour classer des documents, c'est-à-dire à créer un modèle de mappages entrée-sortie. En général, le SVM est un système d'apprentissage linéaire (utilisant la combinaison linéaire de caractéristiques) qui construit des classifieurs à deux classes. La discrimination positive/négative est basée sur le signe de cette combinaison linéaire.

Supposons que nous disposons des exemples d'apprentissage $A = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ où $x_i = (x_{i1}, x_{i2}, \ldots, x_{ik})$ est un vecteur d'entrée de dimension k et les valeurs cibles yi $\{-1,1\}$ (les valeurs 1 et -1 désignent la classe positive la classe négative, respectivement). Le SVM consiste à définir un hyperplan de séparation (une fonction f(x) de forme linéaire), qui sépare les exemples positifs et négatifs les uns des autres avec une marge maximale (voir Figure 12). La marge est la distance entre l'hyperplan et les échantillons les plus proches.

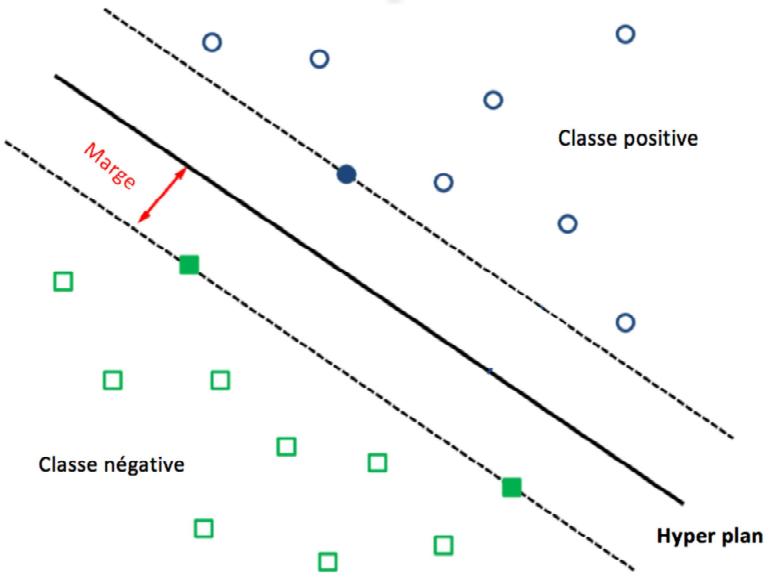


Figure 12 Classification SVM

La fonction f (x) est définie sous la forme suivante :

$$f(x) = < w\ x > + b \qquad (5)$$

$< w\ x >$ est le produit scalaire euclidien de x et w (vecteur de poids). $b \in \mathcal{R}$ et est appelé biais.

Un vecteur d'entrée x_i est assigné à la classe y_i comme suit :

$$y_i = \begin{cases} 1 & si\ \ f(x_i) \geq 0 \\ -1 & si\ \ f(x_i) = 0 \end{cases} \qquad (6)$$

c) K plus proches voisins – k-NN

Toutes les méthodes d'apprentissage précédentes apprennent certains types de modèles à partir des données d'apprentissage (eager learning). En revanche, la méthode des k plus proches voisins ou The k-NN classification (k-Nearest Neighbors) est une méthode d'apprentissage paresseux (lazy learns) dans le sens où aucun modèle n'est appris à partir des données d'apprentissage. En effet, la phase d'apprentissage consiste seulement à mémoriser des exemples dans une forme

optimale de façon à pouvoir les extraire rapidement par la suite. Les calculs quant à eux sont faits dans la phase de classification.

Soit à nouveau A l'ensemble de données d'apprentissage dont les catégories sont connues. Lorsqu'un nouveau texte t à classifier est présenté, l'algorithme calcule la distance ou la similarité avec chaque élément de l'ensemble A. Ensuite, les k exemples les plus proches, appelés les k plus proches voisins de t, seront sélectionnés. t prendra alors la classe la plus fréquente dans les k plus proches voisins.

L'élément clé d'un algorithme k-NN est la fonction de distance / similarité, qui est choisie en fonction des applications et de la nature des données. Pour les documents textuels, la similarité des cosinus est couramment utilisée. Le nombre des k plus proches voisins est généralement déterminé en utilisant la validation croisée sur les données d'apprentissage. C'est-à-dire qu'une gamme de valeurs k est essayée, et la valeur k qui donne la meilleure précision sera sélectionnée. Généralement, k = 1 n'est pas suffisant pour déterminer la classe de t en raison des valeurs aberrantes dans les données.

Le principe de k-NN est extrêmement simple et pourtant très efficace pour la classification de textes (Yang and Liu 1999). En revanche, il est couteux en temps. Etant donné qu'il n'y a pas de construction de modèle, chaque instance de test est comparée à chaque élément de l'ensemble d'apprentissage au moment de la classification, ce qui peut prendre beaucoup de temps surtout lorsque le volume des données est important.

d) Arbres de décision

Les arbres de décision sont très populaires pour générer des règles de classification et de prédiction. Ce sont des outils d'aide à la décision qui représentent un ensemble de choix sous la forme graphique d'un arbre. Les décisions et leurs conséquences possibles, y compris les résultats d'événements aléatoires, sont situées aux extrémités des branches (les « feuilles » de l'arbre), et les chemins de la racine à la feuille représentent les règles de classification. Le processus de décision s'arrête aux feuilles de l'arbre. Un arbre de décision est une structure semblable à un organigramme et peut être converti en un ensemble de règles de contrôle conditionnelles (si-alors). Lorsque la variable à prédire est catégorielle et que ses valeurs représentent donc des classes, nous parlons d'arbre de classification.

Un arbre de décision est composé de trois types de nœuds : des nœuds de hasard, des nœuds de décision et des nœuds terminaux. Un nœud de hasard, illustre plusieurs résultats incertains. Quant au nœud de décision, il représente une décision à prendre, et un nœud terminal représente

le résultat final d'un chemin de décision.

Les arbres de décision sont des outils très efficaces, simples à comprendre et à interpréter. Cependant, les calculs peuvent devenir très complexes, en particulier si de nombreuses valeurs sont incertaines et / ou si de nombreux résultats sont liés.

e) Naïve Bayes

L'apprentissage automatique peut être étudié d'un point de vue probabiliste. En effet, la classification peut être estimée à partir des probabilités postérieures de la classe dans l'exemple de test t : $P(C = c_j|d)$. La classe c_k avec la probabilité la plus élevée est attribuée à t.

Soit $E_1, E_2, \dots, E_{|E|}$ l'ensemble des éléments dans l'ensemble de données A. Et C l'attribut de classe avec $|C|$ valeurs $c_1, c_2, \dots, c_{|C|}$. L'exemple de test t est représenté avec les valeurs d'attributs observées e_1 à $e_{|E|}$. C'est à dire $t =< E_1 = e_1, \dots, e_{|E|} = e_{|E|} >$.

La prédiction est la classe cj telle que $\Pr(C = c_k | E_1 = e_1, \dots, E_{|E|} = e_{|E|})$ est maximale. c_k est appelée une hypothèse maximale a posteriori « maximum a posteriori hypothesis» (MAP).

Selon la règle de Bayes :

$$\Pr(C = c_k | E_1 = e_1, \dots, E_{|E|} = e_{|E|}) = \frac{\Pr(E_1=e_1,\dots,E_{|E|} = e_{|E|}|C=c_j)\Pr(C=c_j)}{\sum_{k=1}^{|C|}\Pr(E_1=e_1,\dots,E_{|E|} = e_{|E|}|C=c_j)\Pr(C=c_k)} \quad (7)$$

$\Pr(C = c_j)$ est la probabilité a priori de classe de c_j, qui peut être estimée à partir des données d'apprentissage.

$\Pr(E_1 = e_1, \dots, E_{|E|} = e_{|E|})$ n'est pas pertinent pour la tâche de classification parce que c'est le même pour chaque classe. Ainsi, seul $\Pr(E_1 = e_1, \dots, E_{|E|} = e_{|E|}|C = c_j)\Pr(C = c_j)$ doit être calculé.

De manière récursive, $\Pr(E_2 = e_2, \dots, E_{|E|} = e_{|E|}|C = c_j)$ est écrit de la même façon, c'est à dire $\Pr(E_2 = e_2, E_3 = e_3, \dots, E_{|E|} = e_{|E|}|C = c_j) \times \Pr(E_3 = e_3, \dots, E_{|E|} = e_{|E|}|C = c_j)$, et ainsi de suite. Cependant, pour poursuivre cette dérivation, nous devons faire l'hypothèse d'indépendance conditionnelle. En effet, les classificateurs bayésiens naïfs supposent que les caractéristiques dans un ensemble de données sont mutuellement indépendantes, d'où vient l'adjectif naïf. Nous supposons donc que $\Pr(E_1 = e_1, \dots, E_{|E|} = e_{|E|}|C = c_j) =$ $\Pr\Pr(E_1 = e_1|C = c_j)$ et de même pou $E_2, \dots, E_{|E|}$. $|E|$

Au final, nous obtiendrons la formule suivante :

$$\Pr(E_1 = e_1, \dots, E_{|E|} = e_{|E|}|C = c_j) = \prod_{i=1}^{|E|}\Pr(E_i = e_i \mid C = c_j) \quad (8)$$

$$\text{Et } \Pr\left(C = c_j \middle| E_1 = e_1, \dots, E_{|E|} = e_{|E|}\right) = \frac{\Pr\left(C = c_j\right) \prod_{i=1}^{|E|} \Pr(E_i = e_i \middle| C = c_j)}{\sum_{k=1}^{|C|} \Pr(C = c_k) \prod_{i=1}^{|E|} \Pr(E_i = e_i | C = c_j)} \quad (9)$$

Les probabilités a priori $\Pr(C = c_j)$ et les probabilités conditionnelles $\Pr\left(E_i = e_i \middle| C = c_j\right)$ sont calculées à partir des données d'apprentissage comme suit :

$$\Pr(C = c_j) = \frac{\text{nombre d'exemples de classe } c_j}{\text{nombre total des exemples d'apprentissage}} \quad (10)$$

$$\Pr\left(E_i = e_i \middle| C = c_j\right) = \frac{\text{nombre des exemple avec } E_i = e_i \text{ et classe } c_j}{\text{nomre des exemples de la classe } c_j} \quad (11)$$

La classe la plus probable pour un cas de test t est défini comme suit :

$$c = \arg_{cj} \max \Pr(C = c_j) \prod_{i=1}^{|E|} \Pr(E_i = e_i | C = c_j) \quad (12)$$

Malgré que l'hypothèse d'indépendance soit souvent violée, les classificateurs Naïve bayes performent dans cette hypothèse irréaliste, surtout pour les échantillons de petite taille (Rish 2001) (Domingos and Pazzani 1997). Ils sont très utilisés grâce à la facilité d'estimation des paramètres et sa rapidité (même sur un volume important de données), la simplicité d'implémentation.

Naïve Bayes est l'un des classificateurs les plus utilisés dans l'analyse des sentiments. Dans le but de traiter le problème du transfert de domaine, commun aux classificateurs de sentiment supervisés, (Ravi 2015) recommande Adapted Naïve Bayes. Dans cette étude, ils ont utilisé les données de l'ancien domaine et les données du nouveau domaine étiqueté, suggérant une mesure efficace pour contrôler les informations provenant des données de l'ancien domaine. Le résultat de leur expérience montre que la méthode recommandée peut améliorer significativement les performances du classificateur de base.

Dans le domaine de l'analyse d'opinion, une étude comparative (Gonçalves et al. 2013) surligne le besoin de combiner plusieurs méthodes parmi celles présentées ci-dessus pour exploiter les avantages de chaque algorithme de classification.

3.2.4. L'évaluation du « classifieur »

Comme évoqué plus haut, une fois qu'un « classificateur » est choisi et construit, nous devons l'évaluer pour mesurer sa performance et son exactitude. C'est une étape importante pour tout projet de classification. Il existe plusieurs manières et mesures pour évaluer un « classifieur » mais la précision de la classification reste la principale mesure. Cette mesure représente le

nombre de documents dans l'ensemble de test correctement classés, divisé par le nombre total de document de l'ensemble de test.

Dans ce qui suit, nous présentons d'autres métriques et méthodes couramment utilisées pour l'évaluation des « classifieurs ».

a) Validation croisée

Cette méthode est utilisée particulièrement lorsque l'ensemble de données est petit. L'objectif de la validation croisée est de définir un ensemble de données pour « tester » le modèle en phase d'apprentissage. Dans cette méthode d'évaluation, les données disponibles sont partitionnées en n sous-ensembles disjoints de taille égale. Chaque sous-ensemble est ensuite utilisé comme ensemble de test et les sous-ensembles n-1 restants sont combinés en tant qu'ensemble d'apprentissage. Cette procédure est ensuite exécutée n fois, ce qui donne n précisions. La précision finale estimée de l'apprentissage à partir de cet ensemble de données est la moyenne des n précisions. En général, les validations croisées 10 et 5 sont les plus utilisées. La validation croisée peut également être utilisée pour l'estimation des paramètres.

b) La précision et le rappel

La précision et le rappel sont deux critères de mesures statistiques évaluant les « classifieurs », aussi appelés valeur prédictive et sensibilité.

Nous notons :

VP : le nombre d'éléments correctement étiquetés positifs (vrai positif)

FN : le nombre de classifications incorrectes d'exemples positifs (faux négatifs)

FP : le nombre d'éléments qui ont été incorrectement étiquetés positifs (faux positifs)

TN : le nombre de classifications correctes d'exemples négatifs (vrai négatif)

Dans une tâche de classification d'opinion, la précision p d'une classe est le nombre de vrais positifs divisé par le nombre total d'éléments catégorisés positifs :

$$p = \frac{VP}{VP+FP} \qquad (13)$$

le Rappel r dans ce contexte est défini comme le nombre de vrais positifs divisé par le nombre total d'éléments qui appartiennent réellement à la classe positive.

$$r = \frac{VP}{VP+FN} \qquad (14)$$

Un score de précision de 1 pour une classe C signifie que chaque élément associé à la classe C appartient réellement à cette classe. Cependant, ce score ne dit rien sur le nombre d'éléments de la classe C qui n'ont pas été correctement étiquetés). Un score de rappel signifie que chaque élément appartenant à la classe C a été correctement étiqueté (mais ce score ne dit rien du nombre des éléments qui ont été incorrectement associés à la classe C). Il existe une relation inverse entre la précision et le rappel, où il est possible d'augmenter l'un au détriment de l'autre.

c) F-score

F-score, est une mesure populaire de la performance d'un test qui combine à la fois la précision et le rappel. Elle est souvent utilisée pour comparer différents classificateurs avec une seule mesure.

$$F = 2 \times \frac{p \times r}{p+r} \quad (15)$$

F-score est également appelé F1-score ou F-mesure, est la moyenne harmonique pondérée de la précision et du rappel :

$$F = \frac{2}{\frac{1}{p}+\frac{1}{r}} \quad (16)$$

d) La fonction d'efficacité du récepteur

La fonction d'efficacité du récepteur, plus fréquemment dite courbe ROC ou courbe sensibilité/spécificité, est une mesure de performance des classifieurs binaires (les systèmes qui catégorisent les éléments en deux classes distinctes). Graphiquement, elle est représentée par le taux de vrai positifs par rapport au taux de faux positifs.

Le taux de vrai positifs TVP désigne la fraction des positifs qui sont réellement détectés et le taux de faux positifs TFP désigne la fraction des négatifs qui sont incorrectement détectés.

$$TVP = \frac{VP}{VP+FN} \quad (17)$$

$$TFP = \frac{FP}{VN+FP} \quad (18)$$

La TVP est le rappel de la classe positive et est également appelée sensibilité. Il y a aussi une autre mesure, appelée spécificité, qui représente le taux de vrai négatif TVN, ou le rappel de la classe négative. TVN est défini comme suit :

$$TVN = \frac{VN}{VN+FP} \quad (19)$$

Un espace ROC est défini par TVP et TFP comme axes x et y, respectivement, qui représentent des compromis relatifs entre vrai positif et faux positif. Chaque résultat de prédiction on représente un point dans l'espace ROC.

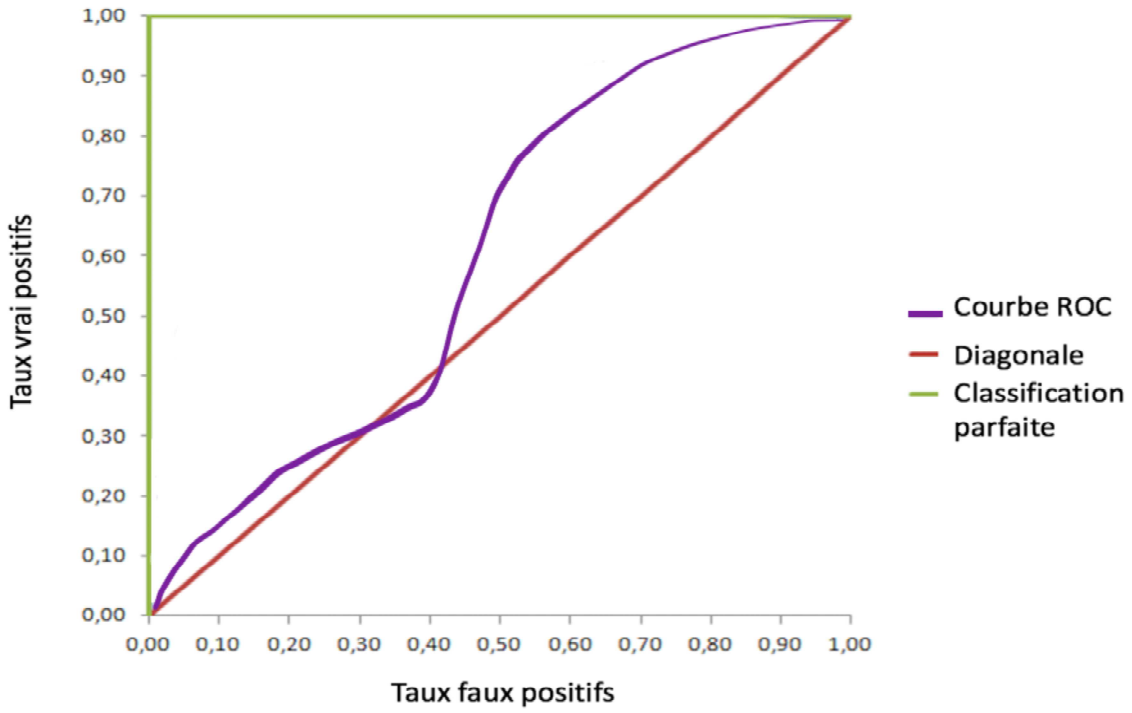


Figure 13 Courbes ROC

Comme illustré dans la figure 13, La diagonale divise l'espace ROC en deux. Les points au-dessus de la diagonale représentent de bons résultats de classification et les points au-dessous de la ligne représentent des résultats médiocres. Une classification parfaite donnerait la coordonnée (0,1) dans l'espace ROC, représentant 100% de sensibilité (aucun faux négatif) et 100% de spécificité (aucun faux positif).

4. Application

Ci-dessous, des exemples qui illustrent l'importance de l'analyse des sentiments et d'opinions dans trois axes de notre vie quotidienne :

a) Amélioration des conditions de vie des citoyens

Cet axe comprend la santé, la maison et l'éducation. Dans le secteur de la santé, Cambria et al. (2012) ont conçu un Framework qui mesure la qualité des soins de santé tout en suivant la sensibilité physio-émotionnelle des patients. Ce Framework exploite l'application des PROM

standard (mesures standards des résultats rapportés par les patients) et l'analyse des sentiments à travers un questionnaire et un texte non structuré. Kashyap et Nahapetian (2014) ont effectué une analyse sémantique et sentimentale sur les tweets de l'utilisateur afin de déterminer son état de santé. Ils ont fourni le score moyen de santé de l'utilisateur à travers différentes catégories tels que les médecins, les personnes commentent un contenu de restauration rapide, et les célébrités. Rodrigues et al. (2016) ont mis au point une nouvelle méthode, appelée SentiHealth, qui détecte l'humeur des patients atteints de cancer en se basant sur les données collectées auprès des communautés cancéreuses sur Facebook. Cette méthode permet aux professionnels de la santé de surveiller l'état émotionnel des patients atteints de cancer. Harous et al. (2018) ont développé des algorithmes d'exploration de données et d'analyse de sentiment afin d'analyser des données collectées à partir de capteurs non invasifs et de Twitter. Le but de ce système est de générer des suggestions intelligentes, des mises en garde et des recommandations pour contrôler et réduire le risque de surpoids et ses maladies connexes. Lim, Tucker, and Kumara (2017) ont proposé un modèle d'apprentissage automatique non supervisé pour l'analyse de sentiment afin de prévoir et de suivre les maladies infectieuses latentes dans un endroit donné. Ce modèle permet également d'identifier les expressions des utilisateurs sur les symptômes, les parties du corps et les localisations de la douleur. Ce modèle utilise les données des médias sociaux ainsi que l'analyse des sentiments et l'extraction des symptômes des utilisateurs, de la partie du corps et de la localisation de la douleur. Dans le même contexte, Jain and Kumar (2017) ont introduit un modèle de surveillance afin de détecter et prédire les endroits touchés par une épidémie de maladie transmise par les moustiques. Cette étude intègre l'analyse des sentiments et la géolocalisation en utilisant les médias sociaux.

Dans le domaine de l'éducation, Altrabsheh, Gaber, and Cocea (2013) ont développé un système d'analyse de sentiment pour l'éducation (SA-E) qui mesure le succès des techniques d'enseignement en temps réel en s'appuyant sur les sentiments des étudiants. Ce Framework permet d'identifier les zones problématiques et de prendre des mesures correctives. Elyasir and Anbananthen (2013) ont mis au point un Framework qui compare les universités en fonction des critères choisis par l'utilisateur. Pour cela, ils ont extrait des données issues de plusieurs sources, telles que Twitter et Facebook, et ont catégorisé la polarité en classes positives, négatives ou neutres pour chaque université.

b) L'administration publique et services

Cet axe comprend la supervision de la sécurité publique, le trafic et le tourisme intelligent. Dans le domaine du trafic intelligent, Kim, Cha and Sandholm (2014) ont introduit un système, appelé

SocRoutes, qui vise à trouver l'itinéraire le plus sûr et le plus agréable basé sur des sentiments déduits de tweets géo-localisés en temps réel. En outre, les chercheurs ont montré la corrélation entre les sentiments et la criminalité dans un cas d'utilisation avec des données historiques de la ville de Chicago. Salas et al. (2017) ont extrait la polarité des tweets liés à la circulation en utilisant l'analyse des sentiments sur les tweets afin de détecter le événements de circulation et le niveau de stress des utilisateurs. Anastasia and Budi (2016) ont comparé les fournisseurs de services de transport en ligne en mesurant la satisfaction de la clientèle grâce à l'analyse du sentiment des tweets.

Dans le tourisme intelligent, Shimada et al. (2011) ont proposé un système d'analyse de l'information touristique pour métropoles. Le système analyse la polarité des tweets liés aux lieux cibles et aux événements touristiques afin d'obtenir des indications utiles pour le développement touristique. De même, Kuhamanee et al. (2017) ont suggéré une analyse des sentiments des touristes étrangers à Bangkok, en Thaïlande, via Twitter. Les chercheurs ont classé les tweets dans cinq contextes de séjour à Bangkok (voyages, affaires, visites familiales, éducation, santé et traitements) et ont extrait le sentiment des touristes pour chaque classe afin de mesurer le degré de satisfaction par rapport à leurs attentes initiales. (K. Kim et al. 2017) ont appliqué l'analyse de sentiments pour extraire la perception des visiteurs par rapport aux offres touristiques dans la ville de Paris : panorama, restaurants, visites touristiques, hôtels, choses à faire, vie nocturne, transports, shopping, sport et plein air, favoris, pièges à touristes, avertissements et dangers, et coutumes locales. A partir de ces analyses, ils ont réussi à identifier les motifs de non satisfaction des touristes.

Dans un souci d'évaluer la qualité de services dans les administrations publiques, l'analyse des sentiments pourrait également aider les services de ce secteur à gérer les crises et à fournir des services adaptés aux besoins des citoyens (Riel, Popescu, and Guanlao 2014) ; Gaspar et al. 2016). Par exemple, Corallo et al. (2015) visent à fournir le sentiment des citoyens sur les institutions, l'efficacité des services et des infrastructures, le degré de satisfaction des événements de l'administration publique, afin de soutenir les décisions de l'administration publique Musto et al. (2015) ont conçu un Framework pour le traitement du sentiment des flux textuels provenant des réseaux sociaux. Ils l'ont déployé sur deux scénarios liés aux villes intelligentes : le premier pour surveiller l'état de récupération de paramètres sociaux (sentiment d'appartenance, confiance et sécurité…) de la ville de L'Aquila après un terrible tremblement de terre et le second pour dresser une carte des zones les plus dangereuses du territoire italien en termes d'homophobie, racisme, violence contre les femmes, antisémitisme…

c) La gestion et la surveillance des ressources

L'analyse de sentiments a également été utilisée à des fins écologique : gestion et surveillance des ressources, notamment l'eau, l'électricité, l'agriculture et l'air. Ainsi, Spiliotis et al. (2015) ont introduit une approche pour évaluer le confort thermique des bâtiments municipaux en se basant sur le sentiment issu de l'analyse des médias sociaux. Le but de cette étude est d'évaluer et d'ajuster les plans d'action de gestion de l'énergie. Manologlou, Tsartas, and Markou (2004) ont examiné les points de vue des résidents de l'île de Milos (Grèce), qui souffre d'un énorme manque d'eau, afin d'identifier la situation concernant l'approvisionnement en eau et l'impact d'une construction d'usine de dessalement à proximité. A partir de l'analyse de sentiments déduits des réseaux sociaux par rapport à la qualité de paramètres environnementaux (exemple : eau et air), Wang et al. (2017) ont pu prédire la qualité de l'environnement dans une région donnée. Le résultat de la comparaison de la qualité de l'environnement entre les différentes provinces était très proche des informations de classement publiées par le ministère de la Protection de l'environnement. Dans le domaine de l'agriculture, Valsamidis et al. (2013) ont appliqué l'analyse des sentiments sur les blogs afin d'enquêter sur des sujets agricoles. Le Framework proposé pourrait contribuer à améliorer le traitement des aspects agricoles par les autorités. Dans le même contexte, Ragkos, Theodoridis, and Batzios (2015) ont analysé les sentiments et les préférences du public à l'égard de l'orientation de l'agriculture, y compris la gestion de l'eau d'irrigation. Cette analyse a montré les axes à cibler pour un développement agricole et rural.

5. Conclusion

Un grand nombre d'utilisateurs partage leur opinion sur les réseaux sociaux, ce qui en fait une plate-forme précieuse pour suivre et analyser le sentiment public. Cette analyse peut fournir des informations cruciales pour la prise de décision dans divers domaines tels que la politique, la santé, ou le marketing. Pour cela, nous avons défini dans ce chapitre les techniques d'analyse des données issues des réseaux sociaux (big social data), les différents courants de recherche et disciplines dont elle est issue, cela inclut l'analyse textuelle et l'analyse de sentiments (approches basées sur le lexique et/ou l'apprentissage automatique). Nous avons donc exposé ces approches, et particulièrement les algorithmes d'apprentissage automatique supervisé de référence pour la classification de sentiments, à savoir : Naïve Bayes, Machine à vecteurs de support... Nous avons également présenté en détails les méthodes de représentation de données textuelles d'une part, et les paramètres utilisés pour l'évaluation des méthodes d'analyse de

sentiments d'autre part. Finalement, nous avons présenté plusieurs applications qui s'appuient sur l'analyse de sentiments dans différents domaines tels que la santé, l'éducation, le tourisme et la politique.

Dans le chapitre suivant, nous présentons notre approche pour le traitement des « big social data» ainsi que les expérimentations que nous avons menées afin de mesurer la robustesse et la performance de la méthode de classification proposée.

Chapitre 4 Nouvelle méthodologie d'analyse big social data

1. Introduction

La mesure du sentiment public en temps réel a toujours été une tâche difficile. Les approches traditionnelles consistent à effectuer des sondages auprès d'un nombre significatif de personnes sur leurs attitudes envers un sujet particulier. Cette tâche nécessite beaucoup de temps et d'argent. Comme exposé dans le chapitre 3, le nombre de « sociaunautes » ne cesse d'augmenter ; ainsi, les réseaux sociaux et les blogs sont devenus une source d'information précieuse s'ils sont pleinement exploités. En effet, les big social data sont utilisées dans divers secteurs tels que le marketing, les soins de santé, l'éducation et la finance, afin d'améliorer les activités grâce aux opinions partagées sur les différents réseaux sociaux tels que Facebook, Twitter et autres. Cependant, les techniques de traitement des big social data montrent des degrés de précision très variables, voire contradictoires. En outre, en raison du caractère contextuel de la plupart des méthodes d'analyse des big social data proposées pour l'analyse de sentiments, celles-ci souffrent d'un manque de flexibilité et d'adaptabilité à différents sujets d'analyse.

2. Problématique

L'analyse des big social data est une tâche complexe et est confrontée à plusieurs défis. En effet, il existe des points de vue contradictoires sur la fiabilité de l'analyse des médias sociaux ; alors que certaines études ont démontré une corrélation directe entre les évènements et leurs publications connexes, d'autres, comme Gayo-Avello (2012), Tumasjan (2010) et Delenn Chin, Jessica Zhao and Anna Zappone (2016), ont montré l'inverse. Bien que ces dernières aient été basées sur des méthodes statistiques et / ou sur des dictionnaires référencés, comme présenté dans le chapitre 3, elles ont été vouées à l'échec. En général, ce problème est lié à des défis d'exploration de l'opinion et du sentiment.

Comme évoqué dans le chapitre 3, la tâche de la classification des sentiments implique l'étiquetage du texte avec une classe de sentiment. Il existe deux familles principales d'approches : les approches basées sur l'apprentissage automatique supervisé et les approches basées sur le lexique. Tout naturellement, les méthodes supervisées utilisent des algorithmes d'apprentissage automatique formés avec des données étiquetées (positive, négative ou neutre)

pour prédire la classe de sentiment des nouveaux documents. Cependant, cette approche devient problématique lorsqu'il est difficile d'obtenir des données d'entraînement fiables et suffisantes comme dans le cas des big social data. De même, les données d'apprentissage ont tendance à être très spécifiques à un domaine ou contexte précis, performantes dans le contexte de l'étude mais inadaptées à un domaine différent. Les big social data sont diversifiés et concernent plusieurs domaines tels que la politique, le marketing, ou la santé. Les systèmes d'analyse des textes issus des réseaux sociaux doivent donc maintenir une performance dans des domaines différents. Par conséquent, les systèmes basés sur l'apprentissage automatique deviennent peu adaptés au contexte du big social data.

Quant aux approches basées sur l'analyse lexicale, elles impliquent l'agrégation des scores de polarité à partir de référentiels génériques pour classifier le texte. Ces approches sont plus flexibles et donc plus adaptées pour la classification des sentiments dans le contexte des big social data. Néanmoins, elles rencontrent elles-mêmes des challenges tels que la définition de l'orientation sémantique des mots qui pourrait être fortement influencée par le contexte. Par exemple, utiliser le terme « sexuel » serait considéré comme un mot négatif dans un tweet lié à Trump, qui a été accusé d'agression sexuelle par quinze femmes, et par opposition, sera considéré comme un mot positif si le tweet est lié à Hillary. Un autre exemple, l'orientation sémantique du mot « email », qui est généralement un mot neutre, pourrait changer en négatif dans le contexte de Hillary, qui a enfreint les lois fédérales en utilisant un compte de courriel personnel pour des affaires gouvernementales. Il n'est donc pas surprenant que les approches basées sur l'analyse lexicale ne donnent pas de très bons résultats si elles ne sont pas contextualisées.

En outre, les approches basées sur le lexique et/ou l'apprentissage automatique ne prennent pas en considération l'informalité des messages publiés dans les réseaux sociaux. En effet, ces messages pourraient contenir des mots spéciaux comme ceux écrits en majuscules ou contiennent la répétition de plus de deux lettres consécutives que l'on qualifiera dans la suite du manuscrit par « mot étendu ». Ces mots spéciaux peuvent être utilisés pour pondérer la charge émotionnelle des publications. Par conséquent, ils pourraient être considérés comme des intensificateurs ou diminutifs de polarité.

Dans ce chapitre, nous cherchons à proposer des solutions pour surmonter les difficultés et limitations liées à l'exploration de l'opinion en termes d'orientation sémantique contextuelle et d'adaptabilité. Adoptant une méthodologie basée sur le lexique, nous présentons une nouvelle approche adaptable qui permet d'associer une polarité aux mots en fonction du contexte à travers la construction de dictionnaires dynamiques. Dans le but d'améliorer la finesse de la

classification de sentiments d'un document, nous nous sommes référés aux méthodes classiques recueillies dans la littérature que nous avons enrichie avec de nouvelles métriques.

3. L'approche proposée

Afin de construire un modèle d'analyse de sentiment, nous proposons une méthodologie composée de trois étapes, comme illustré dans la figure 14 : elle consiste premièrement à construire des dictionnaires dynamiques, puis à classifier et équilibrer cet ensemble avant d'exécuter l'algorithme de prédiction. Les différentes étapes sont détaillées dans les sous-sections suivantes.

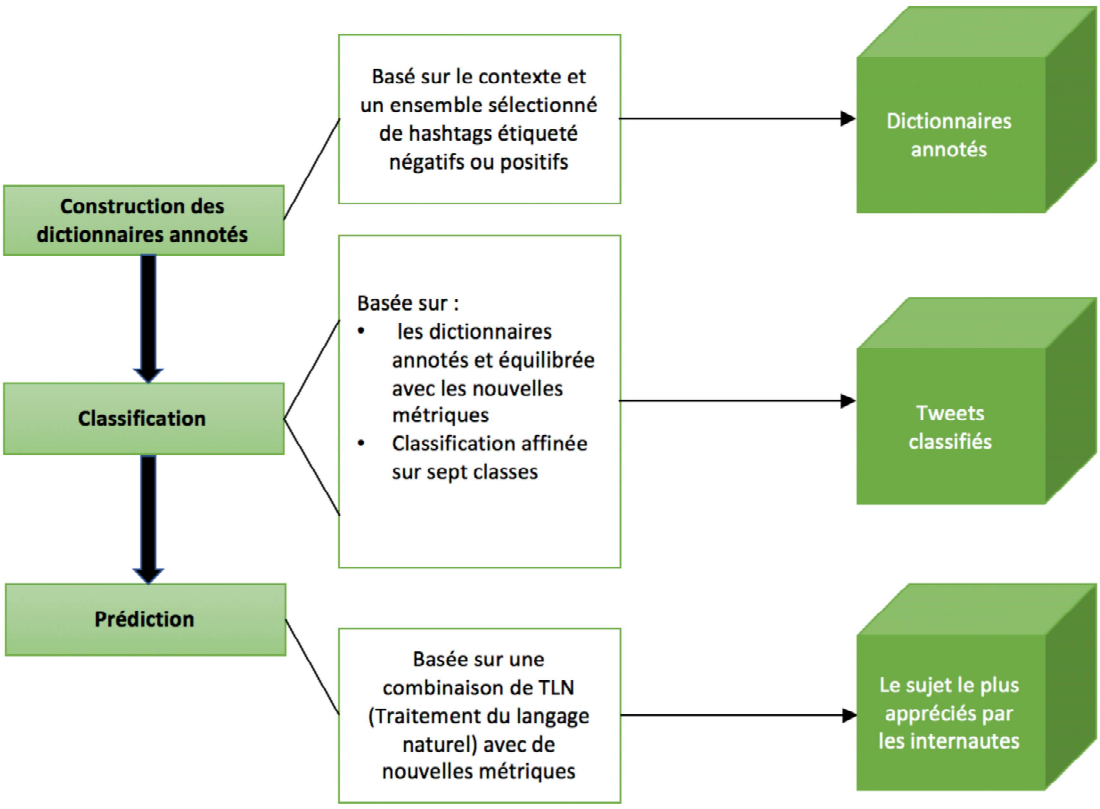


Figure 14: Les étapes de l'approche proposée

Soit : $Y_i, i = 1, \ldots n$ un ensemble de produits, de services ou de personnes que nous souhaitons comparer dans un contexte spécifique.

Considérons $D = \{Y_1, Y_2, \ldots, Y_n\}$ comme contexte ciblé.

Dans ce qui suit, nous allons exposer la procédure adoptée pour construire un dictionnaire contextuel dynamique pour chaque Y_i.

3.1. Première étape : construction des dictionnaires annotés

Diverses recherches dans l'analyse des médias sociaux, telles que celles menées par Speriosu et al. (2011) et Stavrianou et al. (2014), ont déterminé le sentiment d'un poste (message) en se basant sur la polarité des hashtags. Cependant, ils ont utilisé un très grand ensemble de hashtags annotés manuellement (ce qui prend beaucoup de temps) ou ils ont combiné ces derniers avec des dictionnaires afin d'améliorer la précision de la classification des postes. Contrairement à ces travaux, nous utilisons un petit ensemble de hashtags, d'une nouvelle manière, afin de construire des dictionnaires de sentiments, annotés avec l'orientation sémantique du mot pour un contexte donné.

Pour cela, nous supposons que tous les mots d'un tweet contenant un hashtag positif ont une polarité positive et vice versa, puis nous l'affinons par les sous-étapes suivantes :

- **Sous-étape 1** : Le but de cette étape consiste à collecter, stocker et classer les tweets dans une infrastructure big data. Pour cela, nous définissons d'abord les hashtags qui ont une fréquence élevée comme les hashtags les plus populaires pour chaque Y_i. Ensuite, nous classifions manuellement un très petit ensemble d'hashtags populaires (entre deux et trois pour chaque Y_i) dans des classes positives et négatives. Finalement, nous collectons les données de chaque classe séparément. En d'autres termes, les tweets seront classés comme positifs ou négatifs en se basant sur la polarité de l'hashtag. Par exemple, tous les tweets contenant l'hashtag #neverhillary seront évidemment considérés contre Hillary et classés comme négatifs.
- **Sous-étape 2** : le but de cette étape consiste à construire des dictionnaires intermédiaires annotés pour chaque sujet Y_i : $inter - posSW(Y_i)$ et $inter - negSW(Y_i)$
  Les données issues des réseaux sociaux ne sont pas normalisées, d'où le besoin de prétraitement. Pour cela, nous appliquons différents filtres sur les tweets classifiés dans l'étape 1 comme suit :
  - *Suppression de tweets en double* : Comme un tweet pourrait inclure plus d'un hashtag, il pourrait être extrait plusieurs fois. Ainsi, nous supprimons les tweets dupliqués afin d'éviter le surpoids.
  - *Tokenisation :* cette sous-étape consiste à manipuler des noms, des verbes, des adverbes, des adjectifs, des URL, des émoticônes communes, des numéros de téléphone, des balises HTML, des hashtags, la répétition de symboles et les caractères Unicode.

- *Conversion :* Nous convertissons toutes les données en minuscules et remplaçons plus de deux des mêmes lettres consécutives par une seule occurrence de la lettre (par exemple, fuuunny sera remplacé par funny et ANGRY par angry).
- *Stemming :* en éliminant les pluriels, les genres et la conjugaison et en corrigeant les fautes de frappe et les fautes d'orthographe
- *Filtrage :* Différentes recherches prouvent que les adverbes, les adjectifs et les verbes sont de bons indicateurs de sentiment positif et négatif, nous appliquons donc une étape de filtrage afin de garder ces indicateurs. Comme les données sociales peuvent contenir plus d'informations qu'un texte formel, nous améliorons ces indicateurs en ajoutant des indicateurs de sentiment tels que des hashtags.

- **Sous-étape 3 :** Le but de cette étape est d'affiner les dictionnaires intermédiaires annotés afin de construire les dictionnaires de sentiment finaux : positif $posSW()$, négatif $negSW()$ et pour chaque Y_i. Pour cela, nous construisons ces dictionnaire en se basant sur l'occurrence des mots $Occ(w_j)$ dans les différentes classes. L'approche de construction de ces dictionnaires est présentée dans l'algorithme 1. Un exemple d'application est illustré dans la figure 15.

Algorithm 1 Constructing the final dictionaries

Require: $inter\text{-}posSW(Y_i)$,$inter\text{-}negSW(Y_i)$
Ensure: $posSW(Y_i), negSW(Y_i), neutSW(Y_i)$
$posSW(Y_i) = inter\text{-}posSW(Y_i)$
$negSW(Y_i) = inter\text{-}negSW(Y_i)$
for pw in $inter\text{-}posSW(Y_i)$ **do**
  for nw in $inter\text{-}negSW(Y_i)$ **do**
    if $pw == nw$ and $occ(pw) > occ(nw)$ **then**
      $ratio = occ(nw)/occ(pw)$
    else if $pw == nw$ and $occ(nw) > occ(pw)$ **then**
      $ratio = occ(pw)/occ(nw)$
    end if
    if $pw == nw$ and $occ(pw) > occ(nw)$ and $ratio < 0.7$ **then**
      Delete ng from $negSW(Y_i)$
    else if $pw == nw$ and $occ(pw) < occ(nw)$ and $ratio < 0.7$ **then**
      Delete pw from $posSW(Y_i)$
    else if $ratio > 0.7$ **then**
      Add pw in $neut\text{-}SW(Y_i)$
      Delete pw from $posfromSW(Y_i)$
      Delete ng from $negSW(Y_i)$
    end if
  end for
end for

Pour chaqueY_i, l'algorithme cherche les mots qui figurent dans les deux dictionnaires intermédiaires (positif et négatifs). Ensuite, il supprime le mot du dictionnaire qui a la plus petite occurrence. Si le ratio des deux occurrences est supérieur à 0.7 alors le mots est considéré comme neutre et est supprimé des deux dictionnaires.

L'algorithme est répété pour chaque Y_i .

pw et *nw* sont respectivement les mots positifs et négatifs dans les dictionnaires intermédiaires.

Nous avons utilisé un test empirique, qui consiste à tester un nombre de valeurs (entre 0.5 et 0.8), afin de constituer la limite permettant de classer les mots de sentiment avec le plus petit taux d'erreur. Dans notre cas, 0,7 était la meilleure valeur.

Enfin, nous attribuons un score aux mots dans les dictionnaires : 1 pour les mots positifs, et -1 pour les mots négatifs.

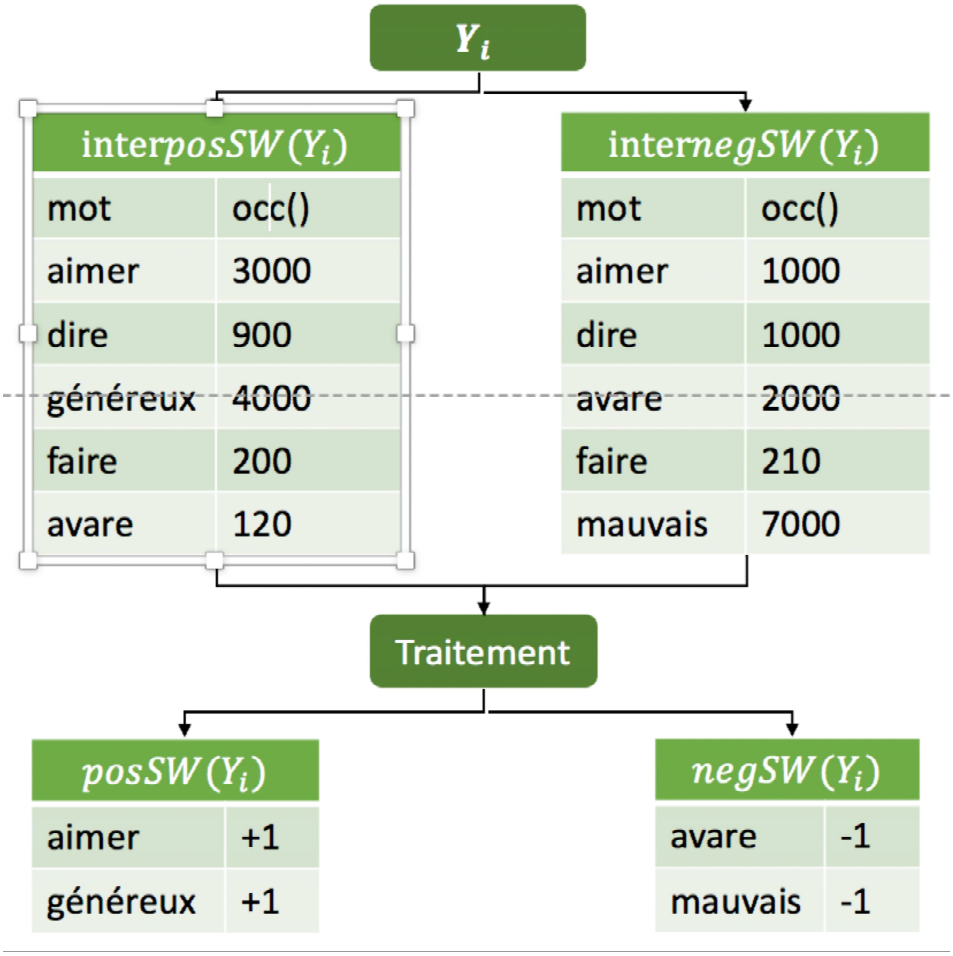


Figure 15: exemple d'application de l'algorithme 1

La figure 16 illustre la première étape de l'approche proposée.

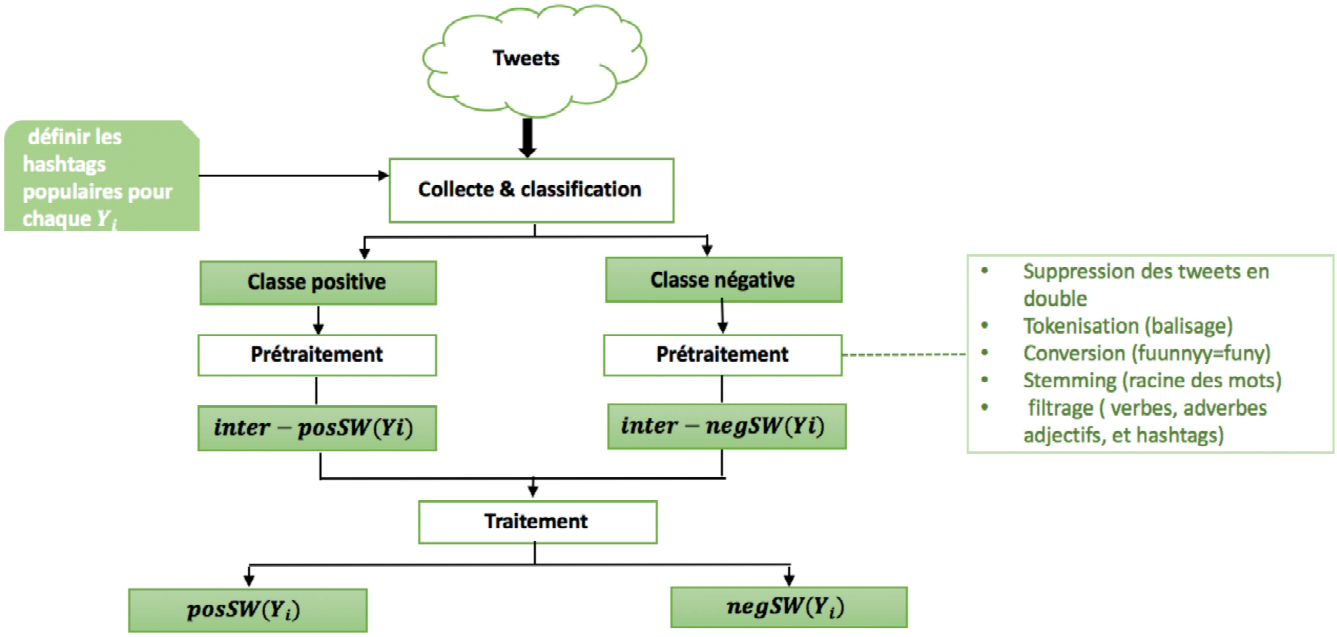


Figure 16: La première étape du modèle

3.2. Deuxième étape : Classification des sentiments

Dans cette étape, nous procédons à la classification des tweets en se basant sur les dictionnaires annotés *SW* construit dans l'étape précédente.

- **Sous-étape 1 :** collecter et stocker de nouveaux tweets pour chaque Yi séparément. Les données collectées passent par les étapes suivantes.
- **Sous-étape 2 :** consiste à prétraiter les données comme suit :
  - *Suppression des données dupliquées :* Comme un message pourrait inclure plusieurs hashtags, il pourrait être extrait plusieurs fois. Pour cela, nous supprimons les tweets dupliqués afin d'éviter de biaiser les résultats de prédiction.
  - *Tokenisation :* la même tokenisation utilisée dans la première étape est appliquée aux nouveaux tweets.
  - *La négation :* Comme les mots de négation (non, ni, ne pas…) affecte significativement la polarité globale d'une phrase, la négation constitue un critère très important dans la classification des sentiments. Comme recommandé dans Jose and Chooralil (2016) et Pang, Lee, and Vaithyanathan (2002), nous inversons la polarité de tous les mots qui viennent après un mot de négation jusqu'à atteindre un signe de ponctuation.

- *Détection des mots spéciaux (intensificateurs et diminutifs)* : Dans cette étape nous détectons les mots écrits en majuscules ou constituant plus de deux des mêmes lettres consécutives. Lorsqu'une séquence de trois lettres ou plus est détectée, le mot étendu, nous réduisons d'abord le nombre des lettres répétées en deux, ensuite, nous vérifions le mot réduit existe dans le dictionnaire. Si le mot réduit n'est pas trouvé, les lettres répétées sont réduites à une lettre. Nous vérifions une séquence à la fois.
- *Stemming :* nous suivons les mêmes règles que la première étape.

- **Sous-étape 3 :** Dans cette étape, nous calculons le degré de polarité des tweets à partir de l'orientation sémantique des dictionnaires construits dans l'étape 1. Pour cela, nous appliquons les deux actions suivantes :
  - *Équilibrage (intensificateurs et diminutifs)* : Il est important de noter que le langage utilisé dans les big social data n'est pas conventionnel et pourrait contenir des mots étendus comme ceux écrits en majuscules ou contiennent la répétition de plus de deux lettres consécutives. Nous pensons que ce type de mots est porteur d'informations alors qu'il n'est pas pleinement exploité dans la plupart des techniques d'analyse sentimentale (Delenn Chin, Jessica Zhao and Anna Zappone 2016 ; Gayo-Avello 2012 ; Tumasjan 2010). En effet, les mots étendus pourraient intensifier ou diminuer la charge émotionnelle (le degré de polarité) du poste. Par conséquent, nous accordons une grande attention à ce genre de mots dans notre approche en équilibrant le score des mots en s'appuyant sur ces métriques.

    Ainsi, nous avons considéré « intensificateurs » les mots de sentiment positif étendus et « diminutifs » ceux de sentiment négatif.

    Comme expliqué ci-dessus, dans notre approche, nous avons introduit la notion de mots étendus pour renforcer la charge émotionnelle des messages ; cependant, nous avons fait le choix de ne pas distinguer les différentes façons d'accentuer un sentiment par son auteur. Par conséquent, nous affectons le même poids +1 ou -1 pour les mots étendus positifs ou négatifs, respectivement. Par exemple, si $score(love) = 1$, alors

    $score\ (looove) = score\ (LOVE) = \ score\ (LOOOVE) = 2.$
  - *Calcul de la polarité* : L'analyse de sentiment basée sur le lexique implique l'extraction des scores de sentiment à partir d'un dictionnaire. La polarité d'un

poste est donc calculée en additionnant les valeurs de score {+1,-1} des mots w indiqués dans le poste prétraité t.

$$p(t) = \sum_{k=1}^{m} score(w) \qquad (20)$$

m représente la taille de t.

- **Sous-étape 4** : pour classer les tweets, nous utilisons $p\ (t)$ comme suit : Nous classons les tweets notés dans sept classes selon leur degré de polarité : $C_{+3}, C_{+2}, C_{+1}, C_0, C_{-1}, C_{-2}, C_{-3}$ (fort positif, modérément positif, légèrement positif, neutre, légèrement négatif, modérément négatif, fort négatif, respectivement). Pour cela, nous avons utilisé un test empirique sur un échantillon de 2000 tweets pour déterminer les limites de chaque classe.

  Fort positif (C_{+3},) : $p\ (t) \geq\ 7$.

  Modérément positif : $4 \leq p(t) \leq 6$.

  Légèrement positif : $0 < p(t) \leq 3$.

  Légèrement négatif : $-3 \leq p(t) < 0$.

  Modérément négatif : $-6 \leq p\ (t) \leq 4$.

  Fort négatif : $p\ (t) \leq -7$.

  Il est possible que le score de sentiment d'un message soit égal à 0, alors le tweet est classé comme neutre.

Les différentes sous-étape de notre approche classification de sentiments sont illustrées dans la figure 17.

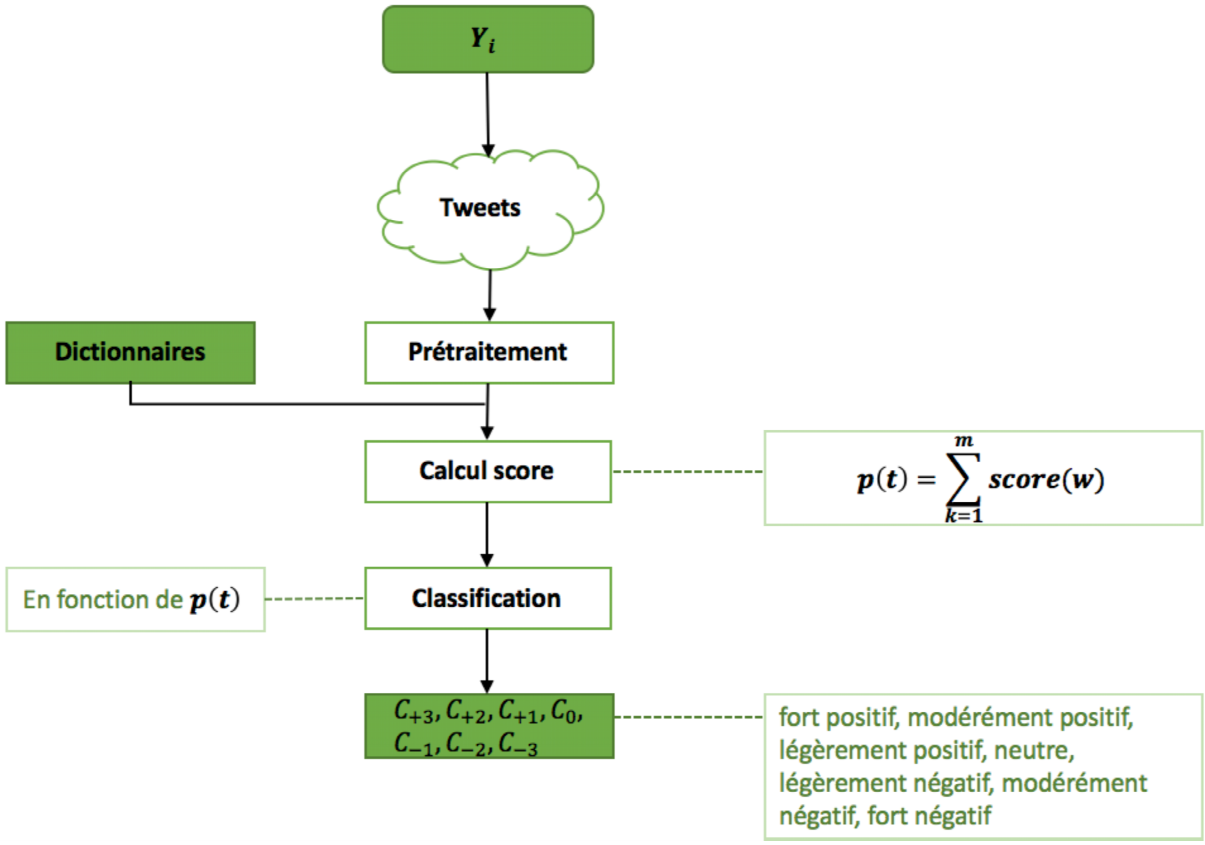


Figure 17: La deuxième étape du modèle

3.3. Troisième étape : la prédiction

Dans cette étape, nous calculons le degré d'appréciation d'un sujet Yi par rapport à l'ensemble des sujets traités dans une domaine donné.

- **Sous-étape 1 :** Plusieurs chercheurs tels que H. Wang and Castanon (2015) ont considéré des classes positives, négatives et neutres pour classifier le sentiment d'un document et seulement quelques-uns, comme Aparup Khatua et al. (2015), ont examiné le degré de polarité (classes fort, modérément, faiblement positives ou négatives). Cependant, pour mesurer l'opinion publique, les auteurs n'ont considéré que des classes fort positives et fort négatives comme indicateurs. Contrairement aux recherches précédentes, nous supposons que chaque classe a un impact spécifique sur la force de l'opinion publique de Y_i dans un domaine donné.

  Pour cela, nous attribuons un poids pour chaque classe comme suit :

  $poids\ C_0 = 0, \quad poids\ C_{+1} = 1, \quad poidsC_{+2} = 2, \quad poids\ C_{+3} = 3,$

$$poids\ C_{-1} = -1, poids\ C_{-2} = -2\ et\ poids\ C_{-3} = -3.$$

- **Sous-étape 2 :** Nous supposons qu'un message qui a été partagé ou aimé de nombreuses fois n'a pas le même poids (degré d'influence) que celui qui n'a pas été partagé ou aimé . Un partage est un moyen de soutenir une opinion ou de partager un message que les utilisateurs considèrent comme intéressant. Dans le contexte étudié, cela peut se faire de deux manières : via le bouton Retweet ou à l'ancienne en écrivant RT, le nom de l'utilisateur et le message à retweeter. Dans notre cas, nous ne considérons que la première manière car le retweet à l'ancienne pourrait être différent du tweet d'origine.
  Pour mesurer le degré d'influence d'un tweet $Inf\ (t)$, nous ajoutons d'autres métriques qui équilibrent le poids d'un tweet dans l'équation 21. Nous définissons ces métriques comme le nombre de retweets et le nombre de likes d'un tweet donné ($NR\ (t), NL\ (t)$ respectivement).

$$Inf\ (t) = poids\ (t) \times \big(NR\ (t) + NL\ (t)\big) \qquad (21)$$

  $poids\ (t)$ est le poids de la classe à laquelle il appartient.
- **Sous-étape 3 :** est la dernière étape de notre approche, dans laquelle nous calculons la note globale qui constitue le degré d'appréciation de chaque sujet Y_i: $\big(SC(Y_i)\big)$. Tout d'abord, nous calculons le degré d'influence pour tous les tweets relatifs à Y_i. $SC(Y_i)$ est déterminé par, la somme de $Inf\ (t_{Y_i})$ et le volume total des messages relatifs à Y_i comme suit :

$$SC\ (Yi) = \frac{\sum_{k=1}^{q} Inf\big(t_{Y_i}\big)}{q} \qquad (22)$$

  $Inf\ \big(t_{Y_i}\big)$ tweets concernant Y_i.
  q est le nombre de tweets relatif à Y_i.
  $SC\ (Yi)$ représente le degré d'appréciation d'un sujet Yi par rapport à l'ensemble des sujets traités dans une thématique donnée.

Les différentes sous-étapes sont illustrées dans la figure 18.

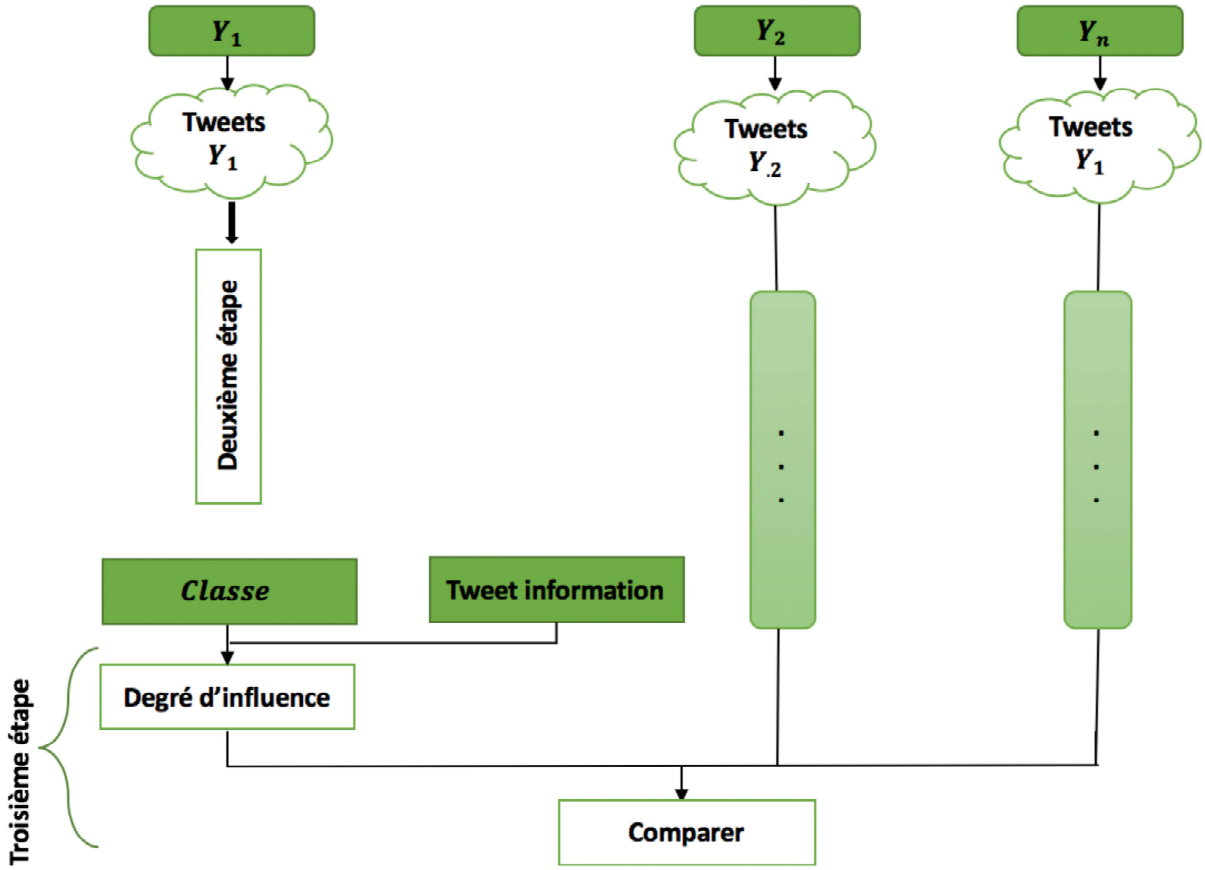


Figure 18: la troisième étape du modèle

.

Dans cette étape, nous calculons le degré d'appréciation d'un sujet Yi par rapport à l'ensemble des sujets traités dans une domaine donné.

4. Expérience et résultats

Dans cette partie, nous présentons les expérimentations que nous avons menées afin de mesurer la robustesse et la performance de la méthode de classification proposée. L'objectif principal de cette expérience est d'évaluer la pertinence des nouvelles métriques introduites à partir de l'analyse contextuelle, par rapport à la classification des sentiments. Pour cela, l'approche de classification est comparée à plusieurs méthodes existantes dans la littérature.

Au cours des dernières années, la littérature s'est intéressée de plus en plus à la possibilité d'analyser les médias sociaux comme complément utile aux sondages traditionnels hors ligne pour suivre une campagne électorale. Certains chercheurs prétendent que ce faisant, nous pouvons également produire une prévision du résultat. S'appuyer sur une méthodologie appropriée pour l'analyse des sentiments reste une question cruciale à cet égard. Du suivi des

discussions sur les débats politiques à la prédiction des résultats électoraux, les réseaux sociaux sont devenus une mine d'or pour la recherche sur le sentiment politique. Pour cette raison, nous menons une étude sur l'analyse des tweets relatifs aux élections présidentielles américaines de 2016 afin de déterminer le candidat favori chez les internautes. Cette expérimentation a pour but d'évaluer l'approche proposée à classifier les sentiments des internautes (dans notre cas, les votants) et à prédire les résultats des élections présidentielles.

Les médias sociaux, tels que Facebook ou Twitter, sont de plus en plus utilisés lors des élections (présidentielles particulièrement), d'un côté par les candidats pour assurer le suivi de leurs campagnes, partager leurs programmes et communiquer avec les électeurs, et de l'autre côté par ces derniers pour exprimer leurs opinions sur les candidats. Dans la présente étude, nous limitons notre étude à Twitter, un service de micro-blogging qui permet aux utilisateurs de poster et d'interagir avec des messages appelés "tweets", limités à 140 caractères. Cette information est largement accessible aux développeurs et aux chercheurs grâce à l'API REST qui permet de collecter des tweets relatifs à un hashtag donné.

4.1. Implémentation

Les systèmes d'analyse de big data doivent avoir suffisamment de mémoire, de bande passante et la capacité d'effectuer un traitement parallèle des tâches en temps réel. Pour répondre aux besoins du big data, nous avons implémenté notre système prototype sur un cluster de 3 serveurs. Chaque serveur dispose de deux processeurs Intel Xeon E5530 quad-core Intel 2,4 GHz fonctionnant sous Linux Ubuntu 64 bits. Les serveurs sont équipés de 24 Go de RAM DDR3 et de 1 To de disque dur.

Concernant la collecte de données, nous avons utilisé Apache Kafka qui est une plateforme de diffusion distribuée. Kafka utilise la messagerie de publication-abonnement et offre un service distribué et répliqué. Plus spécifiquement, nous avons utilisé la bibliothèque d'API Streams intégrée qui permet de créer des applications pour le traitement de flux de données. Ensuite, le grand volume de données collectées est stocké dans HDFS (Hadoop Distributed File System).

Quant au traitement des données (construction des dictionnaires et classification des sentiments), c'est Spark qui est utilisé. En effet, Spark permet un traitement évolutif et tolérant aux pannes des flux de données en temps quasi réel comme expliqué dans le chapitre 2. Dans cette expérimentation, Spark est utilisé sur Hadoop, géré par YARN et distribue le traitement sur 3 nœuds.

De même, afin de comparer la méthode proposée avec d'autres méthodes, nous utilisons l'implémentation de Spark MLlib du classifieur Naïve Bayes pour classifier les tweets en temps réel. Le modèle créé par Naïve Bayes est appliqué aux tweets collectés en temps réel.

Tout l'algorithme de collecte et de traitement des données a été implémenté en utilisant le langage Python.

4.2. La collecte des données de l'expérience

- **Première étape du système :** Pour se limiter aux élections présidentielles américaines de 2016, nous avons utilisé les deux noms de candidats : Hillary et Trump comme mots clés pour collecter les tweets ($D = \{Y_1, Y_2\}\, avec\, Y_1 = Trump\, et\, Y_2 = Hillary$). Les données sont collectées à partir de l'API (Application Programming Interface) Twitter REST en temps réel et les tweets doivent être rédigés en anglais. Afin d'identifier les hashtags les plus populaires, nous avons généré une liste de fréquence des hashtags utilisés dans l'ensemble des données collectées. Nous avons sélectionné les hashtags de cette liste, dont la polarité est évidente à déterminer, puis nous les avons classifiés comme suit :

  Trump positif : #MAGA, #makeamerikagreatagain et #voteTrump

  Négatif Trump : #NeverTrump #dumptrump.

  Positif Hillary : #imwithher, #strongertogether et #voteHillary,

  Hillary négatif : #podestaemails, #NeverHillary et #crokeedHillary.

  Afin de valider la méthode de construction du dictionnaire dynamique, nous avons collecté les données relatives aux hashtags choisis, entre les 5 et 6 octobre 2016. Les données du prototype contiennent un total de 120 000 tweets (30 000 tweets pour chaque classe).
- **Deuxième étape :** Constitue la partie classification des tweets. Pour cela, nous avons recueilli tous les messages twitter postés entre les 6 et 7 novembre 2016 (soit un jour avant les élections présidentielles). Cette base contient un total de 3 600 000 tweets.
- **Troisième étape :** Pour la partie prédiction des résultats des élections présidentielles américaines, nous avons utilisé le résultat des données traitées dans la deuxième étape ainsi que les informations sur les tweets (la classe, le poids de la classe à laquelle appartient le tweet, le nombre de retweets et de likes) pour déterminer le degré d'appréciation d'Hillary et Trump.

4.3. Traitement des données

- **Première étape** : Ici, nous avons construit des dictionnaires positifs et négatifs pour les deux sujets : Hillary et Trump. Pour cela, nous avons traité les données collectées relatives aux hashtags choisis comme mentionné dans la sous-section 3.1. Un exemple est présenté ci-dessous.

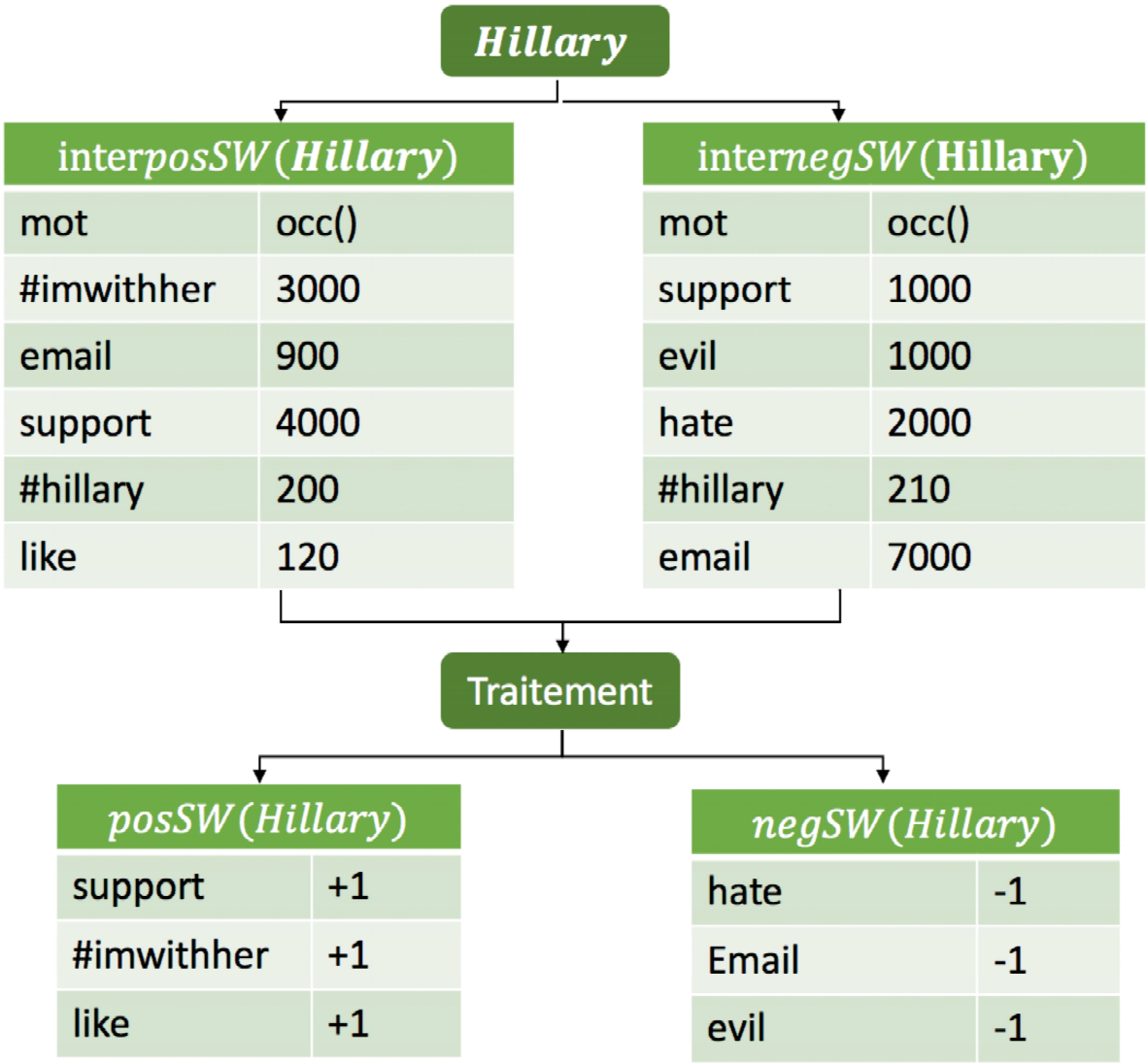


Figure 19: Exemple de construction de dictionnaire Hillary

- **Deuxième étape** : Premièrement, nous avons prétraité les données en supprimant les tweets en double et les mots vides, et en appliquant les filtres comme mentionné dans la deuxième phase de notre système, sous-section 3.2.
  Ci-après un exemple de traitement et de classification :

Tableau 9 Exemple de tweet

Date	Retweet	Likes	Tweet
06/11/2016 23:05:00	4	12	#ImWithHer #Hillary #StrongerTogether I SUPPORT her. #DumpTrump

Tableau 10 Application de la deuxième étape du système

Sous-Etape 2	**Prétraitement**	#ImWithHer #Hillary #StrongerTogether intensifier(support) #DumpTrump
Sous-Etape 3	**Equilibrage**	support
	Score	4
Sous-Etape 4	**Classification**	C_{+2}

- **Troisième étape** : Premièrement, nous calculons le degré d'influence pour les données collectées.

  Ce traitement est appliqué à l'exemple précédent dans le tableau 11.

Tableau 11 Application de la troisième étape du système

Sous-Etape 1	**Poids**	2
Sous-Etape 2	**Influence**	2*(4+12) = 32

Ensuite, nous avons compilé ces résultats afin de déterminer le candidat le plus apprécié et prédire les résultats des élections présidentielles à travers le degré d'appréciation $SC\ (Yi)$ de chaque candidat.

Dans la sous-section suivante, nous évaluons les résultats de l'approche proposée.

4.4. Performances

Afin de valider notre approche, nous avons évalué tout d'abord la performance de la tâche de classification des postes. Ensuite, nous l'avons comparée à d'autres outils d'analyse de

sentiments tels que IBM Waston. Enfin, nous avons étudié la possibilité de prédire correctement les résultats des élections américaines de 2016.

4.4.1. Évaluation de la précision de la classification

Pour évaluer la capacité de notre modèle à classifier les tweets en se basant sur le dictionnaire dynamique, construit automatiquement, nous avons sélectionné au hasard un ensemble de 700 tweets issus de corpus politiques : 50 pour chacune des sept classes. Ces tweets ont été soigneusement inspectés et étiquetés manuellement comme légèrement positifs, modérément positifs, fort positifs, légèrement négatifs, modérément négatifs, fort négatifs ou neutres pour chaque candidat. Ensuite, les mêmes données ont été prétraitées, comme mentionné ci-dessus, en supprimant les mots vides, en appliquant la tokenisation, et divers filtres. Cette étape a été réalisée par TreeTagger, un outil d'annotation de texte qui permet d'identifier des parties du discours, de délimiter des groupes syntaxiques et d'effectuer la lemmatisation. TreeTagger a été modifié pour traiter la négation, les URL, les noms d'utilisateurs, les mentions Twitter, les hashtags, les mots étendus. Nous avons donc associé des étiquettes morphosyntaxiques à chaque mot dans le corpus. L'étiquetage morphosyntaxique consiste à associer chaque mot à une classe morphosyntaxique à savoir sa catégorie grammaticale, nombre, genre…etc (tableau 12)

Tableau 12 Exemple d'étiquetage morphosyntaxique

Etiquette morphosyntaxique	**Descriptif**
ABR	Abréviation
ADJ	Adjectif
VER :pper	Verbe au participe passé
VER :infi	Verbe à l'infinitif
NAM	Nom propre

Voici un exemple d'étiquetage morphosyntaxique de la phrase suivante : « treetagger permet d'annoter plusieurs langues » :

Tableau 13 Exemple d'étiquettage Treetager

Mot	Etiquette	Racine
Treetagger	NAM	<unknown>
permet	VER :pres	permettre
d’	PRP	de
annoter	VER :infi	annoter
plusieurs	PROD :IND	plusieurs
langues	NOM	langue

Nous avons comparé l’approche proposée avec les « classifieurs » suivants en utilisant le même corpus prétraité avec Treetager afin d’évaluer ses performances :

- **Google Cloud Prediction API** : fournit une API RESTful en tant que boîte noire pour créer des modèles d’apprentissage automatique basés sur un ensemble de données d'apprentissage. Ces modèles de prédiction basés sur le cloud, analysent des données dans plusieurs domaines tels que l'analyse de sentiment du client, les systèmes de recommandation ou la détection de spam.
- **Naïve Bayes** : est le « classifieur » le plus simple et le plus utilisé comme expliqué dans le chapitre 3. Ce modèle calcule la probabilité a posteriori d'une classe, basée sur un sac de mots, et utilise le théorème de Bayes pour prédire la probabilité qu'un ensemble de caractéristiques donné appartienne à une classe particulière. Dans notre cas, le modèle classifie les tweets en sous-classes positives et négatives.

Nous avons utilisé les quatre métriques : l’exactitude de la classification « accuracy » (la précision de la classification sans considération des données faux étiquetés), la Précision, le Rappel, et le F-score (voir chapitre 3), les plus utilisées pour évaluer la performance d'une méthode de classification. Dans notre expérience, la performance de classificateurs (Table 14) est calculée en considérant la moyenne des quatre métriques pour chaque classe des deux candidats.

Les résultats obtenus révèlent que la classification utilisant notre approche atteint une bonne précision (90,21%, 90,23, 89,57% et 89,98% précision moyenne, macro précision, macro rappel et macro F-score, respectivement) comparée aux classifieurs Naïve Bayes (34,45%, 33,92%, 31,85% et 32,86%) et à l'API Google Prediction (66,66%, 64,02%, 65,31% et 64,65%). Dans cette expérience, cela confirme la supériorité de notre approche lexicale utilisant le dictionnaire

dynamique. Nous pourrions expliquer cela par le fait que la méthode proposée se base sur un équilibrage de score (les intensificateurs et les diminutifs) pour associer une classe à un message donné.

Tableau 14: comparaison des performances des classifieurs

	Méthode proposée					Naïve Bayes					Google Prediction API			
	Accuracy	**Accuracy moyenne**	**Macro précision**	**Macro rappel**	**Macro F-score**	**Accuracy**	**Accuracy moyenne**	**Macro précision**	**Macro rappel**	**Macro F-score**	**Accuracy moyenne**	**Macro précision**	**Macro rappel**	**Macro F-score**
Fort positive Trump	88,2%					85,71								
Modérément positive Trump	88%					35,71%								
Légèrement Positive Trump	88%					14,21								
Fort positive Hillary	98%					21,42%								
Modérément positive Hillary	86%					14,28%								
Légèrement Positive Hillary	86%	**90,21%**	**90,23%**	**89,57%**	**89,98%**	57,14%	**34.45%**	**33,92%**	**31,85%**	**32,86%**	**66,66 %**	**64,02%**	**65,31%**	**64,65%**
Fort négative Trump	90%					0,70%								
Modérément négative Trump	96%					57,14%								
Légèrement Négative Trump	88%					42,28%								
Fort négative Hillary	98%					14,28%								
Modérément négative Hillary	88%					14,28%								
Légèrement Négative Hillary	88%					57,14%								

4.4.2. Comparaison avec d'autres méthodes existantes dans la littérature

Tableau 15: Performances des méthodes d'analyse de sentiments

Approche	Etude	Données utilisées	Méthodes, référentiels	Performance
Basé sur le lexique	(Kucuktunc et al. 2012)	Yahoo	Sentistrenght	0,4939 (RMSE)
	(Balahur et al. 2013)	reviews on newspaper articles	Sentiwordnet	82.0%(précision)
	(Jaap, Kamps et al. 2004)		wordnet	60,4% (précision)
	(Edison and Aloysius 2017)	Twitter	Synset et SemEva	73,27% (précision)
	(Rehman and Bajwa 2017)	Plusieurs sites web	Dictionnaire anglais traduit en urdu	66% (précision)
Basé sur l'apprentissage automatique	(Pang, Lee, and Vaithyanathan 2002)	IMDb Movie Review	SVM	82,9 %(précision)
	(Duyu, Tang et al. 2014)	Twitter	Les réseaux de neurone	86,48 (F-Score)
	(Zhao et al. 2012)	Twitter	Naive Bayes + emoticons	58,3 % (F-score)
	(Luo, Osborne, and Wang 2015)	Twitter	CoTraining SVM	82.52% (précision)
	(Jin, Wei, Hay Ho, Hung, and K. Srihari, Rohini 2009)	reviews from Amazon	hidden Markov mode + Bootstrapping	74.8% (F-score)
	(Kessler and Nicolov 2009)	Blog	SVM	69.8% (F-score)
Hybride	(Saif, He, and Alani 2012)	Twitter	Analyse sémantique dans Naïve Bayes	84,25% (précision)
	(Zainuddin, Selamat, and Ibrahim 2017)	Twitter	Sentiwordnet + SVM+NB+ autres	76.55%, 71.62 % (F-score)
	(Mukwazvure and Supreethi 2015)	Les commentaires de www.theguardian.com	AFFIN-111 + kNN SVM	55.58% – 74.24 % (précision)

	(Er et al. 2016)	Twitter	lexicon-based and learning-based method(pas d'accès)	81.9% (précision)
	(Rojratanavijit, Vichitthamaros, and Phongsuphap 2016)	Twitter	Wiktionary + SVM	82.97 % (F-measure)

La table ci-dessus résume la performance en % des différentes techniques d'analyse des sentiment issues des tests effectués par divers chercheurs dans différents domaines. Nous pouvons remarquer que les études basées sur l'analyse par l'apprentissage automatique ont une précision relativement élevée. Malgré ce fait, ces méthodes nécessitent un étiquetage laborieux pour construire les données de formation qui par ailleurs ne peuvent être utilisées dans un autre domaine que le contexte étudié. En effet, Zhuang, Jing, and Zhu (2006) ont formé des « classifieurs » hautement précis pour la cinématographie. Cependant, leur modèle ne peut être appliqué qu'aux critiques de films et nécessite un grand volume de corpus étiquetés manuellement.

Nous pouvons également constater que de la méthode proposée améliore la performance et apporte de l'innovation par rapport aux systèmes existant dans la littérature ; en intégrant la construction d'un dictionnaire contextuel dynamique ainsi que de nouvelles métriques tels que les mots étendus.

4.5. Comparaison avec d'autres outils de l'analyse de sentiments

Malheureusement, nous n'avons pas accès aux performances des outils les plus populaires tels que IBM Watson, Rapidminer, Meaning cloud et StreamCrab pour pouvoir situer les résultats obtenus par notre méthode. Cependant, nous nous sommes contentés d'établir une comparaison de notre méthode avec celles des outils précités (tableau 16).

Tableau 16: Comparaison avec d'autres outils de l'analyse de sentiments

	L'approche proposée	**Rapidminer**	**IBM Watson**	**Meaning Cloud**	**Stream Crab**
Dictionnaire construit automatiquement	Oui	Non	Non	Non	Non
Dictionnaire contextuel	Oui	Non / Personnalisable	Non / Personnalisable	Non	Non
Degré de classification	(Fort, Modérément, légèrement) Positive, Négative et Neutre	Positive, Négative et Neutre / Personnalisable	Positive, Négative et Neutre	Positive, Négative et Neutre	Positive et Négative
Intensificateur	Oui	Non	Non	Non	Non
L'utilisation des outils big data	Oui	Non (peut être intégré avec Hadoop)	Oui	Non	Non
Visualisation	Non	Oui	Oui	Oui	Oui (limité)

Contrairement aux outils cités ci-haut, la méthode proposée repose sur la construction automatique d'un dictionnaire dynamique et contextuelle, permettant ainsi de répondre aux défis de l'analyse de sentiments en termes d'adaptabilité. En outre, elle permet une classification affinée (sept classes), en prenant en compte les spécificités du langage informel utilisé dans les big social data telles que les mots étendus (considérés comme intensificateurs ou diminutifs). Notre système se distingue également par l'utilisation des outils big data, permettant ainsi de répondre aux besoins de traitement des données massives en temps réel.

Comparé à ces outils, notre système n'offre pas une représentation graphique de données. Cependant, la visualisation des données est un axe très important dans l'analyse de données comme expliqué dans le chapitre 2. C'est donc dans cette perspective que l'on souhaite améliorer notre système.

4.6. Évaluation de l'exactitude de la prédiction

Bien que de nombreuses études antérieures, telles que celles menées par Gayo-Avello (2012) et Tumasjan (2010) aient démontré le manque de fiabilité de l'estimation des résultats électoraux en utilisant les réseaux sociaux, d'autres chercheurs (Franch 2013 ; Burnap et al. 2016) ont utilisé ces derniers pour mesurer l'opinion publique et prédire les résultats des élections. En effet, ces dernières ont amélioré les précédentes méthodes en intégrant d'autres méthodes dont l'analyse de sentiments.

Comme exposé plus haut, nous avons choisi de démontrer la faisabilité et le potentiel de notre approche via un cas d'application ; la prédiction des résultats des élections présidentielles américaines de 2016. Nous avons donc appliqué la dernière étape de l'approche proposée sur les tweets collectés entre les 6 et 7 novembre 2016 (soit un jour avant les élections). En effet, en calculant le degré d'appréciation des sujets étudiés $SC(Y_i)$, nous avons pu prédire le candidat le plus apprécié par les internautes et donc vainqueur des élections. Les résultats sont présentés dans le tableau ci-dessous.

Tableau 17 Résultat de l'élection présidentielle américaine

	Donald Trump	**Hillary Clinton**
$SC\ (Yi)$	30.85	18.34

A partir de ce tableau, il est clair que le sentiment public est plus en faveur de Trump. Ainsi, notre modèle prédit que Donald Trump sera élu président de 2016 aux États-Unis.

Ces résultats sont les conséquences de la construction d'un dictionnaire contextuel et dynamique, de la finesse de classification de sentiments des tweets en sept classes et l'intégration de plusieurs nouvelles métriques qui prennent en compte les spécificités du langage informel utilisé dans les réseaux sociaux.

5. Conclusion

L'analyse de sentiments ou la polarité de l'opinion s'est avérée efficace pour prédire l'attitude des populations en analysant les big social data. Pour cela, nous avons introduit dans ce chapitre une nouvelle approche adaptable qui vise à extraire le sentiment des internautes sur un sujet spécifique en s'appuyant sur les réseaux sociaux. La nouveauté apportée par cette technique proposée est

l'intégration d'une analyse contextuelle et l'introduction de nouvelles métriques. Elle consiste à construire d'abord un dictionnaire contextuel dynamique à partir d'un ensemble réduit de hashtags relatifs à un sujet donné, classés manuellement positifs ou négatifs. Ensuite, la méthode permet une classification affinée (sept classes) en utilisant de nouvelles métriques telles que les mots étendus.

Un grand nombre d'utilisateurs partagent leurs opinions sur les réseaux sociaux, ce qui en fait une plate-forme précieuse pour suivre et analyser le sentiment public. Cette analyse peut fournir des informations cruciales pour la prise de décision et l'accompagnement des grands événements tels que les élections présidentielles.

Pour cela, nous avons choisi de tester notre méthode pour analyser le sentiment des internautes pendant les élections présidentielles américaines 2016. L'objectif principal de cette évaluation est de déterminer la pertinence des nouvelles métriques introduites à partir de l'analyse contextuelle dans la classification des sentiments. La comparaison de notre méthode avec différents paradigmes rapportés dans la littérature confirme l'apport de notre méthode dans la conception des systèmes d'analyse de sentiments très précis. En effet, notre modèle est capable d'atteindre une précision globale de 90,21%, dépassant largement les modèles de référence actuels dans l'analyse du sentiment des réseaux sociaux.

Conclusions et perspectives

Tout d'abord, une étude bibliographique sur les « big data » et les « big social data » réalisée dans le premier chapitre a montré d'une part la diversité d'outils de traitement big data et d'autre part les limites des méthodes d'analyse de sentiment pour les big social data. Après avoir identifié ces limites, il était nécessaire de fournir une étude comparative actualisée des différents outils utilisés pour extraire l'information stratégique du big data et les mapper aux différents besoins de traitement. Celle-ci a fait l'objet du 2ème chapitre. Ensuite, il était nécessaire de restreindre le contexte d'analyse. Nous avons donc choisi de nous intéresser aux méthodes d'analyse de sentiments des big social data dans le 3ème chapitre. Ce chapitre identifie également les limites d'analyse de sentiments que nous essayons de résoudre dans le 4ème chapitre. Ce dernier présente la méthode proposée qui vise à extraire le sentiment des internautes sur un sujet spécifique en s'appuyant sur les spécificités des big social data. Afin de vérifier la faisabilité de cette approche, un protocole expérimental a été établi. La plate-forme big data expérimentale utilisée pour les besoins de cette recherche est décrite dans le chapitre 4. Egalement, ce chapitre présente les résultats expérimentaux qui ont conforté notre démarche. Nous avons ainsi pu mettre en évidence pertinence de la méthode de classification de sentiments proposée. Le cinquième et dernier chapitre de ce mémoire replace cette recherche dans le contexte global de l'étude et en aborde les développements et les perspectives.

1. Conclusions

L'utilisation des médias sociaux est devenue une opération quotidienne de tout le monde dans l'ère numérique d'aujourd'hui. Par conséquent, d'énormes quantités de données en temps réel sont générées chaque seconde à travers le monde, principalement sous formes de messages textuelles non structurés. Ces données constituent une grande opportunité d'analyse de tendances et d'opinion en temps réel chez les internautes. Le traitement de ce type de données, qualifiées de big social data, implique diverses voies de recherche, en particulier l'analyse textuelle et l'analyse de sentiments. En effet, les big social data regorgent de données textuelles (dogmatique) traduisant des opinions.

La plupart des travaux dans ce domaine utilisent soit des connaissances lexicales antérieures en termes de polarité des mots, soit des méthodes d'apprentissage automatique en traitant la tâche

d'identification des sentiments comme un problème de classification de texte (où la machine apprend à classer le sentiment à base des données de formation étiquetées). Cependant, la couverture limitée des mots spécifiques au domaine dans les lexiques universels existants, tels que SentiWordNet, peut attribuer des scores erronés à des mots spécifiques au domaine et pourrait donc entrainer une classification incorrecte des sentiments. En effet, la polarité d'un mot dépend généralement d'un domaine particulier et pourrait changer en fonction du domaine. Par conséquent, les approches utilisant les données de formation pour un domaine spécifique fonctionnent généralement mieux que les approches basées sur le lexique indépendamment du domaine. Néanmoins, les méthodes d'apprentissage automatique nécessitent un étiquetage laborieux pour construire les données de formation qui ne peuvent être utilisées dans un domaine autre que celui étudié. Par conséquent, Le principal obstacle des applications utilisant les algorithmes supervisés ou le lexique de sentiment spécifiques à un domaine est l'indisponibilité de jeux de données d'apprentissage étiquetés pour chaque domaine.

Actuellement, l'objectif principal des applications utilisant les big social data est de rendre la machine capable d'identifier en temps réel, les émotions et les sentiments dans des domaines variés. L'analyse des big social data apporte donc de nombreux défis tels que l'adaptabilité et le traitement des données massives en temps réel.

Cette thèse se concentre principalement sur deux aspects. Le premier est l'étude et la comparaison des différents outils big data existant afin de choisir les outils adéquats à l'objectif d'étude et la nature des données. Le deuxième est l'élaboration d'une approche générique d'analyse de sentiments qui peut s'adapter à plusieurs domaines. La technique proposée consiste à d'abord construire un dictionnaire contextuel et dynamique basé sur un ensemble réduit de hashtags, relatifs à un sujet, classifiés manuellement positifs ou négatifs liés à un sujet donné. Le poids des sentiments est équilibré en introduisant de nouvelles métriques telles que les mots étendus.

Contrairement aux travaux antérieurs dans l'analyse de sentiments qui reposent principalement sur des classifications binaires (négative ou positive), nous soutenons que le degré de polarité est important. Pour cela, notre méthode procède à une classification plus fine des messages dans sept classes (fort positif, modérément positif, légèrement positif, neutre, légèrement négatif, modérément négatif et fort négatif).

Nous avons mené une étude sur l'analyse des tweets relatifs aux élections présidentielles américaines de 2016 afin de déterminer le candidat favori chez les internautes. Cette expérimentation a pour but d'évaluer l'approche proposée à classifier les sentiments des internautes (dans notre cas, les électeurs) et à prédire les résultats des élections présidentielles. Les tests de

performance ont montré des résultats prometteurs. Nous avons pu effectivement améliorer l'opération de la classification des sentiments en se basant sur un dictionnaire contextuel dynamique et les nouvelles métriques introduites.

Bien que le modèle seul pourrait ne pas être suffisant pour prédire les résultats, il devient un élément décisif lorsqu'il est combiné avec d'autres modèles statistiques et des techniques hors ligne (comme les sondages).

2. Perspectives

La méthode proposée fonctionne bien pour classer les sentiments dans les tweets. Nous croyons que la précision pourrait encore être améliorée. Notre travail ne s'arrête donc pas là.

Plusieurs possibilités d'évolution s'offrent à nous. En effet, nous avons introduit la notion de mots étendus pour renforcer la charge émotionnelle des messages ; cependant, nous avons fait le choix de ne pas distinguer les différentes façons d'accentuer un sentiment par son auteur. Par conséquent, nous affectons le même poids +1 ou -1 pour les mots étendus positifs ou négatifs, respectivement. Dans un court terme, nous pourrons distinguer le degré d'impact des différentes mesures afin d'accentuer un sentiment. Par exemple, si $score(love) = +1$, alors $score\ (looove) = +2\ et\ score\ (LOOOVE) = +3$. Cependant, il reste un problème de recherche pour déterminer les mesures d'accentuation de la charge émotionnelle.

Dans l'expérimentation, nous avons utilisé uniquement des données Twitter. Une information seule n'a pas la même valeur que combinée aux autres. Nous pouvons donc considérer les données provenant de diverses sources telle que facebook, instagram, etc… Cette agrégation impliquera la prise en compte des spécificités des données issues des autres sources notamment en termes de structures et de types de données. Nous pensons que cet axe rendrait notre méthode et plateforme plus générique.

La classification des hashtags est faite de manière manuelle. Dans un moyen terme, il serait intéressant d'automatiser l'opération de la classification des hashtags afin de construire le dictionnaire d'une manière totalement automatique. Le langage d'un hashtag est informel, en effet, il est composé du signe typographique « # », suivi d'un ou plusieurs mots accolés. Par conséquent, l'analyse de sentiment formel ne nous permets pas de classifier automatiquement ce type de langage. Malheureusement, il n'existe pas encore d'études scientifiques traitant la classification de hashtags de manière automatique.

Nous pensons également qu'il serait intéressant d'intégrer la détection des qualificatifs de tournure de phase tels que l'ironie et le sarcasme afin d'améliorer la performance de la classification des sentiments. Ces tournures ont été étudiées dans plusieurs domaines tels que la linguistique et la psychologie. Cependant, la détection de l'ironie et du sarcasme est une tâche qui demeure complexe même pour les humains. Ceci est particulièrement difficile lorsque l'on traite les big data, où le langage est informel.

Enfin, nous prévoyons d'adapter et d'utiliser notre approche d'analyse dynamique et contextuelle des sentiments également dans d'autres domaines, tels que le domaine de marketing.

Bibliographie

A. M. H., Elyasir, and K. S. M., Anbananthen. 2013. “Opinion Mining Framework in the Education Domain.” *World Academy of Science, Engineering and Technology International Journal of Educational and Pedagogical Sciences* 7 (4).

Abdallah, Zahraa S., Mark Carman, and Gholamreza Haffari. 2017. “Multi-Domain Evaluation Framework for Named Entity Recognition Tools.” *Computer Speech & Language* 43 (May): 34–55. https://doi.org/10.1016/j.csl.2016.10.003.

Ahmed, S., M. Pasquier, and G. Qadah. 2013. “Key Issues in Conducting Sentiment Analysis on Arabic Social Media Text.” In *2013 9th International Conference on Innovations in Information Technology (IIT)*, 72–77. https://doi.org/10.1109/Innovations.2013.6544396.

Aivalis, C. J., K. Gatziolis, and A. C. Boucouvalas. 2016. “Evolving Analytics for E-Commerce Applications: Utilizing Big Data and Social Media Extensions.” In *2016 International Conference on Telecommunications and Multimedia (TEMU)*, 1–6. https://doi.org/10.1109/TEMU.2016.7551938.

Almeida, P. D. C. d, and J. Bernardino. 2015. “Big Data Open Source Platforms.” In *2015 IEEE International Congress on Big Data*, 268–75. https://doi.org/10.1109/BigDataCongress.2015.45.

Altawaier, Merfat M., and Sabrina Tiun. 2016. “Comparison of Machine Learning Approaches on Arabic Twitter Sentiment Analysis.” *International Journal on Advanced Science, Engineering and Information Technology* 6 (6): 1067–73. https://doi.org/10.18517/ijaseit.6.6.1456.

Altrabsheh, Nabeela, Mohamed Gaber, and Mihaela Cocea. 2013. “SA-E: Sentiment Analysis for Education.” In *Frontiers in Artificial Intelligence and Applications*. Vol. 255. https://doi.org/10.3233/978-1-61499-264-6-353.

“Analyses Guidées | Logiciel de Business Intelligence | QlikView.” n.d. Accessed April 14, 2018. http://www-prod.qlik.com/fr-fr/products/qlikview.

Anastasia, S., and I. Budi. 2016. “Twitter Sentiment Analysis of Online Transportation Service Providers.” In *2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 359–65. https://doi.org/10.1109/ICACSIS.2016.7872807.

“Apache Flink: Scalable Stream and Batch Data Processing.” n.d. Accessed January 27, 2017. https://flink.apache.org/.

“Apache Hadoop 2.7.1 – Apache Hadoop NextGen MapReduce (YARN).” n.d. Accessed January 23, 2017. https://hadoop.apache.org/docs/r2.7.1/hadoop-yarn/hadoop-yarn-site/YARN.html.

“Apache Hadoop Releases.” 2012. 2012. http://hadoop.apache.org/releases.html.

“Apache Hive TM.” n.d. Accessed December 23, 2016. https://hive.apache.org/.

“Apache Kafka.” n.d. Accessed January 27, 2017. https://kafka.apache.org/intro.

“Apache Mahout: Scalable Machine Learning and Data Mining.” n.d. Accessed January 27, 2017a. http://mahout.apache.org/.

"Apache SAMOA." n.d. Accessed February 15, 2017. https://samoa.incubator.apache.org/.

"Apache Solr -." n.d. Accessed December 23, 2016. http://lucene.apache.org/solr/.

"Apache Spark Officially Sets a New Record in Large-Scale Sorting." 2014. *Databricks* (blog). November 5, 2014. https://databricks.com/blog/2014/11/05/spark-officially-sets-a-new-record-in-large-scale-sorting.html.

"Apache Spark™ - Lightning-Fast Cluster Computing." n.d. Accessed December 22, 2016. http://spark.apache.org/.

"Apache Storm." n.d. Accessed January 27, 2017. http://storm.apache.org/.

Attigeri, G. V., Manohara Pai M. M, R. M. Pai, and A. Nayak. 2015. "Stock Market Prediction: A Big Data Approach." In *TENCON 2015 - 2015 IEEE Region 10 Conference*, 1–5. https://doi.org/10.1109/TENCON.2015.7373006.

Balahur, Alexandra, Ralf Steinberger, Mijail Kabadjov, Vanni Zavarella, Erik van der Goot, Matina Halkia, Bruno Pouliquen, and Jenya Belyaeva. 2013. "Sentiment Analysis in the News." *arXiv:1309.6202 [Cs]*, September. http://arxiv.org/abs/1309.6202.

Begoli, Edmon. 2012. "A Short Survey on the State of the Art in Architectures and Platforms for Large Scale Data Analysis and Knowledge Discovery from Data." In *Proceedings of the WICSA/ECSA 2012 Companion Volume*, 177–183. WICSA/ECSA '12. New York, NY, USA: ACM. https://doi.org/10.1145/2361999.2362039.

Bhargava, K., and R. Katarya. 2017. "An Improved Lexicon Using Logistic Regression for Sentiment Analysis." In *2017 International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN)*, 332–37. https://doi.org/10.1109/IC3TSN.2017.8284501.

Bifet, Albert, Geoffrey Holmes, and Bernhard Pfahringer. 2011. "MOA-TweetReader: Real-Time Analysis in Twitter Streaming Data." In , 46–60. https://doi.org/10.1007/978-3-642-24477-3_7.

Bloom, Burton H. 1970. "Space/Time Trade-Offs in Hash Coding with Allowable Errors." *Commun. ACM* 13 (7): 422–426. https://doi.org/10.1145/362686.362692.

Bollen, Johan, Huina Mao, and Xiao-Jun Zeng. 2011. "Twitter Mood Predicts the Stock Market." *Journal of Computational Science* 2 (1): 1–8. https://doi.org/10.1016/j.jocs.2010.12.007.

Boullier, Dominique, and Audrey Lohard. 2012. *Opinion mining et Sentiment analysis : Méthodes et outils*. Sciences Po | médialab. Marseille: OpenEdition Press. http://books.openedition.org/oep/198.

Bryson, Steve, David Kenwright, Michael Cox, David Ellsworth, and Robert Haimes. 1999. "Visually Exploring Gigabyte Data Sets in Real Time." *Commun. ACM* 42 (8): 82–90. https://doi.org/10.1145/310930.310977.

Burnap, Pete, Rachel Gibson, Luke Sloan, Rosalynd Southern, and Matthew Williams. 2016. "140 Characters to Victory?: Using Twitter to Predict the UK 2015 General Election." *Electoral Studies* 41 (March): 230–33. https://doi.org/10.1016/j.electstud.2015.11.017.

"Business Intelligence and Analytics | Tableau Software." n.d. Accessed April 14, 2018. https://www.tableau.com/.

Cambria, Erik, Tim Benson, Chris Eckl, and Amir Hussain. 2012. "Sentic PROMs: Application of Sentic Computing to the Development of a Novel Unified Framework for Measuring Health-Care Quality." *Expert Systems with Applications* 39 (12): 10533–43. https://doi.org/10.1016/j.eswa.2012.02.120.

Chen, Xiaoyu, Shuai Shao, Zhihua Tian, Zhen Xie, and Peng Yin. 2017. "Impacts of Air Pollution and Its Spatial Spillover Effect on Public Health Based on China's Big Data Sample." *Journal of Cleaner Production*, Special Volume on Improving natural resource management and human health to ensure sustainable societal development based upon insights gained from working within "Big Data Environments," 142, Part 2 (January): 915–25. https://doi.org/10.1016/j.jclepro.2016.02.119.

Chiavetta, F., G. Lo Bosco, and G. Pilato. 2016. "A Lexicon-Based Approach for Sentiment Classification of Amazon Books Reviews in Italian Language." In , 159–70. Scitepress. https://doi.org/10.5220/0005915301590170.

Chua, Alvin, Loris Servillo, Ernesto Marcheggiani, and Andrew Vande Moere. 2016. "Mapping Cilento: Using Geotagged Social Media Data to Characterize Tourist Flows in Southern Italy." *Tourism Management* 57 (December): 295–310. https://doi.org/10.1016/j.tourman.2016.06.013.

"Chukwa - Welcome to Apache Chukwa." n.d. Accessed December 22, 2016. http://chukwa.apache.org/.

Coffman, K. G., and A. M. Odlyzko. 2002. "Handbook of Massive Data Sets." In , edited by James Abello, Panos M. Pardalos, and Mauricio G. C. Resende, 47–93. Norwell, MA, USA: Kluwer Academic Publishers. http://dl.acm.org/citation.cfm?id=779232.779236.

Conover, M. D., B. Goncalves, J. Ratkiewicz, A. Flammini, and F. Menczer. 2011. "Predicting the Political Alignment of Twitter Users." In *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing*, 192–99. https://doi.org/10.1109/PASSAT/SocialCom.2011.34.

Corallo, Angelo, Laura Fortunato, Marco Matera, Marco Alessi, Alessio Camillò, Valentina Chetta, Enza Giangreco, and Davide Storelli. 2015. "Sentiment Analysis for Government: An Optimized Approach." In *Machine Learning and Data Mining in Pattern Recognition*, 98–112. Lecture Notes in Computer Science. Springer, Cham. https://doi.org/10.1007/978-3-319-21024-7_7.

Corbellini, Alejandro, Cristian Mateos, Alejandro Zunino, Daniela Godoy, and Silvia Schiaffino. 2017. "Persisting Big-Data: The NoSQL Landscape." *Information Systems* 63 (January): 1–23. https://doi.org/10.1016/j.is.2016.07.009.

Cox, Michael, and David Ellsworth. 1997. "Application-Controlled Demand Paging for Out-of-Core Visualization." In *Proceedings of the 8th Conference on Visualization '97*, 235–. VIS '97. Los Alamitos, CA, USA: IEEE Computer Society Press. http://dl.acm.org/citation.cfm?id=266989.267068.

D. Schneider. 2012. *Hadoop for Dummies*. IBM, John Wiley & Sons, Inc.

"Data Visualization & Analytics Software | TIBCO Spotfire." n.d. Accessed April 14, 2018. https://spotfire.tibco.com/.

Dean, Jeffrey, and Sanjay Ghemawat. 2004. "MapReduce: Simplified Data Processing on Large Clusters." *OSDI'04: Sixth Symposium on Operating System Design and Implementation* 51 (1): 107–113. https://doi.org/10.1145/1327452.1327492.

Dede, Elif, Madhusudhan Govindaraju, Daniel Gunter, Richard Shane Canon, and Lavanya Ramakrishnan. 2013. "Performance Evaluation of a MongoDB and Hadoop Platform for Scientific Data Analysis." In *Proceedings of the 4th ACM Workshop on Scientific Cloud Computing*, 13–20. Science Cloud '13. New York, NY, USA: ACM.

https://doi.org/10.1145/2465848.2465849.

Delenn Chin, Jessica Zhao, and Anna Zappone. 2016. “Analyzing Twitter Sentiment of the 2016 Presidential Candidates.”

Deng, L., J. Gao, and C. Vuppalapati. 2015. “Building a Big Data Analytics Service Framework for Mobile Advertising and Marketing.” In *2015 IEEE First International Conference on Big Data Computing Service and Applications*, 256–66. https://doi.org/10.1109/BigDataService.2015.27.

DiGrazia, Joseph, Karissa McKelvey, Johan Bollen, and Fabio Rojas. 2013. “More Tweets, More Votes: Social Media as a Quantitative Indicator of Political Behavior.” *PLOS ONE* 8 (11): e79449. https://doi.org/10.1371/journal.pone.0079449.

Ding, Xiaowen, Bing Liu, and Philip S. Yu. 2008. “A Holistic Lexicon-Based Approach to Opinion Mining.” In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, 231–240. WSDM ’08. New York, NY, USA: ACM. https://doi.org/10.1145/1341531.1341561.

Domingos, Pedro, and Michael Pazzani. 1997. “On the Optimality of the Simple Bayesian Classifier under Zero-One Loss.” *Machine Learning* 29 (2–3): 103–30. https://doi.org/10.1023/A:1007413511361.

Dong, T., B. Yang, and T. Tian. 2015. “Volatility Analysis of Chinese Stock Market Using High-Frequency Financial Big Data.” In *2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*, 769–74. https://doi.org/10.1109/SmartCity.2015.234.

Dredze, M. 2012. “How Social Media Will Change Public Health.” *IEEE Intelligent Systems* 27 (4): 81–84. https://doi.org/10.1109/MIS.2012.76.

Duwairi, Rehab, Raed Marji, Narmeen Sha’ban, and Sally Rushaidat. 2014. “Sentiment Analysis in Arabic Tweets.” In *2014 5th International Conference on Information and Communication Systems, ICICS 2014*, 1–6. https://doi.org/10.1109/IACS.2014.6841964.

Duyu, Tang, Furu, Wei, Nan, Yang, Ming, Zhou, Ting, Liu, and Bing, Qin. 2014. “Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification.” In . http://anthology.aclweb.org/P/P14/P14-1146.xhtml.

Edison, M., and A. Aloysius. 2017. “Polarity Detection of Lexicon Based Sentiment Analysis with Negation Handling.” *Journal of Advanced Research in Dynamical and Control Systems* 9 (Special Issue 13): 44–54.

“Elastic - Home.” n.d. Accessed December 23, 2016. https://www.elastic.co/fr/.

Elmasry, Mostafa, Taysir Soliman, and Abdel-Rahman Hedar. 2014. “Sentiment Analysis of Arabic Slang Comments on Facebook.” *International Journal of Computers and Technology (IJCT)* 12 (January): 3470–78.

Ene, Alina, Sungjin Im, and Benjamin Moseley. 2011. “Fast Clustering Using MapReduce.” In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 681–689. KDD ’11. New York, NY, USA: ACM. https://doi.org/10.1145/2020408.2020515.

Er, M.J., F. Liu, N. Wang, Y. Zhang, and M. Pratama. 2016. “User-Level Twitter Sentiment Analysis with a Hybrid Approach.” *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 9719: 426–33. https://doi.org/10.1007/978-3-319-40663-3_49.

Esuli, Andrea, and Fabrizio Sebastiani. 2006. “SENTIWORDNET: A Publicly Available Lexical Resource for Opinion Mining.” In *In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC’06*, 417–422.

Finn, Samantha, Eni Mustafaraj, and P. Metaxas. 2014. “The Co-Retweeted Network and Its Applications for Measuring the Perceived Political Polarization.” *Faculty Research and Scholarship*, January. https://doi.org/10.5220/0004788702760284.

Fléchaux, Reynald. 2015. “Hadoop Sur Le Déclin ? Cloudera, Hortonworks et MapR Répondent.” Silicon. June 2, 2015. http://www.silicon.fr/hadoop-declin-cloudera-hortonworks-mapr-repondent-117774.html.

Franch, Fabio. 2013. “(Wisdom of the Crowds)2: 2010 UK Election Prediction with Social Media.” *Journal of Information Technology & Politics* 10 (1): 57–71. https://doi.org/10.1080/19331681.2012.705080.

Gamallo, Pablo, and Marcos Garcia. 2014. “Citius: A Naive-Bayes Strategy for Sentiment Analysis on English Tweets.” In , 171–75. https://doi.org/10.3115/v1/S14-2026.

García, Salvador, Julián Luengo, and Francisco Herrera. 2016. “Tutorial on Practical Tips of the Most Influential Data Preprocessing Algorithms in Data Mining.” *Knowledge-Based Systems* 98 (April): 1–29. https://doi.org/10.1016/j.knosys.2015.12.006.

Gaspar, Rui, Cláudia Pedro, Panos Panagiotopoulos, and Beate Seibt. 2016. “Beyond Positive or Negative: Qualitative Sentiment Analysis of Social Media Reactions to Unexpected Stressful Events.” *Computers in Human Behavior* 56 (Supplement C): 179–91. https://doi.org/10.1016/j.chb.2015.11.040

ayo-Avello, D. 2012. “No, You Cannot Predict Elections with Twitter.” *IEEE Internet Computing* 16 (6): 91–94. https://doi.org/10.1109/MIC.2012.13

Ghannam, Jeffrey. 2011. “Digital Media in the Arab World One Year After the Revolutions.” Center for International Media.

Ghemawat, Sanjay, Howard Gobioff, and Shun-Tak Leung. 2003. “The Google File System.” In *Proceedings of the Nineteenth ACM Symposium on Operating Systems Principles*, 29–43. SOSP ’03. New York, NY, USA: ACM. https://doi.org/10.1145/945445.945450.

Go, Alec, Richa Bhayani, and Lei Huang. 2009. “Twitter Sentiment Classification Using Distant Supervision.” *Processing* 150 (January).

Goli-Malekabadi, Zohreh, Morteza Sargolzaei-Javan, and Mohammad Kazem Akbari. 2016. “An Effective Model for Store and Retrieve Big Health Data in Cloud Computing.” *Computer Methods and Programs in Biomedicine* 132 (August): 75–82. https://doi.org/10.1016/j.cmpb.2016.04.016.

Gonçalves, Pollyanna, Matheus Araújo, Fabrício Benevenuto, and Meeyoung Cha. 2013. “Comparing and Combining Sentiment Analysis Methods.” In *Proceedings of the First ACM Conference on Online Social Networks*, 27–38. COSN ’13. New York, NY, USA: ACM. https://doi.org/10.1145/2512938.2512951.

Gong, Y., L. Morandini, and R. O. Sinnott. 2017. “The Design and Benchmarking of a Cloud-Based Platform for Processing and Visualization of Traffic Data.” In *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*, 13–20. https://doi.org/10.1109/BIGCOMP.2017.7881699.

"Google Trends." n.d. Google Trends. Accessed February 23, 2017. https://g.co/trends/aes0h.

"Guide D'architecture HDFS." n.d. Accessed January 27, 2017. https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html#Introduction.

Gulzar, Muhammad Ali, Matteo Interlandi, Tyson Condie, and Miryung Kim. 2016. "BigDebug: Interactive Debugger for Big Data Analytics in Apache Spark." In *Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering*, 1033–1037. FSE 2016. New York, NY, USA: ACM. https://doi.org/10.1145/2950290.2983930.

Gupta, S., and P. Tripathi. 2016. "An Emerging Trend of Big Data Analytics with Health Insurance in India." In *2016 International Conference on Innovation and Challenges in Cyber Security (ICICCS-INBUSH)*, 64–69. https://doi.org/10.1109/ICICCS.2016.7542360.

"H2O.ai." n.d. Accessed March 10, 2018. https://www.h2o.ai/.

Han, Jiawei, Micheline Kamber, and Jian Pei. 2012a. "1 - Introduction." In *Data Mining (Third Edition)*, 1–38. The Morgan Kaufmann Series in Data Management Systems. Boston: Morgan Kaufmann. https://doi.org/10.1016/B978-0-12-381479-1.00001-0.

Hansen, Lars Kai, Adam Arvidsson, Finn Aarup Nielsen, Elanor Colleoni, and Michael Etter. 2011. "Good Friends, Bad News - Affect and Virality in Twitter." In *Future Information Technology*, 34–43. Communications in Computer and Information Science. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-22309-9_5.

Harous, Saad, Mohamed El Menshawy, Mohamed Adel Serhani, and Abdelghani Benharref. 2018. "Mobile Health Architecture for Obesity Management Using Sensory and Social Data." *Informatics in Medicine Unlocked* 10: 27–44. https://doi.org/10.1016/j.imu.2017.12.005.

Hashem, Ibrahim Abaker Targio, Ibrar Yaqoob, Nor Badrul Anuar, Salimah Mokhtar, Abdullah Gani, and Samee Ullah Khan. 2015. "The Rise of 'big Data' on Cloud Computing: Review and Open Research Issues." *Information Systems* 47 (January): 98–115. https://doi.org/10.1016/j.is.2014.07.006.

Hatzivassiloglou, Vasileios, and Kathleen R. McKeown. 1997. "Predicting the Semantic Orientation of Adjectives." In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, 174–181. ACL '98. Stroudsburg, PA, USA: Association for Computational Linguistics. https://doi.org/10.3115/976909.979640.

Hatzivassiloglou, Vasileios, and Janyce M. Wiebe. 2000. "Effects of Adjective Orientation and Gradability on Sentence Subjectivity." In *Proceedings of the 18th Conference on Computational Linguistics - Volume 1*, 299–305. COLING '00. Stroudsburg, PA, USA: Association for Computational Linguistics. https://doi.org/10.3115/990820.990864.

He, Wu, Harris Wu, Gongjun Yan, Vasudeva Akula, and Jiancheng Shen. 2015. "A Novel Social Media Competitive Analytics Framework with Sentiment Benchmarks." *Inf. Manage.* 52 (7): 801–812. https://doi.org/10.1016/j.im.2015.04.006.

He, Wu, Shenghua Zha, and Ling Li. 2013. "Social Media Competitive Analysis and Text Mining: A Case Study in the Pizza Industry." *International Journal of Information Management* 33 (3): 464–72.

https://doi.org/10.1016/j.ijinfomgt.2013.01.001.

Hesse, G., and M. Lorenz. 2015. "Conceptual Survey on Data Stream Processing Systems." In *2015 IEEE 21st International Conference on Parallel and Distributed Systems (ICPADS)*, 797–802. https://doi.org/10.1109/ICPADS.2015.106.

Ho, K. F., H. W. Hirai, Y. H. Kuo, H. M. Meng, and K. K. F. Tsoi. 2015. "Indoor Air Monitoring Platform and Personal Health Reporting System: Big Data Analytics for Public Health Research." In *2015 IEEE International Congress on Big Data*, 309–12. https://doi.org/10.1109/BigDataCongress.2015.51.

Hochstetler, J., L. Hochstetler, and S. Fu. 2016. "An Optimal Police Patrol Planning Strategy for Smart City Safety." In *2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, 1256–63. https://doi.org/10.1109/HPCC-SmartCity-DSS.2016.0178.

"Hortonworks." 2011. 2011. http://docs.hortonworks.com/index.html.

Hu, Minqing, and Bing Liu. 2004a. "Mining and Summarizing Customer Reviews." In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 168–177. KDD '04. New York, NY, USA: ACM. https://doi.org/10.1145/1014052.1014073

Hung, Jui-long, and Ke Zhang. 2008. "Revealing Online Learning Behaviors and Activity Patterns and Making Predictions with Data Mining Techniques in Online Teaching." *MERLOT Journal of Online Learning and Teaching*, December. http://scholarworks.boisestate.edu/edtech_facpubs/4.

"Internet Live Stats." n.d. Accessed December 30, 2017. http://www.internetlivestats.com/.

J. S. Brownstein, C. C. Freifeld, Ben Y. Reis, and K. D. Mandl. 1998. "Glossary of Terms." *Mach. Learn.* 30 (2–3): 271–274.

Jaap, Kamps, Maarten, Marx, Robert J., Mokken, and Maarten, de Rijke. 2004. "Using WordNet to Measure Semantic Orientations of Adjectives (PDF Download Available)." *LREC* IV: 1115–1118.

Jacquemin, C., and Bush. 2000. "Fouille Du Web Pour La Collecte D'entités Nommées." *Actes de La 8ème Conférence Nationale Sur Le Traitement Automatique Des Langues Naturelles (TALN 2000)*.

Jagtap, A., B. Bodkhe, B. Gaikwad, and S. Kalyana. 2016. "Homogenizing Social Networking with Smart Education by Means of Machine Learning and Hadoop: A Case Study." In *2016 International Conference on Internet of Things and Applications (IOTA)*, 85–90. https://doi.org/10.1109/IOTA.2016.7562700.

Jahanbakhsh, Kazem, and Yumi Moon. 2014. "The Predictive Power of Social Media: On the Predictability of U.S. Presidential Elections Using Twitter." *arXiv:1407.0622 [Physics]*, July. http://arxiv.org/abs/1407.0622.

Jain, Vinay Kumar, and Shishir Kumar. 2017. "Effective Surveillance and Predictive Mapping of Mosquito-Borne Diseases Using Social Media." *Journal of Computational Science*, July. https://doi.org/10.1016/j.jocs.2017.07.003.

Jeffrey, Dean, and Sanjay, Ghemawat. 2004. "The Google File System." https://research.google.com/archive/mapreduce.html.

Jin, Wei, Hay Ho, Hung, and K. Srihari, Rohini. 2009. “OpinionMiner: A Novel Machine Learning System for Web Opinion Mining and Extraction (PDF Download Available).” In , 1195–120.

Jose, R., and V. S. Chooralil. 2016. “Prediction of Election Result by Enhanced Sentiment Analysis on Twitter Data Using Classifier Ensemble Approach.” In *2016 International Conference on Data Mining and Advanced Computing (SAPIENCE)*, 64–67. https://doi.org/10.1109/SAPIENCE.2016.7684133.

K. G., Coffman, and Odlyzko Andrew. 1999. “The Size and Growth Rate of the Internet.” Center for Discrete Mathematics & Theoretical Computer Science.

Kashyap, R., and A. Nahapetian. 2014. “Tweet Analysis for User Health Monitoring.” In *2014 4th International Conference on Wireless Mobile Communication and Healthcare - Transforming Healthcare Through Innovations in Mobile and Wireless Technologies (MOBIHEALTH)*, 348–51. https://doi.org/10.1109/MOBIHEALTH.2014.7015983.

Katal, A., M. Wazid, and R. H. Goudar. 2013. “Big Data: Issues, Challenges, Tools and Good Practices.” In *2013 Sixth International Conference on Contemporary Computing (IC3)*, 404–9. https://doi.org/10.1109/IC3.2013.6612229.

Kessler, Jason, and N Nicolov. 2009. “Targeting Sentiment Expressions through Supervised Ranking of Linguistic Configurations.” *Proceedings of the Third International AAAI Conference on Weblogs and Social Media*, January, 90–97.

Khan, Farhan Hassan, Usman Qamar, and Saba Bashir. 2016. “SWIMS: Semi-Supervised Subjective Feature Weighting and Intelligent Model Selection for Sentiment Analysis.” *Knowledge-Based Systems* 100 (May): 97–111. https://doi.org/10.1016/j.knosys.2016.02.011.

Khatua, Aparup, Apalak Khatua, Kuntal Ghosh, and Nabendu Chaki. 2015. “Can #Twitter_Trends Predict Election Results? Evidence from 2014 Indian General Election.” In *Proceedings of the 2015 48th Hawaii International Conference on System Sciences*, 1676–1685. HICSS ’15. Washington, DC, USA: IEEE Computer Society. https://doi.org/10.1109/HICSS.2015.202.

Kim, Kun, Oun-joung Park, Seunghyun Yun, and Haejung Yun. 2017. “What Makes Tourists Feel Negatively about Tourism Destinations? Application of Hybrid Text Mining Methodology to Smart Destination Management.” *Technological Forecasting and Social Change* 123 (Supplement C): 362–69. https://doi.org/10.1016/j.techfore.2017.01.001.

Kim, Soo-Min, and Eduard Hovy. 2006. “Automatic Identification of Pro and Con Reasons in Online Reviews.” In *Proceedings of the COLING/ACL on Main Conference Poster Sessions*, 483–490. COLING-ACL ’06. Stroudsburg, PA, USA: Association for Computational Linguistics. http://dl.acm.org/citation.cfm?id=1273073.1273136.

Kucuktunc, Onur, B. Barla Cambazoglu, Ingmar Weber, and Hakan Ferhatosmanoglu. 2012. “A Large-Scale Sentiment Analysis for Yahoo! Answers.” In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*, 633–642. WSDM ’12. New York, NY, USA: ACM. https://doi.org/10.1145/2124295.2124371.

Kuhamanee, T., N. Talmongkol, K. Chaisuriyakul, W. San-Um, N. Pongpisuttinun, and S. Pongyupinpanich. 2017. “Sentiment Analysis of Foreign Tourists to Bangkok Using Data Mining through Online Social Network.” In

2017 IEEE 15th International Conference on Industrial Informatics (INDIN), 1068–73. https://doi.org/10.1109/INDIN.2017.8104921.

Kumar, K. M., Tejasree S, and S. Swarnalatha. 2016. "Effective Implementation of Data Segregation Extraction Using Big Data in E - Health Insurance as a Service." In *2016 3rd International Conference on Advanced Computing and Communication Systems (ICACCS)*, 1:1–5. https://doi.org/10.1109/ICACCS.2016.7586323.

Kumar, P., K. Manocha, and H. Gupta. 2016. "Enterprise Analysis through Opinion Mining." In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, 3318–23. https://doi.org/10.1109/ICEEOT.2016.7755320.

Lailiyah, M., S. Sumpeno, and I. K. E. Purnama. 2017. "Sentiment Analysis of Public Complaints Using Lexical Resources between Indonesian Sentiment Lexicon and Sentiwordnet." In *2017 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, 307–12. https://doi.org/10.1109/ISITIA.2017.8124100.

Landset, Sara, Taghi M. Khoshgoftaar, Aaron N. Richter, and Tawfiq Hasanin. 2015. "A Survey of Open Source Tools for Machine Learning with Big Data in the Hadoop Ecosystem." *Journal of Big Data* 2 (1): 24. https://doi.org/10.1186/s40537-015-0032-1.

Laney, D. 2001. *3D Data Management: Controlling Data Volume, Velocity, and Variety*. https://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf.

Li, Hu, Tianjia Chen, and Wei Xu. 2016. "Improving Spark Performance with Zero-Copy Buffer Management and RDMA." In *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 33–38. https://doi.org/10.1109/INFCOMW.2016.7562041.

Li, J., Z. Xu, Y. Jiang, and R. Zhang. 2014. "The Overview of Big Data Storage and Management." In *2014 IEEE 13th International Conference on Cognitive Informatics and Cognitive Computing*, 510–13. https://doi.org/10.1109/ICCI-CC.2014.6921508.

Lim, Sunghoon, Conrad S. Tucker, and Soundar Kumara. 2017. "An Unsupervised Machine Learning Model for Discovering Latent Infectious Diseases Using Social Media Data." *Journal of Biomedical Informatics* 66 (Supplement C): 82–94. https://doi.org/10.1016/j.jbi.2016.12.007.

Liu, Bin, Shu Gui Cao, and Wu He. 2011. "Distributed Data Mining for E-Business." *Inf. Technol. and Management* 12 (2): 67–79. https://doi.org/10.1007/s10799-011-0091-8.

Liu, Xiufeng, Nadeem Iftikhar, and Xike Xie. 2014. "Survey of Real-Time Processing Systems for Big Data." In *Proceedings of the 18th International Database Engineering & Applications Symposium*, 356–361. IDEAS '14. New York, NY, USA: ACM. https://doi.org/10.1145/2628194.2628251.

Livne, Avishay, Matthew Simmons, Eytan Adar, and Lada Adamic. 2011. "The Party Is Over Here: Structure and Content in the 2010 Election." In *Fifth International AAAI Conference on Weblogs and Social Media*. https://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/view/2852.

Lopez, M. A., A. G. P. Lobato, and O. C. M. B. Duarte. 2016. "A Performance Comparison of Open-Source Stream Processing Platforms." In *2016 IEEE Global Communications Conference (GLOBECOM)*, 1–6.

https://doi.org/10.1109/GLOCOM.2016.7841533.

Lu, R., G. Wu, B. Xie, and J. Hu. 2014. "Stream Bench: Towards Benchmarking Modern Distributed Stream Computing Frameworks." In *2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing*, 69–78. https://doi.org/10.1109/UCC.2014.15.

Luo, Zhunchen, Miles Osborne, and Ting Wang. 2015. "An Effective Approach to Tweets Opinion Retrieval." *World Wide Web* 18 (3): 545–66. https://doi.org/10.1007/s11280-013-0268-7.

Ma, Y., Y. Wang, J. Yang, Y. Miao, and W. Li. 2016. "Big Health Application System Based on Health Internet of Things and Big Data." *IEEE Access* PP (99): 1–1. https://doi.org/10.1109/ACCESS.2016.2638449.

Mahmood, T., T. Iqbal, F. Amin, W. Lohanna, and A. Mustafa. 2013. "Mining Twitter Big Data to Predict 2013 Pakistan Election Winner." In *INMIC*, 49–54. https://doi.org/10.1109/INMIC.2013.6731323.

Manologlou, E, P Tsartas, and A Markou. 2004. "Geothermal Energy Sources for Water Production—socio-Economic Effects and People's Wishes on Milos Island: A Case Study." *Energy Policy* 32 (5): 623–33. https://doi.org/10.1016/S0301-4215(02)00315-4.

"MapReduce Tutorial." n.d. Accessed January 27, 2017. https://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html.

Marron, B. A., and P. A. D. de Maine. 1967. "Automatic Data Compression." *Commun. ACM* 10 (11): 711–715. https://doi.org/10.1145/363790.363813.

Mazumder, Sourav. 2016. "Big Data Tools and Platforms." In *Big Data Concepts, Theories, and Applications*, edited by Shui Yu and Song Guo, 29–128. Springer International Publishing. https://doi.org/10.1007/978-3-319-27763-9_2.

Mertiya, M., and A. Singh. 2016. "Combining Naive Bayes and Adjective Analysis for Sentiment Detection on Twitter." In *2016 International Conference on Inventive Computation Technologies (ICICT)*, 2:1–6. https://doi.org/10.1109/INVENTIVE.2016.7824847.

Miguel, J., S. Caballé, and F. Xhafa. 2015. "A MapReduce Approach for Processing Student Data Activity in a Peer-to-Peer Networked Setting." In *2015 10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC)*, 9–16. https://doi.org/10.1109/3PGCIC.2015.27.

"MINTS - Minnesota Internet Traffic Studies." n.d. Accessed January 19, 2017. http://www.dtc.umn.edu/mints/home.php.

"MLlib | Apache Spark." n.d. Accessed February 15, 2017. http://spark.apache.org/mllib/.

Mohamed, N., and J. Al-Jaroodi. 2014. "Real-Time Big Data Analytics: Applications and Challenges." In *2014 International Conference on High Performance Computing Simulation (HPCS)*, 305–10. https://doi.org/10.1109/HPCSim.2014.6903700.

Mohammed, A. F., V. T. Humbe, and S. S. Chowhan. 2016. "A Review of Big Data Environment and Its Related Technologies." In *2016 International Conference on Information Communication and Embedded Systems (ICICES)*, 1–5. https://doi.org/10.1109/ICICES.2016.7518904.

Moniruzzaman, A. B. M., and Syed Akhter Hossain. 2013. "NoSQL Database: New Era of Databases for Big Data

Analytics - Classification, Characteristics and Comparison." *arXiv:1307.0191 [Cs]*, June. http://arxiv.org/abs/1307.0191.

Montejo-Ráez, Arturo, Eugenio Martínez-Cámara, Maria Martín-Valdivia, and L López. 2012. "Random Walk Weighting over SentiWordNet for Sentiment Polarity Detection on Twitter." In *Proc 3rd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, 3–10.

Mubarok, Mohamad Syahrul, Adiwijaya, and Muhammad Dwi Aldhi. 2017. "Aspect-Based Sentiment Analysis to Review Products Using Naïve Bayes." *AIP Conference Proceedings* 1867 (1): 20060. https://doi.org/10.1063/1.4994463.

Mukwazvure, A., and K. P. Supreethi. 2015. "A Hybrid Approach to Sentiment Analysis of News Comments." In *2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions)*, 1–6. https://doi.org/10.1109/ICRITO.2015.7359282.

Musto, Cataldo, Giovanni Semeraro, Marco de Gemmis, and Pasquale Lops. 2015. "Developing Smart Cities Services Through Semantic Analysis of Social Streams." In *Proceedings of the 24th International Conference on World Wide Web*, 1401–1406. WWW '15 Companion. New York, NY, USA: ACM. https://doi.org/10.1145/2740908.2742132.

Nagy, Ahmed, and Jeannie Stamberger. 2012. "Crowd Sentiment Detection during Disasters and Crises," January.

Nair, Lekha R., Sujala D. Shetty, and Siddhanth D. Shetty. 2018. "Applying Spark Based Machine Learning Model on Streaming Big Data for Health Status Prediction." *Computers & Electrical Engineering*. https://doi.org/10.1016/j.compeleceng.2017.03.009.

Neumann, Saggi. n.d. "Spark vs. Hadoop MapReduce." Xplenty. Accessed December 23, 2016. https://www.xplenty.com/blog/2014/11/apache-spark-vs-hadoop-mapreduce/.

"Newspapers.com - Historical Newspapers from 1700s-2000s." n.d. Newspapers.com. Accessed December 23, 2016. http://www.newspapers.com/.

Nirmal, V. J., and D. I. G. Amalarethinam. 2017. "Real-Time Sentiment Prediction on Streaming Social Network Data Using In-Memory Processing." In *2017 World Congress on Computing and Communication Technologies (WCCCT)*, 69–72. https://doi.org/10.1109/WCCCT.2016.26.

Nodarakis, Nikolaos, Spyros Sioutas, Athanasios Tsakalidis, and Giannis Tzimas. 2016. "Large Scale Sentiment Analysis on Twitter with Spark." In .

Nuaimi, Eiman Al, Hind Al Neyadi, Nader Mohamed, and Jameela Al-Jaroodi. 2015. "Applications of Big Data to Smart Cities." *Journal of Internet Services and Applications* 6 (1): 25. https://doi.org/10.1186/s13174-015-0041-5.

O'Connor, Brendan, Ramnath Balasubramanyan, Bryan Routledge, and Noah Smith. 2010. "From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series." *Proceedings of the International AAAI Conference on Weblogs and Social Media, Washington, DC, May 2010*, January. http://repository.cmu.edu/tepper/559.

"Open Database Connectivity." 2017. *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Open_Database_Connectivity&oldid=758111801.

Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan. 2002a. “Thumbs Up?: Sentiment Classification Using Machine Learning Techniques.” In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, 79–86. EMNLP ’02. Stroudsburg, PA, USA: Association for Computational Linguistics. https://doi.org/10.3115/1118693.1118704.

Park, G., Sooyong Park, Latifur Khan, and L. Chung. 2017. “IRIS: A Goal-Oriented Big Data Analytics Framework on Spark for Better Business Decisions.” In *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*, 76–83. https://doi.org/10.1109/BIGCOMP.2017.7881719.

Philip Chen, C. L., and Chun-Yang Zhang. 2014. “Data-Intensive Applications, Challenges, Techniques and Technologies: A Survey on Big Data.” *Information Sciences* 275 (August): 314–47. https://doi.org/10.1016/j.ins.2014.01.015.

Piyush, Bhardwaj, Suruchi, Gautam, and Payal Pahwa. 2017. “Opinion Mining and Sentiment Analysis of Travel Websites through Twitter.” *International Journal of Applied Engineering Research* 12: 12431–39.

“Power BI | Outils Décisionnels de Visualisation Interactive Des Données.” n.d. Accessed April 14, 2018. https://powerbi.microsoft.com/fr-fr/.

Press, Gil. 2013. “A Very Short History Of Big Data.” Forbes. 2013. http://www.forbes.com/sites/gilpress/2013/05/09/a-very-short-history-of-big-data/.

Price, Derek de Solla. 1961. *Science Since Babylon*. Yale University Press.

Qiu, Junfei, Qihui Wu, Guoru Ding, Yuhua Xu, and Shuo Feng. 2016. “A Survey of Machine Learning for Big Data Processing.” *EURASIP Journal on Advances in Signal Processing* 2016 (1): 67. https://doi.org/10.1186/s13634-016-0355-x.

Raghothama, Jayanth, Vinutha Magal Shreenath, and Sebastiaan Meijer. 2016. “Analytics on Public Transport Delays with Spatial Big Data.” In *Proceedings of the 5th ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data*, 28–33. BigSpatial ’16. New York, NY, USA: ACM. https://doi.org/10.1145/3006386.3006387.

Ragkos, Athanasios, Alexandros Theodoridis, and Christos Batzios. 2015. “Public Awareness Concerning the Multifunctionality of Cypriot Agriculture.” *Agriculture and Agricultural Science Procedia*, Efficient irrigation management and its effects in urban and rural landscapes, 4 (Supplement C): 147–57. https://doi.org/10.1016/j.aaspro.2015.03.018.

Ramanathan, V., and T. Meyyappan. 2013. “Survey of Text Mining.” *International Conference on Technology and Business and Management*, March, 508–14.

Ramteke, J., S. Shah, D. Godhia, and A. Shaikh. 2016. “Election Result Prediction Using Twitter Sentiment Analysis.” In *2016 International Conference on Inventive Computation Technologies (ICICT)*, 1:1–5. https://doi.org/10.1109/INVENTIVE.2016.7823280.

Rasyid, M. U. H. A., W. Yuwono, S. A. Muharom, and A. H. Alasiry. 2016. “Building Platform Application Big Sensor Data for E-Health Wireless Body Area Network.” In *2016 International Electronics Symposium (IES)*, 409–13. https://doi.org/10.1109/ELECSYM.2016.7861041.

Rathore, M. M., A. Ahmad, A. Paul, and G. Jeon. 2015. "Efficient Graph-Oriented Smart Transportation Using Internet of Things Generated Big Data." In *2015 11th International Conference on Signal-Image Technology Internet-Based Systems (SITIS)*, 512–19. https://doi.org/10.1109/SITIS.2015.121.

Ravi, Kumar. 2015. "A Survey on Opinion Mining and Sentiment Analysis: Tasks, Approaches and Applications." *Knowledge-Based Systems* 89 (June): 14–46. https://doi.org/10.1016/j.knosys.2015.06.015.

Razzaq, M. A., A. M. Qamar, and Hafiz Syed Muhammad Bilal. 2014. "Prediction and Analysis of Pakistan Election 2013 Based on Sentiment Analysis." In *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014)*, 700–703. https://doi.org/10.1109/ASONAM.2014.6921662.

Rehman, Z.U., and I.S. Bajwa. 2017. "Lexicon-Based Sentiment Analysis for Urdu Language." In , 497–501. https://doi.org/10.1109/INTECH.2016.7845095.

Reinsel, David, and John Gantz. 2011. "Extracting Value from Chaos." Dell. https://www.emc.com/.

Richter, A. N., T. M. Khoshgoftaar, S. Landset, and T. Hasanin. 2015. "A Multi-Dimensional Comparison of Toolkits for Machine Learning with Big Data." In *2015 IEEE International Conference on Information Reuse and Integration*, 1–8. https://doi.org/10.1109/IRI.2015.12.

Riel, Arthur J., Denisa Popescu, and Luisita Guanlao. 2014. "Social Data Mining and Knowledge Flows Between Government and Its Citizenry in Crisis and Normal Situations." In *Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics (WIMS14)*, 56:1–56:5. WIMS '14. New York, NY, USA: ACM. https://doi.org/10.1145/2611040.2611090.

Rish, I. 2001. "An Empirical Study of the Naive Bayes Classifier."

Ritterman, Joshua, Miles Osborne, and Ewan Klein. 2009. "Using Prediction Markets and Twitter to Predict a Swine

Rodrigues, Ramon Gouveia, Rafael Marques das Dores, Celso G. Camilo-Junior, and Thierson Couto Rosa. 2016. "SentiHealth-Cancer: A Sentiment Analysis Tool to Help Detecting Mood of Patients in Online Social Networks." *International Journal of Medical Informatics* 85 (1): 80–95. https://doi.org/10.1016/j.ijmedinf.2015.09.007.

Saif, Hassan, Yulan He, and Harith Alani. 2012. "Semantic Sentiment Analysis of Twitter." In *Proceedings of the 11th International Conference on The Semantic Web - Volume Part I*, 508–524. ISWC'12. Berlin, Heidelberg: Springer-Verlag. https://doi.org/10.1007/978-3-642-35176-1_32.

Salas, A., P. Georgakis, C. Nwagboso, A. Ammari, and I. Petalas. 2017. "Traffic Event Detection Framework Using Social Media." In *2017 IEEE International Conference on Smart Grid and Smart Cities (ICSGSC)*, 303–7. https://doi.org/10.1109/ICSGSC.2017.8038595.

Saldarriaga, Sebastián Peña. 2010. "Approches textuelles pour la catégorisation et la recherche de documents manuscrits en-ligne." Phdthesis, Université de Nantes. https://tel.archives-ouvertes.fr/tel-00483684/document.

"Samza - Documentation." n.d. Accessed February 2, 2017. http://samza.apache.org/learn/documentation/0.11/.

Seixas, J. L., S. Barbon, and R. G. Mantovani. 2015. "Pattern Recognition of Lower Member Skin Ulcers in Medical Images with Machine Learning Algorithms." In *2015 IEEE 28th International Symposium on Computer-Based*

Medical Systems, 50–53. https://doi.org/10.1109/CBMS.2015.48.

Shi, Lei, Neeraj Agarwal, Ankur Agrawal, Rahul Garg, and Jacob Spoelstra. 2012. *Predicting US Primary Elections with Twitter*.

Shimada, K., S. Inoue, H. Maeda, and T. Endo. 2011. "Analyzing Tourism Information on Twitter for a Local City." In *2011 First ACIS International Symposium on Software and Network Engineering*, 61–66. https://doi.org/10.1109/SSNE.2011.27.

Siddiqa, Aisha, Ibrahim Abaker Targio Hashem, Ibrar Yaqoob, Mohsen Marjani, Shahabuddin Shamshirband, Abdullah Gani, and Fariza Nasaruddin. 2016. "A Survey of Big Data Management: Taxonomy and State-of-the-Art." *Journal of Network and Computer Applications* 71 (August): 151–66. https://doi.org/10.1016/j.jnca.2016.04.008.

Skourletopoulos, Georgios, Constandinos X. Mavromoustakis, George Mastorakis, Jordi Mongay Batalla, Ciprian Dobre, Spyros Panagiotakis, and Evangelos Pallis. 2017. "Big Data and Cloud Computing: A Survey of the State-of-the-Art and Research Challenges." In *Advances in Mobile Cloud Computing and Big Data in the 5G Era*, edited by Constandinos X. Mavromoustakis, George Mastorakis, and Ciprian Dobre, 23–41. Studies in Big Data 22. Springer International Publishing. https://doi.org/10.1007/978-3-319-45145-9_2.

Smailović, J., J. Kranjc, M. Grčar, M. Žnidaršič, and I. Mozetič. 2015. "Monitoring the Twitter Sentiment during the Bulgarian Elections." In *2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, 1–10. https://doi.org/10.1109/DSAA.2015.7344886.

Sodanil, Maleerat. 2016. "Multi-Language Sentiment Analysis for Hotel Reviews." *MATEC Web of Conferences* 75 (January): 3002. https://doi.org/10.1051/matecconf/20167503002.

Soler, J. M., F. Cuartero, and M. Roblizo. 2012. "Twitter as a Tool for Predicting Elections Results." In *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, 1194–1200. https://doi.org/10.1109/ASONAM.2012.206.

"Spark Streaming | Apache Spark." n.d. Accessed December 22, 2016. http://spark.apache.org/streaming/.

Spiliotis, E., G. Anastasopoulos, P. Dede, V. Marinakis, and H. Doukas. 2015. "A Framework for Integrating User Experience in Action Plan Evaluation through Social Media: Transforming User Generated Content into Knowledge to Optimise Energy Use in Buildings." In *2015 6th International Conference on Information, Intelligence, Systems and Applications (IISA)*, 1–6. https://doi.org/10.1109/IISA.2015.7388128.

"Sqoop -." n.d. Accessed December 22, 2016. http://sqoop.apache.org/.

Suguna, S., M. Vithya, and J. I. C. Eunaicy. 2016. "Big Data Analysis in E-Commerce System Using HadoopMapReduce." In *2016 International Conference on Inventive Computation Technologies (ICICT)*, 2:1–6. https://doi.org/10.1109/INVENTIVE.2016.7824798.

Suthaharan, Shan. 2014. "Big Data Classification: Problems and Challenges in Network Intrusion Prediction with Machine Learning." *SIGMETRICS Perform. Eval. Rev.* 41 (4): 70–73. https://doi.org/v.

T. White. 2012. *Hadoop: The Definitive Guide, 3rd Edition*. O'Reilly Media, Inc.

Ta, Van-Dai, Chuan-Ming Liu, and G. W. Nkabinde. 2016. "Big Data Stream Computing in Healthcare Real-Time

Analytics." In *2016 IEEE International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, 37–42. https://doi.org/10.1109/ICCCBDA.2016.7529531.

"The Open Definition - Open Definition - Defining Open in Open Data, Open Content and Open Knowledge." n.d. Accessed February 18, 2017. http://opendefinition.org/.

Thommandram, A., J. E. Pugh, J. M. Eklund, C. McGregor, and A. G. James. 2013. "Classifying Neonatal Spells Using Real-Time Temporal Analysis of Physiological Data Streams: Algorithm Development." In *2013 IEEE Point-of-Care Healthcare Technologies (PHT)*, 240–43. https://doi.org/10.1109/PHT.2013.6461329.

Tsai, Chun-Wei, Chin-Feng Lai, Han-Chieh Chao, and Athanasios V. Vasilakos. 2015. "Big Data Analytics: A Survey." *Journal of Big Data* 2 (1): 21. https://doi.org/10.1186/s40537-015-0030-3.

Tumasjan, Andranik. 2010. "Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment." ttp://www.aaai.org/ocs/index.php/ICWSM/ICWSM10/paper/viewFile/1441/1852.

Tumitan, D., and K. Becker. 2014. "Sentiment-Based Features for Predicting Election Polls: A Case Study on the Brazilian Scenario." In *2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, 2:126–33. https://doi.org/10.1109/WI-IAT.2014.89.

Tunggawan, Elvyna, and Yustinus Eko Soelistio. 2016. "And the Winner Is ...: Bayesian Twitter-Based Prediction on 2016 U.S. Presidential Election." *arXiv:1611.00440 [Cs]*, November. http://arxiv.org/abs/1611.00440.

Udupi, P. K., P. Malali, and H. Noronha. 2016. "Big Data Integration for Transition from E-Learning to Smart Learning Framework." In *2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC)*, 1–4. https://doi.org/10.1109/ICBDSC.2016.7460379.

Urmila, R. 2016. "Big Data Analysis: Comparision of Hadoop MapReduce, Pig and Hive Dr. Urmila R. Pol Assistant Professor, Department of Computer Science, Shivaji University, Kolhapur, India" Vol. 5, Issue 6, June 2016 Copyright to IJIRSET.

Vaidya, M., and S. Deshpande. 2015. "Distributed Data Management in Energy Sector Using Hadoop." In *2015 IEEE Bombay Section Symposium (IBSS)*, 1–6. https://doi.org/10.1109/IBSS.2015.7456653.

Vajjala, V. A. H. 2016. "A Novel Solution to Use Big Data Technologies and Improve Demand Response Program in Aggregated Residential Houses." In *2016 IEEE Conference on Technologies for Sustainability (SusTech)*, 251–56. https://doi.org/10.1109/SusTech.2016.7897176.

Valsamidis, Stavros, Theodosios Theodosiou, Ioannis Kazanidis, and Michael Nikolaidis. 2013. "A Framework for Opinion Mining in Blogs for Agriculture." *Procedia Technology*, 6th International Conference on Information and Communication Technologies in Agriculture, Food and Environment (HAICTA 2013), 8 (Supplement C): 264–74. https://doi.org/10.1016/j.protcy.2013.11.036.

Wang, H., and J. A. Castanon. 2015. "Sentiment Expression via Emoticons on Social Media." In *2015 IEEE International Conference on Big Data (Big Data)*, 2404–8. https://doi.org/10.1109/BigData.2015.7364034.

Wang, Hao, Dogan Can, Abe Kazemzadeh, François Bar, and Shrikanth Narayanan. 2012. "A System for Real-Time Twitter Sentiment Analysis of 2012 U.S. Presidential Election Cycle." In *Proceedings of the ACL 2012 System Demonstrations*, 115–120. ACL '12. Stroudsburg, PA, USA: Association for Computational Linguistics.

http://dl.acm.org/citation.cfm?id=2390470.2390490.

Wang, Zhibo, Lei Ke, Xiaohui Cui, Qi Yin, Longfei Liao, Lu Gao, and Zhenyu Wang. 2017. "Monitoring Environmental Quality by Sniffing Social Media." *Sustainability* 9 (2): 85. https://doi.org/10.3390/su9020085.

"Welcome to Apache Flume — Apache Flume." n.d. Accessed December 22, 2016. https://flume.apache.org/.

"Welcome to Apache Pig!" n.d. Accessed December 23, 2016. https://pig.apache.org/.

"Welcome to Apache™ Hadoop®!" n.d. Accessed December 24, 2016. http://hadoop.apache.org/#Who+Uses+Hadoop%3F.

"What Is Hadoop and NoSQL?" n.d. Accessed December 23, 2016. https://datajobs.com/what-is-hadoop-and-nosql.

Wicaksono, Andy Januar, Suyoto, and Pranowo. 2016. "A Proposed Method for Predicting US Presidential Election by Analyzing Sentiment in Social Media." In *2016 2nd International Conference on Science in Information Technology (ICSITech)*, 276–80. https://doi.org/10.1109/ICSITech.2016.7852647.

Wich, M., and T. Kramer. 2016. "Enrichment of Smart Home Services by Integrating Social Network Services and Big Data Analytics." In *2016 49th Hawaii International Conference on System Sciences (HICSS)*, 425–34. https://doi.org/10.1109/HICSS.2016.59.

Wong, F. M. F., C. W. Tan, S. Sen, and M. Chiang. 2016. "Quantifying Political Leaning from Tweets, Retweets, and Retweeters." *IEEE Transactions on Knowledge and Data Engineering* 28 (8): 2158–72. https://doi.org/10.1109/TKDE.2016.2553667.

Xie, Zheng, Guannan Liu, Junjie Wu, Lihong Wang, and Chunyang Liu. 2016. "Wisdom of Fusion: Prediction of 2016 Taiwan Election with Heterogeneous Big Data." In *2016 13th International Conference on Service Systems and Service Management (ICSSSM)*, 1–6. https://doi.org/10.1109/ICSSSM.2016.7538625.

Xu, G., M. Liu, F. Li, F. Zhang, and W. Shen. 2016. "User Behavior Prediction Model for Smart Home Using Parallelized Neural Network Algorithm." In *2016 IEEE 20th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, 221–26. https://doi.org/10.1109/CSCWD.2016.7565992.

Yadranjiaghdam, B., N. Pool, and N. Tabrizi. 2016. "A Survey on Real-Time Big Data Analytics: Applications and Tools." In *2016 International Conference on Computational Science and Computational Intelligence (CSCI)*, 404–9. https://doi.org/10.1109/CSCI.2016.0083.

Yan, K., X. You, X. Ji, G. Yin, and F. Yang. 2016. "A Hybrid Outlier Detection Method for Health Care Big Data." In *2016 IEEE International Conferences on Big Data and Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom) (BDCloud-SocialCom-SustainCom)*, 157–62. https://doi.org/10.1109/BDCloud-SocialCom-SustainCom.2016.34.

Yan, Ying, Yanjie Gao, Yang Chen, Zhongxin Guo, Bole Chen, and Thomas Moscibroda. 2016. "TR-Spark: Transient Computing for Big Data Analytics." In *Proceedings of the Seventh ACM Symposium on Cloud Computing*, 484–496. SoCC '16. New York, NY, USA: ACM. https://doi.org/10.1145/2987550.2987576.

Yang, H., X. Liu, S. Chen, Z. Lei, H. Du, and C. Zhu. 2016. "Improving Spark Performance with MPTE in Heterogeneous Environments." In *2016 International Conference on Audio, Language and Image Processing*

(ICALIP), 28–33. https://doi.org/10.1109/ICALIP.2016.7846627.

Yang, Yiming, and Xin Liu. 1999. "A Re-Examination of Text Categorization Methods." In *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 42–49. SIGIR '99. New York, NY, USA: ACM. https://doi.org/10.1145/312624.312647.

Zaharia, Matei, Mosharaf Chowdhury, Tathagata Das, Ankur Dave, Justin Ma, Murphy McCauley, Michael J. Franklin, Scott Shenker, and Ion Stoica. 2012. "Resilient Distributed Datasets: A Fault-Tolerant Abstraction for In-Memory Cluster Computing." In *Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation*, 2–2. NSDI'12. Berkeley, CA, USA: USENIX Association. http://dl.acm.org/citation.cfm?id=2228298.2228301.

Zainuddin, N., A. Selamat, and R. Ibrahim. 2017. "Hybrid Sentiment Classification on Twitter Aspect-Based Sentiment Analysis." Article in Press. Scopus. https://doi.org/10.1007/s10489-017-1098-6.

Zamani-Dehkordi, P., L. Rakai, H. Zareipour, and W. Rosehart. 2016. "Big Data Analytics for Modelling the Impact of Wind Power Generation on Competitive Electricity Market Prices." In *2016 49th Hawaii International Conference on System Sciences (HICSS)*, 2528–35. https://doi.org/10.1109/HICSS.2016.316.

Zhang, H., L. Zhang, X. Cheng, and W. Chen. 2016. "A Novel Precision Marketing Model Based on Telecom Big Data Analysis for Luxury Cars." In *2016 16th International Symposium on Communications and Information Technologies (ISCIT)*, 307–11. https://doi.org/10.1109/ISCIT.2016.7751641.

Zhang, Yongzheng, and Marco Pennacchiotti. n.d. *Predicting Purchase Behaviors from Social Media*.

Zhao, Jichang, Li Dong, Junjie Wu, and Ke Xu. 2012. "MoodLens: An Emoticon-Based Sentiment Analysis System for Chinese Tweets." In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1528–1531. KDD '12. New York, NY, USA: ACM. https://doi.org/10.1145/2339530.2339772.

Zheng, J., and A. Dagnino. 2014. "An Initial Study of Predictive Machine Learning Analytics on Large Volumes of Historical Data for Power System Applications." In *2014 IEEE International Conference on Big Data (Big Data)*, 952–59. https://doi.org/10.1109/BigData.2014.7004327.

Zheng, Z, P Wang, J Liu, and S Sun. 2015. "Real-Time Big Data Processing Framework: Challenges and Solutions." *Applied Mathematics and Information Sciences* 9 (January): 3169–90. https://doi.org/10.12785/amis/090646.

Zhu, W., H. Chen, and F. Hu. 2016. "ASC: Improving Spark Driver Performance with SPARK Automatic Checkpoint." In *2016 18th International Conference on Advanced Communication Technology (ICACT)*, 1–1. https://doi.org/10.1109/ICACT.2016.7423489.

Zhuang, Li, Feng Jing, and Xiao-Yan Zhu. 2006. "Movie Review Mining and Summarization." In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, 43–50. CIKM '06. New York, NY, USA: ACM. https://doi.org/10.1145/1183614.1183625.

Zirn, Cäcilia, Mathias Niepert, Heiner Stuckenschmidt, and Michael Strube. 2011. "Fine-Grained Sentiment Analysis with Structural Features." *Proceedings of IJCNLP*, January.

Table des illustrations

Figure 1: la chronologie du big data 26
Figure 2: Batch vs Micro-batch vs Streaming 29
Figure 3: la chaine de valeur big data 30
Figure 4: Hadoop 1.x vs Hadoop 2.x 39
Figure 5: Le concept de MapReduce 40
Figure 6: la phase MapReduce 41
Figure 7: SQL vs NoSQL 46
Figure 8: Tendances de recherche de Spark, Storm, Flink et Samza 51
Figure 9: Les différentes étapes de l'apprentissage automatique 70
Figure 10: Exemple de Naïve Bayes monocouche 76
Figure 11: Exemple de Naïve Bayes multicouches 77
Figure 12 Classification SVM 78
Figure 13 Courbes ROC 84
Figure 14: Les étapes de l'approche proposée 91
Figure 15: exemple d'application de l'algorithme 1 94
Figure 16: La première étape du modèle 95
Figure 17: La deuxième étape du modèle 98
Figure 18: la troisième étape du modèle 100
Figure 19: Exemple de construction de dictionnaire Hillary 103

Table des tableaux

Tableau 1: Approches couramment utilisées dans l'état de l'art 21
Tableau 2: Hortonworks vs Cloudera vs MapR 43
Tableau 3 : CouchBase vs MongoDB vs Hbase vs Cassandra 47
Tableau 4 Hive vs Pig 49
Tableau 5: Spark vs Storm vs Flink vs Samza 51
Tableau 6 : Solr vs ElasticSearch 53
Tableau 7: Qlikview vsTableau vs Spotfire vs Microsoft Power BI 54
Tableau 8: Outils big data couramment utilisés dans les applications 61
Tableau 9 Exemple de tweet 104
Tableau 10 Application de la deuxième étape du système 104
Tableau 11 Application de la troisième étape du système 104
Tableau 12 Exemple d'étiquetage morphosyntaxique 105
Tableau 13 Exemple d'étiquettage Treetager 106
Tableau 14: comparaison des performances des classifieurs 108
Tableau 15: Performances des méthodes d'analyse de sentiments 109
Tableau 16: Comparaison avec d'autres outils de l'analyse de sentiments 111
Tableau 17 Résultat de l'élection présidentielle américaine 112

RÉSUMÉ

Extraire l'opinion publique en analysant les big social data a connu un essor considérable en raison de leur nature interactive, en temps réel. En effet, les données issues des réseaux sociaux sont étroitement liées à la vie personnelle que l'on peut utiliser pour accompagner les grands événements en suivant le comportement des personnes. C'est donc dans ce contexte que nous nous intéressons particulièrement aux méthodes d'analyse du Big data. La problématique qui se pose est que ces données sont tellement volumineuses et hétérogènes qu'elles en deviennent difficiles à gérer avec les outils classiques. Pour faire face aux défis du Big data, de nouveaux outils ont émergés. Cependant, il est souvent difficile de choisir la solution adéquate, car la vaste liste des outils disponibles change continuellement. Pour cela, nous avons fourni une étude comparative actualisée des différents outils utilisés pour extraire l'information stratégique du Big data et les mapper aux différents besoins de traitement.

La contribution principale de la thèse de doctorat est de proposer une approche d'analyse générique pour détecter de façon automatique des tendances d'opinion sur des sujets donnés à partir des réseaux sociaux. En effet, étant donné un très petit ensemble de hashtags annotés manuellement, l'approche proposée transfère l'information du sentiment connue des hashtags à des mots individuels. La ressource lexicale qui en résulte est un lexique de polarité à grande échelle dont l'efficacité est mesurée par rapport à différentes tâches de l'analyse de sentiment. La comparaison de notre méthode avec différents paradigmes dans la littérature confirme l'impact bénéfique de notre méthode dans la conception des systèmes d'analyse de sentiments très précis. En effet, notre modèle est capable d'atteindre une précision globale de 90,21%, dépassant largement les modèles de référence actuels sur l'analyse du sentiment des réseaux sociaux.

mots-clés : big data, big social data, réseaux sociaux, analyse de sentiments, analyse de tendances, outils big data, performance des outils big data

ABSTRACT

Extracting public opinion by analyzing big social data has grown substantially due to its interactive nature, in real time. In fact, our actions on social media generate digital traces that are closely related to our personal lives and can be used to accompany major events by analysing peoples' behavior. It is in this context that we are particularly interested in Big data analysis methods. The volume of these daily-generated traces increases exponentially creating massive loads of information, known as big data. Such important volume of information cannot be stored nor dealt with using the conventional tools, and so new tools have emerged to help us cope with the big data challenges. For this, the aim of the first part of this manuscript is to go through the pros and cons of these tools, compare their respective performances and highlight some of its interrelated applications such as health, marketing and politics. Also, we introduce the general context of big data, Hadoop and its different distributions. We provide a comprehensive overview of big data tools and their related applications.

The main contribution of this PHD thesis is to propose a generic analysis approach to automatically detect trends on given topics from big social data. Indeed, given a very small set of manually annotated hashtags, the proposed approach transfers information from hashtags known sentiments (positive or negative) to individual words. The resulting lexical resource is a large-scale lexicon of polarity whose efficiency is measured against different tasks of sentiment analysis. The comparison of our method with different paradigms in literature confirms the impact of our method to design accurate sentiment analysis systems. Indeed, our model reaches an overall accuracy of 90.21%, significantly exceeding the current models on social sentiment analysis.

keywords : big data, big social data, social media, sentiment analysis, trends detection, big data performance, Big data benchmarking

Présidence de l'université
40 rue de rennes – BP 73532
49035 Angers cedex
Tél. 02 41 96 23 23 | Fax 02 41 96 23 00

ENGAGEMENT DE NON PLAGIAT

Je, soussigné(e) EL ALAOUI IMANE
déclare être pleinement conscient(e) que le plagiat de documents ou d'une partie d'un document publiée sur toutes formes de support, y compris l'internet, constitue une violation des droits d'auteur ainsi qu'une fraude caractérisée.
En conséquence, je m'engage à citer toutes les sources que j'ai utilisées pour écrire ce rapport ou mémoire.

signé par l'étudiant(e) le **02** / **05** / **2018**

Cet engagement de non plagiat doit être signé et joint à tous les rapports, dossiers, mémoires.

Présidence de l'université
40 rue de rennes – BP 73532
49035 Angers cedex
Tél. 02 41 96 23 23 | Fax 02 41 96 23 00

université angers

UNIVERSITE BRETAGNE LOIRE

université angers

Thèse de Doctorat

Imane EL ALAOUI

Transformer les big social data en prévisions - méthodes et technologies

Transforming big social data into forecasts - methods and technologies

Résumé

Extraire l'opinion publique en analysant les Big Social data a connu un essor considérable en raison de leur nature interactive, en temps réel. En effet, les données issues des réseaux sociaux sont étroitement liées à la vie personnelle que l'on peut utiliser pour accompagner les grands événements en suivant le comportement des personnes. C'est donc dans ce contexte que nous nous intéressons particulièrement aux méthodes d'analyse du Big data. La problématique qui se pose est que ces données sont tellement volumineuses et hétérogènes qu'elles en deviennent difficiles à gérer avec les outils classiques. Pour faire face aux défis du Big data, de nouveaux outils ont émergés. Cependant, il est souvent difficile de choisir la solution adéquate, car la vaste liste des outils disponibles change continuellement. Pour cela, nous avons fourni une étude comparative actualisée des différents outils utilisés pour extraire l'information stratégique du Big Data et les mapper aux différents besoins de traitement. La contribution principale de la thèse de doctorat est de proposer une approche d'analyse générique pour détecter de façon automatique des tendances d'opinion sur des sujets donnés à partir des réseaux sociaux. En effet, étant donné un très petit ensemble de hashtags annotés manuellement, l'approche proposée transfère l'information du sentiment connue des hashtags à des mots individuels. La ressource lexicale qui en résulte est un lexique de polarité à grande échelle dont l'efficacité est mesurée par rapport à différentes tâches de l'analyse de sentiment. La comparaison de notre méthode avec différents paradigmes dans la littérature confirme l'impact bénéfique de notre méthode dans la conception des systèmes d'analyse de sentiments très précis. En effet, notre modèle est capable d'atteindre une précision globale de 90,21%, dépassant largement les modèles de référence actuels sur l'analyse du sentiment des réseaux sociaux.

Mots clés

Big data, Big social data, Réseaux sociaux, Analyse de sentiments, Analyse de tendances, Outils big data, Performance des outils big data

Abstract

Extracting public opinion by analyzing Big Social data has grown substantially due to its interactive nature, in real time. In fact, our actions on social media generate digital traces that are closely related to our personal lives and can be used to accompany big events by analyzing peoples' behavior. It is in this context that we are particularly interested in Big Data analysis methods. The volume of these daily-generated traces increases exponentially creating massive loads of information, known as big data. Such important volume of information cannot be stored nor dealt with using the conventional tools, and so new tools have emerged to help us cope with the big data challenges. For this, the aim of the first part of this manuscript is to go through the pros and cons of these tools, compare their respective performances and highlight some of its interrelated applications such as health, marketing and politics. Also, we introduce the general context of big data, Hadoop and its different distributions. We provide a comprehensive overview of big data tools and their related applications.

The main contribution of this PHD thesis is to propose a generic analysis approach to automatically detect trends on given topics from big social data. Indeed, given a very small set of manually annotated hashtags, the proposed approach transfers information from hashtags known sentiments (positive or negative) to individual words. The resulting lexical resource is a large-scale lexicon of polarity whose efficiency is measured against different tasks of sentiment analysis. The comparison of our method with different paradigms in literature confirms the impact of our method to design accurate sentiment analysis systems. Indeed, our model reaches an overall accuracy of 90.21%, significantly exceeding the current models on social sentiment analysis.

Key Words

Big data, Big social data, Social media, Sentiment analysis, Trends detection, Big data performance, Big data benchmarking.