UQAC
Université du Québec
à Chicoutimi

APPRENTISSAGE NON SUPERVISÉ DES ACTIVITÉS DE LA VIE QUOTIDIENNE À PARTIR DE FOUILLE DE DONNÉES SUR INTERNET

PAR CHARLES COUSYN

THÈSE PRÉSENTÉE À L'UNIVERSITÉ DU QUÉBEC À CHICOUTIMI DANS LE CADRE D'UN PROGRAMME EN EXTENSION DE L'UNIVERSITÉ DU QUÉBEC EN OUTAOUAIS EN VUE DE L'OBTENTION DU GRADE DE PHILOSOPHIÆ DOCTOR (PH.D.) EN INFORMATIQUE

QUÉBEC, CANADA

RÉSUMÉ

Au Québec, le vieillissement de la population est un problème majeur qui impacte notamment les dépenses dans le milieu de la santé. L'un des principaux défis que cela implique est le maintien à domicile prolongé des aînés et des personnes semi-autonomes. Afin de permettre ce maintien, il est alors nécessaire d'adapter l'environnement de vie de ces personnes en prenant en considération leurs capacités et leurs limites. Afin de trouver des solutions non intrusives à cette adaptation, une des solutions envisagées est l'utilisation de la technologie par la création d'un habitat intelligent capable de promouvoir l'autonomie et le bien-être des résidents. Cependant, la création d'un tel habitat présente son lot de défis scientifiques. L'un des principaux consiste en l'exploitation des données brutes provenant de capteurs hétérogènes distribués dans l'habitat intelligent, afin de parvenir à reconnaître les Activités de la Vie Quotidienne (AVQ) du résident. Les chercheurs se sont penchés depuis quelques décennies sur ce domaine, qu'on appelle plus communément la reconnaissance d'activités.

L'une des manières de concevoir une approche de reconnaissance d'activités est d'abord la simple d'observation d'individus en train d'effectuer des activités et la création d'un ensemble de règles adaptées aux observations. Néanmoins, cette approche est manuelle et nécessite la réalisation d'expérimentations, c'est-à-dire une observation active par un humain du comportement du résident pour créer des règles adaptées. Une autre solution est l'utilisation d'algorithmes d'apprentissage sur des données tirées d'expérimentations. Cette approche a l'avantage d'automatiser la phase d'apprentissage, mais pas la phase expérimentale ; il est toujours nécessaire de disposer d'individus réalisant des activités dans un habitat intelligent. Afin de nous pencher sur ce problème d'automatisation, nous nous sommes tournés vers le web. Le web représentant une formidable source d'informations, nous pensons qu'il est possible de se passer des données réelles en utilisant les ressources disponibles sur celui-ci. Le contenu disponible étant très abondant et mis à jour régulièrement, il sera alors possible d'extraire un très grand nombre de modèles d'activités, c'est-à-dire une base de connaissances sur un très grand nombre d'activités comme les objets/substances impliqués dans leur réalisation ainsi que des informations temporelles (chronologie des étapes constituantes) et spatiales (lieu de réalisation).

Cette manière de concevoir la reconnaissance d'activités n'ayant été qu'assez peu entrevue dans la littérature scientifique, le défi est de savoir si l'utilisation automatisée du web s'avère être une approche pertinente ou non. La problématique du projet de recherche est alors la suivante : "Dans quelle mesure est-il possible de réaliser automatiquement un apprentissage non supervisé des activités de la vie quotidienne à partir de fouille de données sur internet ?".

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES ABRÉVIATIONS

AVQ	Activité de la Vie Quotidienne
AIVQ	Activité Instrumentale de la Vie Quotidienne
AP	Précision moyenne ("Average Precision")
CNN	Réseau de Neurones Convolutif ("Convolutionnal Neural Network")
FP	Faux Positif(s)
FN	Faux Négatif(s)
HAROUP	Human Activity Recognition using Object Usage Patterns
HMM	Réseau de Markov Caché ("Hidden Markov Model")
LIARA	Laboratoire d'Intelligence Ambiante pour la Reconnaissances d'Activités
MAP	Moyennes des Précisions moyennes ("Mean Average Precision")
MCC	Coefficient de corrélation de Matthews ("Matthews correlation coefficient")
PF-IAF	Pattern Frequency - Inverse Activity Frequency
POS	Étiquetage Morpho-syntaxique ("Part-Of-Speech Tagging")
RFID	Radio Frequency Identification
ROR	Taux d'Objets Reconnaissables ("Recognizable Objects Rate")
SPM	Forage de Motifs Séquentielles ("Sequential Pattern Mining")
TF-IDF	Term Frequency - Inverse Document Frequency
VMCSP1	Vertical mining of Maximal and Closed Sequential Patterns in sequences of 1-uplets
VP	Vrai(s) Positif(s)
VN	Vrai(s) Négatif(s)

REMERCIEMENTS

Rédiger un document aussi sérieux et complet qu'une thèse est une épreuve en soi. Je tiens donc à remercier plusieurs personnes pour le soutien apporté tout au long de cette expérience. Je remercie en tout premier lieu ma compagne Éva Strady de m'avoir soutenu et écouté dans toutes les situations. Je remercie également mes superviseurs Kévin Bouchard et Sébastien Gaboury de m'avoir guidé à travers leurs conseils. Je souhaite aussi remercier mes amis Cédric Démongivert, Geoffrey Glangine, Morgane Cabrol, Killian Lachaux pour leur présence rassurante tout au long de ce doctorat. Pour finir, je remercie ma famille qui, malgré la distance, s'est toujours assuré que j'avais tout ce dont j'avais besoin.

CHAPITRE I
INTRODUCTION

1.1 CONTEXTE DE RECHERCHE

L'évolution de la population humaine est un sujet qui, grâce à l'amélioration du recensement dans de nombreux pays, permet de prendre conscience des tendances existantes. L'une des tendances les plus marquées de ces dernières décennies est le vieillissement de la population mondiale. D'après le rapport *World Population Ageing 2017* de United Nations, Department of Economic and Social Affairs (2017), le nombre de personnes âgées de plus de 60 ans est passé de 382,5 millions en 1980 à 962,3 millions en 2017, soit une augmentation de 151,6%. De plus, les projections pour 2050 parlent d'un nombre de personnes âgées de plus de 60 ans de 2,0805 milliards, soit une augmentation de 116,2%. En plus de cette augmentation, le rapport précise qu'à l'échelle mondiale, le nombre de personnes âgées augmente plus rapidement que le nombre de personnes de tout autre groupe d'âge plus jeune. Ainsi le nombre de personnes âgées de plus 60 ans estimé en 2050 serait au moins 200% supérieur par rapport à l'année 2000 alors que le nombre de personnes âgées de 25 à 59 ans serait seulement 64% supérieur à celui de l'année 2000.

Ce vieillissement que connaît la population humaine a notamment pour conséquence immédiate une augmentation du nombre de personnes en perte d'autonomie, voire sans autonomie. Les maladies neurodégénératives sont étroitement liées au déclin de l'autonomie. On parle alors de troubles comme la maladie d'Alzheimer, la maladie de Parkinson et des différents handicaps que la vieillesse provoque (handicap physique, sensoriel ou intellectuel). D'après le rapport *World Alzheimer Report 2015* de Wimo *et al.* (2015), les maladies neurodégénératives ont un impact à 3 niveaux liés entre eux. Le premier concerne la personne atteinte d'une

ou plusieurs maladies neurodégénératives, qui souffre de problèmes de santé, d'invalidité, d'une mauvaise qualité de vie et d'une espérance de vie réduite. Le second est l'impact sur la famille et les amis de la personne atteinte, qui, dans toutes les régions du monde, sont la pierre angulaire du système de soins et de soutien. Pour finir, il y a également un impact sur la société au sens large, qui soit directement par le biais des dépenses publiques, soit par d'autres moyens, supporte le coût de la fourniture de soins de santé et de services sociaux et le coût d'opportunité de la perte de productivité.

Le vieillissement de la population engendre donc un certain nombre de défis liés aux points précédents tels que la pénurie de travailleurs dans certaines régions, l'adaptation des modèles de soins et la gestion des relations intergénérationnelles. L'un des principaux émerge d'une volonté croissante (Wimo *et al.*, 2015) de maintenir à domicile de manière prolongée les aînés et les personnes semi-autonomes. Les objectifs de cette démarche sont de réduire les dépenses publiques, maintenir la proximité avec la famille, les amis et de favoriser le bien être des personnes en perte d'autonomie. Afin de permettre ce maintien, il est alors nécessaire d'adapter l'environnement de vie de ces personnes en prenant en considération leurs capacités et leurs limites.

Ces dernières décennies, les progrès en matière de technologies de l'information, de technologie embarquée ainsi qu'en microélectronique ont fait émerger de nouveaux concepts permettant de favoriser le maintien à domicile des personnes en perte d'autonomies. L'un de ces concepts est celui de l'Intelligence Ambiante. D'après Ducatel *et al.* (2003), vivre dans un environnement avec Intelligence Ambiante signifie d'être entouré d'interfaces intelligentes soutenues par l'informatique et les technologies de réseaux qui sont intégrés dans les objets du quotidien (meubles, vêtements, véhicules, routes, matériaux). Elle implique un environnement homogène d'informatique, de technologies réseau avancées et d'interfaces spécifiques. Cet environnement devrait être conscient des caractéristiques spécifiques de la présence humaine,

et des particularités de chaque personnalité ; s'adapter aux besoins des utilisateurs ; être capable de répondre intelligemment aux indications verbales ou gestuelles du désir ; et même aboutir à des systèmes capables de s'engager dans un dialogue intelligent. Enfin, l'Intelligence Ambiante devrait également être discrète permettant d'avoir une interaction agréable et facile à prendre en main pour le résident.

En résumé, ce concept représente une solution technologique au besoin de maintenir à domicile les personnes en perte d'autonomie. Cependant, la création d'un habitat utilisant le concept d'Intelligence Ambiante présente son lot de défis scientifiques. L'un des principaux consiste en l'exploitation des données brutes provenant de capteurs hétérogènes distribués dans l'habitat intelligent, afin de parvenir à reconnaître les Activités de la Vie Quotidienne (AVQ) du résident. Les chercheurs se sont penchés sur ce domaine, qu'on appelle la reconnaissance d'activités (Kim *et al.*, 2010; Chapron *et al.*, 2020; Maitre *et al.*, 2019), depuis quelques décennies.

Cette thèse s'inscrit dans l'amélioration des technologies utilisées en Intelligence Ambiante en reconnaissance d'activités et à ce titre, les notions essentielles du domaine sont définies dans les sections qui suivent.

1.2 L'ACTIVITÉ HUMAINE

Comme nous l'avons vu dans la partie précédente, le maintien à domicile des personnes en perte d'autonomie pourrait être prolongé par la mise en pratique du concept d'Intelligence Ambiante. Son application consistant en grande partie à s'adapter aux besoins du résident, il semble alors essentiel de connaître ces derniers. L'un des moyens pour connaître les besoins du résident est de définir quelles sont les actions qu'il est en train de réaliser en temps réel. Ces actions peuvent être regroupées en activités humaines ; chaque activité représentant une

unité de sens contenant la réalisation d'un but ou l'application d'un plan. Par exemple, selon la définition précédente, *regarder la télévision* peut être considérée comme une activité humaine.

Il existe différents formalismes pour définir plus précisément ce qu'est une activité humaine. Le premier formalisme est celui de l'Activité de la Vie Quotidienne ou AVQ (Activity of Daily Living ou ADL). Introduit par Katz *et al.* (1963) et aujourd'hui utilisé dans le milieu de la santé, les AVQs regroupent l'ensemble des activités nécessaires pour prendre soin de son propre corps (se laver, s'habiller, se nourrir, etc.). Une fois les critères de succès et d'échec de ces activités définis, les professionnels de santé disposent alors d'une référence utile à la quantification de la perte d'autonomie d'une personne.

Un second formalisme créé par les chercheurs est celui des activités instrumentales de la vie quotidienne ou AIVQ (*Instrumental Activities of the Daily Living*). À la différence des AVQ, les AIVQs ne sont pas fondamentales à la survie, mais elles permettent à un individu de vivre de façon autonome dans une communauté (gérer son argent, préparer des repas, utiliser le téléphone, prendre ses médicaments, maintenir la propreté de la maison, etc.). Les AIVQs sont des activités plus complexes nécessitant un effort de planification contrairement aux AVQs.

Ces formalismes, qui permettent de quantifier la perte d'autonomie d'une personne dans un environnement intelligent, sont également utiles en tant que référence pour détecter les chutes, les comportements dangereux ou à risque ainsi que tout ce qui pourrait porter atteinte à la sécurité de la personne. Cependant, avant d'utiliser les AVQs et AIVQs comme références et d'effectuer des comparaisons entre des comportements normaux et les comportements réels du résident, il est d'abord nécessaire de procéder à l'identification des activités réalisées par le résident en perte d'autonomie. Cette identification représente un champ de recherche entier que l'on nomme la reconnaissance d'activités humaines (Aggarwal & Ryoo, 2011; Ann & Theng, 2014).

1.3 LA RECONNAISSANCE D'ACTIVITÉS HUMAINES

1.3.1 HISTORIQUE ET DÉFINITION

Avec l'augmentation de la disponibilité des données, les humains disposent de plus en plus de ressources sur lesquelles ils peuvent compter pour raisonner et mieux comprendre le monde qui les entoure. Pour appliquer ces raisonnements, dans de nombreux domaines comme l'Intelligence Ambiante, il est souvent utile d'avoir la capacité de reconnaître les buts et objectifs des autres agents ou personnes considérés. Il est alors possible de comprendre ce qu'ils font, pourquoi ils le font, mais aussi ce qu'ils pourraient faire ensuite. Les recherches effectuées pour traiter ce genre de problème font souvent référence à des termes comme reconnaissance d'activités, reconnaissance de plans ou reconnaissance d'intentions.

Historiquement, selon Sukthankar *et al.* (2014), c'est le domaine de la reconnaissance du plan qui est étudié en premier. Initialement définis par Schmidt *et al.* (1978), les premiers travaux ont utilisé des systèmes basés sur des règles d'inférence [1] créées manuellement par les chercheurs. Au fil du temps, il est devenu évident qu'en l'absence d'une théorie sous-jacente pour leur donner structure et cohérence, ces ensembles de règles définies manuellement sont difficiles à maintenir et ne sont pas très généralisables. Plusieurs paradigmes ont été explorés pour fournir un certain cadre de reconnaissance des plans, comme une représentation des plans sous la forme d'un graphe Kautz & Allen (1986).

Par la suite, beaucoup de travaux dans ce domaine ont été faits dans le cadre de la reconnaissance d'activités. La reconnaissance d'activités ainsi que la reconnaissance de plans peuvent être définies comme suit. L'objectif est d'inférer sur les buts d'une entité capable de

1. Le terme *règle d'inférence* fait référence à des règles logiques se basant sur le processus de déduction. Une règle d'inférence possède des prémisses en entrée et une conclusion en sortie. Le fonctionnement d'une règle est le suivant : si l'on dispose de toutes les prémisses alors on obtient la conclusion. Exemple, dans le domaine de la reconnaissance d'activités, on pourrait avoir la règle suivante : *Si le résident reste immobile pendant plus de 3 h, alors le résident est en train de dormir*

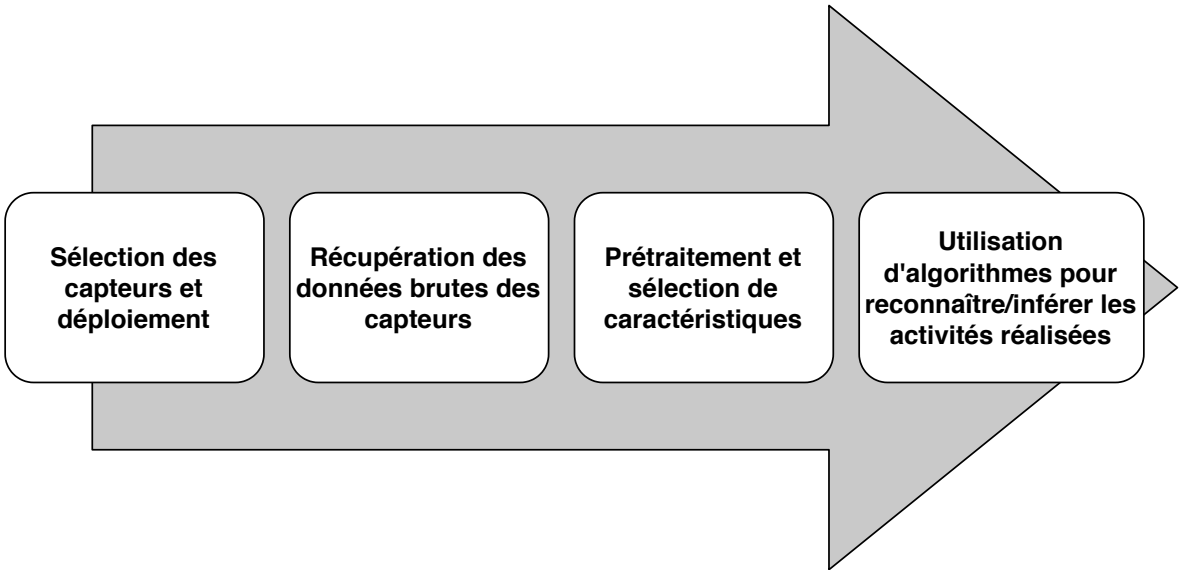

Figure 1.1 : Processus de la reconnaissance d'activités dans un contexte d'Intelligence Ambiante (Hussain *et al.*, 2019). Image Creative Commons.

prendre des décisions (personne, programme informatique, entreprise, etc.) au vu des actions réalisées par celle-ci observées dans un environnement donné. D'autre part, en reconnaissance d'activités, nous essayons de trouver les relations entre les différentes activités qui constituent un plan complet. En pratique, les termes *reconnaissance de plans* et *reconnaissance d'activités* sont largement confondus. Le choix du terme dépend généralement du contexte d'application. Par exemple, dans le contexte de l'Intelligence Ambiante, la reconnaissance d'activités peut être définie comme la capacité à transformer des données brutes de capteurs provenant du milieu de vie du résident, qui sont des données de bas niveau, en des modèles d'activités, qui sont des données de haut niveau, sur lesquels il est possible de raisonner. Ce processus est détaillé dans la Figure 1.1, inspirée du travail de Hussain *et al.* (2019).

1.3.2 CAPTEURS UTILISÉS EN RECONNAISSANCE D'ACTIVITÉS

L'une des premières questions posées lorsque l'on conçoit une approche de reconnaissance d'activités est de savoir quelle sera notre source de données brutes, en d'autres termes, quel choix devons-nous faire par rapport aux technologies existantes permettant d'obtenir de l'information sur un résident dans un habitat intelligent. Pour répondre à cette question, il est important de savoir que les capteurs utilisés pour la reconnaissance d'activités peuvent être très variés ; ayant même donné lieu à une forme de classification des approches de reconnaissances par capteurs.

On pourra par exemple faire de la reconnaissance d'activités basée sur la vidéo en utilisant des caméras (*Vision-based Human Activity Recognition*). Dans son travail, Bux *et al.* (2017) explique le processus comme suit. La vidéo est en général segmentée par la différenciation de son arrière-plan du reste de l'image en utilisant différentes techniques de segmentation d'objets. Après la segmentation, les caractéristiques [2] importantes des silhouettes sont extraites et présentées sous forme d'un ensemble de caractéristiques. Ensuite, ces caractéristiques sont utilisées pour la classification à l'aide de n'importe quel classifieur[2]. D'après Zheng *et al.* (2009), ces approches, en plus d'utiliser des techniques de classification et de segmentation d'images, peuvent également utiliser une structure hiérarchique des activités humaines à trois niveaux : le niveau des actions primitives (ex : *étirer le bras gauche*), le niveau des actions/activités (*parler en bougeant ses bras*) et le niveau des interactions qui se réfère aux activités humaines qui impliquent plus de deux personnes et objets (ex : *avoir une conversation*). Dans son travail, Ann & Theng (2014) explique que l'une des principales contraintes de l'approche utilisant la vidéo est qu'elle nécessite souvent un traitement machine élevé. Ainsi, la performance en temps réel peut être affectée lorsque beaucoup de données sont traitées

2. Nous reviendrons plus tard sur la notion de caractéristiques et de classifieur (Sections 2.1 et 1.3.3). Pour l'instant nous pouvons voir les caractéristiques comme des données brutes prétraitées et le classifieur comme un algorithme prenant en entrée des caractéristiques et qui prédira l'activité en cours

simultanément. Une autre préoccupation soulevée lors de l'utilisation d'une caméra est la question de la protection de la vie privée. Un être humain, c'est-à-dire une personne âgée, peut se sentir mal à l'aise ou gêné à l'idée d'être surveillé et observé en tout temps.

Dans le cas où la vie privée est une priorité et comme peut le dire Subasi *et al.* (2018), il est préférable d'utiliser un ensemble de capteurs hétérogènes non intrusifs comme des capteurs RFID, des capteurs de mouvements et des capteurs de contact magnétique. Cette approche est appelée *Sensor-based Human Activity Recognition*. En raison des questions de vie privée nommées plus haut, cette approche est de plus en plus populaire par rapport à l'approche basée sur la vidéo (Wang *et al.*, 2019; Liu *et al.*, 2016). De plus, grâce à l'accessibilité accrue des dispositifs portatifs et des capteurs, elle a attiré l'attention des chercheurs ces dernières années. Il est remarquable de noter que l'on peut classer les approches de *Sensor-based Human Activity Recognition* en deux grandes catégories en fonction de la nature des capteurs utilisés, à savoir les capteurs distribués et les capteurs portables.

Dans le cas de l'utilisation de capteurs distribués, les capteurs sont en général qualifiés de statiques, car ils restent placés au même endroit pendant toute la durée de leur utilisation. L'idée est de déployer des capteurs dans l'environnement (le logement de la personne en perte d'autonomie) et lorsqu'une personne exécute une activité, les données seront capturées grâce à ces capteurs, qui peuvent ensuite être utilisées pour la reconnaissance d'activités. Néanmoins cette approche comporte aussi certains défis, comme les interférences avec l'environnement. En effet, les données capturées par les capteurs peuvent être perturbées par l'environnement de vie, ce qui peut provoquer du bruit dans les données (Hussain *et al.*, 2019). Finalement, on notera que cette approche a le net avantage d'être pratique parce qu'elle n'exige pas que l'utilisateur porte un capteur sur son corps pendant la réalisation d'une activité.

Certaines approches de reconnaissance d'activités font un choix différent en se tournant vers l'utilisation de capteurs portatifs (*wearable devices*). Aussi appelée *Wearable-based Human Activity Recognition* dans la littérature scientifique, elle nécessite d'utiliser un ou plusieurs capteurs à fixer sur le corps humain. Les capteurs les plus couramment utilisés comprennent l'accéléromètre à 3 axes, le magnétomètre, le gyroscope et les tags RFID comme dans les travaux de Maitre *et al.* (2019); Chapron *et al.* (2018, 2017); Zhang & Sawchuk (2012); Reiss *et al.* (2013). Aussi, avec l'avancement des technologies actuelles des téléphones intelligents, de nombreux travaux comme celui de Reiss *et al.* (2013) utilisent le téléphone intelligent comme dispositif de détection parce qu'ils sont équipés, la plupart du temps, d'un accéléromètre, d'un magnétomètre et d'un gyroscope. Une activité humaine peut alors être facilement identifiée en analysant les données générées par les différents capteurs portables après avoir été traitées et interprétées par un algorithme de classification.

1.3.3 ALGORITHMES DE CLASSIFICATION POUR LA RECONNAISSANCE D'ACTIVITÉS

Dans la partie précédente, nous avons vu qu'il existe plusieurs manières de faire de la reconnaissance d'activités, notamment par le choix des capteurs utilisés. Il existe d'autres critères pour classer les approches de reconnaissance d'activités humaines comme le nombre de résidents dont on souhaite connaître les actions, la capacité à reconnaître des activités anormales, incomplètes et se déroulant parallèlement entre elles. Tous ces critères représentent des choix essentiels à effectuer si l'on souhaite créer une approche de reconnaissance d'activités ; et en tant que tels, ils sont essentiels pour définir le cadre dans lequel on effectue la tâche de reconnaissance. Cependant, malgré leur importance, l'étape la plus importante reste le choix de l'algorithme avec lequel on peut reconnaître la ou les activités effectuées par le ou les résidents en fonction des données disponibles (pouvant être fournies par des capteurs comme dans la

Figure 1.1). Cette étape consiste alors à faire correspondre les données disponibles (fournies par des capteurs) avec la réalisation d'une ou plusieurs activités. Il existe de nombreuses manières d'effectuer cette *correspondance*, mais la plus courante consiste à considérer la reconnaissance d'activités comme un problème de classification. La tâche de classification est un cas particulier de l'apprentissage automatique où les données d'apprentissage sont étiquetées de manière discrète, c'est-à-dire que, dans le jeu de données d'apprentissage, à chaque donnée correspond une catégorie parmi un ensemble préétabli [3]. Ces catégories sont appelées classes en classification. Nous définirons la notion d'apprentissage automatique et verrons les différents types d'algorithmes de classification pour la reconnaissance d'activité dans le chapitre 2.

1.4 LE WEB

Jusqu'à maintenant, nous avons présenté la notion d'activité, le domaine de la reconnaissance d'activités ainsi que les différents types d'approches qui existent pour en faire. Cette thèse tente d'utiliser les ressources du web pour contribuer au domaine de la reconnaissance d'activités. Ainsi, il est important de présenter et de définir la notion de web. La popularité du web de nos jours n'implique pas nécessairement que la notion en elle-même soit bien comprise. Souvent confondu avec d'autres termes comme *internet* ou *cloud*, le web ou World Wide Web (Berners-Lee & Cailliau, 1990) est une application d'Internet comme peuvent l'être les courriels et le service FTP.

Le web est modélisable par un graphe orienté possédant des cycles avec les ressources en tant que sommets et les hyperliens en tant qu'arcs. Aujourd'hui, on peut raisonnablement dire que la facilité avec laquelle il est possible de naviguer entre les pages web grâce aux hy-

3. Par exemple, dans un environnement intelligent, on pourrait avoir l'étiquetage suivant : les données fournies entre l'instant t_0 et t_1 correspondent à la réalisation de l'activité (l'étiquette) *faire la vaisselle* parmi les activités (les étiquettes) *faire la vaisselle*, *faire le ménage* et *regarder la télévision*

perliens et l'avènement des moteurs de recherche comme Google a eu plusieurs conséquences. Premièrement, on remarque que le web de 2019 contient une abondance d'informations sur tous les sujets qui intéressent les humains (science, art, culture, politique, divertissement, etc.). La seconde conséquence est que, dû à cette popularité qui est toujours actuelle, ces informations sont mises à jour très régulièrement. Au vu de ces éléments, notre équipe de recherche pense que l'on peut affirmer, sans trop se risquer, que le web semble pouvoir fournir un certain reflet de l'humanité ; que cela soit au niveau des connaissances, des besoins, des envies, du fonctionnement de notre espèce.

Avec le constat précédent, on peut alors se questionner : est-ce que les ressources disponibles sur le web peuvent servir pour construire ou améliorer une approche de reconnaissance d'activités ? Et si c'est le cas, de quelle manière ? Ces questions représentent les fondements de cette thèse ; par conséquent elle tentera de répondre à ces questions de la manière la plus détaillée possible. Cependant, avant d'émettre une réponse détaillée, on peut au moins tenter de raisonner sur le genre d'informations disponibles. On peut, par exemple, commencer par remarquer que le web regorge de sites web descriptifs sur les activités de la vie quotidienne. Par exemple, on peut citer des sites comme wikihow [4] et ehow [5] qui contiennent une grande quantité d'informations pertinentes sur toute une panoplie d'activités. Les Figures 1.2 et 1.3 montrent, respectivement, la page web descriptive de la réalisation de l'activité *cuire des coquillettes* et une étape présente sur cette page. On constate alors que ces sites fournissent de l'information textuelle sur la réalisation d'une activité (ex : *faire cuire 500g de coquillettes*) ainsi que de l'information en image (ex : une casserole remplie d'eau avec des coquillettes dans la Figure 1.3).

4. https://www.wikihow.com/

5. https://www.ehow.com/

Figure 1.2 : Capture d'écran d'une page décrivant l'activité *cuire des coquillettes* (https://fr.wikihow.com/cuire-des-coquillettes consulté le 14/11/2019). Contenu Creative Commons.

Un autre constat est que les résultats fournis par les moteurs de recherche d'images peuvent être très pertinents et fournir des images ayant une relation forte avec la requête effectuée. Par exemple, en utilisant la requête *faire du thé* sur le moteur de recherche d'image de Google, on obtient un ensemble d'images représentant des tasses, des mugs, du thé, des cuillères, etc. La Figure 1.4 montre ces résultats. En d'autres mots, les moteurs de recherche d'images ont le potentiel de fournir tout ou partie des objets et des substances impliquées dans la réalisation d'activités ; et cette thèse tentera d'évaluer l'ampleur de ce potentiel.

Enfin, on peut également remarquer que les moteurs de recherche semblent devenir de plus en plus performants au fil des années. Pour ne citer qu'un exemple de cette amélioration de performance, en août 2019, le moteur de recherche de Google a atteint une situation très particulière : moins de la moitié des recherches Google aboutissent maintenant à un clic (Fishkin, 2019). En effet, si les premiers résultats sont très pertinents, alors le texte affiché sur ces premiers résultats (aussi appelé *snippet*) a de meilleures chances de directement répondre aux questionnements de l'internaute effectuant la recherche et donc, ne nécessitant pas de clic.

1.5 DÉFINITION DU PROJET DE RECHERCHE

La thèse s'articule autour de l'exploitation du contenu disponible sur le web pour reconnaître les activités des résidents dans les habitats intelligents et permettrait de répondre aux questions suivantes :

1. Dans quelles mesures le web, en termes d'informations disponibles, permet-il de représenter les activités de la vie quotidienne comparativement aux approches existantes ?
2. Comment parvenir à trouver et exploiter de manière automatisée les informations disponibles sur le web en rapport avec les activités de la vie quotidienne ?

Figure 1.3 : Capture d'écran d'une étape d'une page décrivant l'activité *cuire des coquillettes* (https://fr.wikihow.com/cuire-des-coquillettes consulté le 14/11/2019). Contenu Creative Commons.

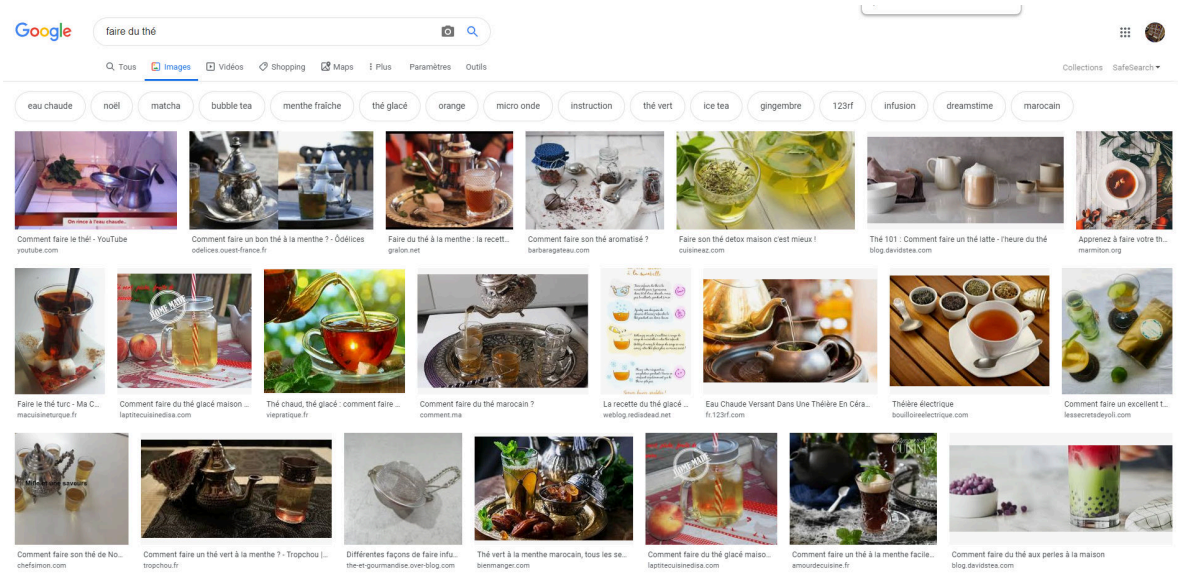

Figure 1.4 : Capture d'écran des résultats du moteur de recherche d'image de Google avec la requête *faire du thé* (14/11/2019)

3. Une approche de reconnaissance d'activités humaines utilisant les connaissances du web peut-elle être performante ?

La contribution de cette thèse consiste en trois éléments distincts. Premièrement, notre contribution théorique est la proposition d'une méthode d'extraction d'informations du web sur la réalisation d'activités de la vie quotidienne ainsi que d'une méthode de reconnaissance adaptée à ces informations et à différents contextes d'applications. Ensuite, notre contribution pratique consiste en un système complet en JavaScript contenant 3 modules :

1. Un module de *web scraping* pour obtenir les résultats de moteurs de recherche
2. Un module d'identification et d'extraction des informations sur les activités
3. Une implémentation d'algorithmes de reconnaissance pour quelques contextes d'application

Enfin, la contribution expérimentale est la réalisation d'expérimentations sur la capacité de reconnaissance d'activités d'algorithmes se reposant sur des informations extraites du web à

l'aide de jeux de données inspirés de jeux de données bien connus comme CASAS (CASAS, 2019).

1.6 ORGANISATION DU DOCUMENT

Le second chapitre traitera du processus et des différents types de reconnaissance d'activités. Le troisième chapitre discutera de l'usage du web dans les approches de reconnaissance d'activités, de son apport pratique et théorique. Le quatrième chapitre est dédié à une méthode de détection d'objets clés à partir d'images venant du web. Le cinquième chapitre présente une méthode d'extraction de motifs séquentiels à partir de texte de page web. Le sixième chapitre, quant à lui, a pour objectif de mettre en application les motifs et les objets clés extraits dans un contexte réel de reconnaissance d'activités. Pour finir, le septième chapitre conclut l'ensemble de la thèse en faisant un retour sur les contributions, les limites rencontrées et les travaux futurs.

CHAPITRE II

PROCESSUS ET TYPES DE RECONNAISSANCE D'ACTIVITÉS

Dans un contexte d'Intelligence Ambiante, c'est-à-dire pour valoriser le bien être et la sécurité du résident, une bonne approche de reconnaissance d'activités humaines devrait être capable d'utiliser les données fournies par des capteurs (des données de bas niveau) pour obtenir des informations ou des connaissances sur l'état et les actions réalisées par le résident (des données de haut niveau). Dans cette vision de la reconnaissance d'activités, on peut aisément comprendre que le niveau d'expressivité des capteurs de l'agent observateur est très important. En effet, des capteurs trop peu expressifs [6], utilisées seuls ne permettront pas d'obtenir de l'information de haut niveau de qualité. En revanche, des capteurs très expressifs comme des caméras permettent d'en obtenir bien plus facilement, mais les données produites peuvent être plus difficiles à traiter.

Il existe de très nombreuses manières de faire de la reconnaissance d'activités, utilisant toutes des capteurs avec des niveaux d'expressivité très différents. Cela a pour conséquence qu'il est parfois difficile de comparer des approches de reconnaissance entre elles à cause de cette variété. Malgré la multiplicité des approches existantes, un très grand nombre d'entre elles se base sur une notion commune depuis un certain nombre d'années, l'apprentissage machine.

Le chapitre précédent introduisait la notion de reconnaissance d'activités ainsi que le contexte et les objectifs du projet de recherche, mais avant d'analyser tout ce que le web pourrait apporter, le présent chapitre commencera par présenter l'intérêt de l'apprentissage

6. À titre d'exemple, les capteurs binaires comme ceux permettant de dire si une porte est ouverte ou fermée ont un faible niveau d'expressivité

machine pour le domaine de la reconnaissance d'activités puis décrira la notion en elle même, ses étapes constituantes et détaillera les approches existantes.

2.1 L'APPRENTISSAGE POUR LA RECONNAISSANCE D'ACTIVITÉS

L'une des manières d'effectuer de la reconnaissance d'activités a longtemps consisté en la programmation en dur de règles de reconnaissance, c'est-à-dire que de la connaissance, trouvée au préalable par les chercheurs, devait être encodées dans un programme informatique afin que ce dernier raisonne pour effectuer des inférences sur l'activité en cours. Cette manière de procéder a eu différents types d'applications. Un premier exemple est les applications reposant sur la notion de *système expert*. Défini par Mitchell (1997) comme un système informatique émulant le comportement d'un expert humain à l'intérieur d'un domaine de connaissance bien défini et étroit, un système expert est un programme composé de 4 éléments distincts :

1. Une base de connaissances (les faits et règles de déduction)
2. Un moteur d'inférence agissant comme un mécanisme de résolution
3. Un module de dialogue qui représente une interface d'interaction avec un usager
4. Un expert humain qui interagit avec le système pour maintenir les connaissances du système

La Figure 2.1 représente l'architecture classique des systèmes experts.

Dans une approche de reconnaissance d'activités reposant sur un système expert, il est tout à fait possible de préciser la nature des règles et des faits utilisés. Un fait peut être toute affirmation ou toute donnée provenant du milieu de vie du résident, allant de *la porte vient de se fermer* à *quelqu'un regarde la télévision* (une activité), en passant par *quelqu'un fait chauffer de l'eau*. En conséquence, ces faits peuvent avoir des niveaux d'expressivité très différents ; le

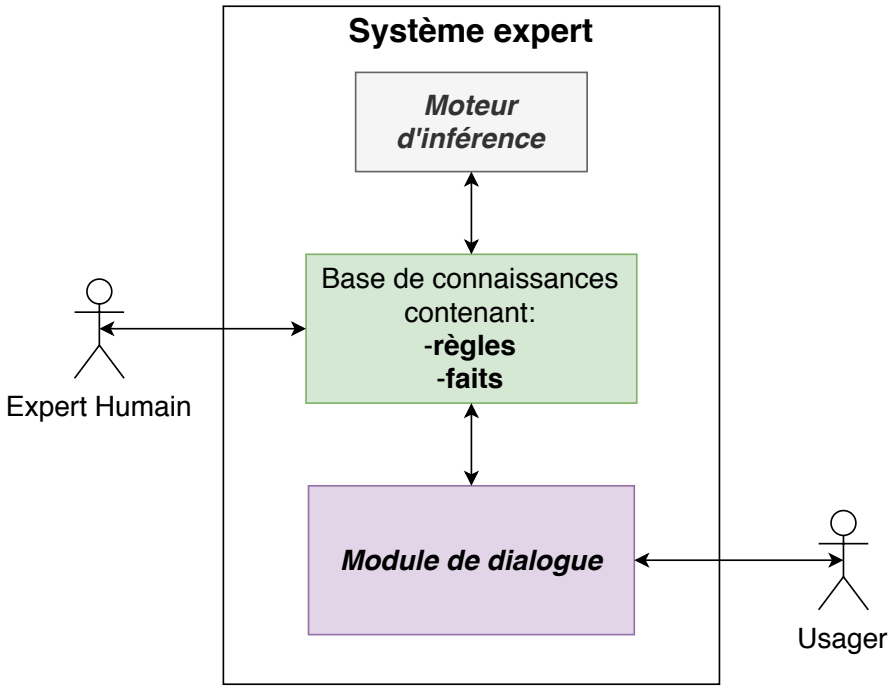

Figure 2.1 : Architecture des systèmes experts. © Charles Cousyn, 2022.

choix des niveaux d'expressivité utilisés dépend de l'architecture informatique utilisée dans l'environnement de vie du résident, des activités à reconnaître et de leur complexité ainsi que du moteur d'inférence utilisé. Les règles, de leur côté, permettent d'établir des relations entre ces faits par un processus de déduction. La représentation ainsi qu'un exemple de règle d'inférence est donnée dans la Figure 2.2.

Cette approche est connue comme une approche basée sur la connaissance (ou *knowledge-based*). Ces approches ne sont en général pas les plus performantes. La cause principale est la qualité insuffisante des règles encodées dans le programme pour résoudre le problème de la reconnaissance d'activités. Ces règles ayant été conçues et encodées par des superviseurs humains, on comprend alors que les humains peuvent avoir de la peine à concevoir des règles d'une complexité suffisante pour résoudre un problème aussi complexe que la reconnaissance d'activités. Comme le dit Goodfellow *et al.* (2016), les difficultés rencontrées par les systèmes

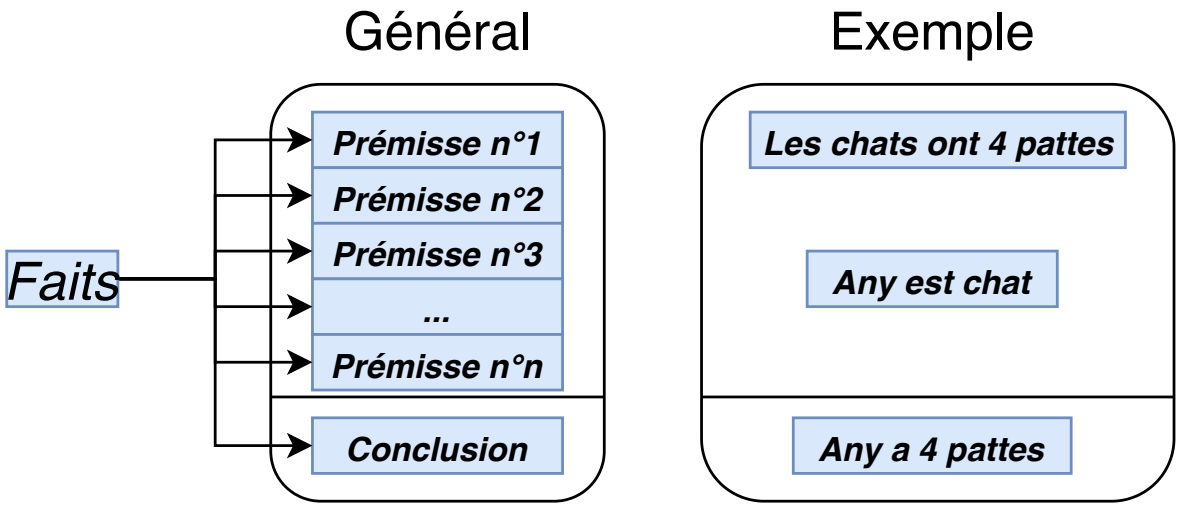

Figure 2.2 : Représentation et exemple de règle d'inférence. © Charles Cousyn, 2022.

programmés en dur suggèrent que les systèmes devraient être capables d'acquérir leurs propres connaissances, en extrayant des motifs de la donnée brute. Cette capacité est connue sous le nom d'*apprentissage automatique* ou *apprentissage machine*. Le terme *apprentissage machine* a pour la première fois été utilisé par Samuel (1959). L'une des définitions les plus formelles est sûrement celle donnée par Mitchell (1997) : "On dit qu'un programme informatique apprend de l'expérience E par rapport à une certaine classe de tâches T et à la mesure de performance P si sa performance aux tâches T, telle que mesurée par P, s'améliore avec l'expérience E.". La signification pratique de cette définition désigne un programme capable d'apprendre en tentant d'améliorer ses propres performances. L'apprentissage automatique a permis aux ordinateurs d'aborder des problèmes complexes impliquant la connaissance du monde réel et de prendre des décisions qui semblent subjectives au premier abord comme la reconnaissance d'objets sur les images (Ren *et al.*, 2017).

Il peut être décrit comme un processus composé des étapes successives suivantes : prétraitement, segmentation, extraction de caractéristiques, apprentissage, évaluation des performances. Ces différents éléments sont décrits dans la section qui suit.

2.1.1 LES ÉTAPES PRÉLIMINAIRES

Dans le contexte d'Intelligence Ambiante, on utilise une variété de capteurs capables de fournir un ensemble d'informations sur le monde. Ces informations sont qualifiées de données *brutes*. Ce sont des données non interprétées provenant d'une source ayant des caractéristiques liées à celle-ci et n'ayant subi aucune manipulation ou aucun traitement. Les données brutes ne sont pas, la plupart du temps, exploitables dans l'immédiat pour une quelconque utilisation. La raison à cela est la présence d'une part d'information non désirée, aussi appelée *bruit*.

On peut définir le bruit comme toute composante non désirée affectant la sortie d'un dispositif, et ce, indépendamment du signal présent en entrée du dispositif. Toute source de données du monde réel peut être sujette au bruit ; il a deux principales origines : les erreurs introduites par les outils de mesure et les erreurs aléatoires introduites par le traitement ou par des experts lors de la collecte des données (Zhu & Wu, 2004). Tout le défi repose sur le fait de réduire la quantité de bruit présente dans les données brutes afin d'avoir le plus d'information désirée possible ; on appelle cette phase, le prétraitement.

Après cette phase de prétraitement, dans plusieurs domaines reposant souvent sur l'apprentissage automatique comme la vision par ordinateur (Szeliski, 2010), les données peuvent nécessiter une phase supplémentaire ayant pour but de les rendre plus informatives et moins redondantes. Cette phase, appelée extraction de caractéristiques, permet de faciliter la phase qui suit, l'apprentissage.

2.1.2 LES ALGORITHMES D'APPRENTISSAGE

Nous avons déjà introduit la notion d'apprentissage automatique au début de la section 2.1, cependant, il faut bien comprendre qu'il existe différents types d'apprentissages permettant de résoudre des problèmes différents. Nous allons définir ces différents types dans cette section.

Pour commencer, l'apprentissage automatique peut être supervisé ou non supervisé. Un apprentissage sera dit supervisé à deux conditions :

— les données utilisées sont étiquetées (un *superviseur* a fourni cette étiquette)

— l'algorithme d'apprentissage apprend en fonction de ces couples donnée-étiquette.

Pour illustrer, on peut imaginer une situation où l'on souhaite construire un système capable de prédire la moyenne d'un étudiant à partir de son nombre d'heures d'absences. Pour faire de l'apprentissage supervisé pour ce problème, il sera alors nécessaire de disposer d'un ensemble de couples heure d'absences-moyenne générale provenant de multiples étudiants. Dans le cas présent, on tente de prédire une variable quantitative (ou numérique), la moyenne d'un étudiant ; c'est ce qu'on appelle une régression. Cependant, la variable à prédire peut également être qualitative, c'est-à-dire qu'elle possède un nombre limité de valeurs ; aussi appelées *classes* ou *catégories* ; c'est ce qu'on appelle de la classification. Par exemple, on peut tout à fait imaginer qu'on veuille construire un système capable de prédire s'il y a un chat ou non sur une image donnée. Dans ce cas, il s'agit d'une classification binaire où les données en entrées sont des images et les classes sont *Présence d'un chat* et *Absence de chat*.

À la différence de l'apprentissage supervisé, l'apprentissage non supervisé effectue son apprentissage sans l'aide d'un superviseur, c'est-à-dire, sans données étiquetées. Il doit donc apprendre à partir de la structure même et des informations latentes des données . La tâche la plus courante d'apprentissage non supervisé est la segmentation (*clustering*) (Xu & Tian,

2015). Elle consiste à exploiter les données en formant des groupes appelés *clusters* et ces *clusters* doivent posséder les propriétés suivantes (Sarle *et al.*, 1990) :

— les données se trouvant dans le même *cluster* doivent être les plus similaires possible,

— les données ne se trouvant pas dans le même *cluster* doivent être les plus distinctes possible,

— les mesures de similarité ou de distance doivent être claires et posséder un sens pratique.

Ces *clusters*, une fois formés, peuvent être analysés par un expert humain qui tentera d'inférer des connaissances à partir de ces regroupements.

2.1.3 ÉVALUATION DES PERFORMANCES EN RECONNAISSANCE EN TANT QUE PROBLÈME DE CLASSIFICATION

Quand on se place dans un contexte de reconnaissance d'activités en tant que problème de classification, avoir une bonne performance signifie que la classe attribuée à un ensemble de données à un instant t devrait correspondre à une véritable activité se déroulant à l'instant t. Par ailleurs, si l'on revient à la définition de l'apprentissage machine donnée en 2.1 par Mitchell (1997), on se rappelle qu'un algorithme d'apprentissage a nécessairement besoin d'avoir la capacité de connaître sa performance afin de pouvoir améliorer celle-ci. Pour cela, on utilise ce qu'on appelle une mesure de performance.

Avant de présenter les mesures de performances les plus utilisées, il est bon de noter qu'il existe un outil extrêmement utile pour mesurer les performances d'un algorithme d'apprentissage, en particulier dans le cas d'un problème de classification ; c'est la matrice de confusion. Cette matrice exprime la qualité qu'un algorithme de classification a de classer les données dans la classe correspondante. Elle se visualise comme un tableau dont chaque ligne correspond à une classe réelle, chaque colonne correspond à une classe estimée par l'algo-

		Classe estimée	
		Présence chat	Absence chat
Classe réelle	Présence chat	36 (vrai positifs)	14 (faux négatifs)
	Absence chat	8 (faux positifs)	42 (vrai négatifs)

Figure 2.3 : Exemple de matrice de confusion montrant une performance non maximale. © Charles Cousyn, 2022.

rithme (ou inversement). La case se trouvant à la ligne L et à la colonne C contiendra le nombre de données classées comme appartenant à la classe C et appartenant en réalité à la classe L. Avec une telle matrice, on peut facilement repérer si la classification se fait correctement en regardant sa diagonale : un algorithme de classification a une performance maximale si toutes les cases, hormis celles de la diagonale, possèdent des valeurs nulles. La Figure 2.3 donne un exemple de matrice de confusion dans le cas d'un problème de classification d'images avec les classes *Présence chat* et *Absence chat*.

Parmi les mesures de performances souvent utilisées, on retrouve la mesure de justesse (*accuracy*). Elle est la mesure de performance la plus intuitive, car il s'agit simplement d'un rapport entre le nombre de données correctement prédites et le nombre total d'observations. Sa formule est la suivante :

$$\text{Justesse} = \frac{\text{vrai positifs} + \text{vrai négatifs}}{\text{vrai positifs} + \text{vrai négatifs} + \text{faux positifs} + \text{faux négatifs}}. \tag{2.1}$$

La mesure de justesse, malgré sa simplicité, possède un problème qui arrive lorsqu'il existe une classe dominante parmi les classes. Par exemple, si l'on se place dans un problème de classification à deux classes (A et B) et que dans nos données d'apprentissage, 99% des

données appartiennent à la classe A, alors avec un classifieur prédisant que chaque donnée est de la classe A, on aura une justesse de 99%. Elle n'est alors pas très représentative de la performance globale du classifieur, car elle privilégie une classe particulière aux dépends des autres. Cette situation est appelée paradoxe de la justesse.

Afin d'éviter ce paradoxe, on peut faire le choix d'utiliser d'autres mesures de performances comme la précision et le rappel. Tout comme la justesse, la précision et le rappel se basent sur la matrice de confusion. Si l'on se place dans un cas de classification binaire, la précision représente la proportion de vrais positifs par rapport au nombre de données prédites comme positives et le rappel représente la proportion de vrais positifs sur la quantité de données réellement positives. Leurs formules sont les suivantes :

$$\text{Précision} = \frac{\text{vrai positifs}}{\text{vrai positifs} + \text{faux positifs}} \tag{2.2}$$

$$\text{Rappel} = \frac{\text{vrai positifs}}{\text{vrai positifs} + \text{faux négatifs}}. \tag{2.3}$$

Pour évaluer correctement les performances d'un classifieur, il est nécessaire de prendre en compte à la fois la précision et le rappel car elles quantifient des propriétés complémentaires de notre classifieur. La précision peut être considérée comme une mesure d'exactitude ou de qualité ; on se pose la question : à quel point ce que nous prédisons est pertinent ? Tandis que le rappel est une mesure d'exhaustivité ou de quantité ; on se pose la question : oublions nous des éléments pertinents dans nos prédictions ? La relation exacte entre la précision et le rappel dépend du pourcentage de cas positifs dans les données. En conséquence, on aura une bonne performance quand on aura une bonne précision et un bon rappel en même temps. À noter

qu'il existe également une mesure permettant de combiner la précision et le rappel en une seule valeur à travers leur moyenne harmonique, appelé le F-score. Sa formule est la suivante :

$$\text{F-score} = 2 \times \frac{\text{Précision} \times \text{Rappel}}{\text{Précision} + \text{Rappel}}. \tag{2.4}$$

On peut également utiliser d'autres mesures de performance pour quantifier la qualité d'un classifieur. On peut citer le kappa de Cohen Kvålseth (1989) qui évalue le degré d'accord entre le classifieur et l'expert humain. Cette mesure permet notamment de vérifier si la justesse d'un classifieur est due au hasard ou non. Plus concrètement, elle représente la proportion avec laquelle un classifieur parvient à classer correctement une donnée comparativement à un classifieur aléatoire. Mathématiquement, cela donne :

$$K = \frac{\text{Justesse Classifieur} - \text{Justesse Classifieur Aléatoire}}{1 - \text{Justesse Classifieur Aléatoire}} \tag{2.5}$$

Parmi les mesures de performances les plus utilisées, on retrouve la mesure d'aire sous la courbe ou AUC (*Area Under the Curve*) de la courbe ROC (*Receiver Operating Characteristics*). La courbe ROC est une courbe de probabilité et l'aire sous celle-ci représente le degré de *séparabilité*. Ayant en abscisse le taux de faux positifs [7] et en ordonnée le rappel, elle indique dans quelle mesure le modèle est capable de distinguer les classes entre elles. La manière le lire cette mesure de performance possède une particularité. Même si un algorithme d'apprentissage ayant de bonnes performances possédera une aire sous la courbe ROC très proche de 1.0 et que le pire classifieur possible aura une aire proche de 0, il faut comparer l'aire obtenue avec la valeur 0.5. En effet, une aire sous la courbe de 0.5 signifie que le classifieur est incapable de séparer les classes et donc qu'il n'est pas performant. Une aire sous la courbe

7. Le taux de faux positifs représente la proportion de faux positifs par rapport aux données réellement négatives ($\frac{\text{faux positifs}}{\text{faux positifs}+\text{vrai négatifs}}$)

proche de 0 signifie que le classifieur inverse les classes (il confond presque systématiquement les classes entre elles). La Figure 2.4 montre un exemple de courbe ROC.

Enfin, une dernière mesure de performance est utilisée dans les problèmes de classification. Créé par Matthews (1975), le coefficient de corrélation de Matthews ou Matthews correlation coefficient (MCC) est une mesure de corrélation similaire au coefficient de Pearson utilisé en statistiques mais appliqué aux cas de variables binaires (Guilford, 1936). C'est une mesure allant de -1.0 à 1.0 ; 1.0 signifiant une corrélation positive parfaite entre classes prédites et classes réelles, -1.0 une corrélation négative parfaite et 0.0 une absence de corrélation. Il a également été généralisé pour les cas avec plus de 2 classes (Gorodkin, 2004). Si nous parlons de cette mesure un peu moins connue, c'est parce qu'un récent travail semble montrer que le MCC présente plusieurs avantages par rapport à la justesse ou au F-score. D'après les travaux de Chicco (2017); Chicco & Jurman (2020), le MCC serait une mesure plus informative car elle prends en compte l'ensemble des termes de la matrice de confusion ; elle serait également plus fiables quand les jeux de données ne sont pas équilibrés mais cette dernière affirmation est encore débattue (Zhu, 2020).

$$MCC = \frac{\mathrm{VP} \times \mathrm{VN} - \mathrm{FP} \times \mathrm{FN}}{\sqrt{(\mathrm{VP}+\mathrm{FP}) \times (\mathrm{VP}+\mathrm{FN}) \times (\mathrm{VN}+\mathrm{FP}) \times (\mathrm{VN}+\mathrm{FN})}} \tag{2.6}$$

2.2 LES APPROCHES POUR LA RECONNAISSANCE D'ACTIVITÉS

La section précédente présente les avantages de l'utilisation des méthodes d'apprentissage automatique pour le problème de la reconnaissance d'activités humaine. Elle nous a également décrit le processus général de l'apprentissage automatique ainsi que les mesures de performances les plus communément utilisées. Ces notions sont essentielles pour bien comprendre les approches utilisées par les chercheurs. En conséquence, la présente section a

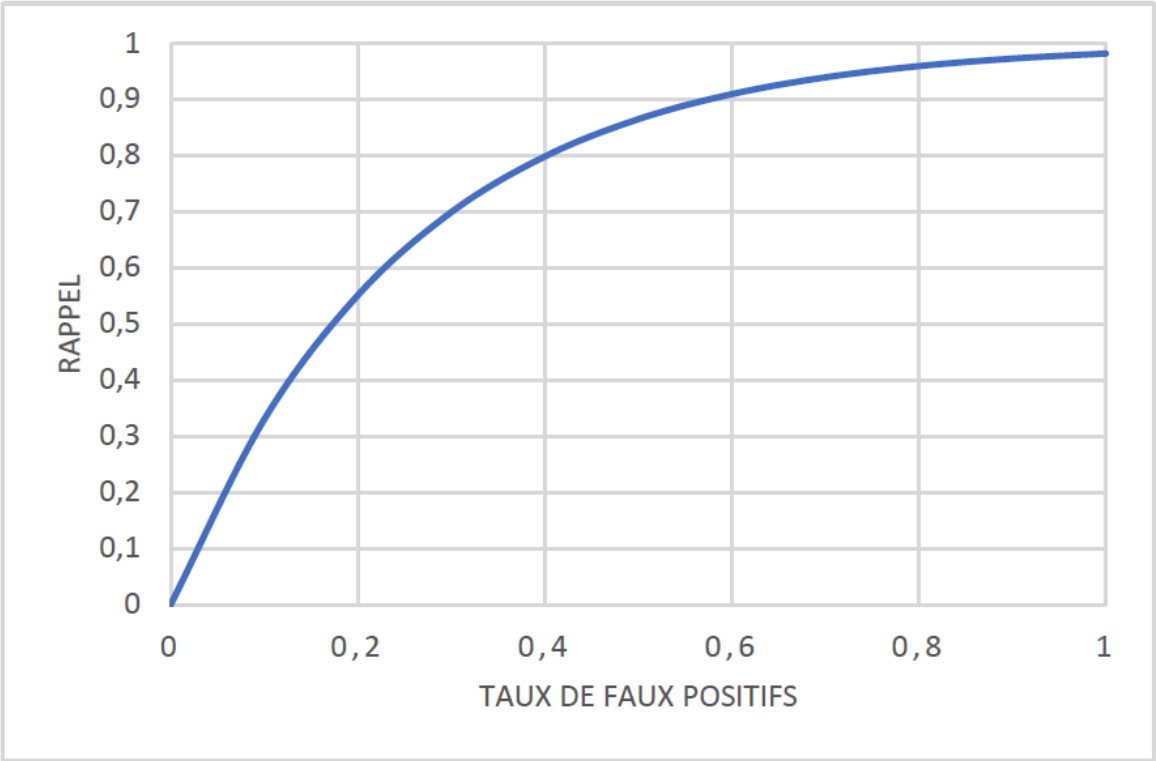

Figure 2.4 : Exemple de courbe ROC. © Charles Cousyn, 2022.

pour objectif de décrire les différents algorithmes pouvant être utilisés pour effectuer de la reconnaissance.

2.2.1 LES APPROCHES BASÉES SUR LES RÈGLES

Depuis quelques décennies, les chercheurs se sont penchés sur le problème du choix de l'algorithme de classification pour le tâche de reconnaissance d'activités humaines. La première idée des chercheurs a été le constat suivant : il suffit parfois de formuler une règle simple pour inférer de manière fiable de la réalisation ou non d'une activité. Par exemple, il semble naturel de dire que s'il est plus de 21 h, que toutes les lumières sont éteintes et que le résident se trouve dans la chambre, alors, il a de grandes chances d'être en train de dormir. Plus concrètement, l'exemple donné précédemment tente d'associer 3 éléments à l'activité *dormir dans son lit* : le temps, la position du résident et l'état de l'éclairage. Cette manière d'effectuer de la reconnaissance d'activités, par l'utilisation de règles d'association, qu'on nomme aussi la recherche de motifs, prend ses origines dans les systèmes experts présentés en 2.1. La différence majeure est, qu'aujourd'hui, les règles sont trouvées de manière automatique sans superviseur humain. Dans la littérature récente sur les approches basées sur les règles, on peut citer le travail de Wen *et al.* (2015) où les auteurs effectuent une recherche des motifs fréquents en ajoutant une notion de poids à chaque motif, le poids quantifiant à quelle point on peut faire confiance à ce motif.

Une chose importante à noter est qu'il est nécessaire de différencier la recherche de motifs classique de la recherche de motifs séquentielle. En effet, en reconnaissance d'activités, les données brutes fournies par les capteurs sont en réalité une série temporelle d'événements [8] ; et donc chaque donnée étant associée à un instant précis, on se retrouve à traiter une ou

8. Voici deux exemples d'événements : sachant $t_2 - t_1 = 1s$, , la lumière s'allume à l'instant t_1 et le résident est présent dans la cuisine à t_2

plusieurs séquences d'événements au lieu de chaque événement indépendamment. Introduite par Agrawal & Srikant (1995), la recherche de motifs séquentielle a, par exemple, été utilisée par Chikhaoui *et al.* (2011) pour réalisée une approche de reconnaissance d'activités en deux étapes : recherche de motifs séquentielle puis utilisation d'une fonction de correspondance entre les motifs fréquents extraits et les activités. Cette fonction de correspondance calcule en réalité la proportion de séquences d'événements correctement associés avec chaque modèle d'activité ; plus ce score est haut, plus il sera probable que l'activité est en cours de réalisation pendant la séquence d'événements.

2.2.2 LES APPROCHES PROBABILISTES

La seconde manière envisagée pour faire de la reconnaissance d'activités est d'utiliser des modèles probabilistes. En reconnaissance d'activités, les approches probabilistes peuvent être définies comme la recherche de l'activité maximisant la probabilité qu'une certaine activité soit en cours sachant les données. Mathématiquement, avec i l'indice d'une activité quelconque, a_i la $i$ème activité, $best$ étant l'indice de l'activité la plus probable et sensorData les données dont on dispose, cela peut se noter comme suit :

$$a_{best} = \underset{a_i}{\operatorname{argmax}} P(a_i|\text{sensorData}). \tag{2.7}$$

Tout le défi des approches probabilistes se trouve dans le calcul de cette probabilité. En effet, si l'on applique la loi de Bayes à cette probabilité, comme en (2.8), on remarque que le dénominateur est une constante par rapport à nos données. La conséquence de cela est que nous pouvons ignorer ce terme et nous concentrer sur le numérateur uniquement pour la recherche de a_{best}.

$$P(a_i|\text{sensorData}) = \frac{P(\text{sensorData}|a_i)\,P(a_i)}{P(\text{sensorData})} \tag{2.8}$$

La difficulté du calcul du terme $P(\text{sensorData}|a_i)\,P(a_i)$ a amené la communauté des chercheurs à poser des hypothèses simplificatrices. L'approche la plus connue posant une hypothèse simplificatrice est celle du classifieur bayésien naïf. L'hypothèse de ce modèle est que les données fournies à un instant donné sont indépendantes les unes des autres. Mathématiquement, en définissant sensorData comme un ensemble de n données provenant de n capteurs ($\text{sensorData} = (\text{sensorData}_1, \text{sensorData}_2, \ldots, \text{sensorData}_n)$), cela a pour conséquence :

$$\begin{aligned} P(\text{sensorData}|a_i) &= P(\text{sensorData}_1, \text{sensorData}_2, \ldots, \text{sensorData}_n|a_i) \\ &= \prod_{j=1}^{n} P\left(\text{sensorData}_j|a_i\right). \end{aligned} \tag{2.9}$$

Bien que facilitant le calcul de notre probabilité $P(a_i|\text{sensorData})$, il est évident que dans le monde réel, cette hypothèse est fausse dans la majorité des cas. En effet, dans la réalisation d'une activité, il est très probable que des événements soient amenés à être émis en même temps à cause d'un lien logique existant entre ces événements [9]. Autrement dit, la donnée émise par un capteur a de grandes chances de dépendre de celles fournies par d'autres capteurs. Malgré la critique évidente de cette hypothèse, les chercheurs ont tout de même tenté de construire des approches de reconnaissance basée sur elle. On peut, par exemple, citer

9. Par exemple, dans le cadre de l'activité *faire du thé*, il est fort probable que l'événement *présence dans la cuisine* s'active en même temps que l'événement *plaque chauffante allumée* dû au fait qu'il est très commun de se trouver dans la cuisine pour allumer sa plaque chauffante

les travaux de Kasteren & Krose (2007) dans lesquels les chercheurs ont construit plusieurs modèles sur cette hypothèse dont un modèle bayésien statique et réseau bayésien dynamique [10].

Parmi les modèles probabilistes les plus utilisés, on retrouve également les modèles graphiques comme les HMMs (*Hidden Markov Model*). Un HMM est un modèle graphique permettant de modéliser un processus supposé markovien contenant des états cachés. Une chaîne de Markov est une manière de représenter ce processus. Expliqué et illustré par Rabiner & Juang (1986), le principe du processus markovien est le suivant : *l'information utile pour la prédiction du futur est entièrement contenue dans l'état présent du processus et n'est pas dépendante des états antérieurs*, c'est l'hypothèse transitionnelle de Markov de premier ordre. Autrement dit, en reconnaissance d'activités, l'état à l'instant t ne dépend que de l'état à l'instant $t-1$. Plus concrètement et comme le dit Kim *et al.* (2010), un HMM est un modèle probabiliste qui est utilisé pour générer des états cachés (les activités dans notre cas) à partir de données observables (les événements dans notre cas). Son objectif principal est de déterminer la séquence d'états cachés $y = (y_1, y_2, ..., y_t)$ la plus probable au vue de la séquence observée $x = (x_1, x_2, ..., x_t)$. Un HMM est un modèle faisant deux hypothèses. La première, comme nous l'avons déjà mentionné, est l'hypothèse transitionnelle de Markov de premier ordre :

$$P(y_t|y_1, y_2, y_3, ..., y_{t-1}) = P(y_t|y_{t-1}). \tag{2.10}$$

La seconde est que les états observés sont indépendants conditionnellement les uns des autres, se traduisant mathématiquement par :

$$P(x_t|y_t, x_1, x_2, x_3, ..., x_{t-1}, y_1, y_2, y_3, ..., y_{t-1}) = P(x_t|y_t). \tag{2.11}$$

10. Un modèle bayésien statique est un modèle qui fait l'hypothèse que l'activité en cours à un instant est indépendante des activités en cours aux instants précédents, soit a_t indépendante de $a_{1:t-1}$ alors qu'un modèle dynamique possède cette dépendance entre a_t et $a_{1:t-1}$

Dit autrement, la variable observable au temps t, x_t dépend uniquement de l'état caché actuel y_t, la probabilité d'observer x pendant l'état caché y est indépendante de toutes les autres variables observables et des états passés. Pour trouver la séquence d'états cachés y_{best} la plus probable à partir d'une séquence observée x, on cherchera la séquence d'états y qui maximise une probabilité conjointe $P(x,y)$ qui est le produit de la probabilité de transition $P(y|y_{t-1})$ et la probabilité d'observation $P(x_t|y_t)$, c'est-à-dire la probabilité que le résultat x_t soit observé dans l'état y_t :

$$P(x,y) = \prod_{t \in T} P(y|y_{t-1}) \times P(x_t|y_t) \tag{2.12}$$

où

T est l'ensemble des indices des instants consécutifs considérés.

La Figure 2.5 illustre un HMM utilisé pour faire de la reconnaissance d'activité. Parmi les travaux utilisant ce type de modèle, on peut citer les travaux de Kim *et al.* (2010, 2016) dans lesquels les auteurs ont créé une approche utilisant plusieurs HMM à travers une méthode d'ensemble (Dietterich, 2000). On peut également citer les travaux de Kolekar & Dash (2017) dans lesquels une approche de *Vision-based Human Activity Recognition* a été conçue en utilisant un HMM entraîné avec des caractéristiques segmentées.

2.2.3 LES AUTRES APPROCHES

Pour finir, il existe un grand nombre de méthodes d'apprentissage automatique pouvant permettre d'effectuer de la reconnaissance d'activités, mais ne rentrant pas dans les catégories des règles d'inférences et des modèles probabilistes. Parmi ces méthodes, on peut remarquer

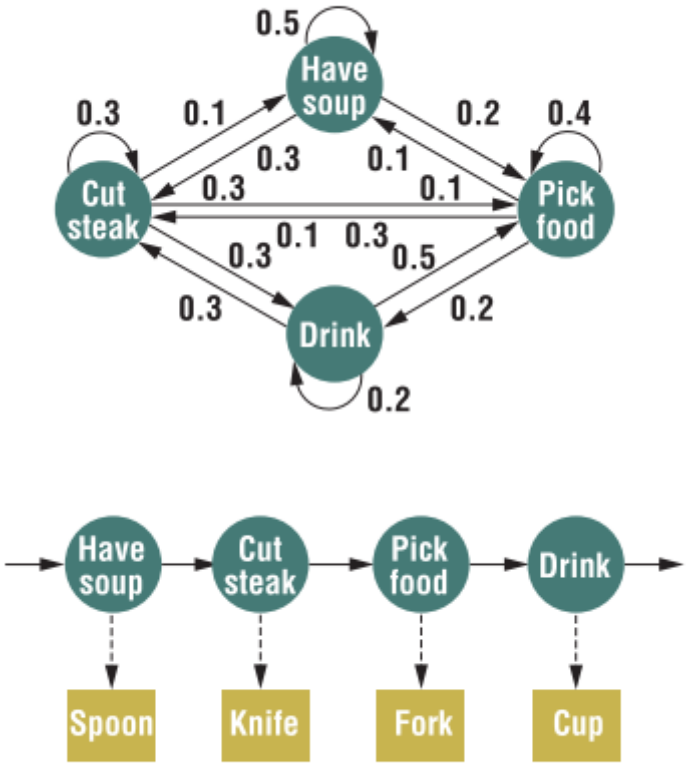

Figure 2.5 : Exemple de HMM pour de la reconnaissance d'activité : les éléments jaunes désignent les événements observables, les éléments en bleu désignent les états cachés. Les flèches en trait continu représentent les probabilités transitionnelles $P(y_t|y_{t-1})$ et les flèches en trait pointillé représentent les probabilités d'émission $P(x_t|y_t)$ (ici elles valent toutes 1.0) (Kim *et al.*, 2010). © 2010 IEEE.

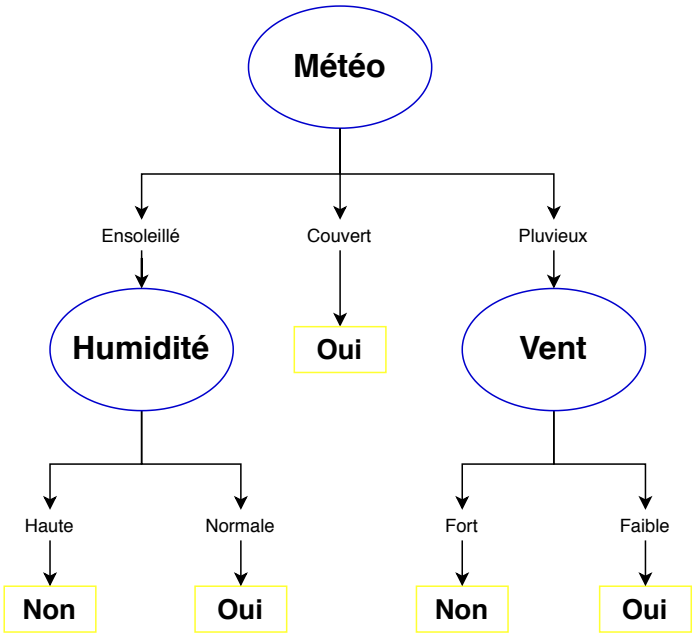

Figure 2.6 : Exemple d'arbre de décision pour un problème de classification à 2 classes : Oui et Non. Image Creative Common.

que chacune d'entre elles utilise un paradigme bien à elle, c'est-à-dire qu'elles sont capables de résoudre le problème selon une certaine vision du monde. On peut, par exemple, citer les méthodes basées sur les arbres de décisions qui tente de résoudre le problème de la reconnaissance d'activités par la création d'un arbre dont chaque branche représente une prise de décision en fonction d'un ou plusieurs critères et dont chaque feuille représente la décision finale (l'activité détectée). Dans cette vision du monde, le but est de créer l'arbre permettant de créer un ensemble de conditions logique discriminant les classes entre elles avec le moins d'incertitude possible. La Figure 2.6 représente un exemple d'arbre de décision bien connu permettant de savoir si l'on devrait aller jouer au tennis en fonction de 3 critères : la météo, l'humidité et le vent.

Les arbres de décision, même s'ils ne représentent qu'une petite portion des algorithmes d'apprentissage automatique, ont été très largement utilisés par les chercheurs en reconnais-

sance d'activités humaines. On peut d'abord citer les travaux de Bouchard *et al.* (2011a,b) dans lesquels les auteurs adaptent un algorithme de création d'arbres de décision (C4.5 par Quinlan (2014)) en tentant de considérer les relations spatiales des objets utilisés pendant la réalisation d'une activité. On peut illustrer les relations spatiales de la manière suivante : imaginons qu'un résident vient d'exécuter une certaine action appelée *faire bouillir l'eau*. Disons que cette observation peut nous conduire à deux activités plausibles, qui sont *préparer une tasse de café* et *cuisiner des pâtes*. En considérant les relations spatiales entre les objets, nous pouvons détecter qu'une tasse est présente dans la zone d'activité, alors qu'il n'y a pas de boîte de pâtes à proximité. Par conséquent, l'hypothèse de la cuisson des pâtes pourrait être éliminée. Ces relations fonctionnent comme un filtre avant le processus de reconnaissance et cela permet de réduire le nombre d'hypothèses à considérer avant d'utiliser un arbre de décision. Si l'on se place dans le cadre d'une approche basée sur les *wearable devices*, on peut citer les travaux de Lara & Labrador (2012); Fan *et al.* (2013) qui utilisent des données comme le rythme cardiaque, la fréquence respiratoire, l'amplitude de la respiration, la température de la peau, la posture, l'amplitude de l'électrocardiogramme et l'accélération 3D pour construire un arbre de décision et faire de la reconnaissance d'activités.

Une autre approche est l'utilisation de réseaux de neurones. Les réseaux de neurones sont des modèles mathématiques basés sur le fonctionnement des neurones biologiques ayant montré de bons résultats pour effectuer un grand nombre de tâches. En ce qui concerne les travaux existants, on peut citer le travail de Ronao & Cho (2016) qui utilisent les données provenant de l'accéléromètre et du gyroscope de smartphones comme données d'apprentissage. Leur méthode consiste en l'utilisation d'un type particulier de réseaux de neurones, un réseau de neurones convolutif pour reconnaître 6 activités avec une justesse pouvant atteindre 95.75%.

2.3 CONCLUSION

Dans ce chapitre, nous avons présenté et décrit la notion d'apprentissage automatique ainsi que ses étapes constituantes, à savoir, le prétraitement, l'extraction de caractéristiques et la phase d'apprentissage. Nous avons également vu qu'il existe différents types d'apprentissages dépendamment du problème à résoudre et du fait de disposer de données étiquetées ou non, à savoir l'apprentissage supervisé et l'apprentissage non supervisé. Nous avons ensuite présenté la manière d'évaluer les performances pour les approches de reconnaissance d'activités avec de la classification. Nous avons vu qu'il existe plusieurs mesures de performances pouvant être utilisées pour les problèmes de classification comme la justesse, la précision, le rappel, le F-score, le kappa de Cohen et l'aire sous la courbe ROC ; chacune d'entre elles tente de représenter la qualité d'un classifieur tentant de séparer les classes entre elles. Dans la seconde section, nous avons ensuite présenté les différentes approches pour effectuer de la reconnaissance d'activités humaines. Les approches basées sur les règles partent du principe que reconnaître une activité peut se faire à travers l'utilisation d'un ensemble de règles d'inférence pouvant être déterminées automatiquement. Les approches probabilistes, elles, voient le problème sous le prisme de la probabilité conditionnelle d'une activité sachant l'existence de données provenant d'ensemble de capteurs ($P(a_i|\text{sensorData})$). Il existe également d'autres approches ne rentrant pas dans les catégories précédentes comme les arbres de décision et les réseaux de neurones ; ces approches ont été brièvement présentées. Enfin, nous avons aussi discuté des apports des travaux utilisant ces approches.

La tâche de reconnaissance d'activité anime la recherche depuis plusieurs décennies. Vu sous l'angle de l'apprentissage automatique, il existe de nombreuses manières de la concevoir. Aucune d'entre elles ne semble se démarquer et la recherche est encore très active à ce jour. Après avoir présenté le problème de la reconnaissance d'activité et de l'apprentissage, cette thèse tentant d'utiliser le web pour contribuer à l'amélioration des approches existantes, le

chapitre suivant a pour rôle de présenter le potentiel du web pour la tâche de reconnaissance d'activités humaines ainsi que les travaux existants sur le sujet.

CHAPITRE III
L'USAGE DU WEB DANS LA RECONNAISSANCE D'ACTIVITÉS

Le chapitre précédent présentait le processus de la reconnaissance d'activités ainsi que les différentes approches qui peuvent être utilisées pour résoudre le problème en se basant sur la littérature scientifique existante. Cette thèse s'articulant sur l'apport du web au problème de la reconnaissance d'activités, le présent chapitre précisera nos espérances quant au potentiel du web pour ce domaine et présentera les travaux qui ont tenté d'exploiter ce potentiel ainsi que les approches envisagées dans cette thèse.

3.1 LE POTENTIEL DU WEB POUR LA RECONNAISSANCE D'ACTIVITÉS

Dans la section 1.4, nous avons montré que certains éléments semblent nous indiquer que le web a un véritable potentiel pour aider à résoudre le problème de la reconnaissance des activités de la vie quotidienne. En effet, le web semble regorger d'informations sur le quotidien de l'humanité sous différentes formes. Ces éléments sont l'existence de sites web descriptifs sur les activités et la possibilité d'obtenir des images contenant les objets liés aux activités par l'utilisation d'un moteur de recherche d'images. Cette thèse s'appuie sur l'hypothèse que ces éléments soient suffisants pour obtenir une bonne représentation des activités de la vie quotidienne ; hypothèse dont la véracité sera testée tout au long du projet.

Une autre hypothèse concernant l'utilisation du web pour la reconnaissance d'activités est la capacité à obtenir une représentation des activités de manière automatique. En effet, si l'on considère la reconnaissance d'activités comme un problème de classification, l'automatisation pourrait apporter plusieurs avantages comparativement aux approches d'apprentissages

plus classiques. Pour bien comprendre cela, il est nécessaire de discuter du processus utilisé par les chercheurs effectuant de l'apprentissage automatisé pour la reconnaissance d'activités.

Toute méthode usant d'apprentissage automatique nécessite l'utilisation de données représentatives du problème à résoudre. En reconnaissance d'activités, ces données sont, la plupart du temps, issues du monde réel et fournies par des capteurs installés dans l'environnement de vie d'un résident ; permettant d'en avoir une vision partielle. Cependant, cette description théorique des données d'apprentissage ne permet pas de rendre compte de certaines considérations pratiques. En effet, d'un point de vue plus pragmatique, afin de récolter des données d'apprentissage, les chercheurs doivent organiser des expérimentations. En reconnaissance d'activités, cela consiste à rechercher des individus acceptant de réaliser des activités dans un environnement de vie donné ; cet environnement de vie étant équipé d'une multitude de capteurs qui vont produire des données liées à ce qui se déroule dans l'environnement. Cette manière de procéder, outre le fait de devoir trouver des individus acceptant de participer aux expérimentations, présente plusieurs inconvénients. Premièrement, on ne peut pas mettre de côté les considérations éthiques des expérimentations. En effet, en plus d'accepter que des données (permettant potentiellement d'identifier le participant) les concernant soient fournies à l'équipe de chercheurs, il est nécessaire de vérifier que l'expérimentation ne met pas en péril la sécurité du participant ; sans oublier que cette expérimentation ne devrait pas favoriser la moindre discrimination. Deuxièmement, le processus de collecte de données peut s'avérer long, car il arrive souvent que l'on demande au participant de répéter plusieurs fois la même activité afin d'avoir plus de données. Si l'on multiplie ça par le nombre d'activités à reconnaître et le nombre de participants, et que l'on ajoute les imprévus sur la disponibilité des chercheurs et des participants, on se retrouve avec des expérimentations pouvant durer plusieurs semaines voire plusieurs mois. Enfin, dans un but de gagner du temps sur cette collecte de données, même s'il est possible de considérer les approches simulant des habitats intelligents avec des

programmes informatiques, une simulation génère des données artificielles pouvant ne pas représenter correctement la réalité. Et ce problème de représentativité peut alors engendrer une mauvaise généralisation [11] par les approches d'apprentissage automatique.

L'automatisation peut être définie comme l'exécution totale ou partielle de tâches techniques par des machines fonctionnant sans intervention humaine. Ainsi, la possibilité d'automatisation représente une solution très pertinente aux problèmes cités plus haut. En effet, si la tâche de chercher des informations représentatives sur la réalisation d'activités de la vie quotidienne peut se faire sans impliquer d'humains, alors, il sera possible d'obtenir ces informations pour un grand nombre d'activités en un temps bien plus court qu'en menant des expérimentations.

3.1.1 RESSOURCES EN LIEN AVEC LES ACTIVITÉS DE LA VIE QUOTIDIENNE DISPONIBLES SUR LE WEB

L'un des principaux éléments permettant de savoir ce que le web pourrait apporter aux approches de reconnaissance d'activités est de prendre connaissance des ressources en lien avec les activités de la vie quotidienne disponibles sur le web. Certains éléments de réponse ont été fournis en section 1.4. Nous avons pu constater que grâce à la popularité et à la démocratisation du web, il peut être considéré comme un recueil de données très pertinent pour obtenir des informations sur les activités de la vie quotidienne, en contenant une grande diversité de ressources disponibles. Un bon exemple de cela est qu'il a été montré que moins de la moitié des recherches effectuées sur le moteur de recherche Google résulte en un clic (Fishkin, 2019). Cela montre à quel point le web, et en particulier ce moteur de recherche, de

11. La généralisation, en apprentissage automatique, désigne la capacité d'un modèle d'apprentissage à s'adapter correctement à de nouvelles données qui n'étaient pas visibles au préalable lors de la phase d'apprentissage

nos jours, répond à nos besoins en nous affichant la réponse à nos questionnements dans la liste des résultats fournie sans même que nous ayons besoin de cliquer.

Cette sous-section illustre un premier aspect qualitatif des différents types de ressources disponibles sur le web en lien avec le domaine de la reconnaissance d'activités. Cependant, cela ne veut pas dire que chacune de ces ressources devrait, dans l'absolu, être exploitée. Pour le bien de cette thèse, il est important d'étudier l'exploitabilité de ces ressources en fonction de bons critères. On peut, raisonnablement, estimer que ces critères sont essentiellement les suivants :

— la suffisance de la quantité disponible de médias correspondant au type de ressource

— la suffisance de présence d'informations pour la reconnaissance d'activités humaines dans le type de ressource

— la capacité d'extraire l'information voulue contenue dans le type de ressource

— la capacité d'automatisation de la collecte du type de ressource

Les deux premiers critères sont abordés dans la présente sous-section, le troisième le sera dans la sous-section 3.1.2 et le quatrième dans la sous-section 3.1.3.

Premièrement, il existe des ressources textuelles sur les activités de la vie quotidienne. Ces informations sont contenues sous forme de titres et de paragraphes présents sur certaines pages web tel qu'illustré dans les Figures 1.2 et 1.3. Nous qualifions ces pages comme des pages web descriptives, car elles fournissent une description par le biais de différents types d'informations sur la réalisation des activités :

— des informations temporelles de par l'ordre dans lequel les paragraphes sont écrits et les connecteurs logiques utilisés (*ensuite*, *après avoir*, *avant de*, *deuxièmement*, etc.)

— des informations spatiales de par les lieux et entités mentionnées (*salon*, *sur la table*, *dans le jardin*, *dans la véranda*, etc.)

— des informations sur les objets et substances impliquées (*eau*, *beurre*, *ordinateur*, *réfrigérateur*, etc.).

Le texte étant historiquement le premier type de ressource disponible sur le web et au vu de la facilité avec laquelle il est possible de trouver du texte concernant la réalisation d'une activité (voir la section 1.4), notre équipe de recherche a fait le choix de considérer que le critère de quantité disponible de données textuelles (premier critère) est valide, a priori. La validité de ce critère pourra être discuter au besoin dans cette thèse.

Deuxièmement, il est possible d'obtenir des images liées aux activités en utilisant les moteurs de recherche d'images comme illustré à la Figure 1.4. En utilisant une requête avec les titres de différentes activités, il semble clair que les images contiennent des informations que nous, humains, pouvons percevoir sans effort :

— des informations spatiales de par les lieux représentées sur les images

— des informations sur les objets et substances impliquées de par les éléments présents sur l'image.

L'image est un type de ressources très populaire sur le web, il est possible de trouver des images en rapport avec tous les sujets. Notre équipe de recherche a effectué une préétude en termes de quantité d'images disponibles pour des requêtes en rapport avec des activités de la vie quotidienne (*make tea*, *cook pasta*, etc.) sur les principaux moteurs de recherche d'image. D'après cette dernière préétude, il semblerait qu'il existe, a priori suffisamment d'images en lien avec les activités de la vie quotidienne. Bien sûr, même si cette affirmation sera plus profondément vérifiée, la préétude visait à savoir si l'image, comme type de ressource, présente du potentiel. Nous montrerons que cela semble être le cas.

Ensuite, malgré le fait que cela soit moins populaire que le texte et les images, il est possible d'obtenir un autre type de ressource sur le web, à savoir le son. En effet, il existe

quelques moteurs de recherche de sons comme http://www.findsounds.com permettant de trouver des sons ambiants facilement identifiables par l'oreille humaine. Les sons peuvent nous fournir :

— des informations temporelles de par l'ordre dans lequel les sons sont enregistrés dans un extrait sonore

— des informations spatiales de par les sons ambiants existants (chant d'oiseau pour le jardin, etc.)

— des informations sur les objets et substances impliqués de par les sons caractéristiques présents (bruit de frottement d'une éponge sur de la vaisselle, bruit d'un aspirateur, bruit de machine à laver, etc.).

En termes de quantité, après une préétude mené par notre équipe de recherche, il semblerait qu'il soit difficile de trouver une quantité suffisante de sons en rapport avec des activités de la vie quotidienne. Le moteur de recherche http://www.findsounds.com que nous avons utilisé pour cette préétude semble fournir des résultats de qualité très inégale. Parfois, on se retrouve avec un nombre très limité de résultats pour certaines activités, ces résultats semblant être de bonne qualité. D'autre fois, on se retrouve avec un très grand nombre de résultats dont la pertinence ne semble pas la meilleure. Les moteurs de recherches de sons étant bien moins populaires que ceux d'images et de page web, notre avons établi que le critère de suffisance de la quantité disponible de médias correspondant au son n'est pas satisfait.

Enfin, sur le web de notre décennie, en plus du texte, des images et du son, nous sommes capables de consulter un dernier type de ressources, les vidéos. En effet, on peut remarquer que tout comme il existe des moteurs de recherche d'images, il existe également des moteurs de recherche de vidéos comme Youtube [12] et Google [13] pouvant fournir des informations sur

12. https://www.youtube.com/results?search_query=faire+du+thé

13. https://www.google.com/search?q=faire+du+thé&tbm=vid

les activités de la vie quotidienne. En supposant que les résultats des moteurs de recherche d'images sont aussi bons que ceux de vidéos, alors la vidéo peut apporter les mêmes types d'informations que les images, mais fournit, en plus, des informations temporelles, car une vidéo peut être vue comme une séquence ordonnée d'images et de sons. En ce qui concerne la quantité de vidéos disponibles en rapport avec des activités de la vie quotidienne, la préétude effectuée semble aller dans le même sens que pour les images. En d'autres mots, les moteurs de recherche de vidéos semblent, a priori, fournir des résultats de bonne qualité et en quantité suffisante.

3.1.2 IDENTIFICATION ET EXTRACTION DES INFORMATIONS PERTINENTES SUR LES ACTIVITÉS À PARTIR DU WEB

Dans la sous-section précédente, nous avons vu qu'il existait quatre types de ressources sur le web ayant le potentiel d'aider à reconnaître les activités de la vie quotidienne : le texte, les images, les sons et les vidéos. Nous avons également estimé le potentiel de ces ressources en termes de quantité disponible pour la reconnaissance d'activités et avons estimé que le son ne satisfait pas ce critère. Chaque type de ressource contient de l'information latente qui ne peut être extraite que sous la condition d'utiliser une approche adaptée à ce type de ressource. Cette sous-section a donc pour but de déterminer les types de ressources exploitables et de décrire les approches existantes pour les exploiter au mieux.

EXPLOITATION DU TEXTE

On peut définir le texte comme un ensemble de mots ou de lettres qui sont organisés pour permettre la compréhension par le lecteur d'un propos, d'une idée, d'un objet, etc. Depuis que l'écriture a été inventée, le texte fait partie des moyens de communication entre humains

les plus utilisés au monde. L'histoire humaine a fait qu'il existe des textes de toutes les natures imaginables (religieux, scientifique, fiction, journalisme, conversations privées, etc.) afin de transmettre une ou plusieurs informations aux lecteurs. Nous, humains, sommes habitués à ce moyen de communiquer grâce à la combinaison de nos capacités cognitives inhérentes à notre espèce et à l'éducation dont nous bénéficions dès notre plus jeune âge.

Cependant, d'un point de vue informatique, le texte est, la majorité du temps, représenté sous la forme d'une chaîne. Une chaîne est traditionnellement une séquence de caractères. Dans les langages informatiques, elle est généralement considérée comme un type de données et est souvent implémentée comme une structure de données en tableau d'octets (ou de mots) qui stocke une séquence d'éléments, typiquement des caractères, en utilisant un certain codage de caractères (ASCII (Association, 1963) par exemple). Une chaîne peut également désigner des tableaux plus généraux ou d'autres types et structures de données en séquence (ou en liste).

Dans le cas général, en informatique, le format par défaut du texte est la chaîne de caractères. Ce format a pour avantage d'être très facile à utiliser, c'est la forme initiale de tout texte en informatique. Cependant, il faut bien comprendre que sans transformations supplémentaires, il est difficile d'en tirer beaucoup d'informations à partir d'un ordinateur. Prenons un exemple pour l'illustrer. Imaginons que nous disposons d'un texte contenant deux phrases : "Le chat chasse désespérément une souris pour la dévorer." et "J'ai acheté un clavier et une souris pour mon ordinateur.". Imaginons que nous souhaitons synthétiser notre texte sous forme de prédicats logiques [14] ; cela donnerait les prédicats donnés dans le groupe d'équations (3.1).

$$\text{Chasser(chat, sourisAnimal)} \tag{3.1a}$$

14. Nos prédicats sont représentés sous la forme : Relation(sujet, objet1, objet2, ..., objetN)

Acheter(je, clavier, sourisObjet) (3.1b)

Pour passer de sa forme initiale à celle des prédicats, nous avons plusieurs problèmes à surmonter. Premièrement, a priori, un ordinateur n'est pas capable de distinguer les deux sens du mot *souris*, du moins quand on ne considère pas le contexte. En effet, quand on regarde la seconde phrase, le mot *clavier* nous donne un indice sur le sens du mot *souris* augmentant drastiquement les chances que l'on parle de l'objet et non de l'animal. Cela implique donc que notre programme informatique ait connaissance des deux concepts distincts malgré une même forme en chaîne de caractères. Deuxièmement, notre programme informatique doit être capable d'identifier les informations grammaticales correspondantes à chacun des mots de nos phrases comme le genre, le nombre, la fonction (verbe, nom, groupe nominal, etc.), etc. On nomme ce processus l'étiquetage morphosyntaxique ou *part-of-speech tagging* (*POS tagging*)(Charniak, 1997). Cette étape est cruciale pour connaître le rôle de chaque mot ou groupe de mots dans la phrase afin de pouvoir créer correctement nos prédicats. Enfin, le programme informatique devrait être capable de mettre en relation les synonymes à travers un même concept afin de pouvoir déterminer les relations entre prédicats. Dans notre exemple, le mot *J'* doit être associé au concept de *je*, tout comme les mots *je* et *Je* devrait l'être aussi (un mot avec ou sans majuscule n'est pas le même mot, en tous cas, au sens de la chaîne de caractères).

L'extraction d'informations à partir du texte est une discipline faisant partie d'un plus grand domaine nommé la fouille de texte. Elle se définit par l'ensemble des méthodes, des techniques et des outils développés pour exploiter les documents non structurés (Venkata *et al.*, 2016; Cousyn, 2018). Elle peut avoir de nombreux objectifs comme l'extraction de sujets, la classification de texte (Agarwal & Mittal, 2014), la reconnaissance d'entités nommées

(Shaalan, 2007), la synthèse automatique (Gambhir & Gupta, 2017), l'extraction de relations entre des concepts (Bach & Badaskar, 2007), la segmentation de textes (Koshorek *et al.*, 2018), etc.

EXPLOITATION DES IMAGES

Une image est un élément qui dépeint la perception visuelle d'un objet, d'une situation, d'un lieu ou de tout autre chose représentable visuellement. Cette représentation visuelle fait office d'entrée pour notre cerveau afin d'effectuer des raisonnements. Dans le cadre de l'informatique, une image est une amplitude distribuée de couleurs qui peut être encodée dans différents formats standardisés (JPEG, GIF, PNG, BMP, WEBP, etc.). Les fichiers d'images sont composés de données numériques dans l'un de ces formats afin que les données puissent être pixellisées pour être utilisées sur un écran d'ordinateur ou une imprimante. La pixellisation est le processus convertissant les données d'une image en une grille de pixels. Chaque pixel a un certain nombre de bits pour désigner sa couleur (et dans certains formats, sa transparence). Un exemple simple est le cas où nous avons une image en nuances de gris où chaque pixel est encodé dans un octet (valeur entre 0 et 255). La Figure 3.1 illustre la manière dont une telle image peut être encodée sous la forme d'une matrice.

On comprend bien alors que d'un point de vue informatique, une image n'est qu'un ensemble de valeurs numériques. Donc, à première vue, il n'est pas évident que l'on puisse déterminer facilement ce que l'image représente ainsi que tous les éléments qu'elle contient. Par exemple, le fait qu'il y ait un *S* dans la matrice de la Figure 3.1 est visible à l'œil nu en cherchant les valeurs inférieures à 100, mais cette lettre que nous voyons est le produit de notre cerveau. Ce qui nous intéresse, c'est la manière d'obtenir l'information de la présence de la lettre *S* en utilisant les capacités calculatoires d'un ordinateur.

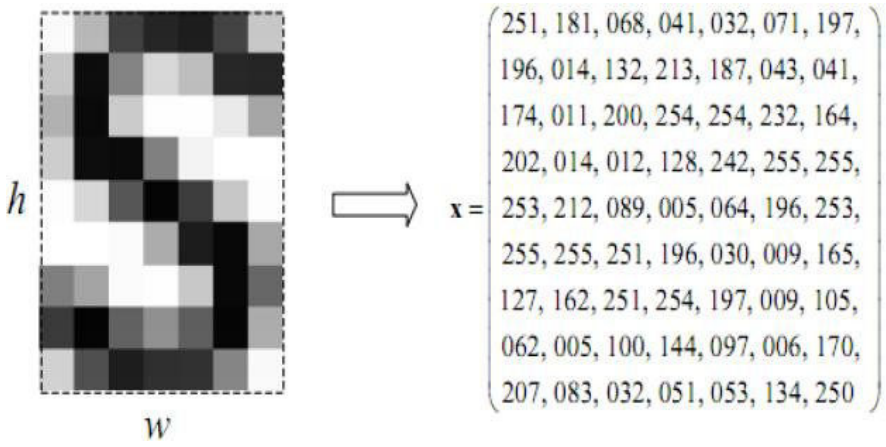

Figure 3.1 : Exemple de représentation d'une image en nuance de gris sous la forme d'une matrice d'octets (Miah *et al.*, 2014). Image Creative Commons.

Cette difficulté de l'interprétation automatique d'images a été le sujet de nombreux travaux de recherche scientifique dans un domaine appelé la vision par ordinateur. Ce domaine représente la discipline scientifique qui s'intéresse à la théorie des systèmes artificiels qui extraient des informations des images. On peut également le voir comme un domaine scientifique combinant informatique et mathématique traitant de la manière dont les ordinateurs peuvent être conçus pour acquérir une compréhension de haut niveau à partir d'images ou de vidéos numériques. Il existe de nombreux sous-domaines de la vision par ordinateur (Szeliski, 2010) comme la classification d'images (Scherer, 2020), reconstitution de scène (Tomasi & Kanade, 1992), la détection d'objet (Ganesh, 2019), l'estimation de mouvement (Wikipedia, 2020), etc.

Durant la dernière décennie, avec le regain en popularité des réseaux de neurones dans la littérature scientifique, de très grands progrès en termes de détection d'objets et de classification d'images ont été effectués notamment grâce à des initiatives comme ImageNet (Deng *et al.*, 2010) consistant en la création d'une large base de données d'images annotées par des humains. Ce projet a permis la mise à disposition pour la communauté scientifique d'un gigantesque jeu de données avec lequel il a été possible de faire de l'apprentissage

automatique. Un autre facteur déterminant en ce qui concerne les performances en détection d'objets a été les progrès effectués dans le domaine des réseaux de neurones, notamment le gain en popularité de réseaux de neurones spécialisés dans le traitement des images, les réseaux de neurones convolutifs ou *Convolutional Neural Network* (LeCun *et al.*, 1989) ou CNN.

EXPLOITATION DES VIDÉOS

On peut définir une vidéo comme un support électronique pour l'enregistrement, la copie, la lecture, la diffusion et l'affichage d'images animées. D'un point de vue informatique, la représentation commune d'une vidéo est une série d'images espacées d'un pas de temps fixe (fonction du nombre d'images par seconde de la vidéo). Tout comme les images, il existe différents formats standardisés (AVI, MP4, MKV, WMV, etc.).

Les challenges pour exploiter les vidéos sont majoritairement les mêmes que pour les images. Il y a toutefois au moins un défi supplémentaire. Les fichiers vidéo peuvent être très volumineux. La taille de ces fichiers est synonyme d'un traitement plus long et d'un besoin de stockage massif si l'on souhaite utiliser un grand nombre de vidéos provenant du web. En conséquence, l'exploitation de vidéos n'est pas le sujet de cette thèse pour le moment, car notre équipe de recherche estime que l'exploitation du texte et des images représentent le meilleur compromis facilité d'exploitation/quantité d'informations latentes disponibles.

INFORMATIONS PERTINENTES

Dans cette sous-section, nous avons tenté de jauger l'exploitabilité des différents types de ressources dont nous disposons sur le web et avons fait le choix de nous focaliser sur

l'exploitation du texte et des images. Et, comme nous l'avons vu dans la sous-section 3.1.1, chacune de ces ressources est en mesure de fournir certains types d'informations sur les activités de la vie quotidienne que l'on peut résumer dans la liste suivante :

— des informations temporelles

— des informations spatiales sur les lieux et entités impliqués dans la réalisation d'une activité

— des informations sur les objets et substances impliqués.

De là, on peut se questionner sur la pertinence de disposer de chacun de ces types d'information. En effet, en dehors du fait que détenir ces trois types d'informations permet de constituer une base de connaissance utile sur les activités de la vie quotidienne, certains travaux ont été capables d'effectuer la tâche de reconnaissance sans disposer de l'ensemble des trois types. Un exemple de cela est le travail de Palmes *et al.* (2010). Dans ce dernier, pour chaque activité, les auteurs utilisent le contenu textuel de pages web pour obtenir une liste d'objets et de substances ayant un poids associé, ce poids faisant office de score de pertinence pour l'activité en question. Pour effectuer de la reconnaissance d'activités humaines, ils ont mis au point une approche dont le principe est le suivant : pour chaque activité, on ne considère que l'objet ou la substance ayant le poids associé le plus élevé, appelé objet clé. La logique sera alors que la présence de cet objet dans une séquence d'événements signifie que l'activité associée est en cours. On peut alors faire la remarque que l'on a pu effectuer de la reconnaissance sans disposer de la moindre information concernant le lieu, la disposition de l'espace, ni l'ordre d'utilisation des entités. Ce travail n'est qu'un exemple, car il existe d'autres travaux tentant d'effectuer de la reconnaissance d'activités humaines à partir du web sans utiliser les trois types d'informations, ces travaux seront présentés dans la section 3.2.

Cette thèse souhaite contribuer à l'extraction automatique d'informations pertinentes et représentatives sur les activités de la vie quotidienne. À ce titre, notre objectif sera d'abord

d'extraire le plus d'informations de qualité possible du web à partir du texte et des images. L'objectif suivant consiste en l'utilisation de ces informations pour effectuer de la reconnaissance d'activités. Nous tenterons donc d'extraire les trois types d'informations mentionnées plus haut puis nous essayerons de toutes les mettre à profit pour la tâche de reconnaissance et nous discuterons de leur pertinence suite à cela.

3.1.3 LA RÉCUPÉRATION AUTOMATIQUE DES RESSOURCES DU WEB

Dans la sous-section précédente, nous nous sommes penchés sur le type de ressources présentes sur le web à utiliser et sur les informations pertinentes à extraire. Ainsi, elle représente une analyse théorique du contenu du web pour le domaine de la reconnaissance d'activités nous permettant de nous faire une meilleure idée de ce que le web pourrait apporter. Cependant, elle ne considère pas certains aspects pratiques du projet comme la manière concrète de récupérer automatiquement nos ressources. De ce fait, la présente sous-section s'occupera de présenter la manière de concevoir une approche automatique pour la récupération des ressources du web.

La manière de récupérer automatiquement une ressource quelconque sur le web se nomme le *web scraping* (Perez, 2019). On peut définir le *web scraping* comme un cas particulier du *data scraping* utilisé pour l'extraction des données des sites web où le *data scraping* est l'ensemble des techniques par laquelle un programme informatique extrait des données d'une sortie lisible par l'homme provenant d'un autre programme. Le but d'un *web scraper* est généralement de récupérer des informations dans une page web, pour les utiliser dans un but différent. On peut, par exemple, imaginer un programme informatique dont le but est de trouver et de copier les noms, les numéros de téléphone et les URLs des sites web des entreprises d'un domaine particulier dans une liste. Un point important concernant le *web scraping* est que même si les pages web sont construites à l'aide de langages structurés

(HTML, XML, JSON, etc.) et peuvent contenir un grand nombre d'informations très utiles sous forme de textes, d'images, de sons ou de vidéos, la plupart des pages web sont conçues pour des utilisateurs finaux humains et non pour faciliter l'utilisation automatisée. C'est le cas des pages web exécutant du code JavaScript (Mozilla Developer Network, 2020) dont le contenu varie en fonction des interactions de l'utilisateur humain. Un autre point est que, la plupart du temps, une approche de *web scraping* est spécifique à un site web en particulier. La raison à cela est que chaque concepteur de site web est entièrement libre d'utiliser la structure qu'il veut pour représenter l'information. Il y a bien sûr des standards sur les langages utilisés (HTML, CSS, Javascript), mais il faut bien comprendre que les navigateurs sont faits pour interpréter du code HTML pour afficher le contenu des pages web et que le langage HTML est un langage qui apporte une structure, mais que cette structure est variable grâce au concept de balisage (Pkdoorn, 2015). Un bon exemple d'outil de *web scraping* est la librairie *Beautiful Soup* développée par Richardson (2020). Codée en Python, elle est une librairie d'analyse syntaxique (*parsing*) de documents HTML et XML. Cette analyse syntaxique permet de produire un arbre syntaxique qui peut être utilisé pour chercher ou modifier facilement des éléments présents dans le fichier. À titre d'exemple, si l'on est dans le cas d'une page en HTML, si l'on cherche le titre de la page ou le texte inclus dans les balises de type <*a*>, *Beautiful Soup* nous permet de le faire en quelques lignes de codes.

Cependant, certains sites web peuvent faire le choix de faciliter grandement la capacité d'automatisation de l'utilisation de leurs pages. Ce choix se fait par la création d'une API ou Interface de programmation. Une API, comme son nom l'indique est une interface par laquelle un programme offre des services à d'autres programmes informatiques. De manière générale, on parle d'API à partir du moment où une entité informatique cherche à agir avec un système tiers, et que cette interaction se fait de manière normalisée en respectant les contraintes d'accès définies par le système tiers. On dit que le système tiers expose une API. Les services d'une

API peuvent être de natures variées, pouvant aller de la gestion de bases de données, à la mise à disposition d'une carte affichant des positions géographiques (Google Maps API par Google (2019)). Dans le monde du web, on les appelle des web API, car elles sont mises en place par un serveur web ou un navigateur web. Un web API côté serveur est une interface composée d'un ou plusieurs points de terminaisons exposés publiquement à travers un système de requêtes/réponses défini dont le format est généralement JSON (ECMA-404, 2013) ou XML (W3C, 2006). Un point de terminaison est l'emplacement à partir duquel les API peuvent accéder aux ressources dont elles ont besoin pour exécuter leurs fonctions. Il est souvent modélisé par une URL. Par exemple, l'URL http://maps.googleapis.com/maps/api/directions/json représente le point de terminaison de l'API Google Maps pour la fonction de calcul des directions entre les emplacements. Cependant, comme dit précédemment, tous les sites web ne fournissent pas une web API côté serveur. D'ailleurs, en termes de proportion, la très grande majorité du web est destiné à un lecteur humain soit parce qu'il n'y a pas d'API disponible, soit parce que l'API qui existe est restreinte à un usage privé.

Après avoir récupéré les ressources désirées (textes, images) que cela soit par le biais de *web scraping* ou d'une API, nous sommes libres de leur exploitation ; ce qui est le sujet de la sous-section suivante.

3.1.4 LE *WEB MINING*

Le *web mining* ou la fouille du web regroupe l'ensemble des techniques utilisées pour découvrir des motifs sur le web impliquant des méthodes se trouvant à l'intersection de l'apprentissage automatique, des statistiques et des systèmes de base de données. En d'autres mots, le *web mining* est du *data mining* appliqué au web. On distingue trois types d'approches de *web mining*. On a d'abord le *web usage mining* dont le but est de trouver des motifs ou des informations pertinentes à partir des logs des serveurs web. On peut voir ces logs comme

une sorte d'historique des transactions des utilisateurs du point de vue d'un serveur. À titre d'exemple, le site web d'une banque, sur lequel on voudrait détecter les virements frauduleux pourrait avoir recours à du *web usage mining* afin d'assurer que seules les transactions considérées légales puissent se faire. Il est également possible de faire du *web structure mining*. Ce type de *web mining* se base sur la théorie des graphes (Bondy & Murty, 2008) et peut avoir deux objectifs distincts. Le premier est celui où l'on souhaite connaître les relations entre plusieurs pages, notamment en utilisant les hyperliens présents sur celles-ci. Un bon exemple de cela est l'algorithme PageRank (Page *et al.*, 1998) qui est un algorithme d'analyse de liens hypertextes attribuant un poids à chaque page d'un ensemble de pages hyperliées dans le but de mesurer son importance relative au sein de l'ensemble. La Figure 3.2 illustre le fonctionnement de l'algorithme. Le deuxième objectif du *web structure mining* peut également être l'analyse de la structure interne des pages web. On pourrait par exemple être intéressé par le nombre de balises du même type utilisées dans la page. Par exemple, l'élément HTML ayant pour balise <*progress*> a été ajouté avec la version 5 de HTML (W3.org, 2008). On pourrait alors tenter de mesurer la proportion des pages web utilisant celle-ci en 2020 et ainsi de se faire une idée de l'utilité de l'élément pour les développeurs de 2020. Enfin, il existe une dernière catégorie de *web mining*, le *web content mining*. On peut définir ce dernier par le processus consistant en l'extraction d'informations contenues dans les documents stockés sur internet pouvant être, comme on a pu le dire précédemment, du texte, de l'image, du son ou de la vidéo. À la différence des deux autres types de *web mining*, on ne s'intéresse ni à la structure, ni à l'utilisation du web, mais bien aux informations contenues. C'est ce *web content mining* qui est utile dans le cadre de cette thèse, car il répond à notre souhait de créer automatiquement une base de connaissances sur les activités de la vie quotidienne à partir du web.

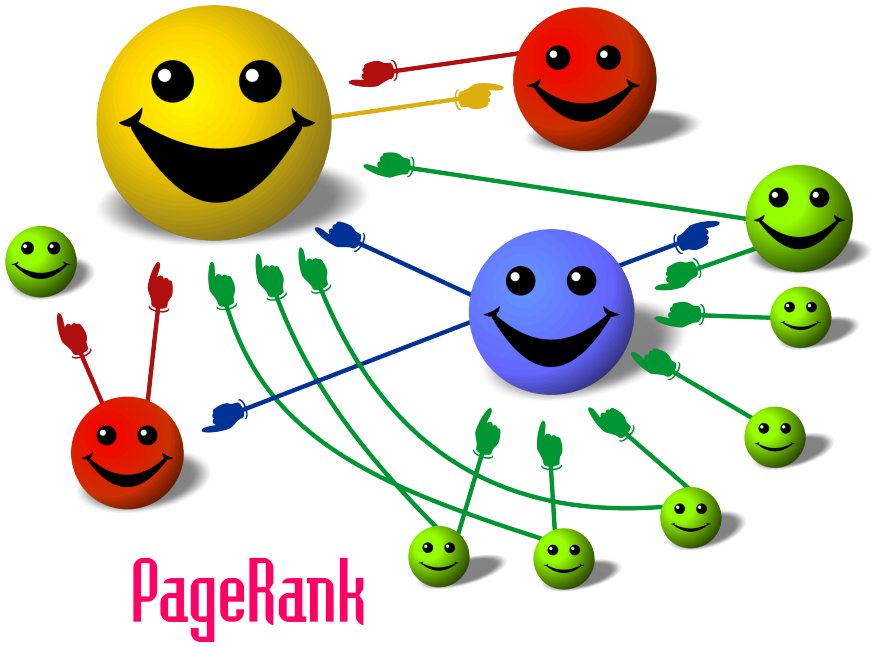

Figure 3.2 : Illustration du principe de base du PageRank. La taille de chaque visage est proportionnelle à la taille totale des autres visages qui le pointent (Mayhaymate, 2012). Image Creative Commons.

3.2 WEB ET RECONNAISSANCE D'ACTIVITÉS

Comme nous l'avons vu précédemment dans le chapitre 2, il existe de nombreuses façons de faire de la reconnaissance d'activités. Cependant, les approches de reconnaissance d'activités basées sur le web (ou *web-based activity recognition*) ont attiré peu d'attention de la part des chercheurs et cela a donné lieu à un petit nombre de travaux scientifiques. Pour commencer, il est nécessaire de définir ces approches correctement. La définition la plus proche est probablement celle donnée par Chen *et al.* (2012), en tant que la reconnaissance d'activités basée sur la fouille (*mining-based activity recognition*). Selon les auteurs, elle comporte quatre étapes. Premièrement, il faut trouver des sources enviables à exploiter ; le web étant la plus couramment utilisée. Deuxièmement, une phase de prétraitement est effectuée pour extraire les informations utiles des sources en utilisant des méthodes de recherche d'informations. Ces informations peuvent être des phrases où sont mentionnés des objets utilisés pendant

l'exécution de l'activité. Ensuite, les probabilités d'utilisation des objets sont estimées à l'aide d'un algorithme. Quatrièmement, en utilisant ces probabilités, on génère une modélisation de nos activités ; cette modélisation pouvant avoir différentes formes. On peut, par exemple, modéliser sous la forme de sacs de mots (*bag of words*) (terme employé pour la première fois par Harris (1954)) ou d'un modèle de Markov caché (Rabiner & Juang, 1986). Nous verrons ces modèles dans la section 3.2.3. Enfin, un algorithme de reconnaissance utilisant les modèles générés est mis en œuvre. Cette définition ne s'applique que partiellement à ce que nous appelons la reconnaissance d'activités basée sur le web. La raison principale est que la reconnaissance d'activités basée sur la fouille est limitée à l'extraction et à l'utilisation des probabilités d'utilisation des objets alors que nous pensons que d'autres informations utiles sont disponibles sur le web comme des informations spatiales, temporelles (l'ordre d'utilisation des objets ou des substances en cause par exemple). Cependant, cela ne change rien au fait que toutes les étapes successives représentent un aspect difficile de la reconnaissance d'activités basée sur le web, ce qui explique en partie pourquoi si peu de travaux ont été réalisés dans ce domaine.

3.2.1 DES SOURCES DÉSIRABLES

Le premier défi à relever lors de l'utilisation du web comme source de données pour la reconnaissance d'activités humaines est de savoir quels types de ressources faisant partie du web sont pertinentes et lesquelles ne le sont pas. Nous avons pu voir dans les sections 3.1.1 et 3.1.2 les différents avantages et inconvénients des différents types de ressources (textes, images, sons, vidéos) au niveau de leur exploitabilité et des informations latentes disponibles. La grande majorité des travaux existants choisissent d'utiliser principalement le texte des pages web comme source principale de données pour faire du *web content mining* (Wyatt *et al.*, 2005; Ihianle *et al.*, 2016; Palmes *et al.*, 2010; Tapia *et al.*, 2006; Perkowitz *et al.*, 2004).

Le texte est un choix pertinent, car de nombreux sites tels que ehows.com ou wikihow.com contiennent une description écrite d'un grand ensemble d'activités de la vie quotidienne (voir la Figure 3.3).

Toutefois, nous savons que le principal inconvénient de l'utilisation du texte est la quantité de bruit qu'il contient. Utiliser du texte provenant du web ajoute une difficulté supplémentaire. En effet, il y a du bruit provenant du texte inutile disponible sur la page web (exemple donné dans la Figure 3.3), mais il y a aussi du bruit dans les phrases pertinentes qui y sont disponibles. Si nous prenons l'expression *faire bouillir les pâtes*, nous pouvons facilement remarquer que si nous prenons la phrase telle quelle, elle contient des mots beaucoup plus importants que d'autres. Ainsi, ce sont les mots *bouillir* et *pâtes* qui sont essentiels à une bonne compréhension. Il est alors primordial de disposer de méthodes et de processus permettant d'extraire l'information essentielle comme nous l'avons stipulé dans la section 3.1.2.

Une minorité de travaux souhaitant exploiter le web ont pris conscience des problèmes de l'utilisation du texte et ont décidé d'utiliser un autre type de ressource, les images. Dans le travail de Riboni & Murtas (2017), les auteurs pensent que les images seraient une manière plus compacte et plus expressive de décrire une activité de la vie quotidienne et qu'en utilisant une technologie appropriée comme les CNNs, il serait parfaitement possible d'extraire des informations des images. Un point intéressant de ce travail est qu'une comparaison en termes de performance de la reconnaissance d'activités est faite avec l'approche plus traditionnelle utilisant le texte. Cette comparaison semble montrer la supériorité de l'utilisation d'images avec un F-score de 0.5998 pour l'approche basée sur le texte et un score F-score de 0.6988 pour l'approche basée sur les images. Plus récemment, la même équipe de recherche a mené des recherches plus approfondies sur l'utilisation d'images pour la reconnaissance d'activités (Riboni & Murtas, 2019). Dans ces nouveaux travaux, ils ajoutent des éléments pertinents.

Figure 3.3 : Capture d'écran d'une page décrivant l'activité *cuire des coquillettes* avec les zones contenant de l'information pertinente encadrées en orange (à partir de https://fr.wikihow.com/cuire-des-coquillettes consulté le 14/11/2019). Contenu Creative Commons.

Tout d'abord, ils développent une ontologie de correspondance ArOnt (DomuSafe, 2019) qui met en correspondance les événements des capteurs survenus dans un environnement donné avec les éléments extraits des images. Deuxièmement, ils étendent l'utilisation des images à l'utilisation de courtes vidéos. En outre, ils testent leur approche avec plusieurs ensembles de données, plusieurs API de vision par ordinateur et avec différents algorithmes de reconnaissance. Comme dans leurs travaux précédents, les résultats trouvés tendent à montrer que l'utilisation d'images au lieu de texte est pertinente, donnant des performances équivalentes ou supérieures.

3.2.2 LA CLASSIFICATION DES PAGES WEB PAR GENRE

Imaginons que nous avons décidé d'utiliser le texte des pages web. Il existe un deuxième défi qui consiste à construire un système capable de déterminer si une page web contient ou non des informations pertinentes. En effet, en raison de l'énorme quantité d'informations disponibles sur le web, un système souhaitant l'utiliser devra nécessairement se doter d'un moteur de recherche capable de fournir des résultats pertinents. Toutefois, il est important de savoir qu'un moteur de recherche est également susceptible de fournir des pages web non pertinentes. La question est donc de savoir comment s'assurer que nous utilisons des pages web qui non seulement parlent d'une activité de la vie quotidienne, mais décrivent aussi les étapes importantes. De notre point de vue, une *bonne* page web devrait alors ressembler à une sorte de recette ou à une procédure avec des étapes détaillées contenant des informations pertinentes, c'est une page web descriptive. La tâche décrite est appelée la classification des sites web par genre. Ce type de classification consiste à distinguer les documents selon leur forme, leur style ou leur public cible sans tenir compte du contenu. Dans notre cas, ce que nous désirons, ce sont des sites web pour le grand public, sans style particulier, mais avec une forme descriptive contenant des étapes détaillées.

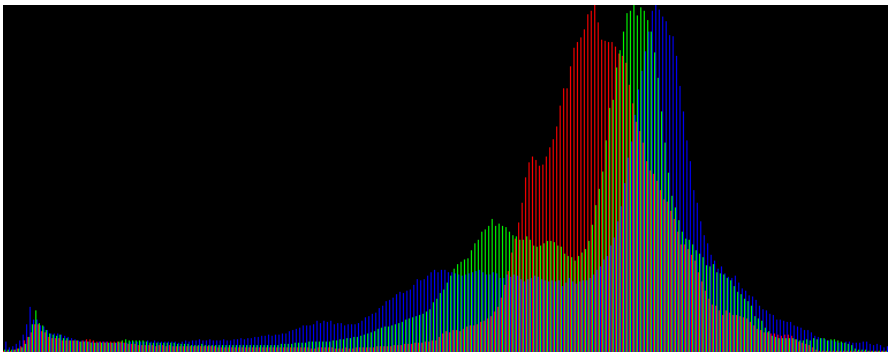

Figure 3.4 : Exemple d'un histogramme de couleurs d'une image avec en abscisse le système Rouge-Vert-Bleu et en ordonnée la fréquence (Muenteroman, 2015). Image Creative Commons.

La littérature est pleine de travaux qui ont tenté de classer par genre en utilisant des caractéristiques très différentes. Ces caractéristiques comprennent des éléments textuels comme les URL, les titres, les rubriques, les ancres ou des caractéristiques structurelles (nombre de liens sur la page, tableaux, paragraphes, etc.), lexicaux (nombre total de mots dans l'attribut *alt* des images, nombre de mots-clés présents, etc.) ou fonctionnels (taille en mémoire de la page, nombre de scripts chargés, etc.) (Pentney *et al.*, 2006; Malhotra & Sharma, 2017). Une approche originale par de Boer *et al.* (2011) est capable d'utiliser une capture d'écran de page web pour extraire des caractéristiques. À partir de cette capture d'écran, les auteurs ont pu extraire 4 groupes de caractéristiques : *Simple Color Histogram* composé de 32 attributs, *Edge Histogram* composé de 80 attributs, les caractéristiques de Tamura (Tamura *et al.*, 1978) composées de 18 attributs et les caractéristiques de Gabor (basées sur les travaux de Daugman (1985)) composées de 36 attributs. Ces groupes sont en réalité différentes représentations possibles d'une image ; par exemple, le *Simple Color Histogram* représente la répartition des couleurs sur une image. La Figure 3.4 montre un tel histogramme pour une image.

Grâce à cet ensemble de caractéristiques, les auteurs ont pu déterminer, à partir d'un échantillon de pages web, une valeur esthétique, une valeur actuelle, mais aussi et surtout

le genre de la page web. Outre le choix de ces caractéristiques, il est important de choisir le type d'algorithme à utiliser pour classer les pages web. Dans l'ouvrage Malhotra & Sharma (2017), les auteurs soulignent qu'il y a trois choix possibles. Le premier consiste à utiliser des algorithmes paramétriques tels que la régression logistique (Desjardins, 2005). Ces algorithmes ont un biais élevé (font des hypothèses fortes), ce qui les rend rapides pour la phase d'apprentissage, mais sont moins flexibles. Le second choix consiste à utiliser des algorithmes non paramétriques tels que les arbres de décision. En choisissant un algorithme de ce type, on constate qu'ils sont souvent plus flexibles et permettent une meilleure précision. Cependant, ils nécessitent un plus grand espace d'échantillonnage et un temps d'apprentissage plus long. Enfin, les méthodes d'ensemble telles que le *bagging* (Breiman, 1996) ou le *boosting* (Zhou, 2012) permettent d'obtenir des classeurs fiables malgré un échantillon d'apprentissage de petite taille grâce à l'utilisation d'une combinaison de classeurs.

La meilleure façon de choisir l'algorithme de classification des pages web est certainement de tester une variété d'algorithmes, de mesurer les performances sur un ensemble de données de test et de garder le meilleur. L'une des méthodes permettant d'estimer la fiabilité des algorithmes est la validation croisée (Kohavi, 1995). Elle est principalement utilisée dans des contextes où l'objectif est la prédiction et où le but est d'estimer avec précision la performance d'un modèle prédictif dans la pratique.

3.2.3 MODÉLISATION DES ACTIVITÉS DE LA VIE QUOTIDIENNE

Une fois que nous disposons de ressources de qualité provenant du web (pages web descriptives, images descriptives, vidéos descriptives, etc.) sur les activités de la vie quotidienne, la question importante concerne la manière de les utiliser, les informations qu'il est pertinent d'extraire, les informations qui seraient en mesure de représenter correctement une activité. Dans la grande majorité des travaux existants en matière de reconnaissance d'activités

basée sur le web (Wyatt *et al.*, 2005; Perkowitz *et al.*, 2004; Riboni & Murtas, 2017; Gu *et al.*, 2010; Tapia *et al.*, 2006), une activité est représentée comme une liste d'objets avec un poids associant l'activité à un objet ; ce modèle est un sac d'objets, inspiré du concept de sac de mots (Wikipedia, 2019; Cousyn, 2018). En effet, l'idée d'associer des objets et des activités a du potentiel, pour plusieurs raisons. Tout d'abord, s'il est assez facile de comprendre que si une activité implique un objet *théière* ainsi qu'un *sachet de thé*, l'activité réalisée est susceptible d'être *faire du thé*, alors on peut imaginer que l'association objet - activité est prometteuse. Deuxièmement, puisque nous avons préalablement choisi des pages web descriptives ou des images précisant tout ou partie des étapes à suivre pour réaliser l'activité, il est très probable que les objets impliqués dans sa réalisation soient présents, quelle qu'en soit la forme. Ainsi, selon certains critères et formules statistiques, tels que TF-IDF (*Term Frequency—Inverse Document Frequency* par Robertson (2004)), chaque activité peut être représentée comme un sac d'objets. La Figure 3.5 montre un exemple de la représentation de l'activité *faire du thé* sous cette forme.

Le seul ajout notable à l'utilisation du modèle du sac d'objets se situe dans le travail de Gu *et al.* (2010) où les auteurs utilisent la notion de modèles de contraste. Ces modèles sont en réalité des règles d'association entre objets où une valeur de liaison ou de relation entre les objets est calculée pour quantifier l'importance d'un n-uplet d'objets.

Cette idée d'association entre objets et activités s'applique également à d'autres types de modèles, comme les HMM. Les HMM est une catégorie de modèle permettant de trouver la séquence d'états cachés la plus probable en fonction d'une séquence d'observations. Ainsi, comme dans les travaux de E. Munguia Tapia *et al.* (2005), nous pourrions construire un modèle de HMM dont les probabilités d'émission sont calculées à partir de ce score d'association entre

Make tea	
Object	Weight
tea	1.00
water	0.85
cup	0.83
sugar	0.75
teapot	0.75
pot	0.74
bowl	0.72
lemon	0.70
kettle	0.70
microwave	0.67

Figure 3.5 : Exemple de représentation de l'activité *faire du thé* sous forme de sac d'objets par Gu *et al.* (2010). Figure réutilisée avec la permission d'Elsevier (voir Annexe B)

activités et objets qui permet d'envisager une approche non supervisée de la reconnaissance d'activités.

En dépit de l'efficacité relative des modèles de sac d'objets, comme le disent les travaux de Riboni & Murtas (2019), les travaux futurs pourraient choisir de s'orienter vers l'utilisation d'informations temporelles qui pourraient être obtenues par l'extraction d'informations d'activités à partir de vidéos afin de calculer des modèles d'activités plus précis.

3.2.4 RECONNAISSANCE D'ACTIVITÉS HUMAINE

Une fois les modèles d'activités à disposition, nous pouvons les utiliser en trouvant une méthode permettant de déduire l'exécution d'une activité à partir de données expérimentales. À noter que par données expérimentales, nous parlons d'un flux d'événements, ces événements étant fournis par un ensemble de capteurs dans un environnement donné et chaque événement étant associé à un objet (par exemple, *prendre la théière* est un événement). Presque tous les travaux sur la reconnaissance d'activités basée sur le web utilisent une méthode différente

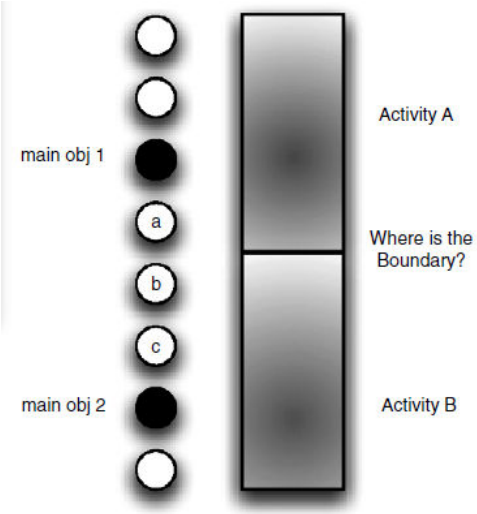

Figure 3.6 : Recherche de l'objet frontière dans une séquence contenant deux activités consécutives. Les ronds noir et blanc représentent respectivement les objets clés et les objets quelconques (Palmes *et al.*, 2010)

pour inférer l'activité à partir des modèles générés, mais la grande majorité choisit d'utiliser le modèle de sac d'objets introduit en section 3.2.3. Une première méthode, que nous avons déjà mentionnée, de Palmes *et al.* (2010) consiste, pour chaque sac d'objets associés à une activité, à ne considérer que l'objet ayant le poids associé le plus élevé, également appelé objet clé. La logique appliquée par les auteurs est alors la suivante : la présence de cet objet dans une séquence d'événements signifiera que l'activité associée est en cours de réalisation. Une fois que tous les objets clés sont identifiés dans une séquence, ils appliquent des algorithmes de segmentation personnalisés pour trouver le meilleur objet afin de séparer les deux objets clés de deux activités distinctes. On se trouve alors dans un problème de recherche de frontière entre les activités consécutives. La Figure 3.6 illustre cette recherche. On obtient alors des sections de séquences dont chacune comporte un objet clé unique. À partir de là, on dira qu'un événement e fait partie d'une activité a si et seulement s'il se trouve dans une section de la séquence où un objet clé associé à l'activité a est présent.

D'autres approches consistent à calculer un score d'importance. Dans les travaux de Riboni & Murtas (2017, 2019), leur première méthode consiste à calculer un poids, pour un temps t_j, pour chaque activité a ; j étant l'indice de l'instant considéré. Ce calcul s'effectue à l'aide d'une fenêtre glissante (Wang, 2018) de taille n. L'activité choisie pour un temps t_j sera celle qui aura le poids le plus élevé. Le calcul de ce poids est donné par l'équation (3.2). Notons que la probabilité utilisée dans cette équation correspond au poids utilisé dans le sac d'objets associés à l'activité a.

$$w(a,t_j) = \prod_{k=j-n+1}^{j} p(a|o(e_j)) \times c^{j-k} \tag{3.2}$$

où

n est la taille de la séquence considérée

k est l'indice d'un des instants de la séquence considérée

$c \in (0, 1]$ est le facteur de lissage temporel

o est la fonction associant à un événement e_j l'objet impliqué dans cet événement.

Dans leur deuxième méthode, ils utilisent les réseaux logiques de Markov (Richardson & Domingos, 2006) (*Markov logic network* ou MLN). Un réseau logique de Markov est un réseau de Markov (Vala, 2019) dans lequel on applique de la logique du premier ordre. Dans ces réseaux, on dispose de formules de logique du premier ordre et chacune est associée à un poids quantifiant l'importance de la formule (University of Waterloo, 2008). La Figure 3.7 montre la représentation associée du problème de reconnaissance d'activités dans ce cas.

Dans d'autres travaux, Gu *et al.* (2010) ont raisonné sur une séquence de longueur L_{a_i} plutôt que sur un moment précis. La fonction de score $f(a_i, S_{t\sim t+L_{a_i}})$ est calculée, qui dépend du modèle de contrainte, de l'activité a_i et d'une séquence d'événements $S_{t\sim t+L_{a_i}}$. De

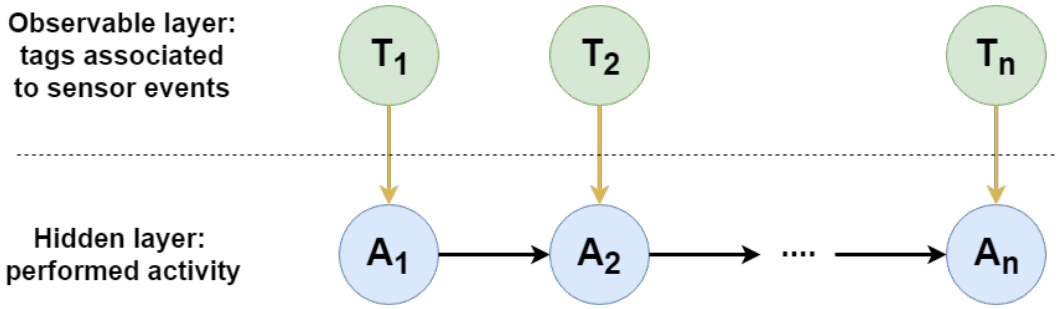

Figure 3.7 : Formulation du problème de reconnaissance d'activités en tant que MLN. T_i est l'élément dérivé de l'événement du capteur observé à l'instant τ_i. A_i est l'activité courante (cachée) à τ_i. On voit alors que A_i dépend à la fois de T_i et de l'activité précédente A_{i-1} (Riboni & Murtas, 2019). Figure réutilisée avec la permission d'Elsevier

la même manière, l'activité choisie pour une séquence $S_{t\sim t+L_{a_i}}$ sera celle qui aura le score le plus élevé. Enfin, des approches probabilistes ont aussi été explorées telles que les HMM et la PLSA (Probabilistic Latent Semantic Analysis Ihianle *et al.* (2016)). Pour rappel, les HMM est une catégorie de modèle permettant de trouver la séquence d'états cachés (activités) la plus probable en fonction d'une séquence d'observations (événements). La PLSA est une technique principalement utilisée pour l'extraction de sujets dans le texte. Vu sous cet angle, la PLSA a 3 composantes (Karani, 2018) :

— des documents : $D = (d_1, d_2, d_3, \ldots d_N)$, N est le nombre de documents. Le terme *document* peut également signifier phrases.

— des mots : $W = (w_1, w_2, \ldots w_M)$, M est la taille du vocabulaire. L'ensemble W est considéré comme un sac de mots. Cela signifie qu'il n'y a pas d'ordre particulier des mots en lien avec l'indice i

— des sujets : $Z = (z_1, z_2, \ldots z_k)$. Ce sont les variables cachées. Le nombre k est spécifié au préalable.

L'intérêt de la PLSA est d'inférer de manière probabiliste les sujets en fonction des documents et des mots. C'est-à-dire qu'elle nous permet de relier les variables cachées aux variables

observées. Appliqué à la reconnaissance d'activités, dans les travaux de Ihianle *et al.* (2016), les documents sont des séquences d'événements, les mots sont les événements et les sujets sont les activités à reconnaître.

3.2.5 LIMITES DES APPROCHES EXISTANTES

Les différentes approches présentées sur la reconnaissance d'activités humaines basée sur le web ont également leur lot de limitations que nous allons présenter dans la présente section.

Tout d'abord, le travail de Palmes *et al.* (2010) n'utilise qu'un unique objet clé pour identifier chacune des activités. Nous pensons que considérer plusieurs objets pour reconnaître une activité donnée a un bénéfice. Nous illustrons ce propos dans les chapitres 4 et 6 en analysant les performances de reconnaissance apportées par l'usage de plusieurs objets extraits.

La deuxième limitation est que certains travaux (Gu *et al.*, 2010; Wyatt *et al.*, 2005; Perkowitz *et al.*, 2004) utilisent pour seule source de données des tags RFID. C'est une limitation, car les objets avec lesquels un résident interagit ne se limitent pas aux objets pouvant se déplacer. Or, les tags RFID permettent surtout de localiser des objets et donc d'analyser leurs déplacements pour reconnaître des activités. Ces travaux étant anciens et les autres travaux utilisant des données d'autres sources, nous utilisons des données de multiples sources dans le chapitre 6.

Troisièmement, la plupart des travaux (Wyatt *et al.*, 2005; Perkowitz *et al.*, 2004; Palmes *et al.*, 2010; Riboni & Murtas, 2017, 2019) n'utilisent pas de notion d'ordre par rapport aux objets utilisés pour la réalisation des activités humaines. Nous pensons que l'ordre d'utilisation des objets a un fort potentiel et nous évaluons son utilisation dans les chapitres 5 et 6.

Enfin, la dernière limitation que nous voyons est qu'aucun des travaux existants ne traite du fait de combiner les informations provenant des images et du texte provenant du web. Cette thèse, et en particulier le chapitre 6 évalue la combinaison de ces deux sources d'informations au niveau des performances de reconnaissance.

3.3 CONCLUSION

Dans ce chapitre, nous commençons par présenter le potentiel du web pour la reconnaissance d'activités. Dans un premier temps, nous identifions les différents types de ressources existantes sur le web, à savoir le texte, les images, les sons et les vidéos. Nous avons identifié les différentes informations que renferment ces ressources, à savoir, des informations sur les objets et substances impliqués dans la réalisation des activités, des informations temporelles ainsi que des informations spatiales. Ensuite, nous avons discuté de l'exploitabilité de chacune de ces ressources et avons déterminé que le texte et les images représentent le meilleur parti en terme de qualité et de quantité. Par la suite, nous avons présenté la manière de récupérer de manière automatisé les ressources du web en définissant des concepts essentiels comme le *web scraping* et les API. Enfin, nous définissons la tâche de *web mining* et les trois types d'approches pour l'appliquer. Nous avons déterminé que la tâche effectuée dans cette thèse fait partie de la catégorie du *web content mining* dont le but est d'extraire des informations des pages web à partir de leur contenu sans considérer la structure, ni l'usage.

Dans la seconde section, nous présentons l'état de l'art de l'approche de la reconnaissance des activités basée sur le web. Nous avons défini et détaillé ses différentes étapes constituant sa réalisation. Nous avons d'abord discuté des types de ressources utilisées dans les travaux existants. Il en ressort que le texte est la ressource la plus utilisée par les chercheurs. Cependant, d'après quelques travaux, les images semblent présenter un fort potentiel. Nous avons ensuite discuté de la classification des pages web par genre afin de nous faire une idée

de la manière d'obtenir des pages web descriptives sur les activités de la vie quotidienne. D'après cette discussion, une grande partie des travaux utilisent des éléments textuels des pages web contenus dans le HTML. Une approche différente existe malgré tout, usant d'une capture d'écran de la page web pour extraire des caractéristiques qui permettront de classifier les pages web par genre. Troisièmement, nous avons identifié les différentes manières utilisées pour modéliser les activités de la vie quotidienne. Les modèles sont soit des sacs d'objets, soit des HMM. Quatrièmement, nous avons discuté des algorithmes de reconnaissance d'activités utilisés pour exploiter les modèles d'activités dans la littérature. Ces algorithmes sont de natures très variées. Pendant que certains calculent un score de pertinence pour chaque activité et gardent l'activité ayant le meilleur score comme hypothèse la plus probable, d'autres utilisent un concept d'objets clés permettant d'identifier l'activité en cours et tentent de segmenter les activités consécutives. Enfin, les approches existantes ont leur lot de limitations que nous allons adresser dans cette thèse. Par exemple, l'utilisation d'images et de texte tout en exploitant une notion d'ordre d'utilisation des objets pour reconnaître les activités humaines semble avoir un potentiel inexploré que nous allons tenter de révéler.

La tâche de reconnaissance des activités de la vie quotidienne utilisant le web comme source principale est difficile, comme le montre le faible nombre d'ouvrages dans la littérature. Cette thèse vise à améliorer la qualité de la démarche de reconnaissance d'activités ainsi qu'à contribuer à l'approfondissement d'une méthode moins connue qui possède un potentiel certain en termes d'informations disponibles. Cette méthode voulant notamment exploiter des images, le chapitre suivant illustrera la capacité de ces dernières à modéliser les activités de la vie quotidienne.

CHAPITRE IV
DÉTECTION D'OBJETS SUR LE WEB POUR DÉCOUVRIR LES OBJETS CLÉS DANS LES ACTIVITÉS HUMAINES

Dans le précédent chapitre, nous avons montré que le web a déjà été utilisé dans des problématiques de reconnaissance d'activités humaines. D'après les travaux de Riboni (Riboni & Murtas, 2017, 2019), une approche intéressante et innovante semble être l'utilisation d'images provenant de résultats de moteurs de recherches connus. Utiliser les images dans cette thèse permettrait d'approfondir ces travaux et de se faire une meilleure idée de ce qui est possible d'en tirer.

Afin d'aller dans la même direction que les travaux de Riboni (Riboni & Murtas, 2017, 2019), le but de ce chapitre est d'évaluer la capacité des images du web à modéliser les activités de la vie quotidienne à l'aide d'objets clés détectés sur ces images. Pour ce faire, nous basons le contenu du présent chapitre sur notre article (Cousyn *et al.*, 2021) publié dans le *Journal of Ambient Intelligence and Humanized Computing*. L'éditeur *Springer* a donné son accord pour traduire toutes les figures et tableaux du travail original et de les intégrer dans cette thèse.

Les questions de recherche auxquelles nous souhaitons répondre sont donc les suivantes. Premièrement, dans quelle mesure les images extraites des moteurs de recherche permettent-elles la représentation des activités de la vie quotidienne par rapport aux approches existantes ? Deuxièmement, comment peut-on trouver et exploiter de manière automatisée les images disponibles sur le web relatives aux activités de la vie quotidienne ? La contribution du chapitre est constituée de deux éléments distincts. Tout d'abord, notre contribution théorique est la proposition d'une méthode d'extraction d'informations du web liées à la réalisation d'activités

de la vie quotidienne. Ensuite, notre contribution pratique consiste en un système complet en JavaScript contenant trois modules : un module de recherche d'images sur le web pour obtenir les résultats des moteurs de recherche [15], un module pour récupérer les images des résultats des moteurs de recherche [16] et un module pour identifier, extraire et évaluer les objets liés aux activités [17].

4.1 MÉTHODOLOGIE

Cette section décrit l'approche permettant d'extraire des objets et des matériaux d'images obtenues à partir d'un moteur de recherche d'images sur le Web. L'approche est représentée dans la Figure 4.1.

4.1.1 MODULE DE RECHERCHE D'IMAGES

La première étape de notre méthode consiste à fournir l'étiquette d'une activité au module de recherche d'image. Ce module est chargé d'utiliser cette étiquette pour effectuer une requête sur le moteur de recherche d'images choisi et récupérer les k premières images qu'il fournit. Dans les faits, cette tâche est réalisée en combinant l'utilisation de deux modules librement accessibles de notre création qui peuvent être utilisés en suivant les instructions fournies dans nos dépôts [18] [19].

15. https://github.com/CharlesCousyn/search_activities
16. https://github.com/CharlesCousyn/image_retrieval
17. https://github.com/CharlesCousyn/image_extractor
18. https://github.com/CharlesCousyn/search_activities
19. https://github.com/CharlesCousyn/image_retrieval

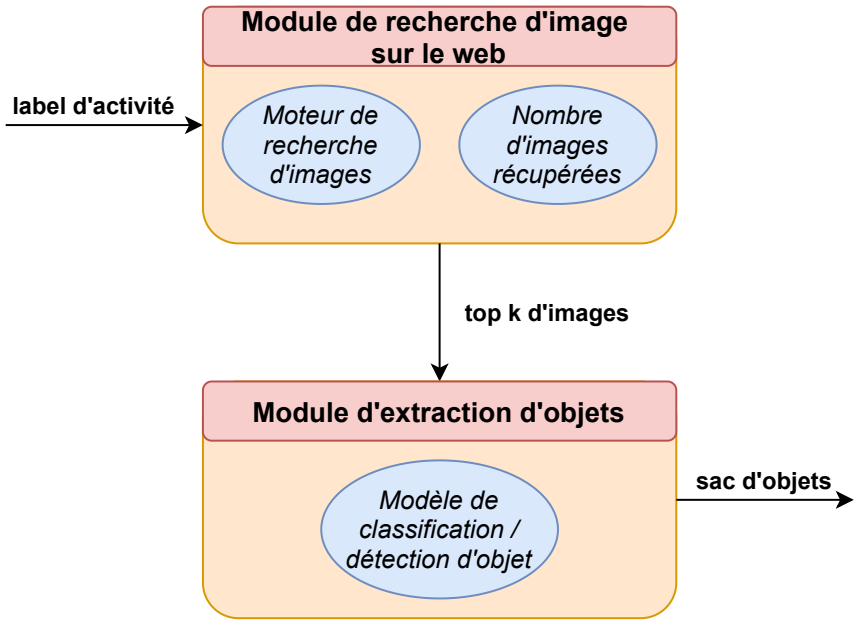

Figure 4.1 : Schéma représentant la méthode d'extraction d'objets à partir d'une étiquette d'activité. © Charles Cousyn, 2022.

4.1.2 MODULE D'EXTRACTION D'OBJETS

Ces images sont ensuite envoyées au module d'extraction d'objets [20], qui se charge de faire passer chacune des images dans un modèle de classification d'images ou de détection d'objets. Pour ce faire, le module filtre les images inutilisables en raison de leur format et les redimensionne dans des dimensions acceptées par le modèle. Pour une image donnée, le modèle de classification/détection d'objets produit un ensemble de couples (objet détecté, score de confiance) que nous appelons sac d'objets (voir exemples en Figure 4.2).

4.1.3 PROCESSUS D'AGRÉGATION DES SACS D'OBJETS

Pour une activité donnée, le module est chargé de compiler et d'agréger les objets détectés sous la forme d'un sac d'objets où le score associé à chaque objet est la somme des

20. https://github.com/CharlesCousyn/image_extractor

scores de confiance associés à cet objet fournis par le modèle de classification ou de détection d'objets. Nous appelons le résultat de cette agrégation : sac d'objets unifié. La Figure 4.2 représente un exemple de cette agrégation. Mathématiquement, pour chaque objet j présent sur les images, le score de confiance associé est :

$$\text{conf}_j = \sum_i \text{conf}_{i,j} \tag{4.1}$$

où

i est l'indice d'une image

j est l'indice de l'objet considéré

$\text{conf}_{i,j}$ est la confiance que l'objet j soit présent sur l'image i.

Le lecteur doit être conscient que la simplicité de la méthode utilisée cache en réalité de grandes incertitudes. Quelles sont les activités à utiliser ? Quel modèle de classification et de détection des objets doit-on utiliser ? Quelle doit être la valeur du nombre k ? Afin de répondre à ces questions, la section suivante présente notre méthode d'analyse de ces différents critères et ce que leur variation implique pour la liste des objets et matériaux obtenus pour chaque activité.

4.2 MÉTHODE D'ÉVALUATION

Cette section présente le processus utilisé pour tester la performance de l'approche pour l'extraction d'objets et de matériaux à partir d'images liées aux activités de la vie quotidienne récupérées à partir de moteurs de recherche d'images.

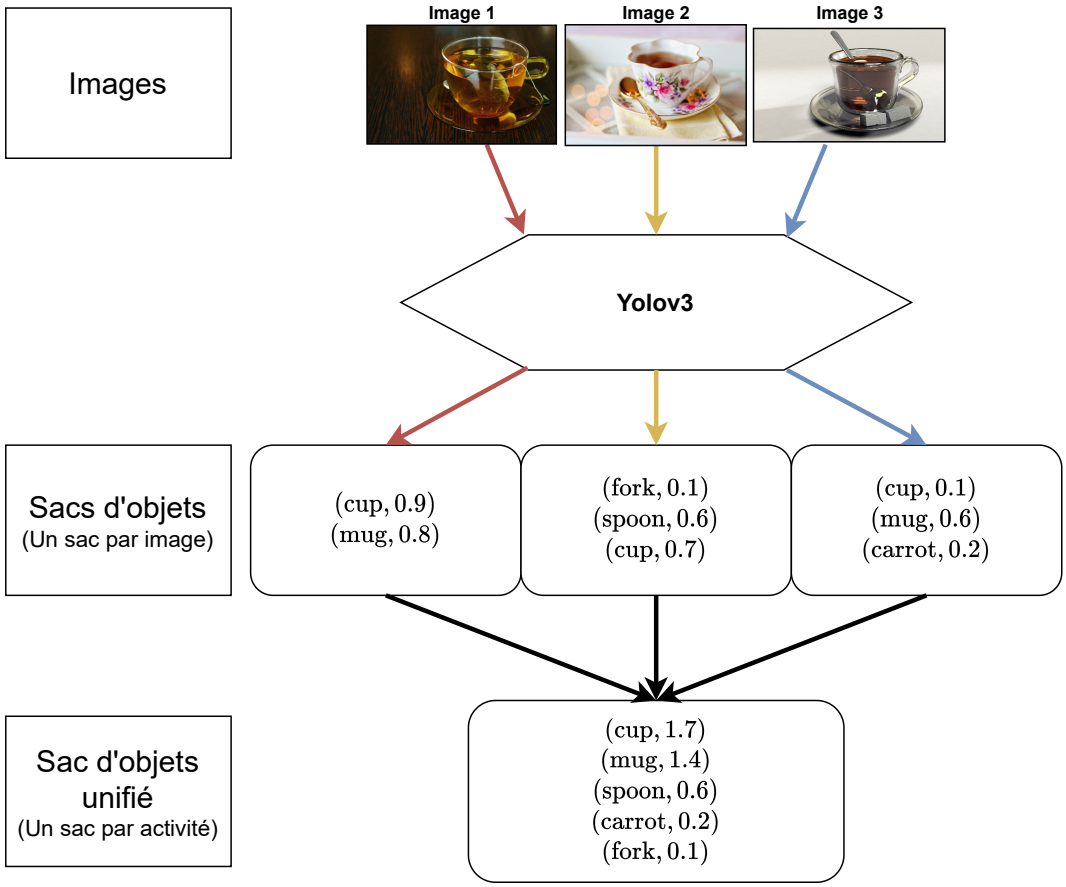

Figure 4.2 : Exemple du processus d'agrégation pour l'activité *make tea* en utilisant le modèle de détection d'objets Yolo v3. Le résultat de l'agrégation est un sac d'objets où le poids de chaque objet est la somme des scores de confiance associés à cet objet fournis par le modèle de classification ou de détection d'objets. Remarque : les objets utilisés dans cet exemple montrent que le modèle de détection d'objets peut détecter des objets qui sont absents des images (par exemple : fourchette, carotte). © Charles Cousyn, 2022.

4.2.1 ACTIVITÉS CONSIDÉRÉES ET JEU DE DONNÉES

La première chose à préciser est l'ensemble des activités de la vie quotidienne qui seront utilisées pour évaluer les performances de l'approche. Notre équipe de recherche a défini un ensemble totalisant 18 activités distinctes. Pour être précis, 8 activités proviennent de celles utilisées dans les travaux de Riboni & Murtas (2017, 2019) qui sont très proches de notre problématique (elles-mêmes inspirées du dataset CASAS *Interweaved ADL Activities*[21]), 6 activités ont été créées en s'inspirant fortement du dataset CASAS *ADL Activities*[22] et 4 activités ont été inventées de toutes pièces. Cet ensemble de 18 activités est séparé en deux sous-ensembles contenant des activités de natures distinctes. Le premier, contenant 14 (8+6) activités, désigne un ensemble d'activités génériques. Nous définissons ces activités par le terme *générique*, car la désignation de ces activités se veut générale et leur interprétation peut varier fortement. Ces activités ont pour étiquettes :

— *fill medicine cabinet*

— *watch tv*

— *water plants*

— *answer the phone*

— *prepare a birthday card*

— *prepare soup*

— *clean*

— *vacuum*

— *choosing outfit*

— *make tea*

21. http://casas.wsu.edu/datasets/adlinterweave.zip

22. http://casas.wsu.edu/datasets/adlnormal.zip

— *make coffee*

— *cook pasta*

— *do homework*

— *play video games*

Le deuxième sous-ensemble, contenant 4 activités imaginées, fait référence aux activités dont la désignation est exprimée avec une granularité plus fine que les activités génériques. Ces activités ont pour étiquettes :

— *bake parmesan turkey meatballs*

— *make hut stuffed crust pizza*

— *make mashed potato casserole*

— *make low carb pancakes*

L'intérêt de cette séparation est de savoir si le fait de fournir plus d'informations dans le libellé de l'activité (donc d'être plus spécifique) peut avoir un impact sur les performances que l'on peut atteindre, donc de savoir si la granularité d'une activité a un impact ou non.

Après avoir sélectionné les activités de la vie quotidienne utilisées pour mesurer la performance, notre équipe de recherche a procédé au développement d'un jeu de données de référence. L'objectif est de permettre la comparaison avec les objets extraits par notre programme informatique. À cette fin, ce jeu de données contient pour chaque activité, tous les objets et/ou matériaux réellement impliqués dans sa réalisation. Notre équipe de recherche a fait de son mieux pour dresser une liste exhaustive pour chaque activité. Cependant, comme ce jeu de données a été compilé manuellement, il est important de considérer que ce jeu de données peut être biaisé. Par exemple, il est tout à fait possible que pour une activité donnée, les objets impliqués soient différents selon la culture et la localisation de ses créateurs. En tant que tel, cet ensemble de données peut souffrir des problèmes suivants : ne pas être

exhaustif pour certaines activités (faux négatif), contenir des objets ou des matériaux peu ou pas pertinents (faux positif). Cependant, nous pensons que ce jeu de données pourrait constituer un premier élément de référence de qualité suffisante pour ce travail. Le jeu de données de référence est disponible en ligne [23].

4.2.2 CRITÈRES DE TEST

Comme pour la granularité des activités, nous avons testé l'approche selon un certain nombre de critères :

— Le moteur de recherche d'images utilisé (Google Image ou DuckDuckGo Image)

— Le modèle de classification/détection d'objets utilisé

— Le nombre d'images par activité utilisées par ordre de pertinence selon les moteurs de recherche (25, 50, 100, 200, 500, 1000)

— La granularité de l'activité (générique ou spécifique)

Outre la granularité des activités, nos choix sur les différentes valeurs de ces critères sont justifiés par les affirmations suivantes. Le choix d'utiliser Google et DuckDuckGo comme moteur de recherche d'images pour notre méthode est justifié par leur haute performance (Fishkin, 2019) et/ou par leur utilisation massive par la population (DuckDuckGo, 2021). Le nombre d'images par activité utilisé ne dépasse jamais 1000 lors de la récupération à partir des deux moteurs de recherche utilisés. Nous suspectons également un effet discriminatoire sur les performances d'un petit nombre d'images, c'est pourquoi nous nous concentrons plutôt sur les valeurs dans l'intervalle $[0, 500]$ que celles de l'intervalle $[500, 1000]$.

23. https://github.com/CharlesCousyn/image_extractor/blob/master/configFiles/groundTruth.json

En ce qui concerne les modèles de classification/détection d'objets utilisés, nous avons choisi de tester plusieurs d'entre eux qui sont assez récents, i. e., PNasnet large (Liu *et al.*, 2017), Inception resnet v2 (Szegedy *et al.*, 2017), Mobilenet v2 140 224 (Sandler *et al.*, 2018), Yolo9000 (Redmon & Farhadi, 2017), Yolov3-tiny (Redmon & Farhadi, 2018) et Yolov3-608 (Redmon & Farhadi, 2018). Les 3 premiers sont des modèles de classification d'images, les 3 derniers sont des modèles de détection d'objets. Contrairement aux modèles de classification, les modèles de détection ont différents paramètres qui doivent être définis avant de les utiliser. Il y a 3 paramètres : le nombre de boîtes de délimitation (*bounding boxes*) utilisées pour la détection, le seuil de confiance pour chaque détection et l'indice de Jaccard (*intersection over union* ou IoU). L'indice de Jaccard est en réalité le quotient de l'aire de l'intersection de deux boites de délimitation et de l'aire de l'union des deux boites de délimitations et est utilisé pour éliminer plusieurs boîtes qui entourent le même objet, en fonction de la boîte qui a le score de confiance le plus élevé. On utilise généralement l'algorithme de suppression non maximale (Ng, 2020) pour effectuer cette suppression. Après de multiples tests pour savoir comment ces paramètres influencent les performances des modèles, nous avons décidé de fixer le nombre de boîtes de délimitation à 20 (il y a rarement plus de 20 objets sur les images utilisées), de fixer l'indice de Jaccard à 0.5 (valeur utilisée dans les travaux de Hui (2019)), mais de faire varier le seuil de confiance, car, dans le contexte de l'implémentation de ces modèles, il semble que ce paramètre ait un impact sur les objets détectés. Ce seuil prendra deux valeurs distinctes : 0.05 et 0.1. En faisant varier ce paramètre, le nombre de configurations de modèles de classification/détection d'objets utilisées passe alors de 6 à 9. Nous nous retrouvons avec les 9 configurations de modèles suivantes (pour les noms des modèles de détection, les tirets inférieurs séparent les paramètres entre eux) : *pnasnet_large*, *inception_resnet_v2*, *mobilenet_-v2_140_224*, *yolo9000_20_0. 05_0.5*, *yolov3-tiny__20_0.05_0.5*, *yolov3-608__20_0.05_0.5*, *yolo9000__20_0.1_0.5*, *yolov3-tiny__20_0.1_0.5* et *yolov3-608__20_0.1_0.5*. Par ailleurs, dans un souci de notation et afin de permettre la lisibilité des figures, le Tableau 4.1 donne

Nom complet et nom abrégé	
inception_resnet_v2	ir2
mobilenet_v2_140_224	m2
pnasnet_large	pl
yolo9000__20_0.05_0.5	y9_0.05_0
yolo9000__20_0.1_0.5	y9_0.01
yolov3-608__20_0.05_0.5	y3_0.05
yolov3-608__20_0.1_0.5	y3_0.1
yolov3-tiny__20_0.05_0.5	y3t_0.05
yolov3-tiny__20_0.1_0.5	y3t_0.1
Google Image	g
DuckDuckGo Image	d

Tableau 4.1 : Tableau de conversion des noms des modèles de classification/détection d'objets et des moteurs de recherche utilisés en noms raccourcis. © Charles Cousyn, 2022.

la correspondance entre les modèles de classification/détection d'objets et leurs noms courts ainsi que celui entre le moteur de recherche et leur nom court.

Normalement, à la lecture rapide de nos critères, avec 2 moteurs de recherche, 9 configurations de modèles de classification/détection d'objets, 6 nombres différents d'images par activité et 2 catégories de granularité, nous devrions avoir 216 configurations possibles pour nos expérimentations (2x9x6x2=216). Cependant, il faut comprendre que le critère de granularité n'a pas d'impact sur le fonctionnement de notre programme informatique, car une activité générique est traitée exactement de la même manière qu'une activité spécifique pour trouver les objets et matériaux associés. L'analyse de la variation des performances en fonction de ce critère est donc simplement réalisable post-expérimentation. Le nombre de configurations à lancer par notre programme tombe donc à (2x9x6=108) 108.

En ce qui concerne la notation, une configuration est créée par la concaténation de valeurs prises par les critères espacés d'un caractère espace. Par exemple, l'une des configurations testées par notre programme est la suivante : *inception_resnet_v2 duckduckgo 100* ; ce qui

entraîne l'extraction d'objets et de matériaux utilisant le modèle *inception_resnet_v2* des 100 premières images extraites du moteur de recherche DuckDuckGo Image.

4.2.3 MESURES DE PERFORMANCES

Enfin, le dernier élément essentiel pour tester la performance de l'approche d'extraction d'objets et de matériaux à partir d'images est le choix des mesures de performance. Puisque pour chaque configuration et activité, notre programme renvoie un ensemble d'objets classés par ordre de pertinence, nous devons utiliser des mesures de performance appropriées. La précision et le rappel sont des mesures de performance à valeur unique basées sur la liste complète des éléments renvoyés par un système. Pour les systèmes qui renvoient une séquence classée d'éléments, il est souhaitable de considérer également l'ordre dans lequel les éléments renvoyés sont présentés. Dans le but de trouver la mesure la plus adaptée et après quelques recherches, nous avons remarqué que trois mesures ressortent et sont souvent utilisées dans les cas similaires au nôtre, à savoir : la précision à k, la R-précision et la *précision moyenne*. La première est une mesure qui peut être pertinente dans certains cas, mais la précision à k présente plusieurs inconvénients (Järvelin & Kekäläinen, 2017; Manning *et al.*, 2008) que les deux autres n'ont pas. La R-précision et la *précision moyenne* étant fortement corrélées (Manning *et al.*, 2008), le choix de l'une ou l'autre ne devrait pas affecter les résultats obtenus. Pour la suite de ce travail, nous avons choisi d'utiliser la *précision moyenne*. Permettant de considérer l'ordre des prédictions sur l'ensemble des éléments prédits, la *précision moyenne* (average precision) ou AP est définie comme l'aire sous la courbe de la précision en fonction du rappel obtenue en calculant la précision et le rappel à chaque position dans la séquence d'éléments classés. La formule pour une activité a_i associée à l'ensemble des objets ordonnés par pertinence O_{a_i} est donnée dans l'équation 4.2.

$$AP(a_i) = \sum_{j=1}^{|O_{a_i}|} (r_j - r_{j-1}) \cdot \left(\frac{p_j + p_{j-1}}{2} \right) \tag{4.2}$$

où r_j dénote le rappel calculé pour les j premiers objets de l'ensemble ordonné O_{a_i} et p_j désigne la précision calculée pour les j premiers objets de l'ensemble ordonné O_{a_i}.

Ces calculs sont illustrés dans le Tableau 4.2 ; on obtient une AP de 0.3474 si on se base sur ce tableau. Si l'on considère la AP comme l'aire sous la courbe précision-rappel, elle peut être calculée de plusieurs façons. En effet, en calcul numérique d'une intégrale, il existe plusieurs méthodes comme la somme de Riemann (Hallett, 2005) et ses formes dérivées telles que la somme de Riemann gauche ou la règle du trapèze [24], c'est cette dernière qui est utilisée dans l'équation (4.2). Le Tableau 4.2 suppose que pour une certaine activité, les seuls objets pertinents sont *cup*, *tea bag* et *spoon*. Notez que si les objets réellement pertinents (*cup*, *tea bag* et *spoon*) étaient les premiers en ordre de pertinence, l'aire sous la courbe serait maximale et vaudrait 1.0. Il est intéressant de noter que la AP ne permet de caractériser la performance que pour une activité spécifique a_i. Afin de connaître la performance globale sur l'ensemble des activités testées A, nous utilisons la mesure de la *moyenne des précisions moyennes* ou MAP (*Mean Average Precision*) qui est la moyenne arithmétique de la AP de chaque activité. Son calcul s'effectue à l'aide de la formule suivante :

$$MAP = \frac{\sum_{a_i \in A} AP(a_i)}{|A|} \tag{4.3}$$

.

Enfin, le dernier point à aborder en relation avec les mesures de performance concerne la manière dont les objets sont considérés comme ayant une relation réelle avec les activités. Pour comprendre cela, il faut d'abord savoir que chaque modèle de classification/détection d'objets

24. Ces règles sont décrites clairement sur https://en.wikipedia.org/wiki/Riemann_sum

Ordre de pertinence	Objets prédits	VP/FP	Précision	Rappel
1	cup	VP	(1/1) 1.00	(1/3) 0.33
2	dog	FP	(1/2) 0.50	(1/3) 0.33
3	tea bag	VP	(2/3) 0.67	(2/3) 0.67
4	plant	FP	(2/4) 0.50	(2/3) 0.67
5	fork	FP	(2/5) 0.40	(2/3) 0.67
6	spoon	VP	(3/6) 0.50	(3/3) 1.00
7	potato	FP	(3/7) 0.43	(3/3) 1.00

Tableau 4.2 : Tableau de calcul pour tracer la courbe précision/rappel avec un exemple de 3 VP (vrais positifs) et 4 FP (faux positifs). Les lignes correspondent aux objets prédits par ordre de pertinence. © Charles Cousyn, 2022.

utilisé a été entraîné par les chercheurs pour reconnaître un nombre limité d'objets. Ainsi, pour un modèle donné, l'ensemble des objets détectables représente une liste fixe d'étiquettes. Dans les modèles que nous utilisons, nous avons 3 listes distinctes d'étiquettes. La première, utilisée par les modèles de classification, est la liste fournie par le projet ImageNet (Deng *et al.*, 2010) qui est disponible en ligne [25] et contient 1001 classes d'objets (1000 classes originales du projet ImageNet et 1 classe *background*). La deuxième liste, utilisée par les modèles *yolov3* et *yolov3-tiny*, utilise le jeu de données COCO [26] contenant 80 classes. La dernière, utilisée par nos modèles *yolo-9000*, contient un peu plus de 9000 classes [27]. La particularité de cette liste est qu'elle est le résultat du mélange de différentes listes. Selon Hui (2019) et Tsang (2019), les jeux de données pour les modèles de détection d'objets ont beaucoup moins de classes que ceux pour la classification. Pour étendre les classes que yolov2 (Redmon & Farhadi, 2017) peut détecter, yolo-9000 propose une méthode permettant de mélanger les images des ensembles de données de détection d'objets et de classification pendant la phase d'apprentissage à l'aide d'un arbre hiérarchique, comme le montre la Figure 4.3. En raison du nombre limité de classes

25. https://storage.googleapis.com/download.tensorflow.org/data/ImageNetLabels.txt

26. https://gist.github.com/AruniRC/7b3dadd004da04c80198557db5da4bda

27. https://github.com/pjreddie/darknet/blob/1e729804f61c8627eb257fba8b83f74e04945db7/data/9k.names

détectables pour nos modèles, ceux-ci se retrouvent incapables de détecter des objets pour lesquels ils n'ont pas été entraînés. Sachant cela, nous devons faire un choix quant à la manière de comparer les objets extraits avec le jeu de données de référence dont nous avons parlé précédemment. En effet, nous devons nous poser la question de savoir comment interpréter le fait qu'un objet soit absent des objets extraits (faux négatif) alors que le réseau était incapable de le détecter a priori (faux négatif a priori). Par exemple, l'objet *disinfectant* est considéré comme lié à l'activité *fill medicine cabinet* dans notre jeu de données de référence, mais ce terme n'existe pas dans la liste d'objets d'ImageNet. Dans un cas comme celui-ci, deux possibilités se présentent à nous. Soit nous considérons ce faux négatif comme une véritable erreur, car il met en évidence l'incapacité des modèles comportant trop peu de classes à fournir tous les objets pertinents, soit nous ignorons ce faux négatif. Notre équipe a décidé de mettre en place un compromis entre ces deux alternatives : nous ignorons ce que nous appelons les *faux négatifs a priori*, mais nous calculons un taux d'objets reconnaissables (*Recognizable Objects Rate* ou ROR) pour chaque couple (type de modèle, activité). Ce taux représente en fait la proportion d'objets reconnaissables d'une activité pour un type de modèle donné (classification, yolov3 ou yolo-9000) parmi les objets liés à l'activité fournis dans le jeu de données de référence. Grâce à ce compromis, nous pouvons obtenir un aperçu de la capacité d'un modèle à trouver des objets, mais aussi de sa limitation par les classes pour lesquelles il a été entraîné. Nous procédons à l'analyse du taux dans la section suivante.

4.3 GRAPHIQUES ET ANALYSES

Une fois que toutes les expérimentations ont été réalisées pour nos 108 configurations possibles, les résultats obtenus contiennent les objets et/ou matériaux extraits pour toutes les configurations et sont compilés dans des fichiers au format JSON. Ces fichiers, au nombre de 108, seraient trop longs pour être comparés manuellement avec notre jeu de données

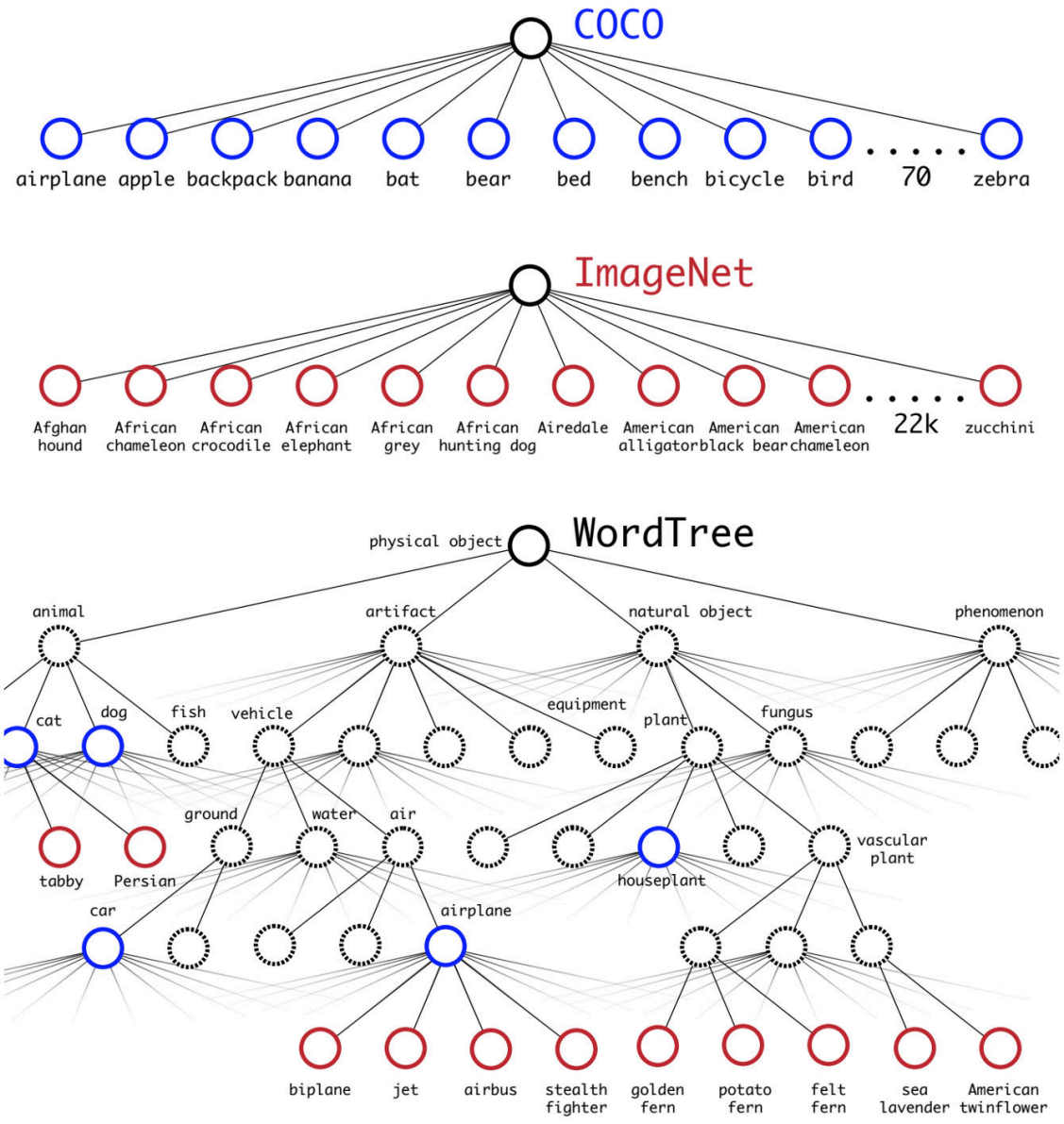

Figure 4.3 : Combinaison des étiquettes de COCO et ImageNet dans un arbre hiérarchique (Redmon & Farhadi, 2017). © 2017 IEEE.

de référence. Nous avons donc développé un outil de visualisation utilisant la bibliothèque Chart.js (Etimberg *et al.*, 2020) et permettant d'effectuer quelques calculs et d'afficher les différentes mesures de performance selon nos critères sous forme de courbe.

4.3.1 LE TAUX D'OBJETS RECONNAISSABLES

Afin d'avoir une idée des valeurs du ROR pour chaque couple (type de modèle, activité) soit 54 (3×18) couples possibles, nous avons calculé la moyenne et l'écart-type du ROR pour toutes les activités par type de modèle et créé une représentation graphique dans la Figure 4.4. L'observation de cette figure montre, sans surprise, que le nombre de classes détectables par un modèle est positivement corrélé avec le ROR. Cependant, il faut noter que la différence de taux entre les modèles de classification et les modèles yolov3 (0.3387-0.2348=0.1039) reste faible par rapport à la différence du nombre de classes (1001-80=921). En d'autres termes, nous pourrions dire que le fait d'avoir 921 classes de plus ne semble donner que 0.1039 ROR de plus. Cela suggère que les classes du jeu de données COCO sont presque aussi diverses que celles du jeu de données ImageNet.

4.3.2 ANALYSE PAR CRITÈRE

LE CRITÈRE DU MOTEUR DE RECHERCHE

Comme décrit dans la Section 4.2.2, le premier critère que nous considérons est le choix du moteur de recherche à utiliser pour cette tâche. Notre outil est capable de combiner plusieurs MAP obtenues à partir de différentes configurations en deux nombres ; la moyenne arithmétique nommée *average MAP*, et l'écart-type nommé *standard deviation MAP*. Dans ce cas, la *average MAP* et la *standard deviation MAP* sont calculées en regroupant les

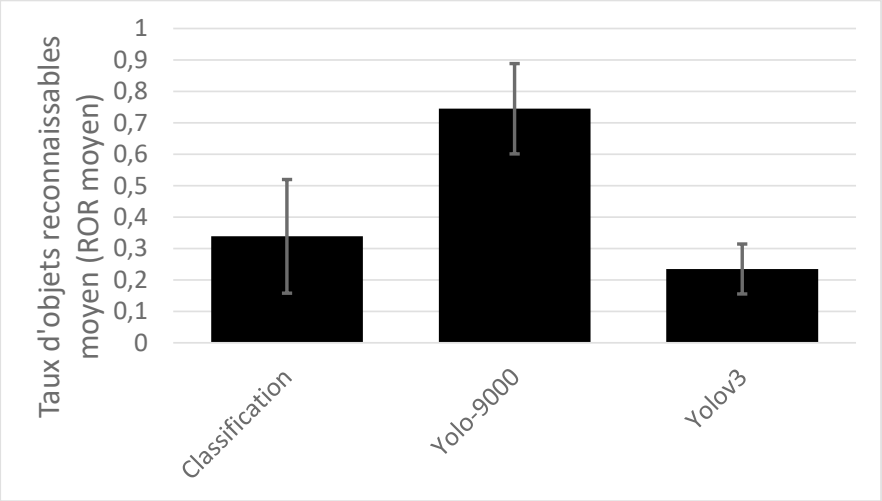

Figure 4.4 : Taux moyen d'objets reconnaissables selon le type de modèle utilisé. © Charles Cousyn, 2022.

configurations ayant le même moteur de recherche (parmi les 108 existantes). Nous nommons le résultat *average MAP par moteur de recherche* et *standard deviation MAP par moteur de recherche*. Les résultats suivants sont obtenus. Le moteur de recherche d'images Google a une *average MAP par moteur de recherche* de 0.4938 alors que le moteur de recherche d'images DuckDuckGo a obtenu 0.5104. Même si ces moyennes semblent légèrement différentes à première vue, il est important de noter que les valeurs de *standard deviation MAP par moteur de recherche* sont bien trop élevées pour affirmer l'existence d'une quelconque différence de performances entre les deux moteurs de recherche : 0.2676 pour DuckDuckGo Image et 0.2584 pour Google Image. Ces données sont présentées dans la Figure 4.5.

La Figure 4.5 donne une vue d'ensemble des performances pour chaque moteur de recherche, cependant il peut être intéressant d'étudier les performances des 108 configurations existantes pour identifier des groupes de configurations plus performantes que d'autres. La Figure 4.6 illustre cela en affichant les 108 configurations, mais en regroupant visuellement les configurations identiques lorsque la valeur du critère étudié est ignorée. Par exemple, comme

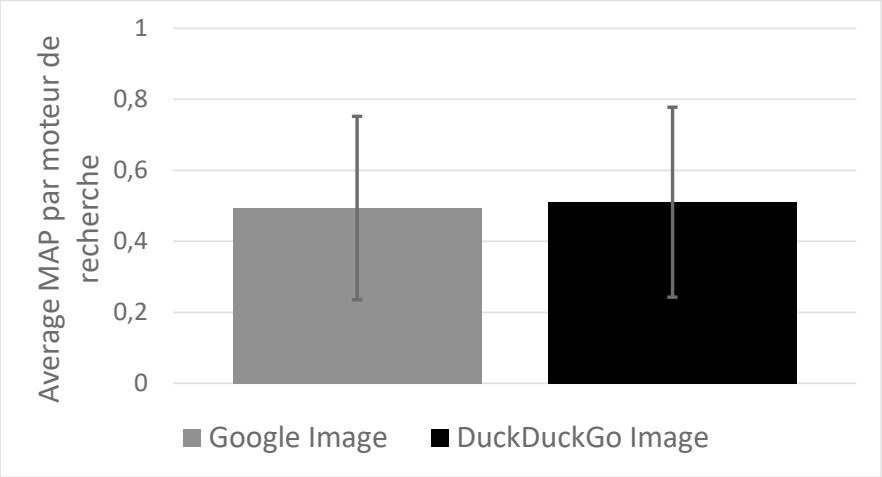

Figure 4.5 : Average MAP par moteur de recherche. © Charles Cousyn, 2022.

nous nous intéressons au critère du moteur de recherche, les configurations *inceptionresnetv2 duckduckgo 100* et *inceptionresnetv2 google 100* sont regroupées sous la valeur en abscisse *inceptionresnetv2 100*; alias *ir2 100*. L'observation de la figure permet de constater qu'il existe 3 groupes de modèles distincts dans les 108 configurations. Le groupe le plus à gauche correspond aux configurations utilisant des modèles de classification, le groupe du milieu correspond aux configurations utilisant le modèle de détection d'objets *yolo-9000* et le groupe le plus à droite correspond aux configurations utilisant les modèles de détection d'objets *yolov3-608* et *yolov3-tiny*. Afin d'avoir une idée de la performance en fonction du moteur de recherche et du type de modèle, la Figure 4.7 montre la *average MAP par couple (moteur de recherche, groupe de modèles)* et la *standard deviation MAP par couple (moteur de recherche, groupe de modèles)* pour chaque groupe précédemment identifié. Cette figure confirme que les 3 groupes de modèles identifiés sont réellement différents. Un classement des configurations peut être établi en fonction du groupe de modèles utilisé. Les configurations les moins performantes utilisent le modèle *yolo9000* avec une *average MAP* de 0.0436 pour DuckDuckGo Image et 0.0449 pour Google Image. Les configurations les plus performantes utilisent les modèles *yolov3-608* et *yolov3-tiny* avec une *average MAP* de 0.7259 pour DuckDuckGo Image et

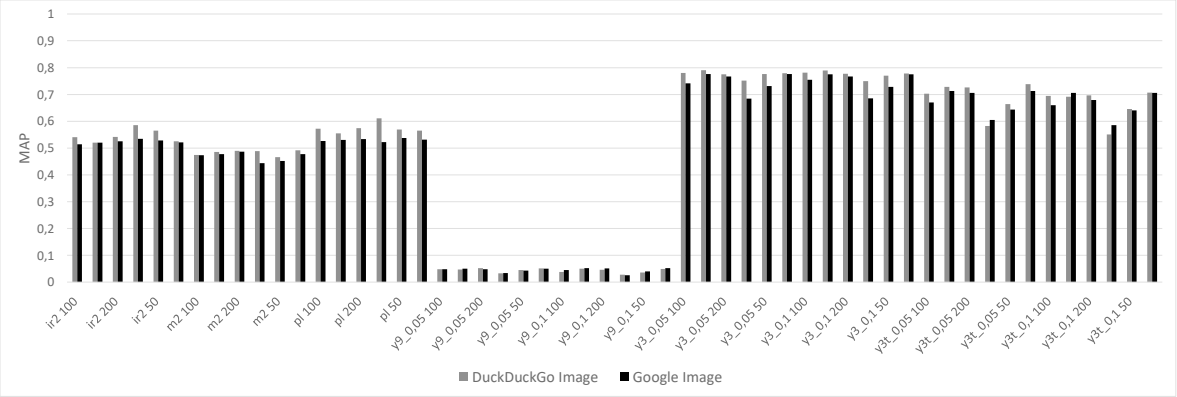

Figure 4.6 : MAP de 108 configurations affichées en regroupant visuellement les configurations identiques lorsque la valeur du moteur de recherche est ignorée. © Charles Cousyn, 2022.

de 0.7079 pour Google Image. Les configurations intermédiaires sont celles qui utilisent les modèles de classification avec une *average MAP* de 0.5344 pour DuckDuckGo Image et de 0.5075 pour Google Image.

LE CRITÈRE DU MODÈLE DE CLASSIFICATION/DÉTECTION D'OBJETS

Le critère suivant que nous voulons analyser est le modèle de classification/détection d'objets utilisé pour extraire les objets et les matériaux des images. La première chose que nous avons faite a été de déterminer si ce critère avait un impact global sur les performances. Pour ce faire, nous avons produit la Figure 4.8 qui exprime la *average MAP par modèle utilisé* et la *standard deviation MAP par modèle utilisé*. L'observation de cette figure nous permet, tout d'abord, d'affirmer que certains modèles fonctionnent réellement mieux que d'autres avec une nette supériorité des modèles *yolov3*. Une autre remarque que l'on peut faire est qu'il n'est pas évident qu'il y ait une valeur du seuil de confiance meilleure que l'autre (0.05 et 0.1).

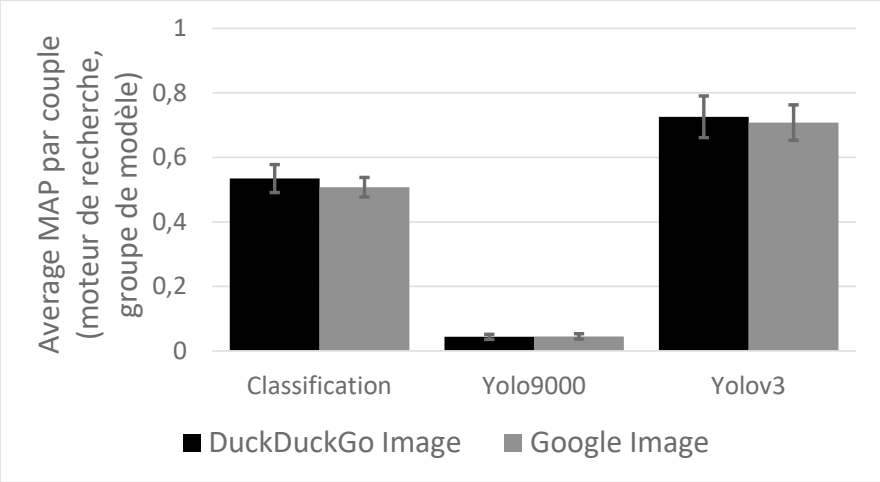

Figure 4.7 : Average MAP calculée pour chaque couple (moteur de recherche, groupe de modèle). © Charles Cousyn, 2022.

De la même manière que dans la Figure 4.6 pour le critère du moteur de recherche, il pourrait être intéressant d'examiner les 108 configurations existantes pour essayer d'identifier des groupes de configurations plus performantes que les autres. La Figure 4.9 illustre cette approche en affichant les 108 configurations, mais en regroupant visuellement les configurations identiques lorsque la valeur du modèle de classification/détection d'objet utilisé est ignorée. Outre les remarques déjà faites pour la Figure 4.6, la Figure 4.9 ne semble pas permettre d'identifier clairement de nouveaux regroupements.

LE CRITÈRE DU NOMBRE D'IMAGES

Le troisième critère concerne le nombre d'images utilisées pour extraire les objets et matériaux liés aux activités de la vie quotidienne. Avant de passer aux résultats obtenus, il convient de noter que ce critère nécessite une interprétation particulière. En effet, les valeurs du nombre d'images correspondent en réalité au nombre d'images souhaitées lors de l'envoi de la requête au moteur de recherche. Les résultats fournis par un moteur de recherche étant limités

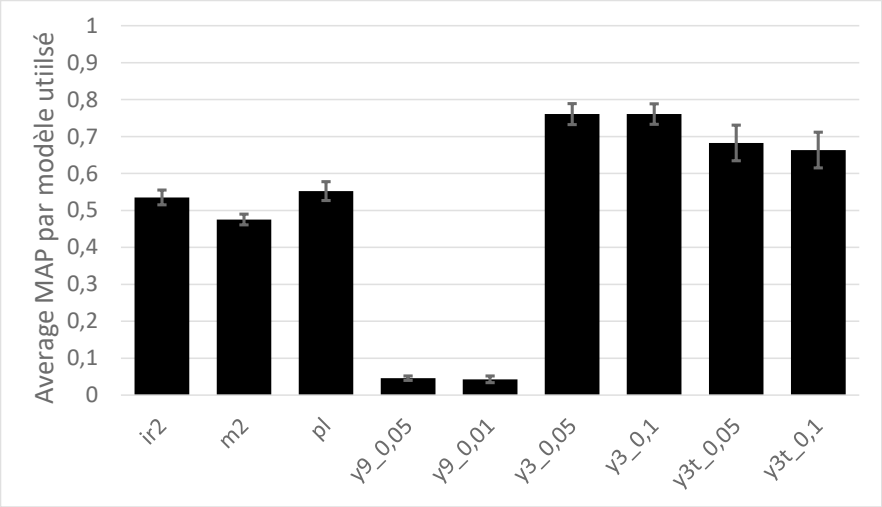

Figure 4.8 : Average et standard deviation MAP par modèle utilisé. © Charles Cousyn, 2022.

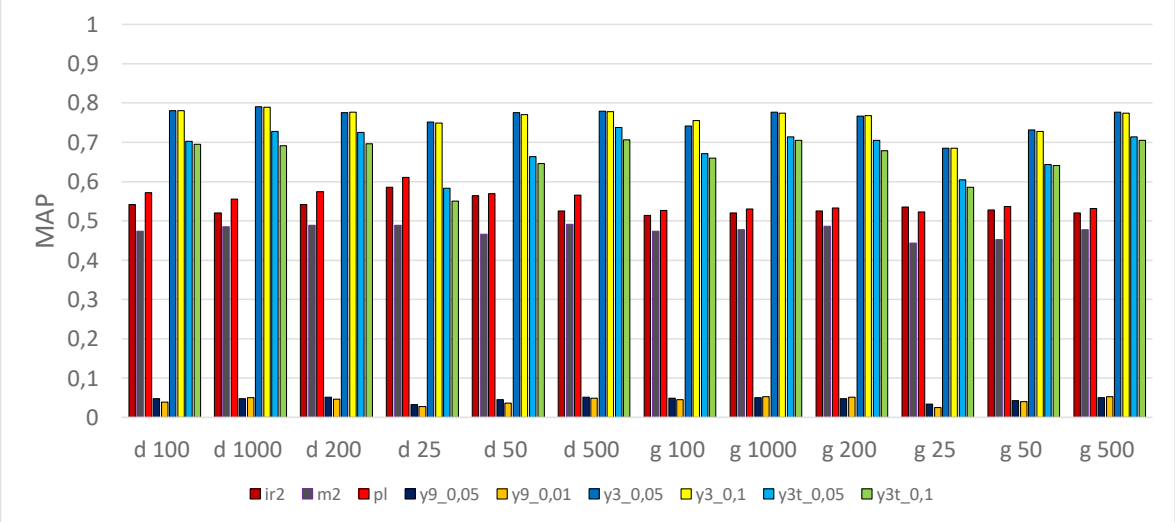

Figure 4.9 : MAP de 108 configurations affichées en regroupant visuellement les configurations identiques lorsque la valeur du modèle est ignorée. © Charles Cousyn, 2022.

et variables, la requête d'une activité peut en effet renvoyer moins d'images que souhaité. La Figure 4.10 représente le nombre d'images disponibles pour chaque activité par moteur de recherche. La figure montre que toutes les requêtes ont abouti à au moins 200 images, mais que c'est différent pour des nombres d'images plus élevés. Pour les requêtes effectuées sur Google Image, 17/18 activités ont plus de 500 images et aucune d'entre elles n'a atteint 1000 images. De même, pour les requêtes effectuées sur DuckDuckGo Image, 16/18 activités ont plus de 500 images et aucune d'entre elles n'a atteint 1000 images. Cela signifie que le nombre maximum d'images disponibles se situe très souvent entre 500 et 1000. Dans cette situation où il est possible de manquer d'images, nous avons plusieurs choix. Pour un nombre d'images utilisées donné (25, 50, 100, 200, 500, 1000), soit on exclut les activités qui n'ont pas le nombre requis, soit on décide de les inclure quand même. Dans le premier cas, la valeur de 1000 images n'a plus de sens, car aucune activité n'a plus de 1000 images et garder les activités avec au moins 500 images implique de considérer moins d'activités que les 18 activités initiales. Dans le second cas, les configurations utilisant 500 et 1000 images doivent être considérées comme ayant un bruit supplémentaire à l'écart-type que nous pourrions calculer, car pour les activités concernées, ces configurations ne fourniront que des résultats basés sur un nombre d'images plus petit que prévu, ce qui biaise ce qu'on attendrait de 500 ou 1000 images. C'est cette deuxième option que nous avons choisie.

Dans ce contexte, en calculant la *average MAP par le nombre d'images utilisées* et la *standard deviation MAP par le nombre d'images utilisées* sur les 108 configurations pour chaque valeur du nombre d'images utilisées (25, 50, 100, 200, 500 et 1000), nous obtenons la Figure 4.11. En observant cette figure, une très légère pente positive semble apparaître, suggérant que plus nous utilisons d'images, meilleures sont les performances. De plus, pour le dernier groupe de configuration (1000 images), la performance est très légèrement inférieure à celle du groupe de configuration avec 500 images. Cependant, l'écart-type pour chaque valeur

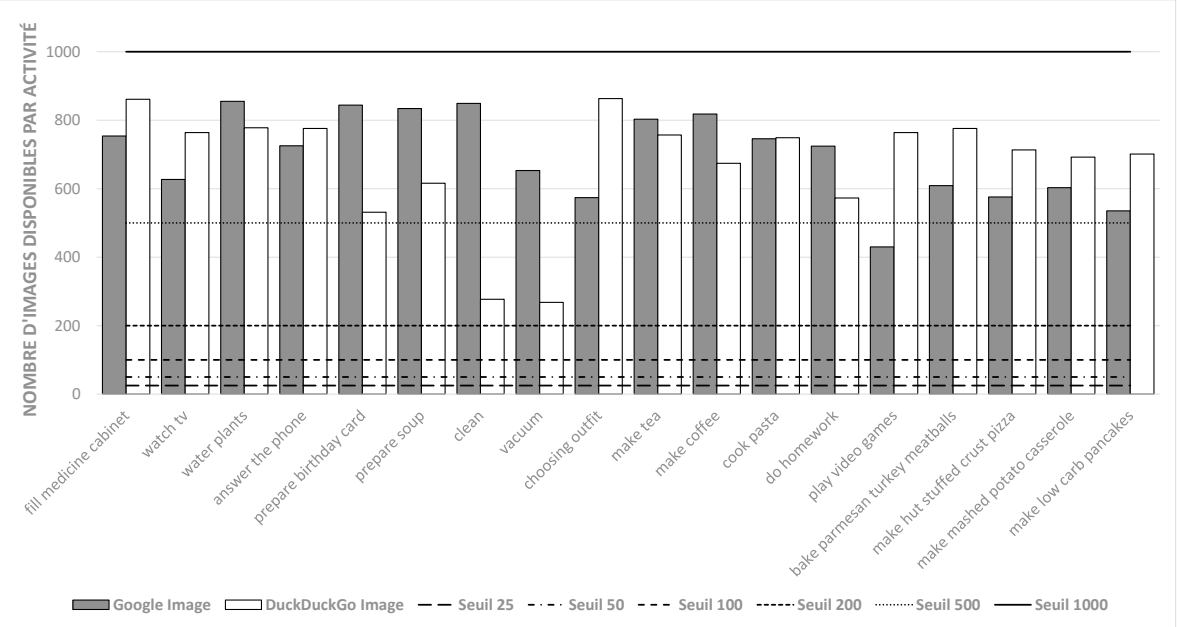

Figure 4.10 : Nombre d'images disponibles par activité et par moteur de recherche. © Charles Cousyn, 2022.

est élevé ce qui signifie que la pente n'est pas significative et que chaque valeur est équivalente jusqu'à preuve du contraire. De toutes ces remarques, on peut déduire qu'en moyenne, le nombre d'images utilisées ne semble pas affecter significativement les performances dans la gamme $[25, 1000]$.

Néanmoins, il serait également intéressant d'examiner plus en détail s'il existe des groupes de configurations qui fonctionnent plus ou moins bien en fonction d'autres critères tels que le modèle de classification/détection d'objets utilisé ou le moteur de recherche. Afin d'explorer cette piste, la Figure 4.12 illustre la MAP des 108 configurations affichées en regroupant visuellement les configurations identiques lorsque la valeur du nombre d'images est ignorée. Outre le fait de voir séparément les 3 groupes précédemment mentionnés (classification, yolov3 et yolo-9000), la figure suggère que, dans le cas des configurations utilisant des

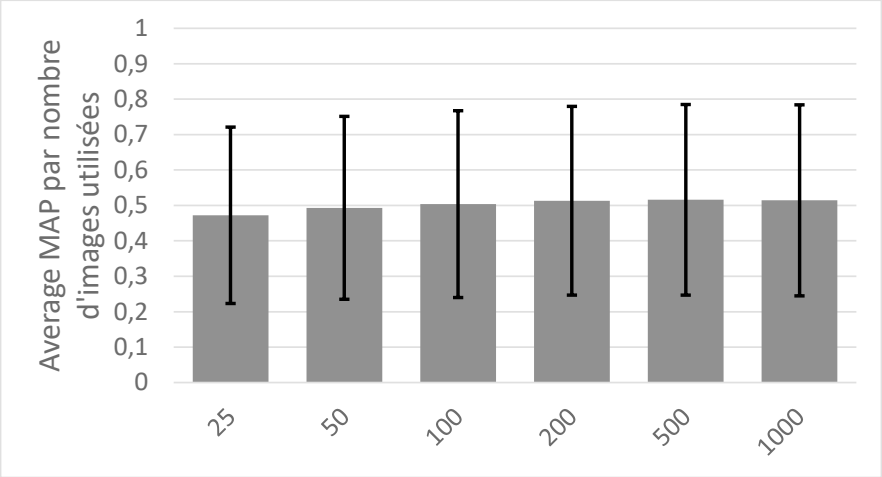

Figure 4.11 : Average MAP par nombre d'images utilisées. © Charles Cousyn, 2022.

modèles de classification, la performance diminue avec l'augmentation du nombre d'images utilisées et que, dans le cas des configurations utilisant des modèles de détection d'objets, la performance augmente avec l'augmentation du nombre d'images utilisées. Afin de vérifier cela, nous avons pu, pour chaque valeur du nombre d'images, afficher la performance moyenne de chacun de ces groupes. La Figure 4.13 illustre cela en donnant la *average MAP par couple (modèle utilisé, nombre d'images utilisées)*. Deux remarques peuvent être faites à ce sujet. La première est qu'il semble effectivement, dans le cas de l'utilisation des modèles yolov3 et yolo-9000, qu'il y ait une légère augmentation des performances lorsque le nombre d'images utilisées augmente. La seconde est que, dans le cas des modèles de classification, la diminution des performances avec l'augmentation du nombre d'images utilisées n'est pas significative en termes d'écarts-types.

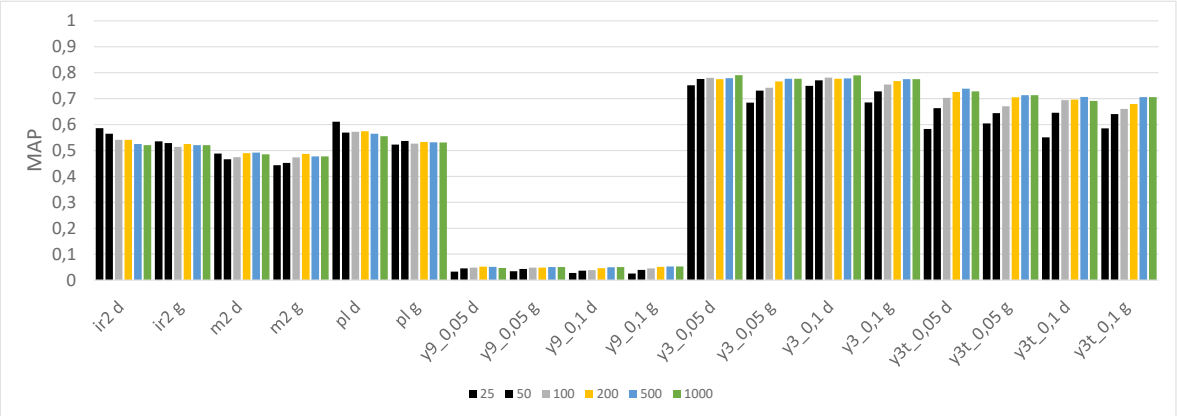

Figure 4.12 : MAP des 108 configurations affichées en regroupant visuellement les configurations identiques lorsque la valeur du nombre d'images est ignorée. © Charles Cousyn, 2022.

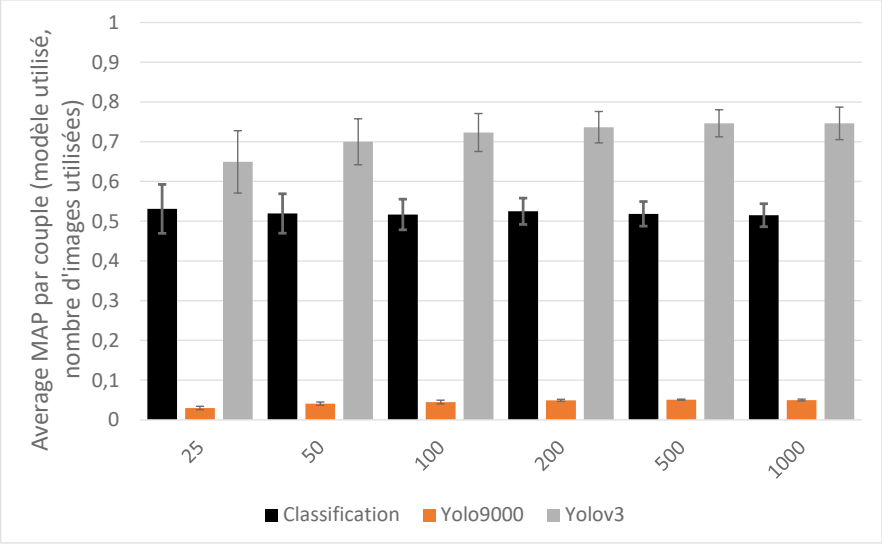

Figure 4.13 : Average MAP par couple (modèle utilisé, nombre d'images utilisées). © Charles Cousyn, 2022.

LE CRITÈRE DE LA GRANULARITÉ

Le dernier critère étudié est l'impact de la granularité d'une activité. Pour rappel, la granularité d'une activité est définie comme étant soit spécifique soit générique, et la répartition entre nos 18 activités est de 4 activités *spécifiques* et 14 activités *génériques*. Les activités spécifiques sont exprimées de manière plus détaillée, tandis que les activités génériques sont plus sujettes à interprétation. De la même manière que pour les autres critères, la performance moyenne pour chaque niveau de granularité est calculée. Cependant, cette fois-ci, la *average MAP par niveau de granularité* et la *standard deviation MAP par niveau de granularité* sur les 108 configurations ne sont pas les mesures préférées, car la mesure de la MAP est une performance calculée pour l'ensemble des 18 activités. Ce qui nous intéresse, dans ce cas, c'est de prendre chaque valeur de la AP pour chaque activité dans chaque configuration, de classer ces valeurs selon qu'elles proviennent d'une activité générique ou spécifique, et de faire une moyenne globale sur toutes les configurations. Nous calculons donc la *average AP par niveau de granularité* et la *standard deviation AP par niveau de granularité*. La Figure 4.14 illustre ces 2 mesures avec une *average AP par niveau de granularité* de 0.5282 pour les activités génériques et 0.4107 pour les activités spécifiques, et avec une *standard deviation AP par niveau de granularité* de 0.2804 pour les activités génériques et 0.3040 pour les activités spécifiques. L'observation de cette figure ne semble pas montrer la supériorité de la performance d'un niveau de granularité sur l'autre.

Se demander si le niveau de granularité peut avoir un impact est intrinsèquement lié à la remise en cause du fait que certaines activités sont plus performantes que d'autres. Pour s'en faire une idée, il suffit de vérifier les performances moyennes sur toutes les configurations par activité. La Figure 4.15 nous montre ces moyennes. L'observation de ces dernières montre une grande variabilité des performances pour chacune des activités et hormis la nette supériorité

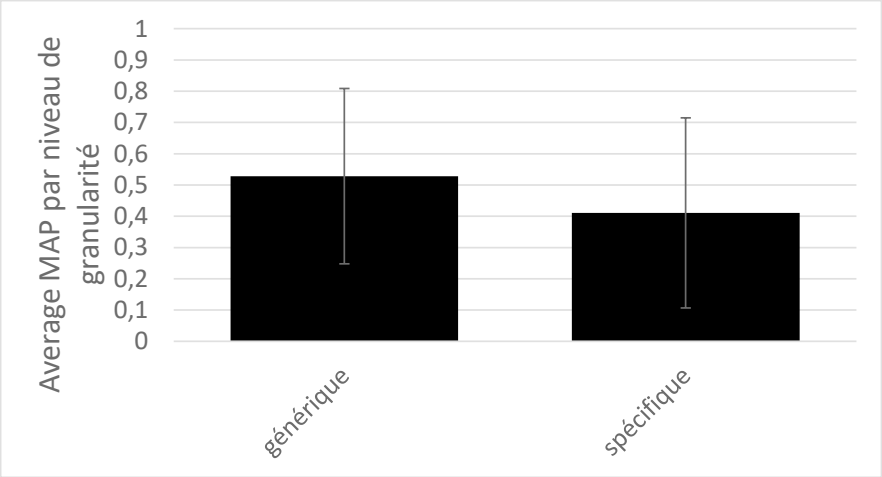

Figure 4.14 : Average AP par niveau de granularité (*générique* ou *spécifique*). © Charles Cousyn, 2022.

de l'activité *make stuffed crust pizza* sur l'activité *bake parmesan turkey meatball*, il semble difficile d'établir un classement par performance des activités en raison de cette variabilité.

4.3.3 LA MEILLEURE CONFIGURATION

Les sections précédentes donnent une vision globale de l'approche en fournissant un point de vue sur le comportement moyen des performances en fonction de tous les paramètres existants (moteur de recherche, modèle de classification/détection d'objets, nombre d'images utilisées, granularité). Dans ce travail, nous avons testé 108 configurations différentes selon les critères que nous avons définis. L'une des choses sur lesquelles nous devons ensuite nous concentrer est de savoir quelle est la meilleure configuration. Pour ce faire, nous devons d'abord définir comment dire qu'une configuration est meilleure qu'une autre. La première chose qui vient à l'esprit est de calculer la MAP pour chaque configuration et de quantifier la qualité d'une configuration par cette seule moyenne. Cependant, il est important de garder à l'esprit que chaque activité est liée à un ROR qui reflète les limites du modèle de classification/détection

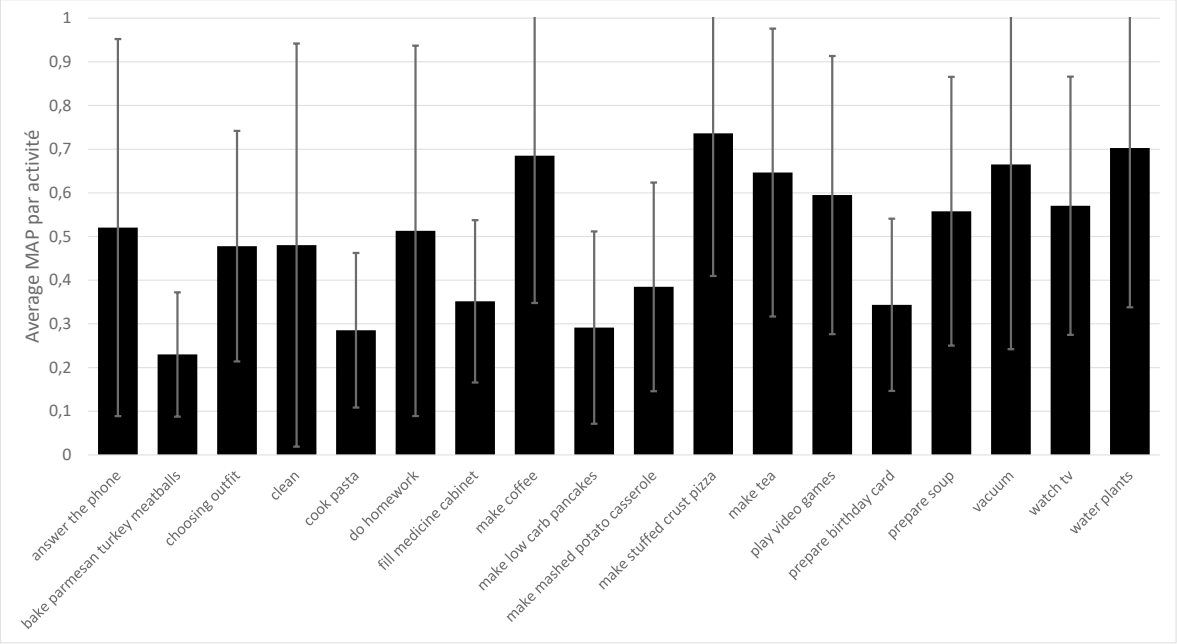

Figure 4.15 : Average AP par activité. © Charles Cousyn, 2022.

d'objets utilisé. Cela signifie que pour évaluer correctement la performance d'une configuration sur une activité, il est nécessaire de prendre en compte ce taux. Dans cet objectif, nous avons choisi d'utiliser une MAP pondérée, nommée WMAP, qui est la moyenne arithmétique pondérée des AP pour une configuration où les poids correspondent aux ROR des 18 activités pour cette configuration particulière. L'équation (4.4) donne la formule de WMAP pour une configuration. Les 10 meilleures configurations sont données dans le Tableau 4.3.

$$WMAP = \frac{\sum_{a_i \in A} AP(a_i) \cdot ROR(a_i)}{\sum_{a_i \in A} ROR(a_i)}. \tag{4.4}$$

Avec la grande variabilité que nous avons trouvée dans l'analyse de chacun des critères, il n'est pas facile de porter un jugement sur la supériorité d'une configuration sur toutes les autres. Néanmoins, le tableau semble nous montrer que les meilleures configurations utilisent le modèle de détection d'objets yolov3-608 et un nombre d'images utilisées supérieur ou égal

Configuration	WMAP
yolov3-608__20_0.05_0.5 duckduckgo 1000	0.7832
yolov3-608__20_0.1_0.5 duckduckgo 1000	0.7818
yolov3-608__20_0.1_0.5 duckduckgo 100	0.7755
yolov3-608__20_0.05_0.5 duckduckgo 100	0.7734
yolov3-608__20_0.05_0.5 google 1000	0.7723
yolov3-608__20_0.05_0.5 google 500	0.7723
yolov3-608__20_0.05_0.5 duckduckgo 50	0.7722
yolov3-608__20_0.05_0.5 duckduckgo 500	0.7715
yolov3-608__20_0.1_0.5 duckduckgo 500	0.7712
yolov3-608__20_0.1_0.5 google 1000	0.7711

Tableau 4.3 : Tableau des 10 meilleures configurations. © Charles Cousyn, 2022.

à 100 ; le moteur de recherche utilisé a probablement peu d'impact comme nous l'avons vu dans la Section 4.3.2.

4.3.4 DISCUSSION DES RÉSULTATS OBTENUS

L'évaluation des objets extraits pour chaque activité selon nos critères peut être résumée comme suit. Premièrement, il ne semble pas y avoir de différence significative entre l'utilisation du moteur de recherche d'images Google Image et DuckDuckGo Image. Deuxièmement, l'utilisation des modèles de détection d'objets de yolov3 semble préférable par rapport aux autres modèles et cela même si le nombre d'objets détectables est plus faible. Troisièmement, le nombre d'images utilisées ne semble n'avoir qu'un très faible impact sur les performances. Nous pensons qu'il est possible que l'utilisation de 25 images soit plus que suffisante au vu des résultats obtenus dans d'autres travaux utilisant seulement 10 images (Riboni & Murtas, 2017, 2019). En ce qui concerne la granularité, il ne semble pas que les activités spécifiques aient des performances différentes des activités génériques. Plus généralement, en considérant les activités choisies pour ce chapitre, les performances obtenues sont, en moyenne sur toutes

les configurations, très similaires et il est difficile de déterminer qu'une activité obtient des résultats significativement meilleurs.

Néanmoins, dans le cas d'une configuration particulière, il peut être intéressant de se demander pourquoi certaines activités sont moins performantes que d'autres. Prenons comme exemple la configuration présentant les meilleures performances dans le Tableau 4.3, la configuration *yolov3-608__20_0.05_0.5 duckduckgo 1000*. Pour cette configuration, l'activité la moins performante est l'activité *cook pasta* avec un AP de 0.3028 et un ROR de 0.2308. En considérant uniquement les objets détectables par le modèle de détection d'objets yolov3-608, les objets pertinents sont : *spoon*, *fork* et *cup*. Le Tableau 4.4 donne les prédictions fournies par la configuration *yolov3-608__20_0.05_0.5 duckduckgo 1000* ainsi que la précision et le rappel associés. La première remarque est qu'il est évident que ces objets ne sont pas assez nombreux et ne caractérisent pas spécifiquement l'activité *cook pasta*. La raison en est expliquée par le ROR : environ 73% des objets véritablement liés à l'activité ne sont pas détectables par le réseau yolov3-608. La deuxième remarque est que le fait qu'il y ait peu d'objets détectables signifie que même si les objets pertinents sont prédits dans le top 8, comme dans le cas de l'activité *cook pasta*, la présence de faux positifs (comme *bowl*, *carrot*, etc.) avant les objets pertinents nuit fortement aux performances.

Ces remarques sont la conséquence d'au moins une des possibilités suivantes. La première possibilité est que l'ensemble de données de référence initial ne contenait pas certains objets pertinents pour l'activité *cook pasta*. La seconde est que les objets pertinents de l'activité *cook pasta* ne font pas partie des 80 objets détectables par le modèle de détection d'objets yolov3. La dernière est que les moteurs de recherche d'images ne montrent pas les objets significatifs dans les images récupérées. Sur la base de l'étude de notre jeu de données de référence et des images récupérées, il semblerait que l'explication la plus plausible soit la

Ordre de pertinence	Objets prédits	VP/FP	Précision	Rappel
1	bowl	FP	(0/1) 0	(0/3) 0
2	carrot	FP	(0/2) 0	(0/3) 0
3	broccoli	FP	(0/3) 0	(0/3) 0
4	person	FP	(0/4) 0	(0/3) 0
5	spoon	VP	(1/5) 0.2	(1/3) 0.33
6	fork	VP	(2/6) 0.33	(2/3) 0.67
7	diningtable	FP	(2/7) 0.29	(2/3) 0.67
8	cup	VP	(3/8) 0.38	(3/3) 1
...	...	...	...	...

Tableau 4.4 : Prédiction et performances de l'activité *cook pasta* avec la configuration *yolov3-608__20_0.05_0.5 duckduckgo 1000*. © Charles Cousyn, 2022.

seconde, à savoir que les objets et matériaux importants tels que *pasta*, *water* sont absents des objets détectables par le modèle yolov3.

4.3.5 CONCLUSION ET RÉPONSES AUX QUESTIONS DE RECHERCHE

Les différents résultats présentés dans cette section ainsi que dans les sections précédentes apportent des réponses aux trois questions de recherche énoncées au début du chapitre. Tout d'abord, nous nous sommes intéressés à la capacité des images issues des moteurs de recherche à représenter les activités de la vie quotidienne par rapport aux approches existantes. D'une part, en comparant les performances obtenues pour chacune des activités considérées, nous nous sommes rendu compte qu'il existe une grande variabilité entre les activités, ce qui signifie qu'il y a encore beaucoup de marge de progression, mais qu'au moins une partie des objets réellement liés aux activités sont actuellement trouvés. D'autre part, la représentation de nos activités sous la forme d'une liste ordonnée d'objets fournie par notre approche semble plus lisible, plus compréhensible et plus généralisable que sous la forme abstraite d'un modèle classique d'apprentissage supervisé.

Ensuite, nous nous sommes intéressés à la façon dont nous pouvons trouver et exploiter de manière automatisée les images disponibles sur le web relatives aux activités de la vie quotidienne. Nous avons montré que nous sommes capables d'obtenir jusqu'à 1000 images automatiquement en utilisant une requête appropriée sur deux moteurs de recherche populaires et efficaces. Nous avons également montré qu'il est possible d'utiliser des modèles de classification/détection d'objets pour automatiser l'extraction d'objets à partir d'images en agrégeant les objets détectés. Cependant, les modèles de classification/détection d'objets ne peuvent reconnaître que ce pour quoi ils ont été entraînés et la proportion de ce qu'ils peuvent reconnaître reste limitée, ne dépassant pas 75% en moyenne pour yolo9000.

CHAPITRE V
EXTRACTION DE MOTIFS SÉQUENTIELS DU WEB POUR LA RECONNAISSANCE DES ACTIVITÉS HUMAINES

Dans le précédent chapitre, nous nous sommes intéressés à l'utilisation d'images provenant du web pour représenter les activités de la vie quotidienne. Nos résultats ont montré qu'il était possible d'extraire des listes d'objets satisfaisantes pour des activités de la vie quotidienne, mais que la plupart de ces listes ne sont pas complètes ; la raison principale étant que les modèles de classification ou de détection d'objets limitent le nombre d'objets détectables. Ces listes d'objets sont évidemment intéressantes en raison de leur manière de représenter les activités de la vie quotidienne, mais elles ne contiennent aucune information en rapport avec la chronologie de l'utilisation des objets. Nous savons que l'ordre dans lequel les objets sont utilisés peut s'avérer important pour différencier des activités utilisant des objets similaires. Par exemple, on pourrait imaginer qu'utiliser de l'eau puis un verre pourrait être une indication que nous sommes en train de faire la vaisselle alors qu'utiliser un verre puis de l'eau pourrait signifier que nous sommes en train de remplir un verre avec de l'eau.

Cette thèse s'intéressant à ce que le web peut apporter pour ce problème, le but de ce chapitre est de répondre à la question de recherche suivante : dans quelle mesure est-il possible d'extraire de manière non supervisée l'ordre d'utilisation des objets liés à la réalisation d'une activité à partir du web ? Pour ce faire, le contenu du présent chapitre est basé sur notre article soumis dans le journal *IEEE Transactions on Knowledge and Data Engineering*.

Notre contribution théorique est une approche non supervisée utilisant le web pour trouver des informations pertinentes sur les activités de la vie quotidienne. La contribution pratique est un outil capable d'extraire des motifs séquentiels à partir de pages web ; l'ensemble

du code est disponible sur notre dépôt Github [28]. Enfin, la contribution expérimentale est l'analyse statistique des performances de la méthode appliquée à un jeu de données de pages web, pour 18 activités de la vie quotidienne, les mêmes utilisées que dans le chapitre précédent.

5.1 DÉCOUVERTE DE MOTIFS ET FORAGE DE MOTIFS SÉQUENTIELS

La découverte de motifs est le processus consistant à identifier et révéler des motifs dans un ensemble de données et représente un domaine de recherche important, notamment dans la discipline de la fouille de textes (Aswini & Lavanya, 2014; Yu, 2018; Ning Zhong *et al.*, 2012; Ningrum *et al.*, 2020). Dans le cas de la découverte de motifs dans un texte, un motif est très souvent un ensemble d'un ou plusieurs termes qui se répètent dans un texte ou un ensemble de textes avec une certaine fréquence. Un premier exemple d'utilisation peut se voir dans les travaux de Yu (2018). Dans celui-ci, les auteurs utilisent un modèle d'espace vectoriel, la similarité cosinus et un outil de visualisation de graphes pour découvrir les tendances de développement dans les rapports de projet d'un certain domaine pendant six années consécutives. Dans le même travail, ils ont également conçu des outils de visualisation pour observer les points chauds ou tendances de la recherche avec des clusters. Un autre exemple est donné dans le travail de Elleuch *et al.* (2020) où les auteurs utilisent le prétraitement de texte, WordNet (Princeton University, 2010a) et un algorithme personnalisé de découverte de motifs pour déterminer les activités fréquentes dans les processus commerciaux à partir de courriels de manière non supervisée. Les auteurs avancent que l'avantage de cette méthode est qu'elle nécessite peu d'intervention humaine et qu'elle est capable d'utiliser des courriels dont le contenu n'est pas structuré et qui contiennent beaucoup d'informations pertinentes sur les processus d'affaires.

28. https://github.com/CharlesCousyn/webpages_objects_spm_extractor

La fouille de motifs séquentiels ou *sequential pattern mining* (SPM) est une méthode de découverte de motifs qui a la particularité d'utiliser une base de données avec des transactions ordonnées (que ce soit par un critère temporel ou autre) et qui permet de trouver les séquences les plus fréquentes d'éléments qui se suivent. Le SPM est peu souvent appliqué au texte (Maylawati *et al.*, 2018) dans la littérature. Il a surtout été utilisé pour obtenir une représentation structurée du texte en essayant d'extraire les séquences importantes et parce qu'il permet de comprendre les relations entre les mots/expressions et les phrases (Doucet & Ahonen-Myka, 2004; Aswini & Lavanya, 2014; Maylawati & Saptawati, 2017). Par exemple, dans les travaux de Doucet & Ahonen-Myka (2004), les auteurs ont montré que l'extraction des séquences fréquentes maximales (maximale signifiant qu'aucune autre séquence fréquente ne contient cette séquence) dans les documents permet de les rechercher plus efficacement qu'en utilisant un simple modèle de sac de mots (Manning, 2008). En utilisant un modèle d'espace vectoriel, la mesure TF-IDF et les séquences maximales fréquentes extraites, ils montrent que leur méthode dépasse les performances utilisant des expressions fréquentes [29]. Plus récemment, dans le travail de Maylawati & Saptawati (2017), les auteurs parviennent à extraire les caractéristiques de l'argot indonésien. Ils montrent qu'une méthode SPM, couplée à un prétraitement et à une sélection de caractéristiques, est capable de préserver le sens avec une justesse de 77.33%.

Comme l'un des objectifs du chapitre est d'évaluer la possibilité d'extraire des ensembles d'objets ordonnés pertinents qui peuvent être utilisés pour la HAR, une méthode de SPM appliquée aux pages web semble pouvoir fournir des résultats intéressants et est étudiée dans les sections qui suivent.

---

29. *Expression fréquente* est la traduction que nous avons fait du concept nommé *statistical phrase* dans l'article original de (Mitra *et al.*, 1997)

5.2 DÉFINITIONS ET JEU DE DONNÉES

Afin de présenter correctement notre recherche, il convient d'abord de définir certains concepts. Ces concepts-clés seront utilisés dans notre méthode d'extraction automatique d'informations sur les activités à partir de pages web. De plus, dans cette section, nous décrivons la collection de pages web qui sont utilisées pour constituer un jeu de données. Ce jeu de données est utilisé pour l'extraction automatique, mais aussi pour valider notre approche à travers un ensemble de tests.

5.2.1 SÉQUENCE D'UTILISATION D'OBJETS

Nous définissons une séquence d'utilisation d'objets comme une séquence ordonnée de chaînes de caractères contenant des informations sur l'utilisation d'un ou plusieurs objets impliqués dans la réalisation d'une activité. Ces séquences peuvent être de deux types distincts. Il peut s'agir de séquences d'objets simples (par exemple, [*water, sugar, tea*]) ou de séquences de couples verbe/objet (par exemple, [*fill*||*bottle, fill*||*water*]). Dans le second cas, afin d'obtenir une chaîne de caractères, nous séparons le verbe et l'objet par la chaîne ||. Dans tous les cas, nous désignons les deux types comme des séquences d'utilisation d'objets.

Chaque élément de ces séquences nous informe sur un objet et/ou l'action associée à cet objet. Afin de rendre l'information latente contenue dans ces séquences aussi faciles à extraire que possible, il est important de considérer comment comparer plusieurs séquences. Notre approche doit être capable de détecter si deux mots sont synonymes afin de les considérer comme une même entité. Par exemple, les séquences [*put*||*water, take*||*pot*] et [*place*||*water, take*||*pot*] devraient être considérées comme la même séquence puisque *put* et *place* ont une signification similaire dans ce contexte. Afin de prendre en compte l'existence de synonymes, un traitement spécifique dédié à l'existence de synonymes est systématiquement appliqué

à chaque séquence. Pour cela, une base de données lexicale bien connue de l'anglais a été sélectionnée. WordNet (Princeton University, 2010a) est une base de données permettant d'obtenir de nombreuses informations sur les mots de la langue anglaise comme leur sens, leur classe grammaticale, leurs synonymes, etc. À l'aide de la librairie wordpos [30] qui utilise WordNet, nous recherchons tous les synonymes des noms et/ou verbes dans nos séquences [31]. De cette manière, pour un nom/verbe donné, nous sommes en mesure de regrouper tous les synonymes possibles sous un même et unique mot. En reprenant notre exemple précédent, le verbe "place" est remplacé par "put" rendant les séquences parfaitement identiques aux yeux de notre méthode [32].

5.2.2 MOTIF D'UTILISATION D'OBJETS

En considérant la définition des séquences d'utilisation d'objets et que notre méthode utilise du SPM, notre tâche est de trouver des motifs appropriés qui apparaissent fréquemment dans un corpus de pages web. La définition stricte d'un motif dans le SPM est une sous-séquence ayant une fréquence associée supérieure à un seuil minimal appelé support minimal. Dans le cas des motifs que l'on souhaite extraire dans cette recherche, la notion de motif d'utilisation d'objet est introduite. Un motif d'utilisation d'un objet est un élément qui est une sous-séquence d'utilisation d'un objet dont la fréquence d'occurrence dans un ensemble de séquences d'utilisation d'objets est supérieure ou égale à la valeur à un support minimal. Par exemple, si l'on prend l'ensemble de séquences fourni dans le Tableau 5.1 avec un support minimal de 0.75 (75%), les motifs d'utilisation d'objets sont alors [*water*], [*spoon*], [*water*, *spoon*] avec une fréquence d'occurrence de, respectivement, 0.83, 0.5 et 0.5.

---

30. https://github.com/moos/wordpos

31. Il est possible de consulter directement sur son navigateur la base de données WordNet sur le site mis à disposition par Princeton University (2010c)

32. On peut constater sur le site de Princeton University (2010b) que "put" et "place" sont bien synonymes

Tableau 5.1 : Exemple d'un ensemble de séquences d'utilisation d'objets. © Charles Cousyn, 2022.

[pot, water]
[water, tea, spoon]
[pot, tea]
[water, spoon, glass]
[water, fork, carrot]
[water, sugar, spoon]

5.2.3 PAGES WEB : NOTRE JEU DE DONNÉES

La première étape du processus d'extraction de motifs consiste à collecter un ensemble représentatif de pages web descriptives pour chaque activité pour laquelle des informations sont nécessaires. Il existe deux méthodes distinctes utilisables pour obtenir les pages web pour notre recherche.

La première méthode consiste à récupérer des pages web à l'aide d'un moteur de recherche tel que DuckDuckGo (DuckDuckGo, 2021) en faisant des requêtes telles que "how to [nom de l'activité]". Pour cela, notre équipe a conçu un premier outil [33] afin de récupérer l'ensemble des résultats de recherche de DuckDuckGo et un second outil [34] pour récupérer les pages web liées à chaque résultat de recherche. DuckDuckGo a été choisi en raison de son utilisation massive par les internautes (DuckDuckGo, 2021). La deuxième méthode consiste à récupérer manuellement des pages web à partir de sites spécifiques contenant assez d'informations sur les activités considérées et présentant une structure facilitant l'extraction d'informations. L'équipe de recherche a identifié le site Wikihow [35] à cette fin. Wikihow est un

33. https://github.com/CharlesCousyn/search_activities
34. https://github.com/CharlesCousyn/webpages_retrieval
35. https://www.wikihow.com/

site web très populaire qui fournit des guides et des étapes pour un très grand nombre d'activités (plus de 210000 articles). Pour chaque activité que nous considérons, nous récupérons le maximum de pages du site Wikihow décrivant l'activité en question. Notre hypothèse est que Wikihow a le potentiel de contenir tous les objets et matériaux liés à la réalisation d'activités de la vie quotidienne ainsi que l'ordre dans lequel ils sont utilisés grâce à ses descriptions structurées sous forme de texte . La liste des pages web utilisées pour la suite du processus est disponible sur notre dépôt [36].

Dans la Section 5.4, les motifs trouvés à l'aide de ces deux méthodes sont comparés et les résultats montrent que la sélection manuelle est préférable, mais pas de beaucoup par rapport à l'utilisation d'un moteur de recherche.

5.3 MÉTHODOLOGIE

L'objectif de cette section est de décrire la méthode d'extraction utilisée pour obtenir des motifs d'utilisation d'objets de manière non supervisée en utilisant des pages web comme principale source de données. Notre méthode est divisée en trois étapes : le processus HTML2Sequences ; responsable de l'utilisation des pages web pour obtenir des séquences d'utilisation d'objets, l'extraction de motifs ; responsable de l'extraction des motifs d'utilisation des objets et le post-traitement. Cette méthode possède plusieurs paramètres (mis en évidence en italique dans le texte) qui ont un impact sur ses performances. Ces paramètres sont systématiquement évalués afin d'optimiser la solution.

36. https://github.com/CharlesCousyn/webpages_retrieval/tree/master/data

5.3.1 LE PROCESSUS HTML2SEQUENCES

Le premier élément de notre méthode est un processus nommé HTML2Sequences dont l'objectif est de transformer les pages web en un ensemble de séquences d'utilisation d'objets. Le processus est décrit étape par étape et la Figure 5.1 est une représentation complète de son fonctionnement détaillé.

ÉTAPE 1 : DE PAGES WEB À PLAN TEXTUELS

Les pages web étant au format HTML (W3.org, 2008), il est nécessaire d'utiliser un outil adapté au traitement de ce type de données. Dans le présent travail, nous utilisons le module JSDOM (Faulkner *et al.*, 2020). Il s'agit d'une bibliothèque pour NodeJS permettant d'analyser et d'extraire des informations d'un fichier HTML de la même manière que le DOM (W3C.org, 2021; MDN contributors, 2021) utilisé dans le navigateur. Une fois la page chargée, la suite du processus consiste à extraire un ou plusieurs blocs contenus dans la page. Cette extraction de blocs peut se faire de deux manières différentes en fonction d'un paramètre booléen nommé *genericOrSpecificParsing*. Si le paramètre a pour valeur vraie, un traitement générique est effectué alors que dans le cas contraire, il s'agit d'un traitement spécifique.

Le traitement spécifique consiste à extraire des blocs spécifiques dans notre page. Ce type de traitement ne s'applique qu'aux pages sélectionnées manuellement sur le site Wikihow. Le choix des blocs sélectionnés est défini manuellement sous un œil humain en essayant de trouver l'ensemble des blocs contenant les étapes de réalisation de l'activité décrite dans la page web. Plus précisément, en utilisant JSDOM (Faulkner *et al.*, 2020), les sélecteurs CSS (W3C.org, 2018) et JavaScript (Mozilla Developer Network, 2020), nous extrayons les titres de chaque étape (voir Figure 5.2) de chaque méthode présentée sur la page pour construire un ou

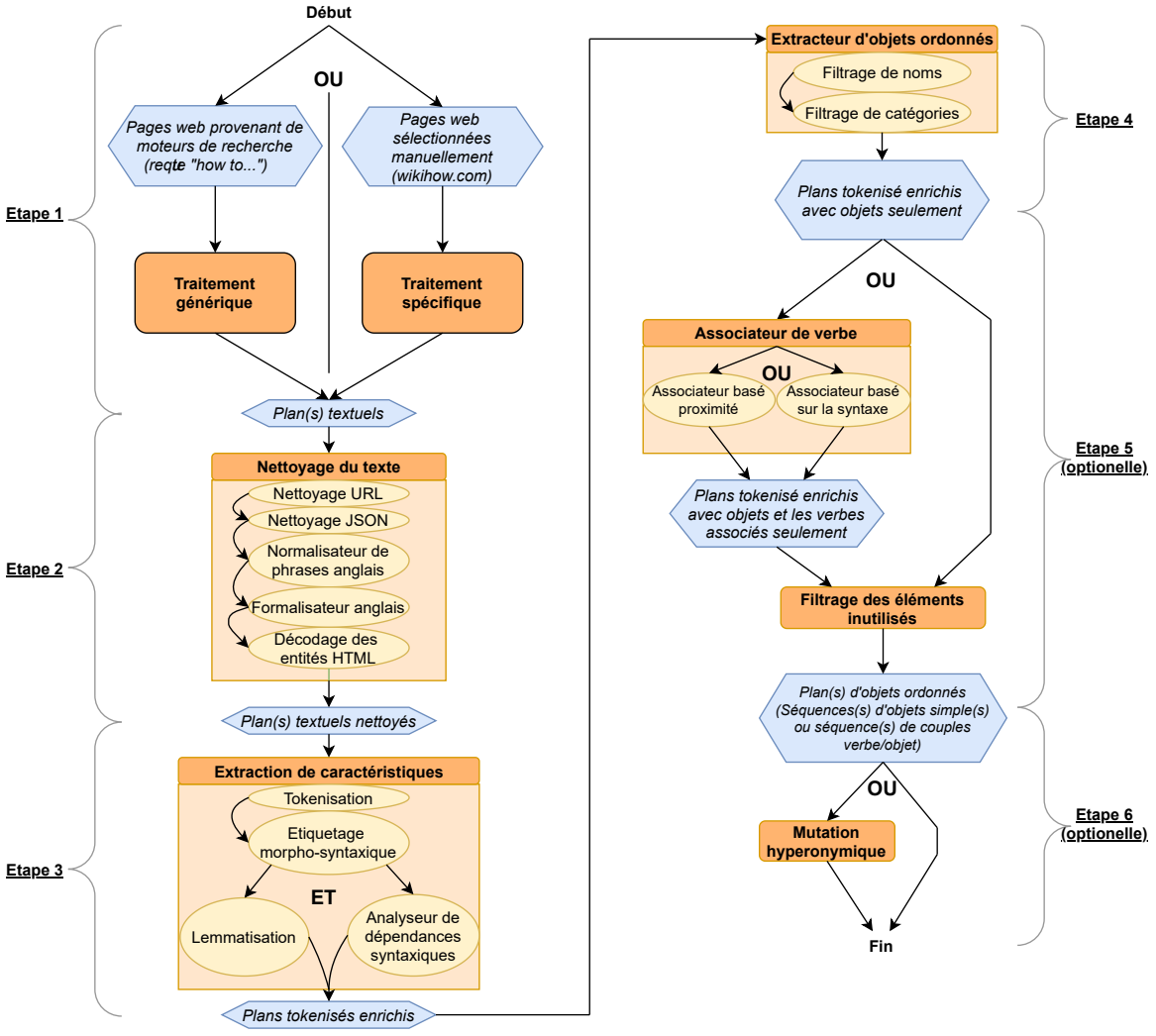

Figure 5.1 : Description du processus HTML2Sequences. Les éléments orange représentent les modules réalisant les différentes étapes du processus. Les éléments en jaune représentent les sous-modules. Les éléments en bleu représentent les données qui circulent dans les modules. Les flèches représentent le flux de données entre les modules et les sous-modules. Le mot "OU" reflète la présence d'un paramètre booléen dans le processus, ce qui implique qu'un seul des deux chemins est emprunté en fonction de la valeur du paramètre. D'autre part, le mot "ET" représente le fait que tous les chemins sont empruntés sans l'utilisation d'un paramètre. © Charles Cousyn, 2022.

plusieurs textes, chaque texte symbolisant un plan d'exécution où chaque bloc est séparé par

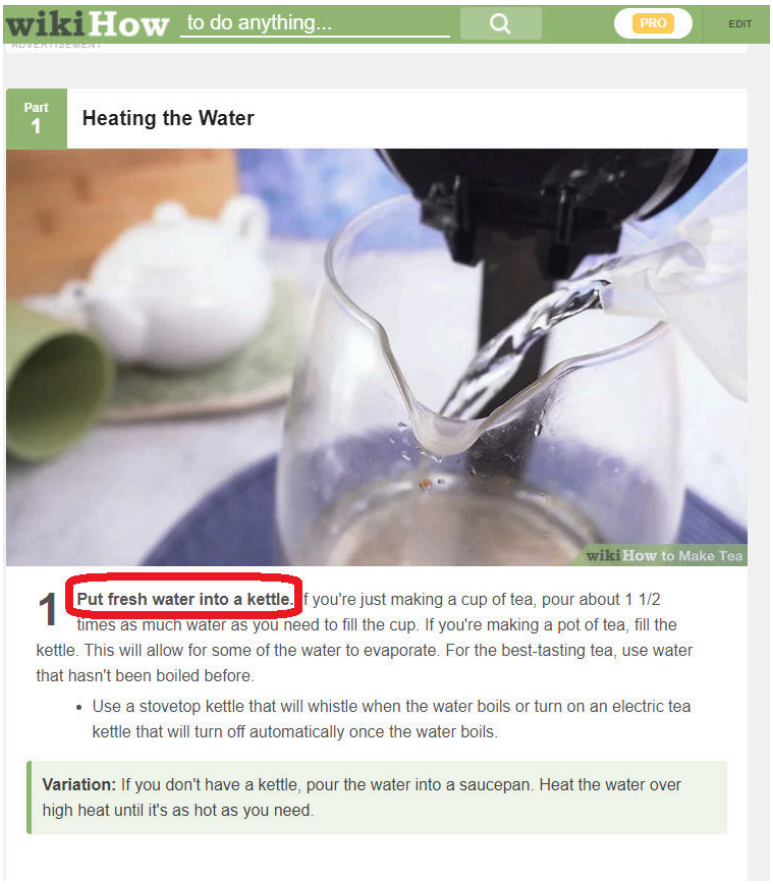

Figure 5.2 : Capture d'écran d'une page décrivant l'activité *faire du thé*. Un titre utilisé pour le traitement spécifique est entouré en rouge (https://www.wikihow.com/Make-Tea, le 27/04/2021). Contenu Creative Commons.

un espace. Il est à noter que pour un traitement spécifique, il est possible d'obtenir plusieurs plans d'exécution avec la même page.

D'autre part, le traitement générique consiste à utiliser l'intégralité du code HTML de la page web. Ainsi, avec ce traitement, toute page web peut être traitée, quelle que soit la structure du site web dont elle fait partie. À noter qu'utiliser l'intégralité du contenu d'une page web, comme le fait le traitement générique, implique qu'une partie de l'information obtenue est non pertinente pour nos séquences d'utilisation d'objets ; ce qui n'est pas le cas

avec le traitement spécifique où la sélection du contenu a été faite manuellement. Enfin, après avoir réussi à extraire les blocs de HTML qui nous intéressaient, nous utilisons un outil [37] pour convertir le HTML en texte en supprimant toutes les balises HTML mais en conservant leur contenu textuel. On obtient alors un ou plusieurs textes représentant un ou plusieurs plans d'exécution ; nous nommons ce type de données plans textuels. Notons qu'avec le traitement générique, un seul plan textuel est obtenu par page web.

ÉTAPE 2 : DES PLANS TEXTUELS AUX PLANS TEXTUELS NETTOYÉS

Les plans textuels obtenus précédemment ne contiennent peut-être plus de balises HTML, mais à ce stade précis, ils contiennent encore un certain nombre d'éléments inutiles, inutilisables ou non normalisés qui doivent être traités. Le texte au format JSON et les URLs font partie de ces éléments. Ils sont supprimés à l'aide d'expressions régulières. Ensuite, un normalisateur de phrases anglais [38] est utilisé pour résoudre les contractions (par exemple, "I'm" devient "I am"), pour remplacer les caractères confondants (par exemple, garder un seul type de guillemets), et pour mettre tout le texte en minuscules (par exemple, "HOW IS ALEX" devient "how is alex"). Ensuite, nous utilisons un formalisateur anglais [39] pour obtenir un vocabulaire plus formel dans notre texte (par exemple, "gr8" devient "great", "plz" devient "please", "imo" devient "in my opinion", etc.). Enfin, nous utilisons un décodeur d'entités HTML [40] pour transformer les caractères spécifiques du HTML en caractères utilisables (par exemple, &amp ; devient &, &quot ; devient ", &#x00025 ; devient %, etc.). On obtient ainsi les plans textuels nettoyés.

---

37. https://github.com/html-to-text/node-html-to-text
38. https://github.com/FinNLP/en-norm
39. urlhttps ://github.com/FinNLP/fin-slang
40. https://github.com/FinNLP/fin-html-entities

ÉTAPE 3 : DES PLANS TEXTUELS NETTOYÉS AUX PLANS TOKÉNISÉS ENRICHIS

Les plans textuels nettoyés obtenus sont ensuite fournis à un module nommé le "String Features Enricher" qui fonctionne comme suit. Tout d'abord, nous procédons à la tokénisation du texte en utilisant l'outil nommé lexed [41] pour obtenir une liste de tokens pour chaque phrase. Ensuite, nous fournissons cette liste à un outil d'étiquetage morpho-syntaxique (Part-of-Speech Tagging ou POS Tagging) (Charniak, 1997) [42] et à un analyseur syntaxique de dépendance [43] (Vadapalli, 2021) afin d'obtenir la classe grammaticale de chaque token ainsi qu'un arbre représentant les relations grammaticales entre les tokens. Cette nouvelle représentation de nos textes nous fournit des informations précieuses qui sont exploitées dans les étapes suivantes. Enfin, le dernier processus consiste à ramener chaque nom et verbe de chaque phrase à sa forme la plus pure en faisant passer les verbes à l'infinitif et les noms au singulier. Ce processus est une lemmatisation (Müller *et al.*, 2015). L'outil appelé en-inflectors [44] est utilisé pour ce faire. Le résultat de cette étape est alors une nouvelle représentation des plans textuels nettoyés sous une forme tokenisée où chaque token est enrichi par l'ensemble des caractéristiques suivantes : la chaîne initiale, la chaîne lemmatisée, son étiquette POS, son noeud correspondant dans l'arbre de dépendance syntaxique. Nous appelons cette représentation un plan tokénisé enrichi.

41. https://github.com/FinNLP/lexed
42. https://github.com/FinNLP/en-pos
43. https://github.com/FinNLP/en-parse
44. https://github.com/FinNLP/en-inflectors

ÉTAPE 4 : TROUVER LES OBJETS DANS LES PLANS TOKÉNISÉS ENRICHIS

Les plans tokénisés enrichis obtenus contiennent tout ou partie des objets impliqués dans l'exécution des activités de la vie quotidienne. L'objectif est alors de les localiser de manière efficace et effective. Tout d'abord, grâce au processus d'étiquetage morpho-syntaxique, nous sommes en mesure d'identifier tous les noms de chaque phrase de notre texte. Par exemple, dans la phrase "Fill a large pot about 2/3 full of water", les noms "water" et "pot" ont le potentiel d'informer sur les objets impliqués dans la réalisation de l'activité "cook pasta". Cependant, identifier chaque nom comme un objet serait insuffisant puisque les objets ne représentent qu'un sous-ensemble limité de noms. Pour y remédier, WordNet Princeton University (2010a) est à nouveau utilisé avec l'outil wordpos [45] pour obtenir la catégorie syntaxique de chaque nom. Par exemple, le nom "spoon" appartient à la catégorie *noun.artifact* qui désigne les noms d'objets fabriqués par l'homme [46]. Les catégories suivantes sont retenues : *noun.artifact, noun.object, noun.substance, noun.plant, noun.animal, noun.food, noun.body*. Celles-ci semblent pouvoir englober la majorité, voire la totalité, des objets liés à la réalisation des activités de la vie quotidienne. Par ailleurs, il convient de noter qu'un mot peut avoir plusieurs définitions, il peut donc avoir plusieurs catégories syntaxiques. Dans ce cas, un nom est considéré comme un objet si au moins une de ses définitions correspond à une des catégories syntaxiques de notre liste. Il en résulte un plan tokenisé enrichi ne comportant que des objets.

45. https://github.com/moos/wordpos

46. Toutes les catégories existantes ainsi que leur description sont disponibles sur le site de (Princeton University, 2022)

ÉTAPE 5 : ASSOCIATEUR DE VERBE

À partir des tokens ordonnés à l'aide des objets liés aux activités, une représentation plus riche pourrait être obtenue en utilisant le verbe associé à chaque objet. Cela permettrait de montrer non seulement les objets utilisés, mais aussi comment ils sont utilisés. En conséquence, un paramètre a été ajouté à notre méthode. Celui-ci est un booléen nommé *verbAssociatorUsed* qui, lorsqu'il a pour valeur vraie, déclenche l'utilisation du module nommé associateur de verbe. Ce module peut effectuer l'association avec un verbe de deux manières distinctes en fonction de la valeur d'un paramètre booléen appelé *verbAssociatorProximityBasedOrSyntacticBased*. Lorsqu'il est défini à vrai, on associera à l'objet le verbe dont la distance en termes de nombre de tokens est minimale dans la phrase. Quand ce paramètre a pour valeur faux, la distance utilisée est le nombre total d'arêtes du plus court chemin dans l'arbre de dépendance syntaxique entre le verbe et l'objet. La Figure 5.3 donne un exemple d'arbre de dépendance syntaxique pour la phrase "Add some salt to the recipe.". Par exemple, plaçons-nous dans l'exemple de cette phrase et de la Figure 5.3 et que notre objectif est d'estimer la distance entre l'objet *salt* et le verbe *Add*. Dans ce cas, il y a une distance de 2 en tokens (*verbAssociatorProximityBasedOrSyntacticBased* à vrai) et une distance de 1 en utilisant l'arbre de dépendance syntaxique (*verbAssociatorProximityBasedOrSyntacticBased* à faux). N'y ayant qu'un unique verbe dans la phrase, il n'y a qu'un choix possible. Dit autrement, utiliser une distance ou l'autre ne fait pas de différence dans cet exemple cependant il existe des cas où le verbe le plus proche en termes de tokens est différent de celui en termes de syntaxe. En résumé, nous avons ajouté ce paramètre afin de connaître l'impact de la mesure plus simpliste (mais plus facile à obtenir) de proximité en termes de tokens par rapport à l'exploitation de la syntaxe même de la phrase.

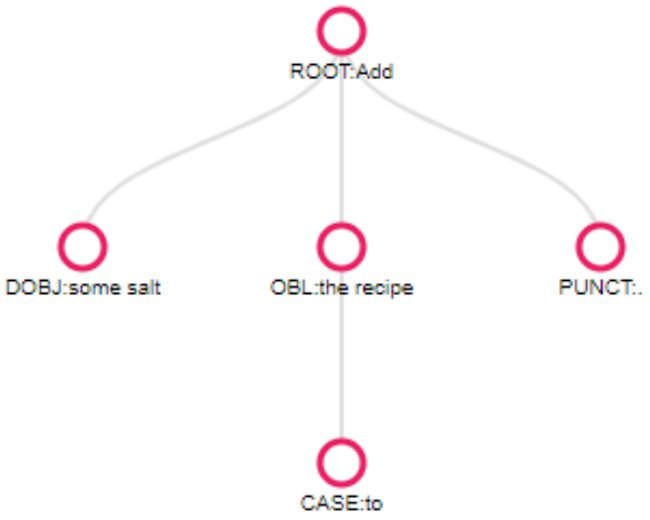

Figure 5.3 : Exemple d'un arbre de dépendance syntaxique pour la phrase "Add some salt to the recipe.". ROOT désigne la racine de l'arbre, DOBJ désigne un objet direct, OBL désigne un oblique nominal, PUNCT désigne la ponctuation et CASE un élément de marquage de la casse (les détails complets des relations peuvent être trouvés sur le site https://universaldependencies.org/u/dep/). © Charles Cousyn, 2022.

ÉTAPE 6 : LA MUTATION HYPERONYMIQUE

Enfin, afin d'améliorer potentiellement les séquences obtenues, un dernier processus a été ajouté et est optionnel en fonction de la valeur du paramètre booléen *hypernymMutation* de notre méthode. Il s'agit de ce que nous appelons la mutation hyperonymique dont le concept est le suivant. Si pour un ensemble de séquences d'utilisation d'objets, il existe une relation d'hyperonymie directe entre deux objets (un hyponyme et un hyperonyme), alors l'objet appelé hyponyme est remplacé par son hyperonyme. L'hyperonymie désigne une relation sémantique hiérarchique d'une unité lexicale à une autre dans laquelle l'extension du premier terme, plus général, englobe l'extension du second terme, plus spécifique (Theissen, 1997). L'hyperonymie peut être directe si l'unité lexicale est le parent direct ou indirecte si l'unité lexicale est un ancêtre de l'autre. Par exemple, *pasta* peut être considéré comme un hyperonyme direct de

noodle. Imaginons un exemple concret d'application de la mutation hyperonymique avec les deux séquences d'utilisation d'objets suivantes provenant de deux pages web : [boil||water, cook||pasta] et [boil||water, cook||noodle]. En observant ces séquences, on constate qu'elles sont très proches à l'exception du terme *pasta* remplacé par *noodle*. Le module repose sur l'hypothèse que les *noodle* sont un type particulier de *pasta*. Nous nous attendons à ce qu'il soit souhaitable de le considérer comme tel dans notre méthode, car cela devrait permettre de ne pas pénaliser les motifs contenant des objets pour lesquels il existe un grand nombre de termes plus spécifiques. Cette hypothèse sera testée dans la Section 5.4.

5.3.2 EXTRACTION DE MOTIFS

Grâce aux étapes précédentes, nous disposons de séquences d'utilisation des objets qui ont le potentiel de contenir tout ou partie des objets liés à la réalisation des activités de la vie quotidienne. La tâche restante est d'extraire les motifs intéressants contenus dans ces séquences, en utilisant dans notre cas, un algorithme de SPM pour extraire les motifs les plus fréquents. L'extraction de tous les motifs fréquents est souvent irréaliste comme le montrent les travaux de Pei *et al.* (2001) : ils sont beaucoup trop nombreux, trop longs à obtenir, ce qui rendrait leur évaluation des motifs beaucoup plus difficile.

Pour contourner le problème, nous avons fait le choix d'extraire des motifs ayant des propriétés particulières. Nous extrayons soit des motifs fermés soit des motifs maximaux, car il a été montré (Fournier-Viger *et al.*, 2014; Gomariz *et al.*, 2013) que l'extraction de ces types de motifs permet de conserver une grande partie de l'information de l'ensemble des motifs séquentiels fréquents et prend un temps raisonnable. D'une part, un motif maximal est un motif p_a pour lequel il n'existe pas de motif séquentiel p_b tel que p_b est un super-motif de p_a (par exemple, [a, b] est un super-motif de [b]). D'autre part, un motif fermé est un motif p_a pour lequel il n'existe pas de motif séquentiel p_b tel que p_b est un super-motif de p_a et que

Tableau 5.2 : Exemple de motifs fréquents. © Charles Cousyn, 2022.

Motif fréquent	Occurrence	Fermé	Maximal
[water]	0.83	Oui	Non
[spoon]	0.5	Non	Non
[water, spoon]	0.5	Oui	Oui

leur support est identique. Par exemple, si nous prenons l'ensemble de séquences d'utilisation d'objets fourni dans le Tableau 5.1 avec un support minimal de 0.5, nous sommes capables d'identifier 3 motifs fréquents dont certains sont fermés et d'autres maximaux. Le Tableau 5.2 montre ces motifs, leur occurrence et s'ils sont fermés et/ou maximaux. On remarquera, dans cet exemple et cela est toujours vrai, que les motifs fermés forment un sous-ensemble de motifs fréquents et que les motifs maximaux sont un sous-ensemble de motifs fermés.

Le choix du type spécifique de motif extrait est contrôlé par la valeur du paramètre booléen *closedOrMaximal* dans notre méthode. En ce qui concerne l'algorithme utilisé pour l'extraction des motifs, une variante de l'algorithme VMSP (Vertical mining of Maximal Sequential Patterns) par Fournier-Viger *et al.* (2014) a été implémentée avec les modifications suivantes.

Tout d'abord, l'algorithme a été modifié pour prendre en entrée une base de données contenant des séquences de 1-uplets, car les jeux de données qui résultent du processus présenté dans la Section 5.3.1 sont des séquences de 1-uplets. Deuxièmement, nous avons modifié VMSP pour qu'il soit capable d'extraire des motifs fermés en plus des motifs maximaux en utilisant le paramètre *closedOrMaximal*. Cette modification s'est avérée utile et ne semble pas affecter de manière significative les performances de l'algorithme VMSP original. Le code de

l'algorithme, que nous nommons "Vertical Mining of Maximal and Closed Sequential Patterns in sequences of 1-uplets" (VMCSP1), est disponible sur notre dépôt [47].

Il convient d'ajouter un certain nombre de précisions sur le fonctionnement de l'extraction des motifs. Premièrement, lorsque le support minimal est trop faible (inférieur à 10%), le temps de calcul devient exponentiellement long. Ceci est attendu puisque la baisse du support minimal provoque une explosion combinatoire sur le nombre de motifs qui peuvent être considérés par l'algorithme. L'article sur l'algorithme VMSP original (Fournier-Viger *et al.*, 2014) le montre très clairement ; le choix d'un support minimum faible peut rendre l'algorithme incapable de se terminer dans un temps raisonnable. Au début, nous voulions optimiser VMCSP1 sur les valeurs de support minimal, mais le temps nécessaire était excessif. Par conséquent, au lieu de fournir un support minimal à notre méthode, celle-ci s'appuie sur un nombre minimal de motifs à extraire.

Avec ces informations, l'algorithme VMCSP1 a été exécuté jusqu'à vingt fois jusqu'à obtenir le nombre minimum de motifs demandés. Il est parti d'une valeur de support minimale de 1.0 et a été décrémentée par pas de 0.05 jusqu'à 0.0 ou jusqu'à ce que le nombre de motifs demandé soit atteint. Cela permet d'obtenir les motifs dans un délai raisonnable. Un des inconvénients est que, en demandant un nombre minimal de X motifs, on obtient, pour certaines activités, un nombre exagéré ou insuffisant de motifs. En effet, a priori, rien ne nous dit que le nombre de motifs réellement pertinents pour une activité donnée se situe autour d'une valeur fixe X. Il est possible d'imaginer qu'une activité possède 3 motifs pertinents alors qu'une autre en possède plus d'une centaine. Néanmoins, après plusieurs tentatives et la contrainte informatique étant extrêmement limitante, le nombre minimal de motifs a été fixé à 10.

47. https://github.com/CharlesCousyn/webpages_objects_spm_extractor/blob/main/sequentialPatternMining/VMSP.js

5.3.3 POST-TRAITEMENT

Dans ce travail, nous considérons que l'occurrence d'un motif est un indicateur de la pertinence dans la réalisation d'une activité, mais aussi que ce n'est pas le seul élément à prendre en compte. En effet, un motif peut être extrêmement fréquent dans une activité, mais si ce motif est présent pour un grand nombre d'activités, alors ce motif peut ne pas permettre d'identifier l'activité de manière spécifique. Un exemple concret d'un motif fréquent mais non pertinent est le motif [add||salt]. Comme ce motif est présent pour de nombreuses activités liées à la cuisine, il ne permet pas de distinguer deux activités comme "cook pasta" et "make pancakes". Nous pouvons voir le parallèle entre ce problème et celui soulevé par l'utilisation de modèles de sacs de mots (Manning, 2008) en fouille de textes. Un simple sac de mots contient tous les mots d'un texte et associe à chaque mot sa fréquence d'apparition dans le texte, ce qui peut refléter l'importance du mot en question. Cependant, il existe des mots extrêmement fréquents comme "the", qui ne sont pourtant que peu pertinents dans la langue anglaise en général. Pour tenter de résoudre ce problème, la mesure statistique "Term Frequency - Inverse Document Frequency" (TF-IDF) (Robertson, 2004) a été utilisée. TF-IDF pondère la fréquence des mots par la fréquence inverse dans plusieurs documents ; elle met en œuvre un facteur qui diminue le poids des termes qui apparaissent très fréquemment dans un ensemble de documents et augmente le poids des termes qui apparaissent rarement. En s'inspirant de cette technique, il est tout à fait possible de créer une mesure similaire pour pondérer l'importance des motifs qui sont trop fréquents pour plusieurs activités comme [add||salt]. Nous introduisons donc la mesure "Pattern Frequency - Inverse Activity Frequency" (PF-IAF). Sa formulation mathématique est donnée par l'équation (5.1).

$$\text{PF-IAF}(p) = \text{PF}(p) \cdot \log\left(\frac{N}{N_p}\right) \tag{5.1}$$

où N désigne le nombre total d'activités considérées, N_p désigne le nombre d'activités ayant le motif p et PF(p) dénote la fréquence du motif p.

Après l'extraction des motifs séquentiels par l'algorithme VMCSP1, un post-traitement est effectué. Il s'agit de l'association de chaque motif extrait à une valeur PF-IAF. Celle-ci devrait mieux refléter la qualité d'un motif que sa fréquence. Nous évaluons la qualité de la mesure PF-IAF dans la Section 5.4.4.

5.4 ÉVALUATION DES MOTIFS SÉQUENTIELS EXTRAITS

Après avoir décrit la méthode permettant d'obtenir les motifs d'utilisation d'objets, cette section se concentre sur l'évaluation de notre approche afin de mieux estimer son applicabilité dans des situations réelles. En particulier, nous cherchons à savoir si les paramètres de notre méthode, à savoir : *genericOrSpecificParsing*, *verbAssociator*, *verbAssociatorProximityBasedOrSyntacticBased*, *hypernymMutation*, *closedOrMaximalPattern* ont un impact significatif sur les performances de notre approche. Ensuite, dans cette section, nous présentons, respectivement, les objectifs précis de cette évaluation, la méthodologie, le jeu de données et enfin les résultats obtenus.

5.4.1 OBJECTIF DE L'ÉVALUATION

L'objectif de l'évaluation est de répondre à 4 questions distinctes qui caractérisent la qualité de notre approche. Soit :

— Quel est l'impact des paramètres de notre méthode sur la qualité des motifs ?

— Quelle(s) configuration(s) des paramètres de notre méthode donne(nt) les motifs de meilleure qualité ?

— Quel est l'impact des paramètres de notre méthode sur le temps de calcul ?

— Notre mesure PF-IAF est-elle une bonne mesure de la pertinence de nos motifs ?

Les réponses à ces questions sont explicitement fournies à la Section 5.4.4.

5.4.2 MÉTHODOLOGIE D'ÉVALUATION

Pour répondre de manière rationnelle à ces quatre questions, nous testerons notre approche avec diverses activités. De la même manière que décrite dans la Section 4.2.1 du Chapitre 4, nous avons défini un ensemble totalisant 18 activités distinctes utilisées dans nos travaux précédents (Cousyn *et al.*, 2021). Rappelons que 8 activités proviennent de celles utilisées dans les travaux de Riboni & Murtas (2017, 2019) qui sont très proches de notre problématique (elles-mêmes inspirées du jeu de données CASAS *Interweaved ADL Activities* (CASAS, 2019)), 6 activités ont été inspirées du jeu de données CASAS *ADL Activities* (CASAS, 2019), et 4 activités ont été créées de toutes pièces. Ces activités sont les suivantes : *fill medicine cabinet*, *watch tv*, *water plants*, *answer the phone*, *prepare a birthday card*, *prepare soup*, *clean*, *vacuum*, *choosing outfit*, *make tea*, *make coffee*, *cook pasta*, *do homework*, *play video games*, *bake parmesan turkey meatballs*, *make hut stuffed crust pizza*, *make mashed potato casserole*, *make low carb pancakes*.

Un autre aspect à considérer est que nous devons être en mesure de tester différentes combinaisons de paramètres. Dans notre méthode, décrite dans la Section 5.3, cinq paramètres booléens différents ont été introduits à savoir *genericOrSpecificParsing*, *verbAssociator*, *verbAssociatorProximityBasedOrSyntacticBased*, *hypernymMutation* et *closedOrMaximalPattern*. A priori, il y a donc $2^5 = 32$ combinaisons possibles. Cependant, les paramètres ne sont pas tous indépendants. Le paramètre *verbAssociatorProximityBasedOrSyntacticBased* n'a un effet que si le paramètre *verbAssociator* a la valeur vraie (voir la Figure 5.1). En tenant compte de cela, le nombre de combinaisons possibles de paramètres est réduit à 24.

Enfin, le point le plus important de notre méthodologie est le type d'analyse réalisée. Dans le but d'avoir des résultats encore plus fiables, nous avons décidé d'utiliser une méthode d'analyse différente de celle utilisée dans le Chapitre 4. La méthode du Chapitre 4 consistait majoritairement à l'interprétation de graphiques en fonction des paramètres à l'aide de moyennes et d'écarts-types. Afin d'améliorer notre analyse, nous avons souhaité nous baser sur des tests statistiques efficaces et utilisant les dernières avancées en recherche, notamment celles des travaux de van Doorn *et al.* (2020).

Nous avons donc choisi de se baser sur l'analyse statistique bayésienne en utilisant le logiciel JASP [48] (JASP Team, 2020). Chaque analyse fournit deux éléments distincts. Le premier est un facteur de Bayes BF_{10} (ou BF_{+0}) symbolisant le quotient de la plausibilité des données sachant l'existence et l'absence d'un effet. L'interprétation d'un facteur de Bayes peut se faire à l'aide du Tableau 5.3. Il est intéressant de noter que le tableau peut être utilisé lorsque le facteur de Bayes est inférieur à 10^0. Pour ce faire, nous pouvons appliquer l'ensemble de la force de la preuve en faveur de l'hypothèse alternative. Par exemple, un facteur de Bayes de $\frac{1}{2\times10^1} = 0.05$ indique une force de preuve de niveau *fort* en faveur de l'hypothèse alternative. Le deuxième élément est une taille d'effet avec un intervalle de crédibilité. La taille de l'effet est généralement interprétée comme le nombre d'écarts-types qui séparent les groupes considérés. Son interprétation peut se faire à l'aide du Tableau 5.4.

48. Chaque analyse réalisée avec JASP peut être consultée en ouvrant les fichiers *.jasp disponibles dans notre dépôt à l'adresse https://github.com/CharlesCousyn/webpages_objects_spm_extractor/tree/main/experimentationResults/JASPFiles

Tableau 5.3 : Facteurs de Bayes et force de preuve associée par Goodman (1999). © Charles Cousyn, 2022.

BF	Force de preuve
$< 10^0$	Négatif (supporte l'hypothèse alternative)
10^0 to $10^{\frac{1}{2}}$	À peine digne d'être mentionnée
$10^{\frac{1}{2}}$ to 10^1	Substantiel
10^1 to $10^{\frac{3}{2}}$	Fort
$10^{\frac{3}{2}}$ to 10^2	Très fort
$> 10^2$	Décisif

Tableau 5.4 : Taille d'effet et description associée par Cohen (2013) et Sawilowsky (2009). © Charles Cousyn, 2022.

Description	Taille d'effet
Très petit	0.01
Petit	0.20
Moyen	0.50
Grand	0.80
Très grand	1.20
Énorme	2.0

JEU DE DONNÉES D'ÉVALUATION

Pour effectuer l'évaluation et notre analyse statistique, nous devons leur fournir tous les motifs générés par notre approche dans chaque combinaison possible de paramètres. Ces motifs pour chaque combinaison et pour chaque activité sont disponibles dans notre dépôt [49].

5.4.3 QUALITÉ DES MOTIFS

Le jeu de données créé associe pour chaque couple motif/activité un score qui dénote sa qualité. Cette qualité doit ensuite être évaluée. La méthode idéale pour le faire serait

49. https://github.com/CharlesCousyn/webpages_objects_spm_extractor/tree/main/experimentationResults/allConfAnnotatedExperimentalResults

de les utiliser dans un contexte réel où l'on mesure le taux de reconnaissance d'activités réalisées par des individus volontaires dans un environnement intelligent. Cette méthode pose plusieurs défis. En plus de nécessiter la participation d'individus, elle requiert une mise en correspondance entre les motifs et les capteurs de l'environnement particulier qui est utilisé.

La seconde méthode consiste à effectuer une annotation manuelle des motifs sous un oeil humain. Cette méthode a l'avantage d'être plus simple à mettre en œuvre et de quantifier clairement la qualité de chaque motif, mais elle suppose que les individus *annotateurs* sont capables d'attester eux-mêmes de la qualité d'un motif. C'est la solution que nous avons choisie, car faire correspondre l'environnement aux motifs est une tâche très dépendante de l'environnement et donc beaucoup plus difficile à généraliser qu'une annotation manuelle.

PROCESSUS D'ANNOTATION

Afin d'effectuer une annotation correcte par un ou plusieurs individus, il existe plusieurs caractéristiques permettant d'attester de la qualité d'un motif. Nous en avons identifié quatre. Premièrement, les objets présents dans les motifs doivent être plausibles dans la réalisation de l'activité spécifique. Deuxièmement, si le motif comporte des verbes, ceux-ci doivent également être plausibles par rapport à l'activité et aux objets auxquels ils sont associés. Troisièmement, l'ordre des objets (ou des paires verbe-objet) doit être plausible pour l'activité réalisée. Enfin, le motif doit être aussi spécifique que possible à l'activité en question, c'est-à-dire qu'il doit s'appliquer au nombre minimum d'activités possibles.

Il est important de noter la dépendance qui existe entre ces caractéristiques. En effet, se demander si les verbes associés sont plausibles n'a de sens que si les objets associés sont eux-mêmes plausibles. De même, se demander si un motif est spécifique n'a de sens que s'il est plausible. Les deux observations précédentes nous ont amené à concevoir le processus

Tableau 5.5 : Exemple d'annotation de motifs (chaque score est normalisé entre 0.0 et 1.0). Pour les détails du calcul de la spécificité, voici un exemple : (1 - 7/17) signifie qu'il y a 17 autres activités et que le motif n'est plausible que dans 7 de ces 17 autres activités. © Charles Cousyn, 2022.

Motif	Activité	Proportion du nombre d'objets plausibles	Proportion du nombre de verbes associés plausibles	Plausibilité de l'ordre (avec verbes)	Plausibilité de l'ordre (sans verbes)	Spécificité	Score d'annotation
[cup, tea]	make tea	1 (2/2)	Non applicable	Non applicable	1.0	1.0 (1 - 0/17)	1.0 (3/3)
[cup, tea]	make coffee	0.50 (1/2)	Non applicable	Non applicable	0.00	0.00	0.17 (0.5/3)
[fill||water, stir||pasta]	cook pasta	1.00 (2/2)	1.00 (2/2)	1.00	Non applicable	0.00	1.00 (4/4)
[add||salt]	cook pasta	1.00 (2/2)	1.00 (2/2)	1.00	Non applicable	0.59 (1 - 7/17)	0.90 (3.59/4)

d'annotation suivant. Pour chaque couple motif-activité, l'annotateur devait répondre à une série de questions avec le fonctionnement suivant : la question Q_i n'est posée que si la question $Q_{(i-1)}$ reçoit la réponse avec la valeur maximale. Dans le cas contraire, la valeur minimale était attribuée comme réponse aux questions qui auraient pu être posées. Les questions ordonnées sont les suivantes :

— Combien d'objets contenus dans le motif sont plausibles avec la réalisation de l'activité ?

— Combien de verbes associés contenus dans le motif sont plausibles avec l'objet et la réalisation de l'activité ?

— Entre 1 et 100, dans quelle mesure l'ordre du motif est-il plausible pour cette activité ?

— Dans combien d'activités autres que celle considérée, le motif serait-il plausible ?

Pour chaque question posée à l'annotateur, un score compris entre 0.0 et 1.0 est calculé et la moyenne est conservée comme score de l'annotation de la paire motif-activité. Un exemple de l'application de ce processus d'annotation est donné dans le Tableau 5.5.

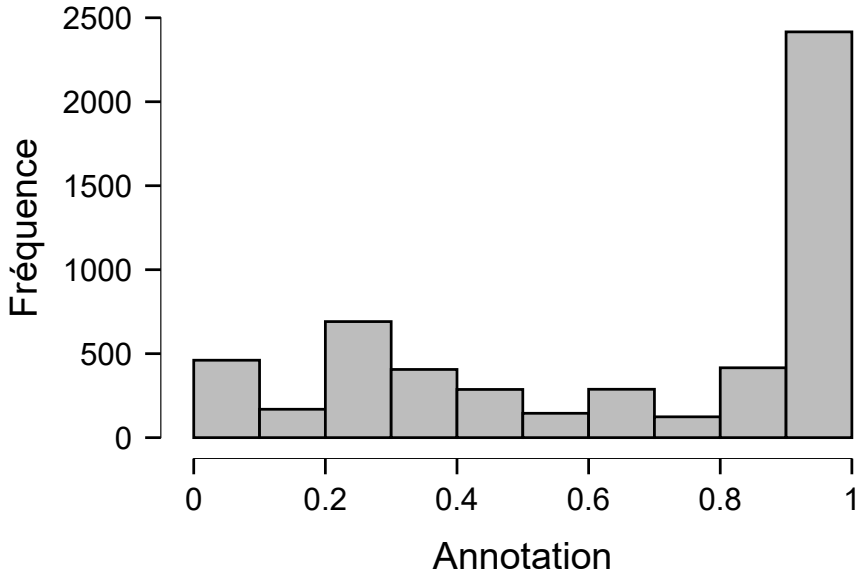

Figure 5.4 : Distribution statistique de la variable d'annotation. © Charles Cousyn, 2022.

5.4.4 RÉSULTATS

Dans cette section, après l'explication de l'objectif d'évaluation, de la méthodologie et du jeu de données, les résultats et l'analyse sont présentés. Chaque sous-section répond à une question spécifique présentée dans la Section 5.4.1.

IMPACT DES PARAMÈTRES SUR LA QUALITÉ DES MOTIFS

La première chose à considérer est l'impact des paramètres sur la qualité/annotation de nos motifs. Sur 5403 motifs annotés, la distribution du score n'est pas normale (voir Figure 5.4). Par conséquent, pour chaque paramètre de notre méthode, nous avons effectué un test statistique bayésien de Mann-Whitney U (van Doorn *et al.*, 2020) qui est un test non paramétrique ne nécessitant pas de données normalement distribuées. Les résultats de ces tests sont disponibles dans le Tableau 5.6.

Afin d'assurer la bonne compréhension des interprétations des résultats des tests statistiques qui suivent, nous rappelons deux choses. Premièrement, les résultats des tests sont consultables en ouvrant les fichiers **.jasp* dans notre dépôt [50] et, en particulier, les résultats interprétés dans la présente section sont issus du fichier nommé *patternData.jasp*. Deuxièmement, elles sont effectuées en utilisant les Tableaux 5.3 et 5.4 permettant respectivement de connaître la force de preuve de l'existence d'un effet et la taille de l'effet s'il existe.

Le premier résultat est que mettre le paramètre *genericOrSpecificParsing* à vrai semble diminuer légèrement la qualité de nos motifs. Cela n'est pas surprenant, car en mettant ce paramètre à vrai, cela signifie que les motifs sont générés à partir de pages web qui n'ont pas été sélectionnées par un individu mais plutôt par un moteur de recherche.

Deuxièmement, les résultats montrent que mettre à vrai le paramètre *verbAssociatorUsed* semble également diminuer faiblement la qualité de nos motifs. Cette diminution est plus importante que celle du paramètre *genericOrSpecificParsing*. Ce résultat peut être interprété comme le fait que l'association d'un verbe avec des objets tend à générer plus de mauvais motifs que de bons. Les motifs de paires verbe/objet de bonne qualité semblent donc être plus difficiles à obtenir que les motifs d'objets simples de bonne qualité. Cela semble logique, car l'une des représentations est plus expressive que l'autre et donc plus sujette aux erreurs lors de sa création.

En ce qui concerne le paramètre *verbAssociatorProximityBasedOrSyntacticBased*, le test bayésien Mann-Whitney U montre qu'il est plus plausible que le paramètre n'ait pas d'impact sur la qualité des motifs générés. Plus concrètement, le test montre qu'il y a 4.827 (1/0.207) plus de chances de ne pas avoir un impact que d'en avoir un.

---

50. https://github.com/CharlesCousyn/webpages_objects_spm_extractor/tree/main/experimentationResults/JASPFiles

Tableau 5.6 : Résultats des tests bayésiens de Mann-Whitney U sur l'annotation (qualité du motif) pour chaque paramètre. Dans notre cas, une taille d'effet positive/négative signifie respectivement une diminution/augmentation de la variable d'annotation lorsque le paramètre a la valeur vraie. © Charles Cousyn, 2022.

Paramètre	BF_{10}	Taille d'effet médian	Intervalle de crédibilité de la taille d'effet (95%)
genericOrSpecificParsing	1716.102	0.141	[0.087, 0.196]
verbAssociatorUsed	3.644e+8	0.275	[.221, 0.330]
verbAssociator ProximityBasedOr SyntacticBased	0.207	-0.058	[-0.118, 0.003]
hypernymMutation	219.218	0.133	[0.078, 0.190]
closedOrMaximalPattern	4.712	-0.093	[-0.147, -0.038]

Pour le paramètre *hypernymMutation*, les résultats tendent à montrer que le fait de mettre ce paramètre à vrai tend à diminuer faiblement la qualité de nos motifs. Cela indique que la mutation des hyperonymes ne serait pas un mécanisme utile pour générer de meilleurs motifs.

Enfin, pour le paramètre *closedOrMaximalPattern*, l'hypothèse selon laquelle il a un effet sur la qualité des motifs est 4.712 fois plus plausible selon le test bayésien de Mann-Whitney U. La force de la preuve est donc substantielle (voir Tableau 5.3). Si cet effet existe, mettre le paramètre *closedOrMaximalPattern* à vrai aurait tendance à augmenter la qualité des motifs entre très faiblement et faiblement. En d'autres termes, en supposant qu'il ait effectivement un impact, il serait préférable d'extraire des motifs fermés plutôt que des motifs maximaux.

LA MEILLEURE CONFIGURATION DE PARAMÈTRES

Grâce à la section précédente, nous connaissons l'impact global de chaque paramètre sur la qualité de nos motifs. Cependant, certaines combinaisons particulières de valeurs de paramètres peuvent ne pas suivre les tendances générales qui ont été déterminées. Il serait alors

Tableau 5.7 : Top 5 des meilleures combinaisons de paramètres (classées par ordre décroissant de l'annotation médiane). © Charles Cousyn, 2022.

genericOr Specific Parsing	verbAssociator Used	verbAssociator ProximityBasedOr SyntacticBased	hypernym Mutation	closedOr Maximal Pattern	Annotation médiane	Annotation écart interquartile
vrai	faux	vrai	faux	faux	0.98	0.358
faux	vrai	faux	faux	vrai	0.971	0.5
faux	vrai	faux	faux	faux	0.941	0.625
faux	vrai	faux	vrai	vrai	0.941	0.485
faux	faux	vrai	faux	vrai	0.941	0.706

intéressant de savoir quelles sont, dans les faits, les combinaisons de paramètres qui permettent d'obtenir les motifs de la plus haute qualité. La distribution de l'annotation n'étant pas normale, la médiane est utilisée comme indicateur de tendance centrale et l'écart interquartile comme mesure de dispersion. Le Tableau 5.7 fournit les 5 meilleures combinaisons de paramètres.

La première chose à remarquer est qu'il y a une très grande dispersion pour ce top 5. De ce fait, il est difficile d'établir avec certitude quelle combinaison est la meilleure. Néanmoins, on peut observer que les 5 meilleures combinaisons respectent, pour la plupart, les tendances déterminées dans la Section 5.4.4 (sauf pour le paramètre *verbAssociatorUsed*). En particulier, la 5e combinaison est celle qui respecte toutes ces tendances. Enfin, le lecteur pourrait être intéressé de savoir que la $10^{\text{ème}}$ combinaison avait une annotation médiane de 0.926, la $15^{\text{ème}}$ de 0.863, la $20^{\text{ème}}$ de 0.4 et la $24^{\text{ème}}$ de 0.375 avec un écart interquartile toujours supérieur à 0.34. Cela suggère que, bien que l'impact des combinaisons de paramètres soit difficile à évaluer en raison de la grande dispersion, les 5 meilleures combinaisons semblent préférables aux 5 moins bonnes (en utilisant l'annotation médiane comme critère de classement).

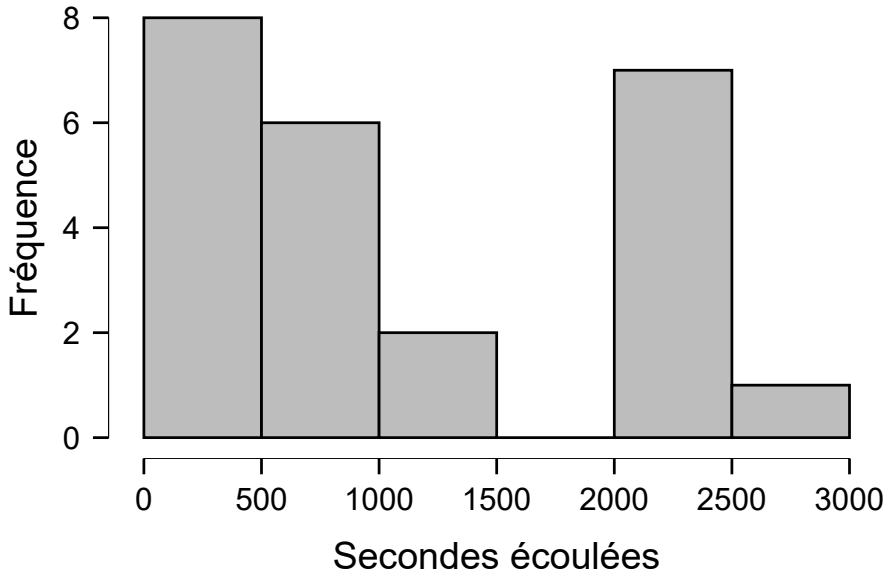

Figure 5.5 : Distribution du temps écoulé pour toutes les combinaisons possibles de paramètres. © Charles Cousyn, 2022.

IMPACT DES PARAMÈTRES SUR LE TEMPS DE CALCUL

Outre l'impact des paramètres de la méthode sur le score d'annotation, il est également important d'évaluer l'impact sur le temps de calcul. Comme la distribution du temps de calcul par configuration n'est pas normale (voir Figure 5.5), nous avons également effectué des tests bayésiens de Mann-Whitney U. Les résultats de ces tests sont disponibles dans le Tableau 5.8 et sont disponibles dans le fichier *config.jasp* de notre dépôt [51].

Pour le paramètre *genericOrSpecificParsing*, les résultats montrent très clairement que le fait de mettre ce paramètre à vrai tend à augmenter le temps de calcul pour obtenir nos motifs avec une intensité au moins faible mais potentiellement énorme (taille d'effet). Ce résultat est attendu parce que mettre ce paramètre à vrai signifie utiliser des pages web provenant

51. https ://github.com/CharlesCousyn/webpages_objects_spm_extractor/tree/main/experimentationResults/JASPFiles

Tableau 5.8 : Résultats des tests bayésiens de Mann-Whitney U sur le temps de calcul pour chaque paramètre. Dans notre cas, une taille d'effet positive/négative signifie respectivement une diminution/augmentation du temps de calcul lorsque le paramètre a la valeur vraie. © Charles Cousyn, 2022.

Paramètre	BF_{10}	Taille d'effet médian	Intervale de crédibilité de la taille d'effet (95%)
genericOrSpecificParsing	32.253	-1.335	[-2.315, -0.411]
verbAssociatorUsed	0.410	-0.001	[-0.745, 0.735]
verbAssociator ProximityBasedOr SyntacticBased	0.415	-0.069	[-0.826, 0.655]
hypernymMutation	0.581	0.317	[-0.360, 1.078]
closedOrMaximalPattern	0.440	0.181	[-0.487, 0.896]

d'un moteur de recherche ; ces pages étant beaucoup plus nombreuses (2523) que celles sélectionnées manuellement (370). En ce qui concerne tous les autres paramètres, les résultats semblent indiquer une absence d'effet de ces derniers, car les facteurs de Bayes obtenus sont tous proches de 0.5 et donc tous inférieurs à 1.0. Cependant, il est important de noter que les facteurs de Bayes associés ne permettent pas de trancher facilement entre l'existence ou l'absence d'un effet, car ils restent très proches de 1.0 (voir Tableau 5.3).

CORRÉLATION ENTRE PF-IAF ET ANNOTATION

La dernière question étudiée pour l'évaluation de la méthode est de savoir si la mesure PF-IAF est une bonne mesure de la pertinence des motifs ou non. Étant donné que la distribution de la mesure PF-IAF (voir Figure 5.6) et la variable d'annotation (voir Figure 5.4) ne suivent pas une distribution normale, un test de corrélation bayésien de Kendall tau-b (van Doorn *et al.*, 2018) est effectué. En effet, ce test nécessite seulement que les données soient ordinales ou continues, et non qu'elles suivent des distributions normales.

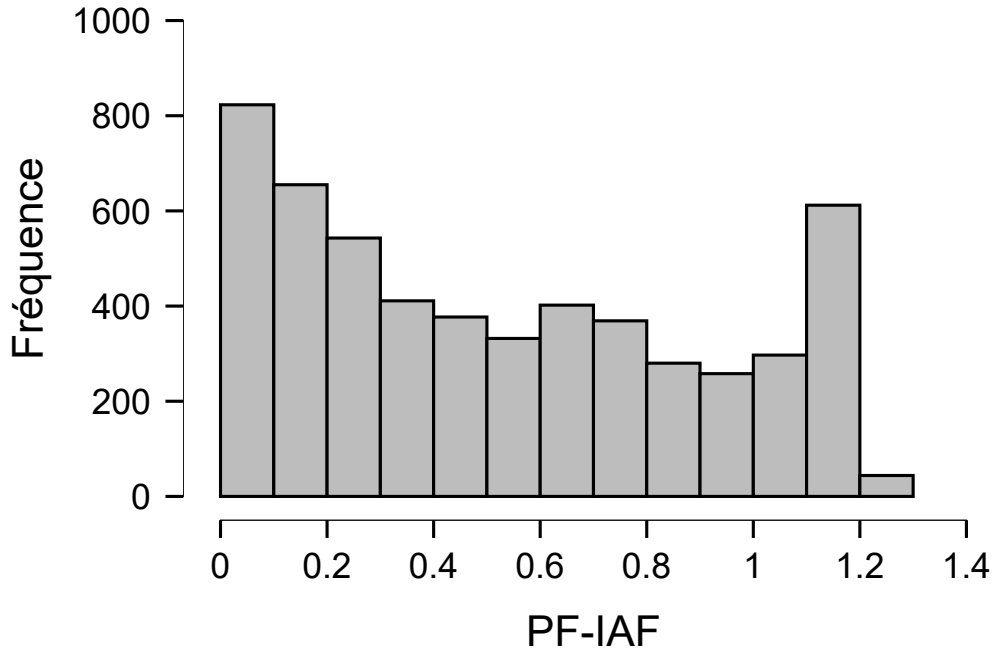

Figure 5.6 : Distribution de la variable PF-IAF. © Charles Cousyn, 2022.

Les résultats de ce test montrent très clairement que la corrélation est quasi inexistante, bien que légèrement positive (BF_{+0} = 9,013e+19, tau-b médian de Kendall = 0.090, intervalle de crédibilité à 95% du tau-b de Kendall : [0.072, 0.108]). Notez que bien que le tau-b de Kendall puisse être considéré comme une taille d'effet (ici, il s'agit de la force de la corrélation), son interprétation est différente de celle fournie dans le Tableau 5.4. Une différence majeure est que le tau-b de Kendall a une plage de [-1.0, 1.0], 1.0 indiquant une corrélation positive parfaite et -1.0 une corrélation négative parfaite. En résumé, selon l'analyse effectuée, PF-IAF est une mesure qui peut attester de la qualité d'un motif, mais en raison de sa corrélation d'environ 0.09 seulement, elle ne peut pas être utilisée comme seul indicateur de la qualité des motifs. Il faudrait l'améliorer ou la combiner avec d'autres indicateurs.

5.4.5 DISCUSSION

L'évaluation des motifs extraits peut être résumée comme suit. Tout d'abord, en ce qui concerne l'impact des paramètres de notre méthode sur la qualité des motifs, plusieurs d'entre eux ont un impact. Premièrement, sans surprise, l'utilisation du traitement générique ne semble diminuer que légèrement la qualité des motifs. C'est une bonne nouvelle, car le traitement générique peut traiter n'importe quelle page et est beaucoup plus généralisable. Ensuite, l'ajout d'un verbe associé à un objet dans nos motifs ne semble pas permettre d'obtenir des motifs de meilleure qualité en moyenne. De même, lorsque celui-ci est associé à un verbe, le choix du verbe par proximité dans le texte ou dans l'arbre de dépendance syntaxique ne semble pas faire de différence notable dans la qualité des motifs. Ensuite, la mutation hyperonymique introduite semble détériorer légèrement la qualité des motifs. Enfin, le choix d'utiliser des motifs fermés ou maximaux semble favoriser sensiblement les motifs fermés. De plus, si l'on considère les 5 meilleures combinaisons de valeurs de paramètres, elles confirment largement les affirmations précédentes.

Ensuite, concernant l'impact des paramètres sur le temps de calcul, seul le choix du traitement semble avoir un impact négatif significatif sur celui-ci. Il est important de noter que malgré ce constat, le temps d'obtention des motifs pour les 18 activités, même avec le traitement générique (qui traite 2523 pages au total), reste inférieur à 3000 secondes, soit moins d'une heure ; ce qui reste un temps parfaitement raisonnable pour obtenir des informations sur la réalisation des activités de la vie quotidienne.

Enfin, concernant la mesure PF-IAF introduite dans ce chapitre, les résultats semblent montrer que PF-IAF utilisée seule ne serait pas une bonne mesure de la qualité d'un motif, car la corrélation existante entre l'annotation et le véritable score est beaucoup trop faible (environ 0.09). Ce résultat montre qu'il est difficile de trouver une mesure de performance

non supervisée et que des recherches supplémentaires sont nécessaires pour trouver une ou plusieurs mesures satisfaisantes.

5.5 CONCLUSION

Dans ce chapitre, nous avons analysé une méthode non supervisée pour extraire des motifs séquentiels d'utilisation d'objets à partir de pages web. L'évaluation de l'approche a été faite à l'aide d'analyses statistiques bayésiennes comme le test bayésien de corrélation tau-b de Kendall et le test bayésien de Mann-Whitney U. Plusieurs paramètres de notre approche permettent de modifier son comportement. Ces paramètres sont : le traitement générique ou spécifique, l'association d'un verbe à nos objets, la recherche du verbe associé, l'utilisation de la mutation hyperonymique et l'extraction de motifs fermés ou maximaux.

Les résultats ont montré qu'à l'exception de la méthode pour trouver le verbe associé, tous les paramètres ont un impact sur la qualité des motifs extraits, permettant d'orienter le choix à faire sur la valeur de ces paramètres pour maximiser la qualité des motifs. Nous avons également montré que l'impact sur le temps de calcul existe, mais que le temps de calcul reste raisonnable. Enfin, nous avons montré qu'il est difficile d'obtenir une mesure de performance qui quantifie correctement la qualité d'un motif sans supervision humaine.

CHAPITRE VI
UTILISATION DE MOTIFS MINÉS POUR LA RECONNAISSANCE D'ACTIVITÉS DE LA VIE QUOTIDIENNE

Le chapitre précédent avait pour but d'extraire et d'évaluer des motifs séquentiels d'utilisation d'objets à partir de pages web. Nous avons évalué ces motifs avec l'aide d'un annotateur humain et grâce à cela nous avons pu évaluer l'extraction des motifs en elle-même en fonction des paramètres de notre méthode. Cependant, cette manière d'évaluer la méthode a deux défauts. D'abord, elle n'évalue pas l'utilisation des motifs dans un contexte réel d'HAR, c'est-à-dire dans un habitat intelligent. Ensuite, elle est sujette au biais d'évaluation de l'annotateur humain qui peut toujours se tromper ou plus probablement évaluer les motifs selon sa conception subjective des activités de la vie quotidienne. Par exemple, une activité *make tea* peut être réalisée de manières très distinctes en fonction des origines, de la position géographique et de la culture de ceux qui la réalisent.

Le Chapitre 4 avait un but similaire à la différence que le but était d'extraire les objets impliqués dans la réalisation des activités de la vie quotidienne à partir d'images trouvées sur le web. Sur le même principe, nous avons évalué les objets extraits à partir d'un jeu de données de référence construit à la main par un humain (Section 4.2.1). Cependant, pour les mêmes raisons que celles évoquées dans le paragraphe précédent, il serait intéressant d'exploiter ces objets extraits comme le fait Riboni & Murtas (2019, 2017) dans un contexte réel d'HAR.

C'est dans ce contexte que s'inscrit le présent chapitre dont le but est de placer les motifs séquentiels et les objets extraits dans un contexte réel. La question de recherche à laquelle nous souhaitons répondre est donc la suivante : dans quelles mesures les motifs d'utilisation

d'objets et les objets extraits à partir d'images peuvent être exploités pour effectuer de l'HAR dans un habitat intelligent ?

Pour répondre à cette question, nous utilisons l'infrastructure du LIARA (Laboratoire d'Intelligence Ambiante pour la Reconnaissance d'Activités), disponible à l'Université du Québec à Chicoutimi. Le LIARA est un laboratoire dans lequel un habitat intelligent est mis à disposition et dans lequel il est possible de récupérer facilement des données. Nous présentons le laboratoire et l'ensemble des outils disponibles dans la Section 6.1.1.

Notre contribution théorique est formée d'un algorithme d'HAR utilisant les motifs d'utilisation d'objets. La contribution pratique est une implémentation en JavaScript de ces algorithmes [52] ainsi qu'un jeu de données récolté au sein du LIARA [53]. Pour finir, la contribution expérimentale est l'analyse des performances de l'algorithme de reconnaissance pour 5 activités de la vie quotidienne.

6.1 MÉTHODOLOGIE

Pour expliquer la méthodologie utilisée, nous présentons d'abord le LIARA, laboratoire où les expérimentations d'HAR seront menées. Ensuite, nous présenterons les données utilisées ainsi que les activités considérées. Enfin, l'algorithme de reconnaissance utilisé sera expliqué et détaillé.

6.1.1 LIARA ET RÉCOLTE DE DONNÉES

Le LIARA est un laboratoire de recherche situé à l'Université du Québec à Chicoutimi. Il a pour objectif de développer des technologies qui permettent d'étendre l'environnement

52. https://github.com/CharlesCousyn/webpages_objects_spm_extractor/blob/main/HAR.js

53. https://github.com/CharlesCousyn/webpages_objects_spm_extractor/tree/main/patternUse/dataset

Figure 6.1 : L'habitat intelligent du LIARA (Bouchard *et al.*, 2014). Reproduit avec la permission de Kévin Bouchard

physique pour s'adapter à l'humain et créer des services et des dispositifs intelligents capables de répondre aux besoins des individus, des groupes et des sociétés. Il se présente sous la forme d'un habitat intelligent de près de 100 m² (Bouchard *et al.*, 2014) possédant une centaine de capteurs et effecteurs différents. Parmi les capteurs, on trouve des capteurs infrarouges, des tapis de pression, des contacts électromagnétiques, divers capteurs de température, des capteurs de lumière, un analyseur de puissance intelligent et huit antennes RFID. Il dispose également de nombreux effecteurs, dont un iPad d'Apple, de nombreuses enceintes IP dans l'appartement, une télévision HD à écran plat, un *home* cinéma et de nombreuses lumières et des diodes électroluminescences (DEL) dissimulées à des endroits stratégiques. Plusieurs photos du laboratoire sont disponibles dans la Figure 6.1.

L'ensemble de ses éléments font que mener des expérimentations d'HAR est tout à fait accessible au LIARA comme cela a pu être fait dans le passé (Bouchard *et al.*, 2020). C'est pour cette raison que pour le présent chapitre, nous utiliserons le LIARA pour mener nos expérimentations.

Étant donné que l'objectif est d'utiliser les motifs d'utilisation d'objets et les objets extraits à partir d'images au sein de cet habitat intelligent, la question que l'on doit se poser est donc de savoir comment mettre en lien ces éléments et les capteurs du laboratoire. Si l'on regarde de quoi ils sont constitués, on peut le résumer comme suit. Définis dans la Section 5.2.2, les motifs d'utilisation d'objets sont des séquences de chaînes de caractères ; ces chaînes pouvant représenter soit des objets, soit des couples verbe-objet. Les objets extraits à partir d'images sont des listes non ordonnées de chaînes de caractères, chaque chaîne représentant un objet.

Du côté du laboratoire, il faut donc trouver quels capteurs sont pertinents à utiliser pour obtenir des correspondances avec l'utilisation des ces objets. Notre réflexion nous a menés vers l'utilisation de deux types de capteurs différents.

Le premier est l'analyseur de puissance intelligent qui est exploité pour analyser le signal électrique de l'habitat pour effectuer de la reconnaissance d'appareils électriques. En effet, nous pouvons très facilement faire le lien entre le fait qu'un appareil soit allumé au fait qu'un appareil soit utilisé. Par exemple, un grille-pain restera allumé seulement pendant son utilisation. Ce lien n'est pas valide uniquement pour les appareils qui restent en fonctionnement 24h sur 24 (comme un réfrigérateur). Une couche logicielle se basant sur les travaux de Maitre *et al.* (2015) pour effectuer de la reconnaissance d'appareils électriques est disponible au laboratoire sous la forme d'un WebSocket (MDN contributors, 2022).

Les seconds sont les 8 antennes RFID permettant de localiser les objets possédant une étiquette RFID. Nous justifions l'utilisation de ce type de capteurs par le fait qu'il est aujourd'hui possible de localiser efficacement une étiquette RFID par trilatération (Bouchard *et al.*, 2013). Avec le bon algorithme de positionnement, les antennes RFID exposent le mouvement des objets ; et ce mouvement peut être facilement transformé en notion d'utilisation. Une couche logicielle permettant d'obtenir les positions de chaque objet à une fréquence de 1 Hz est disponible au laboratoire sous la forme d'un WebSocket (MDN contributors, 2022).

Une remarque qui doit être exprimée concerne les motifs d'utilisation d'objets. Comme définis dans la Section 5.2.2, ces derniers peuvent être des motifs d'objets simples ou des motifs de couple verbe/objet. Si nous souhaitons exploiter à son plein potentiel le second type de motif, alors en plus d'avoir une correspondance entre capteur et objet, il est nécessaire d'avoir une correspondance entre verbe, objet et capteur. Il est déjà peu évident de faire une correspondance entre un capteur et un objet dans un contexte réel, alors nous avons décidé de mettre de côté les motifs de couples verbe/objet dans ce chapitre, car nous estimons qu'ajouter la notion de verbe associé à un objet dans un contexte réel d'HAR est au moins un ordre de grandeur plus complexe que de faire une association entre capteur et objet. Il ne faut pas non plus oublier l'un des résultats du Chapitre 5 qui est qu'ajouter un verbe à chaque objet ne semblait pas améliorer la qualité des motifs (Section 5.4.4).

Établir un lien entre les données fournies par un capteur et un objet est une tâche complexe dont l'évidence peut varier en fonction du contexte, de la force et de la nature de la relation entre la donnée du capteur et l'objet. Par exemple, si un capteur fournit la donnée *Coffee* à un instant t, et que nos motifs contiennent le mot *coffee* sans majuscule, il semble évident de lier ces deux éléments ensemble pour bien faire comprendre qu'ils font référence au même concept. Cependant, dans le cas d'un capteur fournissant la donnée *Cafetière ON*, devrait-on lui associer nécessairement le mot *coffee* provenant d'un motif ? Même si ces

deux données semblent très liées, notamment par le fait qu'une cafetière utilise du café, elles ne font pas référence au même concept. Cette difficulté fait que dans le cadre de notre approche d'HAR, pour que les motifs soient exploités, il est nécessaire d'établir concrètement la correspondance entre les données fournies par les capteurs réels de l'environnement et les mots utilisés dans les motifs d'utilisation d'objets. C'est pour cette raison que nous avons tenté d'établir un dictionnaire pour chaque activité considérée [54]. Nous reviendrons sur ce dictionnaire et les limites qu'il impose dans la Section 6.2.4.

6.1.2 JEU DE DONNÉES ET ACTIVITÉS CONSIDÉRÉES

Maintenant que nous savons quels capteurs et outils devraient être utilisés au LIARA, nous devons savoir sur quelles activités vont être menées nos expérimentations d'HAR. Les objets extraits à partir d'images et les motifs d'utilisations d'objets extraits ont pour source des pages web ou des requêtes qui concernent un ensemble de 18 activités dont nous avons déjà parlé dans la Section 5.4.2. Ces activités sont les suivantes : *fill medicine cabinet*, *watch tv*, *water plants*, *answer the phone*, *prepare a birthday card*, *prepare soup*, *clean*, *vacuum*, *choosing outfit*, *make tea*, *make coffee*, *cook pasta*, *do homework*, *play video games*, *bake parmesan turkey meatballs*, *make hut stuffed crust pizza*, *make mashed potato casserole*, *make low carb pancakes*. Dans l'absolu, nous pourrions tenter de faire de l'HAR sur l'ensemble de ces activités, mais il faut porter une attention particulière aux objets qu'il est possible d'utiliser. En effet, il ne faut pas oublier que nous utilisons deux capteurs distincts : des capteurs RFID nécessitant des étiquettes RFID et un analyseur de puissance qui analyse l'utilisation d'appareils électriques. Donc, les objets que nous pouvons considérer doivent être des appareils électriques ou des objets sur lesquels une étiquette RFID peut être posée. Ces

54. Le dictionnaire est disponible dans notre dépôt https://github.com/CharlesCousyn/webpages_objects_spm_extractor/blob/main/patternUse/experimentationConfig.json, dans l'attribut nommé *objectDictionaries*

objets doivent également être présents dans les motifs d'utilisation d'objets de nos activités. L'ensemble de ces contraintes fait que l'ensemble des objets utilisables est restreint, ce qui, par la même occasion, restreint les activités reconnaissables. Les capteurs et objets disponibles au LIARA font que, dans ce chapitre, nous construisons une approche d'HAR avec un sous-ensemble de nos 18 activités constitué des 5 activités suivantes : *clean*, *vacuum*, *make tea*, *make coffee* et *cook pasta*.

6.1.3 PROCESSUS DE RECONNAISSANCE

L'objectif de cette section est de décrire complètement le processus de reconnaissance utilisé. Afin d'avoir une vue d'ensemble, comme dans les Chapitres 4 et 5, notre méthode possède plusieurs paramètres qui seront détaillés par la suite.

PRÉTRAITEMENT

Afin de pouvoir utiliser un algorithme de reconnaissance correctement, les données d'entrée subissent un prétraitement permettant leur pleine utilisation. Ce prétraitement consiste en deux parties.

La première partie du prétraitement concerne les objets extraits à des images venant du web, en lien avec la contribution du Chapitre 4. Dans un souci d'uniformiser les données d'entrée de l'algorithme de reconnaissance, nous avons fait le constat suivant. La seule différence entre les motifs d'utilisation d'objets et les objets extraits à partir d'images est que les premiers sont ordonnés alors que les seconds ne le sont pas. Dans cette situation, nous nous sommes dits qu'en réalité, un objet extrait à partir d'images est parfaitement équivalent à un motif d'objets simples ne contenant qu'un unique objet. En effet, une liste de taille 1 est toujours ordonnée. Nous avons donc entrepris de convertir les listes d'objets extraites grâce à

la méthode du Chapitre 4 en motifs d'objets simples. Le code de conversion est disponible est notre dépôt [55]. Grâce à cette manoeuvre, nous pourrons concevoir un unique algorithme qui prend en entrée des motifs d'objets simples. Nous précisons également que le fonctionnement de cette conversion possède quelques caractéristiques particulières. D'abord, nous limitons le nombre maximal de motifs par activité à cause du fait que la complexité algorithmique du processus est exponentielle. En effet, si nous ajoutons trop de motifs, le temps pris par notre algorithme pour effectuer l'HAR n'est plus raisonnable. Nous fixons ce nombre maximal à 10 en raison de nos tests préliminaires. Ensuite, les scores de pertinence de chaque objet obtenus grâce au processus d'agrégation (visible en Figure 4.2) n'étant pas situés entre 0.0 et 1.0, nous effectuons une normalisation des scores de pertinence entre 0.0 et 1.0 à l'aide d'une mise à l'échelle min-max [56] afin de s'aligner avec nos motifs d'utilisation d'objets extraits de pages web.

La deuxième partie du prétraitement concerne les données fournies par le WebSocket RFID. Comme mentionné plus haut, pour savoir si un objet est en cours d'utilisation, nous nous basons sur la variation de sa position (son mouvement). Le WebSocket ne fournissant que des positions dans l'espace pour les objets étiquetés, nous avons développé un algorithme de détection de mouvements significatifs nommé $significantMovement$ [57] capable, à partir des positions et de 3 paramètres, de détecter si un mouvement est assez significatif pour considérer que l'objet est en train d'être utilisé. Les paramètres de cet algorithme sont une taille de fenêtre, un seuil minimal et un seuil maximal. Notre équipe a trouvé les valeurs les plus performantes grâce à une méthode essai/erreur avec les objets étiquetés du LIARA.

---

55. https://github.com/CharlesCousyn/webpages_objects_spm_extractor/blob/main/convertImExToSPMEx.js

56. https://en.wikipedia.org/wiki/Feature_scaling#Rescaling_(min-max_normalization)

57. https://github.com/CharlesCousyn/webpages_objects_spm_extractor/blob/main/patternUse/calibrationRFIDPosition.js

ALGORITHMES DE RECONNAISSANCE

L'algorithme de reconnaissance est le coeur d'une approche d'HAR efficace et performante. Comme mentionné précédemment, notre algorithme de reconnaissance doit permettre d'utiliser les motifs d'utilisation d'objets que nous avons extraits. Nous avons donc conçu un algorithme nommé HAROUP (Human Activity Recognition using Object Usage Patterns) dont le pseudo-code est disponible dans l'Algorithme 6.1. En dehors du flux d'événements et des activités considérées, HAROUP possède deux paramètres qui doivent être expliqués. Le premier est la taille de la fenêtre glissante utilisée (une taille en termes de temps, en secondes), le second est l'ensemble des motifs utilisés. Ce sont en fonction de ces deux paramètres que l'évaluation des performances se fera. Le fonctionnement de l'algorithme est le suivant. Chaque événement est d'abord étiqueté en tant que *noActivity* comme classe prédite par défaut. Ensuite, on récupérera l'ensemble des événements à gauche et à droite (les événements sont classés dans l'ordre chronologique) en plaçant l'événement considéré au milieu d'une fenêtre de taille *windowSize*. Pour chaque activité considérée et pour chaque événement (et sa fenêtre), on calcule alors un score de la manière suivante. On trouve les motifs liés à l'activité considérée. Parmi ceux-ci, on garde les motifs activés dans la fenêtre. Un motif est dit activé s'il apparaît au moins une fois dans la fenêtre dans le sens du SPM. Par exemple, pour la fenêtre [*spoon*, *fork*, *pasta*, *apple*, *spoon*], les motifs [*spoon*], [*spoon*, *fork*] et [*fork*, *apple*] apparaissent alors que les motifs [*apple*, *pasta*] et [*pasta*, *fork*] n'apparaissent pas. On va ensuite sommer l'ensemble des poids associés de tous les motifs activés et on va mémoriser cette somme pour l'activité considérée. Une fois cela fait, les sommes de poids de toutes les activités sont normalisées, classées par ordre décroissant. L'activité ayant la plus haute somme est alors utilisée pour étiqueter l'événement considéré.

Pour rappel, les poids utilisés lors de la sommation et de la normalisation décrites sont des poids associés à un motif et une activité particulière. Ces poids ont été obtenus par le processus d'annotation décrit en Section 5.4.3 pour les motifs d'utilisation d'objets provenant de pages web (méthode du Chapitre 5) ou par le jeu de données de référence décrit en Section 4.2.1 pour les motifs d'utilisation d'objets obtenus par la conversion des listes d'objets extraites à partir d'images (méthode du Chapitre 4).

Données : *events* : événements avec un nom et un timestamp, *windowSize* : taille de la fenêtre glissante, *patterns* : motifs d'utilisation d'objets utilisés, *activities* : activités reconnaissables
Résultat : *events* : événements avec un nom, une étiquette et un timestamp

pour $e \in events$ **faire**
    Initialisation;
    $e.label \leftarrow$ "noActivity";
    Obtention des événements situés entre e.timestamp $- \frac{windowSize}{2}$ et e.timestamp $+ \frac{windowSize}{2}$;
    $cut \leftarrow cutTrace(e, windowSize)$;
    Calcul des scores de pertinences pour chaque activité;
    $listWeigths \leftarrow []$;
    pour $a \in activities$ **faire**
        Recherche des motifs liés à l'activité a;
        $potentialPatterns \leftarrow filter(patterns, a)$;
        Recherche des motifs activés dans la fenêtre d'événements;
        $activePatterns \leftarrow findActivePatterns(cut, potentialPatterns)$;
        On somme les poids des motifs actifs;
        $totalWeight \leftarrow sumWeightActivePatterns(activePatterns)$;
        $listWeigths.push(totalWeight)$;
    fin
    Normalisation des poids pour avoir une somme de poids égale à 1.0;
    $normListWeigths \leftarrow normalizeBySum(listWeigths)$;
    Trier les activités par poids décroissant;
    $sortedActivities \leftarrow orderActivitiesByDecreasingWeight(activities, listWeights)$;
    $bestActivityLabel \leftarrow sortedActivities[0]$;
    Étiqueter chaque événement avec l'activité de plus haut score;
    $e.label \leftarrow bestActivityLabel$;
fin

Algorithme 6.1 : HAROUP (Human Activity Recognition using Object Usage Patterns)

6.1.4 LES MOTIFS UTILISÉS

Comme précisé dans la Section 6.1.3, nos motifs d'utilisation d'objets peuvent avoir deux sources distinctes. La première est l'ensemble des pages web minées avec la méthodologie du Chapitre 5 et la seconde est les images analysées avec la méthodologie du Chapitre 4. Nous avons comme hypothèse qu'utiliser des motifs de sources différentes peut amener à des résultats de reconnaissance différents. C'est pour cette raison que nous introduisons dans notre approche un paramètre nommé *useImageExtractorPatternsOrSPMPatterns*. Ce paramètre peut avoir trois valeurs différentes : *IMAGE_EXTRACTOR*, *SPM* ou *BOTH*. La première valeur signifie que la source de motifs est l'ensemble des images analysées, la seconde valeur signifie que la source est l'ensemble des pages web minées et la troisième les deux. En ce qui concerne la troisième valeur possible, utiliser les deux sources de motifs signifie d'effectuer une union des deux ensembles de motifs. Par exemple, si pour l'activité *make tea*, il existe les motifs [*water*], [*spoon*] extraits des images et les motifs [*teabag*, *spoon*], [*spoon*] extraits des pages web, alors les motifs utilisés pour l'activité *make tea* dans le cas où le paramètre *useImageExtractorPatternsOrSPMPatterns* a pour valeur BOTH seront [*water*], [*spoon*] et [*teabag*, *spoon*].

Une deuxième chose à rappeler est que les motifs extraits, que ce soit par la méthode du Chapitre 4 ou celle du Chapitre 5, le sont en fonction de plusieurs paramètres. Pour les motifs extraits à partir de pages web, nous avons sélectionné la combinaison de paramètres suivante : *[genericOrSpecificParsing : faux, verbAssociatorUsed : faux, verbAssociatorProximityBasedOrSyntacticBased : true, hypernymMutation : faux, closedOrMaximalPattern : vrai]*. Pour justifier cette combinaison, nous faisons les remarques suivantes. Premièrement, la combinaison se trouve dans le top 5 des meilleures combinaisons de paramètres (voir le Tableau 5.7). Deuxièmement, c'est une combinaison n'utilisant pas l'association avec un verbe. Troisièmement, cette combinaison n'utilise pas le traitement générique de pages web (gene-

ricOrSpecificParsing : faux) ; qui lors de nos tests préliminaires ne permettait pas d'obtenir des performances satisfaisantes d'HAR. Une autre chose importante est que nous n'utilisons pas la totalité des motifs issus de cette combinaison, car tous les motifs ne sont pas pertinents. Grâce au fait que les motifs ont été annotés pour la phase d'évaluation du Chapitre 5, nous utilisons cette annotation (un score entre 0.0 et 1.0) pour garder les motifs les plus pertinents. Nous avons placé un seuil minimal à 0.8, car la distribution de l'annotation visible en Figure 5.4 nous montre qu'une très grande partie des motifs avec un fort score se trouve au-dessus de 0.8.

Pour les motifs extraits à partir des images, notre choix de la combinaison de paramètres s'est porté sur la combinaison : *pnasnet_large duckduckgo 25* (voir la Section 4.2.2 pour comprendre dans les détails). Nous justifions cette combinaison par les affirmations suivantes qui s'appuient sur nos propos tenus en Section 4.3.4. D'abord, nous avons déterminé qu'utiliser DuckDuckGo ou Google comme moteur de recherche d'images ne faisait pas de différences. Nous avons donc choisi un moteur de recherche au hasard et nos tests préliminaires ont confirmé l'absence de différences en termes de performances. Ensuite, nous avons déterminé qu'utiliser 25 images était amplement suffisant pour extraire les objets pertinents liés aux activités quotidiennes. Dans les faits, également pendant nos tests préliminaires, nous n'avons pas vu de différences en fonction du nombre d'images utilisées. Enfin, dans la Section 4.3.4, nous avons énoncé qu'utiliser le modèle de détection d'objets yolov3 était probablement préférable aux autres modèles. Cependant, nous avons mentionné que les objets détectables par yolov3 sont limités et que plusieurs objets très pertinents de nos activités ne font pas partie des 80 objets détectables par le modèle yolov3. Le problème de cette limitation est que les objets fournis par ce modèle pourraient rendre encore plus difficile la correspondance entre données de capteurs et objets (Section 6.1.1). Nous avons donc fait le choix d'utiliser une combinaison de paramètres utilisant un autre modèle, à savoir *pnasnet_large*, car c'est l'un

des deux modèles de classification d'images qui a eu la meilleure performance moyenne et la meilleure performance quand il est combiné avec un petit nombre d'images et le moteur de recherche DuckDuckGo (Figures 4.8 et 4.12).

6.2 ÉVALUATION DES PERFORMANCES

L'objectif de cette section est de présenter l'évaluation des performances de notre approche d'HAR utilisant des motifs d'utilisation d'objets. Nous commencerons par énoncer l'objectif de cette évaluation, puis sa méthodologie pour ensuite présenter les résultats obtenus. Ces résultats sont ensuite interprétés et discutés. Enfin, nous verrons les limitations de notre approche et quels sont les points à améliorer.

6.2.1 OBJECTIF DE L'ÉVALUATION ET MESURES UTILISÉES

L'objectif de l'évaluation des performances est de savoir si les motifs extraits par la méthode introduite dans le Chapitre 5 peuvent être utilisés pour effectuer l'HAR de manière efficace. Pour notre problème, nous considérons 5 activités. Il s'agira ainsi d'un problème de classification avec 6 classes : une par activité et une classe *noActivity*.

Dans le domaine de l'HAR, les mesures de performance les plus utilisées sont la justesse, le F-score, le kappa de Cohen. En plus de celles-ci, nous utiliserons le MCC également, car cette mesure semble présenter plusieurs avantages que nous avons mentionnés dans la Section 2.1.3. La justesse, le kappa de Cohen, le MCC s'appliquent naturellement à des problèmes de classification multiclasses, mais le F-score est une mesure conçue initialement pour les problèmes de classification binaire. Pour résoudre cela, il suffit de calculer le F-score par classe et de faire une moyenne sur toutes les classes. Nous utiliserons donc la moyenne arithmétique des F-score par classe (aussi appelée *macro-averaging F-score* (Opitz & Burst, 2021)).

6.2.2 MÉTHODOLOGIE D'ÉVALUATION

Pour atteindre cet objectif, nous allons analyser les performances de reconnaissance en fonction des deux paramètres que nous avons introduits dans ce chapitre, à savoir :

— *useImageExtractorPatternsOrSPMPatterns* : un paramètre permettant de choisir la/les source(s) de motifs d'utilisation d'objets qui seront utilisées (IMAGE_EXTRACTOR, SPM ou BOTH),

— *windowSize* : la taille de fenêtre glissante utilisée dans l'algorithme HAROUP.

En ce qui concerne le choix des valeurs considérées pour le paramètre *windowSize*, nos tests préliminaires ont permis de constater que les meilleures performances étaient atteintes avec des valeurs supérieures à 5 secondes. Nous avons donc fait le choix de tester 16 valeurs entre 5 secondes et 20 secondes, à savoir l'intervalle [5, 20] avec un pas de 1.

Avec les deux paramètres et leurs valeurs possibles, il existe 3x16 = 48 combinaisons de paramètres différents. Nous pensons qu'avec ce nombre de combinaisons, des graphiques montrant la variation des mesures de performance en fonction des deux paramètres permettront d'évaluer l'influence de ces paramètres sans nécessiter d'analyse statistique plus poussée. Notre manière d'évaluer les performances consistera donc à interpréter ces graphiques et à les comparer.

6.2.3 RÉSULTATS ET INTERPRÉTATION

L'IMPACT DES PARAMÈTRES

Après avoir exécuté nos 48 combinaisons de paramètres, nous avons obtenu un ensemble de résultats pour chacune d'entre elles. Les résultats bruts avec les matrices de confusion sont disponibles dans notre dépôt dans le fichier *experimentationResults.json* (Cousyn, 2021). La

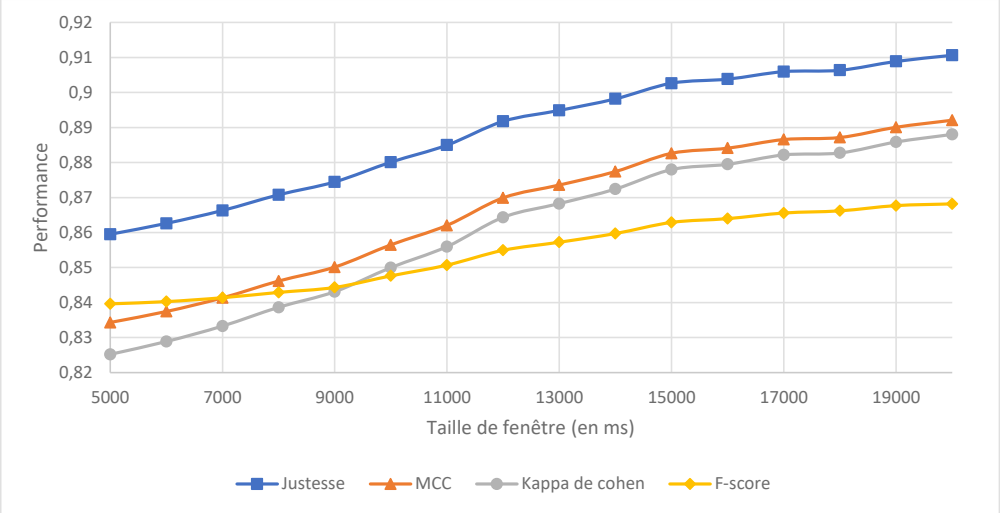

Figure 6.2 : Variation des performances en fonction de la taille de fenêtre (en ms) avec le paramètre de choix de motifs à BOTH. © Charles Cousyn, 2022.

manière la plus concise d'afficher les résultats et de faciliter l'interprétation est de produire 3 graphiques ; chacun montrant les combinaisons utilisant des valeurs de *useImageExtractorPatternsOrSPMPatterns* différentes (IMAGE_EXTRACTOR, SPM ou BOTH). Chacun de ces graphiques montre la variation des mesures de performance en fonction de l'autre paramètre, à savoir *windowSize*. Ces graphiques sont disponibles en Figures 6.2, 6.3 et 6.4.

En ce qui concerne la Figure 6.2, il semble assez clair que mettre le paramètre *useImageExtractorPatternsOrSPMPatterns* à la valeur *BOTH* semble garantir de bonnes performances, avec une justesse variant entre 0.86 et 0.91. Sur les 4 mesures de performance considérées, il semble qu'une augmentation de la taille de la fenêtre se traduit en une augmentation des performances. Pour la Figure 6.3, les performances semblent toutes avoir le même comportement : une très légère hausse jusqu'à 12000 ms puis un comportement stagnant ensuite. La justesse oscille entre 0.827 et 0.845. En ce qui concerne la Figure 6.4, la moyenne des F-score n'était pas valide mathématiquement (division par 0.0) et n'est donc pas affichée. Néanmoins,

cette figure semble montrer des performances très stables variant entre 0.32 et 0.35 pour la justesse même s'il semble exister un très léger effet positif de la taille de la fenêtre sur celle-ci.

Si l'on compare les trois figures, on peut alors facilement effectuer un classement des performances en fonction de la valeur du paramètre *useImageExtractorPatternsOrSPMPatterns*. La valeur IMAGE_EXTRACTOR semble fournir les performances les plus basses ne dépassant pas 0.45 pour la justesse. Elle est suivie de la valeur SPM qui permet d'avoir des performances très satisfaisantes aux alentours de 0.84 pour la justesse. Enfin la valeur, BOTH donne les meilleures performances allant jusqu'à 0.91 de justesse. On peut résumer ces remarques de la manière suivant : le choix des motifs d'utilisation d'objets utilisés dans notre approche a un véritable impact sur les performances. C'est un résultat attendu, car les motifs sont au coeur de notre approche de reconnaissance. On remarquera également que les motifs provenant du texte des pages web de l'approche décrite dans le Chapitre 5 semblent plus efficaces que ceux ayant pour origine les images utilisées dans l'approche du Chapitre 4 pour reconnaître les activités de la vie quotidienne. Enfin, les résultats montrent que combiner les deux types de motifs semble être l'approche préférable ; ce qui signifie que les motifs provenant des images sont capables de compléter l'information apportée par les motifs provenant des pages web.

LA MEILLEURE COMBINAISON DE PARAMÈTRES

Nous connaissons maintenant les effets et les impacts de nos deux paramètres sur les performances de manière globale, c'est-à-dire, sur toutes les activités. Il serait alors intéressant de voir dans le cas de la meilleure combinaison de paramètres quelles activités sont les mieux ou moins bien reconnues. Pour cela, il suffit d'observer la matrice de confusion de cette

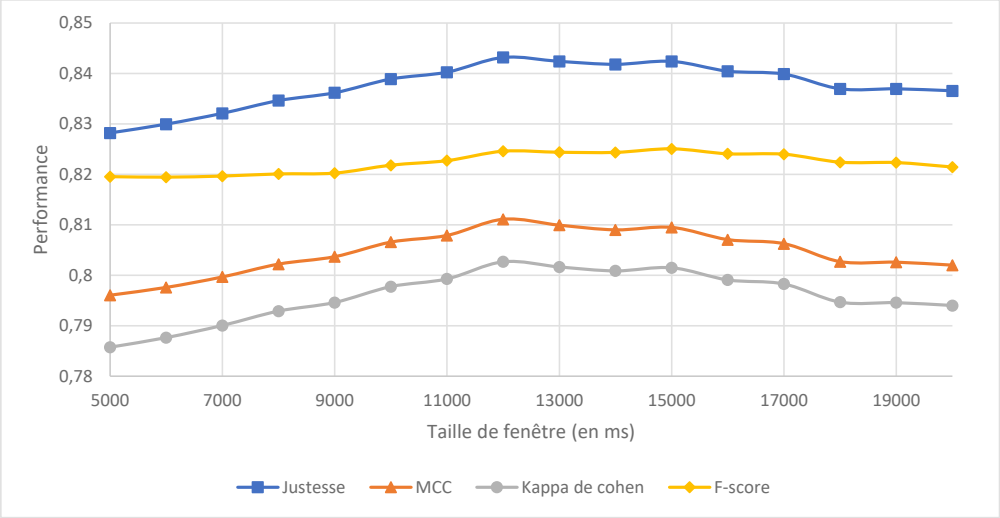

Figure 6.3 : Variation des performances en fonction de la taille de fenêtre (en ms) avec le paramètre de choix de motifs à SPM. © Charles Cousyn, 2022.

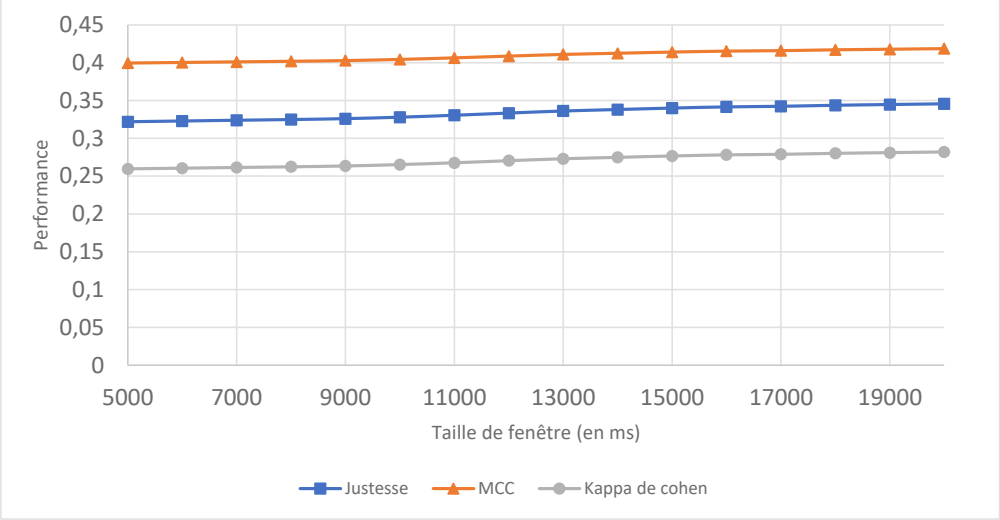

Figure 6.4 : Variation des performances en fonction de la taille de fenêtre (ms) avec le paramètre de choix de motifs à IMAGE_EXTRACTOR. © Charles Cousyn, 2022.

combinaison de paramètres disponible dans la Table 6.1 ainsi que les performances par activité disponible dans la Table 6.2.

La première chose que l'on peut noter est que l'activité *vacuum* est reconnue parfaitement. La raison est que l'objet le plus utilisé dans les motifs est l'objet *vacuum* et que ces

motifs, malgré le fait qu'ils apparaissent dans les activités *clean* et *vacuum*, ont un poids plus important avec l'activité *vacuum* qu'avec l'activité *clean*. Dit autrement, quand l'algorithme HAROUP exploite des motifs utilisant l'objet *vacuum*, il choisit systématiquement l'activité *vacuum*, car les motifs liés à celle-ci ont un plus grand score. La deuxième chose que l'on peut noter est que les activités *make tea* et *make coffee* sont très bien reconnues et que malgré le fait qu'elles se ressemblent, elles sont rarement confondues. D'autres activités (*clean* et *cook pasta*) sont un peu moins bien reconnues, car elles partagent des motifs en commun ([*water*]). Enfin, on voit que l'activité *noActivity* (qui n'en est pas vraiment une) a une précision faible et un rappel parfait. Un rappel parfait est un résultat attendu étant donné que par défaut tous les événements sont étiquetés comme *noActivity* (voir l'Algorithme 6.1). La faible précision de *noActivity* peut être interprétée comme le fait que parmi les prédictions de reconnaissance effectuées, notre approche considère souvent un événement sans activité comme un événement faisant partie d'une véritable activité. Nous pensons que cela s'explique probablement par deux faits distincts. D'abord, une grande taille de la fenêtre implique de considérer parfois trop d'événements et donc que certains événements se retrouvent étiquetés avec les activités à proximité dans le flux d'événements. Ensuite, il arrive que même lorsque nous ne faisions aucune activité au LIARA, certains objets qui sont restés immobiles ont été considérés comme mobile à cause de l'imprécision des positions fournies par l'algorithme de positionnement implémenté au LIARA.

6.2.4 DISCUSSION ET LIMITES

La section précédente nous montre que nous sommes capables d'obtenir de très bons résultats de reconnaissance grâce à l'algorithme HAROUP, avec les données récoltées au LIARA et les bons paramètres. Ces bonnes performances reposent également sur la qualité de la correspondance entre les données fournies par les capteurs et les motifs d'utilisation d'objets.

Activité prédite ↓	clean	vacuum	make tea	noActivity	cook pasta	make coffee
clean	**351**	0	0	51	0	0
vacuum	0	**1209**	0	0	0	0
make tea	0	0	**1069**	0	0	16
noActivity	0	0	0	**112**	0	0
cook pasta	141	0	0	141	**628**	0
make coffee	0	0	0	111	0	**1317**

Tableau 6.1 : Matrice de confusion de l'algorithme HAROUP avec les paramètres de la meilleure combinaison. *useImageExtractorPatternsOrSPMPatterns=BOTH*, *windowSize* = 20s. © Charles Cousyn, 2022.

Activité	**Précision**	**Rappel**	**F-Score**
clean	0.713	0.873	0.785
vacuum	1.000	1.000	1.000
make tea	1.000	0.985	0.993
noActivity	0.2699	1.000	0.425
cook pasta	1.000	0.690	0.817
make coffee	0.988	0.922	0.954

Tableau 6.2 : Précision, rappel et F-score pour chaque activité avec les paramètres de la meilleure combinaison. *useImageExtractorPatternsOrSPMPatterns=BOTH*, *windowSize* = 20s. © Charles Cousyn, 2022.

Dans notre cas, nous avons établi un dictionnaire faisant cette correspondance (Section 6.1.1) en se basant sur les mots utilisés dans les motifs. Cette dépendance aux mots utilisés fait que le dictionnaire fonctionne particulièrement avec les motifs choisis, mais au détriment de motifs provenant d'autres combinaisons de paramètres. Un exemple concret de cela est illustré avec l'utilisation de la combinaison de paramètre *pnasnet_large duckduckgo 25* pour les motifs provenant des images. Dans ce cas, notre dictionnaire utilise les étiquettes du modèle de classification d'image *pnasnet_large*. À cause de ces étiquettes, si l'on essaye d'exploiter les motifs provenant d'une combinaison utilisant un autre modèle comme *yolov3 duckduckgo 25* avec ce même dictionnaire, alors nous obtenons avec des performances de reconnaissance

proche de 0.0 de justesse avec le paramètre *useImageExtractorPatternsOrSPMPatterns* à *IMAGE_EXTRACTOR*.

Ce dictionnaire fait en réalité état de deux dépendances distinctes. Une dépendance avec les données récoltées et une dépendance avec les mots utilisés dans les motifs d'utilisation d'objets. Notre approche d'effectuer de l'HAR à partir de motifs minés d'internet est donc fonctionnelle, mais difficile à généraliser à d'autres contextes de reconnaissance à cause de ce dictionnaire. De l'information pertinente est bien obtenue de manière non supervisée, mais le lien entre cette information et la réalité de l'intelligence ambiante doit être fait de manière supervisé.

6.3 CONCLUSION

Dans ce chapitre, nous avons analysé une approche d'HAR exploitant les motifs d'utilisation d'objets avec des données provenant d'un contexte réel d'habitat intelligent disponible au LIARA. Nous avons d'abord montré que les objets extraits à partir d'images provenant du web étaient équivalents à des motifs d'utilisation d'objets. Ensuite, nous avons présenté notre approche, son contexte, les données utilisées et l'algorithme utilisé. Les performances de l'approche dépendent de plusieurs paramètres que nous avons définis. Son évaluation a été effectuée en faisant varier ces paramètres qui sont la taille de fenêtre et la source des motifs d'utilisation d'objets. Les performances sont ensuite affichées sous la forme de graphiques afin de faciliter leur interprétation.

Les résultats ont montré que le choix de la source de motifs est extrêmement important, avec notamment une nette supériorité de la combinaison de motifs provenant d'images et de motifs provenant de pages web. La taille de la fenêtre avait également son importance dans certains cas. Avec les meilleures combinaisons de paramètres, il est possible d'obtenir d'ex-

cellentes performances d'HAR pour les 5 activités considérées ; les meilleures performances étant obtenues avec la combinaison de paramètres *useImageExtractorPatternsOrSPMPatterns = BOTH*, *windowSize = 20s* Cependant, il a été montré que la généralisation de l'approche peut être grandement améliorée, car la correspondance entre données de capteurs et motifs est effectuée dans un contexte particulier ; c'est-à-dire avec un habitat intelligent particulier. Cette remarque montre que des travaux futurs pour faciliter cette correspondance motif/capteur doivent être effectués afin d'augmenter les capacités de généralisation des approches d'HAR à partir de fouille de données sur le web.

CHAPITRE VII
CONCLUSION GÉNÉRALE

7.1 RAPPEL DES OBJECTIFS

Cette thèse a pour but d'améliorer les approches de reconnaissance d'activités de la vie quotidienne en utilisant une source qui, jusqu'à aujourd'hui, est restée assez peu exploitée ; à savoir le web. En effet, les approches actuelles décrites dans le Chapitre 2 sont très focalisées sur l'apprentissage supervisé qui peut être très dépendant du contexte dans lequel les données d'apprentissage sont récoltées et dont la récolte peut s'avérer chronophage et complexe. Le web offre la possibilité de données disponibles facilement, rapidement sur un grand nombre d'informations sur beaucoup de sujets. Le Chapitre 3 nous a montré qu'il existe différents types de ressources disponibles sur le web qui sont en lien avec la reconnaissance d'activités ainsi que les manières possibles de les exploiter. En commençant par les images, le Chapitre 4 décrit une approche d'extraction d'objets clés dans les activités de la vie quotidienne à partir d'images provenant du web. En nous penchant ensuite sur le texte et sur la notion d'ordre d'utilisation des objets, le Chapitre 5 présente une méthode pour extraire des motifs séquentiels à partir de pages web décrivant la réalisation d'activités de la vie quotidienne. Enfin, le Chapitre 6 présente la conception d'une approche de reconnaissance d'activités exploitant les informations extraites grâce aux informations extraites avec les approches décrites dans les deux précédents chapitres.

Au final, cette thèse tente d'apporter des réponses aux trois questions de recherche suivantes :

1. Dans quelles mesures le web, en termes d'informations disponibles, permet-il de représenter les activités de la vie quotidienne comparativement aux approches existantes ?

2. Comment parvenir à trouver et exploiter de manière automatisée les informations disponibles sur le web en rapport avec les activités de la vie quotidienne ?
3. Dans quelle mesure une approche d'apprentissage machine utilisant les connaissances du web pourrait valoriser les performances de reconnaissance d'activités humaines ?

7.2 RÉPONSES À LA QUESTION 1 : REPRÉSENTATIVITÉ DES INFORMATIONS DISPONIBLES SUR LE WEB

Dans le Chapitre 3, nous avons identifié les informations en lien avec les activités de la vie quotidienne qui sont disponibles sur le web. Cela nous a amenés à extraire des informations à partir d'images provenant de moteurs de recherche et de textes provenant de pages web.

En ce qui concerne la capacité des images à représenter les activités de la vie quotidienne, nous avons montré que dans le Chapitre 4, malgré qu'il y ait encore des améliorations possibles, nous sommes capables d'extraire efficacement une part non négligeable des objets clés véritablement en lien avec les activités considérées. Cette représentation sous forme de liste d'objets pertinents semble plus facile à comprendre, à généraliser.

Dans le Chapitre 5, nous montrons que les motifs séquentiels extraits à partir du texte des pages web peuvent être jugés très pertinents pour les activités considérées, et ce malgré la difficulté de trouver une mesure de performance non supervisée pour évaluer leur qualité. Les motifs, se traduisant par l'ajout de la notion d'ordre d'utilisation des objets, semblent apporter une expressivité supplémentaire qui permet de mieux représenter la manière dont les objets sont utilisés. Aussi, les bonnes performances de reconnaissance obtenues dans le Chapitre 6 appuient le fait que les motifs d'utilisation d'objets sont une représentation correcte et exploitable des activités de la vie quotidienne.

7.3 RÉPONSES À LA QUESTION 2 : EXPLOITATION AUTOMATISÉE DES INFORMATIONS DISPONIBLES SUR LE WEB

Le second point auquel nous souhaitions répondre dans cette thèse était de savoir comment exploiter de manière automatisée les informations en lien avec les activités de la vie quotidienne disponibles sur le web. Tout d'abord, le Chapitre 4 nous a montré qu'il était possible d'exploiter automatiquement jusqu'à 1 000 images avec des requêtes appropriées sur les deux moteurs de recherche considérés ; à savoir Google Image et DuckDuckGo Image. Il a également montré qu'un modèle de classification/de détection d'objets peut être efficacement utilisé pour automatiser l'extraction d'objets à partir d'images. Cette automatisation était cependant limitée par le fait que les modèles de classification/de détection d'objets ne pouvaient que reconnaître ceux pour lesquels ils ont été entraînés ; et que la proportion d'objets reconnaissables n'excède pas 75% en moyenne.

Ensuite, le Chapitre 5 a montré que l'analyse du texte de pages web pour extraire des motifs d'utilisation d'objets peut se faire automatiquement grâce à des processus classiques du traitement du langage naturel comme l'étiquetage morphosyntaxique, la tokenisation, l'analyse de dépendance syntaxique ou des bases de données comme Wordnet (Princeton University, 2010a) qui contiennent des informations précieuses sur les mots, leur utilisation et leur sens. Nous avons également montré que malgré le fait que les algorithmes de SPM étant principalement en complexité temporelle exponentielle, le processus d'extraction de motifs peut être fait en un temps raisonnable à condition de bien choisir le type de motif extrait, de bien paramétrer son approche et d'utiliser le bon algorithme.

7.4 RÉPONSES À LA QUESTION 3 : LA RECONNAISSANCE D'ACTIVITÉ BASÉE SUR LA FOUILLE DU WEB

Les Chapitres 4 et 5 s'articulent sur l'obtention d'informations pertinentes sur les activités de la vie quotidienne. Ces informations et la facilité de leur obtention apportent beaucoup au domaine de l'HAR dans le sens où, avec une approche adaptée, elles peuvent être exploitées pour concevoir une véritable approche de reconnaissance dans des contextes réels d'habitats intelligents. C'est cette possibilité que nous avons creusé dans le Chapitre 6 où nous concevons notre propre approche basée sur les informations pertinentes extraites grâce aux méthodes présentées dans les Chapitres 4 et 5. C'est donc ce chapitre qui permet de répondre à notre troisième question de recherche ; à savoir est-ce qu'une approche d'HAR basée sur la fouille du web peut être performante ?

Le Chapitre 6 a montré qu'il était possible d'utiliser le web pour concevoir une approche de reconnaissance d'activités efficace et performante. Les expérimentations menées au LIARA (Bouchard *et al.*, 2014) pour 5 activités ont montré que les motifs d'utilisation d'objets couplés à notre algorithme HAROUP (Algorithme 6.1) et un bon paramétrage permettent une excellente reconnaissance de certaines activités. Le grand point fort de l'approche est qu'aucun individu n'a dû être présent pour la phase d'apprentissage permettant ainsi d'ouvrir la voie pour des approches d'HAR se passant des contraintes habituelles d'organisation et de gestion de participants. Utiliser des informations aussi génériques que le texte des pages web ou les images issues de moteurs de recherche ont donc le potentiel de permettre la conception d'approches plus rapide, plus simple et plus généralisable.

7.5 LIMITES ET TRAVAUX FUTURS

Malgré les bonnes performances constatées dans la mise en situation proposée par le Chapitre 6, notre approche d'HAR basée sur la fouille de données sur le web possède plusieurs limites clairement identifiables.

La première est la difficulté d'établir une correspondance claire et précise entre les informations extraites du web et les capteurs réellement utilisés dans l'habitat intelligent considéré. Pour l'instant, il revient à l'expertise humaine de lier ce qui semble aller ensemble. Ainsi, si le web fournit l'information générique qu'un objet est important pour une activité donnée, c'est à l'intégrateur de l'approche d'HAR d'associer cet objet à un capteur précis ou à une donnée issue de capteurs particuliers ; la généralisation de l'approche s'en voit alors diminuée.

Il existe plusieurs solutions pour faciliter l'établissement de cette correspondance. On pourrait d'abord penser à uniformiser le format des données fournies par les capteurs utilisés dans les habitats intelligents, car la principale difficulté de la correspondance réside dans le fait que presque chaque type de capteur fournit des informations spécifiques à son concepteur. Avec un format uniformisé, la correspondance aurait le potentiel de se faire de manière quasi automatique. Une autre solution proposée par Riboni & Murtas (2019) avec ArOnt [58] serait d'établir une ontologie représentant les correspondances entre actions, objets et étiquettes RFID qui pourraient être partagées entre plusieurs habitats intelligents. Les auteurs pensent qu'une telle ontologie, accompagnée d'une interface utilisateur adaptée pourrait permettre d'effectuer facilement les correspondances pour chaque contexte particulier. Nous pensons que cette solution, malgré l'expertise humaine encore nécessaire, est moins pénible et pourrait avoir le potentiel de faciliter les approches d'HAR basées sur le web.

58. https ://sites.unica.it/domusafe/aront/

La seconde limite est le biais pouvant exister dans les données du web. En effet, les images et pages web utilisées pourraient parfaitement mal représenter les objets utilisés lors des activités de la vie quotidienne. Les objets extraits pourraient être biaisés par leur utilisation restreinte dans une zone géographique précise, à une époque particulière, à une culture particulière ou même à un contexte particulier. À cause de ce potentiel biais, la généralisation peut s'avérer moins grande que prévue et les approches d'HAR basées sur le web pourraient ne pas s'appliquer à tous les contextes. Dans cette thèse, nous n'avons pas identifié de biais précis dans nos données, car, comme les pages web et les images utilisées sont les résultats de requêtes effectuées sur des moteurs de recherche dont le fonctionnement exact n'est pas connu, et bien, il n'est pas possible de vérifier l'absence de biais dans les résultats fournis par ces moteurs de recherche. Nous pensons que le gain en transparence du fonctionnement de ces outils, très utilisés par l'humanité, pourrait être bénéfique, car les chercheurs souhaitant s'appuyer sur la puissance de ces derniers pourraient jauger les biais existants et améliorer leurs contributions scientifiques dans le même temps.

7.6 APPORT PERSONNEL

Ce doctorat aura été l'une de mes plus grandes épreuves de ma vie universitaire. Devoir travailler avec soi-même et ses superviseurs pour travailler sur un long projet de recherche pendant plusieurs années est quelque chose de difficile, mais qui m'a beaucoup apporté sur le plan personnel. J'ai continué d'apprendre toujours plus sur l'intelligence artificielle et l'apprentissage machine, j'ai écrit des articles dont un publié (Cousyn *et al.*, 2021) et un soumis et j'ai même pu donner une charge de cours à l'université. L'ensemble de ces expériences m'ont permis de mieux identifier ce qui me plaît : pouvoir transmettre du savoir scientifique à d'autres.

BIBLIOGRAPHIE

Agarwal, B. & Mittal, N. (2014). Text classification using machine learning methods-a survey. Dans *Advances in intelligent systems and computing*, Vol. 236 pp. 701–709. Springer, New Delhi

Aggarwal, J. & Ryoo, M. (2011). Human activity analysis. *ACM Computing Surveys*, *43*(3), 1–43. doi: 10.1145/1922649.1922653

Agrawal, R. & Srikant, R. (1995). Mining sequential patterns. Dans *Proceedings - international conference on data engineering*, Vol. 2007, pp. 3–14. IEEE Comput. Soc. Press. doi: 10.1109/icde.1995.380415

Ann, O. C. & Theng, L. B. (2014). Human activity recognition : A review. Dans *Proceedings - 4th IEEE international conference on control system, computing and engineering, ICCSCE 2014*, pp. 389–393. doi: 10.1109/ICCSCE.2014.7072750

Association, A. S. (1963, juin). *American Standard Code for Information Interchange*. Repéré le 2022-04-03, à https://web.archive.org/web/20191130064706/https://worldpowersystems.com/ARCHIVE/codes/X3.4-1963/index.html

Aswini, V. & Lavanya, S. (2014, avril). Pattern discovery for text mining. Dans *2014 international conference on computation of power, energy, information and communication (ICCPEIC)*, pp. 412–416. IEEE. doi: 10.1109/ICCPEIC.2014.6915399

Bach, N. & Badaskar, S. (2007). A review of relation extraction. *Literature review for Language and Statistics II*, *2*, 1–15.

Berners-Lee, T. J. & Cailliau, R. (1990). *WorldWideWeb : Proposal for a HyperText project*. Repéré le 2020-02-15, à https://www.w3.org/Proposal.html

Bondy, A. & Murty, M. R. (2008). *Graph theory*. London: Springer-Verlag.

Bouchard, K., Bouchard, B. & Bouzouane, A. (2011). A new qualitative spatial recognition model based on Egenhofer topological approach using C4.5 algorithm : experiment and results. *Procedia Computer Science*, *5*, 497–504. doi: 10.1016/j.procs.2011.07.064

Bouchard, K., Bouchard, B. & Bouzouane, A. (2011). Qualitative spatial activity recognition using a complete platform based on passive RFID tags : Experimentations and results. Dans *Toward useful services for elderly and people with disabilities* pp. 308–312. Springer, Berlin, Heidelberg

Bouchard, K., Bouchard, B. & Bouzouane, A. (2014). Practical guidelines to build smart homes : Lessons learned. *Opportunistic networking, smart home, smart city, smart systems*, *1*(January 2015), 1–38.

Bouchard, K., Fortin-Simard, D., Gaboury, S., Bouchard, B. & Bouzouane, A. (2013, 6). Accurate RFID Trilateration to Learn and Recognize Spatial Activities in Smart Environment. *International Journal of Distributed Sensor Networks*, p. 936816. doi: 10.1155/2013/936816

Bouchard, K., Maitre, J., Bertuglia, C. & Gaboury, S. (2020). Activity recognition in smart homes using UWB radars. *Procedia Computer Science*, *170*, 10–17. doi: 10.1016/j.procs.2020.03.004

Breiman, L. (1996, 2). Bagging predictors. *Machine Learning*, pp. 123–140. doi: 10.1007/BF00058655

Bux, A., Angelov, P. & Habib, Z. (2017). Vision based human activity recognition : A review. Dans *Advances in intelligent systems and computing*, Vol. 513 pp. 341–371. Springer, Cham

CASAS (2019). *CASAS datasets*. Repéré le 2020-08-20, à http://casas.wsu.edu/datasets

Chapron, K., Bouchard, K., Duchesne, E. & Gaboury, S. (2017, août). Transportable and scalable system for activities and exercises recognition in real-time. Dans *2017 IEEE SmartWorld, ubiquitous intelligence & computing, advanced & trusted computed, scalable computing & communications, cloud & big data computing, internet of people and smart city innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, pp. 1–7. IEEE. doi: 10.1109/UIC-ATC.2017.8397512

Chapron, K., Lapointe, P., Bouchard, K. & Gaboury, S. (2020). Highly accurate bathroom activity recognition using infrared proximity sensors. *IEEE Journal of Biomedical and Health Informatics*, *24*(8), 2368–2377. doi: 10.1109/JBHI.2019.2963388

Chapron, K., Plantevin, V., Thullier, F., Bouchard, K., Duchesne, E. & Gaboury, S. (2018, 1). A more efficient transportable and scalable system for real-time activities and exercises recognition. *Sensors*, p. 268. doi: 10.3390/s18010268

Charniak, E. (1997). Statistical Techniques for Natural Language Parsing. *AI Mag.*, *18*(4), 33–44. doi: 10.1609/aimag.v18i4.1320

Chen, L., Hoey, J., Nugent, C. D., Cook, D. J. & Yu, Z. (2012, 6). Sensor-based activity recognition. *IEEE Transactions on Systems, Man and Cybernetics Part C : Applications and Reviews*, pp. 790–808. doi: 10.1109/TSMCC.2012.2198883

Chicco, D. (2017). Ten quick tips for machine learning in computational biology. *BioData Mining*, p. 35. doi: 10.1186/s13040-017-0155-3

Chicco, D. & Jurman, G. (2020, 1). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, p. 6. doi: 10.1186/s12864-019-6413-7

Chikhaoui, B., Wang, S. & Pigot, H. (2011, mars). A frequent pattern mining approach for ADLs recognition in smart environments. Dans *2011 IEEE international conference on advanced information networking and applications*, pp. 248–255. IEEE. doi: 10.1109/AINA.2011.13

Cohen, J. (2013). *Statistical power analysis for the behavioral sciences*. Academic press.

Cousyn, C. (2018). *Exploration d'articles scientifiques sur les maladies rares pour l'extraction d'informations*. (Mémoire de maîtrise).

Cousyn, C. (2021, novembre). *experimentionResults.json*. Repéré le 2022-04-03, à https://github.com/CharlesCousyn/webpages_objects_spm_extractor/blob/f616016f099b81e280a34a1847e8e62c40d61f48/patternUse/experimentationResults.json

Cousyn, C., Bouchard, K. & Gaboury, S. (2021). Web-based objects detection to discover key objects in human activities. *Journal of Ambient Intelligence and Humanized Computing*. doi: 10.1007/s12652-021-03433-0

Daugman, J. G. (1985, 7). Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America A*, p. 1160. doi: 10.1364/JOSAA.2.001160

de Boer, V., van Someren, M. & Lupascu, T. (2011). Web page classification using image analysis features. Dans J. Filipe & J. Cordeiro (Éds.). *Web information systems and technologies*, pp. 272–285., Berlin, Heidelberg. Springer Berlin Heidelberg. doi: 10.1007/978-3-642-22810-0_20

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. & Li Fei-Fei (2010). ImageNet : A large-scale hierarchical image database. Dans *Japanese Journal of Nephrology*, pp. 248–255. doi: 10.1109/cvpr.2009.5206848

Desjardins, J. (2005). L'analyse de régression logistique. *Tutorial in quantitative methods for psychology*, *1*(1), 35–41. doi: 10.20982/tqmp.01.1.p035

Dietterich, T. G. (2000). Ensemble methods in machine learning. Dans *Multiple classifier systems* pp. 1–15. Berlin, Heidelberg: Springer Berlin Heidelberg

DomuSafe (2019). *aront*. Repéré le 2020-01-18, à https://sites.unica.it/domusafe/aront

Doucet, A. & Ahonen-Myka, H. (2004). Non-contiguous word sequences for information retrieval. Dans *Proceedings of the workshop on multiword expressions integrating processing - MWE '04*, Vol. 26, pp. 88–95., Morristown, NJ, USA. Association for Computational Linguistics. doi: 10.3115/1613186.1613198

Ducatel, K., Bogdanowicz, M., Scapolo, F., Leijten, J. & Burgelman, J.-C. (2003). *Ambient intelligence : From vision to reality*. IST Advisory Group Draft Report, European Commission.

DuckDuckGo (2021). *DuckDuckGo traffic*. Repéré le 2021-01-01, à https://duckduckgo.com/traffic

E. Munguia Tapia, Choudhury, T., Philipose, M. & Wyatt, D. (2005). Using automatically mined object relationships and common sense for unsupervised activity recognition. Dans *Proceedings of the 20th national conference on Artificial intelligence*, Vol. 20, pp. 21–27.

Repéré le 2021-06-12, à http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.61.9138

ECMA-404 (2013). The JSON data interchange format. *ECMA International*, *1st Editio*(October), 8. doi: 10.17487/rfc7158

Elleuch, M., Ismaili, O. A., Laga, N., Gaaloul, W. & Benatallah, B. (2020). Discovering activities from emails based on pattern discovery approach BT - business process management forum. Dans D. Fahland, C. Ghidini, J. Becker, & M. Dumas (Éds.). *Business process management forum*, pp. 88–104., Cham. Springer International Publishing.

Etimberg, Benmccann, Tannerlinsley & Simonbrunel (2020). *Chart.js*. Repéré le 2021-01-01, à https://github.com/chartjs/Chart.js

Fan, L., Wang, Z. & Wang, H. (2013, décembre). Human activity recognition model based on decision tree. Dans *2013 international conference on advanced cloud and big data*, pp. 64–68. IEEE. doi: 10.1109/CBD.2013.19

Faulkner, A., Lang, A., Makowski, A. & Airportyh (2020). *jsdom*. Repéré le 2020-06-15, à https://github.com/jsdom/jsdom

Fishkin, R. (2019). *Less than Half of Google Searches Now Result in a Click*. Repéré le 2019-11-15, à https://sparktoro.com/blog/less-than-half-of-google-searches-now-result-in-a-click

Fournier-Viger, P., Wu, C. W., Gomariz, A. & Tseng, V. S. (2014). VMSP : Efficient vertical mining of maximal sequential patterns. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *8436 LNAI*, 83–94. doi: 10.1007/978-3-319-06483-3_8

Gambhir, M. & Gupta, V. (2017, 1). Recent automatic text summarization techniques : a survey. *Artificial Intelligence Review*, pp. 1–66. doi: 10.1007/s10462-016-9475-9

Ganesh, P. (2019). Object detection : Simplified - towards data science. *Medium*. Repéré le 2020-05-12, à https://towardsdatascience.com/object-detection-simplified-e07aa3830954

Gomariz, A., Campos, M., Marin, R. & Goethals, B. (2013). ClaSP : An efficient algorithm for mining frequent closed sequences. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *7818 LNAI*(PART 1), 50–61. doi: 10.1007/978-3-642-37453-1_5

Goodfellow, I., Bengio, Y. & Courville, A. (2016). Introduction. Dans *Deep learning* p. 1. MIT Press

Goodman, S. N. (1999, 12). Toward evidence-based medical statistics. 2 : The bayes factor. *Annals of Internal Medicine*, p. 1005. doi: 10.7326/0003-4819-130-12-199906150-00019

Google (2019). *Google maps platform*. Repéré le 2020-04-05, à https://developers.google.com/maps/documentation

Gorodkin, J. (2004, 5). Comparing two K-category assignments by a K-category correlation coefficient. *Computational Biology and Chemistry*, pp. 367–374. doi: 10.1016/j.compbiolchem.2004.09.006

Gu, T., Chen, S., Tao, X. & Lu, J. (2010). An unsupervised approach to activity recognition and segmentation based on object-use fingerprints. *Data and Knowledge Engineering*, *69*(6), 533–544. doi: 10.1016/j.datak.2010.01.004

Guilford, J. P. (1936). *Psychometric methods*. New York ; London: McGraw-Hill Book Co. Repéré le 2022-03-16, à http://catalog.hathitrust.org/api/volumes/oclc/16739119.html

Hallett, D. (2005). Integration. Dans *Calculus* p. 340. Hoboken, N.J: J. Wiley.

Harris, Z. S. (1954, 2-3). Distributional structure. *Word-journal of The International Linguistic Association*, pp. 146–162. doi: 10.1080/00437956.1954.11659520

Hui, J. (2019). *Real-time object detection with YOLO, YOLOv2 and now YOLOv3*. Repéré le 2019-01-15, à https://medium.com/@jonathan{_}hui/real-time-object-detection-with-yolo-yolov2-28b1b93e2088

Hussain, Z., Sheng, M. & Zhang, W. E. (2019). Different approaches for human activity

recognition : A survey. *arXiv preprint arXiv :1906.05074*. Repéré le 2020-03-20, à http://arxiv.org/abs/1906.05074

Ihianle, I. K., Naeem, U., Tawil, A.-R. & Azam, M. A. (2016). Recognizing activities of daily living from patterns and extraction of web knowledge. Dans *Proceedings of the 2016 ACM international joint conference on pervasive and ubiquitous computing adjunct - UbiComp '16*, pp. 1255–1262. ACM. doi: 10.1145/2968219.2968440

Järvelin, K. & Kekäläinen, J. (2017). IR evaluation methods for retrieving highly relevant documents. Dans *ACM SIGIR Forum*, Vol. 51, pp. 243–250. ACM New York, NY, USA.

JASP Team (2020). *JASP (version 0.14.1)[Computer software]*. Repéré le 2021-04-03, à https://jasp-stats.org/

Karani, D. (2018). Topic modelling with PLSA - towards data science. *Medium*. Repéré le 2019-02-20, à https://towardsdatascience.com/topic-modelling-with-plsa-728b92043f41

Kasteren, T. V. & Krose, B. (2007). Bayesian activity recognition in residence for elders. Dans *IET conference publications*, Vol. 2007, pp. 209–212. IEE. doi: 10.1049/cp:20070370

Katz, S., Ford, A. B., Moskowitz, R. W., Jackson, B. A. & Jaffe, M. W. (1963). Studies of illness in the aged : the index of ADL : A standardized measure of biological and psychosocial function. *JAMA : the journal of the American Medical Association*, *185*(12), 914–919. doi: 10.1001/jama.1963.03060120024016

Kautz, H. & Allen, J. (1986). Generalized plan recognition. Dans *In proceedings of the 5th national conference on artificial intelligence*, Vol. 86, pp. 32–37. Repéré le 2019-02-20, à http://www.aaai.org/Papers/AAAI/1986/AAAI86-006.pdf

Kim, E., Helal, S. & Cook, D. (2010, 1). Human activity recognition and pattern discovery. *IEEE Pervasive Computing*, pp. 48–53. doi: 10.1109/MPRV.2010.7

Kim, Y. J., Kang, B. N. & Kim, D. (2016, octobre). Hidden markov model ensemble for activity recognition using tri-axis accelerometer. Dans *Proceedings - 2015 IEEE international conference on systems, man, and cybernetics, SMC 2015*, pp. 3036–3041. IEEE. doi: 10.1109/SMC.2015.528

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. Dans *Proceedings of the 14th international joint conference on artificial intelligence - volume 2*, pp. 1137–1143. Morgan Kaufmann Publishers Inc. doi: 10.5555/1643031.1643047

Kolekar, M. H. & Dash, D. P. (2017, novembre). Hidden Markov Model based human activity recognition using shape and optical flow based features. Dans *IEEE region 10 annual international conference, Proceedings/TENCON*, pp. 393–397. IEEE. doi: 10.1109/TENCON.2016.7848028

Koshorek, O., Cohen, A., Mor, N., Rotman, M. & Berant, J. (2018). Text segmentation as a supervised learning task. *arXiv preprint arXiv :1803.09337*. Repéré le 2020-05-24, à http://arxiv.org/abs/1803.09337

Kvålseth, T. O. (1989, 1). Note on cohen's kappa. *Psychological Reports*, pp. 223–226. doi: 10.2466/pr0.1989.65.1.223

Lara, s. D. & Labrador, M. A. (2012, janvier). A mobile platform for real-time human activity recognition. Dans *2012 IEEE consumer communications and networking conference (CCNC)*, pp. 667–671. IEEE. doi: 10.1109/CCNC.2012.6181018

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, *1*(4), 541–551. doi: 10.1162/neco.1989.1.4.541

Liu, C., Zoph, B., Neumann, M., Shlens, J., Hua, W., Li, L.-J., Fei-Fei, L., Yuille, A., Huang, J. & Murphy, K. (2017). *Progressive neural architecture search*.

Liu, Y., Nie, L., Liu, L. & Rosenblum, D. S. (2016). From action to activity : Sensor-based activity recognition. *Neurocomputing*, *181*, 108–115. doi: 10.1016/j.neucom.2015.08.096

Maitre, J., Glon, G., Gaboury, S., Bouchard, B. & Bouzouane, A. (2015). Efficient appliances recognition in smart homes based on active and reactive power, fast fourier transform and decision trees. *AAAI Workshop - Technical Report*, *WS-15-03*(April 2016), 24–29.

Maitre, J., Rendu, C., Bouchard, K., Bouchard, B. & Gaboury, S. (2019). Basic daily activity recognition with a data glove. *Procedia Computer Science*, *151*, 108–115. doi: 10.1016/j.procs.2019.04.018

Malhotra, R. & Sharma, A. (2017, 2). Quantitative evaluation of web metrics for automatic genre classification of web pages. *International Journal of System Assurance Engineering and Management*, pp. 1567–1579. doi: 10.1007/s13198-017-0629-1

Manning, C. (2008). Bag of words model. Dans *Introduction to information retrieval* chapitre 6, p. 117. New York: Cambridge University Press.

Manning, C. D., Raghavan, P. & Schutze, H. (2008). Chap 8 : Evaluation in information retrieval. Dans *Introduction to information retrieval* pp. 151–175. Cambridge: Cambridge University Press

Matthews, B. W. (1975, 2). Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta (BBA) - Protein Structure*, pp. 442–451. doi: 10.1016/0005-2795(75)90109-9

Mayhaymate (2012). *PageRank* [Image]. Repéré le 2022-03-17, à https://commons.wikimedia.org/wiki/File:PageRank-hi-res.png

Maylawati, D. S., Aulawi, H. & Ramdhani, M. A. (2018). The concept of sequential pattern mining for text. *IOP Conference Series : Materials Science and Engineering*, *434*(1). doi: 10.1088/1757-899X/434/1/012042

Maylawati, D. S. & Saptawati, G. A. P. (2017). Set of frequent word item sets as feature representation for text with indonesian slang. *Journal of Physics : Conference Series*, p. 12066. doi: 10.1088/1742-6596/801/1/012066

MDN contributors (2021). *Introduction to the DOM*. Repéré le 2021-06-21, à https://developer.mozilla.org/en-US/docs/Web/API/Document{_}Object{_}Model/Introduction

MDN contributors (2022, février). *WebSockets - Référence Web API*. Repéré le 2022-03-24, à https://developer.mozilla.org/fr/docs/Web/API/WebSockets_API

Miah, M. B., Akter, S. & Bonik, C. (2014). Automatic bangladeshi vehicle number plate recognition system using neural network. *American International Journal of Research in Science, Technology, Engineering & Mathematics*, *9(1)*, pp. 62–66.

Mitchell, T. M. (1997). *Machine learning*. New York, NY, USA: McGraw-Hill.

Mitra, M., Buckley, C., Singhal, A., Cardie, C. & Others (1997). An analysis of statistical and syntactic phrases. Dans *RIAO*, Vol. 97, pp. 200–214.

Mozilla Developer Network (2020). *JavaScript*. Repéré le 2020-04-12, à https://developer.mozilla.org/fr/docs/Web/JavaScript

Muenteroman (2015). *Odd-eyed cat histogram*. Repéré le 2019-05-23, à https://commons.wikimedia.org/wiki/File:Odd-eyed{_}cat{_}histogram.png

Müller, T., Cotterell, R., Fraser, A. & Schütze, H. (2015). Joint lemmatization and morphological tagging with lemming. Dans *Proceedings of the 2015 conference on empirical methods in natural language processing*, pp. 2268–2274., Lisbon, Portugal. Association for Computational Linguistics. doi: 10.18653/v1/D15-1272

Ng, A. (2020). *Non-max suppression - object detection*. Repéré le 2020-07-26, à https://www.coursera.org/lecture/convolutional-neural-networks/non-max-suppression-dvrjH

Ning Zhong, Yuefeng Li & Sheng-Tang Wu (2012, 1). Effective pattern discovery for text mining. *IEEE Transactions on Knowledge and Data Engineering*, pp. 30–44. doi: 10.1109/TKDE.2010.211

Ningrum, P. K., Pansombut, T. & Ueranantasun, A. (2020, 6). Text mining of online job advertisements to identify direct discrimination during job hunting process : A case study in Indonesia. *PLOS ONE*, p. e0233746. doi: 10.1371/journal.pone.0233746

Opitz, J. & Burst, S. (2021). Macro F1 and Macro F1. *arXiv :1911.03347 [cs, stat]*. Repéré le 2022-03-24, à http://arxiv.org/abs/1911.03347

Page, L., Brin, S., Motwani, R. & Winograd, T. (1998). *The PageRank citation ranking :*

Bringing order to the web.

Palmes, P., Pung, H. K., Gu, T., Xue, W. & Chen, S. (2010). Object relevance weight pattern mining for activity recognition and segmentation. *Pervasive and Mobile Computing*, *6*(1), 43–57. doi: 10.1016/j.pmcj.2009.10.004

Pei, J., Han, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U. & Hsu, M. C. (2001). PrefixSpan : Mining sequential patterns efficiently by prefix-projected pattern growth. *Proceedings - International Conference on Data Engineering*, pp. 215–224. doi: 10.1109/icde.2001.914830

Pentney, W., Pentney, W., Popescu, a., Popescu, a., Wang, S., Wang, S., Kautz, H., Kautz, H., Philipose, M. & Philipose, M. (2006). Sensor-based understanding of daily life via large-scale use of common sense. *Proceedings of the National Conference on Artificial Intelligence*, *21*, 906. Repéré le 2019-01-28, à http://www.aaai.org/Papers/AAAI/2006/AAAI06-143.pdf

Perez, M. (2019). What is web scraping and what is it used for ? *ParseHub Blog*. Repéré le 2020-03-26, à https://www.parsehub.com/blog/what-is-web-scraping

Perkowitz, M., Philipose, M., Fishkin, K. & Patterson, D. J. (2004). Mining models of human activities from the web. Dans *Proceedings of the 13th conference on World Wide Web - WWW '04*, p. 573. ACM. doi: 10.1145/988672.988750

Pkdoorn (2015). *Markup language - science europe data glossary*. Repéré le 2020-04-12, à http://sedataglossary.shoutwiki.com/wiki/Markup{_}language

Princeton University (2010). *About WordNet.* Repéré le 2021-02-02, à https://wordnet.princeton.edu/

Princeton University (2010). *WordNet Search - 3.1*. Repéré le 2022-04-03, à http://wordnetweb.princeton.edu/perl/webwn

Princeton University (2010). *WordNet Search - 3.1 : place*. Repéré le 2022-04-03, à http://wordnetweb.princeton.edu/perl/webwn?o2=1&o0=1&o8=1&o1=1&o7=1&o5=1&o9=&o6=1&o3=1&o4=1&s=place&i=16&h=00000000000000000000000000000000#c

Princeton University (2022). *lexnames(5WN) | WordNet*. Repéré le 2022-04-03, à https://wordnet.princeton.edu/documentation/lexnames5wn

Quinlan, J. R. (2014). *C4.5 : programs for machine learning*. Elsevier.

Rabiner, L. & Juang, B.-H. (1986). An introduction to hidden Markov models. *IEEE ASSP Magazine*, *3*(1), 4–16. doi: 10.1109/MASSP.1986.1165342

Redmon, J. & Farhadi, A. (2017, juillet). YOLO9000 : Better, faster, stronger. Dans *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 6517–6525. IEEE. doi: 10.1109/CVPR.2017.690

Redmon, J. & Farhadi, A. (2018). YOLOv3 : An incremental improvement. *CoRR*. Repéré le 2019-08-20, à http://arxiv.org/abs/1804.02767

Reiss, A., Hendeby, G. & Stricker, D. (2013). A competitive approach for human activity recognition on smartphones. Dans *ESANN 2013 proceedings, 21st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pp. 455–460., Bruges.

Ren, S., He, K., Girshick, R. & Sun, J. (2017). Faster r-cnn : Towards real-time object detection with region proposal networks. Dans C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, & R. Garnett (Éds.), *IEEE transactions on pattern analysis and machine intelligence*, Vol. 39 pp. 1137–1149. Curran Associates, Inc.

Riboni, D. & Murtas, M. (2017). Web mining & computer vision : New partners for object-based activity recognition. Dans *Proceedings - 2017 IEEE 26th international conference on enabling technologies : Infrastructure for collaborative enterprises, WETICE 2017*, pp. 158–163. doi: 10.1109/WETICE.2017.38

Riboni, D. & Murtas, M. (2019). Sensor-based activity recognition : One picture is worth a thousand words. *Future Generation Computer Systems*. doi: 10.1016/j.future.2019.07.020

Richardson, L. (2020). *Beautiful soup*. Repéré le 2020-04-05, à https://www.crummy.com/software/BeautifulSoup/bs4/doc

Richardson, M. & Domingos, P. (2006). Markov logic networks. *Machine learning*, *62*(1-2), 107–136. doi: 10.1007/s10994-006-5833-1

Robertson, S. (2004). Understanding inverse document frequency : On theoretical arguments for IDF. *Journal of Documentation*, *60*(5), 503–520. doi: 10.1108/00220410410560582

Ronao, C. A. & Cho, S.-B. (2016). Human activity recognition with smartphone sensors using deep learning neural networks. *Expert Systems with Applications*, pp. 235–244. doi: 10.1016/j.eswa.2016.04.032

Samuel, A. L. (1959, 3). Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, pp. 210–229. doi: 10.1147/rd.33.0210

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A. & Chen, L. C. (2018). MobileNetV2 : Inverted residuals and linear bottlenecks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 4510–4520. doi: 10.1109/CVPR.2018.00474

Sarle, W. S., Jain, A. K. & Dubes, R. C. (1990, 2). Algorithms for clustering data. *Technometrics : a journal of statistics for the physical, chemical, and engineering sciences*, p. 227. doi: 10.2307/1268876

Sawilowsky, S. S. (2009, 2). New effect size rules of thumb. *Journal of Modern Applied Statistical Methods*, pp. 597–599. doi: 10.22237/jmasm/1257035100

Scherer, R. (2020). *Computer vision methods for fast image classification and retrieval*, Vol. 821 de *Studies in computational intelligence*. Cham: Springer International Publishing. doi: 10.1007/978-3-030-12195-2

Schmidt, C. F., Sridharan, N. S. & Goodson, J. L. (1978). The plan recognition problem : An intersection of psychology and artificial intelligence. *Artificial Intelligence*, *11*(1-2), 45–83. doi: 10.1016/0004-3702(78)90012-7

Shaalan, K. (2007, 1). A survey of named entity recognition and classification. *Lingvisticae Investigationes*, pp. 3–26. doi: 10.1075/li.30.1.03nad

Subasi, A., Dammas, D. H., Alghamdi, R. D., Makawi, R. A., Albiety, E. A., Brahimi, T. & Sarirete, A. (2018). Sensor based human activity recognition using adaboost ensemble classifier. *Procedia Computer Science*, *140*, 104–111. doi: 10.1016/j.procs.2018.10.298

Sukthankar, G., Geib, C., Bui, H. H., Pynadath, D. & Goldman, R. P. (2014). An introduction to plan, activity, and intent recognition. Dans G. Sukthankar, C. Geib, H. H. Bui, D. Pynadath, & R. P. Goldman (Éds.), *Plan, activity, and intent recognition* chapitre An Introdu. Elsevier

Szegedy, C., Ioffe, S., Vanhoucke, V. & Alemi, A. A. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. Dans *Thirty-first AAAI conference on artificial intelligence*. doi: 10.5555/3298023.3298188

Szeliski, R. (2010). *Computer vision : algorithms and applications*. Springer Science & Business Media.

Tamura, H., Mori, S. & Yamawaki, T. (1978). Textural features corresponding to visual perception. *IEEE Transactions on Systems, man, and cybernetics*, *8*(6), 460–473. doi: 10.1109/TSMC.1978.4309999

Tapia, E. M., Choudhury, T. & Philipose, M. (2006). Building reliable activity models using hierarchical shrinkage and mined ontology. Dans K. P. Fishkin, B. Schiele, P. Nixon, & A. Quigley (Éds.). *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*, Vol. 3968 LNCS, pp. 17–32., Berlin, Heidelberg. Springer Berlin Heidelberg. doi: 10.1007/11748625_2

Theissen, A. (1997). Relation d'hypo/hyperonymie et choix lexical. Dans *Le choix du nom en discours (Langue et cultures) (French Edition)* pp. 61–64. Libr. Droz, (paperback)

Tomasi, C. & Kanade, T. (1992, 2). Shape and motion from image streams under orthography : a factorization method. *International Journal of Computer Vision*, pp. 137–154. doi: 10.1007/BF00129684

Tsang, S.-H. (2019, mars). *Review : YOLOv2 & YOLO9000 you only look once (object detection)*. Repéré le 2019-04-01, à https://towardsdatascience.com/review-yolov2-yolo9000-you-only-look-once-object-detection-7883d2b02a65

United Nations, Department of Economic and Social Affairs, P. D. (2017). *World population ageing 2017*. United Nations, Department of Economic and Social Affairs.

University of Waterloo (2008). *Markov logic networks*. Repéré le 2022-01-20, à https://cs.uwaterloo.ca/{~}ppoupart/teaching/cs486-fall08/slides/Lecture21.pdf

Vadapalli, P. (2021, mars). *Dependency parsing in NLP [Explained with examples]*. Repéré le 2021-06-21, à https://www.upgrad.com/blog/dependency-parsing-in-nlp

Vala, K. (2019). Markov networks : Undirected graphical models - towards data science. *Medium*. Repéré le 2020-01-20, à https://towardsdatascience.com/markov-networks-undirected-graphical-models-dfb19effd8cb

van Doorn, J., Ly, A., Marsman, M. & Wagenmakers, E.-J. (2018, 4). Bayesian inference for kendall's rank correlation coefficient. *The American Statistician*, pp. 303–308. doi: 10.1080/00031305.2016.1264998

van Doorn, J., Ly, A., Marsman, M. & Wagenmakers, E.-J. (2020). Bayesian rank-based hypothesis testing for the rank sum test, the signed rank test, and Spearman's ρ. *Journal of Applied Statistics*, *47*(16), 2984–3006. doi: 10.1080/02664763.2019.1709053

Venkata, N., Padmasree, L. & Mangathayaru, N. (2016). Survey of text mining techniques, challenges and their applications. *International Journal of Computer Applications*, *146*(11), 30–35. doi: 10.5120/ijca2016910908

W3C (2006). *Extensible markup language (XML) 1.1 (second edition)*. Repéré le 2020-04-12, à https://www.w3.org/TR/xml11

W3C.org (2018). *Selectors level 3*. Repéré le 2021-06-21, à https://www.w3.org/TR/selectors-3/

W3C.org (2021). *DOM*. Repéré le 2021-06-21, à https://dom.spec.whatwg.org/{#}what

W3.org (2008). *HTML5 : A vocabulary and associated APIs for HTML and XHTML*. Repéré le 2020-08-20, à https://dev.w3.org/html5/html-author

Wang, J., Chen, Y., Hao, S., Peng, X. & Hu, L. (2019). Deep learning for sensor-based activity recognition : A survey. *Pattern Recognition Letters*, *119*, 3–11. doi: 10.1016/j.patrec.2018.02.010

Wang, Z. (2018). Sliding window technique - zengrui wang - medium. *Medium*. Repéré le 2022-02-20, à https://medium.com/@zengruiwang/sliding-window-technique-360d840d5740

Wen, J., Zhong, M. & Wang, Z. (2015). Activity recognition with weighted frequent patterns mining in smart environments. *Expert Systems with Applications*, *42*(17-18), 6423–6432. doi: 10.1016/j.eswa.2015.04.020

Wikipedia (2019). *Bag-of-words model - Wikipedia*. Repéré le 2019-06-03, à https://en.wikipedia.org/wiki/Bag-of-words{_}model

Wikipedia (2020). *Motion estimation*. Repéré le 2020-04-12, à https://en.wikipedia.org/wiki/Motion{_}estimation

Wimo, P. A., Guerchet, D. M., Ali, M. G.-C., Wu, D. Y.-T., Prina, D. M. & Alzheimer's Disease International (2015). *World alzheimer report 2015*. Alzheimer's Disease International.

Wyatt, D., Philipose, M. & Choudhury, T. (2005). Unsupervised activity recognition using automatically mined common sense. Dans *Proceedings of the 20th national conference on Artificial intelligence*, Vol. 20, pp. 21–27. doi: 10.1093/bmb/ldx041

Xu, D. & Tian, Y. (2015, 2). A comprehensive survey of clustering algorithms. *Annals of Data Science*, pp. 165–193. doi: 10.1007/s40745-015-0040-1

Yu, N. (2018, août). A visualized pattern discovery model for text mining based on TF-IDF weight method. Dans *2018 10th international conference on intelligent human-machine systems and cybernetics (IHMSC)*, pp. 183–186. IEEE. doi: 10.1109/IHMSC.2018.10148

Zhang, M. & Sawchuk, A. A. (2012, janvier). Motion primitive-based human activity recognition using a bag-of-features approach. Dans *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*, IHI '12, pp. 631–640., New York, NY, USA.

Association for Computing Machinery. doi: 10.1145/2110363.2110433

Zheng, V. W., Hu, D. H. & Yang, Q. (2009). Cross-domain activity recognition. Dans *Proceedings of the 11th international conference on ubiquitous computing*, UbiComp '09, pp. 61–70., New York, NY, USA. ACM. doi: 10.1145/1620545.1620554

Zhou, Z.-H. (2012). *Ensemble methods : foundations and algorithms*. CRC press.

Zhu, Q. (2020). On the performance of Matthews correlation coefficient (MCC) for imbalanced dataset. *Pattern Recognition Letters*, pp. 71–80. doi: 10.1016/j.patrec.2020.03.030

Zhu, X. & Wu, X. (2004, 3). Class noise vs. Attribute noise : A quantitative study. *Artificial Intelligence Review*, pp. 177–210. doi: 10.1007/s10462-004-0751-8