Institut National des Langues et Civilisations Orientales

Département Textes, Informatique, Multilinguisme

Développement en open source et évaluation d'un système de classification multi-classes pour des articles de presse du domaine bancaire

MASTER

TRAITEMENT AUTOMATIQUE DES LANGUES

Parcours :

Ingénierie Multilingue

par

Xi RONG

Directeur de mémoire :

Patrick Paroubek

Encadrant :

Amanda Bouffier

Michel Bernardini

Année universitaire 2017/2018

TABLE DES MATIÈRES

Liste des figures **5**

Liste des tableaux **5**

Résumé **7**

Remerciements **9**

Introduction **11**

I Contexte général **13**

1 État de l'art **15**
1.1 Introduction 15
1.2 Classification des textes 15
1.3 Evaluation d'un système 18

2 Contexte Métier **21**
2.1 Introduction 21
2.2 La plate-forme LEOnard 21
2.3 L'outil de fouille de texte - Cogito 22
2.4 La classification en secteurs 24

II Expérimentations **27**

3 système existant **29**
3.1 Introduction 29
3.2 Le corpus et les secteurs 29
3.3 L'outil et les méthodes utilisées 32
3.4 Conclusion 34

4 Le nouveau système basé sur de l'open source **35**
4.1 Introduction 35
4.2 Le corpus 35
4.3 Le processus de traitement 41
4.4 Expériences 44

4.5 Conclusion 45

5 Résultats et évaluations **47**
5.1 Introduction 47
5.2 Les mesures d'évaluation classiques 48
5.3 Évaluations sur le corpus de test 49
5.4 Évaluations sur les nouveaux documents 55
5.5 Insuffisances des mesures classiques 58
5.6 Mesures d'évaluation complémentaires 58
5.7 Conclusion 61

Conclusion **63**

Bibliographie **65**

A Annexe **67**
A.1 Plan de classement sectoriel complet 67

Liste des figures

1.1 Organigramme du processus de classification des textes 16
2.1 Intégration des documents dans LEOnard 22
2.2 Entités nommées et nuage de mots dans LEOnard 23
2.3 Interface de Cogito Studio Express 24
2.4 Plan d'annotation de projet 24
2.5 Exemple de secteurs dans LEOnard 25
3.1 Exemple de fichier HTML 30
3.2 Exemple de fichier TMX 30
3.3 Classement des catégories dans LEOnard [1] 31
4.1 Plan factoriel de l'analyse des correspondances pour le corpus français . . . 38
4.2 Plan factoriel de l'analyse des correspondances pour le corpus anglais . . . 38
4.3 Schéma général de l'approche automatique pour la classification de textes . 41
5.1 Étapes d'une validation croisée à 10 plis 52
5.2 Résultat par secteur du SGD pour les documents français 53
5.3 Résultat par secteur du SGD pour les documents anglais 53
5.4 Résultats français par secteur après l'ajout du corpus d'apprentissage . . . 54
5.5 Résultats anglais par secteur après l'ajout du corpus d'apprentissage 54
5.6 Évaluation sur le français avec le système existant 55
5.7 Évaluation sur le français avec le nouveau système 55
5.8 Évaluation sur l'anglais avec le système existant 56
5.9 Évaluation sur l'anglais avec le nouveau système 56
5.10 Matrice de confusion du nouveau système pour le français 57
5.11 Matrice de confusion du nouveau système pour l'anglais 57
A.1 Plan de classement sectoriel complet 68

Liste des tableaux

3.1 Paramètres du modèle d'apprentissage 33
4.1 Informations principales du corpus 36
4.2 Répartition du corpus 36
4.3 Les 10 termes les plus fréquents par catégorie 37
4.4 Les 10 termes les plus spécifiques par catégorie (français) 39
4.5 Les 10 termes les plus spécifiques par catégorie (anglais) 40

5.1 Tableau de contingence de classification 48
5.2 Résultats du système existant 49
5.3 Résultats après les nettoyages 50
5.4 Résultats avec les stems 50
5.5 Résultats avec les groupes nominaux 51
5.6 Résultats des deux systèmes 51
5.7 Résultats de la validation croisée sur SVC et SGD 52
5.8 Exemple du résultat de classification 58
5.9 Leo-Score des deux systèmes 60

Résumé

Ce travail s'inscrit dans le projet LEOnard des Études Économiques de BNP Paribas. Il s'agit du redéveloppement en open source et de l'évaluation d'un système de classification multi-classes pour des articles de presse du domaine bancaire. Pour construire le système, différentes expériences ont été menées afin de trouver les paramètres optimaux : pré-traitements du corpus, sélection de *features*, choix de l'algorithme, etc. C'est l'algorithme SGD (Stochastic Gradient Descent) qui a finalement été retenu avec une F-Mesure de 95% sur le corpus de test.

Le nouveau système doit s'adapter aux contraintes réelles de la tâche et les évaluations doivent aussi être adaptables et comparables avec le dernier. Cependant, les mesures classiques ne permettent pas d'intégrer les évaluations adaptées à la tâche ni de qualifier globalement la performance du système. Par conséquent, nous avons proposé des métriques complémentaires, dont le « Leo-Score », pour évaluer le système de classification intégré dans la plate-forme. Notre système a eu un Leo-Score de 81,76% tandis que le système existant a eu un Leo-Score de 64,88% pour des nouvelles données présentées dans la plate-forme LEOnard.

Mots clés : *classification multi-classes des textes, apprentissage supervisé, système de classification, classification automatique, open source, mesure d'évaluation, validation croisée, Leo-Score*

Remerciements

Je souhaite en premier lieu remercier toute l'équipe pédagogique de pluriTAL pour la richesse et la qualité de leur enseignement puis pour leurs grands efforts qui assure à leurs étudiants une formation actualisée.

Je remercie également mon tuteur de stage, Monsieur Michel Bernardini, Responsable Informatique et Communication des Etudes Economiques, pour sa disponibilité et son écoute, l'attention qu'il a apporté a mon travail et la confiance qu'il m'a accordé pendant ce stage, ce qui m'a permis de travailler avec une grande autonomie.

Je tiens à remercier tout particulièrement Amanda Bouffier, qui m'a encadrée durant ce stage avec une patience à toute épreuve, qui a pris tout le temps nécessaire pour répondre à mes questions, débloquer des situations, m'aider à chercher des réponses à chaque fois que j'en avais besoin. Je la remercie également pour sa précieuse aide à la relecture et à la correction de mon mémoire.

J'adresse des chaleureux remerciements à l'équipe LEOnard qui m'a accompagné pendant ces six mois : Mickaëlle Fils Marie-Luce, Thomas Leloutre, Benigna Nekrosiute, Aude Brugère, Henri Garnier et Ines Messaoudi. Les tâches réalisées en équipe, leur aide à la rédaction de ce mémoire et leur bonne humeur quotidienne resteront un très bon souvenir.

Je souhaite remercier mon tuteur-enseignant, Monsieur Patrick Paroubek, pour m'avoir guidé dans la préparation de ce mémoire et pour sa disponibilité.

Enfin, j'adresse mes plus sincères remerciements à tous mes proches et amis, qui m'ont accompagné, aidé, soutenu et encouragé tout au long de la réalisation de ce mémoire.

Introduction

Maîtriser l'information est devenue une des clés de voûte de l'économie actuelle. Dans le cadre de la transformation digitale de la société, les nouvelles technologies disruptives comme Intelligence Artificielle (AI), le *NLP (Natural Language Processing)*, le *Machine Learning*, jouent un rôle de grande importance. Ces technologies qui sont en train de bouleverser l'ensemble de l'économie touchent fortement le secteur financier et bancaire : ces derniers font appel de plus en plus à ces technologies pour automatiser le traitement des informations et rendre leur traitement efficace et pertinent.

Avec la popularisation de l'IA on assiste à un changement d'échelle : de l'innovation à l'industrialisation, du « big data » au « smart data ». Pour des raisons économiques et de confidentialité, il est plus que jamais nécessaire que ces systèmes soient fiables, robustes et reproductibles. De plus, dans les situations réelles, des mises à jours et des remplacements de systèmes en raison de l'offre de plus en plus nombreuse, de montée en maturité de celle-ci, ou de changement de stratégies d'entreprises ne restent pas rares. Les résultats générés par le nouveau système doivent alors être adéquats et sa qualité doit être comparable avec le dernier. Comment construire un système de qualité, comparable et comment le mesurer ? Comment donner des clés d'appréciation pour le client qui soient rassurantes et efficaces ? Ces questions sont devenus un enjeu crucial pour les entreprises.

Ce stage s'inscrit dans le projet LEOnard des Etudes Economiques de BNP Paribas. Étant le leader bancaire dans la zone européenne, BNP Paribas est toujours prête à anticiper des changements et des opportunités pour maintenir et améliorer la compétitivité à l'ère digitale, comme le dit son slogan « la banque d'un monde qui change ». Le projet LEONard a été lancé par Michel Bernardini en 2004 et a pour objectif de proposer des ressources économiques répondant aux attentes des collaborateurs et de leur faciliter la recherche et le filtrage d'informations.

Une des fonctionnalités de LEOnard s'appuie sur l'intégration d'un système de classification consistant à catégoriser les articles de la presse par secteurs d'activité. Ce système a été développé par des stagiaires précédents avec un outil de Temis (société devenue « Expert System ») reposant sur du *Machine Learning*. Après le changement de stratégie de l'entreprise, Expert System a décidé d'arrêter le support de cet outil. Dans ce contexte, reconstruire un nouveau système avec des modules

open source est devenu une piste prometteuse.

Le nouveau système de classification doit s'adapter aux contraintes réelles de la tâche et les évaluations doivent aussi être adaptables et comparables avec le dernier. Les critères de mesure des performances les plus populaires pour cette tâche sont la précision, le rappel et le F-mesure. Ces critères sont employés indifféremment dans le domaine de recherche d'information et de classification depuis presque 30 ans. Cependant, ces mesures présentent quelques défauts : elles consistent plutôt à évaluer les résultats générés par le système d'un point de vue logiquement binaire pour le cas général et ne permettent pas d'intégrer les évaluations adaptées à la tâche ni de qualifier globalement la performance du système. Elles sont donc insuffisantes pour évaluer et comparer les deux systèmes, notamment pour ceux qui sont intégrés dans les domaines qui demandent davantage de fiabilité, de stabilité et de reproductibilité comme la banque.

Nous pouvons dès lors dégager la problématique suivante : Comment construire un modèle fiable de classification multi-classes de texte et comparer sa qualité avec le système existant dans le cadre de la stratégie d'intelligence économique de la banque ? Pour répondre au mieux à cette problématique, il faut d'abord avoir une connaissance solide du système existant, puis concentrons sur les étapes principales de la construction du modèle d'apprentissage, à savoir : la représentation des documents dans un *VSM (Vector Space Model)*, la sélection des *features* (caractéristiques), la transformation ou extraction des *features*, etc. Enfin, nous comparons les deux systèmes aux plusieurs niveaux : les différents mesures d'évaluation pour les résultats générés, la robustesse et la stabilité, la reproductibilité, etc.

Ce travail est organisé comme suit : dans le chapitre 1 nous présentons l'état de l'art des méthodes de classification multi-classes et des mesures d'évaluation. Dans le chapitre 2, nous présentons le contexte métier du travail. Dans le chapitre 3, nous présenterons le système de classification existant dans la plate-forme. Dans le chapitre 4, nous exposons le nouveau système construit basé sur l'open sources. Enfin, dans le chapitre 5, nous abordons les évaluations sur les deux systèmes avant la conclusion de notre travail.

Première partie

Contexte général

CHAPITRE 1

ÉTAT DE L'ART

Sommaire

1.1 Introduction **15**
1.2 Classification des textes **15**
1.2.1 Les types de classification automatique des textes 15
1.2.2 Le processus de classification de texte en apprentissage . . . 16
1.3 Evaluation d'un système **18**

1.1 Introduction

Nous présentons dans ce chapitre un état de l'art sur la classification de textes et les mesures d'évaluation liées à ce type de système.

1.2 Classification des textes

La classification de texte consiste à classer de nouveaux documents en classes [Liu, 2007] [Schütze et al., 2008]. Il existe plusieurs approches pour construire ce type de modèle, dont des approches à bases de règles. Cependant, le paradigme de l'apprentissage automatique [Mitchell et al., 1997] [Sebastiani, 2002]s'est imposé comme efficace et prédominant.

1.2.1 Les types de classification automatique des textes

Les méthodes de classification se distinguent en la classification supervisée (ou catégorisation) et la classification non supervisée (ou *clustering*). La catégorisation consiste à apprendre un modèle pour prédire la classe d'un document à partir des documents pré-étiquetées dans des classes définies à l'avance alors que le *clustering* consiste à trouver un regroupement « naturel » des documents à partir de données non étiquetées.

Il existe deux grandes variantes de problèmes de classification à distinguer : la classification multi-classes et la classification multi-labels. La classification multi-

classes désigne une tâche de classification avec plus de deux classes ; par exemple, classer un ensemble d'images de fruits qui peuvent être des oranges, des pommes ou des poires. La classification multi-classes repose sur l'hypothèse que chaque échantillon est attribué à une seule étiquette (label) : un fruit peut être une pomme ou une poire mais pas les deux à la fois. La classification multi-labels assigne à chaque échantillon un ensemble d'étiquettes cibles. Cela peut être considéré comme une prédiction de propriétés d'un point de données qui ne s'excluent pas mutuellement, telles que des rubriques pertinentes pour un document. Un texte peut concerner à la fois la religion, la politique, la finance ou l'éducation, ou aucune de ces thématiques.

1.2.2 Le processus de classification de texte en apprentissage

La littérature décrit principalement les quatre étapes du processus de classification de texte [Aas and Eikvil, 1999] [Guzella and Caminhas, 2009] [Adeva et al., 2014] :

— constitution du corpus (l'acquisition des données avec étiquetage en catégories, la division en corpus d'apprentissage et corpus de test) ;
— représentation du corpus (les pré-traitements, la sélection et la pondération des caractéristiques, la modélisation de documents) ;
— construction du classifieur (les algorithmes d'apprentissage) ;
— évaluation du modèle d'apprentissage.

Le processus peut être illustré ainsi :

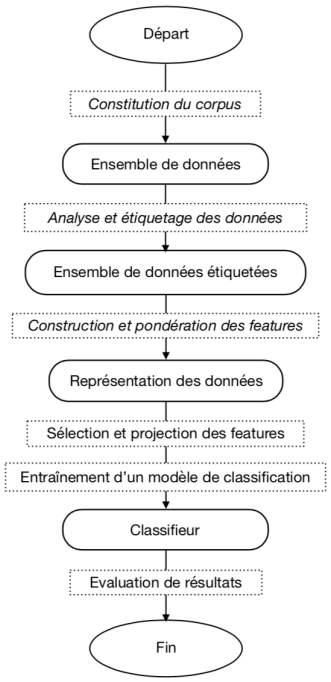

FIGURE 1.1 – Organigramme du processus de classification des textes

Représentation du corpus

Le corpus doit être représenté en une forme appropriée pour que les algorithmes d'apprentissage puissent le traiter. D'abord, les pré-traitements des corpus consistent à nettoyer et préparer des corpus en mots bruts. Il s'agit de la tokenisation, la suppression des mots vides, et le *stemming* ou la lemmatisation. Ensuite, la sélection des *features* ou descripteurs linguistiques permets de réduire la complexité de document en ne conservant que les *features* les plus informatifs. Ces descripteurs peuvent être des mots (lemmatisés ou pas), des n-grams, des annotations sémantiques, etc. Des méthodes comme le gain d'information (IG), l'information mutuelle (MI), le chi-square (χ^2) et TF-IDF permettent non seulement de nous aider à sélectionner les *features* les plus informatifs mais aussi de pondérer ces *features*. Enfin, la modélisation des documents consiste à représenter des données textuelles en d'autres formats avec lesquels les calculs peuvent se réaliser. Il y a deux représentations de données textuelles :

— le modèle vectoriel (VSM) [Hand et al., 2001] [Manning et al., 2008] dans lequel un document est représenté comme un vecteur qui contient les poids des features sélectionnés ;
— la représentation de graphe [Mihalcea and Radev, 2011] où un document est modélisé en format de graphe, par exemple, les nœuds comme des mots.

Ainsi, la représentation du corpus est une étape importante pour construire un modèle de classification performant. D'après la littérature, il est indispensable de faire des pré-traitement du corpus, la sélection et la pondération avec la méthode TF-IDF pour représenter le document dans un modèle vectoriel est la plus performante et stable pour la classification des textes.

La construction du classifieur

Il existe plusieurs algorithmes classiques pour entraîner un modèle de classification, à savoir :

— **K-nearest neighbor (KNN).** Il consiste à calculer les distances ou les similarités pour la classification. Mais il est peu utilisé pour la classification des textes parce que le temps de calcul est long et qu'il est difficile de trouver le valeur optimal de « k » ;

— **Arbres de décision.** Ils sont surtout appréciés pour leur lisibilité, qui les rend en général compréhensibles par un humain (dans les limites d'une certaine taille). Mais ils ont tendance à être de moins en moins utilisés de nos jours. Sur les textes, leurs performances (en termes de précision / rappel / F-score) sont habituellement moins bonnes que celles obtenues avec d'autres méthodes ;

— **Naïve Bayes.** Les programmes de type « Naive Bayes » sont simples, rapides et relativement efficaces pour les données textuelles. Dans des tâches de clas-

sification des textes, ils sont plutôt utilisés pour ranger les mails en « spam » et « non spam » ;

— **SVM (Support Vector Machines).** Il s'agit de méthodes très puissantes issues d'une analyse mathématique précise et avancée du problème de l'apprentissage d'un séparateur binaire dans un espace vectoriel. Jusqu'à récemment, ces méthodes donnaient la plupart du temps les meilleurs résultats. Cependant, comme elles reposent sur le séparateur binaire, quand il s'agit de classification multi-classes , la stratégie habituelle consiste à lancer plusieurs apprentissages indépendants pour chercher à séparer une classe de toutes les autres, et le temps de calcul augmente de façon exponentielle.

En plus de ces algorithmes classiques, les nouvelles technologies plus récentes comme *Stochastic Gradient Descent (SGD)*et *lightGBM* ont vu le jour. Leur avantage principal est qu'ils sont très efficaces et performants pour un grand nombre de données et des représentations vectorielles clairsemées. Le *SGD Classifieur (Stochastic Gradient Descent)* correspond à un classifieur linéaire (SVM, la régression logistique, etc.) avec l'apprentissage *SGD*. Et le *LightGBM* classifieur est un *gradient boosting framework* basé sur un algorithme d'arbre de décision.

1.3 Evaluation d'un système

Une fois qu'un modèle de classification est formé, nous pouvons choisir et établir des indicateurs pour mesurer sa performance. L'indicateur le plus utilisé dans le domaine du *NLP* est le taux d'exactitude, mais il s'agit d'une évaluation peu profonde et partial. Dans le domaine de la classification et la recherche d'information, la précision et le rappel sont les mesures d'évaluation les plus utilisées, mais ce sont toujours des indicateurs scalaires. La F-mesure est un indicateur de synthèse, il correspond à une moyenne harmonique de la précision et du rappel, qui permet d'évaluer le système plus objectivement. Le problème que pose la F-mesure est qu'elle ne permet pas la différenciation des erreurs et qu'elle reste sensible à la distribution des classes car basée sur la précision et le rappel. Il est donc préférable d'utiliser une nouvelle mesure afin de comparer les deux.

Toutefois, lors d'une expérience d'apprentissage automatique, il est courant de conserver une partie des données disponibles sous la forme d'un ensemble de test (corpus de test). La méthode de division en corpus d'apprentissage et corpus de test influencent aussi la performance du système. Généralement, le corpus est divisé au hasard par des pourcentages (par exemple, 20% pour le test, 80% pour l'entraînement). Mais il présente aussi des inconvénients : comme le corpus a été divisé, le nombre d'échantillons pouvant être utilisés pour l'apprentissage du modèle est considérablement réduit, et les résultats peuvent dépendre d'un choix aléatoire particulier pour la paire de séries (entraînement, test). Pour remédier à ces inconvénients, la validation croisée (*cross-validatio*n en anglais) est proposée. *LeaveOutOut (LOO)* et le

k-fold cross validation sont les deux applications les plus populaires. Le *k-fold cross validation* divise tous les échantillons en groupes d'échantillons, appelés "plis" (*fold* en anglais), de tailles égales (si possible). La fonction de prédiction est apprise à l'aide de plis, et le pli laissé de côté est utilisé pour le test. *LOO* est la validation croisée simple, il s'agit de prendre tous les échantillons comme corpus d'entraînement sauf un, qui est laissé à côté comme corpus de test. Ainsi, pour les échantillons, nous avons différents corpus d'apprentissage et différents corpus de tests. En général, la plupart des auteurs et les preuves empiriques suggèrent qu'une validation croisée à 5 ou 10 plis devrait être préférée à la *LOO*.

Cependant, tous ces mesures d'évaluation listées sont utilisées largement pour estimer la performance d'un modèle de classification. Bien que la procédure d'évaluation soit assez générale, sa mise en œuvre est strictement liée à un objectif précis de classification des documents. Adapter des mesures convenables pour des tâches différentes reste encore un enjeu. En outre, mesurer la qualité d'un système ne se limite pas aux résultats obtenus. La robustesse et la stabilité, la reproductibilité, la complexité en temps et en espace sont aussi des facteurs importants à exploiter pour qualifier un système de classification.

CHAPITRE 2

CONTEXTE MÉTIER

Sommaire

2.1 Introduction **21**
2.2 La plate-forme LEOnard **21**
2.3 L'outil de fouille de texte - Cogito **22**
2.4 La classification en secteurs **24**

2.1 Introduction

Ce chapitre présente le contexte professionnel du stage, la plate-forme LEOnard sur laquelle est déployée la catégorisation automatique, et l'outil de fouille de texte utilisé dans la plate-forme Cogito permettant de construire les ressources et modules linguistiques.

2.2 La plate-forme LEOnard

La plate-forme de veille collaborative de BNP Paribas, baptisée LEOnard (Navigateur Assistant de Recherche Documentaire) a été introduit au sein des Études Économiques en 2004 par Michel Bernardini. Il s'agit d'un outil qui a pour objectif de diffuser et partager des informations à tous les collaborateurs de BNP Paribas.

Dans la plate-forme LEOnard, les articles proviennent de plusieurs sources : la base de données interne GIMADOC qui est alimenté par le centre de documentation du Groupe BNP Paribas, la presse quotidienne et hebdomadaire (envoyés par le prestataire VMH), et des articles d'internet crawlé par l'outil de veille automatique KB Crawl.

La plupart des documents collectés sont préalablement au format PDF. Tous ces documents sont d'abord convertis en format HTML et ensuite indexés par le moteur de recherche, CustomerMatrix. Les documents sont alors analysés par Cogito, un outil d'analyse sémantique. Les documents présentés dans LEOnard peuvent être filtrés par entités nommées et concepts économiques, ils sont aussi classé par secteurs.

Les documents qui attirent plus d'intérêt sont classés plus finement dans les catégories « Macro économie », « ALMT » (*Asset and Liability Management and Treasury*) et « Future of work ».

Le processus d'intégration des documents dans LEOnard peut être schématisé de la manière suivante :

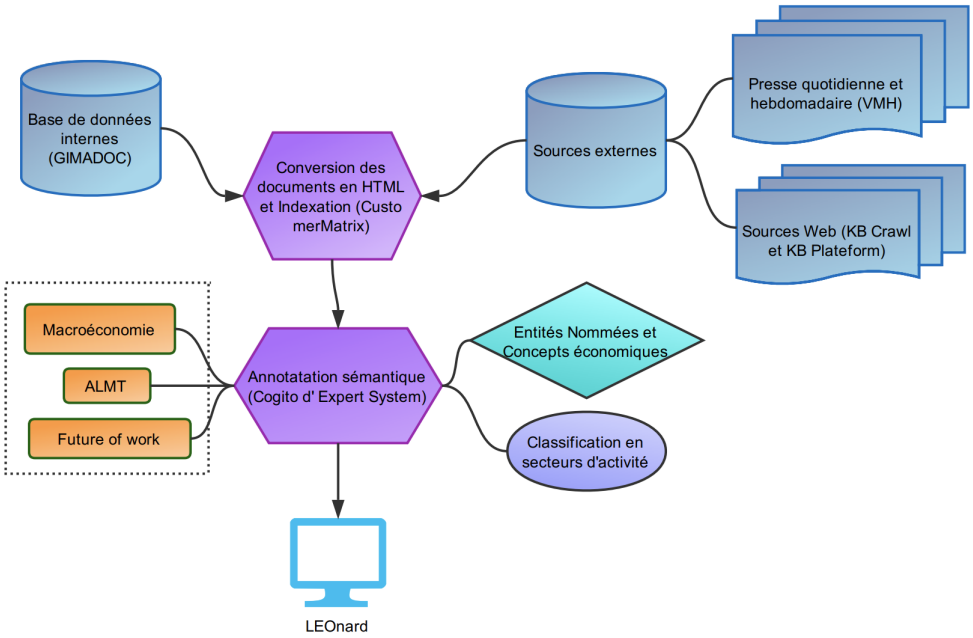

FIGURE 2.1 – Intégration des documents dans LEOnard

2.3 L'outil de fouille de texte - Cogito

Cogito, l'outil d'analyse cognitive de texte développé par Expert System, permet de faire des analyses sémantiques sur des données textuelles. Le concept de cartouche de connaissance est développé par la société Temis (le prédécesseur d'Expert System), il s'agit des modules qui effectuent des tâches différents (identification de la langue, annotation morphosyntaxique, extractions sémantiques ou traitements périphériques comme la gestion de contexte, etc.) afin de réaliser des tâches d'analyse sémantique selon les besoins.

Les cartouches de connaissance s'appuient principalement sur deux approches. La première est l'approche symbolique qui repose sur des ressources linguistiques comme les thésaurus et des règles de grammaire. Expert System fournit des cartouches standards qui permettent de faire l'annotation morphosyntaxique et d'extraire des entités nommées.

FIGURE 2.2 – Entités nommées et nuage de mots dans LEOnard

Expert System permet également de développer des cartouches spécifiques qui consistent à extraire des concepts dans un domaine précis, classer des documents ou effectuer un certain nombre de calcul pour les annotations. Avec Cogito Studio Express, comme illustré dans la figure 2.3, nous pouvons construire le thésaurus en format SKOS et l'intégrer dans les cartouches de connaissance. Ensuite, nous configurerons le plan d'annotation avec des cartouches de post-processus (calcul les scores de terms, gestion de contexte, etc), comme illustré dans la figure 2.4. Enfin, nous déployons le projet dans l'administration et l'annotation et la classification se fait automatiquement. Dans notre cas, les thesaurus sont créés pour « Macro économie », « ALMT », « Future of work » et « Innovation bancaire ».

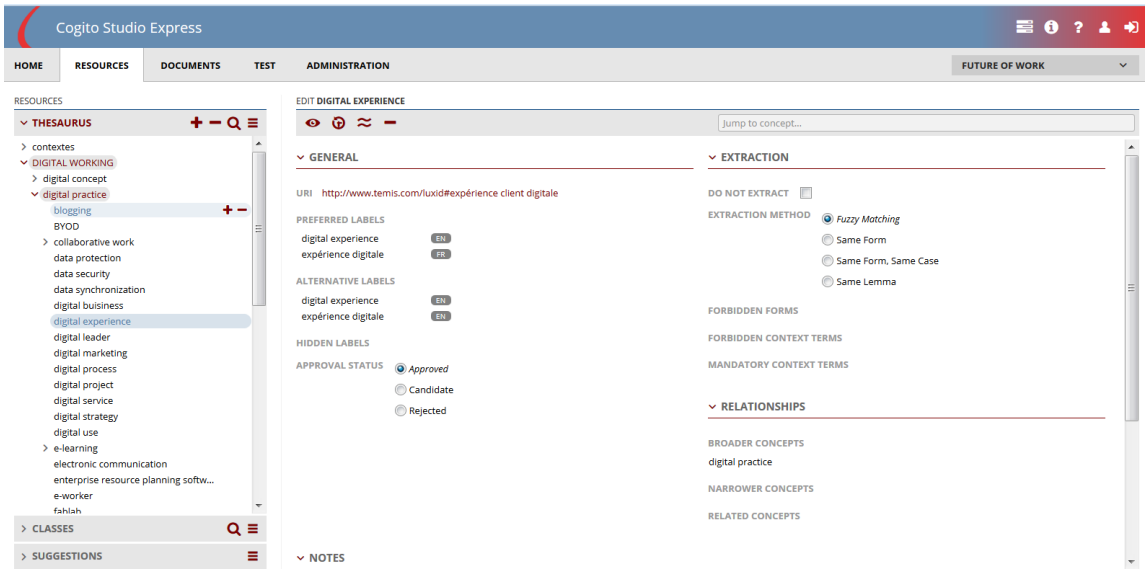

FIGURE 2.3 – Interface de Cogito Studio Express

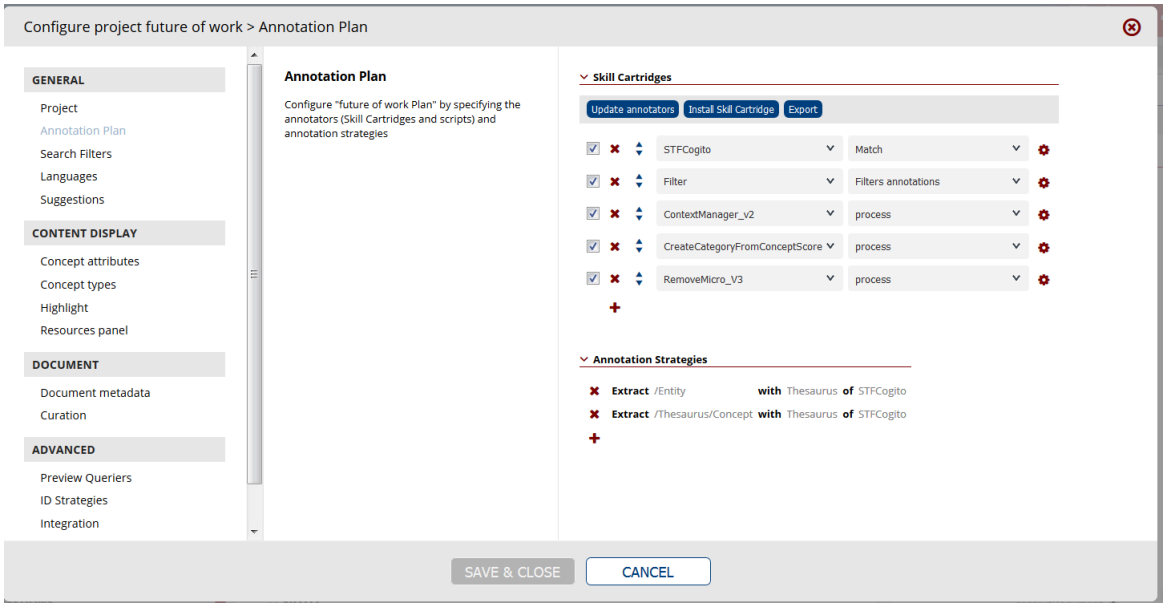

FIGURE 2.4 – Plan d'annotation de projet

2.4 La classification en secteurs

La seconde approche des cartouches de connaissance est l'apprentissage automatique. Elle consiste à créer des cartouches qui permettent de faire la classification automatique supervisée pour les documents par l'apprentissage automatique. Il s'agit des produits de la génération précédente proposés par Expert System. Néanmoins, la classification des secteurs d'activité qui est basée sur cette approche reste toujours en production dans LEOnard comme illustré ci-dessous :

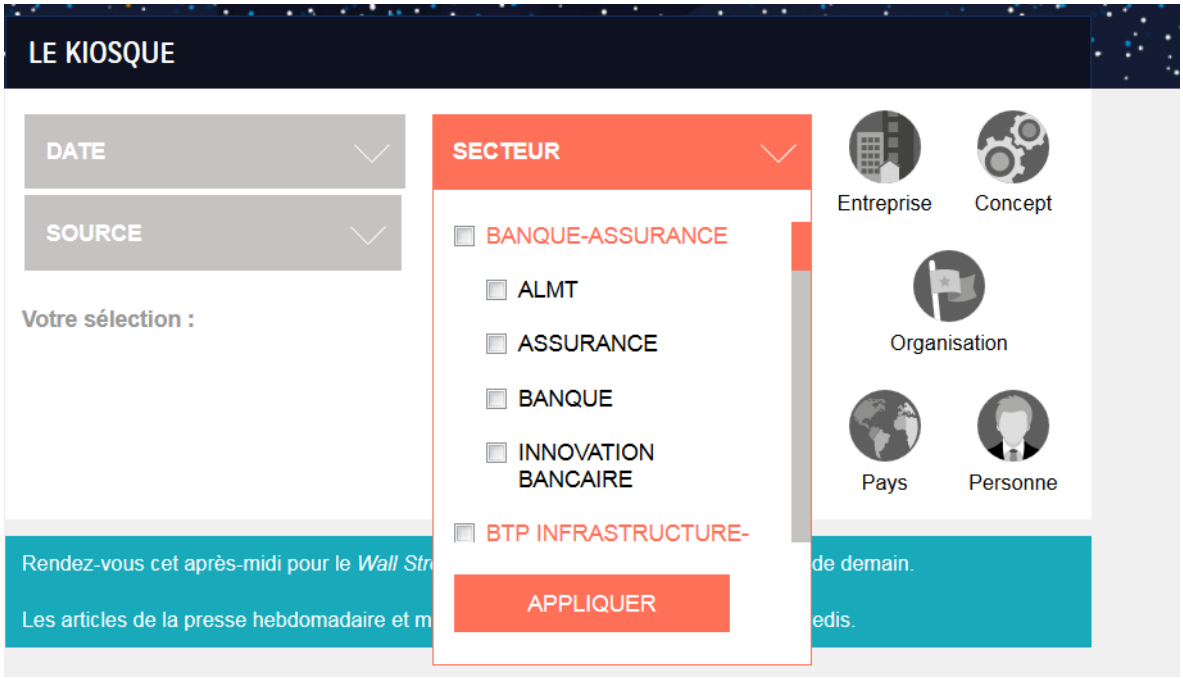

FIGURE 2.5 – Exemple de secteurs dans LEOnard

La mise en place de ces secteurs qui est faite par des travaux précédents seront détaillés dans le chapitre suivant. Les ressources et des matériels n'étant plus accessibles après la migration des outils et la réorganisation d'Expert System, il est nécessaire de refaire un système pour la catégorisation sectorielle. Plusieurs approches sont possibles pour ce faire : le thésaurus, la hiérarchie de concepts, l'apprentissage automatique, etc. Étant donné que le nombre de secteurs est assez important (52) et que le regroupement des termes est entremêlé, la construction de thésaurus et de règles est possible mais extrêmement fastidieuse. De plus, le système en vigueur a été développé par l'apprentissage automatique et le résultat est satisfaisant. Nous avons donc choisi d'explorer la piste de l'apprentissage automatique supervisé en utilisant des technologies indépendantes *open source* pour refaire le système afin que les améliorations et les mises à jour éventuelles selon des besoins ultérieurs puissent s'effectuer.

Deuxième partie

Expérimentations

CHAPITRE 3

SYSTÈME EXISTANT

Sommaire

3.1 Introduction 29
3.2 Le corpus et les secteurs 29
3.3 L'outil et les méthodes utilisées 32
3.4 Conclusion 34

3.1 Introduction

Ce chapitre présente le système de classification en vigueur dans la plate-forme LEOnard réalisé lors de précédents stages en précisant le corpus, les outils et les méthodes utilisés.

3.2 Le corpus et les secteurs

Le corpus d'apprentissage utilisé provient de la base de données interne GIMADOC. Ce sont des documents archivés par les documentalistes des Études Économiques et de Risk Management. Des descripteurs sont attribués par les documentalistes aux documents comme métadonnées. L'équipe a d'abord créé un plan de classement à partir des ces descripteurs, et ensuite affecté les documents aux catégories avec les experts et analystes de BNP Paribas. Le corpus d'apprentissage est au format TMX (format propriétaire XML d'Expert System). Chaque élément (document) du fichier xml contient sa catégorie ainsi que le chemin vers le texte du document en HTML. Chaque document a une seule catégorie. Voici les exemples pour le document HTML et le fichier TMX :

```
<title>Image 1</title>
</head>
<body>
<PRE><FONT color=blue><B>descripteur:</B></FONT> <FONT color=black>PECHE
</FONT></PRE>
<HR color=red>

<p><font color="#373434" size="1" face="Prelo-Extra"><b>CLAIRE GALLEN</b
></font></p><p><font color="#373434" size="1" face="Prelo">BRUXELLES</
font> <font color="#CE3133" size="1" face="PreloSlab-Extra"><b>UNION
EUROPÉNNE </b></font><font color="#373434" size="1" face="GlosaText-Semi
">Petit à petit, l'Europe avance vers un moratoire sur le thon rouge.
La Commission a recommandé hier que ce poisson très prisé des Japonais
soit classé parmi les espèces menacées, et donc d'en interdire le
commerce international dans le courant de l'année prochaine. Ce qui
gèlerait de fait toute pêche à grande échelle.</font> <font color="
#373434" size="1" face="GlosaText-Semi">Bruxelles s'attaque là à un
dossier complexe. Le thon rouge, pêché essentiellement en Méditerranée,
est massivement exporté vers le Japon, qui consomme 80 % des captures
mondiales malgré une forte hausse des prix. C'est d'ailleurs pourquoi </
font><font color="#373434" size="1" face="GlosaText-SemiMedium"><i>« il
y a peu de risque</i></font> <font color="#373434" size="1" face="
GlosaText-SemiMedium"><i>que l'on mange cette sorte de thon en Europe »
</i></font><font color="#373434" size="1" face="GlosaText-Semi">, selon
un spécialiste de la protection des espèces. Mais cet engouement
pourrait bien menacer la conservation de l'espèce.</font> <font color="
#373434" size="1" face="GlosaText-SemiMedium"><i>« Au cours des
soixante dernières années, la surpêche a provoqué un déclin grave des
stocks, on est aujourd'hui à 15 % des niveaux</i></font> <font color="
#373434" size="1" face="GlosaText-SemiMedium"><i>souhaités »</i></font><
font color="#373434" size="1" face="GlosaText-Semi">, a estimé hier le
commissaire européen à l'Environne-</font> <font color="#373434" size="1
" face="GlosaText-Semi">ment, Janez Potocnik. Le Parlement s'est déjà
prononcé pour un moratoire. La balle est à présent dans le camp des
```

FIGURE 3.1 – Exemple de fichier HTML

```
<?xml version="1.0" encoding="UTF-8"?>
 <tm xmlns:dc="http://purl.org/dc/elements/1.1/">

<doc id="68">
<dc:title>68</dc:title>
<categories><c>ENERGY-ENVIRONMENT/ENERGY/OIL</c></categories>
<features>
<ft f="1">/Metadata/COMPAGNIE PETROLIERE</ft>
<ft f="1">/Metadata/CONTRAT</ft>
<ft f="1">/Metadata/CHINE</ft>
<ft f="1">/Metadata/IRAN</ft>
</features>
<text><file format="html" path="
./Corpus-Gimadoc-Francais-Anglais/30197543446122.htm"/></text>
</doc>

<doc id="69">
<dc:title>69</dc:title>
<categories><c>INDUSTRY/AEROSPACE-AERONAUTICS-AIRLINE INDUSTRY</c></
categories>
<features>
<ft f="1">/Metadata/TRANSPORT AERIEN</ft>
<ft f="1">/Metadata/FUSION ACQUISITION</ft>
<ft f="1">/Metadata/ITALIE</ft>
<ft f="1">/Metadata/FRANCE</ft>
</features>
<text><file format="html" path="
./Corpus-Gimadoc-Francais-Anglais/30197543447274.htm"/></text>
</doc>
```

FIGURE 3.2 – Exemple de fichier TMX

[Jouannet, 2015] a modifié le secteur *High Tech* sur la base de travail de [Ma, 2014] pour qu'il soit plus clair et adapté à l'actualité. Pendant la constitution du corpus, seuls les documents dont la taille supérieure à 4ko (environ 450 mots) sont conservés. Le plan de classement sectoriel est construit sur 4 niveaux de profondeur et compte au total 52 catégories feuilles. Cependant, dans l'interface de LEOnard, seuls deux niveaux de classement sont affichés, soit 8 secteurs et 25 sous-catégories, pour des raisons ergonomiques. Le modèle automatique de catégorisation a été fait essentiellement par apprentissage automatique (3 catégories supplémentaires ont été générées par la suite avec une approche par thésaurus). Le plan de classement affiché dans LEOnard est présenté ci-dessous et le plan de classement sectoriel complet est présenté dans A.1.

1. BANKING_INSURANCE
   - 1.1 ALMT *
   - 1.2 BANKING
   - 1.3 BANKING INNOVATION *
   - 1.4 INSURANCE
2. BUILDING CIVIL ENGINEERING-REAL ESTATE
   - 2.1 BUILDING CIVIL ENGINEERING
   - 2.2 REAL ESTATE
3. ENERGY-ENVIRONMENT
   - 3.1 ENERGY
   - 3.2 ENVIRONMENT-WASTE MANAGEMENT
4. FOOD INDUSTRY-RETAILING-LUXURY-FASHION
   - 4.1 FOOD INDUSTRY
   - 4.2 RETAILING-LUXURY-FASHION AND TEXTILE
5. INDUSTRY
   - 5.1 AEROSPACE-AERONAUTICS-AIRLINE INDUSTRY
   - 5.2 AUTOMOTIVE
   - 5.3 CAPITAL GOOD
   - 5.4 CHEMISTRY-PHARMACY
   - 5.5 METALWORKING INDUSTRY-STEEL INDUSTRY
   - 5.6 ROAD AND RAIL AND GOODS TRANSPORT
   - 5.7 SHIPBUILDING AND SEA TRANSPORT
   - 5.8 WOOD PAPER-PACKAGING-FURNITURE
6. IT-DIGITAL-TELECOMS
   - 6.1 DIGITAL
   - 6.2 IT
   - 6.3 TELECOMS **
7. MEDIA-ADVERTISING
   - 7.1 ADVERTISING
   - 7.2 MEDIA
8. SERVICES-LEISURE ACTIVITIES
   - 8.1 LEISURE ACTIVITIES
   - 8.2 SERVICES

FIGURE 3.3 – Classement des catégories dans LEOnard [1]

1. Les secteurs suivis par * sont des secteurs classés par le thésaurus ; les secteurs suivis par ** sont des secteurs non classés dans notre système en raison de la manque de ressource

3.3 L'outil et les méthodes utilisées

L'apprentissage automatique sur les secteurs a été effectué avec l'outil « Category Workbench » (développé par Temis) permettant la classification automatique à partir du « vecteur sémantique » de documents. Le processus complet est précisé par les étapes suivantes :

— **Pré-traitement du corpus.** Le logiciel nécessite un corpus d'apprentissage dans un format spécifique (TMX) dans lequel les informations sont présentées comme desmétadonnés avec des balises, à savoir : l'identifiant du document, le titre du document, la catégorie associée au document, les descripteurs attribués au documents (*features*), le chemin vers le document ;

— **Le corpus est annoté par des cartouches de connaissances.** Plusieurs cartouches sont appelées : d'abord l'identification de la langue du document, ensuite la tokenisation, l'analyse morphologique, la désambiguïsation syntaxique ainsi que l'extraction de termes, etc. Les documents annotés sont stockés dans une base de données à laquelle se connecte « Category Workbench ». Le document est représenté par des termes qui sont sous la forme */Term/COMMON-NOUN/documents*, */Terms/VERB/voir*, */Terms/NP/laboratoire vétérinaire*, ... La catégorie de référence est sous la forme */Category/Insurance* ;

— **Apprentissage automatique statistique avec « Category Workbench ».** Les algorithmes d'apprentissage implémentés sont basés sur un naïve bayes aves différentes variantes : scalar et pachinko. Le principe de ces algorithmes consiste à comparer le vecteur du document au vecteur de chaque catégorie. Pour effectuer l'apprentissage, le corpus doit être divisé en deux : le corpus d'apprentissage et le corpus de test. Dans les expériences, 85% du corpus est aléatoirement choisi comme le corpus d'apprentissage et les 15% restant comme corpus de test. D'abord, chaque document d'apprentissage est transformé en un « vecteur sémantique » qui est composé par l'ensemble des paires *terme*/*fréquence* (dans logiciel, *Feature/Frequency*), et dans les expériences, les descripteurs (*features*) ayant le pos tagger « NP » (Noun Phrase)/(groupes nominaux), « PROPER-NAME » (nom propre) et « COMMON-NOUN » (nom commun) ont été retenus comme *features* parce qu'ils sont plus informatifs que les verbes et adjectifs. Les seuils minimums et maximums du nombre de descripteurs dans le corpus, dans une catégorie et dans un document sont fixés afin de déduire les bruits occasionnels. Un vecteur moyen pour chaque catégorie est calculé sur cette base ensuite. Le vecteur moyen pour une catégorie $W(C)$ correspond à :

$$W(C) = W(i,j) \times W(j,C) \tag{3.1}$$

dont $w(i,j)$ est le score de terme i dans le document j, $W(j,C)$ est le score de document j dans toute la catégorie C. $w(i,j)$ est calculé avec le « smoothed-frequency scoring » comme suit :

$$W(i,j) = \begin{cases} log(1 + Frequency(i,j)) & ,i \in Document(j) \\ 0 & ,i \notin Document(j) \end{cases} \tag{3.2}$$

pour normaliser les scores :

$$W_{normalized}(i,j) = \frac{W(i,j)}{norm(j)} \quad ,where \quad norm(j) = \sqrt{\sum_i W(i,j)^2} \tag{3.3}$$

Le poids du document dans la catégorie :

$$W(j,C) = log(1 + N(j)) \tag{3.4}$$

dont $N(j)$ est la taille du document, qui correspond au nombre de termes retenus dans le document. En conclusion, l'équation finale de vecteur moyen de la catégorie est :

$$W(C) = W(i,j) \times log(1 + N(j))/norm(j) \tag{3.5}$$

Tous les vecteurs sémantiques pour chaque catégorie sont stockés dans le fichier de modèle. Parmi les différentes options de l'algorithme, la méthode Scalar a été choisie parce que plus performante. Dans cette méthode, toutes les catégories sont considérées comme plates (sans hiérarchie) avec une normalisation au niveau de la longueur du texte. Voici un résumé des paramètres choisis :

Proportion du corpus d'apprentissage	85%
Descripteurs utilisés	Groupes Nominaux, Noms Communs, Noms Propres
Nombre minimal de documents dans une categorie	30
Frequence minimum des descripteurs dans le corpus	3
Frequence maximum d'un descripteur dans un document	10 000
Frequence minimum d'un descripteur dans un document	2
Frequence minimum des descripteurs dans une categorie	5
Nombre maximum de descripteurs	1 000 000

TABLE 3.1 – Paramètres du modèle d'apprentissage

— **Phase d'assignement.** Pour tester le modèle généré au corpus de test, le corpus de test est annoté et converti en vecteur de la même manière que le corpus d'apprentissage. Les distances entre ce vecteur et les vecteurs de chaque catégorie dans le modèle sont calculées. « Category Workbench » donne les scores selon les distances, la catégorie ayant le meilleur score, soit la distance la plus proche est attribuée au document. Enfin, pour le corpus de test ayant les références, « Workbench » donne trois sorties pour évaluer : *correct*, *missed* et *false*.

3.4 Conclusion

Dans ce chapitre, nous avons présenté le classifieur basé sur un processus d'apprentissage automatique existant avec lequel nous comparons notre propre classifieur. Ce modèle est développé avec le logiciel « Category Workbench ». Les noms propres, noms communs et groupes nominaux sont retenus comme *features* et transformés en vecteurs sémantiques pendant l'apprentissage pour représenter les documents et les catégories. La catégorie assignée à un nouveau document est la catégorie ayant la distance la plus proche au niveau du secteur sémantique avec celui du nouveau document.

CHAPITRE 4

LE NOUVEAU SYSTÈME BASÉ SUR DE L'OPEN SOURCE

Sommaire

4.1 Introduction **35**
4.2 Le corpus **35**
4.2.1 Les statistiques du corpus 35
4.2.2 Analyses du corpus 36
4.2.3 Limitations du corpus 40
4.3 Le processus de traitement **41**
4.4 Expériences **44**
4.4.1 Création de la classe vide 44
4.4.2 Ajout du corpus d'apprentissage 45
4.5 Conclusion **45**

4.1 Introduction

Ce chapitre présente le nouveau système de classification. Comme ce système est basé sur de l'apprentissage automatique, nous présentons en premier lieu le corpus d'apprentissage. Nous présentons ensuite le processus de traitement ainsi que les expériences faites pour rendre le système optimal.

4.2 Le corpus

4.2.1 Les statistiques du corpus

Le corpus utilisé pour l'apprentissage et les expériences est le corpus construit dans les travaux précédents, comme le décrit le chapitre 3. Notre corpus couvre 8 secteurs d'activité donnant lieu à 52 sous-catégories sur 4 niveaux. Il s'agit de documents en français et en anglais provenant de deux sources : une source interne issu de la base de données Gimadoc et une source de documents web. Les informations principales sont présentées dans la table 4.1 :

	Nombre de documents	Nombre de tokens	Tokens par document	Taille
EN	8600	8,074,938	940	54,5 Mo
FR	25654	20,602,794	803	149,1 Mo

TABLE 4.1 – Informations principales du corpus

La répartition du corpus en huit secteurs est présentée dans les tables suivantes :

	Français		Anglais	
Secteurs	Nombre de Documents	Pourcentage	Nombre de Documents	Pourcentage
ENERGY	2560	9,98 %	1520	17,67 %
AEROSPACE-AERONAUTICS-AIRLINE INDUSTRY	1308	5,10 %	590	6,86 %
RETAILING-LUXURY-FASHION AND TEXTILE	2841	11,07 %	379	4,41 %
MEDIA	1850	7,21 %	440	5,12 %
AUTOMOTIVE	1339	5,22 %	537	6,24 %
METALWORKING INDUSTRY-STEEL INDUSTRY	248	0,97 %	161	1,87 %
FOOD INDUSTRY	2412	9,40 %	392	4,56 %
INSURANCE	2261	8,81 %	288	3,35 %
BANKING	3716	14,49 %	2234	25,98 %
LEISURE ACTIVITIES	908	3,54 %	193	2,24 %
ROAD AND RAIL AND GOODS TRANSPORT	546	2,13 %	64	0,74 %
SERVICES	471	1,84 %	37	0,43 %
REAL ESTATE	759	2,96 %	201	2,34 %
CHEMISTRY-PHARMACY	1322	5,15 %	845	9,83 %
BUILDING CIVIL ENGINEERING	723	2,82 %	121	1,41 %
WOOD PAPER-PACKAGING-FURNITURE	689	2,69 %	67	0,78 %
CAPITAL GOOD	351	1,37 %	48	0,56 %
ADVERTISING	455	1,77 %	84	0,98 %
ENVIRONMENT-WASTE MANAGEMENT	315	1,23 %	79	0,92 %
SHIPBUILDING AND SEA TRANSPORT	263	1,03 %	55	0,64 %
DIGITAL	139	0,54 %	125	1,45 %
IT	178	0,69 %	140	1,63 %
MOYENNE	1166	4,55 %	391	4,55 %
TOTAL	25654	100,00 %	8600	100,00 %

TABLE 4.2 – Répartition du corpus

4.2.2 Analyses du corpus

Pour avoir une meilleurs connaissance du corpus et nous permettre de mieux évaluer les résultats du système de classification, nous avons procédé à une analyse textométrique (statistiques textuelles) à l'aide du logiciel TXM. Pour que les résultats soient les plus représentatifs, les analyses se font sur le niveau intermédiaire du classement, soit 22 catégories. Et comme le corpus complet est trop gros pour être traité par TXM, nous avons tiré au hasard 20% du corpus total pour chaque catégorie

en guise d'échantillons.

Tout d'abord, nous avons créé pour chaque langue une table lexique qui contient des fréquences de chaque mot dans chaque catégorie. Cela nous donne une première vue globale sur le corpus par catégorie. Nous présentons les dix termes les plus fréquents par catégorie dans les tableaux suivants :

Catégorie	Les 10 termes les plus fréquents
advertising	Internet, média(s), annonceurs, marketing, agence, publicitaires, Publicité, presse, croissance, Havas
aerospace-aeronautics-aireline inudustry	Air, compagnie, groupe, France, euros, Airbus, compagnies, marché, avions, Boeing
automotive	groupe, Renault, euros, marché, véhicules, constructeur, société, vente, automobile, voiture
Banking	font, gestion, euros, banque(s), marché, société, milliard, actifs, crédit, taux
Building civil engineering	euros, groupe, millions, construction, affaires, Eiffage, milliard, libre, chiffre,prix
capital good	euros, groupe, société, marché, affaires, millions, chiffre, entreprise, années, libre
chemistery-pharmacy	euros, groupe, société, produits, affaires, chiffre, médicaments, croissance, France,laboratoire
digital	données, Facebook, Data, Big, réseau(x), sécurité, entreprise(s), social, information, utilisateur(s)
energy	prix, production, gaz, dollars, société, euros, pétrole, énergie, France, EDF
environment-waste manangement	eau(x), déchets, marché, pays, millions, euros, émissions, traitement, carbone, gestion
food industry	marché, groupe, euros, produits, marque, affaires, chiffre, prix, production, gamme
insurance	assurance, marché, euros, assureurs, société, contrat, millions, gestion, risque, santé
IT	3D, cloud, impression, fabrication, Apple, production, application, Google, marché, entreprise
leisure activities	euros, millions, affaires, tourisme, chiffre, hôtel, jeux, produits, gamme, marque
medias	millions, euros, France, presse, société, Internet, films, marché, site, chaîne
metalworking industry-steel industry	Mittal, groupe, acier, Arcelor, ArcelorMittal, Steel, Lakshmi, mondial, production, tonnes
real estate	euros, immobilier, marché, prix, logement, hausse, taux, actifs, secteur, loyer
retailing-luxury-fashion and textile	groupe, marché, produits, France, marque, euros, magasins, ventes, luxe, croissance
road and rail and goods transport	transport, SNCF, groupe, affaire, service, activité, logistique, transporteurs, ferroviaire, français
services	euros, société, conseil, formation, marché, cabinets, entreprises, secteur, activité, travail
ship building and sea transport	conteneurs, trafic, euros, transport, tonne, Marseille, marché, portuaire, construction, bateaux
wood paper-packaging-furniture	marché, France, produits, papier, prix, gamme, production, bois, entreprise, ventes

TABLE 4.3 – Les 10 termes les plus fréquents par catégorie

Ensuite, la table lexique nous permet de faire l'AFC (Analyse Factorielle des Correspondances) sur la partition du corpus. Il s'agit d'une méthode statistique d'analyse des données qui admet en entrée la table de lexique et produit une carte qui montre la dépendance entre les mots et les catégories. Elle donne un point de vue différent et nous aide à comprendre les liens entre les mots et les catégories. Très grossièrement, les points dans le plan de l'AFC peuvent être considérés comme une représentation des catégories. Les figures suivantes montrent l'AFC sur le corpus :

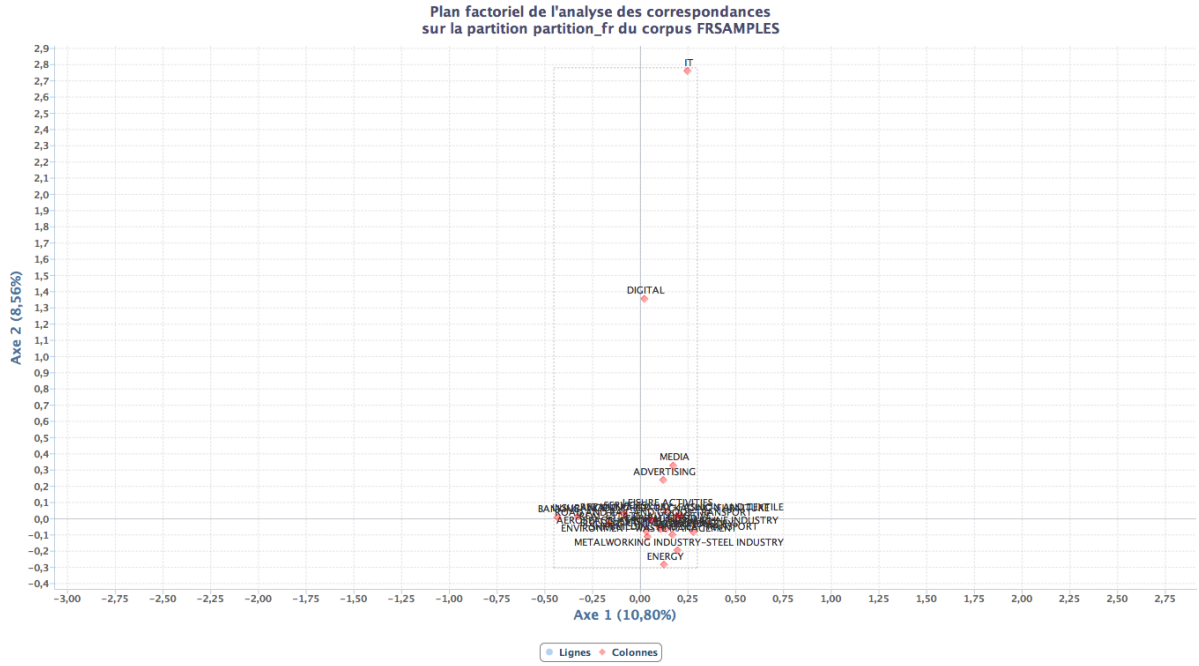

FIGURE 4.1 – Plan factoriel de l'analyse des correspondances pour le corpus français

La figure 4.1 présente le plan d'AFC pour le corpus français. Dans ce plan, les catégories IT, Digital, media, advertising se sont bien distinguées, et les autres catégories sont plutôt proches les unes des autres. Le plan d'AFC pour le corpus anglais est présenté dans la figure suivante :

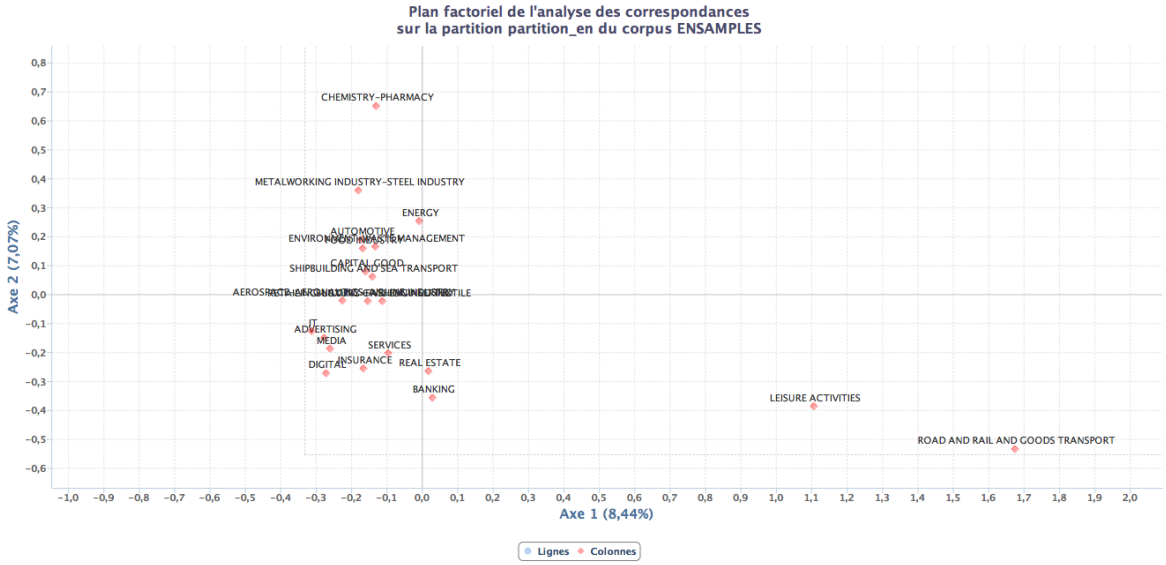

FIGURE 4.2 – Plan factoriel de l'analyse des correspondances pour le corpus anglais

Par rapport au corpus français, le corpus anglais est beaucoup plus éclaté, les catégories qui se distinguent le plus sont "leisure activities" et "road and rail and goods transport". De plus, la table de lexique nous permet de calculer les spécificités de chaque mot pour chaque catégorie. Il s'agit de calculs statistiques indiquant si les oc-

currences d'un mot paraissent en surnombre (valeur positive) ou en sous-effectif (valeur négative) dans chaque catégorie. L'analyse des specificites permet donc de porter un jugement sur la frequence de chaque mot dans chaque catégorie. Dans chaque catégorie, les mots ayant la valeur de spécificité plus grande sont considérés comme étant les plus discriminants par rapport à l'ensemble du corpus. Nous les considérons donc comme des descripteurs linguistiques importants de chaque catégorie. Les tableaux suivants présentent les dix mots les plus spécifiques (ayant les plus grandes valeurs de spécificité) dans chaque catégorie pour les deux langues :

Catégorie	Les 10 termes les plus spécifiques
advertising	annonceurs,média(s), Havas, Publicis, publicité, Internet, publicitaire(s), agence(s), marketing, affichage
aerospace-aeronautics-aireline inudustry	Boeing, EADS, Airbus, Air, Compagnie, Airways, Alitalia, avions, passagers, aérien
automotive	constructeur, voitures, PSA, véhicules, Renault, automobile, Ghosn, Volkswagen, Peugeot, Toyota
Banking	crédit, gérants, actifs, banque, gestion, fonds, hedge, funds, BNP, encours
Building civil engineering	Eiffage, Sacyr, Lafarge, BTP, ciment, construction, Vinci, WC, Fayat, cimentier
capital good	Nexans, Rexel, Alstom, Pubert, batteries, Schneider, Saft, câbles, Electric, machines
chemistery-pharmacy	médicaments, labaratoire, génériques, pharmaceutique, vaccins, Sanofi, chimie, Merck, Novartis
digital	données, Facebook, Data, Big, Twitter, darknet, sécurité, réseaux, attaques, utilisateurs
energy	électricité, EDF, pétrole, gaz, énergie, Suez, or, GDF, réserves, nucléaire
environment-waste manangement	eau, déchets, émissions, carbone, Veolia, recyclage, Kyoto, traitement, CO2, assainissement
food industry	vin, lait, rayon, viande, fruit, légumes, marque, tonnes, café, boissons
insurance	assurance, vie, assureur, contrats, Axa, mutuelles, dommages, Md, prévoyance, Generali
IT	3D, cloud, impression, Apple, Android, Windows, fabrication, smartphone, additive, IOS
leisure activities	tourisme, hôtels, jouets, jeux, Smoby, Accor, hôtellerie, vacances, clubs,étoiles
medias	films, cinéma, Monde, audience, chaînes, presse, musique, télévision, numérique, Internet
metalworking industry-steel industry	Mittal, acier, Arcelor, ArcelorMittal, Steel, Lakshmi, Gandrange, sidérurgiste, Corus, sidérurgie
real estate	immobilier, logement, Nexity, loyers, foncière, Gecina, Colonial, bureaux, SIIC, m2
retailing-luxury-fashion and textile	magasins, marque, luxe, Carrefour, enseigne, Casino, mode, textile, griffe, lingerie
services	formation, cabinets, EPI, conseil, consultants, Vedior, emploi, avocats, audit, stagiaire
ship building and sea transport	ports, conteneurs, portuaire, bateaux, fluvial, trafic, Havre, navires, EVP, manutention
wood paper-packaging-furniture	papier, bois, carton, palettes, linge, emballage, Mt, PNP, pâte, papetiers

TABLE 4.4 – Les 10 termes les plus spécifiques par catégorie (français)

Catégorie	Les 10 termes les plus spécifiques
advertising	advertising, Google, Microsoft, Yahoo, search, advertisers, Doubleclick, online, Saatchi, internet
aerospace-aeronautics-aireline inudustry	Air, airline, carrier, aircraft, Airways, flights, Boeing, Alitalia, routes, Airbus
automotive	Ford, car, auto, Chrysler, GM, sales, vehicles, Toyota, Motor, VM
Banking	Islamic, bank, funds, banking, equity, private, credit, mortage, loans, investment
Building civil engineering	projects, PPP, construction, roofing, cement, infrastructure, bituminous, bitumen, road, shingles
capital good	fiber, Yamazaki, cable, Agco, Erie, Richenhagen, machines, Sonepar, Internet, cables
chemistery-pharmacy	drug, chemical, tonne, ICIS, ethylene, BASF, pharmaceutical, PET, Bayer, generic
digital	data, cybersecurity, cyber, security, organizations, hackers, information, mobile, malware, social
energy	gaz, oil, energy, LNG, power, nuclear, production, barrels, BHP, electricity
environment-waste manangement	carbon, emissions, Kyoto, CDM, permits, cap-and-trade, allowances, water, ETS, credits
food industry	Busch, food, beer, wine, rice, drinks, Inbev, farmers, Anheuser, Champagne
insurance	insurance, Ping, insurers, life, KBC, Re, An, Lloyd's, AIG, Aviva
IT	cloud, 3D, Windows, your, Microsoft, printing, smartphone, PC, printer, tablet
leisure activities	hotel, rooms, gambling, tourism, casinos, Macau, gaming, Jurys, Schrager, toys
medias	TV, music, Murdoch, content, News, Warner, iTunes, Viacom, media, digital
metalworking industry-steel industry	steel, mill, iron, Mittal, Posco, tpy, slab, Siemens, bells, steelmaker
real estate	property, estate, Retis, Reit, real, Zhang, Kwok, real-estate, office, Shenzhen
retailing-luxury-fashion and textile	fashion, goods, retail, luxury, Loro, retailers, brand, Ports, vicuna, Carrefour
services	Adecco, Audit, Brothers, Chuo, firms, japan, Four, assisted-living, Quigley, accounting
ship building and sea transport	Hutchison, container, boat, yacht, Port, terminal, SIPG, marine, Mombasa
wood paper-packaging-furniture	PAV, Packaging, PP, plastic, materials, paper, film, solution, pack, airless

TABLE 4.5 – Les 10 termes les plus spécifiques par catégorie (anglais)

En analysant les termes les plus spécifiques pour chaque catégorie, nous avons une vue globale sur toutes les catégories. Selon les termes relevés, les secteurs d'activité de la plupart des catégories sont explicites pour les deux langues, sauf pour les catégories « capital goods » et « services ». Les documents de ces deux catégories sont probablement plus difficiles à classer que les autres. De plus, au niveau des relations entre les catégories, nous trouvons qu'il y a des catégories qui recouvrent des notions proches, par exemple, les catégories « building civil engineering », « ship building and sea transport » et « automotive » ainsi que les catégories « advertising », « digital » et « media ». Les documents de ces catégories risquent d'être plus difficiles à classer par le système.

4.2.3 Limitations du corpus

Après avoir analysé notre corpus, nous avons remarqué aussi quelques caractéristiques au niveau du format qui pourraient influer négativement sur les résultats de catégorisation.

La première limitation est que certains documents PDF sont mal numérisés. La présence de signes spéciaux et d'entités html, la segmentation mal faite, les fautes de numérisation, nous demandent de travailler davantage sur le nettoyage et les prétraitements de corpus.

La deuxième limitation est que le nombre de documents dans chaque catégorie ne sont pas équilibrés. Dans certaines grandes catégories comme Banque et Assurances, les documents sont beaucoup plus nombreux que les documents dans des catégories comme Services-Leisure activities. Les résultats de l'apprentissage pourraient être moins performants par rapport à ceux de grandes catégories.

La troisième limitation est que la plupart des documents du corpus proviennent de la presse de l'année 2007 sauf les secteurs « IT-Digital-Télécoms ». Comme il s'agit d'un corpus d'il y a plus de dix ans, le système entraîné de ce corpus risque d'avoir des difficultés à classer des articles de nos jours.

4.3 Le processus de traitement

Le processus de traitement pour la classification suit le schéma présenté dans la figure 4.3 . Des expériences seront faites pour améliorer le programme afin d'avoir des résultats optimaux (cf 4.4).

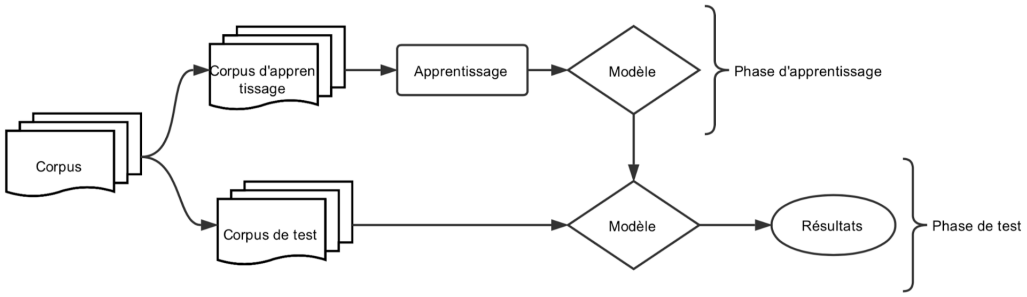

FIGURE 4.3 – Schéma général de l'approche automatique pour la classification de textes

Nous avons choisi Python comme langage de programmation pour implémenter ce processus. Il est facile à manipuler et à interpréter et les bibliothèques open sources sont très riches. Dans notre travail, nous nous appuyons principalement sur le package d'apprentissage automatique *Scikit-Learn* et des bibliothèques de TAL comme *NLTK* et *Tree tagger*.

Dans cette section, nous présentons les méthodes utilisées et les implémentations pour chaque étape en nous laissant le plus de liberté possible afin d'effectuer des expériences à plusieurs niveaux et nous donner davantage de leviers d'améliorations. Les résultats des expériences seront présentés dans le chapitre suivant pour évaluer les performances.

La phase d'apprentissage est composée de quatre étapes :
— les pré-traitements des corpus ;
— la sélection des *features* ;
— la représentation vectorielle du corpus ;

— les algorithmes et les paramétrages.

Les pré-traitements du corpus. Parmi les pré-traitements réalisés, il y a la mise au format du corpus qui est contrainte par l'utilisation de Python et des bibliothèques. A partir des fichiers HTML et XML, nous avons extrait le contenu et sa catégorie pour chaque document au format txt sans tenir compte des balises [1]. Comme les textes proviennent de fichiers HTML, des caractères spéciaux et des entités HTML doivent être décodé [2]. Une fois les textes bruts et les catégories obtenus, nous devons les nettoyer pour diminuer le plus possible les bruits et réduire les calculs. Dans cette étape, les mots vides (stop words), les signes de ponctuation, les chiffres [3] et des mots inférieurs à 3 caractères et supérieurs à 15 caractères sont supprimés du corpus afin d'éliminer les scories ou les erreurs de transcription.

La sélection des features. Il s'agit de la sélection des descripteurs linguistiques (features) à l'aide des annotations morphosyntaxiques pour représenter de manière optimale les documents. D'un point de vue linguistique, le français et l'anglais (les deux langues de notre corpus) sont des langues flexionnelles. Il existe donc la flexion (les modifications subies par le signifiant des mots d'une langue flexionnelle pour dénoter les traits grammaticaux voulus). Pour que le système tienne compte de la sémantique au lieu de la forme des mots, nous avons effectué une racinisation (stemming) pour tout le corpus. Ensuite l'analyse du corpus montre que la plupart des termes les plus spécifiques dans chaque catégorie sont des nominaux. Les nominaux sont généralement les plus informatifs et signifiants par rapport aux autres POS (*Part Of Speech*). Nous avons donc appliqué le pos tagger pour extraire les noms et le chunking pour extraire des groupes nominaux sur le texte brut. Avec le texte initial, les stems et les (groupes) nominaux donnent trois représentations différentes comme features.

La représentation vectorielle du corpus. Afin d'effectuer l'apprentissage, les données textuelles doivent être transformées au format numérique pour que la machine puisse les traiter. Il existe deux représentations numériques : la représentation distribuée, comme le BOW (sac de mots), et la représentation discrète, comme Word2Vec. Pour une tâche de la classification classique comme la nôtre, la représentation distribuée est performante et beaucoup moins compliquée. Nous l'avons donc choisie pour notre travail. Dans une représentation distribuée, chaque contenu est ainsi représenté par un vecteur v, dont la dimension correspond à la taille du vocabulaire. Chaque élément v_i du vecteur v consiste en un poids associé au terme d'indice i. La composante du vecteur représente donc le poids du terme i dans le document. Pour calculer les poids, nous avons utilisé la méthode TF-IDF (*Term Frequency-Inverse Document Frequency*). Le TF-IDF permet d'évaluer l'importance d'un terme contenu dans un document par rapport à un corpus. Le poids augmente proportionnellement

1. Avec les bibliothèques *os*, *xml.etree.ElementTree* et *BeautifulSoup*
2. Avec la bibliothèque *html*
3. Avec les bibliothèques *String* et *html*

au nombre d'occurrences du mot dans le document. Les formules appliquées sont :

$$tf_{ij} = \frac{n_{ij}}{\sum_k n_{kj}} \tag{4.1}$$

$$idf_i = log \frac{|D|}{|d_j : t_i \in d_j|} \tag{4.2}$$

$$tfidf_{i,j} = tf_{i,j} * idf_i \tag{4.3}$$

où n_{ij} est la fréquence d'un terme i dans le document d_j ; $|D|$ est le nombre total de documents dans le corpus ; $|d_j : t_i \in d_j|$ est le nombre de documents où le terme t_i apparaît (c.-à-d. $n_{ij} \neq 0$) [4].

Les algorithmes et les paramétrages. Nous avons comparé des algorithmes supervisés de la classification multi-classes, y compris les méthodes classiques et émergentes, pour trouver l'algorithme et les paramétrages les plus performants.

— **KNN (K-plus proches voisins).** L'idée est de chercher les k documents qui sont les plus « proches » (au sens de la distance vectorielle) de la nouvelle donnée et d'associer à celle-ci la classe majoritaire au sein de ces k voisins. Le paramètre crucial est donc le nombre de K et il est fixé à 5 dans nos expériences ;

— **SVM (Support Vector Machines) ou Séparateurs à Vastes Marges.** Un SVM est un séparateur qui vise à séparer au maximum les données d'apprentissage qui se trouvent dans un espace vectoriel en classes prédéfinies avec des hyperplans. Afin de trouver les meilleurs hyperplans possibles, des astuces comme « pénalité » et « fonction du noyau » sont utilisées. La pénalité consiste à prendre en compte les données mal classées pour ajuster le modèle et la fonction du noyau permet de faciliter les calculs compliqués. Dans nos expériences, nous avons utilisé l'algorithme SVC (*C-Support Vector Classification*) avec C = 1 qui est le paramètre de pénalité et le « rbf » la fonction du noyau ;

— **Arbres de décision.** Il s'agit d'un modèle prédictif qui représente l'application entre les données et leurs catégories en forme d'arbre. Pour construire un arbre performant, il faut trouver dans chaque noeud l'attribut le plus discriminant. Différents critères sont possibles pour l'évaluer : Gain d'information, Impureté Gini, etc. Dans nos expériences, nous avons utilisé l'impureté Gini comme critère ;

— **Naïve Bayes.** Il s'agit d'un modèle probabiliste qui consiste à assigner à la nouvelle donnée la classe qui détient la plus grande probabilité de prédiction

4. L'implémentation se réalise vec le package *Scikit Learn*. Les méthodes *count vectorizer* et *tfidf vectorizer* sont utilisées pour transformer les termes en valeurs *tfidf*. La méthode *LabelEncoder* pour encoder les catégories (les transformer en valeurs numériques)

parmi toutes les classes. Dans nos expériences, nous avons utilisé l'algorithme bayésien naïf multinomial qui prend comme l'hypothèse que la distribution des features respecte la loi multinomiale ;

— **SGD Classifiers (Stochastic Gradient Descent).** Il s'agit de classifieurs linéaires (SVM, la régression logistique, etc) avec l'apprentissage SGD. Dans nos expériences, nous avons utilisé le SGD Classifier à base de SVM avec la loss fonction de Hinge (la loss fonction privilégiée aux SVM) et la pénalité L2 (la méthode standard de régularisation pour les SVM linéaires) ;

— **LightGBM (Light Gradient Boosting Machine).** LightGBM est un gradient boosting framework basé sur un algorithme d'arbre de décision, utilisé pour le ranking, la classification et de nombreuses autres tâches d'apprentissage automatique. Il fait partie du projet DMTK (*Microsoft Distributed Machine Learning Toolkit*) de Microsoft. Comme il s'agit d'un algorithme de boosting, le paramètre le plus important est le type de boosting. Dans nos expériences, nous avons utilisé GBDT (**Gradient Boosting Decision Tree**) comme type de boosting.

4.4 Expériences

Nous avons effectué plusieurs expériences pour construire notre système de classification. Au niveau de la sélection des descripteurs linguistiques, nous avons retenu trois formes comme features pour représenter une texte :

— le texte brut sans sélection ;
— les stems du texte ;
— tous les noms et les groupes nominaux du texte.

Au niveau de la représentation vectorielle du corpus, nous avons fait varier les deux pondérations de features : l'occurrence des termes dans les documents, le poids TF-IDF des termes dans les documents. Et pour les algorithmes de classification, nous avons testé les différents algorithmes.

D'autres expériences ont été par ailleurs réalisées pour adapter davantage le système à la situation de production réelle, tel que présenté ci-après :

4.4.1 Création de la classe vide

Dans le corpus d'apprentissage, chaque document correspond à une catégorie. Le modèle entraîné prédit donc la catégorie pour chaque nouveau document d'entrée. Cependant, dans la situation réelle, LEOnard reçoit tous les jours des centaines d'articles de presse qui ne font pas forcément partie d'un secteur d'activité précis. C'est le cas par exemple des articles traitant de politique. Si nous y appliquons directement le modèle entraîné, tous les documents seront classés dans au moins un secteur même si certains documents n'appartiennent à aucun secteur. Dans ce cas-là, il est

nécessaire de créer pour ces documents une « classe vide » gcomme premier filtrage de notre système.

L'idée principale consiste à comparer la distance entre le nouveau document et les documents dans chaque catégorie. D'abord, nous avons regroupé tous les documents dans la même catégorie et les avons transformé en vecteurs à l'aide de la méthode TF-IDF. Le vecteur moyen de tous les vecteurs est calculé comme le vecteur central de cette catégorie. Ensuite, nous supposons que la distribution de documents dans la même catégorie respecte la loi normale. Le vecteur central est considéré comme l'espérance μ et la portée de cette catégorie est fixé à $\mu \pm \delta$ (l'écart type). Enfin, la distance entre le nouveau document et le vecteur central de chaque catégorie sera calculées, si la distance est supérieure à la portée de toutes les catégories, le document ne doit appartenir à aucun des secteurs et être classé dans la « classe vide ». A l'inverse, si la distance est inférieure, le document sera classé dans une des catégories.

4.4.2 Ajout du corpus d'apprentissage

Comme mentionné dans les limitations du corpus, la plupart des documents dans le corpus sont récoltés dans la presse de 2007. Le classifieur entraîné avec ces documents risque d'être moins performant et moins adéquat pour les données actuelles, surtout pour les catégories autre que « IT-Digital-Télécoms ». Pour optimiser le résultat, nous avons collecté manuellement pour chaque catégorie 5 à 10 documents récents (03/2018 - 09/2018) et les avons ajoutés dans le corpus d'apprentissage. Comme la proportion de ces documents ajoutés dans le corpus est très petite par rapport au corpus initial, nous avons associé une pondération plus grande à ces documents pour accentuer leur importance pendant l'apprentissage du modèle. Les pondérations sont calculées selon la proportion de nouveaux documents dans chaque classe pour avoir une influence équilibrée sur le modèle.

4.5 Conclusion

Dans ce chapitre, nous avons présenté le processus de construction de notre propre système avec des modules open source. D'abord, nous avons analysé notre corpus avec des outils textométriques afin d'avoir une connaissance globale de notre corpus. Ensuite, nous avons présenté la méthodologie pour construire le classifieur. Enfin, nous avons présenté les expériences permettant d'optimiser les résultats.

CHAPITRE 5

RÉSULTATS ET ÉVALUATIONS

Sommaire

5.1 Introduction **47**
5.2 Les mesures d'évaluation classiques **48**
5.2.1 Qu'est-ce que l'évaluation ? 48
5.2.2 Les indicateurs et mesures 48
5.3 Évaluations sur le corpus de test **49**
5.3.1 Résultats du système existant 49
5.3.2 Résultats du nouveau système et les expériences 50
5.3.3 Validation croisée 51
5.3.4 Résultats du nouveau système après l'ajout du corpus d'apprentissage 54
5.4 Évaluations sur les nouveaux documents **55**
5.4.1 Matrice de confusion 56
5.5 Insuffisances des mesures classiques **58**
5.6 Mesures d'évaluation complémentaires **58**
5.6.1 Les demandes de LEOnard 58
5.6.2 Une nouvelle mesure adaptée : Leo-Score 59
5.6.3 Les critères complémentaires 60
5.7 Conclusion **61**

5.1 Introduction

Évaluer les performances d'un système de classification est un enjeu de grande importance. Dans ce chapitre, nous présentons d'abord les mesures d'évaluation classique pour la classification. Nous comparons ensuite les résultats des deux systèmes en appliquant ces mesures. Cependant elles se révèlent insuffisantes, c'est pourquoi nous proposons des métriques complémentaires permettant de donner une évaluation plus complète de notre modèle de classification.

	Correct	Non correct	Total
Proposé par le système	Vrais Positifs (VP) : a	Faux Négatifs (FN) : b	a+b
Non proposé par le système	Faux Positifs (FP) : c	Vrais Négatifs (VN) : d	c+d
	a+c	b+d	a+b+c+d = N

TABLE 5.1 – Tableau de contingence de classification

5.2 Les mesures d'évaluation classiques

5.2.1 Qu'est-ce que l'évaluation ?

L'évaluation consiste à mesurer la différence entre un résultat attendu et un résultat obtenu. Pour cela, il faut disposer de données de référence, pour lesquelles le résultat attendu du programme est connu et validé par des humains. Ce type de ressource est appelé un *Gold Standard*. Le résultat obtenu est un ensemble de prédictions générées pour des nouvelles données par le modèle après la phase d'apprentissage. Les métriques associées à l'évaluation sont en général des indicateurs scalaires ou multicritères compris entre 0 et 1 pour en faciliter l'interprétation.

5.2.2 Les indicateurs et mesures

Un système qui utilise le tableau de contingence ci-dessous permet la différenciation des erreurs selon chaque classe en vue d'évaluer un classifieur :

Définissons maintenant plusieurs indicateurs de mesure de manière formelle :

$$L'exactitude(accuracy) = \frac{VP + VN}{N} = \frac{a + d}{N} \tag{5.1}$$

$$precision = \frac{VP}{VP + FN} = \frac{a}{a + b} \tag{5.2}$$

$$Rappel = \frac{VP}{VP + FP} = \frac{a}{a + c} \tag{5.3}$$

Une représentation ensembliste est plus parlante, l'exactitude correspond à la proportion de documents bien classés pour chaque catégorie ; la précision indique la proportion du nombre de documents correctement classés d'une classe par rapport au nombre total de documents classés dans cette classe ; le rappel mesure la proportion du nombre de documents correctement classes d'une classe par rapport au nombre total de documents de référence appartenant à cette classe.

La F-Mesure a été proposée pour équilibrer la précision et le rappel. Il s'agit d' un indicateur de synthèse qui correspond à une moyenne harmonique de la précision et du rappel et elle est considérée comme un résultat global. Le pa-

ramètre β permet de pondérer la précision ou le rappel et vaut généralement 1 [Nakache and Métais, 2005].

$$Mesure - F = \frac{((1+\beta^2) * Precision * Rappel)}{((\beta^2 * Precision) + Rappel)} \quad , avec \quad \beta^2 = 1 \tag{5.4}$$

Les mesures listées ci-dessus sont utilisées pour évaluer la performance d'une ou de chacune des classes. Pour calculer la précision globale (ou le rappel) d'un algorithme sur toutes les classes pour la classification multi-classes, il existe deux méthodes reconnues : la Macro-Moyenne et la Micro-Moyenne. La Macro-Moyenne consiste à calculer d'abord la précision (ou le rappel) pour chaque classe, puis à faire la moyenne de celles-ci pour calculer les moyennes globales. Alors que la Micro-Moyenne consiste à calculer d'abord les totaux d'a, b, c et d pour toutes les classes, puis à utiliser ces totaux pour calculer la précision (ou le rappel) globale. Il existe une distinction importante entre ces deux types de calcul de la moyenne : le calcul de la Micro-Moyenne donne un poids égal à chaque document tandis que la Macro-Moyenne donne un poids égal à chaque classe.

5.3 Évaluations sur le corpus de test

Dans nos expériences, nous avons d'abord utilisé les mêmes ressources et les mêmes mesures pour que les résultats du nouveau système soient comparables avec ceux du système existant. Nous avons en plus, pour faire la moyenne de ces mesures et donner une résultat global, efffectué une Macro-Moyenne pondérée par le support (le nombre de documents vrais pour chaque classe) qui tient compte du déséquilibre des classes.

5.3.1 Résultats du système existant

Le système existant a utilisé le corpus Gimadoc : 85% des documents comme corpus d'apprentissage et 15% comme corpus de test pour les évaluations. Comme nous l'avons présenté dans le chapitre 3, le secteur *High Tech* a été modifié en *IT-Digital-Télécoms* pour qu'il soit plus clair et adapté à ce qui est présenté dans LEOnard. Les résultats du système existant sont présentés dans le tableau suivant :

Source	Gimadoc			IT-Digital-Télécoms			Total
Langue	FR	EN	Total	FR	EN	Total	
Précision	82%	78%	80%	N/A	N/A	75,9%	**76,4%**
Rappel	92%	85%	88,5%	N/A	N/A	72,8%	**80,65%**
F1-Mesure	87%	81%	84%	N/A	N/A	74,35%	**79,17%**

TABLE 5.2 – Résultats du système existant

5.3.2 Résultats du nouveau système et les expériences

Nous utilisons d'abord le même corpus et la même proportion du corpus d'apprentissage et de test pour entraîner notre système et l'évaluer. Dans le chapitre précédent, nous avons expliqué notre choix de de laisser ouverts un grand nombre de paramètres afin de pouvoir les faire varier au sein de plusieurs expériences.

Au niveau de la sélection des *features*, nous avons gardé trois représentations pour comparer les résultats entre eux, à savoir, les textes bruts sans sélection, les stems de tous les mots dans le texte, les noms et les groupes nominaux dans le texte. Chaque représentation est transformée par la suite en vecteurs dans un VSM (*Vector Space Model*) avec les valeurs de TF-IDF. Nous présentons dans les tableaux ci-dessous les résultats générés par les différents algorithmes avec des *features* différents pour deux langues.

Les résultats en utilisant les textes bruts après les nettoyages :

Algorithmes	Précision		Rappel		F1-Mesure	
	FR	EN	FR	EN	FR	EN
KNN	90%	90%	89%	88%	88%	88%
Naive Bayes	89%	85%	88%	86%	87%	83%
Decision Tree	70%	74%	70%	75%	70%	74%
SVC	**95%**	92%	**95%**	92%	**95%**	91%
SGD	**95%**	**95%**	**95%**	**95%**	**95%**	**94%**
LightGBM	92%	90%	92%	91%	92%	90%

TABLE 5.3 – Résultats après les nettoyages

Les résultats en utilisant les stems :

Algorithmes	Précision		Rappel		F1-Mesure	
	FR	EN	FR	EN	FR	EN
KNN	92%	91%	92%	91%	92%	91%
Naive Bayes	91%	87%	91%	88%	90%	86%
Decision Tree	71%	74%	71%	74%	71%	74%
SVC	95%	93%	95%	93%	**95%**	92%
SGD	**96%**	**95%**	**96%**	**95%**	**95%**	**95%**
LightGBM	92%	90%	92%	90%	92%	90%

TABLE 5.4 – Résultats avec les stems

Les résultats en utilisant les groupes nominaux :

Algorithmes	Précision		Rappel		F1-Mesure	
	FR	EN	FR	EN	FR	EN
KNN	89%	91%	89%	91%	89%	90%
Naive Bayes	91%	88%	91%	89%	91%	87%
Decision Tree	68%	73%	68%	74%	68%	73%
SVC	93%	93%	93%	93%	**93%**	92%
SGD	**94%**	**95%**	**94%**	**95%**	**93%**	**95%**
LightGBM	90%	90%	90%	91%	90%	90%

TABLE 5.5 – Résultats avec les groupes nominaux

Après avoir comparé les résultats donnés dans les tableaux ci-dessus, nous avons constaté que la représentation en stem donne de meilleurs résultats parmis les trois représentations de features. Et au niveau des algorithmes, SVC et SGD sont plus performants que les autres. En observant le temps de calcul de l'exécution des algorithmes, SVC a pris beaucoup plus de temps (plus de 30 minutes) que SGD (moins d'une minute) pour une seule validation. Nous avons donc retenu la représentation en stem comme les features et SGD comme algorithme pour les expériences ultérieures.

Nous comparons les résultats de notre système avec les paramètres retenus avec le système existant dans le tableau suivant. Nous pouvons en conclure que notre système donne un meilleur résultat sur les mesures de précision, rappel et F1-Mesure pour le corpus Gimadoc.

	Précision	Rappel	F1-Mesure
Système existant	76,4%	80,65%	79,17%
Nouveau système	**95,5%**	**95,5%**	**95%**

TABLE 5.6 – Résultats des deux systèmes

5.3.3 Validation croisée

La méthode de la division du corpus en corpus d'apprentissage et corpus de test correspond à une validation simple est le modèle entraîné risque d'avoir un effet d'*overfitting* car elle repose sur un choix arbitraire de découpage entre échantillons d'apprentissage et de validation (dans notre cas, le corpus d'apprentissage et de test). Pour un résultat plus stable, nous avons fait une moyenne sur plusieurs découpages, ce que l'on appelle la validation croisée. Elle désigne le processus qui permet de tester la précision prédictive d'un modèle dans un échantillon test (parfois aussi appelé échantillon de validation croisée) par rapport à la précision prédictive de l'échantillon d'apprentissage à partir duquel le modèle a été développé. Elle consiste à procéder de la façon suivante :

— diviser l'ensemble des exemples en k sous-ensembles disjoints (ici k = 10) ;
— réaliser k opérations d'apprentissage/évaluation distinctes : chaque opération consiste à prendre $k-1$ des sous-ensembles pour l'apprentissage, et le $k^{ème}$

sous-ensemble restant pour le test ;
— faire la moyenne les k évaluations effectuées comme mesure de qualité.
Le processus est schématisé dans la figure ci-dessous :

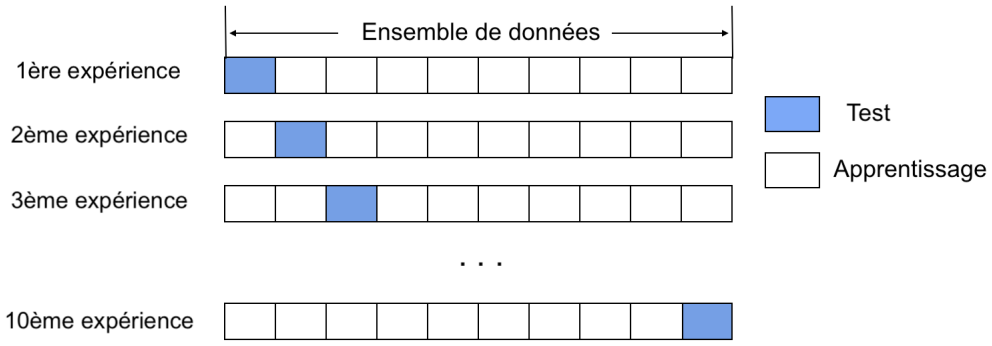

FIGURE 5.1 – Étapes d'une validation croisée à 10 plis

Nous appliquons donc la validation croisée sur les deux algorithmes les plus performants pour éviter le « biais » dû à un tirage au sort malencontreux qui peut-être fausserait l'évaluation. Les résultats sont présentés dans le tableau ci-dessous :

	Précision		Rappel		F1-Mesure	
	FR	EN	FR	EN	FR	EN
SVC	95%	94,89%	94%	95,11%	94%	94,78%
SGD	**95%**	**95,11%**	**94%**	**95,33%**	**94%**	**95%**

TABLE 5.7 – Résultats de la validation croisée sur SVC et SGD

Les résultats de la validation croisée montrent que les deux systèmes avec des algorithmes différents sont tous généralisés. Mais nous avons gardé SGD comme algorithme dans notre système au vu de la complexité en temps. Les résultats pour chaque secteur sont présentés dans les figures suivantes :

	precision	recall	f1-score	support
BANKING	0.93	0.97	0.95	539
INSURANCE	0.98	0.95	0.96	340
BUILDING CIVIL ENGINEERING	0.97	0.84	0.90	116
REAL ESTATE	0.92	0.96	0.94	107
ENERGY	0.96	0.98	0.97	402
ENVIRONMENT-WASTE MANAGEMENT	0.96	0.89	0.93	57
FOOD INDUSTRY	0.97	0.99	0.98	396
RETAILING-LUXURY-FASHION AND TEXTILE	0.96	0.97	0.96	415
AEROSPACE-AERONAUTICS-AIRLINE INDUSTRY	0.98	0.98	0.98	209
AUTOMOTIVE	0.97	0.96	0.97	176
CAPITAL GOOD	0.90	0.79	0.84	47
CHEMISTRY-PHARMACY	0.96	0.95	0.96	175
METALWORKING INDUSTRY-STEEL INDUSTRY	0.92	0.83	0.88	42
ROAD AND RAIL AND GOODS TRANSPORT	0.96	0.94	0.95	87
SHIPBUILDING AND SEA TRANSPORT	0.92	1.00	0.96	34
WOOD PAPER-PACKAGING-FURNITURE	0.91	0.88	0.89	104
DIGITAL	1.00	0.91	0.95	23
IT	0.96	1.00	0.98	23
ADVERTISING	0.95	0.86	0.90	70
MEDIA	0.94	0.99	0.96	270
LEISURE ACTIVITIES	0.98	0.97	0.97	144
SERVICES	1.00	0.85	0.92	73
avg / total	0.96	0.96	0.95	3849

FIGURE 5.2 – Résultat par secteur du SGD pour les documents français

	precision	recall	f1-score	support
BANKING	0.94	0.97	0.95	345
INSURANCE	0.89	0.79	0.84	39
BUILDING CIVIL ENGINEERING	1.00	0.57	0.73	14
REAL ESTATE	0.86	0.91	0.89	35
ENERGY	0.96	0.97	0.97	250
ENVIRONMENT-WASTE MANAGEMENT	0.85	0.92	0.88	12
FOOD INDUSTRY	0.97	0.92	0.94	61
RETAILING-LUXURY-FASHION AND TEXTILE	0.96	0.96	0.96	57
AEROSPACE-AERONAUTICS-AIRLINE INDUSTRY	1.00	0.99	0.99	86
AUTOMOTIVE	0.99	0.99	0.99	79
CAPITAL GOOD	0.75	0.60	0.67	10
CHEMISTRY-PHARMACY	0.96	0.99	0.97	108
METALWORKING INDUSTRY-STEEL INDUSTRY	0.91	1.00	0.95	30
ROAD AND RAIL AND GOODS TRANSPORT	0.83	0.71	0.77	7
SHIPBUILDING AND SEA TRANSPORT	0.92	1.00	0.96	11
WOOD PAPER-PACKAGING-FURNITURE	1.00	0.69	0.82	13
DIGITAL	1.00	0.92	0.96	13
IT	0.95	1.00	0.98	21
ADVERTISING	1.00	0.91	0.95	11
MEDIA	0.95	0.96	0.95	55
LEISURE ACTIVITIES	0.88	0.85	0.86	26
SERVICES	1.00	0.43	0.60	7
avg / total	0.95	0.95	0.95	1290

FIGURE 5.3 – Résultat par secteur du SGD pour les documents anglais

5.3.4 Résultats du nouveau système après l'ajout du corpus d'apprentissage

Nous avons ajouté de nouveaux documents pour chaque secteur et les avons pondérés dans la phase d'apprentissage pour que le système soit plus adapté aux documents actuels. Les résultats sont toujours stables et aussi performants :

	precision	recall	f1-score	support
BANKING	0.93	0.96	0.94	610
INSURANCE	0.99	0.96	0.97	389
BUILDING CIVIL ENGINEERING	0.95	0.92	0.94	151
REAL ESTATE	0.92	0.96	0.94	151
ENERGY	0.96	0.98	0.97	428
ENVIRONMENT-WASTE MANAGEMENT	0.95	0.94	0.94	78
FOOD INDUSTRY	0.97	0.98	0.97	395
RETAILING-LUXURY-FASHION AND TEXTILE	0.94	0.96	0.95	439
AEROSPACE-AERONAUTICS-AIRLINE INDUSTRY	0.97	0.99	0.98	235
AUTOMOTIVE	0.97	0.97	0.97	234
CAPITAL GOOD	0.95	0.71	0.81	49
CHEMISTRY-PHARMACY	0.95	0.96	0.95	223
METALWORKING INDUSTRY-STEEL INDUSTRY	1.00	0.95	0.98	64
ROAD AND RAIL AND GOODS TRANSPORT	0.98	0.95	0.96	134
SHIPBUILDING AND SEA TRANSPORT	0.99	1.00	0.99	72
WOOD PAPER-PACKAGING-FURNITURE	0.94	0.92	0.93	142
DIGITAL	1.00	0.97	0.98	59
IT	0.97	0.98	0.97	57
ADVERTISING	0.99	0.94	0.97	105
MEDIA	0.98	0.98	0.98	324
LEISURE ACTIVITIES	0.96	0.97	0.96	184
SERVICES	0.99	0.83	0.90	94
avg / total	0.96	0.96	0.96	4617

FIGURE 5.4 – Résultats français par secteur après l'ajout du corpus d'apprentissage

	precision	recall	f1-score	support
BANKING	0.95	0.97	0.96	361
INSURANCE	0.97	0.93	0.95	72
BUILDING CIVIL ENGINEERING	0.93	0.83	0.88	52
REAL ESTATE	0.88	0.92	0.90	63
ENERGY	0.93	0.98	0.96	276
ENVIRONMENT-WASTE MANAGEMENT	0.89	0.94	0.91	50
FOOD INDUSTRY	0.98	0.95	0.96	93
RETAILING-LUXURY-FASHION AND TEXTILE	0.98	0.95	0.96	111
AEROSPACE-AERONAUTICS-AIRLINE INDUSTRY	0.99	0.97	0.98	119
AUTOMOTIVE	0.99	0.99	0.99	130
CAPITAL GOOD	1.00	0.83	0.91	12
CHEMISTRY-PHARMACY	0.95	0.98	0.97	143
METALWORKING INDUSTRY-STEEL INDUSTRY	0.94	0.89	0.92	57
ROAD AND RAIL AND GOODS TRANSPORT	0.97	0.97	0.97	59
SHIPBUILDING AND SEA TRANSPORT	1.00	1.00	1.00	49
WOOD PAPER-PACKAGING-FURNITURE	1.00	0.92	0.96	37
DIGITAL	0.98	0.98	0.98	57
IT	1.00	1.00	1.00	62
ADVERTISING	0.98	0.98	0.98	54
MEDIA	0.96	0.98	0.97	113
LEISURE ACTIVITIES	0.98	0.91	0.94	66
SERVICES	1.00	0.83	0.91	12
avg / total	0.96	0.96	0.96	2048

FIGURE 5.5 – Résultats anglais par secteur après l'ajout du corpus d'apprentissage

5.4 Évaluations sur les nouveaux documents

Pour évaluer la généralité et l'adaptabilité de notre système, nous avons pris comme nouveau corpus de test la presse du vendredi 19 octobre 2018 qui contient la presse quotidienne et hebdomadaire. Il y a au total 459 articles dont 336 en français et 123 en anglais.

	precision	recall	f1-score	support
BANKING	0.88	0.71	0.78	69
INSURANCE	0.78	0.74	0.76	19
BUILDING CIVIL ENGINEERING	1.00	0.50	0.67	2
REAL ESTATE	0.60	0.38	0.46	8
ENERGY	0.80	0.89	0.84	9
ENVIRONMENT-WASTE MANAGEMENT	1.00	0.71	0.83	7
FOOD INDUSTRY	0.00	0.00	0.00	4
RETAILING-LUXURY-FASHION AND TEXTILE	0.89	0.85	0.87	20
AEROSPACE-AERONAUTICS-AIRLINE INDUSTRY	0.86	1.00	0.92	6
AUTOMOTIVE	1.00	0.50	0.67	6
CAPITAL GOOD	0.00	0.00	0.00	0
CHEMISTRY-PHARMACY	1.00	0.67	0.80	6
ROAD AND RAIL AND GOODS TRANSPORT	0.50	1.00	0.67	1
SHIPBUILDING AND SEA TRANSPORT	1.00	0.33	0.50	3
DIGITAL	0.62	0.58	0.60	31
IT	0.67	0.89	0.76	9
ADVERTISING	1.00	0.83	0.91	6
MEDIA	1.00	0.90	0.95	10
LEISURE ACTIVITIES	0.00	0.00	0.00	0
SERVICES	0.50	0.50	0.50	12
CLASSE VIDE	0.69	0.86	0.77	108
avg / total	0.76	0.75	0.74	336

FIGURE 5.6 – Évaluation sur le français avec le système existant

	precision	recall	f1-score	support
BANKING	0.97	0.93	0.95	69
INSURANCE	0.93	0.74	0.82	19
BUILDING CIVIL ENGINEERING	1.00	0.50	0.67	2
REAL ESTATE	0.80	1.00	0.89	8
ENERGY	1.00	0.44	0.62	9
ENVIRONMENT-WASTE MANAGEMENT	0.80	0.57	0.67	7
FOOD INDUSTRY	0.75	0.75	0.75	4
RETAILING-LUXURY-FASHION AND TEXTILE	0.87	0.65	0.74	20
AEROSPACE-AERONAUTICS-AIRLINE INDUSTRY	1.00	0.67	0.80	6
AUTOMOTIVE	1.00	0.83	0.91	6
CHEMISTRY-PHARMACY	0.71	0.83	0.77	6
ROAD AND RAIL AND GOODS TRANSPORT	0.50	1.00	0.67	1
SHIPBUILDING AND SEA TRANSPORT	0.25	0.33	0.29	3
WOOD PAPER-PACKAGING-FURNITURE	0.00	0.00	0.00	0
DIGITAL	0.81	0.68	0.74	31
IT	1.00	0.67	0.80	9
ADVERTISING	0.75	1.00	0.86	6
MEDIA	0.86	0.60	0.71	10
SERVICES	0.92	1.00	0.96	12
CLASSE VIDE	0.69	0.84	0.76	108
avg / total	0.83	0.80	0.80	336

FIGURE 5.7 – Évaluation sur le français avec le nouveau système

	precision	recall	f1-score	support
BANKING	0.64	0.50	0.56	14
INSURANCE	1.00	1.00	1.00	1
ENERGY	0.86	1.00	0.92	6
ENVIRONMENT-WASTE MANAGEMENT	0.00	0.00	0.00	1
FOOD INDUSTRY	1.00	0.50	0.67	4
RETAILING-LUXURY-FASHION AND TEXTILE	1.00	0.75	0.86	4
AEROSPACE-AERONAUTICS-AIRLINE INDUSTRY	0.75	1.00	0.86	3
AUTOMOTIVE	1.00	0.25	0.40	4
CHEMISTRY-PHARMACY	1.00	1.00	1.00	2
SHIPBUILDING AND SEA TRANSPORT	1.00	1.00	1.00	1
WOOD PAPER-PACKAGING-FURNITURE	0.00	0.00	0.00	1
DIGITAL	0.00	0.00	0.00	2
IT	1.00	0.50	0.67	4
ADVERTISING	1.00	1.00	1.00	1
MEDIA	0.50	1.00	0.67	1
LEISURE ACTIVITIES	1.00	0.50	0.67	2
CLASSE VIDE	0.80	0.93	0.86	72
avg / total	0.79	0.80	0.77	123

FIGURE 5.8 – Évaluation sur l'anglais avec le système existant

	precision	recall	f1-score	support
BANKING	0.87	0.93	0.90	14
INSURANCE	1.00	1.00	1.00	1
ENERGY	1.00	0.83	0.91	6
ENVIRONMENT-WASTE MANAGEMENT	0.00	0.00	0.00	1
FOOD INDUSTRY	1.00	1.00	1.00	4
RETAILING-LUXURY-FASHION AND TEXTILE	1.00	0.75	0.86	4
AEROSPACE-AERONAUTICS-AIRLINE INDUSTRY	1.00	0.67	0.80	3
AUTOMOTIVE	1.00	1.00	1.00	4
CHEMISTRY-PHARMACY	1.00	1.00	1.00	2
SHIPBUILDING AND SEA TRANSPORT	1.00	1.00	1.00	1
WOOD PAPER-PACKAGING-FURNITURE	1.00	1.00	1.00	1
DIGITAL	1.00	1.00	1.00	2
IT	1.00	0.75	0.86	4
ADVERTISING	1.00	1.00	1.00	1
MEDIA	1.00	1.00	1.00	1
LEISURE ACTIVITIES	1.00	1.00	1.00	2
CLASSE VIDE	0.92	0.97	0.95	72
avg / total	0.93	0.93	0.93	123

FIGURE 5.9 – Évaluation sur l'anglais avec le nouveau système

5.4.1 Matrice de confusion

Les résultats présentés ci-dessus indiquent la précision, le rappel et la F1-Mesure pour chaque classe ainsi que la moyenne pour l'ensemble des classes. Pour mesurer l'écart entre le résultat du système et le résultat attendu (la référence) et les analyser, nous faisons appel à un outil clé appelé « matrice de confusion », qui comptabilise, pour chaque classe, tous les documents bien ou mal classés. Dans une matrice de confusion, chaque colonne représente le nombre d'occurrences d'une classe estimée, tandis que chaque ligne représente le nombre d'occurrences d'une classe réelle (ou de référence). L'intérêt principal de la matrice de confusion est qu'elle montre rapidement si un système de classification parvient à classifier correctement.

Les matrices de confusion pour le nouveau système pour le français et l'anglais sont présentées ci-dessous :

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	60	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	6
1	1	14	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	3
2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
3	1	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	1	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
5	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	4
6	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	1
7	0	0	0	0	0	0	1	13	0	0	0	0	0	0	0	0	1	0	0	5
8	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	4
9	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	1
10	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	1
11	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	2
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	2	0	0	0	0	0	0	0	0	0	0	0	0	0	18	0	0	0	0	11
15	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	4	0	0	0	3
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0
17	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	6	0	3
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0
19	14	0	2	2	0	1	0	0	0	0	2	0	3	2	3	0	0	1	1	77

	index
0	BANKING
1	INSURANCE
2	BUILDING CIVIL ENGINEERING
3	REAL ESTATE
4	ENERGY
5	ENVIRONMENT-WASTE MANAGEMENT
6	FOOD INDUSTRY
7	RETAILING-LUXURY-FASHION AND TEXTILE
8	AEROSPACE-AERONAUTICS-AIRLINE INDUSTRY
9	AUTOMOTIVE
10	CHEMISTRY-PHARMACY
11	ROAD AND RAIL AND GOODS TRANSPORT
12	SHIPBUILDING AND SEA TRANSPORT
13	WOOD PAPER-PACKAGING-FURNITURE
14	DIGITAL
15	IT
16	ADVERTISING
17	MEDIA
18	SERVICES
19	CLASSE VIDE

FIGURE 5.10 – Matrice de confusion du nouveau système pour le français

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	1
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
4	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	1
6	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	1
7	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1
13	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0
16	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	70

	index
0	BANKING
1	INSURANCE
2	ENERGY
3	ENVIRONMENT-WASTE MANAGEMENT
4	FOOD INDUSTRY
5	RETAILING-LUXURY-FASHION AND TEXTILE
6	AEROSPACE-AERONAUTICS-AIRLINE INDUSTRY
7	AUTOMOTIVE
8	CHEMISTRY-PHARMACY
9	SHIPBUILDING AND SEA TRANSPORT
10	WOOD PAPER-PACKAGING-FURNITURE
11	DIGITAL
12	IT
13	ADVERTISING
14	MEDIA
15	LEISURE ACTIVITIES
16	CLASSE VIDE

FIGURE 5.11 – Matrice de confusion du nouveau système pour l'anglais

5.5 Insuffisances des mesures classiques

Nous avons présenté dans les sections précédentes des mesures d'évaluations classiques : la précision, le rappel et la F1-mesure qui sont utilisés largement dans la recherche d'information et la classification. Cependant, ces mesures présentent quelques défauts : elles répondent à une logique binaire et ne permettent pas d'intégrer certains niveaux d'exigence en matière de résultats. Par exemple, supposons un résultat de classification ci-dessous :

	Précision	Rappel	F1-Mesure
Classe A	50%	100%	67%
Classe B	100%	50%	67%
Moy/Total	75%	75%	75%

TABLE 5.8 – Exemple du résultat de classification

Pour les classes A et B, la F1-Mesure sont tous 67% tandis que leur précision et rappel sont très différents. Les moyennes de précision et rappel, elles, se calculent à partir des précisions et rappels de chaque classe sans pondération particulière ou en pondérant le nombre de références pour chaque classe. Mais l'importance de chaque classe ne correspond pas forcément au nombre de documents dans cette classe. En revanche, dans la situation réelle, il existe souvent des relations entre les classes. Certaines classes sont plus proches entre elles. On acceptera davantage qu'un document « energy » soit classé dans « environment » que dans « insurance ». La précision et le rappel ne tiennent pas compte de cela et appliquent la même pénalité (0) pour chaque document mal classé.

5.6 Mesures d'évaluation complémentaires

5.6.1 Les demandes de LEOnard

Étant le navigateur assistant de recherche documentaire, LEOnard a pour objectif d'offrir aux utilisateurs le meilleur de l'actualité. Ce projet met l'accent sur deux points :

— Les contenus : Comme il s'agit d'un projet de la banque BNP Paribas, les secteurs liés à la banque et aux activités financières comme « Banking » et « Insurance » intéressent davantage les collaborateurs que les autres secteurs ;
— La fiabilité des résultats (choix de privilégier le silence) : les bruits (Faux Négatifs) doivent être plus pénalisé que les silences (Faux Positifs) pendant l'évaluation.

Pour évaluer la qualité du système, les erreurs ne sont pas au même niveau. Certaines erreurs sont moins graves que d'autres, notamment quand leurs classes sont sémantiquement proche, par exemple « Advertising » et « Media ».

5.6.2 Une nouvelle mesure adaptée : Leo-Score

Dans cette section, nous allons définir une nouvelle mesure à partir de la précision, du rappel et de la F1-Mesure, qui est mieux adaptée à la tâche réelle pour évaluer les résultats générés. Nous avons implémenté cette nouvelle mesure à partir de la matrice de confusion en répondant aux besoins suivants :

— **Les relations entre les classes.** L'idée est de bien distinguer les différents types d'erreurs et de leur associer des pénalités différentes. Nous considérons que la précision et le rappel sont insuffisants puisqu'ils ne hiérarchisent pas les erreurs. Pour implémenter cette idée dans l'évaluation, nous avons d'abord créé une nouvelle matrice avec la même dimension que celle de la matrice de confusion généré par le système. Nous avons ensuite complété la matrice avec des chiffres entre 0 et 1 selon la relation avec le secteur donné. Par exemple, pour le secteur « banking », les documents classés en « banking » ont un poids « 1 » et les documents classés en « Insurance » qui est sémantiquement proche du secteur « banking » ont un poids « 0.5 » Et pour les secteurs éloignés comme « Food industry », ils ont un poids très petit ou « 0 ». Nous avons appliqué cette matrice de poids à la matrice de confusion générée et obtenu une nouvelle matrice sur laquelle nous effectuons des calculs par la suite.

— **L'hétérogénité de classes.** Il s'agit d'une contrainte utilisée souvent dans l'évaluation du clustering [Amigó et al., 2009] [Moreno and Dias, 2015]. L'idée principale est que le système qui permet de classer dans un secteur donné des documents appartenant au moins de classes possibles est le plus performant. Idéalement, il y a une seul classe dans un secteur, soit le secteur de référence. Nous avons pris la matrice de confusion générée dans l'étape précédente et avons compté le nombre de classes de prédiction. Si le nombre de classes de prédiction d'une classe de référence est inférieur à 4, nous avons considéré que les résultats de cette classe respectent cette contrainte et leur avons associé un poids supérieur à 1.

— **Les pondérations de classes.** L'idée est que certaines classes sont plus importantes que d'autres dans LEOnard. Nous avons choisi avec l'équipe LEOnard 10 secteurs qui sont plus importants parmi tous les secteurs, à savoir, « banking », « insurance », « building civil engineering », « real estate », « energy », « environment-waste management », « aerospace-aeronautics-airline industry », « automotive », « digital » et « IT ». Nous avons attribué aux résultats de ces secteurs plus d'importance lors du calcul de la moyenne.

— **La pondération de précision.** Dans le domaine bancaire, la pertinence des informations est primordiale. La précision est donc privilégiée pour évaluer la qualité du système. Dans nos expériences précédentes, nous avons utilisé F-Mesure avec β = 1 qui pondère précision et rappel de manière égale. D'après

la formule générale de F-Mesure, le paramètre β est un coefficient qui permet de pondérer la précision ou le rappel. Dans notre cas, nous souhaitons obtenir un système de classification ayant une bonne précision, soit peu bruité, même si les résultats fiables sont partiels. Nous optons ainsi pour β=0.5 afin d'être plus exigeant sur la précision.

Après avoir appliqué toutes ces contraintes sur notre résultat, nous avons obtenu un score nommé Leo-Score pour évaluer la qualité du système. Cet indicateur unique est capable de mesurer la qualité des résultats donnés par un système de classification dans la plate-forme LEOnard. Voici les Leo-Score pour le système existant et notre système :

	Système Existant	Nouveau Système
Français	60,78%	71,14%
Anglais	48,94%	89,37%
Moy/Total	64,88%	81,76%

TABLE 5.9 – Leo-Score des deux systèmes

Selon les Leo-Scores obtenus par les deux systèmes, le nouveau système est plus performant que le système existant en tenant compte des critères spécifiques au niveau des résultats de classification.

5.6.3 Les critères complémentaires

Pour le système de classification implémenté dans la plate-forme LEOnard, la qualification ne se limite pas aux résultats obtenus. D'autres critères sont aussi importants tels que la reproductibilité, etc. Nous avons également évalué notre système à l'une de ces critères.

— **La reproductibilité.** Elle s'appuie sur le fait que les mêmes résultats doivent être générés si les même expériences se sont effectuées dans les mêmes conditions [Cohen et al., 2016]. C'est un enjeu de grande importance car notre système réalisé sur un l'ordinateur local sera implémenté dans la plate-forme LEOnard. Pour tester la reproductibilité de notre système, nous avons exécuté notre script plusieurs fois sous l'environnement Linux et Windows. Tous les tests montrent que notre système est reproductible.

— **La complexité.** La complexité d'un système consiste en la quantité de ressources (par exemple de temps ou d'espace) nécessaires à l'exécution de ce système. Notre système est composé de deux phases : la phase d'apprentissage et la phase de prédiction. La première consiste à entraîner le modèle à partir du corpus d'apprentissage et à le sauvegarder et la seconde consiste à prédire les classes pour de nouvelles données. C'est sur la phase de prédiction que nous avons donc calculé la complexité. Comme l'algorithme est le cœur de

la phase de prédiction, les calculs de complexité peuvent se simplifier en complexité de l'algorithme utilisé - le classifieur SGD. D'après la documentation de *Scikit-Learn*, le classifieur SGD est fondamentalement linéaire dans le nombre d'exemples. Si X est une matrice de taille (n, p), la complexité est $O(n\bar{p})$, où $\bar{p}$ est le nombre moyen d'attributs non nuls par échantillon. Sur l'échelle des mesures de complexité, il correspond au cas favorable. Plus concrètement, nous avons calculé le temps d'exécution du processus complet (y compris le nettoyage, les pré-traitements, la détection de classe vide, etc) pour tous les articles (459) du vendredi 19 octobre 2018 : nous obtenons 793 s (environ 13 mins).

— **La stabilité et la robustesse.** Une autre contrainte pour évaluer un bon système : le modèle doit dépendre aussi peu que possible de l'échantillon d'apprentissage et se généraliser à d'autres échantillons. Pour éviter l'*overfitting* (sur-apprentissage) et tester la robustesse de notre système, nous avons appliqué la validation croisée dans nos expériences d'évaluation. Les résultats ont montré que notre système est satisfaisant au niveau de la stabilité et de la robustesse.

5.7 Conclusion

Dans ce chapitre, nous avons évalué les deux systèmes avec les mesures d'évaluation classiques et une nouvelle mesure : le « Leo-Score ». Ce score a été construit pour améliorer et simplifier les évaluations sur plusieurs dimensions tout en répondant aux besoins spécifiques de notre tâche. Les résultats montrent que notre système est plus performant et adapté à LEOnard que le système existant du point de vue de la classification. De plus, nous avons évalué notre système avec des critères supplémentaires portant sur la qualification globale du nouveau système. Il reste toujours meilleur en stabilité, robustesse, complexité et reproductibilité. Nous avons donc réussi dans ce travail à construire un système de classification comparable au système existant, voire plus performant encore.

Conclusion

Au cours de cette étude nous avons répondu au besoin suivant : effectuer la refonte d'un système de classification automatique de textes existant en open source. Cette demande repose sur le fait que le système de classification en vigueur dans la plate-forme LEOnard ne peut plus être actualisé à cause de codes sources inaccessibles. Notre objectif est de fournir un système en open source qui est comparable avec celui-ci.

Après différentes expériences, nous avons retenu comme paramètres optimaux : stems des mots comme *features* et leurs poids TF-IDF dans un modèle vectoriel. Au niveau de l'algorithme, nous avons choisi SGD (*Stochastic Gradient Descent*) avec la validation croisée pour entraîner le modèle. Notre système a obtenu une F1-mesure de 96% en moyenne alors que le système existant a eu 79,17% sur le corpus de test qui provient de la même source du corpus d'apprentissage.

Seulement, Les mesures d'évaluation classiques comme la précision, le rappel et la F1-Mesure ne sont pas suffisantes pour évaluer les résultats comme il existe des contraintes spécifiques dans LEOnard. Nous avons donc défini une nouvelle mesure Leo-Score en intégrant ces contraintes pour évaluer les résultats. Le Leo-Score obtenu pour notre système est de 81,76% par rapport au 64,88% pour le système existant, ce qui confirme la performance du nouveau système.

En conclusion, nous avons développé un système de classification multi-classe pour des articles de presse du domaine bancaire et défini une nouvelle mesure d'évaluation Leo-Score pour la plate-forme. D'après les évaluations, notre système est plus performant que le système existant.

Perspectives

Les futurs travaux prévoient d'exploiter de nouvelles pistes pour le développement et l'évaluation de la classification. Un système de classification multi-labels peut être aussi très intéressant lorsqu'il s'agit de la classification de la presse. Au niveau de la transformation vectorielle des features, il serait intéressant de faire des expériences sur la représentation « Word Embedding » comme *Word2vec* et *Glove* qui obtiennent de bonnes performances dans la littérature. De plus, d'autres algorithmes de classification restent à être testés, par exemple, les réseaux de neurones

[Lai et al., 2015] [Zhang et al., 2015] et d'autres algorithmes de boosting comme *Adaboost*, etc. Au niveau de l'évaluation, nous envisageons d'exploiter davantage le LeoScore et d'essayer de le généraliser afin qu'il puisse s'adapter à davantage de tâches.

BIBLIOGRAPHIE

[Aas and Eikvil, 1999] Aas, K. and Eikvil, L. (1999). Text categorisation: A survey. technical report, norwegian computing center. *Available online: citeseer. ist. psu. edu/aas99text. html*. – Cité page 16.

[Adeva et al., 2014] Adeva, J. G., Atxa, J. P., Carrillo, M. U., and Zengotitabengoa, E. A. (2014). Automatic text classification to support systematic reviews in medicine. *Expert Systems with Applications*, 41(4):1498–1508. – Cité page 16.

[Amigó et al., 2009] Amigó, E., Gonzalo, J., Artiles, J., and Verdejo, F. (2009). A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, 12(4):461–486. – Cité page 59.

[Cohen et al., 2016] Cohen, K. B., Xia, J., Roeder, C., and Hunter, L. E. (2016). Reproducibility in natural language processing: a case study of two r libraries for mining pubmed/medline. In *LREC... International Conference on Language Resources & Evaluation:[proceedings]. International Conference on Language Resources and Evaluation*, volume 2016, page 6. NIH Public Access. – Cité page 60.

[Guzella and Caminhas, 2009] Guzella, T. S. and Caminhas, W. M. (2009). A review of machine learning approaches to spam filtering. *Expert Systems with Applications*, 36(7):10206–10222. – Cité page 16.

[Hand et al., 2001] Hand, D. J., Mannila, H., and Smyth, P. (2001). *Principles of data mining (adaptive computation and machine learning)*. MIT press Cambridge, MA. – Cité page 17.

[Jouannet, 2015] Jouannet, A. (2015). Apports de la catégorisation automatique à la veille collaborative. – Cité page 31.

[Lai et al., 2015] Lai, S., Xu, L., Liu, K., and Zhao, J. (2015). Recurrent convolutional neural networks for text classification. In *AAAI*, volume 333, pages 2267–2273. – Cité page 64.

[Liu, 2007] Liu, B. (2007). *Web data mining: exploring hyperlinks, contents, and usage data*. Springer Science & Business Media. – Cité page 15.

[Ma, 2014] Ma, Y. (2014). L'intégration du thésaurus dans le traitement de la catégorisation automatique. – Cité page 31.

[Manning et al., 2008] Manning, C. D., Raghavan, P., and Schütze, H. (2008). Text classification and naive bayes. *Introduction to information retrieval*, 1(6). – Cité page 17.

[Mihalcea and Radev, 2011] Mihalcea, R. and Radev, D. (2011). *Graph-based natural language processing and information retrieval*. Cambridge university press. – Cité page 17.

[Mitchell et al., 1997] Mitchell, T. M. et al. (1997). Machine learning. 1997. *Burr Ridge, IL: McGraw Hill*, 45(37):870–877. – Cité page 15.

[Moreno and Dias, 2015] Moreno, J. G. and Dias, G. (2015). Adapted b-cubed metrics to unbalanced datasets. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 911–914. ACM. – Cité page 59.

[Nakache and Métais, 2005] Nakache, D. and Métais, E. (2005). Evaluation: nouvelle approche avec juges. In *INFORSID*, volume 5, pages 555–570. – Cité page 49.

[Schütze et al., 2008] Schütze, H., Manning, C. D., and Raghavan, P. (2008). *Introduction to information retrieval*, volume 39. Cambridge University Press. – Cité page 15.

[Sebastiani, 2002] Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47. – Cité page 15.

[Zhang et al., 2015] Zhang, X., Zhao, J., and LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657. – Cité page 64.

ANNEXE

A.1 Plan de classement sectoriel complet

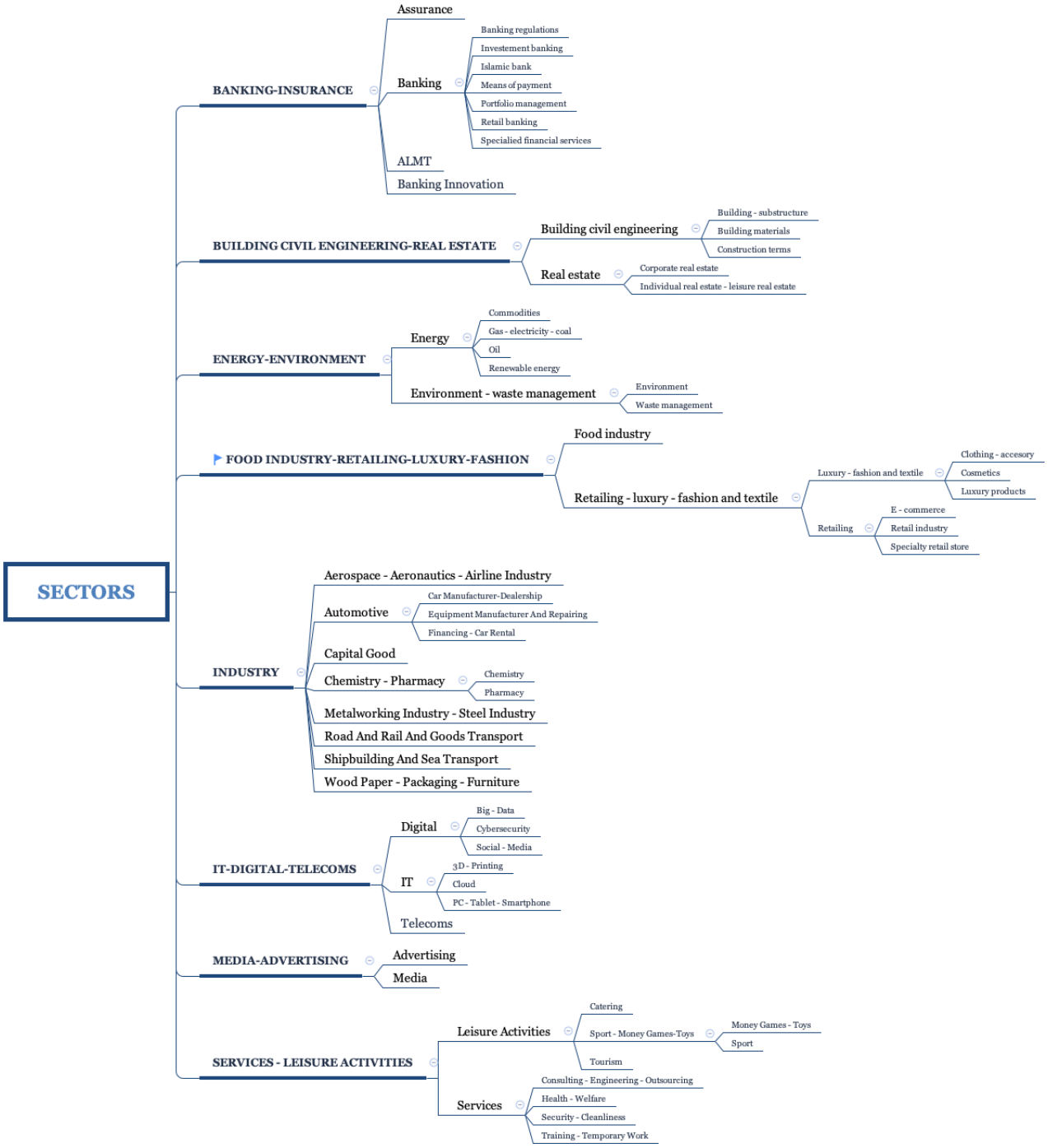

FIGURE A.1 – Plan de classement sectoriel complet