Classification automatique de comptes rendus anatomo-pathologiques en texte libre : application au sein d'un registre de cancers

Benjamin Naffrechoux

Master Sciences, Technologies, Santé
Mention Santé Publique

Parcours
Systèmes d'Information et Technologies Informatiques pour la Santé
Promotion 2018-2019

Classification automatique de comptes rendus anatomo-pathologiques en texte libre : application au sein d'un registre de cancers

Mémoire réalisé dans le cadre d'une mission effectuée
du 01/03/2019 au 29/08/2019
Structure d'accueil : Registre général des cancers de la Gironde
Maître de stage: Mr. Brice AMADEO (MCU)
Encadrants : Georgeta Bordea, Sylvain Maurisset

Soutenu publiquement le 10/09/2019
Par Benjamin NAFFRECHOUX

Jury de soutenance
Mr. Rabia AZZI, tuteur universitaire
Mr. Gayo DIALLO, rapporteur

Remerciements

Je tiens à remercier l'ensemble de l'équipe du registre général des cancers de la Gironde ainsi que l'équipe ERIAS pour leur accueil bienveillant.

Un grand merci à Brice, pour sa patience, ses conseils avisés ainsi que pour sa relecture constante au fil de ce stage et des nombreuses versions (plus ou moins abouties) de ce présent rapport.

Georgeta, Sylvain, merci pour votre expertise dans le domaine de l'informatique et de l'intelligence artificielle : sans vous, je crois que j'y serai encore… ☺

Enfin, merci à mes proches d'avoir supporté mon anxiété durant ces six mois (qui s'est parfois transformée en procrastination, il faut bien le dire). Et tout particulièrement à Fanny : merci d'être là, tout simplement (et aussi merci pour la traduction du résumé, mais ça, ça reste entre nous…).

Sommaire

Table des figures, tableaux et annexes

Abréviations

CRAP : compte rendu anatomo-cyto-pathologique
IA : intelligence artificielle
TAL : traitement automatique de la langue
ADICAP : Association pour le Développement de l'Informatique en Cytologie et Anatomo-Pathologie
CIMO3 : classification internationale des maladies pour l'oncologie

Introduction

Le cancer constitue la deuxième cause de mortalité dans le monde, soit un décès sur six En France, cette pathologie est même la première cause de décès chez les hommes et la première cause de décès prématurés (avant 65 ans) tout sexe confondu (sources : OMS, DREES). Les registres de cancers, en recueillant des données épidémiologiques, s'inscrivent ainsi dans une démarche d'observation et d'évaluation des politiques de lutte contre cette pathologie.

Avec le développement de l'informatique au sein des systèmes d'information en santé, l'accès à ces données ainsi que leur traitement ont été facilités. De plus, les progrès réalisés dans ce domaine, notamment dans le domaine de l'intelligence artificielle (IA), ouvrent la voie à de nouvelles perspectives : ainsi, il est désormais envisageable d'automatiser certaines tâches autrefois réservées à un travail exclusivement manuel.

C'est dans ce cadre que s'inscrit notre travail : après avoir décrit le contexte scientifique, nous détaillerons la méthodologie envisagée pour automatiser le traitement de comptes rendus anatomo-cyto-pathologiques (CRAP), une tâche récurrente au sein d'un registre de cancers. Nous discuterons alors nos résultats puis suggèreront des perspectives d'amélioration pour mettre en application notre objectif.

1 Contexte

1.1 Principe de fonctionnement d'un registre de cancer

Un registre de morbidité est défini comme « un recueil continu et exhaustif de données nominatives intéressant un ou plusieurs événements de santé dans une population géographiquement définie, à des fins de recherche et de santé publique, par une équipe ayant les compétences appropriées» [1].

Dans le cas d'un registre de cancer, ce recueil concerne des données intéressant la survenue d'une pathologie cancéreuse. En France, il n'existe pas de registre national de cancer : le recueil est départemental [2]. Celui-ci concerne donc tous les nouveaux cas de cancer d'un département donné, dans le but d'assurer la mission principale de ces registres : participer à la surveillance épidémiologique des cancers [3]. Afin d'être exhaustif (et d'éviter une sous-estimation de l'incidence des cancers), ils utilisent de multiples sources d'information (figure 1).

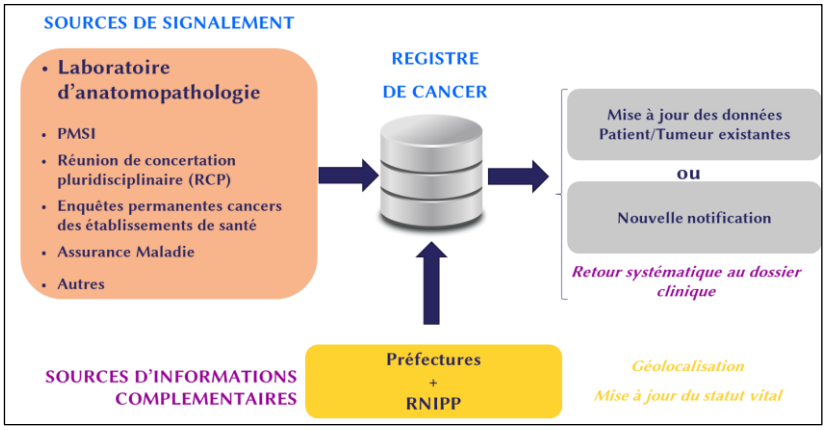

Figure 1 : Principe général et sources d'information des registres de cancer en France
RNIPP : répertoire national d'identification des personnes physiques

1.2 Le compte rendu anatomo-cyto-pathologique (CRAP) : une des principales sources de données pour un registre de cancer

Parmi les sources d'information d'un registre, les CRAP en constituent la principale pour le recueil de la typologie (topographie et morphologie) d'un cancer vérifié microscopiquement [4,5]. En effet, lorsque cela est possible, effectuer un prélèvement tumoral constitue le Gold Standard pour le diagnostic d'un cas de cancer. Le CRAP constitue alors une donnée textuelle détaillée de ce type de prélèvement.

La topographie (ou localisation anatomique) et la morphologie (ou type histologique) d'un cancer font partis des items d'informations dits « basiques » pour un registre : ce sont des items essentiels pour décrire le cas de cancer recueilli et qui doivent être codés au sein de la base du registre [5]. L'attribution d'un code à une donnée sert à résumer celle-ci par un concept (médical ou non), identifié par ce code. Cela facilite la réutilisation secondaire de celle-ci [6], c'est-à-dire l'utilisation de cette donnée dans un contexte autre que celui dans lequel elle a été produite.

Pour le codage de la typologie, la terminologie recommandée par l'IARC est la classification internationale des maladies pour l'oncologie, 3ème édition [7] (CIMO3). C'est une terminologie à deux axes (un axe topographique et un axe morphologique) réservée à la description de pathologies tumorales. La figure 2 présente la structure d'un code CIMO3, composé d'un code topographique et/ou d'un code morphologique (ces codes faisant partie de deux axes différents, ils sont indépendants). Le code topographique est composé de 4 caractères alphanumériques : les trois premiers indiquent la localisation de la tumeur et le 4ème apporte une précision par rapport à cette localisation. Le code morphologique est quant à lui composé de cinq chiffres : les quatre premiers indiquent le type histologique précis et le 5ème caractérise le comportement de la tumeur (malin, bénin,…).

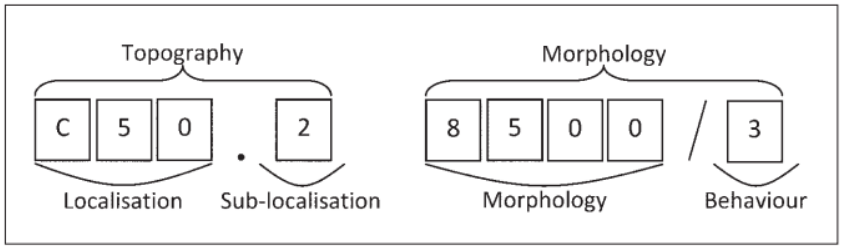

Figure 2 : Structure d'un code CIMO3, par Jouhet et al. [8]
(carcinome canalaire (8500) infiltrant (/3) du quadrant supéro-interne (.2) du sein (c50))

A partir de l'exemple du registre général des cancers de la Gironde, nous allons détailler comment sont réutilisés les CRAP pour en extraire ces deux items d'information. Dans un premier temps, il faut distinguer deux catégories de CRAP recueillis par le personnel d'un registre auprès des laboratoires d'anatomo-cyto-pathologie :

1) les CRAP codés directement par le laboratoire, contenant à la fois du texte et un code permettant d'identifier la typologie du prélèvement.
2) les CRAP non codés, où seul le compte-rendu au format texte est disponible. On parle de données textuelles non structurées, ou données « texte libre ».

1.2.1 Réutilisation des CRAP codés

En France, les laboratoires utilisent principalement la terminologie ADICAP [9] pour coder leurs CRAP. Elle a été élaborée par les pathologistes français de l'Association pour le Développement de l'Informatique en Cytologie et Anatomo-Pathologie (ADICAP). Un code ADICAP est composé de huit caractères alphanumériques obligatoires : le 1er identifie le mode de prélèvement, le 2ème le type de technique, les 3ème et 4ème au « code organe » (topographie du prélèvement), et les 5ème à 8ème au « code lésion » (morphologie du prélèvement). Grâce à ce code annoté au compte-rendu, les registres intègrent dans leur base uniquement les CRAP relatifs à une pathologie tumorale maligne. Par exemple, les CRAP ayant pour 2ème caractère du code lésion les chiffres 0, 1, 2 ou 3 (chiffres associés à des tumeurs bénignes) sont ignorés.

Puis, au sein du registre, les codes ADICAP sont « traduits » en codes CIM-O3 grâce à une table de transcodage ADICAP-CIMO3, pour respecter les recommandations de standardisation de l'IARC.

1.2.2 Réutilisation des CRAP en texte libre

Pour les CRAP non codés, une première sélection est faite sur les patients résidents en Gironde et sur les patients avec un code postal vide dont le prescripteur est Girondin. Ils sont alors extraits de la base des laboratoires sans autre sélection. Puis, un travail manuel est réalisé à partir du texte libre des CRAP, en suivant ces étapes:

1) Distinction des CRAP concluant à un cancer de ceux concluant à l'absence de cancer
2) Codage en ADICAP des CRAP relatifs à un cancer, à l'exception des CRAP en lien avec les autres registres de la région Aquitaine (ex: registre des hémopathies malignes)
3) Transcodage ADICAP-CIMO3, puis enregistrement dans la base du registre.

Ce travail manuel est couteux en termes de temps et de ressources, et sujet à un risque d'erreur. Il est pourtant indispensable pour répondre au devoir d'exhaustivité d'un registre. Or, dans un contexte où l'informatique est omniprésente au sein des systèmes d'information en santé et notamment au sein des registres de cancer (une étude conduite en 1991 a montré que l'ensemble des registres audités utilisaient en partie l'informatique dans leur procédure d'enregistrement des cas de cancer [4]), il semble nécessaire d'utiliser cette technologie pour automatiser cette tâche.

Le champ de l'informatique qui traite ce type de données « texte libre » se nomme la fouille de texte (« text mining » en anglais). Dans la suite de ce rapport, nous introduirons cette discipline et nous verrons son application au traitement des CRAP en texte libre.

1.3 Fouille de texte (« text mining »)

1.3.1 Définition

Avec le développement d'Internet, une grande quantité de textes a été rendu accessible. Ces données textuelles constituent une source d'information intéressante à exploiter dans de nombreux domaines (finance, marketing, médecine,…). Il n'est donc pas surprenant de voir se développer des technologies informatiques permettant d'optimiser le traitement de ces données. La fouille de texte est la discipline qui regroupe ces technologies, héritière directe de la fouille de données (« data mining » en anglais) née dans les années 1990.[10]

De façon similaire à la fouille de données, la fouille de texte vise à traiter de grandes quantités de données textuelles pour obtenir une information exploitable en sortie. Pour répondre de façon simple à cet objectif, et même si elle emprunte parfois les outils linguistiques des technologies de traitement automatique de la langue (TAL), la fouille de texte ne cherche pas à comprendre le sens profond d'un texte mais décompose ce traitement en tâches élémentaires. Il est alors possible de combiner ces tâches élémentaires pour réaliser une tâche plus complexe.

1.3.2 Principales tâches élémentaires

Au sens informatique, une tâche est la spécification d'un programme qui mime une compétence précise d'un être humain [10]. Schématiquement, une tâche de fouille de texte est caractérisée par des données textuelles en entrée (ayant subies ou non un prétraitement) et des données en sortie (le résultat de la tâche).

On peut décomposer la fouille de texte en 4 tâches élémentaires : la recherche d'information (RI), la classification (c'est la tâche qui se rapproche le plus de celles utilisées en fouille de données), l'annotation et l'extraction d'information (EI). Ces tâches sont en réalité très liées les unes aux autres. En effet, une simple reformulation du problème permet souvent de « remplacer » une tâche par une autre. Cela a un intérêt lorsque l'on veut comparer deux tâches entre elles lors de la résolution d'un même problème.

Par exemple, l'extraction d'information peut être vue comme une tâche d'annotation : extraire des informations d'un texte est similaire à annoter le texte en indiquant la position des unités porteuses de l'information. Une tâche d'annotation peut aussi être reformulée comme une tâche de classification ; ainsi, par extension, on peut extraire des informations d'un texte grâce à des méthodes de classification.

Pour construire le programme réalisant cette tâche et aboutir au résultat, il existe deux approches principales : (1) une approche dite « manuelle », (2) une approche par apprentissage automatique (« machine learning » en anglais).

L'approche manuelle utilise des méthodes de l'intelligence artificielle dite symbolique, basée sur des règles construites par un expert du domaine d'étude (appelées règles de décision) et codées explicitement à l'intérieur du programme. De tels programmes sont appelés "systèmes experts" et étaient les plus utilisés jusque dans les années 1980.

A l'inverse, un programme d'apprentissage automatique est capable d'apprendre seul ces règles, en s'entraînant avec des exemples de la tâche envisagée. Ces exemples permettent de construire un modèle, c'est à dire un programme paramétré ("entraîné") capable de réaliser cette tâche sur de nouvelles données. Selon les exemples fournis, on distingue deux grands types de programme de machine learning (figure 3):

1) Apprentissage supervisé, où les exemples sont des couples <entrée, résultat>, c'est à dire des données dont on sait quelle est l'information à obtenir en sortie du programme
2) Apprentissage non supervisé, où les exemples sont uniquement des données d'entrée, sans information sur le résultat à obtenir.

L'apprentissage semi-supervisé combine ces 2 types de programme.

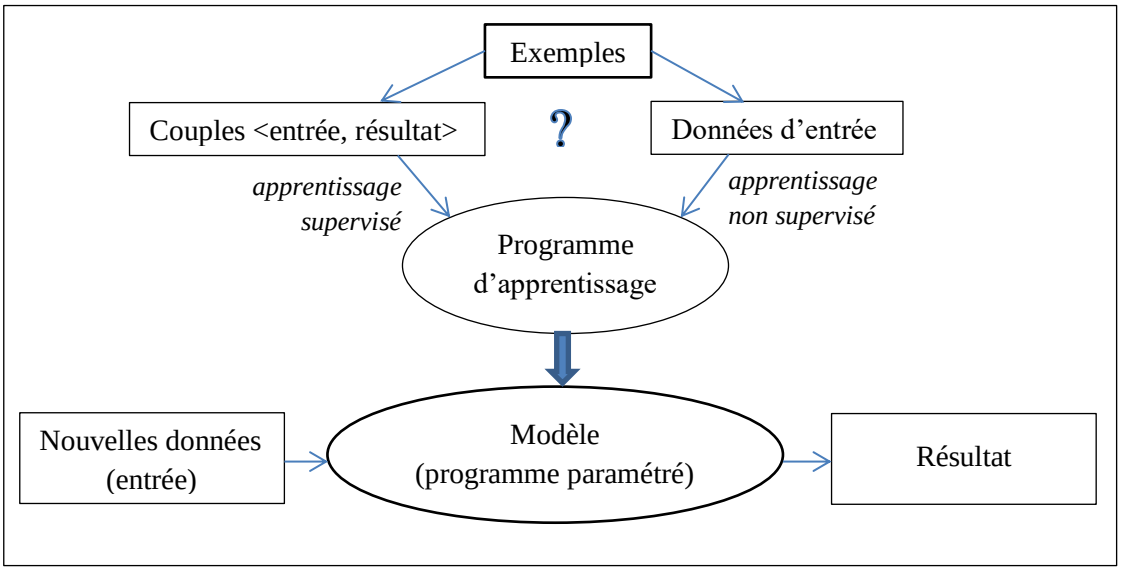

Figure 3 : Schéma général d'un programme de machine learning

Là où dans l'approche manuelle la justesse du programme dépendait de la qualité (au sens de représentativité du domaine) des règles établies par un expert, la justesse d'un programme de machine learning dépend de la qualité des exemples fournis en entrée. En effet, un tel programme aura de bons résultats seulement pour des données qui statistiquement ressemblent à celles lui ayant servi d'exemples.

1.4 Application de la fouille de texte au traitement des CRAP en texte libre

L'application de la fouille de texte au domaine biomédical est un champ de recherche en plein essor. Ainsi, plusieurs systèmes ont déjà été développés pour traiter les données textuelles relatives à un cas de cancer.

1.4.1 Systèmes existants

CaFE a été développé pour améliorer et accélérer la procédure de recueil des cas de cancer par les registres de l'université du Michigan. A l'aide d'une liste de termes (servant de « règles de décision » pour le programme) construite par des employés expérimentés, il réalise une tâche d'annotation en surlignant les phrases pertinentes (par rapport à cette liste) d'un document médical. Grâce à ces annotations, il arrive à exclure automatiquement les CRAP ne relevant pas d'un cas de cancer, avec une sensibilité de 100% et une spécificité de 85% [11].

L'outil caTIES s'intègre dans un projet d'intégration et de partage de données anatomopathologiques, initié par le SPIN (Shared Pathology Informatics Networks) aux USA. Il permet de rechercher des CRAP avec une précision moyenne de 94% [12]. Pour cela, il annote ces CRAP avec des concepts médicaux, grâce à une tâche d'extraction d'information utilisant un vocabulaire contrôlé (ex: le NCI-thesaurus). Ce sont ces annotations qui servent d'index pour la fonction de recherche.

MedTAS/P est un outil d'annotation et d'extraction d'information de CRAP. Il permet ainsi de faire correspondre les éléments annotés d'un CRAP en texte libre à des concepts d'une terminologie (ex: codes de topographie et morphologie de la CIMO3) ou à des termes d'un dictionnaire (vocabulaire contrôlé). Il a été évalué en analysant des CRAP de patients atteints de cancer du côlon, et a montré une bonne précision [13].

MEDTEX est également un système d'extraction d'information. Il se base sur la terminologie SNOMED-CT ainsi que sur un dictionnaire établi par le registre des cancers de Queensland en Australie pour extraire les concepts médicaux présents dans les CRAP en texte libre. Ces extractions correspondent à ce que les auteurs appellent "items de notification de cancer", et sont retrouvées par le système avec une justesse moyenne variant de 74 à 82% en comparaison au Gold Standard [14]. Ceux-ci permettent alors d'identifier les CRAP relatifs à un cas de cancer avec une spécificité de 96 à 100%. L'item le plus dur à extraire était le site primaire du cancer, avec une justesse de 51 à 60%. En modifiant la granularité du site primaire (Cxx plutôt que Cxx.x), la justesse du système augmente à 80% pour cet item.

Les systèmes ci-dessus ont été construits à partir d'une approche dite « manuelle ». D'autres systèmes intègrent du machine learning pour traiter ces données « texte libre », de façon exclusive ou en combinaison avec l'approche manuelle :

CRCP a été développé à l'université d'Alabama. Il utilise plusieurs tâches de fouille de texte, basées à la fois sur une approche manuelle et sur du machine learning. Une première tâche d'extraction d'information permet d'identifier les concepts médicaux relatifs à un cancer dans un CRAP, et d'annoter ces derniers. La deuxième tâche est une tâche de classification par apprentissage automatique : elle utilise entre autres les concepts extraits par la première tâche en tant qu'attributs (« features » en anglais, cf partie 4.3.2) pour l'algorithme de classification. Les auteurs montrent que la première tâche seule améliore l'efficacité du registre dans le recueil des cas de cancer [15]. Ils montrent également que l'apport du machine learning améliore la précision du recueil mais pas le rappel.

ARC est un système de recherche d'information développé par le MAVERIC. En permettant d'évaluer différents algorithmes de classification (fonctionnant avec différentes combinaisons de features), ce système a montré de bonnes performances pour classer des CRAP "relatifs à un cancer", avec par exemple un rappel de 0.97 et une précision de 0.95 pour le cancer de la prostate [16].

1.4.2 Le cas des CRAP en langue française

En France, il n'existe pas à notre connaissance de systèmes similaires, c'est-à-dire un système automatisant le processus de traitement d'un CRAP en texte libre dans son ensemble. Par exemple, Jouhet et al. [8] ont utilisé le machine learning et la classification de texte pour attribuer un code topographique et morphologique aux CRAP relatifs à un cancer : ils n'ont pas traité l'étape préalable de distinction de ces CRAP relatifs à un cancer.

Une des solutions serait de réutiliser un système existant. Or, ces systèmes ont été développés à partir de CRAP en langue anglaise et nécessitent donc des adaptations pour prendre en compte les spécificités linguistiques du français.

De plus, la plupart des systèmes ci-dessus utilisent la SNOMED-CT ou le NCI-thesaurus pour traiter ces CRAP. Ces deux outils de représentation de l'information médicale sont très utilisés dans la fouille de texte appliquée au domaine biomédical, notamment en oncologie, mais sont peu adaptées pour traiter des textes en langue française. En effet, la SNOMED-CT est une ontologie qui regroupe près de 350 000 concepts dans sa version 2019 et dont seulement 10% ont été traduits en français canadien [source : ihtsdo]. Le NCI-thesaurus est une terminologie créée par le National Cancer Institute (NCI) définissant près de 100 000 termes biomédicaux spécifiques au domaine de l'oncologie. A notre connaissance, il n'existe pas de traduction française du NCI-thesaurus.

Il n'est donc pas possible d'utiliser ces ressources en l'état sans risquer d'être imprécis dans la tâche réalisée.

Il semble alors plus pertinent de développer un système de fouille de texte spécifique aux CRAP écrits en langue française.

1.4.3 L'apport du machine learning

Pour construire un tel système, utiliser le machine learning semble intéressant. En effet, nous avons vu qu'en l'état actuelle des connaissances, il était risqué d'utiliser une terminologie en tant que règle de décision pour des textes rédigés en français : une approche manuelle imposerait donc la construction préalable d'un jeu de règles, ce qui nécessite une excellente expertise du domaine et est couteux en termes de temps et de ressources.

De plus, l'efficacité d'un programme de machine learning est comparable voir supérieur aux systèmes experts [17,18]. Par exemple, Solti et al [17] ont montré que dans un problème de classification binaire de comptes rendus radiologiques, un programme de machine learning est plus performant qu'une approche manuelle par mots-clés, avec une F-mesure maximum de 0.91 et 0.85 respectivement.

Enfin, il permet une bonne généralisation du programme à d'autres domaines ou sous-domaines [19].

Le principal inconvénient d'un système utilisant du machine learning est la nécessité de posséder un jeu de donnée conséquent, pour permettre un apprentissage correct et suffisamment généralisable. Dans le cas d'un apprentissage supervisé, le jeu de donnée doit en plus être labellisé.

Dans son livre *Deep learning with Python* [20], F. Chollet décrit les étapes nécessaires à la construction d'un programme de machine learning. La première étape est de définir précisément le problème, qui peut être vu comme la tâche à réaliser par le programme. Pour cela, il est indispensable d'identifier quelles seront les données en entrée du programme ("input") et quel est le résultat attendu ("output").

Dans notre cas, nos données sont des CRAP en texte libre, et le résultat du programme doit mimer le travail effectué par un registre de cancer sur ces CRAP non codés (partie 1.2.2). Ce résultat s'obtient donc en 2 étapes :

1) Répartir les CRAP en 2 catégories : « cancer » ou « bénin »
2) Affecter un code CIMO3 aux CRAP catégorisés « cancer »

On remarque que chacune de ces étapes consiste à attribuer une catégorie (une classe), parmi un ensemble prédéfini (« cancer » ou « bénin » d'un côté, codes CIMO3 de l'autre), à une donnée textuelle, ce qui décrit une tâche de classification de texte. Bien que l'on puisse remplacer cette tâche par une autre (partie 1.3.2), celle-ci reste centrale lorsque l'on souhaite utiliser le machine learning pour traiter un texte. En effet, une façon courante d'extraire de l'information à l'aide d'un programme d'apprentissage automatique est de considérer le problème comme une tâche de classification [21–23].

Au final, une solution pour automatiser le traitement des CRAP en texte libre au sein d'un registre de cancer en France serait de construire un programme de classification de texte par apprentissage.

1.5 Classification automatique de comptes rendus médicaux et machine learning : état de l'art

Le tableau 1 résume un état de l'art non exhaustif sur la classification automatique de comptes rendus médicaux par machine learning. Pour chaque article, nous présentons le type de compte rendu ainsi que sa langue, le type et le nombre de classes, et le meilleur modèle de classification implémenté (représentation des données textuelles, algorithme de classification (« algo »), performance du modèle).

Tableau 1 : Classification de comptes rendus médicaux par machine learning

Données (langue)	Représentation*	Algo	Classes (nombre)	Performance**	Ref.
CRAP (**français**)	BoW	SVM	Codes topographiques CIMO3 complets (26)	0.72 (micro F)	[8]
			Codes morphologiques CIMO3 complets (18)	0.85 (micro F)	
CRAP (anglais)	Concepts, Negex	SVM	Stades T (4)	0.65 (justesse)	[24]
			Stades N (3)	0.82 (justesse)	
CRAP (anglais)	BoW	SVM	Marges chir. (3)	0.97 (justesse)	[25]
CRAP (anglais)	Concepts, Negex	SVM	Stades T (5)	0.74 (justesse)	[26]
			Stades N (4)	0.87 (justesse)	
CRAP (anglais)	BoW	SVM	Catégories nominales (36)	0.65 (macro F)	[27]
CRAP (anglais)	BoW, concepts	SVM	Codes topographiques CIMO3 3 digits (58)	0.72 (macro F), 0.90 (micro F)	[28]
CRAP (anglais)	BoW, n-grams, nombre de codes CIM9	LR	Maladie cœliaque oui/non (2)	0.92 (f-mesure)	[29]
CRAP (portugais)	BoW	SVM	Codes topographiques CIMO3 3 digits (18)	0.82 (micro F)	[30]
			Codes morphologiques CIMO3 3 digits (49)	0.73 (micro F)	
Certificats de décès (anglais)	BoW, concepts	SVM	Cancer oui/non (2)	0.94 (f-mesure)	[31]
			Codes CIM10 3 digits (85)	0.61 (macro F)	
CR médicaux (allemand)	BoW, concepts	SVM	Diagnostic (4)	0.98 (micro F)	[32]
			Stade de Salmon (4)	0.97 (micro F)	
Certificats de décès (anglais)	BoW, bi-gram, concepts	SVM	Cancer oui/non (2)	0.99 (f-mesure)	[18]
CR d'autopsie (anglais)	BoW	SVM	Codes CIM10 3 digits (8)	0.78 (macro F)	[33]

* BoW : bag of words, negex : détection de la négation par expressions régulières, n-gram : découpage du texte tous les n caractères

** F : f-mesure, variantes micro et macro pour les tâches de classification multi-classe
justesse : pourcentage de données correctement classées

1.6 Objectif de travail

Dans le cadre de ce travail de Master, nous souhaitons proposer une démarche par machine learning supervisé capable d'automatiser le traitement, réalisé au sein d'un registre de cancer, des CRAP rédigés en langue française et non codés par les laboratoires d'anatomopathologie.

Pour cela, nous proposerons trois modèles de classification : un modèle de classification binaire, permettant de distinguer les CRAP relatifs à un cancer des autres, et deux modèles de classification multi-classe, permettant de coder la typologie des CRAP relatifs à un cancer.

2 Matériel

2.1 Source des données

Les données sont issues du registre général des cancers de la Gironde. Il a pour mission de recenser l'ensemble des nouveaux cas de cancer dans le département de la Gironde. Pour certains de ces cas, le registre général a seulement un rôle de signalement : ceux-ci sont alors transmis puis validés par des registres dits « spécialisés ». Cela concerne les tumeurs du système nerveux central (SNC), les hémopathies malignes et les mésothéliomes.

Le recueil a débuté en janvier 2005, en respectant les critères de la figure 4. Il s'agit d'un recueil actif des nouveaux cas auprès des laboratoires d'anatomo-cyto-pathologie du département, grâce à un passage régulier du personnel du registre. Ce dernier récupère les CRAP, comme on l'a vu à la partie 1.2.

Dans le cadre de ce travail, nous avons utilisé des CRAP provenant d'un seul laboratoire. Ils ont été rédigés en langue française à partir de prélèvements réalisés entre 2005 et 2017, et ne sont initialement pas codés. Ils ont alors été annotés manuellement par les ARC du registre : un motif d'exclusion a été attribué pour les carcinomes basocellulaires (« exclu baso »), pour les tumeurs transmises aux registres spécialisés, ainsi que pour les autres tumeurs ne respectant pas les critères d'inclusion de la figure 4 (« ligne exclue avant enregistrement »). Pour les tumeurs traitées par le registre général, un ou plusieurs codes Cimo3 leur a été proposés.

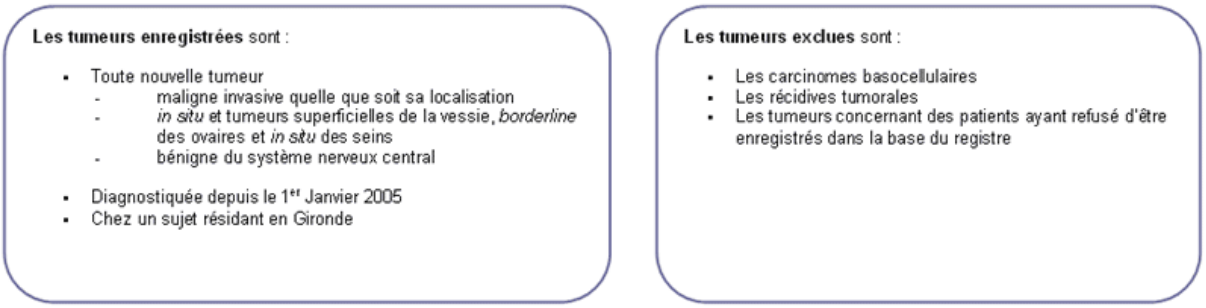

Figure 4: Critères d'inclusion et d'exclusion du registre général des cancers de la gironde

2.2 Outils informatiques

L'ensemble des étapes décrites ci-dessous ont été réalisées à l'aide du langage informatique Python v3.7.1, et des librairies NLTK v3.4 et scikit-learn v0.20.2.

3 Méthodologie

Après avoir défini la tâche à réaliser, ici une tâche de classification, il reste plusieurs étapes à suivre pour développer notre programme de machine learning [20]. Ce sont celles-ci que l'on va détailler dans ce chapitre.

Le choix de présenter certains résultats de l'exploration de nos données dans cette partie est volontaire. En effet, construire un programme de machine learning est une démarche inductive [10,19] : les choix méthodologiques dépendent beaucoup des données en notre possession et de leur analyse. Il n'existe pas non plus de règles précises ou de principes permettant de standardiser ces choix.

3.1 Construction du jeu de données

3.1.1 Données en entrée (« input »)

Certains CRAP annotés n'ont pas été inclus dans l'analyse. En effet, soit ils ne correspondaient pas à une tumeur primitive, soit ils traitaient de plusieurs localisations tumorales, soit ils étaient ambigus. Les critères de non inclusion étaient donc les suivants :

- CRAP avec plusieurs codes Cimo3 (proposés par les ARC du registre)
- CRAP avec un code Cimo3 incomplet
- CRAP avec un code Cimo3 de comportement évolutif /1 (indéterminé si bénin ou malin), /6 (siège métastatique ou secondaire) ou /9 (incertain si primitif ou métastatique)

Au total, 84 745 CRAP ont été inclus et ont été utilisés pour la tâche de classification binaire. Parmi eux, 10 933 possédaient un code CIMO3 complet (résumant à la fois la topographie et la morphologie de la pathologie tumorale maligne) et ont été utilisés pour les tâches de classification multi-classe.

Les données en entrée du programme de classification correspondent au texte de ces comptes rendus. Pour chacun des modèles, nous évalueront 2 choix méthodologiques :

1) L'utilisation du texte du CRAP dans son ensemble
2) L'utilisation de la conclusion seule

3.1.2 Cibles de classification (« output »)

Dans le cas d'une tâche de classification, l'apprentissage est supervisé : les données servant à paramétrer et à évaluer le modèle de classification doivent être des couples <entrée, résultat>.

Pour chacun des modèles, nous avons défini le résultat attendu, c'est-à-dire les classes servant de Gold Standard à notre programme.

3.1.2.1 Classification en « cancer » et « bénin »

La distinction entre tumeur maligne et tumeur bénigne s'est basé ici sur le motif d'exclusion des ARC du registre. En effet, il n'a pas été possible de se baser sur le code Cimo3 car les tumeurs exclues du registre ou transmises aux registres spécialisés ne possèdent pas cette annotation.

La classe « cancer » correspond donc aux CRAP ne possédant pas de motif d'exclusion, et à ceux exclus car relatifs à un carcinome basocellulaire, une tumeur du SNC, une hémopathie ou un mésothéliome.

La classe « bénin » correspond aux CRAP annotés avec le motif « ligne exclue avant enregistrement ». Nous avons en effet fait l'approximation que les tumeurs avec ce motif étaient supposées bénignes.

La figure 5 représente la répartition des données dans ces 2 classes : 63 418 données dans la classe « bénin » et 21 327 dans la classe « cancer ».

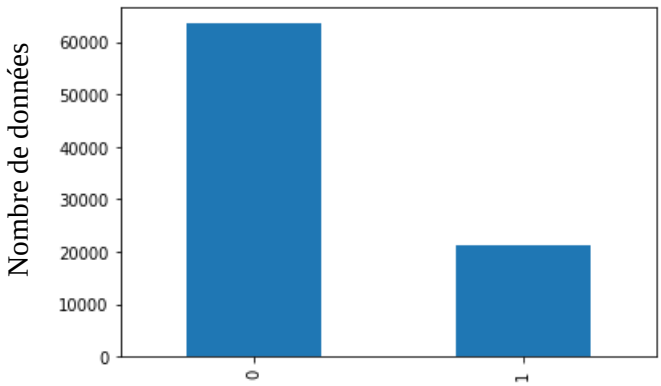

Figure 5: Répartition des classes "cancer" (1) et "bénin" (0)

3.1.2.2 Classification selon la terminologie CIMO3

La terminologie CIMO3 est une terminologie à 2 axes : un axe topographique et un axe morphologique. Pour chacun de ces axes, nous avons défini un premier ensemble de classes cibles constitué de codes CIMO3 : codes topographiques complets (à 4 digits) d'un côté et codes morphologiques à 4 digits de l'autre (le code de comportement ayant été retiré car nous avons considéré que la précision « in situ » (/2) ou « tumeur maligne primaire » (/3) rajoutait trop de bruit au modèle par rapport au gain d'information que l'on obtenait en conservant ce code).

La répartition des données dans ces premiers ensembles est très déséquilibrée, comme le montre les figures 6 et 7. Or, comme nous le verrons à la partie 3.1.3, ce déséquilibre pose problème pour paramétrer un programme de classification. Nous avons donc défini, pour chaque axe, un deuxième ensemble constitué de classes à plusieurs granularités, selon le principe suivant :

Soit une classe du premier ensemble. Si celle-ci comporte 100 données ou plus, alors la classe du deuxième ensemble sera identique (c'est-à-dire un code topographique ou un code morphologique à 4 digits). Sinon la classe du deuxième ensemble sera un code CIMO3 tronquée à 3 digits. Enfin, pour ces classes à 3 digits, si celles-ci comportent moins de 50 données, elles seront remplacées par un groupe (topographique ou morphologique) de codes ; ces groupes sont définis dans le chapitre « Listes numériques » de la CIMO3 et sont présentés en annexe.

Pour chacun de ces 2 ensembles, les classes avec moins de 40 données ont été regroupées dans une classe « autres ».

Au final, chaque axe comprend 2 ensembles de classes distincts (figures 6 et 7), pour autant de modèles de classification :

1) Codes à 4 digits, topographiques (43 classes) ou morphologiques (25 classes)
2) Classes à granularité mixte, topographiques (41 classes) ou morphologiques (28 classes)

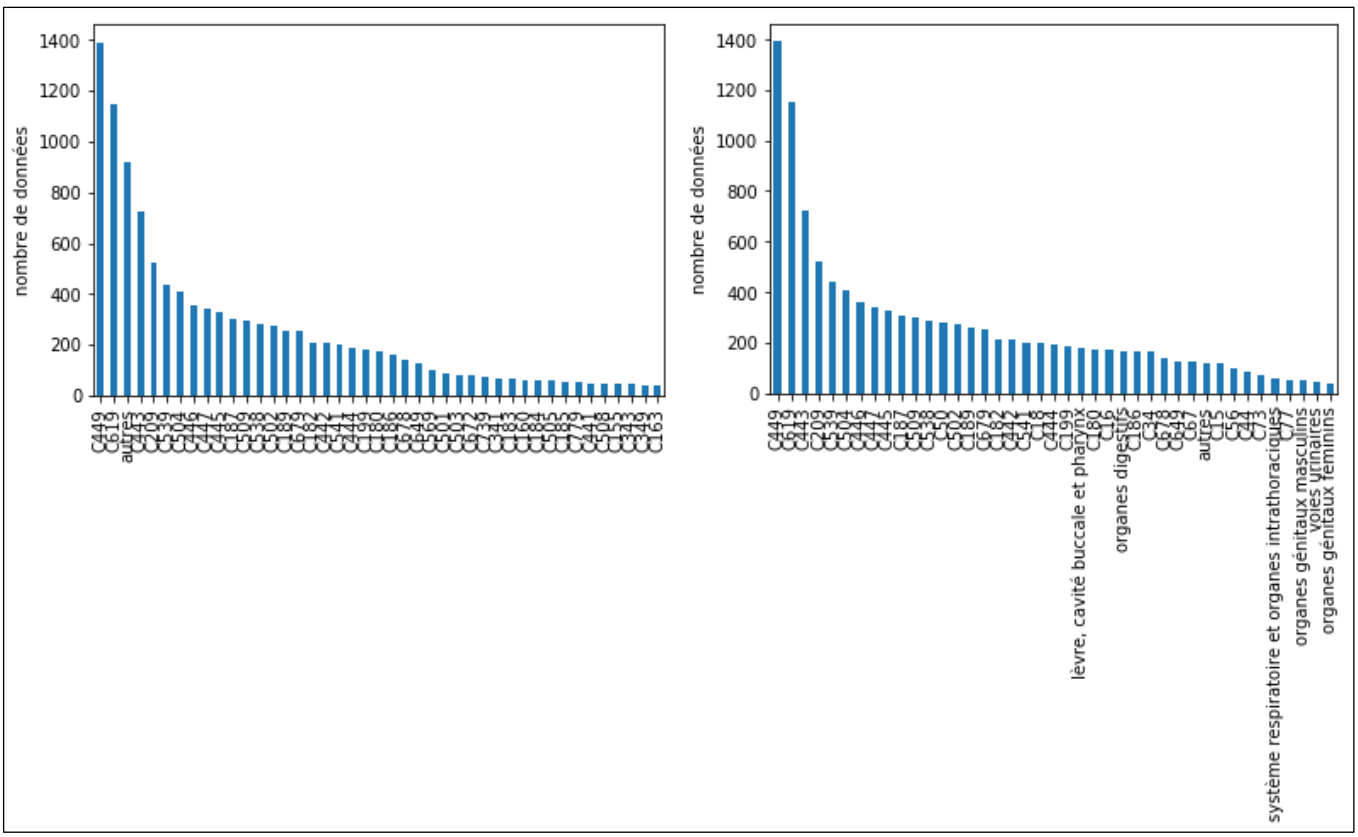

Figure 6 : Répartition des topographies
(codes cimo3 complets à gauche, granularité mixte à droite)

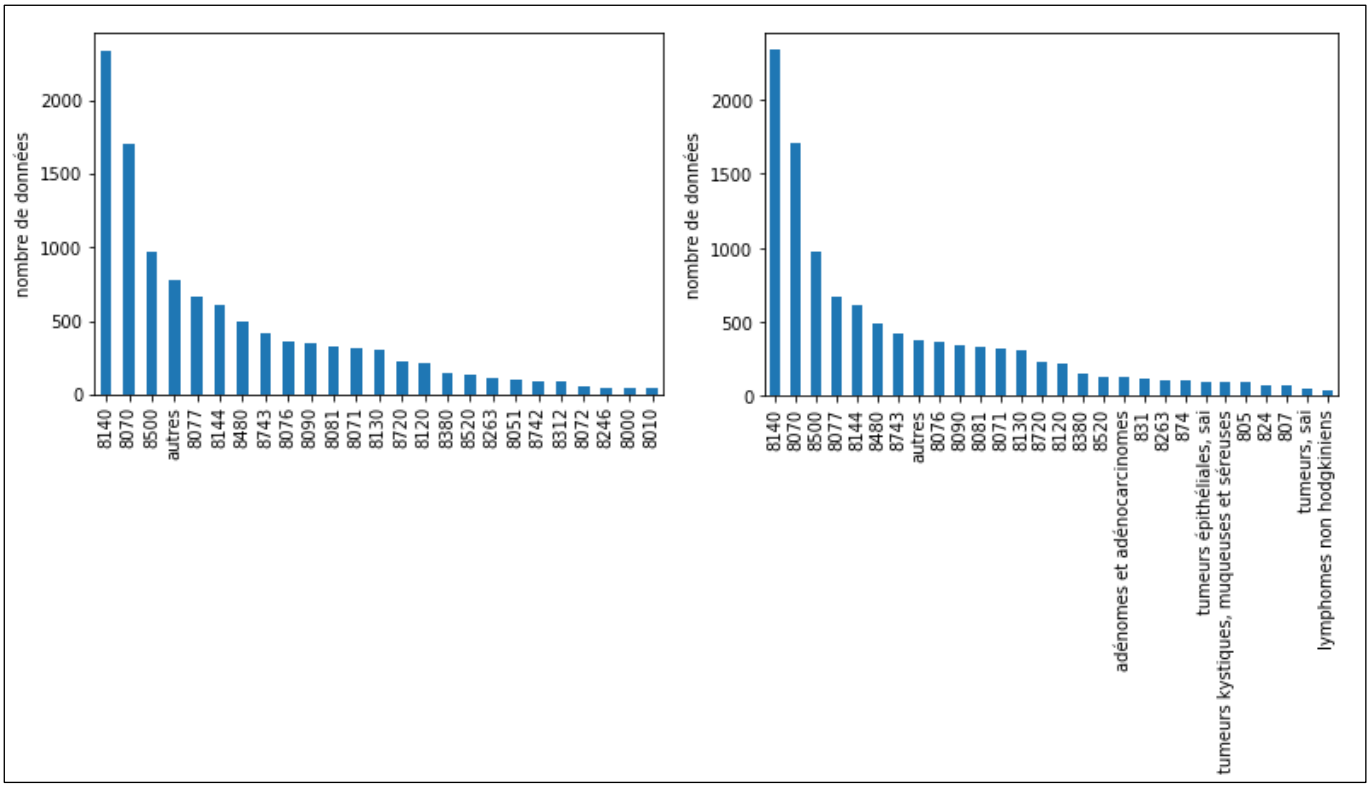

Figure 7 : Répartition des morphologies
(codes cimo3 à 4 digits à gauche, granularité mixte à droite)

3.1.3 La problématique d'un jeu de données déséquilibré

Les algorithmes de machine learning sont construits sur l'hypothèse que le jeu de données est équilibré. Pour une tâche de classification, cela suppose que la répartition des données dans l'ensemble prédéfini de classes est identique. Or, c'est rarement le cas dans la réalité et cela diminue souvent les performances du programme [34].

Pour améliorer le paramétrage d'un programme de machine learning avec un jeu de données déséquilibré, il existe 3 grandes approches :

1) Traitement au niveau du jeu de données lui-même
2) Traitement au niveau de l'algorithme de machine learning utilisé
3) Approche hybride, combinant les avantages des deux premières approches

L'approche utilisée ici est un traitement du jeu de données, et qui consiste en un ré-échantillonnage (« resampling » en anglais) des données servant à paramétrer l'algorithme de classification.

Pour la tâche de classification binaire, nous avons réalisé un sous-échantillonnage (« undersampling » en anglais) en supprimant aléatoirement des données de la classe majoritaire « bénin » ; pour aboutir à une répartition parfaitement équilibrée. Nous avons également paramétré un programme sans ré-échantillonnage, en conservant la répartition initiale (figure 5).

Pour le codage de la topographie et de la morphologie, le traitement a consisté en l'utilisation de deux ensembles cibles de classification : un ensemble avec les codes Cimo3 à 4 digits, et un autre avec plusieurs granularités de codage.

3.2 Protocole d'évaluation

Pour comprendre comment évaluer un programme de machine learning, il faut voir notre jeu de données comme étant divisible en trois sous-ensembles [35] :

1) un jeu d'entraînement (« training set » en anglais) qui permet de paramétrer le programme (on parle alors de modèle d'apprentissage)
2) un jeu de validation (« validation set » en anglais) qui sert à évaluer différents modèles (modèles de classification dans notre cas)
3) un jeu de test (« test set » en anglais) qui permet de vérifier la stabilité des performances du ou des meilleurs modèles évalués.

En effet, évaluer les performances d'un modèle sur les mêmes données que celles ayant servies à construire celui-ci conduit à des résultats biaisés, faussement optimistes. Pour cela il est nécessaire d'utiliser de nouvelles données : les jeux de validation et de test peuvent alors jouer le rôle de « nouvelles données » si l'on considère ces données comme indépendantes et identiquement distribuées (i.i.d.). [36].

Dans un premier temps, notre jeu de données est séparé en deux sous-ensembles : un jeu de développement[1] et un jeu de test. Le jeu de développement fera office de jeu d'entrainement pour paramétrer le ou les meilleurs modèles retenus ; ceux-ci seront alors testés grâce au jeu de test. En général, 80% des données sont réservées au jeu de développement et 20% au jeu de test.

Le jeu de développement permet d'évaluer différents modèles et de sélectionner le ou les meilleurs modèles : on parle de « model selection ». Les figures 8 et 9 résument les deux principales approches pour cela : la « hold-out validation » et la validation croisée (« k-fold cross validation », avec k le nombre de « plis »).

La validation hold-out est l'approche la plus simple à mettre en œuvre. Le jeu de développement est séparé en un jeu d'entrainement et un jeu de validation, dans les mêmes proportions que précédemment : 80% des données du jeu de développement sont réservés pour l'entrainement des différents programmes, 20% pour l'évaluation de ceux-ci.

Cette approche est suffisante lorsqu'on estime que l'on possède suffisamment de données pour que les jeux de validation et de test soient statistiquement représentatifs de celles-ci.

La validation croisée est plus complexe. En effet, elle est beaucoup plus coûteuse en ressources informatiques et est sujette à un risque d'erreurs d'implémentation, notamment si l'on pratique un ré-échantillonnage sur le jeu d'entrainement. Cette approche est utile lorsqu'on fait l'hypothèse que notre jeu de données n'est pas de taille suffisante pour être représentatif au niveau des jeux de validation et de test, car elle permet d'estimer la variance de notre modèle.

[1] Ce choix de nom est personnel, dans un souci de clarification des concepts

Son principe est le suivant : le jeu de développement est divisé en k sous-ensembles de taille égale, avec k-1 sous-ensembles servant de jeu d'entrainement et le dernier de jeu de validation. Puis il faut procéder par itération de façon à ce que chaque sous-ensemble serve en tant que jeu de validation. La performance du modèle correspond alors à la moyenne des k mesures obtenues. La figure 9 schématise une validation croisée à trois sous-ensembles (« 3-fold cross validation »).

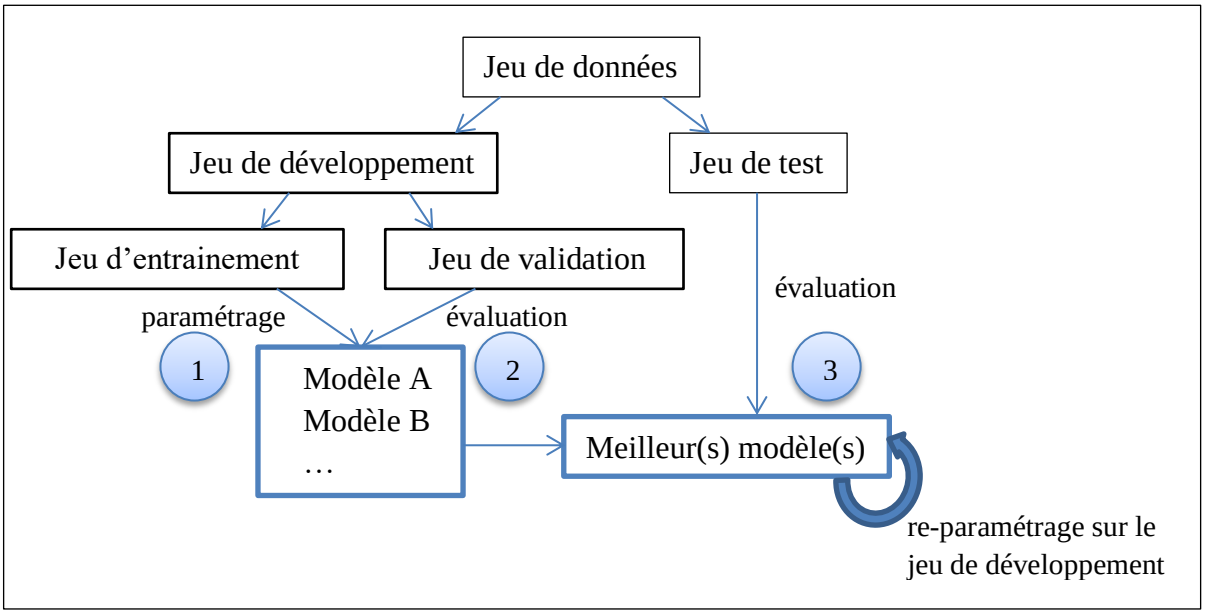

Figure 8 : Validation hold-out (① et ②)

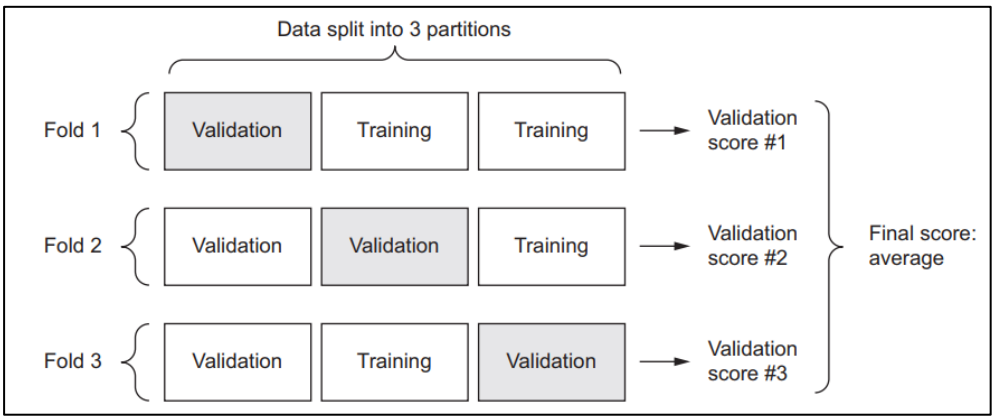

Figure 9: 3-fold cross-validation, par F Chollet, deep learning with python

** erreur: 1 seul jeu de validation par "fold"

Le choix du protocole de validation dépend donc essentiellement du nombre de données disponible pour chaque tâche de classification, et notamment au nombre de données réservé aux jeux de validation et de test. Pour autant, il n'existe pas de standard dans le choix de tel ou tel protocole.

Pour la tâche de classification binaire, nous avons considéré que nos 84 745 CRAP étaient suffisants pour utiliser le protocole de validation hold-out, et que réaliser une validation croisée serait trop coûteux informatiquement.

Pour le codage de la topographie et de la morphologie, nous utiliserons une validation croisée à 5 sous-ensembles.

3.2.1 Mesures d'évaluation

Il existe différentes mesures pour évaluer la performance d'un modèle de classification. Pour comprendre comment elles sont calculées, on peut organiser les résultats du modèle dans une matrice de confusion (tableau 2). Ainsi, les vrais positifs et vrais négatifs correspondent aux CRAP correctement classés, les faux négatifs aux CRAP de la classe i prédits dans une autre classe et les faux positifs aux CRAP prédits i alors qu'ils appartiennent à une autre classe.

Tableau 2 : Matrice de confusion

		Classes prédites	
		Classe i	~ Classe i
Classes réelles (Gold standard)	Classe i	**Vrais positifs (VP)**	**Faux négatifs (FN)**
	~ Classe i	**Faux positifs (FP)**	**Vrais négatifs (VN)**

Dans le domaine médical et lorsque le jeu de données est déséquilibré, c'est-à-dire lorsque la répartition des données au sein des différentes classes prédéfinies est inégale, les mesures les plus couramment utilisées sont le rappel (ou sensibilité), la précision (ou valeur prédictive positive) et la F-mesure. Leurs valeurs sont données par l'équation 1.

Pour une tâche de classification, on définit le rappel comme étant la proportion de données correctement classées parmi l'ensemble des données d'une classe i. Cela reflète la capacité du modèle à retrouver les données de cette classe i.

La précision est la proportion de données correctement classées parmi les données prédites dans une classe i. Cela reflète la capacité du modèle à identifier et distinguer les données de cette classe i.

La F-mesure est la moyenne harmonique du rappel et de la précision.

$$Rappel = \frac{VP}{VP + FN}$$

$$Précision = \frac{VP}{VP + FP}$$

$$F - mesure = 2 * \frac{rappel * précision}{rappel + précision}$$

Équation 1 : Rappel, précision et f-mesure

Dans le cas d'une classification multi-classe, il existe 2 grandes variantes pour le calcul de ces mesures :

1) Mesure 'micro', mesure globale qui compte le nombre total de vrais positifs, faux positifs et faux négatifs.
2) Mesure 'macro', qui correspond à la moyenne non pondérée des mesures calculées au niveau de chaque classe.

Ces mesures ont donc servi à évaluer chaque modèle créé et à sélectionner le meilleur modèle pour chaque tâche (qui sera alors ré-entrainé et réévalué sur le jeu de test), en suivant ces critères : les modèles avec la meilleure F-mesure ont été privilégiés dans un premier temps. Puis, lorsque les valeurs de F-mesure étaient proches, nous avons comparé soit le rappel soit la précision : pour la tâche de classification binaire, nous avons choisi le modèle avec le meilleur rappel car il est nécessaire pour un registre d'être exhaustif dans son recueil des cas. Pour le codage des items de typologie, le modèle final sera le modèle avec la meilleure précision ; en effet, nous avons considéré qu'il était préférable pour cette tâche d'automatiser le codage de moins de CRAP mais de façon plus précise, plutôt que l'inverse.

3.3 Représentation des données

Cette étape est fondamentale et consiste à transformer les données de façon à ce qu'elles puissent être traitées par un algorithme de machine learning. Dans le cas d'un corpus (jeu de données textuelles) et d'une tâche de classification, cette transformation se déroule classiquement en 3 étapes :

1) Etape de prétraitement : normalisation du corpus
2) Indexation de chaque texte : un texte dj est représenté par un vecteur numérique $\overrightarrow{dj}$ = $\langle w_{1j},\ldots,w_{|T|j}\rangle$ où T est le jeu de termes, et $0 \leq w_{kj} \leq 1$ est le poids du terme k dans dj.
3) Réduction de la dimension des vecteurs créés

3.3.1 Normalisation du corpus

Cette étape consiste à harmoniser la syntaxe des textes d'un corpus. Ainsi, chaque CRAP a été normalisé de cette façon :

- Texte en minuscule
- Suppression des accents, de la ponctuation et des caractères spéciaux
- Suppression des nombres entiers ou décimaux
- Remplacement des contractions (j', l', n',…) par leur forme pleine (je, le, ne,…)
- Suppression des mots vides (« stopwords »), c'est-à-dire des mots trop fréquents pour être informatifs (ex : articles, pronoms,…)

Puis 2 choix d'implémentation ont été retenus : un jeu normalisé avec racinisation des mots (« stemming » en anglais) et un jeu sans racinisation.

En linguistique, la racinisation est un procédé de transformation des flexions en leur radical ou racine. Cela consiste à supprimer les préfixes et suffixes d'un mot. Par exemple, le verbe *chercher* a pour racine *cherch*.

3.3.2 Indexation en sac de mots (“bag of words”)

Pour indexer notre corpus normalisé, il faut définir ce qu’est un terme à l’échelle d’un texte et comment pondérer ces termes.

Un terme peut être vu comme la plus petite unité lexicale d’un texte. On parle également de « token », en référence au procédé consistant à découper un texte en termes : la tokénisation.

Nous avons choisi de définir un terme comme étant un mot : on parle alors de tokénisation en mots et d’indexation (ou de représentation) en sac de mots (« bag of words » en anglais).

Nous avons également choisi d’élargir la définition d’un mot aux abréviations et scores (ex : T2N1M0), en faisant l’hypothèse que ces unités lexicales étaient informatives pour notre tâche. Nous avons ainsi, grâce à des expressions régulières, extrait les mots simples, les abréviations et les scores de chaque texte du corpus.

A ce stade, il est important de préciser à quoi correspond notre jeu de termes : celui-ci est créé à partir du jeu d’entrainement (défini plus haut) et servira de features (attributs) pour décrire l’ensemble du corpus.

Ensuite, pour pondérer un terme, c’est-à-dire pour estimer numériquement la contribution sémantique d’un terme k dans un texte dj du corpus, il existe plusieurs possibilités. Le choix le plus simple est d’utiliser une valeur binaire : 0 si le terme est absent du texte, 1 s’il est présent. Un des choix les plus performants dans le domaine de la classification de texte est d’utiliser la mesure tf.idf [11,20]. Sa valeur est donnée par l’équation 2, où $\#(t_k, d_j)$ est la fréquence du terme k dans le texte dj, $|Tr|$ est le nombre total de textes dans le jeu d’entrainement et $\#Tr(t_k)$ est la fréquence des textes de Tr dans lesquels le terme k apparait. C’est le choix que nous avons fait dans ce travail, en normalisant les valeurs pour qu’elles restent dans l’intervalle [0,1].

$$tf.idf(t_k, d_j) = \#(t_k, d_j) \,.\log \frac{|Tr|}{\#Tr(t_k)}$$

Équation 2: tf.idf

Au final, notre corpus peut être représenté sous forme d’une matrice terme-document : un tableau dont les lignes correspondent aux textes, les colonnes aux k features (jeu de termes) et les valeurs au poids (tf.idf) du terme k dans le texte dj.

Par exemple, prenons un corpus fictif de 3 CRAP : (CRAP 1) « Le stade du cancer est T2N1M0. », (CRAP 2) « Masse de 3cm du sein droit, signes en faveur d’une néoplasie. », (CRAP 3) « prélèvement du sein droit : limite d’exérèse à 2 cm, marges chirurgicales saines. ». Un stemming a été réalisé, et seuls les CRAP 1 et 2 font partis du jeu d’entrainement. Nous obtenons alors la matrice terme-document suivante :

Tableau 3 : Matrice terme-document d’un corpus fictif de trois CRAP

	canc	cm	droit	faveur	mass	neoplas	sein	sign	stad	t2n1m0
CRAP 1	0.577	0	0	0	0	0	0	0	0.577	0.577
CRAP 2	0	0.378	0.378	0.378	0.378	0.378	0.378	0.378	0	0
CRAP 3	0	0.577	0.577	0	0	0	0.577	0	0	0

3.3.3 Réduction de la dimensionnalité

Grossièrement, on peut définir la dimension d'un espace vectoriel comme étant la taille des vecteurs le composant. Par exemple, $\vec{v}$ = (3,5,2) est un vecteur de dimension 3.

Dans notre cas, cette dimension correspond au nombre de termes (features) dans notre jeu de termes. On remarque que malgré un premier filtrage par les étapes de pré-traitement et d'indexation, la dimension des vecteurs créés restent très grandes : plusieurs milliers de termes sont présents dans un corpus de CRAP.

A la différence d'une tâche de recherche d'information, cette haute dimensionnalité peut être problématique, avec un risque de sur-apprentissage (« overfitting » en anglais) [19,37], c'est-à-dire la production d'un modèle trop spécifiques (on parle de spécialisation) aux données du jeu d'entrainement et qui s'adaptera mal à de nouvelles données (mauvaise généralisation du modèle). Fuhr et al. [37] préconise l'utilisation d'un terme pour 50-100 données d'entrainement. Pour notre tâche de classification binaire, il faudrait donc idéalement utiliser entre 677 et 1356 termes pour tester notre modèle final.

Cependant, diminuer le nombre de termes fait perdre de l'information au modèle et, malgré les suggestions ci-dessus, il est dangereux de se limiter à l'évaluation d'une seule dimension. Ainsi, il est courant de construire plusieurs modèles lors de l'étape de « model selection » (sur le jeu de développement) pour tester plusieurs dimensions, en allant d'un modèle simple vers un modèle plus complexe (à plus haute dimension). Par ailleurs, il ne faut pas procéder aléatoirement pour réduire cette dimension, mais utiliser une des 2 grandes approches suivantes :

1) Sélection de termes (« term selection » en anglais), avec T' un sous-groupe de T
2) Extraction de termes (« term extraction » en anglais), avec T' ne contenant pas forcément le même type de termes que T (ex: T' peut contenir des concepts alors que T contient seulement des mots).

L'approche choisie ici est une approche par sélection de termes : un deuxième filtrage des termes est réalisé grâce à des fonctions qui mesurent l' « importance » des termes pour la tâche de classification à réaliser. Une des fonctions les plus performantes est le test du Chi² [33], qui va calculer l'association entre un terme et une classe : $\chi^2(t_k,c_i)$. La dimension finale correspond alors aux k termes les plus significatifs.

En suivant les principes précédents, nous avons évalué des modèles avec 10, 50, 200, 500, « significatif » et « significatif *2 » termes (avec « significatif » les termes ayant un χ^2 significatif, c'est-à-dire $\chi^2 > 3.84$ pour $\alpha = 0.05$).

Nous avons également évalué un modèle sans réduction de dimensionnalité (k = « all »).

3.4 Algorithmes de classification

Une fois les données transformées, elles peuvent être traitées par l'algorithme de machine learning, qui est ici un algorithme de classification.

Pour la tâche de classification binaire, quatre algorithmes de classification ont été évalués :

1) Algorithme à support vectoriel (« support vector machine » en anglais, SVM), avec 4 noyaux différents (linéaire, rbf, polynomial et sigmoïde)
2) Forêt aléatoire (« random forest » en anglais, RF)
3) Arbre de décision (« decision tree » en anglais, DT)
4) Régression logistique (« logistic regression » en anglais, LR)

Grâce à une démarche inductive, seuls les algorithmes avec les meilleurs résultats pour la tâche de classification binaire ont été utilisés pour le codage de la typologie des cancers. Soit les algorithmes SVM (avec les noyaux linéaire et rbf), RF et LR.

Ces algorithmes ont été utilisés avec leurs paramètres (on parle d' « hyper-paramètres ») par défaut, sans chercher à optimiser ceux-ci lors de la phase de développement.

3.5 Résumé des modèles évalués

3.5.1 Classification en « cancer » et « bénin »

La figure 10 représente les variations méthodologiques implémentées dans les différents modèles pour la tâche de classification binaire. Au sens strict, 448 modèles ont été évalués. Pour simplifier la présentation des résultats, nous parlerons de modèles pour caractériser les choix faits au niveau des données et de leur représentation, ainsi qu'au niveau du ré-échantillonnage du jeu d'entrainement. Au final, huit grands modèles ont été évalués, chacun avec une variation dans le nombre de features et dans le choix de l'algorithme de classification :

Modèles avec CRAP complet			Modèles avec conclusion seule		
	Undersampling	**Stemming**		**Undersampling**	**Stemming**
Modèle A	Oui	Oui	**Modèle E**	Oui	Oui
Modèle B	Oui	Non	**Modèle F**	Oui	Non
Modèle C	Non	Oui	**Modèle G**	Non	Oui
Modèle D	Non	Non	**Modèle H**	Non	Non

3.5.2 Classification selon la terminologie Cimo3

La figure 11 représente les variations méthodologiques implémentées dans les différents modèles pour l'attribution d'un code topographique et morphologique aux CRAP relatifs à un cancer. De la même façon que pour la tâche précédente, nous parlerons de modèles pour caractériser les choix faits au niveau des données et de leur représentation, ainsi qu'au niveau du choix des classes cibles. Ainsi, pour chaque axe de la terminologie CIMO3 (topographique et morphologique), huit grands modèles ont été évalués, chacun avec une variation dans le nombre de features et dans le choix de l'algorithme de classification :

Modèles avec CRAP complet			Modèles avec conclusion seule		
	Classes cibles	**Stemming**		**Classes cibles**	**Stemming**
Modèles A_t et A_m*	Codes CIMO3	Oui	**E_t et E_m**	Codes CIMO3	Oui
Modèles B_t et B_m	Codes CIMO3	Non	**F_t et F_m**	Codes CIMO3	Non
Modèles C_t et C_m	Granularité mixte	Oui	**G_t et G_m**	Granularité mixte	Oui
Modèles D_t et D_m	Granularité mixte	Non	**H_t et H_m**	Granularité mixte	Non

* t pour topographie, m pour morphologie

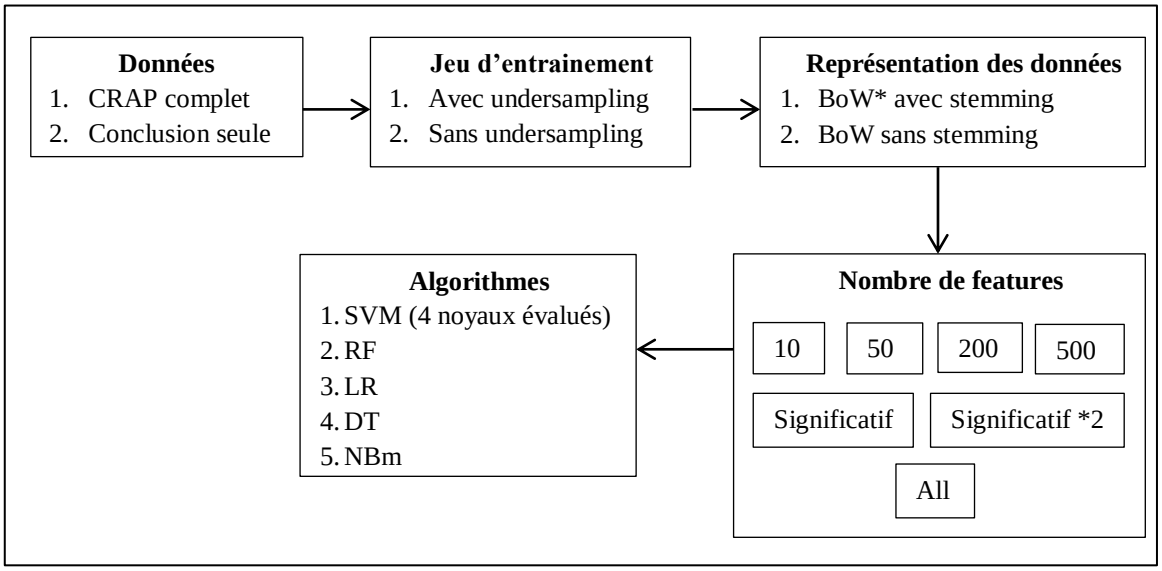

Figure 10: Variables des modèles de classification binaire

* bow signifie « bag of words »

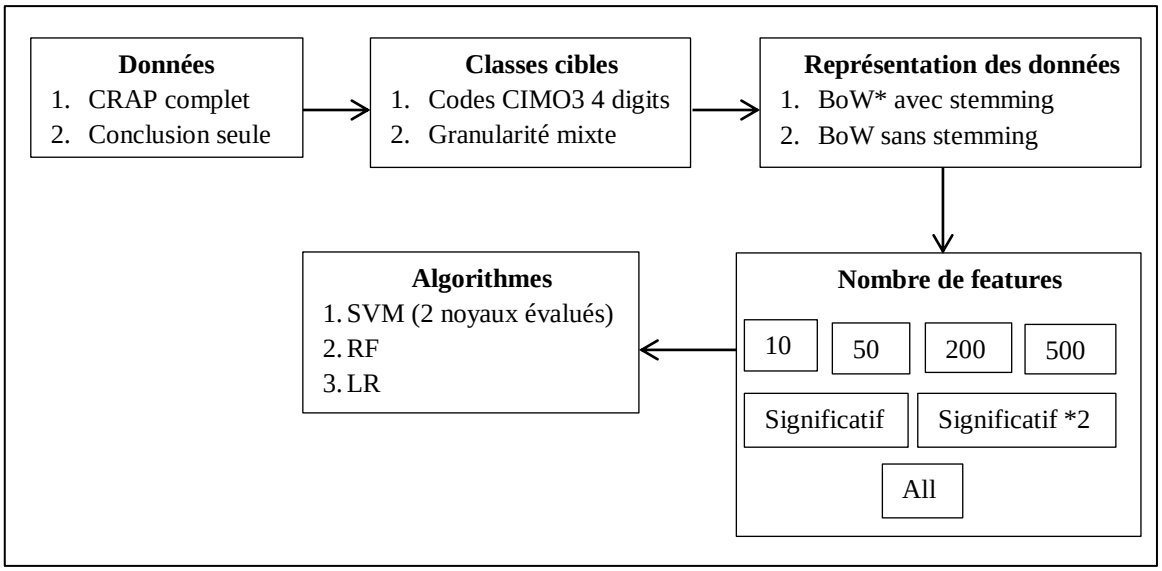

Figure 11 : Variables des modèles de classification CIMO3

4 Résultats

La majorité des résultats sont fournis en annexe. Pour chaque modèle de classification, les mesures présentées correspondent à celles pour lesquelles le nombre de features maximise la F-mesure. Pour les tâches de classification multi-classe, ce sont les mesures « macro » qui sont fournies (sauf indication contraire).

4.1 Classification binaire

4.1.1 Evaluation des modèles de classification

Lorsque l'on utilise seulement la conclusion des CRAP en tant que donnée d'entrée du programme, le meilleur modèle sur le jeu de validation est le modèle G sans réduction de dimensionnalité et avec l'algorithme de classification RF. Sa F-mesure est de 0.959, avec un rappel de 0.951 et une précision de 0.967.

Globalement, les performances des modèles avec conclusion seule sont inférieures à celles des modèles avec le CRAP complet en tant que donnée d'entrée.

Le tableau 3 présente les résultats des trois meilleurs algorithmes pour les modèles de classification utilisant les CRAP complets. En termes de F-mesure, les modèles les plus performants sont les modèles C et D avec l'algorithme RF (0.963). On note que les modèles avec undersampling sur le jeu d'entrainement (modèles A et B) améliorent le rappel de façon non négligeable (écart de 1.5% entre les meilleurs modèles), sans trop dégrader la F-mesure (écart de 0.3%). Le choix du modèle final se portera donc sur un de ces deux modèles, comme expliqué à la partie 3.2.1.

Tableau 4: Performances des modèles de classification binaire avec CRAP complet, sur le jeu de validation

Résultats du jeu de validation				
	Meilleure F-mesure	Rappel	Précision	Nombre de features
Modèle A				
RF	**0.955**	**0.972**	0.938	« significatif *2 »
LR	0.953	0.963	0.944	« all »
SVM (linéaire)	**0.959**	0.971	0.947	« all »
Modèle B				
RF	**0.955**	0.97	0.941	« all »
LR	0.954	0.964	0.944	« all »
SVM (linéaire)	**0.96**	**0.972**	0.949	« all »
Modèle C				
RF	**0.963**	0.954	0.972	« significatif »
LR	0.954	0.944	0.965	« all »
SVM (linéaire)	**0.961**	**0.957**	0.965	« significatif *2 »
Modèle D				
RF	**0.963**	0.954	0.972	« significatif »
LR	0.954	0.942	0.967	« significatif *2 »
SVM (linéaire)	**0.962**	**0.957**	0.967	« significatif *2 »

Les figures 12 et 13 présentent les valeurs de rappel et de F-mesure en fonction du nombre de features sélectionné, pour les trois meilleurs algorithmes des modèles A et B.

On remarque que les performances de l'ensemble des modèles augmentent rapidement jusqu'au nombre « significatif », c'est-à-dire au nombre de features ayant un χ^2 significatif. Puis les valeurs soient augmentent soient diminuent lentement selon les modèles et les algorithmes. On note aussi que les mesures maximales de rappel sont obtenues pour les deux modèles avec l'algorithme RF et le sous-groupe de features « significatif » (0.972 pour le modèle A et 0.973 pour le modèle B). Les mesures maximales de F-mesure sont quant à elles obtenues avec l'algorithme SVM à noyau linéaire et le sous-groupe « all » pour les deux modèles (0.959 pour le modèle A et 0.96 pour le modèle B)

On remarque aussi que l'écart de rappel entre les modèles avec RF et les modèles avec SVM est faible, alors que l'écart de F-mesure est plus significatif. Pour le choix du modèle final, nous avons donc décidé de privilégier un modèle avec l'algorithme SVM à noyau linéaire. Avec cet algorithme, les valeurs de rappel et de F-mesure sont maximisées en utilisant toutes les features.

Nous avons également vu à la partie 3.3.3 qu'utiliser des vecteurs de plus faible dimension était préférable ; malgré le fait qu'avec cet algorithme, les performances du modèle B soient légèrement supérieures à celles du modèle A (F-mesure à 0.96 contre 0.959, rappel à 0.972 contre 0.971) sur le jeu de validation, la dimension plus petite des vecteurs créés (9336 pour le modèle A contre 13802), liée à l'étape de stemming, nous a fait retenir le modèle A avec l'algorithme SVM linéaire et le sous-groupe de features « all » en tant que modèle final.

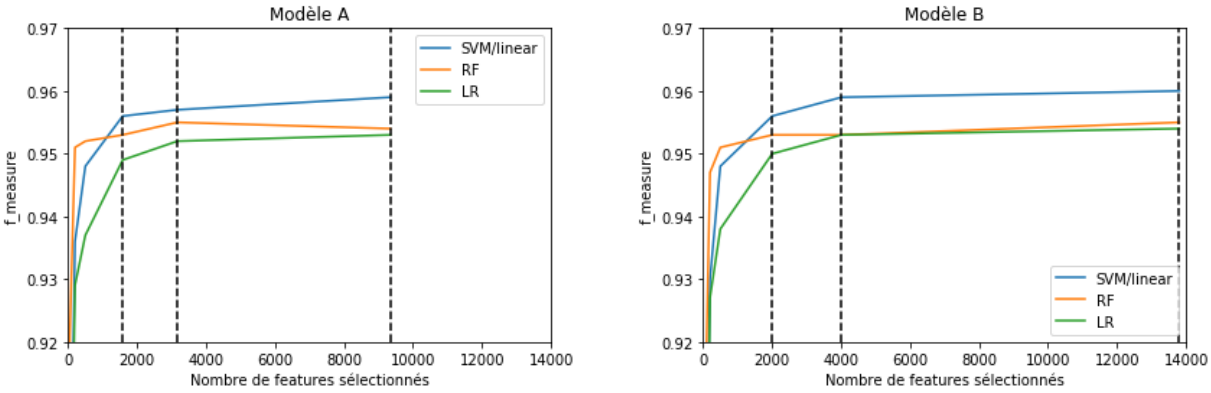

Figure 12 : F-mesure des modèles A et B en fonction du nombre de features

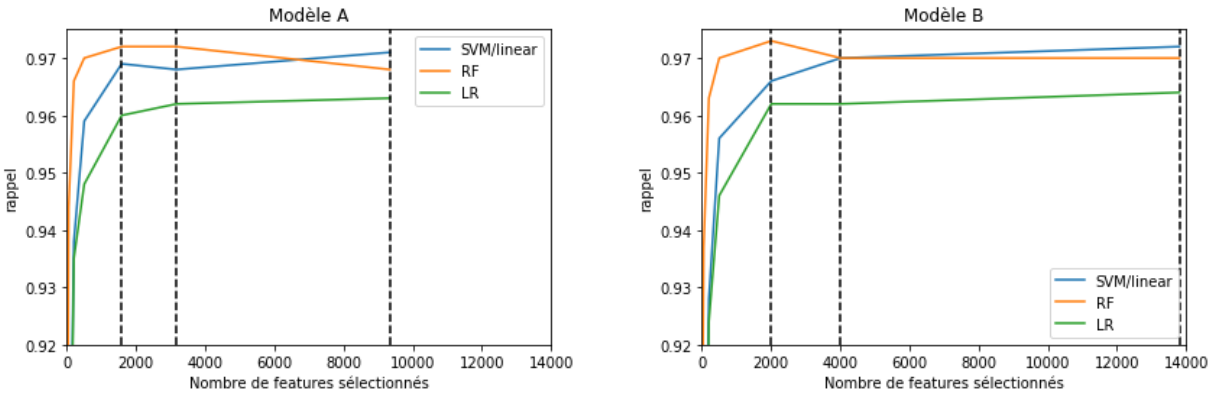

Figure 13 : Rappel des modèles A et B en fonction du nombre de features

4.1.2 Modèle final

Le modèle final a donc été entrainé sur le jeu de développement puis évalué sur le jeu de test.

A partir du jeu de développement, 10 286 termes « racinisés » ont été extraits et correspondent aux features utilisées pour représenter notre corpus.

Les prédictions du modèle pour les 16 949 CRAP complets du jeu de test sont résumées dans le tableau 4. Parmi les 4265 CRAP relatifs à un cancer, 4152 ont été correctement classés (vrais positifs), 113 ont été classés comme étant relatifs à une tumeur bénigne (faux négatifs). Parmi les 12684 CRAP relatifs à une tumeur bénigne, 12488 ont été correctement classés (vrais négatifs) et 196 ont été mal classés (faux positifs).

Au final, la F-mesure est de 0.964, le rappel de 0.974 et la précision de 0.955.

Tableau 5 : Matrice de confusion

		Classes prédites		
		Cancer	Bénin	*Total*
Classes réelles	Cancer	**4152**	**113**	*4265*
	Bénin	**196**	**12488**	*12684*
	Total	*4348*	*12601*	***16949***

4.1.2.1 Analyse des erreurs de classification

Pour comprendre pourquoi le modèle de classification s'était trompé, nous avons analysé le texte de 10% des faux négatifs (soit 11 CRAP) et 10% des faux positifs (soit 20 CRAP). Suite à cette analyse, nous avons estimé qu'il y avait 3 grandes causes d'erreurs :

Premièrement, une mauvaise définition du Gold standard. En effet, 9 faux négatifs sur 11 possèdent le motif d'exclusion « carcinome basocellulaire » et ont donc été initialement classés en « cancer ». Or, à la relecture, aucun de ces CRAP ne parle de carcinome basocellulaire. Parmi les faux positifs, 10 sur 20 correspondent bien à des carcinomes (dont 8 étaient des carcinomes basocellulaires), mais ne rentrent pas dans les critères d'inclusion du registre. Aussi, deux comptes rendus concluent à une maladie de Bowen (un faux négatif et un faux positif) ; or, le classement de ce cancer est aléatoire avec notre méthodologie de construction du Gold standard. Au total, 67.7% des erreurs de classification analysées proviennent d'un mauvais Gold standard et non de l'algorithme en lui-même.

Deuxièmement, une absence de prise en compte des lésions précancéreuses par nos cibles de classification. En effet, deux faux positifs comprennent le groupe nominal « lésion de bas grade ». Or, il existe une différence de considération selon la localisation de ce type de lésions : certaines sont considérées comme des cancers (ex : tumeurs du SNC) alors que d'autres n'impliquent pas la même prise en charge qu'un cancer et il est difficile de les séparer entre nos deux classes « cancer » et « bénin ».

Enfin, une ambiguïté syntaxique dans l'écriture des comptes rendus, qu'il n'est pas possible de prendre en compte informatiquement avec notre représentation des données décrites à la partie 3.3 . Par exemple, 6 faux positifs sur 20 utilisent la négation pour exprimer une idée : « absence de résidu carcinomateux », « pas de critère de transformation maligne », « ne permet pas d'éliminer une lésion invasive »,… On remarque également d'autres syntagmes nominaux ambigus, tel que « hypothèse d'un foyer de néoplasie », « prolifération mélanocytaire », « marges chirurgicales saines »,… Ces syntagmes ne peuvent en effet pas être pris en compte par le modèle de classification puisque chaque mot de ces groupes est séparé et pris en compte individuellement par l'algorithme.

4.2 Classification en codes topographiques

4.2.1 Evaluation des modèles de classification

Pour chaque ensemble de classes cibles, les modèles utilisant seulement la conclusion des CRAP sont globalement moins performants que ceux utilisant le CRAP complet : lorsque les classes correspondent aux codes CIMO3, le meilleur modèle sur le jeu de validation est le modèle F_t sans réduction de dimensionnalité et avec l'algorithme SVM à noyau linéaire, avec une F-mesure de 0.465 (écart-type = 0.018), une précision de 0.52 (0.027) et un rappel de 0.454 (0.017). Pour l'ensemble de classes à granularité mixte, le modèle G_t sans réduction de dimension et avec l'algorithme SVM à noyau linéaire est le plus performant, avec une F-mesure de 0.581 (0.02), une précision de 0.648 (0.031) et un rappel de 0.57 (0.018).

Le tableau 4 présente les résultats des trois meilleurs algorithmes pour les modèles de classification utilisant les CRAP complets. On remarque que les modèles avec l'ensemble de classes à granularité mixte sont significativement supérieurs aux autres (au risque $\alpha = 0.05$). Pour ceux-ci (modèles C_t et D_t), l'algorithme SVM à noyau linéaire donne la meilleur F-mesure (0.642 et 0.632), avec une différence statistiquement significative par rapport aux autres algorithmes (au risque $\alpha = 0.05$).

Pour déterminer quel modèle maximise la précision (tout en conservant une bonne F-mesure), nous présentons les valeurs de précision et de F-mesure en fonction du nombre de features, pour l'algorithme SVM linéaire des modèles C_t et D_t (figure 14). On voit que la précision est maximale (0.692, écart-type = 0.022) avec le modèle C_t et un nombre de features égal à deux fois le nombre de features avec un χ^2 significatif (« significatif *2 »). La F-mesure est quant à elle maximale (0.642, écart-type = 0.013) pour ce même modèle combiné à l'ensemble des features. Pour autant, que ce soit en termes de précision ou de F-mesure, il n'y a pas de différence statistiquement significative entre le modèle C_t combiné à « significatif *2 » features et ce même modèle combiné à l'ensemble des features (au risque $\alpha = 0.05$).

Nous avons donc entrainé deux modèles sur le jeu de développement, construits sur la base du modèle C_t et dont les données sont décrites soit par « significatif *2 » features, soit par toutes les features disponibles.

Tableau 6 : Performances des modèles de classification topographique avec CRAP complet, sur le jeu de validation

Résultats du jeu de validation				
	Meilleure F-mesure	Rappel	Précision	Nombre de features
Modèle A_t				
SVM (linéaire)	**0.487 (0.012)**	0.483 (0.014)	**0.537 (0.027)**	« all »
LR	0.446 (0.008)	0.439 (0.01)	0.488 (0.011)	« all »
RF	0.431 (0.01)	0.427 (0.011)	0.463 (0.021)	200
Modèle B_t				
SVM (linéaire)	**0.479 (0.012)**	0.476 (0.013)	**0.52 (0.022)**	« all »
LR	0.44 (0.011)	0.43 (0.012)	0.489 (0.009)	« all »
RF	0.436 (0.013)	0.43 (0.015)	0.469 (0.018)	200
Modèle C_t				
SVM (linéaire)	**0.642 (0.013)**	0.634 (0.011)	**0.69 (0.018)**	« all »
LR	0.607 (0.01)	0.596 (0.013)	0.68 (0.019)	« all »
RF	0.551 (0.01)	0.542 (0.011)	0.595 (0.014)	200
Modèle D_t				
SVM (linéaire)	**0.632 (0.013)**	0.624 (0.012)	**0.679 (0.016)**	« all »
LR	0.598 (0.012)	0.586 (0.013)	0.664 (0.021)	« all »
RF	0.546 (0.005)	0.538 (0.004)	0.601 (0.013)	500

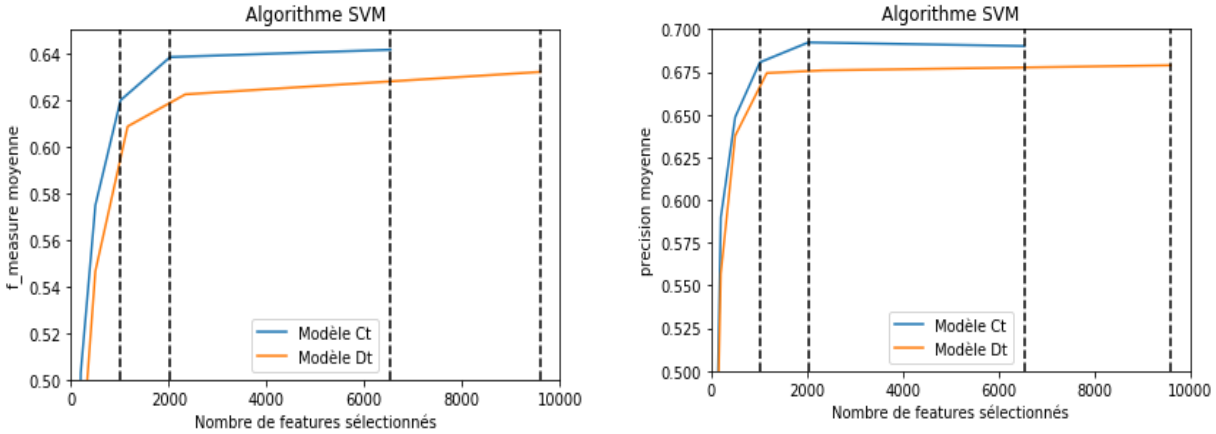

Figure 14 : F-mesure et précision des modèles C_t et D_t en fonction du nombre de features

4.2.2 Modèles finaux

Nous présentons ici les résultats du jeu de test.

Pour le modèle C_t avec « all » features, 7031 termes « racinisés » ont été extraits. La F-mesure est de 0.634, la précision de 0.682 et le rappel de 0.623. La micro F-mesure est de 0.702.

Pour le modèle C_t avec « significatif *2 » features, 2366 termes « racinisés » ont été extraits. La F-mesure est de 0.633, la précision de 0.691 et le rappel de 0.622. La micro F-mesure est de 0.703. Du fait d'une précision plus élevée, et comme détaillé à la partie 3.2.1, c'est ce dernier modèle que nous avons considéré comme étant le plus adapté pour attribuer un code topographique aux CRAP relatifs à un cancer. Ce code correspond à une des 41 catégories prédéfinies pour ce modèle, et n'aura donc pas la même granularité pour chaque CRAP.

Pour vérifier si cette perte d'information avait un intérêt, nous avons comparé les performances des 25 classes communes à ce modèle et au modèle A_t paramétré avec le même nombre de features et le même algorithme (pour rappel, les classes du modèle A_t sont toutes des codes CIMO3 à 4 digits). Dans l'ensemble de classes à granularité mixte, la classe « autres » regroupe les codes CIMO3 C53 et C54 ainsi que les groupes CIMO3 « site primaire inconnu », « tissu conjonctif, tissu sous-cutané et autres tissus mous », « rétro-péritoine et péritoine », « œil, cerveau et autres localisations du SNC », « os, articulations et cartilage articulaire » et « autres localisations et localisations mal définies » (pour un total de 117 CRAP). Avec le modèle C_t, la F-mesure de cette classe est de 0.455, le rappel de 0.435 et la précision de 0.476. Dans l'ensemble de codes CIMO3 à 4 digits, elle regroupe 132 codes pour un total de 921 CRAP. Avec le modèle A_t, sa F-mesure est de 0.683, son rappel de 0.815 et sa précision de 0.588. Pour les 24 autres classes communes aux deux ensembles, la F-mesure moyenne du modèle C_t est de 0.642 (écart-type = 0.184) alors que celle du modèle A_t est de 0.646 (0.178). Globalement, pour ces classes, le choix de l'ensemble de classes cibles n'a donc pas d'influence sur les performances du programme de classification. Le tableau 6 présente les codes CIMO3 à 4 digits pour lesquels on note un écart de performance de plus de 3% entre les modèles C_t et A_t.

Tableau 7 : Comparaison des performances des classes communes aux deux ensembles de classes cibles*

	Modèle C_t			Modèle A_t		
Codes topographiques	F-mesure	rappel	précision	F-mesure	rappel	précision
C649	0.92	0.92	0.92	**0.96**	**0.96**	**0.96**
C541	**0.892**	**0.925**	0.86	0.817	0.85	**0.895**
C447	**0.75**	0.706	**0.8**	0.729	0.691	0.77
C182	0.554	0.548	**0.561**	0.533	0.571	0.5
C180	0.533	0.457	0.64	**0.603**	**0.543**	**0.679**
C504	0.526	0.671	0.433	0.532	**0.768**	0.406
C509	0.49	0.417	**0.595**	0.472	0.417	0.543
C502	0.382	0.309	**0.5**	**0.416**	**0.382**	0.457

* sont présentées les classes avec un écart > 3%

L'analyse des erreurs du modèle de classification topographique ainsi que l'évaluation du modèle de classification morphologique n'a pas pu être réalisée dans le temps imparti.

5 Discussion

Les modèles de classification développés dans le cadre de ce travail sont performants et montrent qu'il est possible d'adapter les outils informatiques de fouille de texte et de machine learning aux comptes rendus médicaux écrits en langue française. En effet, nos résultats sont proches de ceux décrits dans notre état de l'art : pour la classification binaire, nous obtenons une F-mesure de 0.964, contre 0.94 pour Koopman et al et 0.99 pour Butt et al. Pour la classification des CRAP selon la topographie du cancer analysé, nous obtenons une micro F-mesure de 0.703 et une macro F-mesure de 0.633 pour 41 classes cibles. Du fait d'un ensemble de classes cibles différent en terme de nombre et de granularité, il est difficile de comparer ces résultats à ceux décrits dans la littérature. Malgré tout, cela apporte certaines indications : Kavuluru et al. [28] ont obtenu une micro F-mesure de 0.90 et une macro F-mesure de 0.72 pour 58 classes cibles (codes CIMO3 en 3 digits). Avec des CRAP rédigés en français, Jouhet et al. [8] ont obtenu une micro F-mesure de 0.72 pour 26 classes cibles (codes CIMO3 complets, en 4 digits).

Ces résultats sont donc encourageants. Néanmoins, plusieurs axes d'amélioration sont possibles pour rendre nos modèles plus performants et les utiliser en « vie réelle » au sein d'un registre de cancers.

5.1 Jeu de données

Nous notons plusieurs limites dans la construction de notre jeu de données.

Premièrement, nous avons sélectionné des CRAP avec une seule localisation cancéreuse et ne traitant ni des sites métastatiques ni des tumeurs à comportement incertain. Avec ces critères de sélection, Jouhet et al. ont estimé à 16% les CRAP non traités par les modèles de classification. Ceci empêche leur mise en production au sein d'un registre de cancers puisque ces CRAP sont nécessairement traités en routine. De plus, on note la coexistence de pathologies tumorales et non tumorales dans certains CRAP. Pour analyser ces CRAP décrivant plusieurs topographies à l'aide du machine learning, une des solutions serait d'utiliser une classification multi-label : au lieu d'attribuer une seule classe aux CRAP, ceux-ci se verraient attribuer plusieurs classes, rangées par ordre de vraisemblance.

Une autre limite pour la mise en production de nos modèles est que nous utilisons les données provenant d'un seul laboratoire. Les algorithmes de machine learning étant paramétrés grâce à ces données, il est probable que nos modèles se généralisent mal à de nouvelles données provenant d'autres laboratoires d'anatomo-pathologie.

Ensuite, notre analyse des erreurs a montré un problème de Gold standard pour la tâche de classification binaire. Dans le cas d'un apprentissage supervisé, établir un Gold standard sert à évaluer les performances de notre programme mais est surtout indispensable au paramétrage des algorithmes de machine learning. Dans notre cas, nous estimons à 67,7% les erreurs de classement de la tâche binaire liées à une mauvaise construction de notre Gold standard. Ces erreurs diminuent ainsi artificiellement les performances de nos modèles, sans que cela soit dû à une erreur de l'algorithme.

Bien que nous n'ayons pas fait d'analyse qualitative des erreurs du modèle de classification retenu pour le codage de la topographie, cette difficulté de construction du Gold standard intervient aussi pour cette tâche. Ainsi, McCowan et al. [26] ont montré qu'il y avait 10% de désaccord entre deux experts pour l'annotation manuelle de la topographie des cancers dans un corpus de 817 CRAP.

Enfin, l'ensemble de classes cibles à granularité mixte utilisé pour la tâche de classification de la topographie des cancers n'a pas obtenu les résultats escomptés. En effet, nous avions fait l'hypothèse que le rééquilibrage des données induit par cet ensemble permettrait d'améliorer les performances de classement des classes CIMO3 à 4 digits, par rapport à l'utilisation d'un ensemble de classes composé uniquement de codes CIMO3 4 digits.

Bien que les performances globales du modèle sont significativement meilleures (macro F-mesure de 0.642 contre 0.487 sur le jeu de validation), les performances obtenues pour les 24 classes communes (codes CIMO3 4 digits) à ces deux modèles sont presque identiques : macro F-mesure de 0.642 pour le modèle à granularité mixte (modèle C_t) contre 0.646 pour le modèle à classes composées uniquement de codes CIMO3 4 digits (modèle A_t). Cette méthode de ré-échantillonnage ne nous parait donc pas pertinente, dans le sens où elle complexifie inutilement l'annotation des CRAP et la construction de l'ensemble de classes cibles, rendant difficile la comparaison de nos résultats avec ceux de la littérature.

Un autre résultat remarquable est la faible F-mesure de la classe « autres » du modèle C_t : celle-ci est à 0.455, contre 0.683 pour le modèle A_t, alors qu'elle regroupe seulement huit groupes CIMO3 (soit 117 CRAP) contre 132 pour le modèle A_t (soit 921 CRAP). Il semble donc que le paramétrage des algorithmes de machine learning (au moins l'algorithme SVM, utilisé dans le modèle final) soit plus sensible au nombre de données d'entrainement par classe qu'à la qualité de construction de ces mêmes classes. La littérature confirme cette hypothèse : Koopman et al. [31] ont montré une corrélation entre le nombre de données d'entrainement par classe et les performances de classification (coefficient de corrélation de Pearson = 0.65). Plusieurs autres études ont remarqué ce résultat, avec notamment un impact négatif sur le rappel plutôt que la précision pour les classes avec moins de 100 données d'entrainement [8,25,26,28,31]. Pour améliorer les performances de ces classes, qui correspondent aux cancers dits « rares » (à faible prévalence), plusieurs solutions sont proposées par les auteurs : augmenter le seuil du nombre de données d'entrainement par classe, utiliser des règles de décision (combinées ou non avec des algorithmes de machine learning), ou encore procéder à un ré-échantillonnage des données comme nous l'avons fait pour la tâche de classification binaire. En effet, le sous-échantillonnage effectué sur la classe majoritaire « bénin » a permis d'améliorer le rappel de la classe minoritaire « cancer » (0.972 contre 0.957).

5.2 Représentation des données

Optimiser la façon dont les données sont présentées à un algorithme de machine learning est un champ de recherche en plein essor, notamment dans le domaine biomédical [38–42]. Les méthodes développées dans ce cadre sont regroupées sous le terme anglo-saxon « feature engineering », que l'on peut traduire par « ingénierie des caractéristiques/attributs des données » en français.

Nous avons vu à la partie 3.3 qu'il y avait trois grandes étapes pour représenter une donnée textuelle : normalisation du texte, indexation et réduction de dimensionnalité. Notre état de l'art a montré que les choix réalisés à chacune de ces étapes avaient une forte influence sur les performances des modèles de classification [18,27,28,31,33]. Par exemple, Butt et al. ont montré qu'une indexation en sac de mots avec des mots racinisés étaient supérieure à une indexation en sac de mots sans racinisation pour leur tâche de classification.

Pour l'étape de normalisation, nous avons évalué deux choix de normalisation : un corpus avec racinisation des mots (« stemming ») et un corpus sans stemming. En effet, les librairies informatiques permettant de réaliser ce stemming sont en général créées à partir d'un vocabulaire généraliste, peu adapté aux spécificités du domaine médical. De plus, en travaillant avec des textes en français, le choix des outils est d'autant plus réduit par rapport à des textes anglais. Nous avions alors fait l'hypothèse que ces outils, utilisés sans adaptation au vocabulaire médical et à la langue française, étaient susceptibles de diminuer les performances de nos modèles de classification. Au final, nos résultats montrent que le stemming n'a pas ou peu d'influence sur la F-mesure calculée sur le jeu de validation. Pour autant, en influant positivement sur d'autres étapes de transformation des données (diminution de la dimension des vecteurs), nous considérons que les modèles de classification utilisant le stemming sont supérieurs aux autres. Mais d'autres investigations sont nécessaires à ce niveau, notamment dans le but d'adapter les outils de stemming au vocabulaire médical français.

Après avoir normalisé le corpus, il faut indexer chacun de ces textes à l'aide d'un jeu de termes. Ces termes peuvent être de nature variée (mots, concepts, etc…) et sont extraits (on parle de « features extraction ») à l'aide de différentes méthodes de TAL. Ce sont les choix opérés à ce niveau que nous discuterons, le choix de la pondération de ces termes ayant peu d'influence sur les performances des modèles de classification [18].

Dans ce travail, nous avons choisi de réaliser une « tokénisation en mots » pour indexer notre corpus de CRAP. En effet, dans une tâche de classification de comptes rendus d'autopsie, Mujtaba et al. [33] ont montré la supériorité d'une tokénisation en mots par rapport à une tokénisation en N-grams[2].

[2] Notion provenant de la théorie de l'information : séquence de n unités lexicales (lettres, mots,…)

Pour autant, ce type de représentation a plusieurs limites :

1) de la même façon qu'avec les outils de stemming, les outils de tokénisation sont construits à l'aide d'un vocabulaire généraliste et initialement anglais : leurs performances diminuent lorsqu'ils doivent s'adapter aux spécificités de la langue française [43] et du vocabulaire médical [44].

2) des difficultés d'ordre linguistique, confirmées lors de l'analyse des erreurs de notre modèle de classification binaire. En effet, la syntaxe et le sens d'un texte sont perdus lors de ce processus.

La première limite sort du cadre de ce Master et impose de construire de nouveaux outils de tokénisation. Pour la deuxième, l'utilisation d'autres méthodes de TAL est nécessaire. Par exemple, l'analyse des erreurs du modèle de classification binaire a montré que six CRAP mal classés sur 20 utilisaient la négation pour exprimer une idée. Pour prendre en compte cette syntaxe particulière, Chapman et al. [45] ont développé un algorithme à base d'expressions régulières qui permet de détecter si un terme est exprimé dans une phrase affirmative ou négative. Cette algorithme fait référence dans le domaine et est régulièrement mis à jour. Une traduction française a été réalisée par Deléger et al. [46] mais n'est pas disponible en open-source. Si nous avions pu utiliser cet algorithme, il est probable que les performances de nos modèles aient été améliorées.

Une autre façon de limiter ces ambiguïtés linguistiques est d'utiliser des outils comme MetaMap [47], qui permettent de faire correspondre un terme à un concept médical d'une ontologie ou d'une terminologie (d'un dictionnaire de façon plus général). En utilisant un jeu de termes composés de mots et de concepts, la littérature a montré que ces features amélioraient les performances des modèles de classification [18,28,31]. Pour autant, les résultats dépendent de la qualité de l'extraction et du dictionnaire utilisé : Kavuluru et al. ont montré que l'ajout de concepts UMLS améliorait de seulement 1% la F-mesure de leur modèle par rapport à une représentation en sac de mots. Aussi, Koopman et al. suggèrent que les concepts de la terminologie CIMO3 sont plus discriminants que ceux de l'ontologie SNOMED, mais que leur faible fréquence au sein d'un corpus de comptes rendus médicaux conduit à une forte variance des performances des modèles. Enfin, il n'existe pas ou peu de ressources équivalentes en langue française (seulement 10% de la SNOMED est traduite en français canadien, et il n'existe pas de traduction française du NCI-thesaurus à notre connaissance). Ainsi, il nous semble nécessaire d'adapter ces ressources à la langue française pour pouvoir les utiliser de façon optimale au sein d'un modèle de classification.

Enfin, une autre représentation intéressante pour prendre en compte la sémantique d'un texte est d'utiliser ce qu'on appelle le « plongement lexical », « word embedding » en anglais. Sans rentrer dans les détails d'implémentation, cette technique utilise des algorithmes d'apprentissage statistique pour modéliser les mots d'un corpus dans un espace vectoriel dans lequel leur co-occurrence au sein d'un texte les rapproche. Ainsi, les documents du corpus sont décrits par des features contenant une information sur le contexte d'apparition des mots. Bien que cette approche soit très récente (Mikolov et al, 2013 [48]), son application dans des tâches de classification de comptes rendus médicaux a montré une bonne efficacité, en combinaison avec les représentations classiques comme le sac de mots [49–52].

Une fois le corpus indexé, il est souhaitable de réduire la dimension des vecteurs créés (c'est-à-dire de diminuer le nombre de termes du jeu de termes), dans le but de limiter la complexité des modèles de classification et d'optimiser leurs performances. Nous avons vu à la partie 3.3.3 qu'il existait deux principales approches pour cela : sélection de termes et extraction de termes. En pratique, aux vues de notre état de l'art, l'extraction de termes est plutôt utilisée pour ajouter des features discriminantes au modèle que pour réduire la taille du jeu de termes. Quant à la sélection de termes, nos résultats montrent qu'il n'existe pas de « meilleure pratique » dans le choix du nombre de termes à conserver ; ce choix dépendant de l'algorithme de classification utilisé. Par exemple, pour la tâche de classification topographique, l'algorithme SVM fournit de meilleurs résultats lorsqu'on utilise l'intégralité du jeu de termes (soit sans réduction de dimensionnalité) alors que l'algorithme RF performe mieux avec seulement 200 termes. Par rapport aux méthodes de sélection de termes, Mujtaba et al. [33] ont montré que le test du Chi^2 et le calcul du gain d'information ("information gain" en anglais) étaient supérieurs au calcul du coefficient de Pearson pour une tâche de classification de comptes rendus d'autopsie. D'autres investigations sont nécessaires pour confirmer ces résultats et analyser les avantages et inconvénients de chaque méthode.

5.3 Intelligence artificielle : un domaine en constante évolution

Pour remplir notre objectif de travail, qui était d'automatiser le processus de traitement manuel des CRAP au sein d'un registre de cancers, nous avons utilisé le machine learning et une tâche de classification supervisée. Or, il existe plusieurs autres méthodes d'intelligence artificielle dite statistique (par opposition à l'intelligence artificielle symbolique ou déterministe) susceptibles de réaliser ce même objectif.

Chacun de nos modèles utilisaient un seul algorithme de classification pour prédire la classe d'un CRAP. Grâce à des méthodes dites ensemblistes, il est possible de combiner les prédictions de différents algorithmes et ainsi diminuer le biais et la variance des résultats. Pour un CRAP donné, un système de vote plus ou moins évolué (vote majoritaire, pondérations des prédictions, etc.) permettra alors de sélectionner sa classe finale. Nous distinguons ici ces méthodes des méthodes de « bagging » et de « boosting », dont le principe est de combiner des algorithmes de même type [19], et dont l'algorithme RF (« random forest ») est un exemple (combinaison de plusieurs arbres de décision). Par exemple, Nguyen et al. [24] ont montré dans une tâche de classification multi-classe que décomposer le problème en sous-tâches de classification binaire, à l'aide d'une hiérarchie de classifieurs SVM, était supérieur à l'utilisation d'un seul SVM multi-classe.

Une autre limite à l'utilisation de techniques d'apprentissage supervisé est la disponibilité d'un jeu de données labellisé. Dans le domaine médical, du fait d'une nécessité de confidentialité des données, un tel jeu est rarement disponible en open-source. Il est alors nécessaire de le construire, ce qui peut être couteux en temps : McCowan et al. [26] ont estimé à 40 heures le temps d'annotation de comptes rendus de 179 patients, soit 13 minutes par patient. Pour réduire le temps de construction d'un jeu de données apte à entrainer un modèle de machine learning, il est possible d'utiliser l' « active learning ».

L’active learning est une méthode d’apprentissage semi-supervisé qui nécessite une intervention humaine dans le but d’utiliser des données non labélisées et ainsi améliorer les performances d’un modèle de machine learning. Dans le domaine médical, cette méthode a montré de bonnes performances, supérieures à celles obtenues avec un apprentissage supervisé dit « passif » [53–55].

Enfin, un champ de recherche en plein essor est l’apprentissage profond (« deep learning » en anglais), une branche du machine learning. Alors qu’un modèle de machine learning « apprend » à partir d’une seule (en général) représentation des données, l’avantage du deep learning est d’intégrer dans un même modèle plusieurs représentations (parfois des centaines) dans son processus d’apprentissage [20]. Ces représentations sont organisées en couches successives, c’est pourquoi on parle de « réseau de neurones » pour faire référence à un modèle de deep learning, par analogie avec les couches neuronales du cortex cérébral humain. Dans le domaine de la classification de textes médicaux, ces modèles ont montré de bonnes performances, sensiblement supérieures à celles de modèles de machine learning classique [56–61]. Le principal inconvénient d’un réseau de neurones est l’importance (en terme de performance) des ressources informatiques nécessaires à l’entrainement d’un tel modèle. Avant d’utiliser cette approche à la place d’une approche par machine learning classique pour un problème de classification de textes, d’autres investigations nous semblent nécessaires notamment dans l’évaluation de la balance bénéfice/coût des modèles d’apprentissage profond.

Conclusion

En conclusion, nos résultats montrent qu'il est possible d'automatiser le traitement des CRAP en texte libre rédigés en langue française, dans le but de recueillir les informations nécessaires au signalement des nouveaux cas de cancer, au sein d'un registre des cancers.

Pour autant, avant d'intégrer les modèles de classification développés dans ce travail au sein d'un système de fouille de texte, de façon similaire aux systèmes décrits à la partie 1.4.1, des adaptations sont nécessaires. En effet, la confrontation de nos résultats aux données de la littérature suggère plusieurs axes d'amélioration, dont voici ceux qui nous semblent être les principaux :

1) Optimiser la construction de notre jeu de données, notamment au niveau du Gold Standard de la tâche de classification binaire. En effet, si les annotations des ARC du registre de la Gironde sont adaptées à leur protocole de traitement manuel des CRAP, elles ne le sont pas pour distinguer de façon correcte les cancers des pathologies tumorales bénignes ; dans le sens où dans ce dernier cas, les critères de sélection du registre ne doivent jamais intervenir pour cette distinction. Pour une partie du jeu de données au moins, cela passera vraisemblablement par un temps d'annotation manuelle et il sera alors utile d'optimiser le nombre de données à annoter pour permettre à la fois un bon paramétrage et une bonne évaluation des modèles de machine learning.
2) L'utilisation de méthodes de TAL permettant de lever les ambiguïtés sémantiques relevées par notre analyse des erreurs. Pour cela, deux ressources nous semblent indispensables : l'algorithme NegEx traduit en langue française et une ontologie permettant l'extraction de concepts médicaux. Pour cette dernière ressource, deux voies de développement nous semblent préférables : soit la traduction en français d'ontologies existantes (telles la SNOMED, l'UMLS ou le NCI-thesaurus), soit la création d'une nouvelle ontologie en français et plus ou moins spécifique au domaine de la cancérologie.
3) Le développement et l'évaluation de modèles utilisant d'autres branches de l'intelligence artificielle, notamment le deep learning.

Enfin, pour la mise en production de ce système, il sera nécessaire de traiter l'ensemble des CRAP recueillis par un registre de cancers, ce qui reste un défi important aux vues des données de la littérature. Aussi, il semble illusoire de pouvoir automatiser à 100% le traitement manuel décrit dans ce travail : des investigations seraient utiles pour déterminer un objectif d'automatisation acceptable, et ainsi permettre d'évaluer les ressources minimales nécessaires et le coût d'un tel projet.

Bibliographie

[1] Spécificités et perspectives du programme de travail partenarial 2011-2013, relatif à la surveillance des cancers à partir des registres n.d. http://beh.santepubliquefrance.fr/beh/2013/43-44-45/2013_43-44-45_1.html (accessed July 29, 2019).

[2] N.d. https://www.santepubliquefrance.fr/docs/les-registres-des-cancers-en-france (accessed July 11, 2019).

[3] Maurisset PRSDMSAMDFCPBHBCGIBAJJCS. Registre des cancers en Aquitaine n.d. http://etudes.isped.u-bordeaux2.fr/REGISTRES-CANCERS-AQUITAINE/Rg/R_Accueil.aspx (accessed February 13, 2019).

[4] RJ B, L S, HH S, E D. Automated Data Collection in Cancer Registration. 1998.

[5] Jensen OM, International Agency for Research on Cancer, World Health Organization, International Association of Cancer Registries, editors. Cancer registration: principles and methods. Lyon, France : New York: International Agency for Research on Cancer ; Distributed in the USA by Oxford University Press; 1991.

[6] Safran C, Bloomrosen M, Hammond WE, Labkoff S, Markel-Fox S, Tang PC, et al. Toward a National Framework for the Secondary Use of Health Data: An American Medical Informatics Association White Paper. J Am Med Inform Assoc 2007;14:1–9. doi:10.1197/jamia.M2273.

[7] Fritz AG. Classification internationale des maladies pour l'oncologie: CIM-O. Genève: Organisation mondiale de la santé; 2008.

[8] Jouhet V, Defossez G, Burgun A, Beux PL, Levillain P, Ingrand P, et al. Automated Classification of Free-text Pathology Reports for Registration of Incident Cases of Cancer. Methods Inf Med 2012;51:242–51. doi:10.3414/ME11-01-0005.

[9] adicap_version5_4_1_2009_0.pdf n.d.

[10] Tellier I. Introduction `a la fouille de textes universit´e de Paris 3 - Sorbonne Nouvelle n.d.:98.

[11] Hanauer DA, Miela G, Chinnaiyan AM, Chang AE, Blayney DW. The registry case finding engine: an automated tool to identify cancer cases from unstructured, free-text pathology reports and clinical notes. J Am Coll Surg 2007;205:690–7. doi:10.1016/j.jamcollsurg.2007.05.014.

[12] Crowley RS, Castine M, Mitchell K, Chavan G, McSherry T, Feldman M. caTIES: a grid based system for coding and retrieval of surgical pathology reports and tissue specimens in support of translational research. J Am Med Inform Assoc JAMIA 2010;17:253–64. doi:10.1136/jamia.2009.002295.

[13] Coden A, Savova G, Sominsky I, Tanenblatt M, Masanz J, Schuler K, et al. Automatically extracting cancer disease characteristics from pathology reports into a Disease Knowledge Representation Model. J Biomed Inform 2009;42:937–49. doi:10.1016/j.jbi.2008.12.005.

[14] Nguyen A, Moore J, Lawley M, Hansen D, Colquist S. Automatic extraction of cancer characteristics from free-text pathology reports for cancer notifications. Stud Health Technol Inform 2011;168:117–24.

[15] Osborne JD, Wyatt M, Westfall AO, Willig J, Bethard S, Gordon G. Efficient identification of nationally mandated reportable cancer cases using natural language processing and machine learning. J Am Med Inform Assoc JAMIA 2016;23:1077–84. doi:10.1093/jamia/ocw006.

[16] D'Avolio LW, Nguyen TM, Farwell WR, Chen Y, Fitzmeyer F, Harris OM, et al. Evaluation of a generalizable approach to clinical information retrieval using the automated retrieval console (ARC). J Am Med Inform Assoc 2010;17:375–82. doi:10.1136/jamia.2009.001412.

[17] Solti I, Cooke CR, Xia F, Wurfel MM. Automated Classification of Radiology Reports for Acute Lung Injury: Comparison of Keyword and Machine Learning Based Natural Language Processing Approaches. Proc IEEE Int Conf Bioinforma Biomed 2009;2009:314–9. doi:10.1109/BIBMW.2009.5332081.
[18] Butt L, Zuccon G, Nguyen A, Bergheim A, Grayson N. Classification of cancer-related death certificates using machine learning. Australas Med J 2013;6:292–9. doi:10.4066/AMJ.2013.1654.
[19] Sebastiani F. Machine Learning in Automated Text Categorization. ArXiv:Cs/0110053 2001.
[20] Chollet F. Deep Learning with Python (2).pdf n.d. https://www.xodo.com/app/#/pdf (accessed April 16, 2019).
[21] Freitag D. Information Extraction from HTML: Application of a General Machine Learning Approach n.d.:7.
[22] Téllez-Valero A, Montes-y-Gómez M, Villaseñor-Pineda L. A Machine Learning Approach to Information Extraction. In: Gelbukh A, editor. Comput. Linguist. Intell. Text Process., Springer Berlin Heidelberg; 2005, p. 539–47.
[23] Freitag D. Machine Learning for Information Extraction in Informal Domains. Mach Learn 2000;39:169–202. doi:10.1023/A:1007601113994.
[24] Nguyen A, Moore D, McCowan I, Courage M-J. Multi-class classification of cancer stages from free-text histology reports using support vector machines. Conf Proc Annu Int Conf IEEE Eng Med Biol Soc IEEE Eng Med Biol Soc Annu Conf 2007;2007:5140–3. doi:10.1109/IEMBS.2007.4353497.
[25] D'Avolio LW, Litwin MS, Rogers SO, Bui AAT. Automatic identification and classification of surgical margin status from pathology reports following prostate cancer surgery. AMIA Annu Symp Proc AMIA Symp 2007:160–4.
[26] McCowan IA, Moore DC, Nguyen AN, Bowman RV, Clarke BE, Duhig EE, et al. Collection of cancer stage data by classifying free-text medical reports. J Am Med Inform Assoc JAMIA 2007;14:736–45. doi:10.1197/jamia.M2130.
[27] Li Y, Martinez D. Information Extraction of Multiple Categories from Pathology Reports. Proc. Australas. Lang. Technol. Assoc. Workshop 2010, Melbourne, Australia: 2010, p. 41–48.
[28] Kavuluru R, Hands I, Durbin EB, Witt L. Automatic Extraction of ICD-O-3 Primary Sites from Cancer Pathology Reports. AMIA Summits Transl Sci Proc 2013;2013:112–6.
[29] Chen W, Huang Y, Boyle B, Lin S. The utility of including pathology reports in improving the computational identification of patients. J Pathol Inform 2016;7:46. doi:10.4103/2153-3539.194838.
[30] Oleynik M, Patrão DFC, Finger M. Automated Classification of Semi-Structured Pathology Reports into ICD-O Using SVM in Portuguese. Stud Health Technol Inform 2017;235:256–60.
[31] Koopman B, Zuccon G, Nguyen A, Bergheim A, Grayson N. Automatic ICD-10 classification of cancers from free-text death certificates. Int J Med Inf 2015;84:956–65. doi:10.1016/j.ijmedinf.2015.08.004.
[32] Löpprich M, Krauss F, Ganzinger M, Senghas K, Riezler S, Knaup P. Automated Classification of Selected Data Elements from Free-text Diagnostic Reports for Clinical Research. Methods Inf Med 2016;55:373–80. doi:10.3414/ME15-02-0019.
[33] Mujtaba G, Shuib L, Raj RG, Rajandram R, Shaikh K. Prediction of cause of death from forensic autopsy reports using text classification techniques: A comparative study. J Forensic Leg Med 2018;57:41–50. doi:10.1016/j.jflm.2017.07.001.
[34] Learning from imbalanced data: Open challenges and future directions. ResearchGate n.d. https://www.researchgate.net/publication/301596547_Learning_from_imbalanced_data_Open_challenges_and_future_directions (accessed May 28, 2019).

[35] Tibshirani S, Friedman H. The elements of statistical learning n.d.:764.
[36] Arlot S, Celisse A. A survey of cross-validation procedures for model selection. Stat Surv 2010;4:40–79. doi:10.1214/09-SS054.
[37] Fuhr N, Darmstadt T, Buckley C. A probabilistic learning approach for document indexing. ACM Trans Inf Syst 1991;9:223–248.
[38] Yetisgen-Yildiz M, Gunn ML, Xia F, Payne TH. Automatic identification of critical follow-up recommendation sentences in radiology reports. AMIA Annu Symp Proc AMIA Symp 2011;2011:1593–602.
[39] Bleik S, Mishra M, Huan J, Song M. Text categorization of biomedical data sets using graph kernels and a controlled vocabulary. IEEE/ACM Trans Comput Biol Bioinform 2013;10:1211–7. doi:10.1109/TCBB.2013.16.
[40] Kavuluru R, Rios A, Lu Y. An empirical evaluation of supervised learning approaches in assigning diagnosis codes to electronic medical records. Artif Intell Med 2015;65:155–66. doi:10.1016/j.artmed.2015.04.007.
[41] Mujtaba G, Shuib L, Raj RG, Rajandram R, Shaikh K, Al-Garadi MA. Automatic ICD-10 multi-class classification of cause of death from plaintext autopsy reports through expert-driven feature selection. PloS One 2017;12:e0170242. doi:10.1371/journal.pone.0170242.
[42] Mujtaba G, Shuib L, Raj RG, Rajandram R, Shaikh K, Al-Garadi MA. Classification of forensic autopsy reports through conceptual graph-based document representation model. J Biomed Inform 2018;82:88–105. doi:10.1016/j.jbi.2018.04.013.
[43] Bernhard D, Todirascu A, MARTIN F, Erhart P, Steible L, Huck D, et al. Problèmes de tokénisation pour deux langues régionales de France, l'alsacien et le picard. DiLiTAL 2017, Orléans, France: 2017, p. 14–23.
[44] Rabary C, Lavergne T, Névéol A. Etiquetage morpho-syntaxique en domaine de spécialité: le domaine médical 2015:7.
[45] Chapman WW, Bridewell W, Hanbury P, Cooper GF, Buchanan BG. A Simple Algorithm for Identifying Negated Findings and Diseases in Discharge Summaries. J Biomed Inform 2001;34:301–10. doi:10.1006/jbin.2001.1029.
[46] Deléger L, Grouin C. Detecting negation of medical problems in French clinical notes. Proc. 2nd ACM SIGHIT Symp. Int. Health Inform. - IHI 12, Miami, Florida, USA: ACM Press; 2012, p. 697. doi:10.1145/2110363.2110443.
[47] Aronson AR. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. Proc AMIA Symp 2001:17–21.
[48] Mikolov T, Chen K, Corrado G, Dean J. Efficient Estimation of Word Representations in Vector Space. ArXiv13013781 Cs 2013.
[49] Lilleberg J, Zhu Y, Zhang Y. Support vector machines and Word2vec for text classification with semantic features. 2015 IEEE 14th Int. Conf. Cogn. Inform. Cogn. Comput. ICCICC, Beijing, China: IEEE; 2015, p. 136–40. doi:10.1109/ICCI-CC.2015.7259377.
[50] Enríquez F, Troyano JA, López-Solaz T. An approach to the use of word embeddings in an opinion classification task. Expert Syst Appl 2016;66:1–6. doi:10.1016/j.eswa.2016.09.005.
[51] Yang B, Dai G, Yang Y, Tang D, Li Q, Lin D, et al. Automatic Text Classification for Label Imputation of Medical Diagnosis Notes Based on Random Forest. In: Siuly S, Lee I, Huang Z, Zhou R, Wang H, Xiang W, editors. Health Inf. Sci., Springer International Publishing; 2018, p. 87–97.
[52] Wang Y, Liu S, Afzal N, Rastegar-Mojarad M, Wang L, Shen F, et al. A comparison of word embeddings for the biomedical natural language processing. J Biomed Inform 2018;87:12–20. doi:10.1016/j.jbi.2018.09.008.

[53] Garla V, Taylor C, Brandt C. Semi-supervised clinical text classification with Laplacian SVMs: an application to cancer case management. J Biomed Inform 2013;46:869–75. doi:10.1016/j.jbi.2013.06.014.
[54] Figueroa RL, Zeng-Treitler Q, Ngo LH, Goryachev S, Wiechmann EP. Active learning for clinical text classification: is it better than random sampling? J Am Med Inform Assoc JAMIA 2012;19:809–16. doi:10.1136/amiajnl-2011-000648.
[55] Chen Y, Mani S, Xu H. Applying active learning to assertion classification of concepts in clinical text. J Biomed Inform 2012;45:265–72. doi:10.1016/j.jbi.2011.11.003.
[56] Hughes M, Li I, Kotoulas S, Suzumura T. Medical Text Classification Using Convolutional Neural Networks. Stud Health Technol Inform 2017;235:246–50.
[57] Karimi S, Dai X, Hassanzadeh H, Nguyen A. Automatic Diagnosis Coding of Radiology Reports: A Comparison of Deep Learning and Conventional Classification Methods. BioNLP 2017, Vancouver, Canada,: Association for Computational Linguistics; 2017, p. 328–32. doi:10.18653/v1/W17-2342.
[58] Yoon H-J, Ramanathan A, Tourassi G. Multi-task Deep Neural Networks for Automated Extraction of Primary Site and Laterality Information from Cancer Pathology Reports. In: Angelov P, Manolopoulos Y, Iliadis L, Roy A, Vellasco M, editors. Adv. Big Data, Springer International Publishing; 2017, p. 195–204.
[59] Gao S, Young MT, Qiu JX, Yoon H-J, Christian JB, Fearn PA, et al. Hierarchical attention networks for information extraction from cancer pathology reports. J Am Med Inform Assoc JAMIA 2017. doi:10.1093/jamia/ocx131.
[60] Qiu JX, Yoon H-J, Fearn PA, Tourassi GD. Deep Learning for Automated Extraction of Primary Sites From Cancer Pathology Reports. IEEE J Biomed Health Inform 2018;22:244–51. doi:10.1109/JBHI.2017.2700722.
[61] Banerjee I, Ling Y, Chen MC, Hasan SA, Langlotz CP, Moradzadeh N, et al. Comparative effectiveness of convolutional neural network (CNN) and recurrent neural network (RNN) architectures for radiology text report classification. Artif Intell Med 2018. doi:10.1016/j.artmed.2018.11.004.

Annexes

Annexe 1 : Performances des modèles de classification binaire avec crap complet, sur le jeu de validation

Modèles avec CRAP complets				
	Meilleure F-mesure	Rappel	Précision	Nombre de features
Modèle A				
RF	**0.955**	**0.972**	0.938	« significatif *2 »
LR	0.953	0.963	0.944	« all »
SVM (linéaire)	**0.959**	0.971	0.947	« all »
SVM (rbf)	0.94	0.952	0.929	« significatif »
SVM (poly)	0.571	0.403	0.982	10
SVM (sigmoid)	0.934	0.945	0.924	« significatif »
DT	0.921	0.954	0.891	« significatif »
NBm	0.874	0.953	0.807	« all »
Modèle B				
RF	**0.955**	0.97	0.941	« all »
LR	0.954	0.964	0.944	« all »
SVM (linéaire)	**0.96**	**0.972**	0.949	« all »
SVM (rbf)	0.941	0.951	0.931	« significatif »
SVM (poly)	0.577	0.408	0.982	10
SVM (sigmoid)	0.933	0.94	0.926	« significatif »
DT	0.924	0.956	0.894	« significatif »
NBm	0.881	0.955	0.818	« all »
Modèle C				
RF	**0.963**	0.954	0.972	« significatif »
LR	0.954	0.944	0.965	« all »
SVM (linéaire)	**0.961**	**0.957**	0.965	« significatif *2 »
SVM (rbf)	0.942	0.924	0.962	« significatif »
SVM (poly)	0.667	0.507	0.976	10
SVM (sigmoid)	0.933	0.902	0.965	500
DT	0.94	0.938	0.941	« significatif »
NBm	0.913	0.933	0.894	« all »
Modèle D				
RF	**0.963**	0.954	0.972	« significatif »
LR	0.954	0.942	0.967	« significatif *2 »
SVM (linéaire)	**0.962**	**0.957**	0.967	« significatif *2 »
SVM (rbf)	0.941	0.918	0.965	« significatif »
SVM (poly)	0.636	0.472	0.975	10
SVM (sigmoid)	0.927	0.887	0.969	500
DT	0.942	0.942	0.942	« significatif*2 »
NBm	0.917	0.938	0.897	« all »

Annexe 2 : Performances des modèles de classification binaire avec conclusion seule, sur le jeu de validation

Modèles avec conclusion seule				
	Meilleure F-mesure	Rappel	Précision	Nombre de features
Modèle E				
RF	0.953	0.965	0.941	« significatif*2 »
LR	0.949	0.956	0.941	« significatif*2 »
SVM (linéaire)	0.952	0.96	0.945	« all »
SVM (rbf)	0.943	0.939	0.946	« significatif »
SVM (poly)	0.608	0.442	0.977	10
SVM (sigmoid)	0.938	0.929	0.947	« significatif »
DT	0.93	0.952	0.91	« significatif »
NBm	0.914	0.941	0.889	« all »
Modèle F				
RF	0.953	0.966	0.939	« all »
LR	0.95	0.956	0.944	« all »
SVM (linéaire)	0.954	0.96	0.948	« all »
SVM (rbf)	0.944	0.937	0.95	« significatif »
SVM (poly)	0.601	0.434	0.977	10
SVM (sigmoid)	0.939	0.926	0.952	« significatif »
DT	0.932	0.953	0.912	« significatif »
NBm	0.918	0.946	0.892	« all »
Modèle G				
RF	0.959	0.951	0.967	« all »
LR	0.95	0.938	0.963	« all »
SVM (linéaire)	0.956	0.949	0.963	« all »
SVM (rbf)	0.941	0.921	0.962	500
SVM (poly)	0.687	0.531	0.976	10
SVM (sigmoid)	0.936	0.91	0.965	« significatif »
DT	0.942	0.934	0.949	« significatif »
NBm	0.933	0.926	0.939	« significatif*2 »
Modèle H				
RF	0.958	0.951	0.965	« significatif*2 »
LR	0.951	0.939	0.964	« all »
SVM (linéaire)	0.956	0.948	0.964	« significatif*2 »
SVM (rbf)	0.939	0.916	0.962	« significatif »
SVM (poly)	0.661	0.501	0.974	10
SVM (sigmoid)	0.928	0.897	0.961	500
DT	0.946	0.936	0.955	« significatif »
NBm	0.933	0.925	0.941	« all »

Annexe 3 : Performances des modèles de classification topographique

Résultats du jeu de validation				
	Meilleure F-mesure	Rappel	Précision	Nombre de features
Modèles avec CRAP complet				
Modèle A_t				
SVM (linéaire)	**0.487 (0.012)**	0.483 (0.014)	**0.537 (0.027)**	« all »
SVM (rbf)	0.304 (0.009)	0.311 (0.01)	0.351 (0.016)	200
LR	0.446 (0.008)	0.439 (0.01)	0.488 (0.011)	« all »
RF	0.431 (0.01)	0.427 (0.011)	0.463 (0.021)	200
Modèle B_t				
SVM (linéaire)	**0.479 (0.012)**	0.476 (0.013)	**0.52 (0.022)**	« all »
SVM (rbf)	0.281 (0.008)	0.286 (0.009)	0.347 (0.012)	200
LR	0.44 (0.011)	0.43 (0.012)	0.489 (0.009)	« all »
RF	0.436 (0.013)	0.43 (0.015)	0.469 (0.018)	200
Modèle C_t				
SVM (linéaire)	**0.642 (0.013)**	0.634 (0.011)	**0.69 (0.018)**	« all »
SVM (rbf)	0.394 (0.004)	0.389 (0.005)	0.526 (0.022)	200
LR	0.607 (0.01)	0.596 (0.013)	0.68 (0.019)	« all »
RF	0.551 (0.01)	0.542 (0.011)	0.595 (0.014)	200
Modèle D_t				
SVM (linéaire)	**0.632 (0.013)**	0.624 (0.012)	**0.679 (0.016)**	« all »
SVM (rbf)	0.365 (0.012)	0.357 (0.01)	0.499 (0.03)	200
LR	0.598 (0.012)	0.586 (0.013)	0.664 (0.021)	« all »
RF	0.546 (0.005)	0.538 (0.004)	0.601 (0.013)	500
Modèles avec conclusion seule				
Modèle E_t				
SVM (linéaire)	**0.462 (0.014)**	0.452 (0.014)	**0.514 (0.013)**	« all »
SVM (rbf)	0.268 (0.006)	0.275 (0.009)	0.333 (0.012)	200
LR	0.424 (0.009)	0.41 (0.01)	0.494 (0.015)	« all »
RF	0.413 (0.005)	0.402 (0.005)	0.463 (0.015)	500
Modèle F_t				
SVM (linéaire)	**0.465 (0.018)**	0.454 (0.017)	**0.52 (0.027)**	« all »
SVM (rbf)	0.259 (0.008)	0.264 (0.009)	0.329 (0.012)	200
LR	0.42 (0.011)	0.403 (0.012)	0.495 (0.019)	« all »
RF	0.411 (0.01)	0.402 (0.008)	0.452 (0.013)	500
Modèle G_t				
SVM (linéaire)	**0.581 (0.02)**	0.57 (0.018)	**0.648 (0.031)**	« all »
SVM (rbf)	0.329 (0.01)	0.333 (0.012)	0.481 (0.022)	200
LR	0.56 (0.018)	0.545 (0.017)	0.632 (0.021)	« all »
RF	0.512 (0.011)	0.503 (0.008)	0.566 (0.023)	« significatif »
Modèle H_t				
SVM (linéaire)	**0.579 (0.015)**	0.568 (0.014)	**0.646 (0.021)**	« all »
SVM (rbf)	0.309 (0.007)	0.314 (0.01)	0.452 (0.027)	200
LR	0.554 (0.018)	0.54 (0.017)	0.628 (0.026)	« all »
RF	0.505 (0.003)	0.496 (0.001)	0.548 (0.006)	500

Résumé

Les CRAP constituent une des principales sources d'information d'un registre de cancers. Ce sont des données textuelles non structurées dont le contenu peut être résumé par l'ajout d'un code décrivant un ou plusieurs concepts médicaux. Lorsque ces données ne sont pas codées, elles nécessitent un traitement manuel pour extraire l'information voulue. Notre objectif est d'automatiser ce traitement des CRAP non codés au sein des registre de cancers français, grâce à des modèles de classification automatique de texte par machine learning.

A partir de 84 745 CRAP extraits de la base de données du registre général des cancers de la Gironde, nous avons construit trois modèles de classification et évalué différents choix au niveau des données et de leur représentation, ainsi qu'au niveau des algorithmes de classification utilisés. Les mesures d'évaluation étaient le rappel, la précision et la F-mesure.

Pour le modèle de classification binaire, permettant de distinguer les CRAP relatifs à un cancer, nous obtenons une F-mesure de 0.964. Pour le codage de la topographie des cancers, le meilleur modèle obtient une micro F-mesure de 0.703 et une macro F-mesure de 0.633.

Ces résultats sont performants et suggèrent la possibilité d'utiliser ces modèles pour automatiser le traitement des CRAP en texte libre écrits en français. Néanmoins, des améliorations sont nécessaires pour permettre leur mise en production au sein d'un registre de cancers.

Mots clés: classification automatique, texte libre, comptes rendus anatomo-pathologiques, apprentissage automatique, informatique médicale

Abstract

Pathology reports are ones of the main information sources concerning a cancer register. They are unstructured textual data, their content can be sum up by adding a describing code of one or several medical concepts. When these data are not coded, they need a manual processing in order to extract the wanted information. Our aim is to automate this not coded pathology reports' processing among french cancer registers through automatic textual categorization models by learning machine.

Based on 84 745 pathology reports' extracts from the Gironde general cancer register' data, we created three categorization models and evaluated various choices according to: their data, representation, and algorithms of categorization used. The evaluation measures were: the recall, the precision and the F-measure.

For the binary classification model, allowing to distinguish the pathology reports relating to a cancer, we get a 0.964 F-measure. For the cancer topography transcription, the best model gets a 0.703 micro F-measure and a 0.633 macro F-measure.

These results are relevant and suggest the possibility of using those models to automate in free texts written in French, the pathology reports' processing. Nevertheless, improvements are necessary to let their production start within a cancer register.

Keywords: automated categorization, free text, pathology reports, machine learning, medical informatics