Analyse explicable et personnalisable de données hétérogènes multi-niveaux : une approche guidée par l'apprentissage automatique et les ontologies

Maxime Perrot

THÈSE

pour l'obtention du Grade de

Docteur de l'École Nationale Supérieure de Mécanique et d'Aérotechnique

(Diplôme National - Arrêté du 25 mai 2016)

École Doctorale : MIMME - Mathématiques, Informatique, Matériaux, Mécanique, Energétique
Secteur de Recherche : Informatique et Applications

Présentée par :

Maxime PERROT

Analyse explicable et personnalisable de données hétérogènes multi-niveaux : une approche guidée par l'apprentissage automatique et les ontologies

---

Sous la direction de **Mickael BARON**, **Brice CHARDIN** et **Stéphane JEAN**

Soutenue le 13 décembre 2024 devant la Commission d'Examen

JURY

Président :

Arnaud SOULET	Professeur des Universités, Université de Tours, Blois

Rapporteurs :

Nadine CULLOT	Professeure des Universités, Université de Bourgogne, Dijon
Marie-Jeanne LESOT	Professeure des Universités, Sorbonne Université, Paris

Examinateurs :

Mickael BARON	Ingénieur de Recherche, ISAE-ENSMA, Poitiers
Brice CHARDIN	Maître de Conférences, ISAE-ENSMA, Poitiers
Stéphane JEAN	Professeur des Universités, Université de Poitiers, Poitiers
Hala SKAF-MOLLI	Professeure des Universités, Université de Nantes, Nantes

Invités :

Jérôme CREIGNOU	Responsable sécurité des SI, Orisha Retail Shops, La Roche-Sur-Yon
Allel HADJALI	Professeur des Universités, ISAE-ENSMA, Poitiers
Vincent TRICOIRE	Ingénieur des données, Orisha, Paris

Remerciements

Je tiens tout d'abord à exprimer ma gratitude à mon directeur de thèse, Stéphane Jean, pour m'avoir donné l'opportunité de réaliser ce projet. Durant ces trois années, il a su m'encadrer avec rigueur et bienveillance, apportant son soutien aussi bien dans les moments d'enthousiasme que dans les périodes plus complexes. Ses qualités humaines et scientifiques ont été essentielles à l'accomplissement de ce travail et à mon développement personnel et professionnel.

Je remercie également Mickael Baron et Brice Chardin, collègues du LIAS, pour leur accompagnement précieux. Mickael, pour sa passion pour les aspects opérationnels de l'informatique, sa rigueur rédactionnelle, sa disponibilité, et pour avoir joué un rôle central dans la réussite de cette collaboration entre le laboratoire et l'entreprise. Brice, par sa maîtrise des sujets liés à l'apprentissage automatique et sa rigueur scientifique, a été un atout majeur pour ce projet, renforçant mon approche méthodologique et ma formation de chercheur.

Ce travail n'aurait pas pu être mené à bien sans le soutien d'Orisha Retail Shops. Je tiens à remercier Vincent Tricoire pour ses conseils avisés dans le domaine des sciences des données. Son expertise et sa pédagogie m'ont permis de développer des compétences solides tout au long de ce projet. Merci également à Jérôme Creignou pour ses connaissances techniques et son enthousiasme, ainsi qu'à Vincent Coulet, dont l'implication et l'intérêt pour le projet ont été des moteurs essentiels, malgré ses nombreuses responsabilités.

Je souhaite aussi remercier mes collègues et amis du LIAS et d'Orisha Retail Shops. Que ce soit par vos conseils ou simplement les moments partagés, vous avez enrichi ces trois années.

Je tiens à remercier ma compagne Lisa, mes parents, mes frères, ainsi que ma famille et mes amis pour leur soutien indéfectible tout au long de cette aventure.

Un remerciement tout particulier à Graziello Geneau, dont les conseils et l'orientation vers l'informatique ont marqué un tournant décisif dans mon parcours, ainsi qu'à Annie Geniet, qui m'a transmis sa passion pour cette discipline. Je vous en suis profondément reconnaissant.

Résumé

Cette thèse s'intègre dans le contexte industriel de Orisha Retail Shops, qui propose des caisses enregistreuses et logiciels pour points de vente comme les boulangeries et bureaux de tabac. Ces systèmes génèrent un volume considérable de données sur les ventes, cruciales pour améliorer le suivi opérationnel et la prospection, conférant un avantage concurrentiel significatif à l'entreprise.

Le premier problème traité est l'identification précise des activités des points de vente, complexifiée par la liberté de nommage et de catégorisation des produits. Ceci constitue un défi de classification, peu abordé dans la littérature malgré sa pertinence en apprentissage automatique, du fait de la faible qualité intrinsèque des données et des libellés de produits à haute cardinalité. Pour adresser cela, un banc d'essai spécialisé a été conçu pour évaluer les méthodes de classification, mettant en lumière les limites des techniques actuelles, notamment dans l'encodage des données.

La seconde contribution, *Thesaurus-BT*, propose une méthode de classification basée sur un thésaurus construit à partir de connaissances métier, permettant une classification globale et effective des produits. Cette méthode, testée expérimentalement, surpasse les encodeurs traditionnels et a été mise en production chez Orisha Retail Shops.

Enfin, la troisième contribution répond aux besoins d'analyse de l'entreprise par la mise en œuvre de la méthode Thesaurus-BT, couplée aux technologies du Web sémantique pour modéliser les concepts et relations non explicitement présents dans les données existantes. Cette approche, complétée par des évaluations empiriques de différentes architectures, démontre le potentiel et les limites de ces technologies dans un usage industriel, offrant une vue d'ensemble sur leur applicabilité et scalabilité dans des scénarios réels.

Mots clés : Apprentissage automatique, Codage, Classification automatique, Intégration de données (informatique), Ontologies (informatique), Web sémantique Banc d'essai, Thesausus-Based Transformation

Abstract

This thesis is set within the industrial context of Orisha Retail Shops, which offers cash registers and software solutions for retail outlets such as bakeries and tobacco shops. These systems generate a significant volume of sales data, crucial for enhancing operational tracking and prospecting, thus providing a substantial competitive advantage to the company.

The first issue addressed is the precise identification of retail activities, complicated by the freedom of product naming and categorization. This poses a classification challenge that is rarely discussed in the literature despite its relevance to machine learning, due to the poor intrinsic quality of the data and the high cardinality of product labels. To address this, a specialized benchmark was designed to evaluate classification methods, highlighting the limitations of current techniques, particularly in data encoding.

The second contribution, *Thesaurus-BT*, introduces a classification method based on a thesaurus constructed from industry knowledge, enabling a comprehensive and effective product classification. This method, experimentally tested, surpasses traditional encoders and has been implemented at Orisha Retail Shops.

Finally, the third contribution meets company's analysis needs by implementing the Thesaurus-BT method, paired with semantic web technologies to model concepts and relationships not explicitly present in the existing data. This approach, complemented by empirical evaluations of various architectures, demonstrates the potential and limitations of these technologies in industrial use, providing an overview of their applicability and scalability in real scenarios.

Keywords : Machine learning, Coding theory, Automatic classification, Data integration (Computer science), Ontologies (Information retrieval), Sematic Web, Benchmark, Thesausus-Based Transformation

Table des matières

Liste des figures

Liste des tableaux

Introduction générale

Dans le milieu industriel, la collecte et l'analyse de données jouent un rôle primordial. De nombreuses entreprises, quelle que soit leur taille ou leur secteur, s'efforcent d'exploiter les données issues de leurs processus opérationnels pour optimiser leurs activités, améliorer leur efficacité et, plus généralement, renforcer leur compétitivité. L'essor des données massives (Big Data) et des technologies d'intelligence artificielle (IA) a ouvert la voie à de nouvelles opportunités pour extraire des connaissances significatives à partir des données transactionnelles.

Orisha Retail Shops [1], entreprise industrielle dans laquelle s'est déroulée cette thèse CIFRE, est spécialisée dans la fourniture de solutions d'encaissement pour les points de vente, tels que les bureaux de tabac et les boulangeries. Le cœur des activités d'Orisha Retail Shops repose sur ses caisses enregistreuses et son logiciel d'encaissement, permettant aux commerçants d'effectuer des transactions. Cependant, l'ambition de l'entreprise va au-delà de la simple gestion des encaissements. Grâce à ses solutions, Orisha Retail Shops collecte chaque jour un volume considérable de données relatives aux ventes de produits. Ces données transactionnelles offrent un aperçu de l'activité commerciale de ses clients. Ainsi, en exploitant les données issues de ces transactions, Orisha Retail Shops voit l'opportunité d'améliorer ses services.

Les données issues des transactions de vente constituent donc une ressource stratégique essentielle pour renforcer les services et l'assistance qu'Orisha Retail Shops fournit à ses clients. Exploitées efficacement, elles pourraient permettre d'améliorer différentes activités telles que :

— Le suivi opérationnel : les commerçants et les gestionnaires d'Orisha Retail Shops pourraient surveiller en temps réel les ventes, identifier les tendances et adapter leurs stratégies d'approvisionnement.

— La prospection et le marketing ciblé : en analysant les habitudes d'achat des clients, Orisha Retail Shops pourrait segmenter sa clientèle et cibler ses actions marketing pour répondre de manière personnalisée aux besoins de chaque groupe.

— L'optimisation de l'activité publicitaire : Orisha Retail Shops fournit aux annonceurs des données sur l'activité des points de vente, et pourrait renforcer cette offre en leur proposant des analyses croisées. Celles-ci combineraient les informations relatives aux performances de leurs produits avec les données démographiques, sociales et géographiques des points de vente, permettant ainsi aux annonceurs de mieux cibler et adapter leurs campagnes.

1. https://retail-shops.orisha.com/

Cependant, exploiter pleinement les données transactionnelles d'Orisha Retail Shops pose des défis considérables. L'un des principaux obstacles réside dans la qualité et la structure des données, qui sont souvent fragmentées, incohérentes ou mal standardisées. Ces limitations compliquent la mise en place d'analyses fiables et de prévisions précises. De plus, les différences dans les formats de données, ainsi que les éventuelles lacunes dans les informations collectées, compliquent les processus d'intégration et nécessitent des étapes supplémentaires de nettoyage et de transformation pour garantir une interprétation et une exploitation fiables. C'est dans ce contexte que les travaux de thèse présentés dans ce manuscrit ont été réalisés. Nous détaillons ci-après les problématiques abordées par ces travaux ainsi que les objectifs fixés.

Problématiques et objectifs

L'une des problématiques principales posées par les données collectées par Orisha Retail Shops est la classification des produits vendus dans les points de vente. Contrairement à des environnements contrôlés où les catégories de produits sont bien définies, dans les petits commerces tels que les bureaux de tabac ou les boulangeries, les commerçants sont libres de nommer et de catégoriser leurs produits sans suivre de normes spécifiques. Cela conduit à une grande hétérogénéité des libellés de produits. Par exemple, un même produit peut être désigné par plusieurs noms différents selon le point de vente, avec des variations syntaxiques ou sémantiques significatives.

Ce problème de variabilité et d'ambiguïté dans les noms de produits se traduit par un besoin de classification complexe. D'un point de vue scientifique, il s'agit d'un problème de classification dans le cadre de l'apprentissage automatique. Toutefois, contrairement aux ensembles de données habituellement utilisés dans la littérature, les données d'Orisha Retail Shops présentent des caractéristiques particulières qui compliquent leur traitement :

— *Faible qualité des données* : les libellés des produits et des familles de produits sont définis librement et sans cohérence entre les différents points de vente.

— *Synonymie et variations morphologiques* : un même produit peut être désigné de plusieurs manières, avec des différences mineures ou majeures dans les termes utilisés.

— *Cardinalité élevée* : les ensembles de produits et de familles de produits sont très larges, impliquant un grand nombre de catégories différentes.

Des approches de classification en apprentissage automatique ont été développées ces dernières décennies et appliquées dans divers domaines [Jordan, 2015]. Elles sont souvent évaluées sur des bancs d'essai destinés à démontrer leur efficacité [Olson, 2017]. Cependant, la plupart de ces bancs d'essai reposent sur des données synthétiques qui ne reflètent pas la grande variabilité des libellés de produits, leur haute cardinalité, ainsi que les problèmes de synonymie et de variations morphologiques mentionnés précédemment. Par conséquent, ces bancs d'essai ne permettent pas d'évaluer l'adéquation des approches de classification en apprentissage automatique actuelles à la problématique de la classification des produits dans le contexte d'Orisha Retail Shops.

La première problématique abordée dans cette thèse consiste à établir un banc d'essai basé sur des données réelles rencontrées dans l'industrie, permettant ainsi d'évaluer les différentes approches de classification de produits dans le contexte spécifique des points de vente d'Orisha Retail Shops.

Les approches de classification en apprentissage automatique reposent sur des techniques d'encodage standard, telles que TF-IDF [Sparck Jones, 1972] ou les techniques d'incorporation de type Word2Vec [Mikolov, 2013]. Cependant, comme nous le montrerons dans nos travaux grâce au banc d'essai proposé, ces techniques ne parviennent pas toujours à capturer les nuances spécifiques des libellés de produits. Cela peut conduire à une mauvaise catégorisation des produits et, par conséquent, à des analyses erronées.

La deuxième problématique abordée dans cette thèse consiste à élaborer une nouvelle méthode d'encodage qui soit robuste face à des données non structurées et de faible qualité.

Au-delà de la classification des produits, Orisha Retail Shops souhaite tirer parti des données transactionnelles pour répondre à des analyses plus complexes, telles que la segmentation des clients ou l'identification de nouvelles opportunités commerciales. Toutefois, ces demandes engendrent de nouveaux défis, car elles nécessitent l'intégration de sources de données hétérogènes, qu'elles soient internes ou externes à l'entreprise, ainsi que l'utilisation de concepts qui ne sont pas explicités dans les sources de données disponibles.

Dans ce contexte, les technologies du Web sémantique présentent un intérêt potentiel. Ces technologies reposent sur divers langages (par exemple, RDFS [Dan, 2004] ou OWL [McGuinness, 2004]), des systèmes de gestion de bases de données appelés triplestore (par exemple, Jena [McBride, 2002] ou Virtuoso [Erling, 2009]) et des plateformes (par exemple, Ontop [Calvanese, 2017]). Bien que ces technologies aient été largement étudiées d'un point de vue théorique, leur application dans des environnements industriels, comme celui d'Orisha Retail Shops, reste peu explorée [Hogan, 2020]. En particulier, bien que différents bancs d'essai aient été conçus pour évaluer les performances des triplestores (par exemple, LUBM [Guo, 2005b] ou WatDiv [Aluç, 2014b]), peu de travaux se sont intéressés à une évaluation globale des performances des différentes architectures possibles pour mettre en œuvre ces technologies.

La troisième problématique abordée dans cette thèse consiste à proposer une méthodologie pour la mise en œuvre des technologies du Web sémantique dans un contexte industriel, tout en évaluant les performances des différentes architectures d'implémentation possibles.

Démarche et contributions

La première contribution de cette thèse est la création d'un banc d'essai destiné à évaluer les différentes approches de classification sur des données réelles rencontrées dans l'industrie. Ce banc d'essai vise à tester des méthodes de classification des produits dans le contexte spécifique des points de vente d'Orisha Retail Shops. Cependant, son objectif est

également d'être suffisamment détaillé pour pouvoir être utilisé dans d'autres contextes industriels confrontés aux mêmes problématiques, notamment la variabilité des libellés de produits, leur haute cardinalité, ainsi que les questions de synonymie et de variations morphologiques.

Pour concevoir ce banc d'essai, nous avons adopté une démarche méthodique en commençant par une définition générique des problématiques à résoudre. En collaboration avec des experts métiers, nous avons identifié précisément les enjeux liés à la classification des commerces par activités et à la complexité des libellés de produits. Nous avons sélectionné un échantillon représentatif des données réelles de l'entreprise, tout en maintenant une taille permettant une annotation manuelle fiable des activités d'un sous-ensemble de commerces par des experts, garantissant ainsi la validation des résultats. Nous avons également défini des métriques d'évaluation appropriées et établi des procédures expérimentales standardisées. Cette approche nous a permis de structurer nos expérimentations, de garantir la reproductibilité des résultats et de comparer efficacement les différentes méthodes, afin de choisir la solution la plus adaptée en termes de qualité, de coût et de délai.

La deuxième contribution de cette thèse est l'élaboration d'une nouvelle méthode d'encodage de données, nommée Thesaurus-BT. Cette méthode repose sur un thésaurus construit à partir des connaissances métier d'Orisha Retail Shops et permet une classification approximative, mais globalement efficace, des produits. Contrairement aux méthodes d'encodage classiques, Thesaurus-BT intègre les relations sémantiques entre les produits et les catégories, en s'appuyant sur des structures hiérarchiques spécifiques au domaine de la distribution.

L'efficacité de Thesaurus-BT est démontrée expérimentalement en comparaison avec les encodeurs traditionnels, et la méthode est intégrée dans les systèmes d'Orisha Retail Shops. Bien que cette méthode soit fondée sur un thésaurus spécifique aux connaissances métiers d'Orisha Retail Shops, elle est conçue pour être générique et adaptable à d'autres contextes industriels rencontrant les mêmes problématiques de fiabilité des données.

La troisième contribution de cette thèse concerne l'application des technologies du Web sémantique dans le cadre de l'analyse avancée des données d'Orisha Retail Shops. Nous proposons une démarche complète pour modéliser les concepts et les relations présents dans les demandes d'analyse, les relier aux sources de données réelles et les exploiter via des requêtes sémantiques. De plus, nous démontrons que plusieurs architectures peuvent être mises en œuvre pour ces technologies. Nous réalisons donc une évaluation expérimentale sur des données réelles afin de comparer ces architectures, mettant en lumière à la fois les forces et les limites de ces technologies dans un cadre opérationnel, notamment en termes de scalabilité et de capacité à intégrer des sources de données hétérogènes.

Organisation du manuscrit

Ce manuscrit de thèse est organisé en cinq chapitres.

— Le premier chapitre présente le contexte industriel et scientifique de la thèse, en détaillant les données sur lesquelles repose ce travail, leur utilisation actuelle chez Orisha Retail Shops, ainsi que les nouveaux besoins et objectifs identifiés.

— Le deuxième chapitre offre une revue de l'état de l'art portant sur les thématiques clés de cette thèse : l'encodage, l'incorporation (embedding), l'étiquetage multiple, et les bancs d'essai pertinents pour évaluer les approches de classification des activités des commerces.

— Le troisième chapitre s'intéresse à la conception d'un banc d'essai personnalisé, conçu pour répondre aux défis de la classification des produits du catalogue des clients de l'entreprise. Il met en lumière les enjeux et les bénéfices d'une telle démarche.

— Le quatrième chapitre propose une nouvelle méthodologie d'encodage des produits, fondée sur un thésaurus, qui permet une amélioration significative des performances pour la classification des produits et des clients de Bimedia.

— Enfin, le cinquième et dernier chapitre aborde la structuration des données et des connaissances à travers l'utilisation des ontologies et des graphes de connaissances, ouvrant ainsi la voie à de nouveaux usages, à la fois pour les données existantes et pour celles produites par les nouveaux traitements mis en place.

Publications

Maxime Perrot, Mickaël Baron, Brice Chardin et Stéphane Jean *Thesaurus-based Transformation : A Classification Method for Real Dirty Data.* European Conference on Advances in Databases and Information Systems (ADBIS), Sep 2023, Barcelona, Spain

Maxime Perrot, Mickaël Baron, Brice Chardin et Stéphane Jean *Knowledge Graphs for Data Integration in Retail.* 27th International symposium on methodologies for intelligent systems (ISMIS), Jun 2024, Poitiers - Futuroscope, France

Chapitre 1

L'exploitation hors ligne des données chez Orisha Retail Shops

Objectifs

L'objectif de ce chapitre est de présenter comment Orisha Retail Shops, une entreprise pionnière dans les solutions pour commerces de proximité, notamment avec sa solution Bimedia, a évolué pour devenir une plateforme complexe, centrée sur l'exploitation des données. À travers l'analyse de son histoire, de son architecture technique et de ses canaux de création de données, ce chapitre met en lumière les forces de l'entreprise dans la gestion des données, ainsi que les défis auxquels elle fait face, notamment en matière de classification des produits et des commerces, et d'intégration des sources de données externes. Le chapitre conclut en identifiant quatre familles de nouveaux besoins, qui seront abordées dans le reste du manuscrit, et qui visent à optimiser les processus décisionnels et à maximiser la valeur des données pour Orisha Retail Shops.

Sommaire

1 Introduction

Orisha Retail Shops [1], anciennement nommée Bimedia, est une entreprise qui donne la priorité à l'innovation et à l'amélioration continue des services proposés à ses clients et collaborateurs. Elle a notamment été la première en France à proposer une caisse enregistreuse composée de deux écrans LCD couleur tactiles : l'un du côté du commerçant, et l'autre du côté du client, d'où le nom de sa solution « Bimedia » [2]. Cette innovation présente plusieurs avantages. D'une part, l'aspect pratique et ergonomique de l'écran tactile a considérablement amélioré l'expérience utilisateur. D'autre part, la possibilité de diffuser des informations utiles au client de l'autre côté du comptoir s'est développée au fil du temps : d'abord le panier d'achat, puis des informations telles que la météo ou les actualités locales, jusqu'à aujourd'hui, avec des publicités ciblées, locales ou nationales. Enfin, cette solution offre une apparence plus moderne, rompant avec l'image vieillissante des caisses enregistreuses à boutons physiques.

Désormais, Orisha Retail Shops ne se distingue plus seulement par une simple caisse enregistreuse, mais par une solution de plus en plus complexe, composée de multiples terminaux interconnectés, d'une suite logicielle complète offrant de nombreux services tiers avec un catalogue de partenaires s'étoffant au fil du temps, et même d'applications mobiles à disposition des clients finaux [3]. Le traitement des données est devenu un enjeu central pour l'entreprise, que ce soit à des fins de commercialisation des données, de rapport des performances aux différents acteurs externes à l'entreprise (commerçants, distributeurs, annonceurs), mais aussi, et surtout, pour diverses analyses internes accompagnant la prise de décision, la détection d'anomalies, et bien plus encore. Dans ce cadre, Orisha Retail Shops a su mettre en place des flux de gestion, de stockage et de traitement des données automatisés et complexes, gérant des volumes de données importants à de grandes cadences, tout en garantissant une disponibilité 24 heures sur 24 et 7 jours sur 7.

Dans cette dynamique de valorisation des données à travers tous les services de l'entreprise, de nouveaux besoins ont émergé. Le projet de thèse s'inscrit dans cette démarche pour répondre à une partie de ces nouveaux besoins.

Dans ce chapitre, nous commencerons par une présentation d'Orisha Retail Shops et de son évolution (section 2). Ensuite, nous aborderons les bases de données internes utilisées par l'entreprise (section 3), ainsi que l'architecture et les flux de données associés (section 4). Enfin, nous explorerons les nouveaux besoins identifiés pour améliorer la valorisation des données (section 5).

1. https://retail-shops.orisha.com/
2. https://retail-shops.orisha.com/logiciels/bimedia/
3. Dans ce manuscrit, le terme « client » désigne généralement les clients d'Orisha, c'est-à-dire les commerces. Les « clients finaux » sont les personnes qui fréquentent les commerces équipés de la solution.

2 Orisha Retail Shops

Orisha Retail Shops est une filiale du groupe Orisha [4]. Orsiha compte plus de 1300 collaborateurs et fait partie des plus grands fournisseurs de logiciels européens. Le groupe est présent dans un peu plus de 50 pays et fournit des solutions pour un large panel de clients (Carrefour, Decathlon, Adidas, Vinci, Honda, etc.). Depuis sa création, le groupe connaît une croissance constante, avec un chiffre d'affaire passant de 75 millions d'euros en 2021 à 200 millions d'euros en 2023. Le groupe opère dans cinq secteurs différents, constituant ainsi cinq unités métier (*business unit*) : la santé, l'immobilier, la construction, l'agroalimentaire, et enfin le commerce et la distribution.

Cette dernière unité métier correspond à Orisha Retail Shops, qui équipe plus de 17000 commerces en France avec trois solutions logicielles, totalisant plus de 1,2 milliard de transactions par an. L'une de ces solutions, nommée Bimedia, est composée d'une suite de matériels et de logiciels spécialisés pour les commerces de proximité et équipe plus de 6500 commerces en France. Cette solution, qui existe depuis 22 ans, est développée et animée par un peu plus de 140 collaborateurs. Ses clients sont principalement des bureaux de tabac, des maisons de la presse, des boulangeries, des épiceries, des librairies, des restaurants, des hôtels, des fleuristes, des bars, des cafés ou des combinaisons de ces activités.

La solution Bimedia est globalement composée de :

— une ou plusieurs caisses enregistreuses avec deux écrans, dont celui orienté vers le commerçant est tactile, visible en figue 1.1 ;

— un périphérique de paiement ;

— un ou plusieurs périphériques additionnels portables permettant l'encaissement, le paiement et la gestion, la « ScanUp », présentée en figure 1.2 ;

— une application mobile de pilotage à destination des commerçants, permettant notamment le suivi des ventes, visible en figure 1.3 ;

— une suite logicielle modulaire pour répondre à des besoins spécifiques tels que : l'encaissement, le paiement, la facturation, la gestion des stocks, la billeterie, le transfert d'argent, la gestion des commandes et des tables pour les restaurants, etc.

Aujourd'hui, l'entreprise se démarque de ses concurrents principalement par le segment premium de sa solution, par le très grand nombre de services additionnels – dont les principaux sont représentés sur la figure 1.4 – directement intégrés au logiciel de la caisse et aux périphériques portables, et par l'activité de sa régie publicitaire. En effet, les caisses enregistreuses proposées dans la solution Bimedia étant composées de deux écrans, dont l'un est tourné vers le consommateur, l'entreprise a su développer une activité de diffusion d'annonces publicitaires afin de capitaliser sur cette visibilité. Une dizaine de personnes travaillent autour de cette activité au sein de l'entreprise. Leur mission est de sourcer puis rester en relation avec les annonceurs, négocier les contrats en valorisant au mieux les espaces à disposition, et répondre aux demandes

4. https://www.orisha.com/

FIGURE 1.1 – Caisse Bimedia

FIGURE 1.2 – ScanUP Bimedia

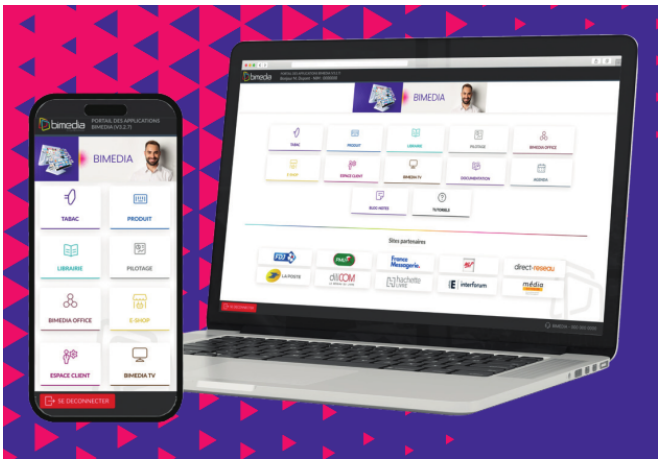


FIGURE 1.3 – Solution pilotage

FIGURE 1.4 – Services tiers

d'analyses spécifiques ou d'exportation de données à l'initiative des annonceurs ou des entreprises partenaires. Autour de cette activité, des traitements de données relativement complexes peuvent être demandés aux analystes, afin d'apporter des arguments lors de la négociation de diffusion d'annonces, ou encore pour répondre à des besoins spécifiques. Les travaux atour de ce projet de thèse sont notamment liés à ces nouveaux besoins analytiques.

Finalement, la filiale d'Orisha qui commercialise le produit Bimedia dispose de quatre grands axes de revenus : les abonnements pour la solution d'encaissement, les frais sur les services additionnels, la valorisation des espaces publicitaires dans les commerces, et la revente de données, par exemple les statistiques des ventes autour du tabac à des fournisseurs de cigarettes. Nous allons, dans les sections suivantes, résumer le fonctionnement de l'architecture technique derrière la solution Bimedia.

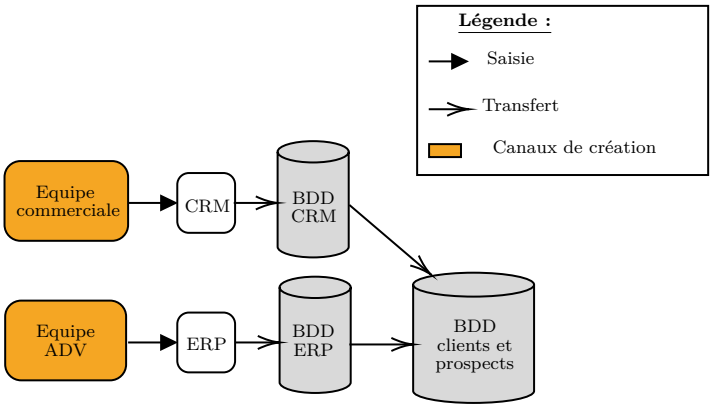


FIGURE 1.5 – Architecture de création et stockage des données légales

3 Bases de données support

Cette section présente les différentes sources internes de données utilisées, en précisant comment elles sont exploitées et quelles sont leurs limites actuelles pour Orisha Retail Shops.

3.1 Données légales sur les commerces

Les données légales sur les commerces sont saisies au moment de l'inscription d'un commerce dans le CRM (*Customer Relationship Management*, logiciel de gestion de la relation client) et l'ERP (*Enterprise Resource Planning*, logiciel de gestion des processus opérationnels) de l'entreprise en tant que prospect. Dans les bases de données de ces applications, nous retrouvons diverses informations sur les commerces, principalement des données légales. La portée de ces informations est assez conséquente, car Orisha Retail Shops répertorie plus de 60 000 commerces en France (incluant des prospects, des commerces clients, et d'anciens clients).

Les informations du CRM sont ensuite vérifiées, complétées et corrigées lors de la transformation du prospect en client dans l'ERP. Nous y retrouvons diverses informations, principalement sur l'entreprise avec laquelle Orisha Retail Shops signe un contrat de leasing du matériel et du logiciel. Nous y trouvons notamment la raison sociale de l'entreprise, l'enseigne, le SIRET (Système d'Identification du Répertoire des ÉTablissements), le code NAF (Nomenclature d'Activités Française), le numéro de TVA, l'adresse, ainsi que les contrats, les différentes communications avec le client, les factures, les documents administratifs, et tout l'historique des échanges internes à l'entreprise au sujet du client.

Toutes ces données sont enfin consolidées par des traitements manuels réalisés par le service d'administration des ventes (ADV), puis importées dans une base de données non relationnelle, contenant donc les données légales des clients et des prospects, ce qui permet à différents applicatifs et utilisateurs d'y accéder plus facilement. Cette organisation est illustrée dans la figure 1.5.

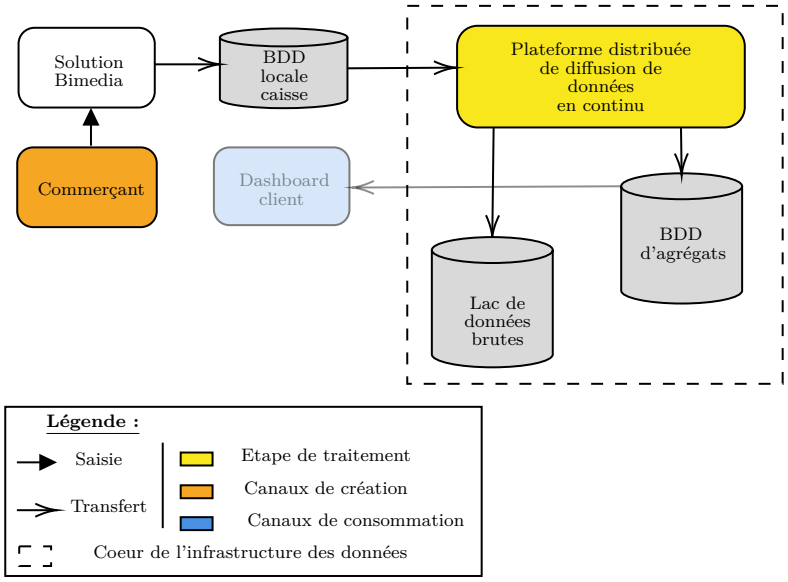


FIGURE 1.6 – Architecture de création et stockage des données des ventes

3.2 Données des ventes

Les clients ayant un contrat actif avec Orisha Retail Shops produisent et transmettent quotidiennement un grand nombre de données, notamment par la remontée des ventes mais aussi des enregistrements (*logs*) sur l'usage des fonctions du matériel ou des incidents. Ces données sont stockées dans une base de données locale au point de vente, et sont transférées à l'entreprise via une plateforme de streaming d'événements distribuée. Cette plateforme permet notamment la gestion de volumes de données colossaux grâce à un système de queue, nous parlons ici de flux pouvant atteindre une centaine de tickets par seconde. Deux flux de données sortent de ce système distribué, l'un alimentant un lac de données contenant l'historique des tickets de vente sous forme brute, et l'autre alimentant une base de données nommée « base d'agrégats », car les ventes y sont agrégées par jour et par produit de manière incrémentale. Un schéma de ce système de gestion des données est présenté en figure 1.6.

De nombreux autres canaux de création de données existent au sein de l'entreprise, mais ils ne sont pas abordés dans ce chapitre car ils ne sont pas pertinents pour la compréhension générale du sujet.

3.3 Traitements récurrents et automatiques

Ces données servent de support à de nombreuses applications, et font notamment l'objet de traitements périodiques, comme les exports quotidiens, hebdomadaires ou mensuels de données spécifiques destinées à des collaborateurs pour des vérifications ou des études, ainsi que celles destinées à des partenaires dans le cadre d'accords. Ces exports peuvent par exemple être des listes de ventes agrégées par semaine liées à une marque de cigarette ou de boisson. Un autre type d'analyse automatisée, réalisée en tâche de fond, est le rapport de la détection d'anomalies effectués pour le service de conformité de l'entreprise, afin de détecter de potentielles tentatives de transferts d'argent frauduleux par exemple. Les clients de la solution Bimedia disposent

également de tableaux de bord en temps réel affichant diverses statistiques sur les ventes, les marges, les stocks et la fréquentation. Enfin, une partie des analyses récurrentes concerne la consolidation automatique de données, produisant un entrepôt de données issues de diverses sources internes ou externes, permettant par exemple le géocodage d'adresses.

3.4 Analyses ponctuelles

Le principal demandeur d'analyses ponctuelles est la régie publicitaire d'Orisha Retail Shops, pour valoriser les espaces publicitaires. Les typologies d'analyses sont très variées : il peut s'agir de simples exports de données pour un commerce sur une période donnée ; ou bien d'analyses plus complexes, comme pour identifier tous les points de vente à moins de 500 mètres d'une station de métro de l'agglomération parisienne, enregistrés comme vendeurs de boissons par leur code APE/NAF dans les référentiels de l'État français.

Les fonctions liées au commerce et à la prospection de nouveaux clients sont également des services qui formulent régulièrement des demandes d'analyses spécifiques afin d'affiner le ciblage géographique pour la prospection. Par exemple, une demande peut s'intéresser uniquement aux commerces ayant une activité de restauration à emporter afin de leur faire découvrir un nouveau service de gestion des commandes et des livraisons.

3.5 Limites

Dans le cadre des analyses, qu'elles soient récurrentes ou ponctuelles, nous pouvons relever au moins cinq principales limites auxquelles l'entreprise fait face.

1. **L'absence de référentiel à jour sur la caractérisation des activités des commerces.** En effet, l'activité commerciale est une information difficile à identifier dans le contexte industriel de l'entreprise. Cette information est saisie manuellement par les commerciaux lors de l'ajout en tant que prospect dans le système d'information. Elle est parfois vérifiée et corrigée lors de la transformation du prospect en client. Cependant, ce référentiel n'est pas pleinement exploitable pour plusieurs raisons. D'une part, l'activité des commerces évolue avec le temps, et la liste des choix d'activités proposée aux commerciaux dans le logiciel a également évolué. Par exemple, jusqu'à récemment, la catégorie « restaurant » n'était pas présente dans ces formulaires. Cette information peut également être obtenue via le référentiel des entreprises tenu par l'État, qui, pour chaque SIRET, inclut un code NAF/APE correspondant à l'activité principale exercée. Cependant, cette information est parfois erronée et possède une granularité qui n'est pas toujours pertinente pour Orisha. Une autre méthode consiste à utiliser des sources externes, comme Google Maps, mais ce processus n'est pas automatisé et n'est utilisé que pour confirmation. L'information d'origine dans nos référentiels n'est que rarement corrigée par les utilisateurs, même si elle s'avère être incorrecte. Enfin, la caractérisation des commerces par l'analyse automatique des ventes pourrait être une autre solution. Bien que cette solution ne soit pas triviale, c'est en grande partie sur cette approche que nous avons travaillé, comme nous le verrons dans ce manuscrit.

2. **La classification des produits limitée aux familles fixes.** Pour les produits à distribution fortement contrôlée par l'État français, tels que le tabac, la presse, les jeux d'argent, ou les services dématérialisés comme la billetterie, les commerçants disposent de catalogues préconfigurés par l'entreprise, organisés en familles de produits. Ces familles de produits sont dites « fixes » car elles ne peuvent pas être modifiées par le commerçant. Orisha Retail Shops a une maîtrise totale de ces familles de produits et peut identifier à quelle typologie de produits appartient une vente dès lors qu'elle concerne l'une de ces familles. Tant qu'une étude se réfère à des produits répertoriés dans les familles fixes, il n'y a pas de réelle difficulté. Cependant, de plus en plus d'études portent sur des produits appartenant à des familles non-fixes, tels que des boissons, des confiseries, ou encore des ventes liées à la restauration, révélant ainsi une limite sur la catégorisation partiellement connue des produits et des services. L'entreprise a besoin d'un système pour organiser les produits afin de faciliter ce type d'étude. Se limiter aux familles fixes n'est plus suffisant ; l'absence totale de catégorisation fiable pour tout un pan des ventes n'est plus envisageable.
3. **La non-intégration automatique de sources de données externes utilisées à répétition.** Bien que certaines données de sources externes, comme le géocodage des adresses, soient dupliquées et intégrées aux bases de données internes, la plupart ne le sont pas, ce qui nécessite un travail de chargement, de traitement et d'alignement répétitif. Avec la multiplication du nombre d'études ponctuelles et récurrentes qui font appel à des sources de données externes, telles que les référentiels des entreprises, les catalogues de centres de formation d'études supérieures ou encore les recensements de population, l'entreprise souhaite trouver des solutions pérennes et réutilisables pour intégrer ces données.
4. **La complexité des outils.** Le personnel non informaticien de l'entreprise dispose d'outils développés pour simplifier la définition et l'exécution de requêtes. Cependant, un certain nombre de sources d'informations ne sont pas intégrées à ces outils, rendant leur récupération difficile voire impossible. Les analystes ont fréquemment recours à un expert pour réaliser une partie du traitement, alors que ces derniers n'ont qu'une faible valeur ajoutée sur ce type de tâche.
5. **L'impossibilité de valoriser de manière automatique et en temps réel les espaces publicitaires.** Cette impossibilité est principalement due aux trois premières limites citées plus tôt : les commerces sont mal, voir pas catégorisés, tant sur leurs activités que sur leur environnement socio-géo-démographique ; et le type des produits associés à la transaction en cours de traitement en caisse est régulièrement inconnu. Cette mauvaise connaissance du contexte d'une transaction limite les possibilités de réaliser des annonces ciblées, et diminue ainsi la valeur de l'offre.

Dans ce manuscrit, nous allons chercher à résoudre les quatre premières limites. La cinquième n'est résolue qu'indirectement, par la résolution des quatre autres.

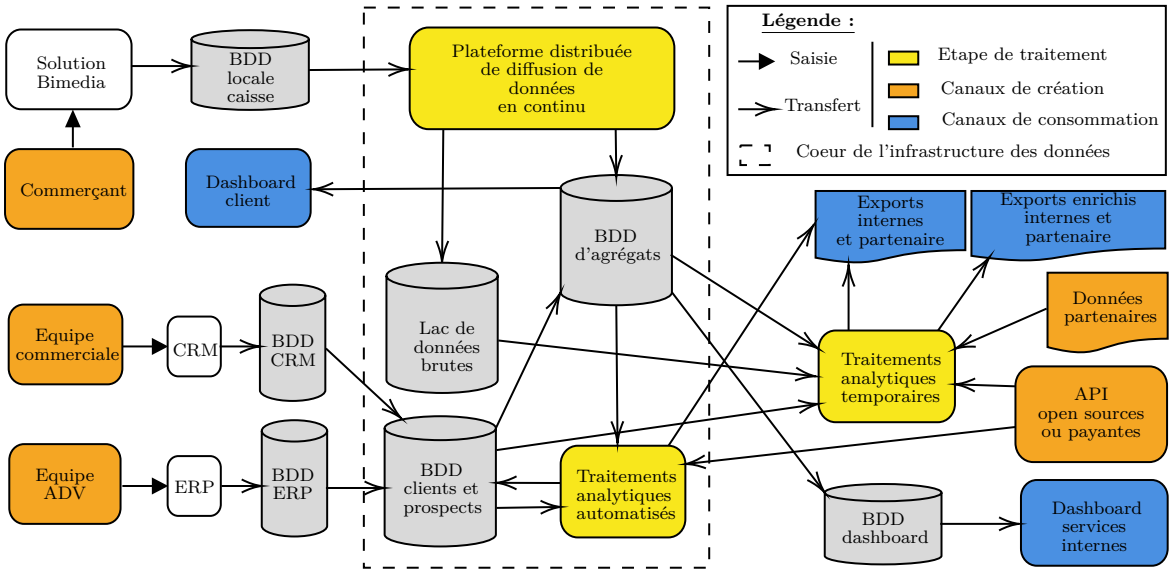


FIGURE 1.7 – Flux de données

4 Architecture

Afin de comprendre plus concrètement les flux de données concernés par ce manuscrit, nous avons choisi de les représenter de manière schématique dans la figure 1.7. Nous retrouvons par ailleurs dans cette figure les éléments de deux figures présentées plus tôt dans ce chapitre, les figures 1.5 et 1.6. Dans ce schéma complet, nous pouvons identifier les sources de données principales. La première provient des commerces clients de la solution Bimedia, lesquels génèrent un large éventail de données, notamment liées aux ventes, aux événements de caisse, à l'utilisation des services tiers, et à d'autres activités. La deuxième source émane des équipes commerciales et de l'administration des ventes, qui renseignent les fiches des clients et prospects avec un ensemble complet d'informations légales. Une troisième source provient des partenaires de l'entreprise, principalement des fournisseurs et des annonceurs, par la mise à disposition d'APIs et de bases de données. Enfin, une quatrième source regroupe des données externes, qu'elles soient ouvertes – comme les rapports de recensement de la population fournis par l'INSEE – ou payantes – comme les API de Societe.com ou de Google Places.

Les deux sources de données internes mentionnées précédemment sont centralisées dans une base de données appelée « base d'agrégats », qui contient un ensemble de données agrégées avec différents niveaux de granularité. Par exemple, les ventes y sont agrégées par jour, par produit et par commerce. Cette base de données est de loin la plus utilisée pour les traitements de données internes. Par ailleurs, plusieurs types d'études nécessitent des traitements de données réalisés par des analystes ou d'autres fonctions de l'entreprise, en croisant les données internes avec des sources de données externes issues d'internet ou fournies par des partenaires. Actuellement, aucun lien n'est fait avec ces sources mis à part le géocodage automatique des adresses ; l'intégration des données de sources externes est donc une tâche à la charge de l'analyste.

5 Nouveaux besoins

Dans cette section, nous allons explorer les nouveaux besoins identifiés au sein de l'entreprise pour améliorer la valorisation des données, la compréhension des comportements des clients finaux et des commerces, ainsi que pour optimiser l'exploitation des données à des fins décisionnelles. Ces besoins ont été classés en quatre grandes catégories qui reflètent les limites décrites en section 3.5 : la classification des produits, la classification des commerces par activité, l'intégration de données de sources internes et externes, et la démocratisation de l'utilisation des nouvelles connaissances.

5.1 Classification des produits

L'un des besoins les plus urgents concerne la classification des produits vendus dans les commerces équipés de la solution Bimedia. Actuellement, une partie des produits est classée dans des familles « fixes », prédéfinies pour des catégories spécifiques comme le tabac, la presse ou les jeux d'argent. Si cette approche fonctionne bien pour ces types de produits, elle n'est malheureusement pas applicable aux autres catégories.

Il est donc nécessaire de mettre en place un système de classification plus dynamique et extensible, capable de catégoriser automatiquement les produits en fonction des données de vente et des informations disponibles. Un tel système permettrait non seulement de mieux comprendre les comportements d'achat des clients finaux, mais aussi de fournir des analyses plus précises et pertinentes pour les études marketing et la gestion de l'approvisionnement. Ce système devra être suffisamment flexible pour s'adapter aux évolutions du marché, en intégrant de nouvelles catégories de produits à mesure qu'elles apparaissent.

Enfin, la classification devra être simple à maintenir pour le personnel de l'entreprise et suffisamment robuste pour limiter les erreurs, de sorte que son fonctionnement requière un minimum d'interventions humaines. Les objectifs applicatifs liés à ces résultats seront, par exemple, de réaliser des études de tendance ou d'analyser globalement les ventes d'un commerce. Pour cela, la classification des produits n'a pas besoin d'être parfaite, d'autant que nous ne disposons pas des moyens pour en quantifier la précision exacte.

Dans ce manuscrit, nous détaillons comment ce problème a été résolu grâce au développement d'un transformateur basé sur un thésaurus de mots clés, et sur la combinaison des classifications partielles des produits par les commerçants. Ce système nous a permis de créer une classification flexible et évolutive, tout en répondant aux besoins de précision et de maintenabilité. Les résultats et les perspectives de cette solution seront abordés dans le chapitre 4.

5.2 Classification des commerces par activité

Actuellement, l'activité principale des commerces est renseignée manuellement lors de l'inscription d'un prospect dans le système d'information. Cette méthode, bien que fonctionnelle, est sujette à des erreurs et ne garantit pas des mises à jour adéquates, en particulier lorsque les commerces diversifient ou modifient leurs activités au fil du temps.

L'objectif est de développer une méthode automatisée pour identifier leurs activités. Cette approche pourrait s'appuyer sur des algorithmes d'apprentissage automatique, capables d'analyser les données de vente pour identifier les principales activités commerciales. Une telle solution pourra tirer parti d'une catégorisation correcte des produits vendus. En complément, des informations issues de sources externes, telles que les codes NAF/APE ou des bases de données commerciales, pourraient être intégrées pour affiner davantage cette classification.

Dans le chapitre 4, nous verrons comment nous avons répondu à ce besoin en développant une approche combinant l'utilisation d'un transformateur pour l'encodage des produits, suivi de l'entraînement d'un modèle d'apprentissage automatique capable de prédire les étiquettes d'activités des commerces en fonction de leurs ventes transformées. Ce processus a permis d'automatiser la classification tout en ouvrant des perspectives intéressantes pour des applications futures, que nous explorerons plus en détail dans la suite du manuscrit.

5.3 Intégration de données de sources internes et externes

Le troisième besoin identifié concerne l'intégration fluide et automatisée des données provenant de sources internes et externes. Actuellement, bien que certaines sources externes, telles que le géocodage des adresses, soient déjà intégrées automatiquement, une grande partie des données doit encore être insérée manuellement, ce qui entraîne de la redondance, une perte de temps et augmente les risques d'erreurs.

Il est essentiel de mettre en place des processus d'intégration de données plus automatisés afin d'enrichir les analyses et d'améliorer la connaissance de l'entreprise sur ses clients et leurs comportements. Cela implique l'intégration de référentiels externes, comme les codes NAF/APE ou des bases de données démographiques, mais aussi l'exploitation de sources internes encore sous-utilisées. L'automatisation de ces intégrations permettra non seulement de créer des modèles prédictifs plus précis, mais aussi d'optimiser les campagnes marketing et de personnaliser davantage les services offerts aux clients.

Dans le chapitre 5, nous détaillons comment nous avons réalisé avec succès ce travail d'intégration de données en utilisant une approche originale basée sur les graphes de connaissances et le traitement de données.

5.4 Partage des nouvelles connaissances

Enfin, un besoin fondamental pour l'entreprise est de démocratiser l'accès et l'utilisation des connaissances générées par ces nouvelles analyses avec des solutions adaptées. Actuellement, l'accès aux données et la capacité à les exploiter sont souvent limités aux experts techniques. Cette situation limite les bénéfices apportés par ces nouvelles connaissances sur l'ensemble de l'organisation.

Il est donc crucial de développer des outils et des interfaces plus intuitifs qui permettent à un plus grand nombre d'utilisateurs, même sans compétences techniques avancées, d'accéder aux données, de les analyser et de les exploiter pour prendre des décisions éclairées. Cela pourrait inclure des tableaux de bord interactifs, des rapports automatisés, et des outils de requêtage

simplifiés. La démocratisation de ces connaissances est un levier essentiel pour améliorer la prise de décision à tous les niveaux de l'entreprise et pour maximiser l'impact des données sur la performance de l'entreprise.

Nous détaillons dans le chapitre 5 les outils et interfaces que nous avons choisis de mettre en place, et comment ces derniers abaissent drastiquement le coût d'accès à l'information pour les utilisateurs non techniques.

6 Conclusion

Ce chapitre a permis de présenter les activités d'Orisha Retail Shops liées à la valorisation de la donnée. Nous avons vu comment l'entreprise, initialement reconnue pour sa caisse enregistreuse innovante, s'est transformée en une plateforme complexe et interconnectée, gérant d'importants volumes de données dont l'exploitation évolue d'un besoin strictement opérationnel vers une intégration quasiment systématique de l'information dans le processus de décision.

L'analyse des différents canaux de création et de valorisation des données a mis en lumière non seulement les forces de l'entreprise dans la gestion et l'exploitation des données de vente, mais également les limites rencontrées, notamment en matière de classification des produits et des commerces, d'intégration de sources de données externes, et d'autonomie des utilisateurs dans l'exploitation des données. Ces défis, une fois relevés, permettront à l'entreprise d'optimiser ses processus décisionnels, de mieux comprendre les comportements des clients finaux, et de maximiser la valeur des services offerts.

Dans les chapitres suivants, nous explorerons en détail les solutions envisagées pour répondre à ces nouveaux besoins, en mettant un accent particulier sur les méthodes innovantes de classification et d'intégration des données, ainsi que sur les outils destinés à rendre ces nouvelles connaissances accessibles à un plus large public au sein de l'entreprise.

Chapitre 2

État de l'art des approches d'encodage, d'étiquetage multiple et des bancs d'essai

Objectifs

L'objectif de ce chapitre est de présenter une revue complète de l'état de l'art autour de plusieurs concepts clés liés à l'encodage, l'incorporation (embedding), l'étiquetage multiple, ainsi que les bancs d'essai relatifs à notre contexte de recherche. L'étude s'inscrit dans le cadre d'une problématique complexe d'étiquetage multiple pour classifier les commerces en fonction de leurs ventes, un enjeu stratégique pour l'entreprise Orisha Retail Shops. Ce chapitre met en lumière différentes stratégies d'encodage et d'incorporation de variables textuelles courtes à haute cardinalité et d'étiquetage multiple. Il souligne également l'absence de bancs d'essai adaptés à notre contexte industriel, rendant nécessaire la création d'un banc d'essai personnalisé pour évaluer et comparer les différentes approches de classification de manière rigoureuse et pertinente.

Sommaire

1 Introduction

L'attrait croissant des entreprises pour l'exploitation et la valorisation des données a entraîné une évolution rapide des techniques de traitement, donnant naissance à des disciplines spécialisées comme le traitement d'images, le traitement du langage naturel et le traitement du signal. Autrefois distincts, ces domaines convergent désormais vers l'apprentissage automatique, un paradigme qui s'est imposé comme une solution incontournable pour résoudre de nombreuses problématiques industrielles. Depuis quelques années, une grande partie des recherches dans ces disciplines s'oriente vers l'apprentissage automatique, qui non seulement relève des défis complexes, mais trouve également des applications dans des secteurs tels que la prédiction des ventes, la segmentation des clients, l'analyse de marché ou encore la publicité ciblée.

L'efficacité de ces applications repose toutefois sur des méthodes de traitement de données de plus en plus sophistiquées, où l'étape de préparation ou de pré-traitement des données joue un rôle central. Cette phase permet de transformer des données brutes, souvent désorganisées et hétérogènes, en données structurées et exploitables par les algorithmes. C'est grâce à ce processus que les algorithmes peuvent non seulement extraire des caractéristiques pertinentes, mais aussi apprendre de manière efficace et robuste, garantissant ainsi des résultats fiables. L'étape de prétraitement comprend des tâches variées comme le nettoyage des données, la normalisation, la sélection de caractéristiques et la réduction de dimensionnalité, chacune pouvant potentiellement maximiser les performances des modèles d'apprentissage automatique.

L'un des principaux objectifs de cette étude est de comparer différentes approches d'étiquetage multiple, ou classification multi-labels, des commerces en fonction de leurs ventes.

Dans le cadre de cet objectif, nous passons alors en revue plusieurs approches liées au domaine de l'apprentissage automatique. L'apprentissage automatique (*machine learning*) est une branche de l'intelligence artificielle qui se concentre sur la création de systèmes capables d'apprendre à partir de données. Au cœur de cette discipline se trouvent des algorithmes d'apprentissage supervisé et non supervisé, qui, grâce à un entraînement sur des ensembles de données, peuvent faire des prédictions sans avoir besoin d'instructions explicites (au sens algorithmique).

Dans cette étude, nous travaillons avec des **données tabulaires et hétérogènes**, comprenant à la fois des variables qualitatives (catégoriques) et quantitatives. Pour que ces données soient utilisables par les algorithmes d'apprentissage automatique, comme ceux utilisés pour l'étiquetage multiple, il est nécessaire de les **transformer** en une forme compatible avec ces modèles, à savoir des vecteurs de nombres réels dans un **espace vectoriel** (Vector Space Model ou VSM). Cette transformation est essentielle car elle permet d'utiliser des outils mathématiques tels que l'algèbre linéaire pour analyser, comparer et classer les données, permettant ainsi ce que l'on appelle l'apprentissage. Les approches courantes de transformation sont l'encodage et l'incoporation (*embedding* en anglais).

Sous cette problématique globale de caractérisation des activités des commerces, nous identifions trois axes de recherche, chacun étant abordé dans une section de cet état de l'art :

— encodage et incorporation (section 2),

— transformations pour entrées à taille variable (section 3),

— étiquetage multiple (section 4).

Une dernière partie (section 5) se concentre elle sur les outils et jeux de données utilisés pour l'évaluation de ces solutions.

2 Encodage/incorporation des référentiels produits : affronter la haute cardinalité

Le premier axe de recherche porte sur l'encodage ou l'incorporation des variables qualitatives. Comme mentionné précédemment, les algorithmes d'apprentissage automatique, utilisés notamment pour des tâches telles que l'étiquetage multiple (attribuer une ou plusieurs catégories d'activités à un commerce), nécessitent que ces variables soient converties en valeurs numériques.

L'encodage ou l'incorporation des variables qualitatives dans le jeu de données fourni par l'entreprise présente des défis particuliers. En effet, certaines de ces variables contiennent de nombreux synonymes, avec différentes valeurs textuelles renvoyant à une même catégorie (par exemple, « vape », « cigarette électronique » et « e-cigarette »). De plus, ces variables subissent des variations morphologiques, telles que des abréviations ou des fautes d'orthographe (par exemple, « Jeux à gratter » et « Jeux à grt »). L'enjeu central ici est donc d'identifier une approche capable de gérer efficacement les synonymes et les variations morphologiques.

Dans cette section, nous explorons le défi d'intégrer des labels textuels avec une haute cardinalité, c'est-à-dire comportant un grand nombre de valeurs distinctes, tout en prenant en compte les synonymes et les variations morphologiques. Ces particularités posent des problèmes spécifiques en termes de complexité et de performance des modèles d'apprentissage automatique.

2.1 Critères de sélection des approches

Pour cet état de l'art, nous avons retenu quatre critères principaux afin de garantir la pertinence des approches sélectionnées par rapport à notre problématique.

— Les approches doivent concerner l'encodage de variables qualitatives avec une haute cardinalité, en tenant compte des variations morphologiques, des synonymes ou des deux.

— Les jeux de données utilisés pour la validation des approches doivent inclure au moins une variable qualitative non ordinale, sous forme textuelle, présentant des synonymes, des variations morphologiques ou des incohérences. Étant donné que ces aspects sont rarement détaillés dans la littérature, nous chercherons des jeux de données dits « sales » (*dirty*), marqués par des variations ou incohérences textuelles.

— Nous privilégions les approches non supervisées, car elles évitent la nécessité de créer manuellement un jeu de données étiqueté, tâche longue et coûteuse.

— Les valeurs textuelles doivent être des libellés et non des phrases, afin d'assurer une cohérence avec notre propre jeu de données et de répondre à nos besoins spécifiques.

2.2 Étude des approches d'encodage de variables qualitatives

L'**encodage**, est l'une des approches qui permet de rendre les données de diverses formes (texte, images, signaux) intelligibles par les algorithmes d'apprentissage automatique. Pour les données textuelles, des méthodes comme le **Bag-of-Words**[Zhang, 2010] ou le **TF-IDF** (Term Frequency-Inverse Document Frequency) [Aizawa, 2003] sont souvent utilisées pour représenter les mots sous forme numérique, en tenant compte de leur fréquence d'apparition dans un texte ou un corpus.

Cependant, comme exprimé en début de section 2, les caractéristiques de notre jeu de de données rendent les techniques classiques peu adaptées car elles ne tiennent pas compte des relations entre les différentes valeurs textuelles. La figure 2.1 illustre comment l'application de *One-hot encoding*, une approche répandue pour l'encodage de valeurs textuelles, à la variable « Type de produit » génère un grand nombre de colonnes supplémentaires, augmentant ainsi la complexité du traitement. Cette figure montre également l'objectif recherché si cette méthode prenait en considération les synonymes (par exemple, « Vape » et « Cigarette électronique ») ainsi que les variations morphologiques (par exemple, « Jeux à gratter » et « Jeux à grt »).

Encodage réel

Store ID	Libellé produit	Quantité
A1234	Vape	12
	Billeterie	31
	ecigarette	58
	Jeux à gratter	14
B4327	Cigarette electronique	17
	Boulangerie	76
	Jeux à grt	47
D8075	Pains et baguettes	36

Encodage souhaité

Interprétation réelle de la variable « Libellé produit » par *One hot encoding*

Store ID	Vape	Billetterie	ecigarette	Jeux à gratter	Cigarette electronique	Boulangerie	Jeux à grt	Pains et baguettes	Quantité
A1234	1	0	0	0	0	0	0	0	12
	0	1	0	0	0	0	0	0	31
	0	0	1	0	0	0	0	0	58
	0	0	0	1	0	0	0	0	14
B4327	0	0	0	0	1	0	0	0	17
	0	0	0	0	0	1	0	0	76
	0	0	0	0	0	0	1	0	47
D8075	0	0	0	0	0	0	0	1	36

Interprétation souhaitée de la variable « Libellé produit » par *One hot encoding*

Store ID	Vape Cigarette électronique ecigarette	Billetterie	Jeux à gratter Jeux à grt	Boulangerie Pains et baguettes	Quantité
A1234	1	0	0	0	70
	0	1	0	0	31
	0	0	1	0	14
B4327	1	0	0	0	17
	0	0	0	1	76
	0	0	1	0	47
D8075	0	0	0	1	36

FIGURE 2.1 – Encodage de la variable « Type de produit » avec la technique *One-hot encoding*

Deux grandes catégories d'approches se démarquent dans la littérature pour l'encodage de labels : d'une part, les publications traitant de l'encodage de variables incluant des synonymes, et d'autre part, celles se concentrant sur l'encodage des variables présentant des morphologies multiples.

Encodage de variables qualitatives avec synonymes Les recherches sur l'encodage de variables qualitatives (catégoriques) avec synonymes sont peu nombreuses. En effet, les synonymes

sont étudiés dans le cadre du traitement de langage naturel (NLP pour *natural language processing* en anglais), mais rarement dans le contexte de données tabulaires. Pour rappel, comme cela est présenté dans la figure 2.1, des synonymes sont différentes valeurs textuelles qui font référence à un seul et même concept (exemple de « ecigarette » et de « cigarette électroniques »).

Les travaux présentés dans l'article [Turney, 2008] introduisent le modèle *PairClass*, qui semble performant pour l'identification de synonymes de manière non-supervisée. Cependant, dans cet article comme pour d'autres [Turney, 2006], l'identification des synonymes repose sur un contexte dans lequel les mots sont utilisés, le reste de la phrase, ce qui n'est bien entendu pas applicable dans notre cas, les valeurs textuelles étant des libellés et non des phrases, nous n'avons pas ce type de contexte à disposition.

Encodage de variables qualitatives à morphologies multiples Pour traiter les variables qualitatives présentant des morphologies multiples, une approche courante et utilisée depuis de nombreuses années consiste à pré-traiter les données textuelles à l'aide d'outils dédiés au traitement du texte, avant d'appliquer des méthodes classiques d'incorporation de variables qualitatives [Pyle, 1999 ; Rahm, 2000]. C'est précisément ce type d'approche qui a été adoptée par Reilly et al. [Reilly, 2022], où la résolution du problème repose sur un prétraitement des données textuelles visant à corriger les fautes d'orthographe. Cependant, dans notre cas, cette solution n'est pas toujours applicable en raison de la nature spécifique du vocabulaire utilisé dans notre domaine, qui inclut des marques et des abréviations absentes des dictionnaires traditionnels.

Par ailleurs, Pachpande et al. [Pachpande, 2022] présentent des résultats prometteurs en ce qui concerne la déduplication des valeurs catégoriques. Ce travail se concentre principalement sur la gestion des variations morphologiques, telles que les abréviations, l'utilisation de caractères spéciaux, la présence de majuscules et les fautes d'orthographe, ainsi que sur les techniques permettant de les identifier. L'article teste trois approches de prétraitement et cinq modèles de classification, et démontre que le calcul de distances entre chaînes de caractères, utilisé pour générer de nouvelles variables, couplé à un modèle supervisé de type forêt aléatoire (*Random Forest*), donne de meilleurs résultats, avec une précision de 95% dans leurs expérimentations. La figure 2.2 présente le fonctionnement de cette méthode supervisée sur des exemples tirés du vocabulaire métier.

Trois autres solutions sont également évaluées par Cerda et al. [Cerda, 2020] : *similarity encoding*, la factorisation Gamma-Poisson et l'encodage MinHash.

Similarity encoding [Cerda, 2018], dérivée de la technique *one-hot encoding*, prend en compte les similarités entre les valeurs catégoriques. Cette approche consiste à construire des vecteurs d'entrée à partir des similarités entre catégories. Un gain de performance significatif a été observé avec cette solution en comparaison avec le *one-hot encoding*, le *bag of character n-grams*, et d'autres approches, grâce à l'utilisation de la similarité sur des *3-grams*. Cette technique est particulièrement efficace pour les variables catégoriques dites « sales » (*dirty*), car elle capture les ressemblances morphologiques. Une fois cette méthode appliquée, chaque valeur catégorique est représentée non plus par un texte, mais par un vecteur contenant les distances entre valeurs catégoriques. Un exemple de cette approche est illustré en figure 2.3.

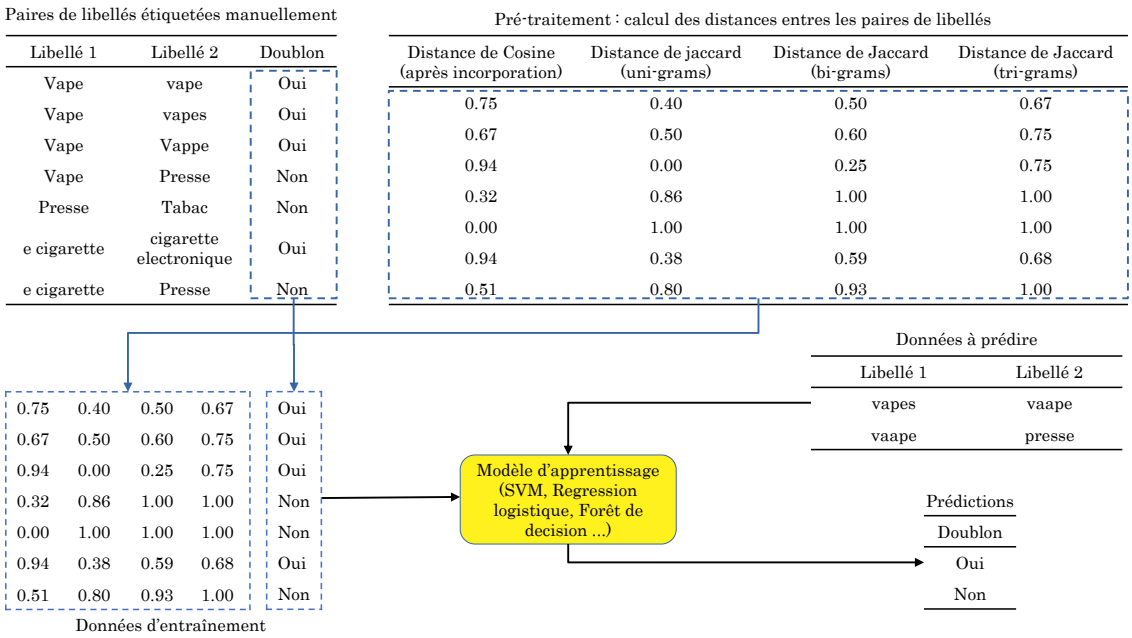


FIGURE 2.2 – Méthode présentée dans les travaux de PACHPANDE et al.

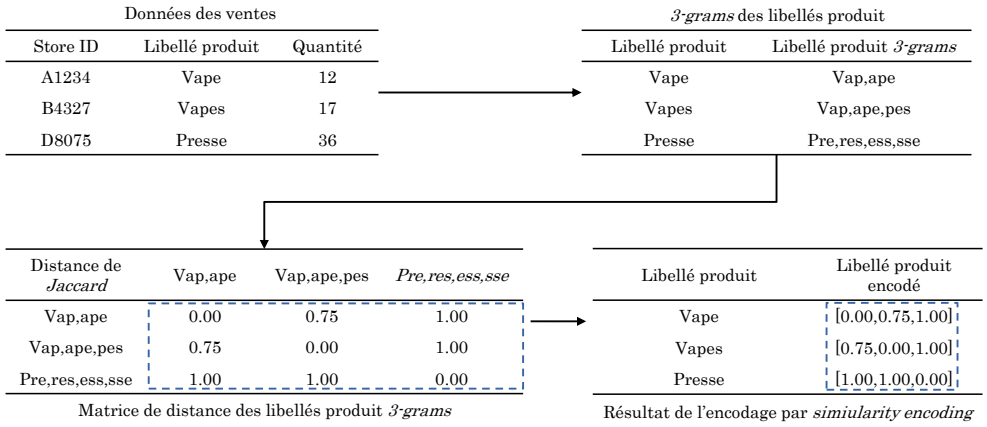


FIGURE 2.3 – Méthode *similarity encoding* présentée dans les travaux de CERDA et al.

2-gramme	h_1	h_2	h_3
va	e72a78	b0f0a0	9f994a
ap	f64905	47bc70	3095ae
pe	5119fa	80dc92	c06963
pp	efb311	41d0e3	fd3499
MinHash pour « vape »	5119fa	47bc70	3095ae
MinHash pour « vappe »	5119fa	41d0e3	3095ae

TABLE 2.1 – Exemple de calcul de MinHash

La factorisation Gamma-Poisson appliquée aux matrices de comptage de sous-chaînes de caractères. Cette méthode utilisée pour l'encodage de variables textuelles modélise les occurrences de sous-chaînes (séquences de caractères) comme des données de comptage suivant une distribution de Poisson, tandis que les paramètres latents suivent une distribution Gamma. La décomposition produit des matrices latentes, qui capturent des motifs cachés ou des régularités dans les données textuelles. Par exemple, cette approche permet d'encoder des textes en identifiant des motifs récurrents de sous-chaînes, ce qui peut ensuite être utilisé pour des tâches comme la classification ou l'analyse sémantique, en représentant les textes sous forme de combinaisons de ces motifs [Canny, 2004].

L'encodage MinHash est une technique pour calculer efficacement une valeur approchée de l'indice de Jaccard entre deux ensembles de n-grammes. MinHash fait pour cela appel à plusieurs fonctions de hachage – leur nombre est configurable, et représente un compromis entre la précision de l'approximation et le temps de calcul. Chacune d'entre elles est appliquée sur tous les n-grammes, et seule la valeur minimale est conservée. Ce fonctionnement est illustré pour le calcul de la similarité entre les deux chaînes « vape » et « vappe ». Leurs 2-grammes sont respectivement {va, ap, pe} et {va, ap, pp, pe}. L'indice de Jaccard est défini comme :

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

$$\text{Ici, } J(\{va, ap, pe\}, \{va, ap, pp, pe\}) = \frac{3}{4}$$

On considère trois fonctions de hachage h_1, h_2 et h_3, appliquées à chaque 2-gramme. Les valeurs calculées sont présentées dans la table 2.1. Une valeur minimale est conservée pour chaque fonction de hachage et pour chaque label. La valeur de l'indice de Jaccard est alors approximé par la proportion de MinHash égaux, soit ici un tiers – les valeurs de MinHash calculées avec h_2 sont différentes ; les valeurs calculées avec h_1 et h_3 sont identiques.

CERDA et al. [Cerda, 2020] considèrent le résultat de MinHash (les deux dernières lignes de la table 2.1) comme une technique d'encodage des labels.

Ces trois approches semblent particulièrement adaptées à notre problématique d'encodage des variables qualitatives présentant des variations morphologiques. Elles offrent une alternative efficace aux méthodes d'encodage usuelles pour traiter des données textuelles complexes. Elles ont d'ailleurs montré des performances intéressantes dans la compagne d'évaluation réalisée par

Jeu de données	Onehot enc. SVD	Similarity enc.	TfIdf SVD	Fast-Text SVD	Bert SVD	Gamma Poisson	Min-hash enc.
drug directory	0.971	0.979	0.980	**0.982**	0.979	0.980	0.981
midwest survey	0.575	0.635	0.646	0.636	0.605	0.651	**0.653**
traffic violations	0.782	0.789	0.789	0.790	0.792	0.792	**0.793**

TABLE 2.2 – Performance des techniques d'encodage [Cerda, 2020]

CERDA et al. [Cerda, 2020], et dont les résultats pour des problèmes de classification multiclasses sont reproduits dans le tableau 2.2. Ce tableau présente l'exactitude médiane obtenue à partir de 20 validations croisées sur des échantillons aléatoires issus de trois jeux de données avec des problématiques de haute cardinalité. Le modèle utilisé pour les tâches de prédiction est XGBoost avec une dimension des entrées fixée à 30. Une étape de réduction de dimension est parfois appliquée à certaines approches d'encodage ou d'incorporation afin d'optimiser les performances, indiquée par le sigle SVD (*singular value decomposition*).

À noter que ces trois jeux de données sont entièrement en anglais. Cependant, une étude complémentaire [Cerda, 2020] montre une baisse des performances lors de l'utilisation du modèle pré-entraîné FastText sur des données en français.

2.3 Étude des approches d'incorporation de variables qualitatives

L'**incorporation** (*embedding* en anglais), est couramment utilisée dans le domaine du traitement automatique du langage (TAL), où elle permet de convertir des éléments tels que des mots ou des phrases en vecteurs, en essayant de conserver une partie de leur sémantique. Des techniques comme **Word2Vec** [Church, 2017] et **GloVe** [Pennington, 2014] créent des représentations vectorielles en apprenant les relations contextuelles entre les mots, de manière à ce que des mots similaires se retrouvent proches dans l'espace vectoriel.

Alors que l'encodage convertit les données qualitatives en représentations numériques simples, comme le *One-hot encoding* qui génère des vecteurs binaires sans capturer de relations sémantiques entre les catégories, l'incorporation produit des vecteurs denses et continus qui saisissent les relations sémantiques entre ces catégories, offrant ainsi une représentation plus compacte et riche en informations.

Les **transformateurs** (transformers) représentent eux une approche moderne de l'incorporation. Ces modèles d'apprentissage automatique ont révolutionné des domaines comme le TAL grâce à leur mécanisme d'**attention**, qui permet à chaque élément d'une séquence de données de considérer toutes les autres positions dans cette séquence. Cette capacité à comprendre le contexte global améliore considérablement les performances dans des tâches comme la traduction automatique, la génération et la compréhension de texte. Un exemple bien connu est le modèle **BERT** (Bidirectional Encoder Representations from Transformers) [Tenney, 2019], pré-entraîné sur de vastes corpus textuels et affiné pour des tâches spécifiques. BERT dépasse les encodages traditionnels en exploitant les relations sémantiques entre les mots et en s'appuyant

sur un modèle statistique pré-entraîné pour améliorer la qualité des représentations vectorielles.

Dans la plupart des travaux sur l'incorporation de variables textuelles à haute cardinalité, l'utilisation de modèles d'incorporation pré-entraînés de type **transformateur** tels que **BERT** [Tenney, 2019] ou **FastText** [Wu, 1992] est fréquente. Ces modèles, souvent basés sur des réseaux de neurones, sont conçus pour transformer des mots ou des phrases en vecteurs, en essayant de traduire une similarité sémantique en une proximité numérique. Ces techniques sont communément appelées *modèles de langage*.

Cependant, leurs performances ne sont pas toujours concluantes sur ce type de données comme a pu le montrer les travaux de Cerda et al. avec l'usage des modèles pré-entrainés **BERT** et **FastText**, dont les résultats sont présentés en tableau 2.2.

En figure 2.4, nous illustrons les limites de l'utilisation de ces approches sur les libellés de produits de notre contexte. Nous avons sélectionné deux modèles largement utilisés :

— **Sentence Transformer**, basé sur le modèle pré-entraîné MPNet [1]

— **Word2Vec**, dans sa version pré-entraînée sur le WaCky Wide Web français [2]

Pour chacun de ces modèles, nous présentons une matrice de similarité en figure 2.4, où les valeurs correspondent aux similarités cosinus entre les vecteurs des libellés. Ces matrices permettent d'identifier rapidement où les modèles ont réussi ou échoué à rapprocher des libellés sémantiquement proches.

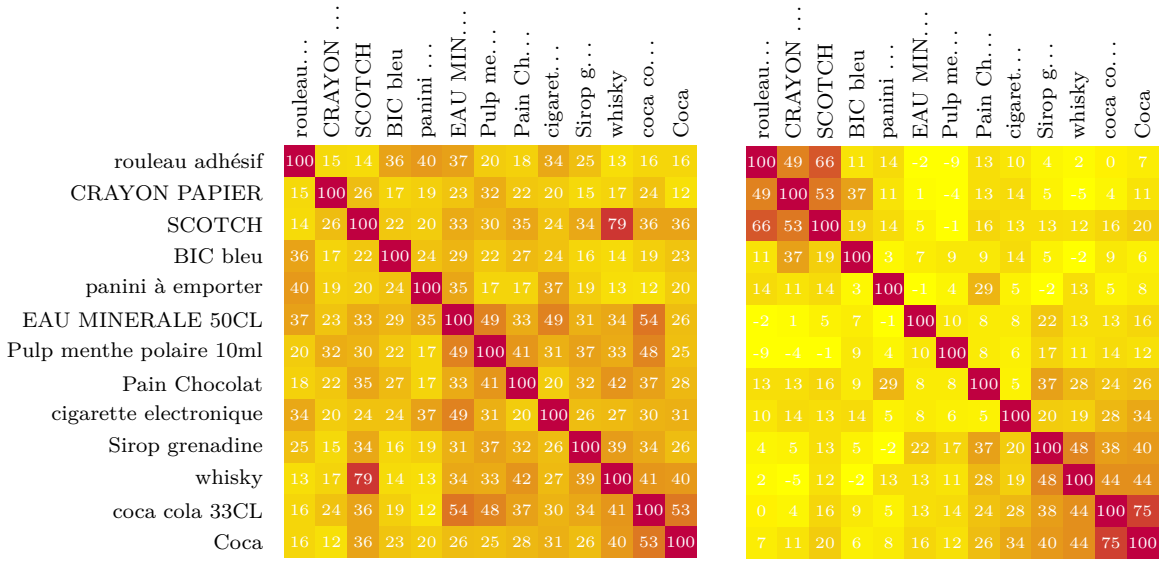
Les résultats d'incorporation par le modèle *Sentence Transformer* (figure 2.4a) présente des performances discutables. Par exemple, le liquide de cigarette électronique (« *pulp menthe polaire 10 ml* ») est considéré comme plus proche de l'eau minérale et du soda (« *coca cola 33cl* ») que de la cigarette électronique. Cependant, certaines similarités sont correctement identifiées, comme la proximité entre la canette de soda et son synonyme (« *coca* »), ou entre le stylo (« *bic bleu* ») et le rouleau adhésif.

Les résultats d'incorporation par le modèle *Word2Vec* (figure 2.4b) a également quelques erreurs, mais le modèle parvient globalement à regrouper les produits en deux catégories : les articles de papeterie et les boissons, visibles par deux zones sombres dans la matrice de similarité (respectivement en haut à gauche et en bas à droite).

Dans la figure 2.5, nous utilisons à nouveau *Word2Vec* pour incorporer trois libellés. Les deux premiers réfèrent au même produit (une confiserie) avec des variations mineures, tandis que le troisième est un liquide de cigarette électronique. Le modèle identifie à tort une grande proximité entre le second libellé et le troisième, alors qu'une proximité était attendue entre les deux premiers. Ce type de variabilité dans les libellés est fréquent dans les commerces Bimedia.

Considérant les résultats issus des travaux de Cerda et al. présentés dans le tableau 2.2 ainsi que les quelques tentatives d'incorporation réalisées sur les données de vente, nous pouvons observer des performances limitées de ce type d'approche sur des données textuelles courtes à haute cardinalité. Nous pouvons supposer que ces faibles performances sont liées à plusieurs raisons.

1. https://huggingface.co/sentence-transformers/all-mpnet-base-v2
2. https://fauconnier.github.io/#data



(a) Sentence Transformers (b) Word2Vec

FIGURE 2.4 – Exemples d'incorporation avec différents modèles de language pré-entrainés

FIGURE 2.5 – Exemples visuels des résultats de l'incorporation par le modèle *Word2Vec*

Premièrement, lors de nos premières analyses de l'état de l'art en 2021, les modèles de langage n'étaient pas aussi avancés qu'aujourd'hui. Leur adoption massive et l'intérêt majeur qu'ils suscitent actuellement dans la recherche étaient encore limités. De plus, leurs performances, en particulier pour le traitement du français, étaient nettement moins développées.

Deuxièmement, le langage utilisé dans les catalogues de produits, notamment pour les articles liés au tabac, est souvent très spécifique et hors des dictionnaires standards. Cela crée un décalage important avec les bases de connaissances utilisées pour entraîner ces modèles de langage.

Cela dit, nous sommes convaincus qu'au vu des avancées significatives réalisées au cours des deux dernières années, il serait pertinent de réévaluer les solutions de l'état de l'art en les comparant avec les nouveaux modèles de langage pré-entraînés. Nous avons notamment relevé plusieurs nouveaux modèles prometteurs [Costa, 2023 ; Petukhova, 2024] sur des problématiques d'incorporation comme *text-embedding-ada-002* d'OpenAI, *LLaMA2-chat* de Meta, ou *falcon-180B* du *Technology Innovation Institute*.

2.4 Modèles hybrides capables de traiter directement des variables qualitatives

Dans cette section, nous abordons quelques exemples de modèles capables d'effectuer des tâches de classification directement sur des jeux de données comprenant des variables qualitatives, sans phase d'encodage préalable. La majorité des approches utilisées pour traiter des données tabulaires hétérogènes sont des modèles hybrides. Les plus répandus sont les arbres de décision avec renforcement de gradient, une famille d'algorithmes appelée *Gradient Boosting Decision Tree* (GBDT), dont les deux exemples les plus connus sont *XGBoost* [Chen, 2016] et *CatBoost* [Prokhorenkova, 2018]. Les arbres de décision sont particulièrement adaptés au traitement de données tabulaires, car ils gèrent efficacement les valeurs qualitatives ou discrètes sans nécessiter de pré-traitements complexes. Par exemple, les chaînes de caractères peuvent simplement être encodées en entiers. Le renforcement de gradient optimise les prédictions en minimisant les erreurs au sein de chaque arbre de décision, les forçant à converger vers la bonne réponse collective.

Une autre approche hybride prometteuse est celle des *Neural Oblivious Decision Ensembles* (NODE) [Popov, 2019], qui permet de traiter directement les données brutes. Une contribution similaire avec des avantages comparables est proposée par Borisov et al. [Borisov, 2022a], via la solution nommée *DeepTLF*.

Bien que ces modèles soient intéressants, ils ne se rattachent pas directement à notre problématique d'encodage, mais plutôt à la problématique plus générale de classification multi-label (ou étiquetage multiple).

2.5 Synthèse des travaux connexes sur l'encodage et l'incorporation de variables qualitatives

Un tableau de synthèse des différentes approches abordées dans cette section, associées aux critères définis en amont, est présenté en figure 2.3.

Méthode	Traitement des synonymes	Traitement des morphologies multiples	Traitement des variables qualitatives « sales »	Algorithme non supervisé	Adapté à des libellés
PairClass [Turney, 2008]	Oui	Non	Oui	Oui	Non
Categorical Data Deduplication [Pachpande, 2022]	Non	Oui	Oui	Non	Oui
Similarity encoding [Cerda, 2018]	Non	Oui	Oui	Oui	Oui
Factorisation de Gamma-Poisson [Canny, 2004]	Non	Oui	Oui	Oui	Oui
MinHash [Broder, 1997]	Non	Oui	Oui	Oui	Oui

TABLE 2.3 – Comparatif des approches d'encodage

Considérant les critères cités en section 2.1, et la performance des traitements sur des jeux de données présentant des caractéristiques similaires [Cerda, 2020], les approches les plus performantes sont : l'encodage *min-hash*, la factorisation Gamma-Poisson, et la *similarity encoding*.

Pour finir, il est essentiel de souligner les points suivants :

— L'approche *similarity encoding* repose sur le calcul de matrices de distances pour encoder les variables. Cependant, cette méthode tend à perdre la signification des valeurs initiales puisqu'elles sont remplacées par des mesures de similarité.

— L'approche par factorisation Gamma-Poisson génère des matrices creuses (sparse), ce qui peut poser des problèmes d'efficacité lors de leur traitement par certains algorithmes d'apprentissage automatique, particulièrement ceux nécessitant des représentations denses.

— L'approche *min-hash* utilise le calcul de hachages, une méthode irréversible. Cela implique qu'une fois les données encodées, il n'est plus possible de retrouver les valeurs initiales, ce qui est un aspect à prendre en compte pour l'implémentation et l'interprétation des résultats.

3 Gestion des entrées à tailles variables en apprentissage automatique

Le deuxième axe de recherche concerne la forme des entrées. Dans les exemples standards d'apprentissage automatique, les données d'entrée sont souvent présentées sous forme de matrices, ce qui facilite leur traitement par la majorité des algorithmes disponibles. Par exemple, l'élément A de la figure 2.6 illustre ce type de structure, une matrice, où chaque ligne représente une instance du jeu de données (dans notre cas, un commerce), et chaque symbole ($\square, \bigcirc, \triangle, \star$ et $\diamondsuit$) correspond à une référence de produit vendue. Si tous les commerces ven-

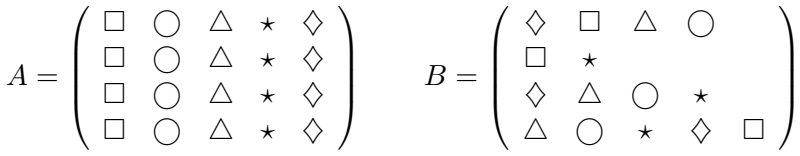


FIGURE 2.6 – Matrice homogène en A et listes irrégulières en B

daient exactement les mêmes produits, nous obtiendrions une matrice de la forme de l'élément A.

Cependant, dans notre contexte, ce n'est pas le cas. Un commerce peut vendre seulement vingt références de produits, tandis qu'un autre en vendra cent, ce qui est illustré par l'élément B de la figure 2.6. A noter que certains commerces ne vendent tout simplement pas de produits en communs, cela correspondrait à des lignes avec aucun symbole en communs dans notre exemple. Cette structure non rectangulaire, caractérisée par des tailles variables d'entrée, complexifie le traitement des données et demande des approches adaptées.

Dans les cas où les données sont sous forme de matrices, les approches de pré-traitement se concentrent principalement sur le nettoyage des données et sur des transformations telles que la réduction de dimensions ou l'encodage des variables qualitatives. Ces transformations sont souvent bien documentées et faciles à mettre en œuvre grâce à l'abondance d'outils dédiés.

Dans notre étude, les données ne suivent pas ce format simple. Elles sont représentées par l'élément B, où les entrées sont de tailles variables en fonction des produits vendus par chaque commerce. C'est cette spécificité qui constitue notre deuxième problématique : comment traiter efficacement des entrées à tailles variables tout en garantissant une cohérence et une comparabilité des résultats entre commerces ?

Pour relever ce défi de gestion des entrées à tailles variables en apprentissage automatique, diverses stratégies ont été développées. Celles-ci vont du pré-traitement des données, visant à adapter les entrées à taille variable aux modèles classiques, à l'utilisation d'algorithmes spécialement conçus pour gérer directement ces entrées hétérogènes.

3.1 Critères de Sélection des Approches

Dans cette état de l'art, nous avons séparé les typologies d'approches en trois parties, correspondant à trois axes de recherche différents. Une première sur les simples pré-traitements des entrées par des transformations usuelles. Une seconde qui couvre les approches par transformation de problème, dont l'objectif est de transformer la problématique en une autre plus connue ou plus facile à prendre en charge. Une troisième sur les approches par traitement direct, qui couvre les types de modèles capables de prendre directement les entrées à taille variable pour leur apprentissage et leurs prédictions.

3.2 Approches de pré-traitement des données sur la forme des entrées

Les approches par pré-traitement des données d'entrées correspondent globalement aux transformations les plus courantes et les plus simples à mettre en place. L'objectif est alors de modifier

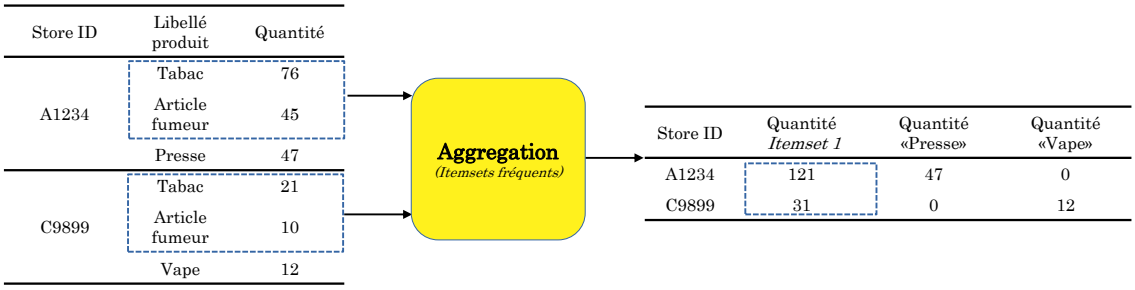


FIGURE 2.7 – Agrégation de deux variables avec la méthode *itemsets fréquents*

la structure des données afin de former une matrice qui sera facilement prise en charge par les modèles d'apprentissage automatique classiques. Cette section couvre notamment les approches telles que l'agrégation, l'augmentation, la troncation et d'autres approches hybrides.

3.2.1 Agrégation

L'agrégation permet de simplifier la représentation de certaines variables en les regroupant. Une technique classique d'agrégation est l'exploitation des *itemsets fréquents* [Borgelt, 2012 ; Moens, 2013], également appelée découverte des motifs fréquents. Cette méthode consiste à créer de nouvelles variables à partir de motifs récurrents dans un jeu de données. Cette approche a l'avantage de limiter l'augmentation du nombre de dimensions après transformation, ce qui peut se produire avec des techniques comme le *one-hot encoding* (voir figure 2.1). En effet, avec cette méthode, les variables ne sont plus traitées individuellement, mais par groupes.

3.2.2 Troncation

La troncation est une solution simple pour traiter des entrées à tailles variables. Cette approche consiste à réduire les données en alignant les variables des différentes entrées, puis à sélectionner celles qui sont partagées par toutes les entrées. Les données sont donc réduites à la taille de l'entrée la plus courte (celle avec le moins de variables). Cette méthode est facile à implémenter et réduit la complexité des données en supprimant des variables. Toutefois, elle entraîne une perte d'information importante. La figure 2.8 illustre cette méthode, où les informations sur les produits de type « Presse » et « Vape » sont perdues après la transformation. Dans notre cas, cette méthode n'est pas adaptée, car il n'existe aucun produit vendu par l'ensemble des commerçants.

3.2.3 Augmentation

Contrairement à la troncation, l'augmentation (*padding*) consiste à aligner toutes les variables présentes dans les entrées, puis à remplir les valeurs manquantes par une valeur arbitraire (souvent 0, méthode appelée *zero-padding*). Cette méthode est également simple à implémenter et permet de conserver toutes les informations. Cependant, elle génère souvent des matrices creuses, ce qui augmente la complexité des données, le temps de traitement, et les ressources

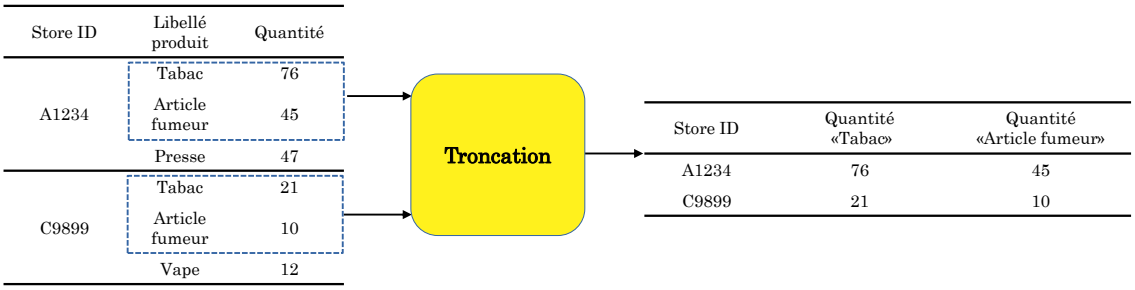


FIGURE 2.8 – Illustration de la troncation des entrées

matérielles nécessaires. Un exemple de la méthode *zero-padding* est visible en figure 2.9. De plus, ces matrices creuses ne sont pas toujours prises en charge par les algorithmes d'apprentissage automatique, ce qui peut mener à des résultats de faible qualité [Dwarampudi, 2019].

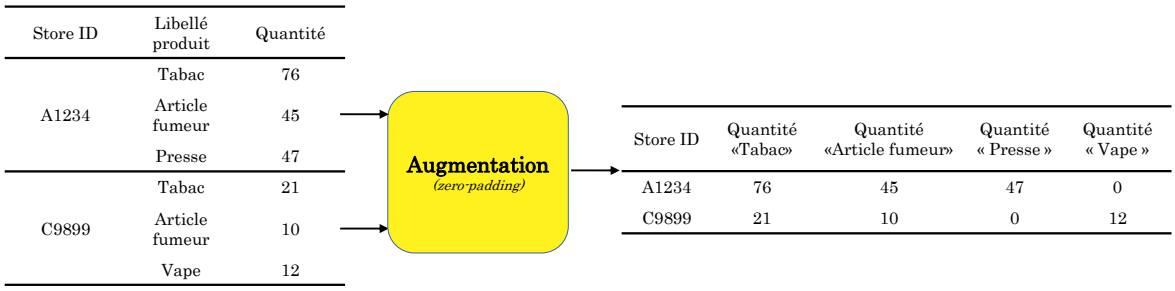


FIGURE 2.9 – Illustration de l'augmentation des entrées par la méthode *zero-padding*

3.2.4 Hybride

Plusieurs approches hybrides combinent troncation et augmentation avec d'autres techniques. Deux méthodes sont présentées ici.

La première consiste à augmenter les entrées du jeu de données, puis à réduire la dimensionnalité à l'aide de l'analyse en composantes principales (ACP ou PCA en anglais) [Güler, 2005 ; Gupta, 2016]. L'ACP transforme les variables corrélées en nouvelles variables non corrélées, réduisant ainsi le nombre de variables tout en préservant un maximum d'informations. Toutefois, un inconvénient majeur est que les variables résultantes perdent leur signification initiale [Abdi, 2010].

La seconde méthode consiste à augmenter les données, puis à sélectionner les variables pertinentes à l'aide des algorithmes SVM et RBF, ou du modèle statistique AFDM (analyse factorielle de données mixtes) [Visbal-Cadavid, 2020]. Contrairement à l'ACP, cette approche conserve la signification des variables, tout en réduisant la dimensionnalité de manière efficace. Ces méthodes de sélection de variables sont détaillées dans les travaux de CHANDRASHEKAR et al. [Chandrashekar, 2014].

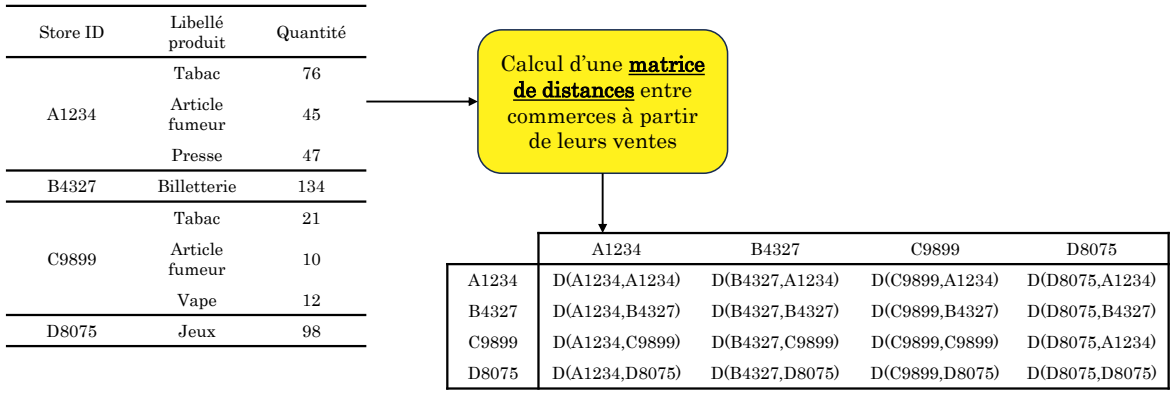


FIGURE 2.10 – Illustration de la transformation par calcul d'une matrice de distances

3.3 Approches par la transformation de problème

L'objectif des approches par transformation de problème est de convertir un problème initial, ici le traitement de données tabulaires à tailles variables, en un problème plus simple ou plus courant à résoudre. L'exemple le plus parlant est la transformation par calcul d'une matrice de distances. Concrètement, dans notre cas, les données des ventes des commerces seraient transformées en matrice de distances entre commerces, en fonction des similitudes dans leurs ventes.

3.3.1 Transformation par calcul d'une matrice de distances

Le calcul d'une matrice des distances entre les données d'entrée à l'aide de techniques adaptées aux vecteurs de tailles variables (comme *Longest Common Subsequence (LCSS)* ou *Hausdorff Distance*) est une méthode envisageable pour nos données. Une matrice de distance étant nécessairement carrée, elle peut être directement utilisée comme entrée pour les algorithmes d'apprentissage classiques. Une illustration de cette approche est présentée en figure 2.10, bien que le calcul des distances exactes n'ait pas été réalisé dans cette figure, car la principale difficulté réside dans le choix de la formule appropriée pour mesurer ces distances.

LIU et al. [Liu, 2016] proposent la solution *AED matrix*, qui consiste à transformer les données d'entrée en calculant une matrice de distances, en utilisant des mesures adaptées aux entrées à tailles variables telles que la *path distance* ou la distance de *manifold*. Malheureusement, cette solution ne peut être utilisée dans notre cas, car elle repose sur la notion de séquence, or nos données ne sont pas ordonnées (il n'y a pas de relation d'ordre entre les produits).

La difficulté ici consiste à identifier une distance appropriée pour nos données.

3.3.2 Transformation par calcul de textes

Une approche alternative, qui consiste à traiter des données tabulaires hétérogènes de manière analogue à des textes, est proposée par YIN et al. [Yin, 2020]. Ce modèle, nommé *TaBERT*, a été conçu pour extraire des informations structurées à partir de données hétérogènes contenues dans des tableaux. Cependant, ce modèle suppose que les tableaux sont liés à des textes, et adapte

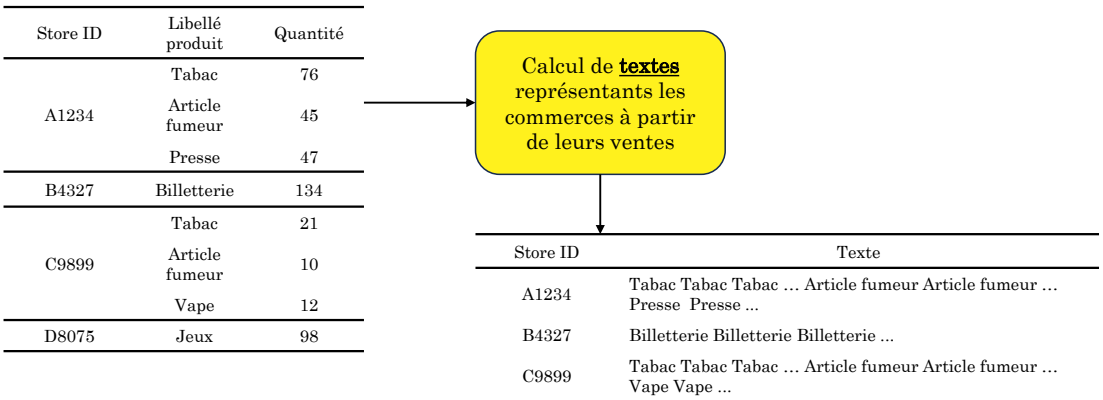


FIGURE 2.11 – Illustration de la transformation par calcul de textes

son extraction en fonction du contexte textuel environnant. Malheureusement, ce modèle ne convient pas à nos données, car elles ne comportent pas de notion de contexte, et *TaBERT* a été principalement entraîné sur des données textuelles en anglais. La figure 2.11 illustre un exemple de transformation par calcul de textes, où un texte est généré pour chaque commerce en fonction des types de produits et des quantités vendues.

3.3.3 Transformation par calcul d'images

Un autre axe de recherche explore la transformation de données tabulaires en images. Avec les progrès de la vision par ordinateur, notamment grâce aux réseaux de neurones convolutifs (CNN), il peut être intéressant de transformer les données en images pour tirer parti de ces algorithmes. Des exemples notables incluent *SuperTML* [Sun, 2019] et *IGTD* [Zhu, 2021], tous deux mentionnés dans la synthèse de BORISOV et al. [Borisov, 2022b]. L'algorithme *IGTD*, en particulier, minimise la différence entre l'ordonnancement des variables dans les données et la distance entre les pixels de l'image, chaque pixel représentant une valeur de variable.

Bien que les résultats obtenus soient prometteurs, cette méthode n'est pas adaptée à notre jeu de données, où certains commerces ne vendent qu'une petite quantité de produits, rendant difficile la constitution d'images cohérentes. Un exemple de transformation par calcul d'images est montré en figure 2.12, où chaque pixel représente un type de produit vendu.

3.3.4 Transformation par calcul de vecteurs

L'article de revue de BORISOV et al. [Borisov, 2022b] décrit l'utilisation d'algorithmes d'apprentissage automatique pour transformer des données d'entrée de taille variable en vecteurs de taille fixe. L'algorithme *VIME* [Yoon, 2020], par exemple, est un encodeur auto-supervisé qui représente les données originales sous une forme simplifiée, adaptée aux traitements ultérieurs. Ce processus comporte deux étapes : d'abord, un masque est généré à partir de la structure et des caractéristiques des données ; ensuite, les données d'entrée originales sont reconstruites à partir de ce masque. Une fois ces étapes réussies, le masque peut être utilisé comme entrée dans un modèle d'apprentissage supervisé. Cette méthode permet d'encoder des variables quantitatives

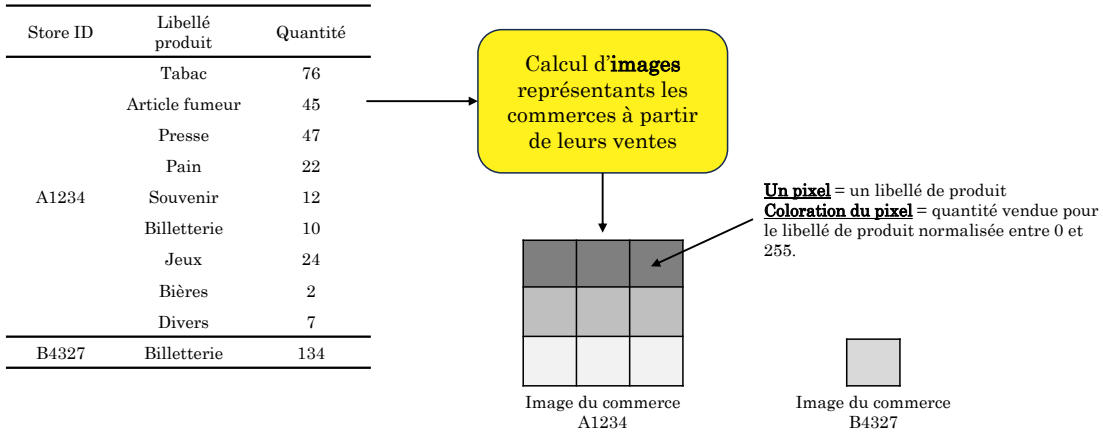


FIGURE 2.12 – Illustration de la transformation par calcul d'images

et qualitatives issues de données hétérogènes en vecteurs de taille fixe.

3.4 Approches par traitements directs

Une dernière famille d'approches existe, celle des approches par traitements directs. Lorsque nous parlons d'approches par traitement direct, cela signifie que ces techniques sont capables de prendre en charge directement des entrées à tailles variables sans nécessiter de phase de pré-traitement. Ces méthodes sont particulièrement intéressantes dans des contextes où la variabilité des données en termes de dimensionnalité ou de structure ne peut être facilement normalisée ou agrégée. Elles offrent une flexibilité accrue pour traiter des données hétérogènes tout en évitant les étapes de transformation qui peuvent introduire des biais ou une perte d'information.

3.4.1 Réseaux de neurones récursifs

CHO et al. [Cho, 2014] proposent l'utilisation de réseaux de neurones convolutionnels récursifs pour des tâches de traitement de langage naturel. Ce qui nous intéresse particulièrement dans cet article est la capacité des réseaux de neurones récursifs à traiter des entrées de tailles variables sans pré-traitements nécessaires. Cependant, le modèle proposé repose sur la structure grammaticale des phrases, ce qui ne correspond pas à la forme de nos entrées, rendant cette approche inapplicable à notre cas.

3.4.2 Réseaux de neurones récurrents

Les réseaux de neurones récurrents sont souvent mis en avant pour leur capacité à traiter des séquences à tailles variables. Ces réseaux sont utilisés pour des tâches telles que les fonctions de calcul sur les mots, l'approximation de systèmes dynamiques, ou le calcul de valeurs réelles à partir de séquences [Hammer, 2000]. Cependant, les données traitées par ce type de modèle présentent toujours une notion de séquence ou de dynamique temporelle, qui exige que les observations ne soient pas analysées individuellement, mais comme un ensemble cohérent. Par exemple, il est impossible de tirer une émotion d'une phrase en analysant chaque mot indé-

pendamment ; il faut considérer l'enchaînement des mots. De la même manière, pour modéliser un système dynamique, il est nécessaire d'observer la succession des valeurs. Nos données ne sont pas séquentielles, elles n'ont ni ordre ni temporalité, ce qui rend l'utilisation des réseaux de neurones récurrents inadaptée à notre étude.

3.5 Synthèse des travaux connexes sur le traitement des entrées à tailles variables

Dans le cadre de cet état de l'art sur le traitement des entrées à tailles variables, nous avons sélectionné des approches couramment utilisées pour traiter ce type de données, tout en évitant les méthodes nécessitant une phase d'apprentissage supervisé. En effet, l'objectif était de ne pas recourir à des modèles supervisés, compte tenu de la taille du jeu de données et du temps qu'il faudrait pour constituer des jeux d'apprentissage et de validation étiquetés de manière cohérente.

Pour choisir les approches les plus adaptées, nous avons défini trois critères d'évaluation, notés sur une échelle de 1 à 5, en fonction de nos recherches :

— **facilité d'implémentation**, évaluée en fonction des bibliothèques disponibles et de leur facilité d'intégration à l'environnement technique de Bimedia ;

— **faisabilité des approches**, évaluée à l'aide de notre compréhension du jeu de données fourni par Bimedia et de ses particularités ;

— **popularité dans la communauté scientifique**, mesurée par la fréquence d'utilisation et la reconnaissance des approches dans les articles scientifiques.

Dans le tableau 2.4, nous présentons une synthèse des différentes approches pour le traitement des entrées à tailles variables, ainsi que les scores associés à chacune d'entre elles. Les trois approches les plus prometteuses, identifiées en gras, ont été sélectionnées sur la base des scores totaux obtenus.

Famille d'approche	Type d'approche	Approche	Facilité d'implémentation (de 1 à 5)	Faisabilité de l'approche (de 1 à 5)	Popularité de l'approche (de 1 à 5)	Total
Pré-traitement des données sur la forme des entrées	**Agrégation**	**Itemsets fréquents**	4	4	3	**11**
	Troncation	Troncation	4	0	4	10
	Augmentation	**Zero-padding**	**5**	**5**	**5**	**15**
		Zero-padding + PCA	3	3	2	8
		Zero-padding + RBF	3	3	2	8
Transformation de problème	Calcul d'une matrice de distances	Hausdorff distance, AED matrix, etc.	3	0	0	3
	Génération de textes	TaBERT	3	0	4	7
	Génération d'images	SuperTML, IGTD	2	1	0	3
	Génération de vecteurs	VIME	2	1	4	7
Traitements directs	Réseaux de neurones récursifs	RvNN	3	4	1	8
	Réseaux de neurones récurrents	RNN	3	4	1	8
	Modèles hybrides	**XGBOOST, CATBOOST**	4	4	**5**	**13**

TABLE 2.4 – Synthèse des scores obtenus par les différentes approches pour le traitement des entrées à tailles variables

4 Panorama des techniques d'étiquetage multiple

Le troisième axe de recherche porte sur l'étiquetage multiple des commerces en fonction de leurs ventes. Dans notre contexte, un commerce peut exercer plusieurs activités simultanément (par exemple, tabac, presse et bar), ce qui nécessite l'utilisation d'approches d'étiquetage multiple pour classifier chaque commerce selon ses diverses activités. L'objectif est d'identifier la méthode d'étiquetage multiple la plus adaptée à notre jeu de données, marqué par la présence de variables qualitatives hétérogènes et des tailles d'entrées variables.

L'étiquetage multiple présente des défis spécifiques, car les modèles doivent non seulement identifier correctement les différentes activités commerciales, mais aussi prendre en compte les corrélations entre ces activités. Par exemple, un commerce vendant des produits de presse est souvent associé à une activité de vente de tabac. Le modèle doit donc être capable de capturer ces relations tout en assurant une performance satisfaisante pour des classes plus rares.

De nombreux travaux sont disponibles pour aborder la problématique de l'étiquetage multiple [Tsoumakas, 2007 ; Tsoumakas, 2009 ; Zhang, 2013]. Dans la littérature, trois grandes familles d'approches sont couramment utilisées pour effectuer de l'étiquetage multiple : la transformation de problème, l'adaptation d'algorithme et le traitement direct par des techniques ensemblistes. Dans cette section, nous aborderons les deux premières, la troisième étant moins commune. **La transformation de problème** consiste globalement à adapter la problématique d'étiquetage multiple en plusieurs sous-problèmes afin d'utiliser des algorithmes de classification classiques. **L'adaptation d'algorithmes** consiste à créer un nouveau type de modèle en modifiant des approches existantes pour qu'elles puissent répondre à la problématique d'étiquetage multiple.

Étant donné que nous ne prévoyons pas de proposer de nouvelles contributions autour de cette problématique spécifique, et compte tenu du grand nombre de publications et d'approches disponibles sur le sujet, cet état de l'art n'a pas vocation à couvrir l'ensemble de la littérature, mais de présenter les approches les plus communes.

4.1 Transformation de problème

L'approche par transformation de problème est largement répandue [Tsoumakas, 2007] pour aborder les problématiques d'étiquetage multiple. Cette méthode peut être subdivisée en trois catégories principales : la décomposition en plusieurs problèmes binaires, la décomposition en plusieurs problèmes de classification multi-classes, et l'utilisation d'approches basées sur des ensembles, telles que l'agrégation et l'empilement. Dans les sections suivantes, nous discuterons d'une méthode représentative pour chacun de ces trois types.

4.1.1 Décomposition en problèmes binaires

Les approches par décomposition en problèmes binaires se divisent en deux techniques principales : *OVA* et *OVO* [Raziff, 2017].

OVA (*One-vs-All*) consiste à entraîner un classifieur pour chaque étiquette. Chaque classifieur est entraîné à prédire une seule étiquette, et il y a donc autant de classifieurs que d'étiquettes. Chaque classifieur répond par « Oui » ou « Non » concernant l'étiquette pour laquelle

il a été entraîné. Par exemple, un classifieur pourrait être spécialisé dans la reconnaissance des bureaux de tabac, tandis qu'un autre identifierait les magasins de presse. Si pour un commerce donné, les deux classifieurs répondent positivement, le commerce est alors considéré à la fois comme un bureau de tabac et un magasin de presse.

OVO (*One-Vs-One*) entraîne des classifieurs pour identifier une classe dominante entre deux étiquettes à la fois. Cela nécessite d'entraîner autant de classifieurs que de combinaisons possibles de paires d'étiquettes. La sélection des étiquettes finales se fait via un système de votes. Par exemple, un classifieur est entraîné à prédire si un commerce est un bureau de tabac ou un magasin de presse, et un autre est entraîné à choisir entre magasin de presse et boulangerie. Les prédictions des classifieurs sont ensuite ordonnées selon leur probabilité, et un seuil est défini pour sélectionner les étiquettes au-dessus d'une certaine probabilité.

Ces deux techniques (OVA et OVO), initialement conçues pour résoudre des tâches de classification multi-classe à partir de modèles de classifications binaires, sont utiles pour résoudre les problématiques d'étiquetage multiple quand elles sont combinées avec d'autres méthodes, que nous aborderons ci-dessous.

Binary Relevance (BR) La méthode *Binary Relevance (BR)* est l'une des approches les plus connues et les plus performantes de la décomposition en problèmes binaires. Elle consiste à décomposer le problème d'étiquetage multiple en autant de problèmes binaires qu'il y a d'étiquettes. Ses avantages et limites sont détaillés par Read et al. [Read, 2011] et Zhang et al. [Zhang, 2018]. Le principal inconvénient de cette méthode est qu'elle ne prend pas en compte la corrélation entre les étiquettes [Bogatinovski, 2022]. Habituellement, cette méthode est couplée à des classifieurs SVM (Support Vector Machine), où le SVM est utilisé comme classifieur pour chaque étiquette [Madjarov, 2012]. Chaque modèle, spécifique à une classe, décide si l'observation appartient à la classe ou non. Un exemple permettant d'illustrer son fonctionnement est présent en figure 2.13.

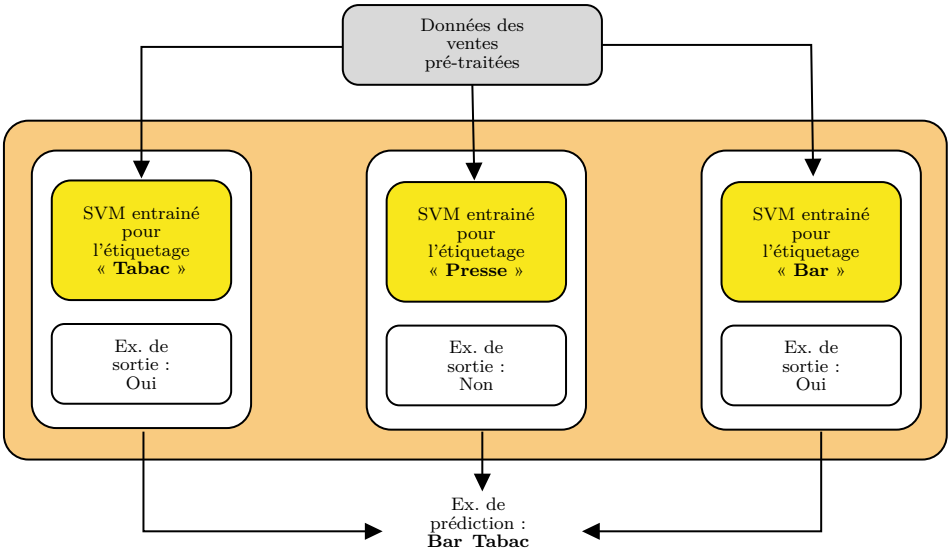


Figure 2.13 – Étiquetage multiple par *Binary Relevance*

Classifier Chains (CC) La méthode *Classifier Chains (CC)* [Bogatinovski, 2022] est une extension de la méthode *Binary Relevance (BR)*. Elle présente de meilleures performances en prenant en compte les corrélations entre les étiquettes, contrairement à *BR*. Dans cette approche, chaque modèle de type *BR* est chaîné, les prédictions de chaque modèle étant utilisées comme caractéristiques d'entrée pour les modèles suivants. Cela permet de capturer les relations entre les étiquettes. Le fonctionnement de ce modèle est illustré avec un exemple simple en figure 2.14.

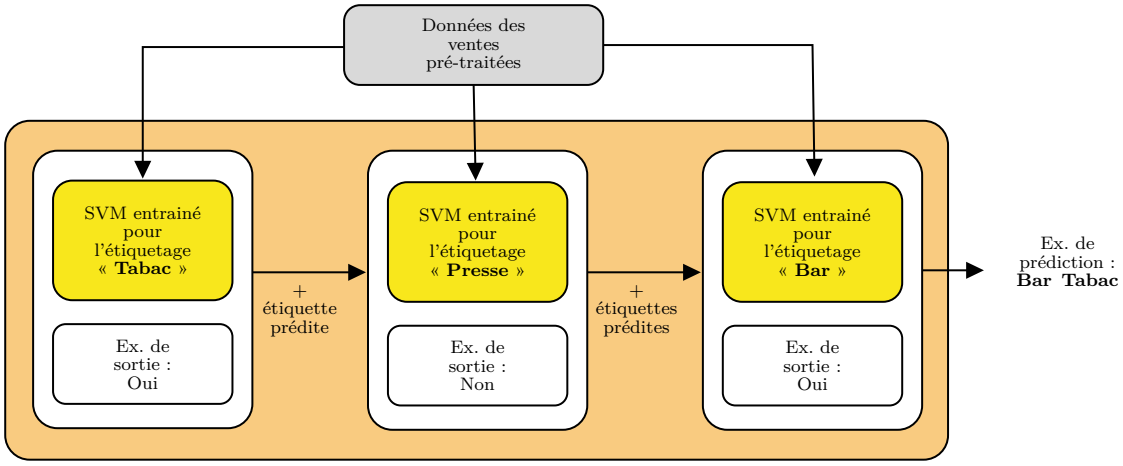


FIGURE 2.14 – Étiquetage multiple par *Classifier Chains (CC)*

4.1.2 Transformation en problèmes multi-classes

La transformation en problème multi-classe consiste cette fois à utiliser un ou plusieurs modèles capables chacun de prédire l'une des différentes étiquettes à prédire. Contrairement aux approches précédentes se basant sur plusieurs classificateurs binaires (sortie « Oui » ou « Non »), les modèles déterminent ici une étiquette unique chacun. Les suggestions d'étiquettes faites par les différents modèles sont ensuite transformées pour obtenir un étiquetage multiple.

Label Powerset (LP ou LC). *Label Powerset* est l'approche la plus couramment utilisée pour ce type de transformation de problème. Cette technique consiste à transformer un problème d'étiquetage multiple en un problème de classification où chaque combinaison possible d'étiquettes devient une classe distincte. Cette méthode permet d'utiliser tous les classifieurs multi-classes et préserve les relations entre étiquettes. Cependant, elle peut se montrer inefficace si le nombre de combinaisons possibles d'étiquettes devient trop important, et est sujette au sous-apprentissage. Les deux classifieurs les plus utilisés avec cette méthode sont le SVM et K-means. Avec cette approche, le modèle serait, par exemple, entraîné à prédire l'étiquette « Boulangerie », mais également « Boulangerie Presse », « Boulangerie Tabac », et ainsi de suite pour toutes les combinaisons possibles. Cela peut représenter un grand nombre d'étiquettes, sachant que certains commerces d'Orisha cumulent, en pratique, jusqu'à neuf activités.

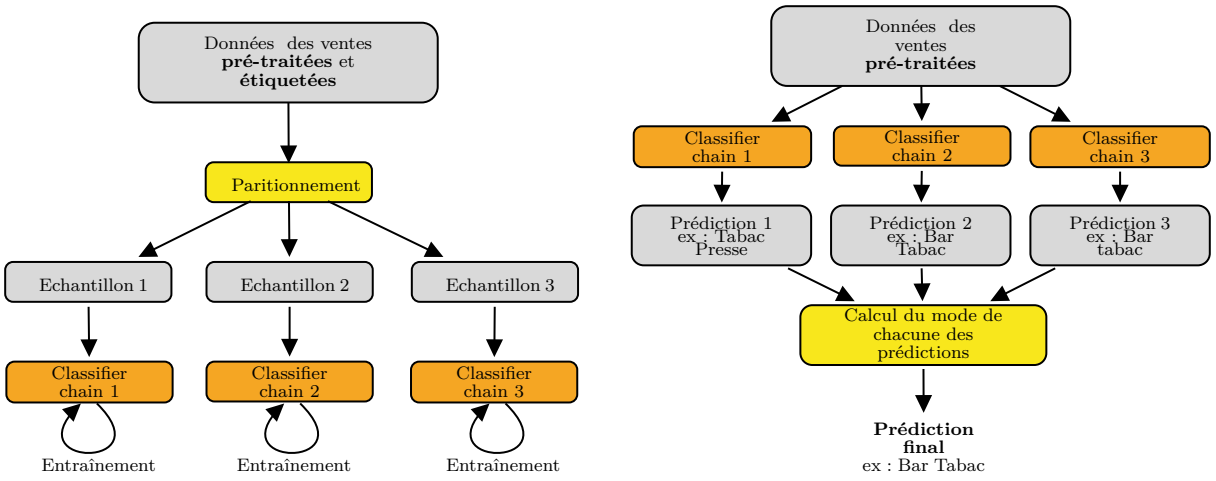
Hierarchy of Multi-label Classifiers (HOMER). *HOMER* [Tsoumakas, 2008] est une approche dérivée de l'approche *Label Powerset*. En se basant sur une approche « diviser pour

régner », cette méthode construit une hiérarchie d'étiquettes afin de réduire progressivement le nombre de combinaisons possibles. Cette méthode est particulièrement utile si le jeu de données contient un grand nombre de combinaisons d'étiquettes. Cependant, elle est moins efficace dans le cas contraire.

4.1.3 Ensembles

Il existe de nombreuses approches d'étiquetage multiple qui mettent en œuvre un apprentissage ensembliste [Bogatinovski, 2022] et combinent ainsi plusieurs modèles. Ces modèles sous-jacents peuvent eux-mêmes permettre un étiquetage multiple ou non.

Ensemble of Chain Classifiers (ECC) est une méthode qui utilise un ensemble de *Chain Classifiers (CC)*, eux-mêmes basés sur le chaînage de plusieurs classifieurs de type *Binary Relevance*. À chaque phase, un échantillon différent avec remplacement des instances contenues dans chaque échantillon est utilisé [Read, 2011]. Un exemple permettant d'illustrer son fonctionnement est présent en figure 2.15. Le principal avantage de cette méthode par rapport à celles précédemment mentionnées est qu'elle permet de traiter plus facilement de plus grands jeux de données, étant donné que l'apprentissage est réalisé sur des échantillons et non sur le jeu de données complet.



(a) Méthode ECC en phase d'entraînement (b) Méthode ECC en phase de prédiction

FIGURE 2.15 – Étiquetage multiple par *Ensemble of Classifier Chains (ECC)*

4.2 Adaptation d'algorithmes

Dans cette section, nous présentons l'une des deux grandes approches pour adapter des algorithmes de classification classique (où une seule étiquette est attribuée par instance) en algorithmes d'étiquetage multiple. Ces deux approches se distinguent par les familles d'algorithmes qu'elles exploitent : soit les algorithmes singuliers (appelés *Singleton algorithms*), soit les algorithmes basés sur des ensembles de modèles (*ensemble methods*).

Random Forest of Predictive Clustering Trees (RFPCT) [Kocev, 2013] est une méthode qui utilise des forêts d'arbres de décision avec des arbres de prédiction par partitionnement (*Predictive Clustering Trees* ou PCT) comme modèles d'apprentissage de base. Ces arbres de décision prennent en entrée des données partitionnées de manière hiérarchique. Pour obtenir cette structure d'entrée, les données sont partitionnées de manière récursive dans le but de réduire la variance, calculée en sommant les indices de Gini des étiquettes. La fonction prototype retourne un vecteur avec les probabilités qu'un individu soit étiqueté avec différentes combinaisons d'étiquettes. Un critère nommé *F-test* permet de limiter la complexification des arbres. Dans cette méthode, l'induction des arbres de décision se fait de haut en bas (*Top-Down Induction of Decision Trees*, TDITD en anglais).

Cette méthode est rapide et prend en compte les corrélations entre étiquettes, mais elle n'est pas adaptée en cas de matrices de variables creuses. En effet, le tirage aléatoire sur une matrice creuse pour la construction de l'arbre risque de mener à de faibles performances. Cependant, elle reste la méthode la plus performante rapportée dans les travaux de Bogatinovski et al. [Bogatinovski, 2022] et Madjarov et al. [Madjarov, 2012].

4.3 Synthèse des travaux connexes sur l'étiquetage multiple

Pour conclure cette section, nous pouvons constater qu'il existe de nombreux types d'approches pour répondre aux problématiques d'étiquetage multiple. Chaque typologie d'approches présentée dans cette section propose une multitude de techniques, parfois complexes. Cependant, un point important à retenir est que seule une faible proportion de ces approches est accompagnée d'outils accessibles, rendant ces méthodes moins abordables pour les chercheurs qui ne sont pas spécialisés dans ce domaine. Comme nous le verrons plus tard, la notoriété, la facilité d'accès et la pérennité d'un modèle d'apprentissage automatique jouent un rôle majeur dans sa sélection pour un usage dans un contexte industriel.

5 Bancs d'essai adapté au contexte scientifique ou industriel

En informatique, les bancs d'essai, également appelés *benchmarks*, jouent un rôle fondamental dans l'évaluation des performances et la comparaison de différentes solutions face à une problématique donnée. Ils constituent une référence objective, standardisée et reproductible, permettant aux chercheurs et aux professionnels de mesurer la pertinence et l'efficacité de leurs approches. Grâce à ces bancs d'essai, il devient possible d'évaluer la robustesse des algorithmes, leur capacité à résoudre un problème spécifique et leur adaptabilité à des contextes variés, tout en facilitant la comparaison directe avec d'autres méthodes.

Dans notre cas, nous sommes à la recherche d'un banc d'essai qui s'inscrit dans le cadre d'une problématique de classification, dans un contexte à la fois scientifique et industriel, semblable à celui étudié dans cette thèse. Un tel banc d'essai nous permettrait de tester différentes approches de manière rigoureuse et de valider les choix méthodologiques retenus.

Plus précisément, le banc d'essai que nous cherchons doit répondre à certaines caractéristiques essentielles :

— un contexte de données de ventes hétérogènes en termes de qualité de nommage des produits. Par exemple, un banc d'essai basé sur les données issues d'une place de marché en ligne où les utilisateurs sont libres de nommer les produits comme ils le souhaitent, ce qui entraîne des incohérences et des variations dans les libellés ;

— un contexte impliquant des données textuelles courtes et à haute cardinalité, présentant des synonymes et des variations morphologiques multiples, rendant les techniques classiques d'encodage inefficaces et nécessitant des approches plus sophistiquées.

L'objectif de cette démarche est de trouver une base de comparaison pertinente pour mesurer les performances de nos méthodes de classification dans un environnement de données complexes, semblable à celui dans lequel cette thèse se déroule.

5.1 Critères de sélection des approches

L'objectif principal de ce travail est de sélectionner un banc d'essai adapté pour évaluer les performances des modèles de classification des commerces à partir de leurs ventes. Actuellement, bien que les informations sur les activités des commerces soient disponibles, elles sont considérées comme incomplètes, obsolètes et peu fiables, car elles sont figées au moment de l'inscription d'un nouveau client à la solution Bimedia. Pourtant, les activités des commerces évoluent considérablement au fil du temps, rendant ces informations critiques.

Un banc d'essai pertinent doit permettre une analyse rigoureuse et précise des performances des modèles de classification des commerces, en tenant compte de la diversité et de la complexité des données propres au secteur du commerce de détail. Il doit offrir un cadre d'évaluation réaliste, reflétant les défis concrets posés par l'hétérogénéité des libellés de produits et des familles de produits, rédigés majoritairement en français. Cette diversité des libellés constitue un enjeu majeur dans le contexte industriel qui nous concerne.

Les métriques d'évaluation devront permettre de quantifier objectivement les performances des modèles. Elles devront également tenir compte d'un jeu de données fortement déséquilibré, car, dans notre contexte, les typologies d'activités telles que les bureaux de tabac et les magasins de presse sont largement majoritaires par rapport à d'autres types de commerces.

Atteindre cet objectif impliquera une exploration approfondie des bancs d'essai existants. Si aucune solution ne s'avère satisfaisante, nous envisagerons le développement d'un banc d'essai sur mesure, spécifiquement conçu pour répondre aux exigences et aux particularités de notre problématique.

5.2 État des lieux des bancs d'essai existants

Pour évaluer les méthodes de classification dans un contexte de données présentant une grande hétérogénéité, des synonymes, des homonymes et diverses morphologies, nous n'avons pas identifié de banc d'essai spécifiquement conçu avec toutes ces caractéristiques. Cela s'applique également à notre recherche d'un banc d'essai reconnu pour l'évaluation de la classification de produits dans le domaine du commerce de détail, qui se distingue du commerce de distribution.

Cependant, au cours de nos recherches, nous avons identifié des bancs d'essai pertinents pour des problématiques inhérentes à la nôtre, telles que l'encodage de variables qualitatives à haute cardinalité. Parmi ces contributions, nous pouvons notamment citer le travail présenté par Pargent et al. [Pargent, 2022].

Dans notre contexte, nous faisons face à cette problématique lorsque nous traitons les données de vente. En effet, les variables qualitatives permettant de définir les produits dans ces données incluent les libellés de produit et de famille de produits. Ces variables sont de très haute cardinalité, avec de nombreuses valeurs différentes présentes dans notre jeu de données. Elles jouent un rôle crucial dans la modélisation nécessaire à la classification des commerces.

Pargent et al. [Pargent, 2022] explorent des techniques d'encodage efficaces pour les variables catégorielles à haute cardinalité, couramment utilisées dans l'analyse de données pour les algorithmes de machine learning. L'étude se concentre sur l'impact de ces méthodes sur la performance prédictive des algorithmes, et vise à établir des bonnes pratiques pour choisir la meilleure technique. Un vaste banc d'essai a été élaboré, comparant différentes stratégies d'encodage avec cinq algorithmes de ML. Les résultats montrent que les versions régularisées de l'encodage par cible (*target encoding*) sont les plus performantes, surpassant les méthodes plus classiques telles que l'encodage entier ou l'encodage binaire (*one-hot encoding*).

Une autre publication intéressante autour du même sujet [Cerda, 2020] a également été identifiée et est abordée plus en détails dans la section 2.

Globalement, bien que ces publications soient pertinentes, elles ne considèrent pas notre problématique dans son ensemble et ne répondent pas pleinement à nos critères de recherche.

5.3 Conclusion sur l'absence de banc d'essai pertinent

À travers cette étude, nous avons mis en évidence l'absence d'un banc d'essai spécifiquement conçu pour répondre aux besoins de la classification des commerces dans le secteur du commerce de détail. Bien que des benchmarks existent pour l'étiquetage multiple, ainsi que pour l'incorporation et l'encodage des données, ces approches ne permettent pas de mesurer efficacement la performance globale des modèles dans un environnement où plusieurs traitements doivent être combinés. En effet, une évaluation séparée de ces composants ne reflète pas fidèlement les résultats lorsque ces traitements interagissent dans des scénarios réels.

De plus, les jeux de données actuels, couvrant souvent des domaines variés, ne garantissent pas des performances optimales pour des données spécifiques telles que celles de Bimedia. Ces données, principalement des libellés courts et en français, sont caractérisées par des homonymes, des synonymes, ainsi que des erreurs d'orthographe ou de saisie, et présentent un champ lexical relativement restreint (lié aux produits) par rapport à l'ensemble du langage. Ces particularités, propres au secteur du commerce de détail, rendent difficile l'évaluation fiable des algorithmes avec les benchmarks existants.

L'absence d'un banc d'essai adapté limite donc la comparaison objective des algorithmes de classification, et restreint la capacité à identifier des améliorations pertinentes. Il est crucial de développer un banc d'essai sur mesure qui prenne en compte non seulement la diversité des données, mais aussi les spécificités des catalogues de produits.

En définitive, la création d'un tel banc d'essai permettrait une évaluation rigoureuse et transparente des performances des algorithmes, tout en garantissant une meilleure prise en charge des données spécifiques à Orisha Retail Shops. Cela offrirait une classification plus fine et adaptée aux besoins stratégiques et commerciaux de l'entreprise.

Chapitre 3

Banc d'essai pour la classification des commerces

Objectifs

L'objectif de ce chapitre est de mettre en place un banc d'essai, afin d'évaluer les différentes approches de classification des commerces par activités commerciales chez Orisha Retail Shops. En réponse aux défis posés par la diversité des catalogues de produits et l'évolution constante des activités commerciales, ce banc d'essai doit offrir un environnement rigoureux pour tester et comparer efficacement les solutions de classification. Le chapitre se concentre sur la définition précise des jeux de données, des critères de performance, et des protocoles expérimentaux, garantissant ainsi des résultats fiables et reproductibles. Ce banc d'essai constitue également une base solide pour l'amélioration des approches actuelles et l'intégration de nouvelles solutions répondant aux besoins stratégiques d'Orisha Retail Shops.

Sommaire

1 Introduction

Orisha Retail Shops fournit sa solution Bimedia a des commerces couvrant une large gamme d'activités. Actuellement, les données générées par ces commerces sont utilisées pour des tableaux de bord et des analyses ad hoc, mais elles pourraient permettre une variété plus large d'applications, telles que la prévision des ventes, l'optimisation des stocks, la recommandation de produits ou la caractérisation des activités des commerces.

Historiquement, l'enregistrement des activités des commerces, telles que « épicerie » ou « tabac », était effectué par le personnel commercial d'Orisha Retail Shops lors de la configuration initiale des contrats avec les clients. Cette démarche visait initialement à personnaliser la solution offerte à chaque commerce en fonction de ses activités spécifiques.

Avec le temps, le besoin de caractérisation des activités des commerces s'est accru, notamment avec le développement des activités liées à la régie publicitaire d'Orisha Retail Shops, la filiale Bimedia Adgency. Cette information peut notamment être utilisée lors de la valorisation des espaces publicitaires. Par exemple, elle permet un ciblage précis lors de la diffusion de publicités relatives à une activité spécifique, comme la vente de boissons non alcoolisées ou de confiseries. En diffusant les publicités uniquement dans les commerces pertinents, l'efficacité des campagnes publicitaires s'en trouve améliorée, ce qui contribue également à accroître la satisfaction des annonceurs.

D'autres études, telles que des analyses de corrélations entre type d'activité et position géographique, seraient particulièrement valorisables pour l'entreprise. Par exemple, dans une étude récente réalisée en interne, l'entreprise souhaitait dresser le profil type d'un point de vente performant pour les ventes de services de transfert d'argent. L'étude de corrélation entre un type d'activité particulier et la vente de ce type de service aurait pu fournir des données intéressantes à étudier. Elle n'est actuellement pas disponible de manière fiable et à jour, notamment parce qu'elle est complexe à calculer automatiquement. De plus, les points de vente tendent à diversifier leurs activités entre le moment de leur enregistrement dans le référentiel et le moment où l'information est utilisée.

Au cœur de cette problématique de classification des commerces par activité commerciale se trouve un autre défi majeur : la compréhension des catalogues de produits des commerces clients, ou plus généralement des ventes réalisées. En raison de la manière disparate dont ces catalogues sont constitués par les gérants des commerces, ainsi que de la diversité des activités de ces derniers, l'alignement, la compréhension et, plus généralement, le traitement automatisé des catalogues de produits sont des tâches complexes.

Face à ces défis, plusieurs approches sont envisageables, dont deux principales.

— Une campagne d'étiquetage manuelle des activités des commerces par les employés du service commercial de l'entreprise. Bien que relativement précise, cette méthode pourrait présenter un problème de subjectivité, chaque commercial pouvant avoir une perception différente des activités commerciales. Par exemple, dans un secteur où les services de transfert d'argent sont souvent vendus, un commercial pourrait ne pas caractériser un commerce réalisant « seulement » 1 000 euros par semaine, alors qu'un autre commercial considérerait

ce même commerce comme un client important pour cette typologie de produit. De plus, cette solution serait extrêmement coûteuse et chronophage, sans pour autant résoudre la problématique de disposer d'information à jour.

— L'entraînement de modèles d'apprentissage automatique entièrement autonomes, qui, bien que potentiellement plus évolutifs, pourraient nécessiter des ajustements fins pour atteindre une performance acceptable.

Cependant, il est difficile de choisir une solution efficace pour résoudre ce manque d'information précise et à jour sur les activités des commerces clients de la solution Orisha Retail Shops sans un cadre expérimental rigoureux. Ce cadre doit définir précisément les données à utiliser, les métriques à relever, ainsi que les contraintes à respecter pour garantir des résultats fiables et comparables.

C'est donc ce que nous avons souhaité réaliser à cette étape du projet de thèse : définir clairement la problématique et établir un cadre expérimental précis afin de pouvoir ensuite effectuer des protocoles expérimentaux comparables, permettant de choisir la solution la plus adéquate, autrement dit, le meilleur compromis entre la qualité, le coût et le délai (en référence au triangle QCD, ou triangle d'or, un élément essentiel dans la gestion de projets dans un contexte industriel).

Ce cadre expérimental, communément appelé *banc d'essai*, est essentiel pour valider et comparer les différentes approches. Comme nous l'avons montré dans le chapitre précédent, nous n'avons identifié aucun banc d'essai existant pertinent pour notre contexte et couvrant intégralement notre problématique. Par conséquent, nous devons en constituer un de toutes pièces. Cela implique de définir les jeux de données pertinents, les critères de performance et les procédures expérimentales standards.

Ce chapitre est structuré pour fournir une vue d'ensemble complète de la conception et de l'évaluation d'un banc d'essai dédié à la classification des commerces par activités commerciales. Dans ce qui suit, nous explorerons nos problématiques ainsi que les principaux intérêts et objectifs clés associés à la création d'un banc d'essai personnalisé (section 2). Ensuite, nous présentons les données à notre disposition (section 3). Puis, les critères d'évaluation des performances seront discutés (section 4). Nous terminerons par l'évaluation de l'approche initiale de classification des commerces (section 5). Enfin, le chapitre se conclura par un récapitulatif des conclusions et des perspectives pour les travaux futurs (section 6).

2 Problématique, intérêts et objectifs

Dans cette section, nous analysons en profondeur la problématique à laquelle Orisha Retail Shops est confrontée, en nous appuyant sur des illustrations issues de données réelles pour contextualiser les défis rencontrés. Nous expliquons ensuite les avantages pour l'entreprise de résoudre cette problématique, ainsi que les objectifs clés liés aux résultats attendus.

2.1 Problématique

La problématique que nous abordons est l'identification des activités des commerces à partir de leurs ventes. D'un point de vue scientifique, il s'agit d'un problème de classification des commerces de détail à partir d'un jeu de données réel, en se basant sur les ventes réalisées. Bien que la classification soit un problème bien connu en apprentissage automatique, nous montrons dans ce chapitre que les solutions classiques sont inefficaces pour classer précisément les commerces dans notre contexte industriel. Cela est principalement dû à la qualité hétérogène des données. De notre expérience, cela est souvent le cas avec de nombreux jeux de données issus du monde réel, où le travail de préparation des données est généralement plus long et complexe que les étapes de traitement, qui sont souvent plus standardisées.

Dans le réseau de commerces utilisant la solution Bimedia, les propriétaires gèrent leurs catalogues de produits avec une grande flexibilité, ce qui entraîne des divergences dans la manière dont les produits sont regroupés. Comme détaillé dans la section 3, ces produits sont organisés en familles de produits. Il existe deux typologies : les familles globales, gérées centralement par Orisha Retail Shops et communes à tous les commerces, et les familles locales, propres à chaque commerce.

L'objectif est de classer les commerces selon leurs activités à partir de leurs ventes, ce qui nécessite une classification ou un encodage des produits pour identifier les similarités et les différences entre les catalogues. Les familles globales sont relativement simples à traiter grâce à leur structure centralisée et homogène. Certaines activités, comme les bureaux de tabac ou les magasins de presse, sont ainsi plus faciles à identifier, car les ventes associées sont facilement reconnaissables. Toutefois, plus d'un tiers des commerces offrent des produits et services dans d'autres secteurs, comme les hôtels, restaurants, boulangeries, épiceries, fleuristes, bars, etc. Ces produits appartiennent aux familles locales, rendant leur gestion plus complexe en raison des variations dans les noms des produits (synonymes, acronymes, fautes d'orthographe), des codifications (codes-barres normés, codes locaux, erreurs de saisie) et des arrangements en familles (certains commerces classent les produits par marque, d'autres par type avec différents niveaux de granularité). Comparer les produits entre commerces et identifier les activités devient alors une tâche ardue sans transformations complexes des données.

La principale difficulté réside dans la classification et l'encodage des produits. Les commerces vendent environ deux millions de produits différents, organisés en familles qui varient significativement d'un commerce à l'autre, tant sémantiquement que syntaxiquement. Ces produits sont vendus au sein des 6 500 commerces équipés de la solution Bimedia, avec des chiffres d'affaires et des quantités de vente variables. Comme nous l'avons montré dans notre revue de l'état de l'art, ce problème d'encodage et de classification des produits n'est pas propre à Orisha Retail Shops. D'autres industries, telles que l'automobile ou le commerce de détail, rencontrent également des défis similaires lorsqu'elles permettent à leurs clients de structurer librement leurs catalogues tout en cherchant à identifier les activités principales de leurs utilisateurs.

Nous illustrons ce problème à travers un exemple simple dans le tableau 3.1, qui présente des données réelles issues du jeu de données. Les onze premières lignes se réfèrent au même produit : une portion de frites, mais exprimée sous onze libellés différents. De plus, ces produits sont

TABLE 3.1 – Extrait du jeu de données des ventes avec les noms de produits et leur famille au sein de différents commerces (les colonnes inutiles ne sont pas affichées)

	Store ID	**Libellé produit**	**Libellé famille produit**
1	ccb...2d6	Barquette de Frite	Restauration sur place
2	969...8c8	BARQUETTE DE FRITES	Boissons emportees 10
3	6c5...714	Petite frite	BRASSERIE
4	3e1...cf7	grande frite	Restauration a emporter 10 - 707140
5	609...ba4	BARQUETTE FRITE	ALIM5.5
6	db2...5c2	Bol De Frites	Traiteur
7	be9...702	FRITE	PLAT
8	379...949	ASSIETTE FRITE SEULE	BRASSERIE
9	379...949	MOYEN. BARQUETTE FRITE EMPORTER	Restauration a emporter
10	aa1...590	Frites Barquette	Snack
11	bc3...3d3	BARQUETTE DE FRITE	RESTAURATION 10
12	ab1...8ef	NOODLE FRITE	Tabletterie
13	8ab...c19	FRITE PIK	Confiserie 20

organisés dans diverses familles selon les commerces ou même selon le mode de consommation (sur place ou à emporter, comme le montrent les lignes 8 et 9). Comme illustré, il est impossible de se fier aux valeurs brutes de ces variables. Dans de nombreux cas, un identifiant de produit (par exemple, un code-barres) ne peut pas être utilisé pour effectuer des correspondances entre les commerces, car cet identifiant est parfois généré localement, comme pour le produit considéré ici. Une étape de transformation des données est donc nécessaire.

Le cas exposé dans cet exemple avec une portion de frites se retrouve pour de nombreux autres produits, ce qui rend la classification des activités commerciales des commerces particulièrement complexe.

Le problème de classification des commerces peut être décomposé en trois sous-problèmes principaux. Premièrement, nous devons traiter l'encodage de variables catégorielles à haute cardinalité, car les données de ventes sont structurées en catégories avec une hiérarchie à deux niveaux (produits et familles). Ces catégories sont majoritairement non partagées entre commerces et comportent des valeurs textuelles imparfaites, comme illustré dans le tableau 3.1. Deuxièmement, il s'agit d'un problème lié à la gestion de formes d'entrée variables. Les données brutes de ventes, utilisées comme variables d'entrée pour chaque commerce, ont des longueurs aléatoires et ne sont pas séquentielles. Par exemple, un commerce peut vendre seulement 50 produits, tandis qu'un autre peut en vendre plus de 1000. Les données d'entrée, si elles sont utilisées brutes sans prétraitement, sont donc de tailles variables, rendant leur transformation complexe. Troisièmement, c'est un problème de classification à étiquetage multiple, car une ou plusieurs activités commerciales peuvent être assignées aux commerces en fonction de leurs ventes. Ces trois sous-problèmes ont été abordés plus en détail dans le chapitre précédent.

Le tableau 3.2 illustre un extrait de données brutes extraites des transactions pour trois commerces que nous souhaitons classer, ainsi que le résultat attendu. Dans cet échantillon, nous pouvons observer que les commerces vendent un nombre varié de produits, ce qui se traduit par un nombre différent de lignes par commerce.

En conclusion de ces deux exemples illustrés dans les tableaux 3.1 et 3.2, la partie la plus difficile dans la problématique de caractérisation des activités commerciales des commerces n'est pas d'assigner des étiquettes aux commerces, mais d'identifier les relations d'équivalence entre les

TABLE 3.2 – Exemple de jeu de données des ventes avec prédictions des activités ciblées (les colonnes non pertinentes ne sont pas affichées, les informations non pertinentes sont tronquées)

store ID	Code Barre	Libellé produit	Libellé famille de produit	Quantité	Activités (à prédire)
db...5c2	31...01	Pain Au Mais	Pain	4	Boulangerie
	31...02	Pain	Pain	2338	
	31...04	Baguette	Pain	13377	
	31...51	Cookie	Viennoiserie	2378	
e9...f43	00...03	Tourte aux alouettes	Boulangerie	6315	Boulangerie
	00...19	Tradition	Boulangerie	135445	
	31...07	Galette creme amande...	Patisserie	3	
	31...38	BROWNIE	Patisserie	1070	
bb...49f	04...94	ESSENCE ZIPPO	Autres articles fumeurs	40	Tabac café
	11...10	Clearomiser Q16 PRO	Divers 20.	3	
	31...45	CAFE	CHAUD	83253	
	_AL...36	PHILIP MORRIS 20	Cigarettes	26125	

produits de différents commerces. En effet, un processus d'étiquetage multiple basé sur des modèles robustes devrait fournir des performances acceptables uniquement si les ventes de produits sont correctement transformées. Cela signifie que les produits équivalents entre les commerces doivent être identifiés lors de l'étape de transformation, en amont de la phase d'étiquetage multiple des activités commerciales des commerces. La transformation nécessaire pour assurer un traitement de qualité inclut donc l'encodage approprié des valeurs textuelles (libellés de produits et de familles) et l'intégration des ventes de produits.

Dans ce chapitre, nous considérons le problème dans son ensemble. Nous ne dissocions pas la tâche de transformation des libellés de produits, l'intégration des ventes des différents produits et familles de produits, de la tâche d'étiquetage multiple, pour deux raisons.

— Nous ne sommes pas en mesure de produire un ensemble de validation cohérent pour les produits (il y en a environ deux millions), alors que cela est faisable pour les activités des commerces (avec 6 500 commerces et neuf activités différentes). En effet, pour évaluer un modèle, il est nécessaire de disposer d'un jeu de données de validation suffisamment représentatif de l'ensemble des données. Or, pour évaluer la qualité de l'étiquetage des produits, un jeu de validation de 10% du jeu de données total représenterait environ 200 000 produits à étiqueter manuellement, ce qui est irréalisable avec les ressources humaines disponibles.

— Nous sommes convaincus, par expérience avec des tentatives d'incorporation basées sur des modèles tels que Word2Vec, Glove, FastText, BERT, CamemBERT, et ELMo, que les modèles généralistes d'incorporation de valeurs textuelles ne suffisent pas comme prétraitement pour l'entraînement de modèles d'étiquetage multiple dans le cadre de la caractérisation de l'activité des commerces. Cela est dû à l'hétérogénéité de la qualité des libellés, à la langue utilisée (français), à la petite taille des valeurs textuelles (seulement des libellés, pas de phrases), et à leur champ lexical relativement restreint.

Par conséquent, la tâche d'étiquetage multiple est incluse dans cette étude principalement comme moyen d'évaluer la qualité de l'étape de transformation des données. C'est dans cette étape de transformation que réside la complexité majeure, et l'étiquetage multiple des activités commerciales des commerces est l'une des seules tâches que nous pouvons évaluer avec un jeu

de données de validation conséquent.

En somme, le problème de classification des commerces par activité présente une véritable complexité en raison de la très haute cardinalité des libellés des produits et des familles de produits. La première étape consiste à définir clairement la problématique et les objectifs de classification attendus, ainsi qu'à mesurer les performances des différentes solutions. Autrement dit, il est nécessaire de définir un banc d'essai adapté à cette problématique.

2.2 Intérêts et objectifs clés

Un banc d'essai permet de quantifier les résultats des méthodes existantes. En mesurant leur performance à l'aide de métriques précises, il est possible d'évaluer objectivement l'efficacité des solutions en place. Cette quantification permet d'identifier les forces et faiblesses des approches actuelles, de détecter les lacunes, et d'estimer leur impact sur les objectifs commerciaux de l'entreprise. Avant le projet de thèse, l'entreprise utilisait une approche d'apprentissage automatique basée sur le clustering. Celle-ci semblait produire des résultats insuffisants et n'était donc pas exploitée. Cependant, en l'absence de cadre expérimental normalisé, cette solution n'avait pas pu être évaluée autrement que de manière qualitative et subjective.

De plus, le banc d'essai facilite la comparaison rigoureuse des approches existantes avec de nouvelles méthodes. Cela permet de tester des solutions basées sur des approches innovantes et de valider les hypothèses concernant leur efficacité. En identifiant les configurations optimales des solutions, il devient possible d'optimiser les performances des modèles développés. La reproductibilité des résultats est également assurée, renforçant ainsi la crédibilité des conclusions tirées des expérimentations. Ce banc d'essai s'est avéré particulièrement utile lors de la comparaison de différentes approches, notamment dans le cadre du développement de la solution ThesaurusBT, abordée dans le chapitre suivant.

La standardisation des procédures expérimentales constitue un autre avantage majeur du banc d'essai. En définissant des protocoles précis et reproductibles, nous garantissons que les expérimentations sont menées de manière cohérente. Cette approche facilite également la collaboration entre les chercheurs et accélère le développement de nouvelles approches en fournissant un cadre méthodologique bien défini.

Enfin, un banc d'essai personnalisé favorise l'innovation et le partage des connaissances. En publiant les résultats et les méthodologies utilisées, nous encourageons l'innovation ouverte et renforçons la position de l'entreprise dans son domaine. Les retours de la communauté scientifique permettent d'améliorer continuellement les méthodes, garantissant qu'elles restent à la pointe de la technologie.

En résumé, la conception d'un banc d'essai personnalisé est une démarche stratégique pour Orisha Retail Shops. Elle permet de quantifier et comparer les approches, de standardiser les expérimentations et de promouvoir l'innovation. Ce processus garantit la pertinence et l'efficacité des solutions face aux défis complexes de classification et d'analyse de données dans un environnement commercial dynamique.

3 Contexte des données d'entrée, échantillonnage et statistiques

L'une des étapes clés dans la conception d'un banc d'essai est la description et l'explication des données. C'est ce que nous proposons dans cette section pour le jeu de données des ventes des commerces équipés de la solution Bimedia.

3.1 Données

Le banc d'essai que nous proposons repose principalement sur deux jeux de données fournis par l'entreprise Orisha Retail Shops.

Le jeu de données des ventes contient les ventes agrégées par produit et par commerce pour 2 325 commerces anonymisés (sur plus de 6 500 commerces clients) sur une période non divulguée d'une durée de 12 mois. Cette anonymisation a été motivée par des impératifs de confidentialité en vue de sa diffusion. Dans ce jeu de données, sept variables sont observées : un identifiant de commerce, un code-barres de produit, un libellé de produit, un identifiant de famille, un libellé de famille, le montant total vendu pendant l'année, ainsi que le taux de TVA. Un extrait de ce jeu de données est visible dans le tableau 3.3.

Table 3.3 – Extrait du jeu de données des ventes par produit et par commerces

store_id	barcode	product_label	family_label	family_id	quantity	vat
96a...3f2	888010664581	COUP_DOUBLE	Jeux	2110	1.0	0.0
561...3bf	3296992852063	BOUILLETTES VANILLE-FRAISE 18mm 1kg	PECHE	76	1.0	20.0
a36...6ca	P00105000510000	EQUIPE no 210528	Presse	100	6.0	0.0
00e...3a0	0000030200006	COTON BRANCHE	Fleurs Piece	30200	15.0	10.0
243...2ff	P05280000001000	+COLL BOOSTER POKEMO no 1	Hors Presse 20	9103	18.0	20.0
6f2...693	9782253087663	VERNON SUBUTEX (TOME 1)	Librairie 5.50%	81	3.0	5.5
b96...ba4	P00123000227000	MG PELE MELE MEGA no 177	Presse	200	2.0	0.0
12d...b12	ITUNE050	iTunes 50 €	Itunes	9852	19.0	0.0
a54...44e	_ALTA_61832	MAYA ORIGINAL SPIRIT 100% TABAC EN 20	Cigarette	1101	476.0	0.0
ebc...803	648053833804	JEU_CASH	Jeux	2110	61.0	0.0
d6c...1ab	8719964026538	MY BLU STARTER	E CIGARETTE	63	2.0	20.0
140...9dd	9782401063556	LES MILLE ET UNE NUITS OEUVRES & THEMES HATIER	LIBRAIRIE	104	1.0	5.5
b18...b81	P14433000123000	ECHOS no 220802	Presse	100	1.0	0.0
f0e...0cb	3020122350293	BRISTOL 125X200 ASSORTI	Papeterie	32	1.0	20.0
af0...ba7	_ALTA_85613	WINSTON CIGARILLOS EN 10	Cigares	1102	61.0	0.0
bf6...45f	9782733847343	THEO ET LE MONSTRE DE L'EAU (COLL. MON PREMIER...	Librairie 5.5	40	1.0	5.5
8e2...de1	_ALTA_60734	CHE ESSENTIAL EN 20	Cigarette	1101	778.0	0.0
150...90c	P15667004390000	TELE Z TNT no 2023	Presse	200	1.0	0.0
7e8...13a	3038354190402	Sauce Tomate Basilic Panzani 210 gr	Alimentation 5.5	9	3.0	5.5

Les sept variables qui composent ce jeu de données sont des attributs issus de quatre classes d'objets distinctes : les commerces, les produits, les familles de produits, et les activités commerciales. L'organisation de ces attributs et classes est illustrée dans la figure 3.1. La classe *Activity* (activités en français), colorée en jaune, n'est pas fournie dans le jeu de données initial ; elle correspond à la sortie attendue de la classification. La classe *Sales* n'est également pas explicitement présente dans le jeu de données original, mais elle représente le volume des ventes (quantités vendues) d'un produit pour un commerce donné sur la période considérée.

Chaque commerce peut vendre plusieurs produits, qui sont eux-mêmes classés dans des familles. Dans le système, un produit est identifié par son code-barres (qui peut être généré si le produit n'en possède pas). Un commerce ne peut pas avoir plusieurs produits avec le même code-barres. Un produit ne peut être classé que dans une seule famille de produits par commerce

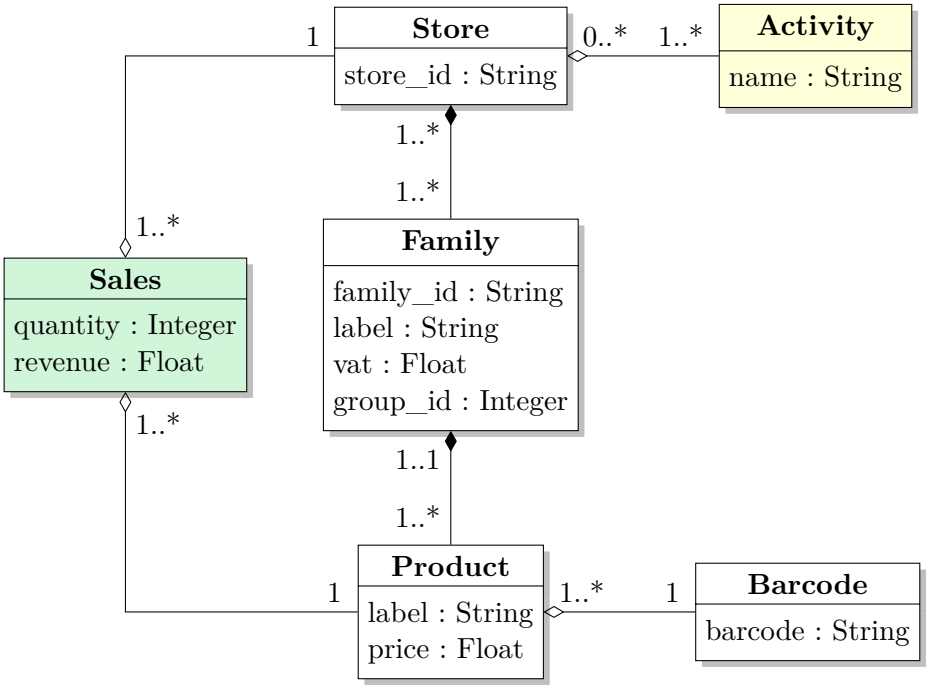


FIGURE 3.1 – Diagramme UML des éléments du jeu de données

(un commerçant ne peut donc pas placer un produit dans plusieurs familles à la fois dans son système). Une famille de produits regroupe un ou plusieurs produits spécifiques. Comme exprimé dans la section 3.2.2, il existe deux types de familles : les familles fixes et non-fixes.

Le jeu de données des commerces étiquetés est composé de 400 commerces dont les activités ont été annotées par des experts. Le processus d'annotation était le suivant : sélectionner aléatoirement un commerce du jeu de données des ventes et annoter l'activité de ce commerce en considérant les ventes agrégées par famille de produits contenues dans le premier ensemble de données, ainsi que la page Google Maps du commerce. Cet ensemble de données contient deux variables : un identifiant de commerce et des étiquettes d'activités. Il existe 12 étiquettes d'activités possibles, dont seules neuf sont présentes dans le jeu de données de validation (les autres étant anecdotiques). Cet ensemble de données présente 29 combinaisons uniques de ces neuf étiquettes. Un extrait de ce jeu de données est visible dans le tableau 3.4.

3.2 Contexte et explications des données

Dans cette section, nous détaillons les différentes variables présentes dans le jeu de données, en expliquant leur origine, leur utilisation, ainsi que leurs spécificités. Chaque variable joue un rôle crucial dans l'analyse, et leur compréhension est essentielle pour une exploitation efficace des données. Nous mettrons en lumière la manière dont elles sont collectées, les contextes dans lesquels elles sont utilisées, et les particularités qu'elles présentent, telles que leur type, leur granularité ou encore leur niveau de qualité.

TABLE 3.4 – Jeu de données des commerces avec leurs activités commerciales étiquetées manuellement

Store ID	Types
b25...d17	[tabac, presse]
bb8...abb	[bar, tabac, presse, cafe]
7b8...173	[tabac, presse, bar, cafe]
e8b...0e2	[presse]
1ec...529	[tabac, presse]
240...281	[tabac, presse, cafe]
43a...b33	[tabac, presse]
139...60a	[tabac, presse]
5fd...6a0	[tabac, presse]
9a8...8d6	[tabac, presse]

3.2.1 Produits

Les produits correspondent aux articles vendus par un sous-ensemble de 2325 commerces sur 12 mois. Le tableau 3.5 répertorie les attributs associés à chaque produit.

Attributs	Description	Source
code-barres	identifiant d'un produit	scanné sur le produit ou saisi par le client ou généré par Bimedia
prix	prix du produit (TTC)	saisi par le client ou fourni par Bimedia
libellé produit	nom du produit	saisi par le client ou fourni par Bimedia
libellé de famille	nom de la famille de produits	saisi par le client ou fourni par Bimedia
TVA	taxe sur la valeur ajoutée	saisi par le client ou fourni par Bimedia
identifiant commerce client	identifiant unique du commerce	fourni par Bimedia

TABLE 3.5 – Description des attributs des produits

Chaque commerce identifie les produits par des **codes-barres produits**. De nombreux formats différents de codes-barres peuvent être observés dans le jeu de données, résultant de l'existence de divers standards (EAN-8, EAN-12, EAN-13, EAN-18, etc.) et des différentes méthodes utilisées pour ajouter des codes-barres personnalisés dans les systèmes de caisses enregistreuses (codification interne, erreurs de frappe, inventions, etc.). Les gérants de commerces peuvent ainsi ajouter des codes-barres personnalisés à leur inventaire via la caisse enregistreuse de plusieurs manières :

— scanner le code-barres imprimé sur l'étiquette du produit ;

— saisir manuellement le code-barres imprimé sur l'étiquette du produit ;

— saisir manuellement un code-barres inventé ;

— laisser le champ du code-barres vide, permettant à la caisse enregistreuse de générer un nouveau code-barres.

Dans le cas des codes-barres générés, l'entreprise utilise plusieurs règles pour leur création, ce qui aboutit à plusieurs formats de codes-barres. Prendre en compte ces règles peut être crucial lors du prétraitement des données, afin de maximiser l'identification correcte des différents produits, en particulier pour les solutions de classification des activités commerciales basées sur les produits. En effet, en raison des règles de génération de codes-barres de Bimedia, plusieurs produits (au sens articles) peuvent partager le même code-barres généré (appelés homonymes) et un même produit peut avoir plusieurs codes-barres générés (appelés synonymes).

Les règles de génération de codes-barres sont les suivantes.

Si un produit est ajouté manuellement à une famille de produits sans scanner son code-barres (ou sans saisir de code-barres), la solution génère un code-barres de 13 chiffres commençant par 31, suivi de 11 chiffres représentant un compteur incrémental, basé sur le nombre de produits ajoutés de cette manière dans le catalogue de produits du commerce. **Exemple :** Le produit `3100000000001` est le premier produit ajouté de cette manière, `3100000000002` étant le deuxième, etc. Si dans le catalogue de produits d'un commerce A, le premier produit ajouté sans code-barres est un briquet, et que dans un commerce B, c'est un stylo, ces deux produits auront le même code-barres généré.

Si un produit est virtuel et n'a pas de support physique (par exemple, un abonnement, un billet virtuel, un reçu de transfert d'argent, etc.), le code-barres généré commence par un `F`, suivi de chiffres et parfois d'un ou plusieurs tirets ('-'). Ces ventes sont appelées « ventes par famille », car le produit est représenté par sa famille, et le code-barres est simplement un identifiant de transaction généré au moment de la vente (local au commerce).

Exemple : Pour un transfert d'argent, le code-barres commence par `F`, suivi d'un identifiant de la famille de produits attribué à la société de transfert d'argent, `3321` par exemple, suivi de l'identifiant de la transaction. Le transfert `F3321-001` est obtenu pour le premier et `F3321-002` pour le deuxième, etc. L'incrément en suffixe est local au commerce. Deux codes-barres représentent donc le même produit, à savoir « Transfert d'argent ». Il est à noter que le processus de construction du code-barres n'est pas toujours identique, mais doit commencer par le caractère `F` pour toutes les « ventes par famille ».

Une autre source potentielle d'erreurs lors de l'analyse des codes-barres réside dans ceux saisis manuellement. Certains gérants de commerces saisissent manuellement les codes-barres pour diverses raisons : le code-barres ne peut pas être scanné (étiquette endommagée ou absence de code-barres) donc ils en inventent un, ou ils préfèrent identifier un produit avec un code-barres personnel. Dans ces cas, il existe une forte probabilité de problèmes d'unicité ou d'erreurs de saisie.

Voici quelques règles supplémentaires concernant les codes-barres, utiles pour le traitement des données.

— Les codes-barres du tabac sont fournis par Logista et commencent par « _ALTA_ ».

— Les codes-barres de la presse sont fournis par NAP et commencent souvent par `P` suivi d'une série de chiffres.

— Les codes-barres à 3 ou 4 chiffres sont dans la plupart des cas liés au système de ventes dites « bouton » de la caisse, qui permet au commerçant d'enregistrer une vente sans scanner de produit, mais en appuyant simplement sur un bouton spécifique de la caisse. Cette méthode génère alors un code à 3 ou 4 chiffres correspondant à l'identifiant de la famille de produits vendus. Il est également possible que ces codes-barres aient été entrés manuellement par le gérant du commerce.

3.2.2 Familles

Les produits sont regroupés en familles de produits, divisées en deux catégories : les familles globales (dites fixes), créées par l'entreprise, et les familles locales (dites non fixes), définies par les gérants des commerces clients.

Les familles globales, également appelées **familles fixes**, sont définies par l'entreprise. Elles couvrent des produits :

1. soumis à une législation spéciale limitant les fournisseurs potentiels, tels que le tabac et la presse ;
2. dématérialisés, tels que le transfert d'argent, les cartes prépayées de téléphone ou les cartes cadeaux.

Ces familles sont immuables et identiques d'un point de vente à un autre. Elles ne sont présentes dans le logiciel de gestion que si le point de vente souhaite exercer des activités liées à ces familles globales.

Les familles locales, également appelées **familles non-fixes**, sont définies par les gérants des commerces. Ces familles reflètent souvent les préférences individuelles des gérants ou le niveau de granularité souhaité pour la gestion des produits. Elles peuvent varier considérablement d'un commerce à l'autre, bien que certaines familles locales portent fréquemment des noms similaires. Par exemple, un gérant pourrait créer des familles basées sur les taux de TVA (5%, 10%, etc.), tandis qu'un autre pourrait organiser ses produits par type (Alimentation, Livres, etc.) ou par marque. Les produits de ces familles sont souvent définis sans identifiant de produit global, tel qu'un code-barres normalisé (c'est notamment le cas pour les produits alimentaires sans emballage). L'organisation des produits dans ces familles doit respecter deux règles.

1. Pas de doublons d'identifiant de produit : un identifiant de produit (souvent le code-barres) ne peut appartenir à plus d'une famille.
2. Tous les produits d'une même famille doivent avoir le même taux de TVA.

L'organisation des produits par famille présente deux avantages principaux : faciliter la recherche rapide d'un produit dans le catalogue et offrir un niveau de granularité plus simple et personnalisable dans les tableaux de bord des statistiques de ventes.

Il est important de noter que le système de facturation des caisses enregistreuses ne permet pas d'appliquer des taux de TVA différents à des produits au sein d'une même famille. Le taux de TVA est en effet assigné au niveau de la famille, et non au produit individuel. En d'autres termes, tous les produits d'une même famille sont facturés au même taux de TVA. Cette caractéristique est utile pour distinguer certains produits, notamment compte tenu des réglementations strictes de l'État français concernant les différents taux de TVA applicables à certains types de produits.

Des exemples de groupements de produits par familles avec des taux de TVA variables sont illustrés dans le tableau 3.6.

Étiquette de famille	TVA	Étiquettes de produits
Boisson non alcoolisée à emporter	5,5%	Eau, Canette soda
Boisson non alcoolisée sur place	10%	Verre cola, verre jus
Boisson alcoolisée	20%	Vin, Bière

TABLE 3.6 – Exemples de familles

3.2.3 Commerces clients

Un commerce client de Bimedia correspond à un commerce équipé d'une ou plusieurs caisses enregistreuses fournies par Orisha Retail Shops. Ces commerces peuvent être des franchises, bien que la majorité d'entre eux soient des commerces indépendants. Les types de commerces les plus courants dans le réseau de client Bimedia incluent des bureaux de tabac, des bars, des magasins de presse, des épiceries, des boulangeries, des fleuristes, ou une combinaison de plusieurs de ces types d'établissements.

3.2.4 Activités des commerces

Les activités des commerces constituent les résultats attendus de la classification des commerces par activités commerciales. L'entreprise a défini douze catégories d'activités commerciales, dont une catégorie générique « Autre ». Ces activités ont été définies par des experts métiers de l'entreprise, en adéquation avec leur connaissance du parc de clients équipés de la solution Bimedia, mais aussi des activités qu'il est pertinent pour eux de détecter. Les activités sont les suivantes :

- Tabac
- Presse
- Bar
- Café
- Fleuriste
- Épicerie
- Restaurant
- Boulangerie
- Hôtel
- Station-Service*
- Blanchisserie*
- Autre*

Il est important de noter que les activités marquées d'une étoile (*) ne sont pas présentes dans le jeu de données de validation. Ces activités sont rares dans l'ensemble de données initial et n'ont pas été sélectionnées lors de l'échantillonnage du jeu de données de validation (aucune

stratification n'était possible en raison de l'absence de données de référence). De ce fait, elles peuvent également ne pas être représentées dans le jeu de données d'entraînement.

De plus, certaines activités commerciales, telles que celles liées à la téléphonie ou au transfert d'argent, n'ont pas été isolées dans ce cadre. Actuellement, l'entreprise ne cherche pas spécifiquement à caractériser ces activités, car elles sont souvent pratiquées en parallèle d'activités principales comme la vente de tabac ou de presse. Bien que ces activités puissent parfois constituer une part significative du chiffre d'affaires de certains commerces, elles sont considérées comme additionnelles et ne font pas partie des cibles pour la classification.

3.2.5 Types de produits

Les types de produits correspondent à une catégorisation grossière des produits et services commercialisés par les commerces Bimedia. Bien qu'aucune donnée de référence n'existe pour un tel jeu de données, ces étiquettes sont fournies à titre indicatif pour les solutions de classification des commerces par activités commerciales qui s'appuieraient sur une classification préalable des produits. Il est important de noter que cette classification des produits n'est pas obligatoire pour réussir la classification des commerces par activité commerciale, et elle ne peut être directement évaluée, car aucun jeu de validation n'existe pour le moment.

L'entreprise a fourni une liste de 22 types de produits qui sont présents dans le jeu de données. Cette liste inclut le type « inconnu » pour les produits non identifiables et le type « divers » pour les produits identifiés qui n'appartiennent à aucun autre type de la liste. Le type de produit peut être considéré comme un attribut de la classe produit, mais il pourrait également être assigné à des familles de produits si la granularité de celles-ci le permet.

Les 22 types de produits (ou de familles de produits) sont les suivants :

- Alcool
- Alimentaire
- Dépôt*
- Article fumeur
- Banque*
- Billetterie
- Carte cadeau*
- CBD
- Divers
- Inconnu
- Carburant
- Fleur
- Papeterie
- Jouet
- High tech
- Jeu à gratter et pari
- Librairie*
- Moyen de paiement*
- Publicité*
- Presse*
- Service additionnel
- Tabac*
- Téléphonie*
- Timbres fiscaux*
- Transfert d'argent*
- Vape
- Web2store*

Les types de produits marqués d'une étoile (*****) correspondent également à des familles fixes (globales) que l'on peut retrouver dans le système Bimedia. Cependant, il est tout à fait possible que des produits correspondant à certaines de ces typologies soient vendus en dehors d'une famille fixe (c'est par exemple souvent le cas avec la librairie).

3.3 Stratégies d'échantillonnage et élaboration des jeux de données

Compte tenu de l'utilisation d'un sous-ensemble de données, la qualité de l'échantillonnage constitue un pilier fondamental de notre approche, garantissant une représentativité optimale de la diversité et de la complexité des catalogues de produits. Dans cette section, nous détaillons la manière dont nous avons constitué les jeux de données et expliquons les méthodes utilisées pour l'échantillonnage.

3.3.1 Élaboration et échantillonnage : jeu de données des ventes

Dans les systèmes d'information d'Orisha Retail Shops, un module est responsable de l'agrégation en temps réel de toutes les ventes des commerces à partir des tickets. Ce processus est suivi d'une consolidation quotidienne pour renforcer leur fiabilité (en cas de problème de connexion). Dans la base de données résultant de cette agrégation, une vente est identifiée de manière unique par un identifiant de commerce, un identifiant de produit (code-barres) et une date, comme illustré dans le tableau 3.7. Une vente possède également de nombreux autres attributs, notamment la quantité vendue, qui nous intéresse dans ce cas d'étude.

Pour constituer le jeu de données des ventes utilisé dans ce banc d'essai, nous avons commencé par sélectionner les ventes sur une période non divulguée d'une durée de 12 mois, et les avons agrégées par commerce et produit, en calculant la somme des quantités vendues, comme illustré dans le tableau 3.8, reprenant les données du tableau 3.7. La clé d'identification unique (clé primaire) de notre jeu de données est donc composée de l'identifiant de commerce et de l'identifiant de produit (colonnes nommées respectivement *ID commerce* et *ID produit* dans les tableaux 3.7 et 3.8).

ID commerce	ID produit	Libellé	Famille	ID famille	TVA	Qte	Date
aer...134	051131592292	SCOTCH	Papeterie	32	20	3	03/12/20XX
aer...134	051131592292	SCOTCH	Papeterie	32	20	6	04/12/20XX
aer...134	051231549890	CRAYON PAPIER HB	Papeterie	33	20	5	05/12/20XX
fgh...987	431126592191	coca cola 33CL	Boisson	42	10	12	06/12/20XX
fgh...987	431126592191	coca cola 33CL	Boisson	42	10	20	07/12/20XX
jkl...321	421786549320	Pain Chocolat	Boulangerie	12	5.5	8	07/12/20XX
jkl...321	421786549320	Pain Chocolat	Boulangerie	12	5.5	6	08/12/20XX
jkl...321	421786549320	Pain Chocolat	Boulangerie	12	5.5	9	09/12/20XX
abc...123	123456789012	EAU MINERALE 50CL	Boisson	44	10	15	08/12/20XX
def...456	987654321098	BIC bleu	Papeterie	35	20	7	09/12/20XX
def...456	987654321098	BIC bleu	Papeterie	35	20	2	10/12/20XX

TABLE 3.7 – Exemple d'extrait de la base de données d'agrégats

À l'aide du référentiel produit des clients, dans sa version la plus récente, nous avons associé à chaque vente, initialement composée de l'identifiant de commerce de l'identifiant de produit et de la quantité vendue (*Qte*), les variables suivantes : le libellé du produit (*Libellé*), le libellé de la famille de produit (*Famille*), le taux de TVA de la famille (*TVA*) et l'identifiant de la famille (*ID famille*), comme illustré dans le tableau 3.8.

Ensuite, nous avons procédé à l'échantillonnage aléatoire en sélectionnant 2325 commerces parmi les plus de 6500 présents dans le jeu de données d'origine, ne conservant que les ventes

ID commerce	ID produit	Libellé	Famille	ID famille	TVA	Qte
aer...134	051131592292	SCOTCH	Papeterie	32	20	9
aer...134	051231549890	CRAYON PAPIER HB	Papeterie	33	20	5
fgh...987	431126592191	coca cola 33CL	Boisson	42	10	32
jkl...321	421786549320	Pain Chocolat	Boulangerie	12	5.5	23
abc...123	123456789012	EAU MINERALE 50CL	Boisson	44	10	15
def...456	987654321098	BIC bleu	Papeterie	35	20	9

TABLE 3.8 – Extrait du jeu de données des ventes

des commerces sélectionnés. Cette sélection d'un sous-ensemble du jeu de données original a été validée avec l'entreprise afin de respecter les exigences de confidentialité.

Enfin, nous avons anonymisé ce jeu de données en générant un nouvel identifiant unique pour chaque commerce.

3.3.2 Élaboration et échantillonnage : jeu de données de validation

Comme mentionné précédemment, il est difficile pour l'entreprise de constituer un jeu de données significatif pour l'étiquetage manuel des produits, bien que la classification automatique des produits soit une problématique majeure pour l'entreprise. Dans ce banc d'essai, nous avons donc défini une problématique inhérente à cette dernière, mais pour laquelle nous étions en mesure de constituer un jeu de données de validation significatif : la classification des commerces par activités commerciales.

Sur la base du jeu de données des ventes, constitué des ventes de 2325 commerces, nous avons lancé une campagne d'étiquetage manuel des activités commerciales de 400 commerces, en nous appuyant notamment sur les fiches *Google My Business* des commerces. Cette tâche a été réalisée par des experts métiers (commerciaux et analystes) à l'aide d'une interface développée en interne.

Cette interface fonctionnait de la manière suivante.

1. Un algorithme sélectionnait aléatoirement un commerce parmi les 2325 présents dans le jeu de données des ventes.
2. Les ventes du commerce sélectionné étaient agrégées par famille de produits, en calculant la somme des quantités vendues et en listant les libellés de produits uniques.
3. Les résultats de l'étape précédente étaient organisés par ordre décroissant et affichés sous forme de tableau.
4. La page *Google My Business* du commerce était affichée à côté du tableau résultant de l'étape précédente.
5. Des boutons correspondant à diverses activités commerciales apparaissaient à l'utilisateur, lui permettant de sélectionner une ou plusieurs activités commerciales pour le commerce en question.
6. Une fois le ou les choix validés, l'étiquetage recommençait avec un nouveau commerce. Le processus s'arrêtait quand 400 commerces avaient été étiquetés.

Un exemple est présenté dans la figure 3.2, certains éléments ont été masqués par des rectangles rouges pour des questions de confidentialité, tous les boutons d'activités ne sont pas visibles sur la figure.

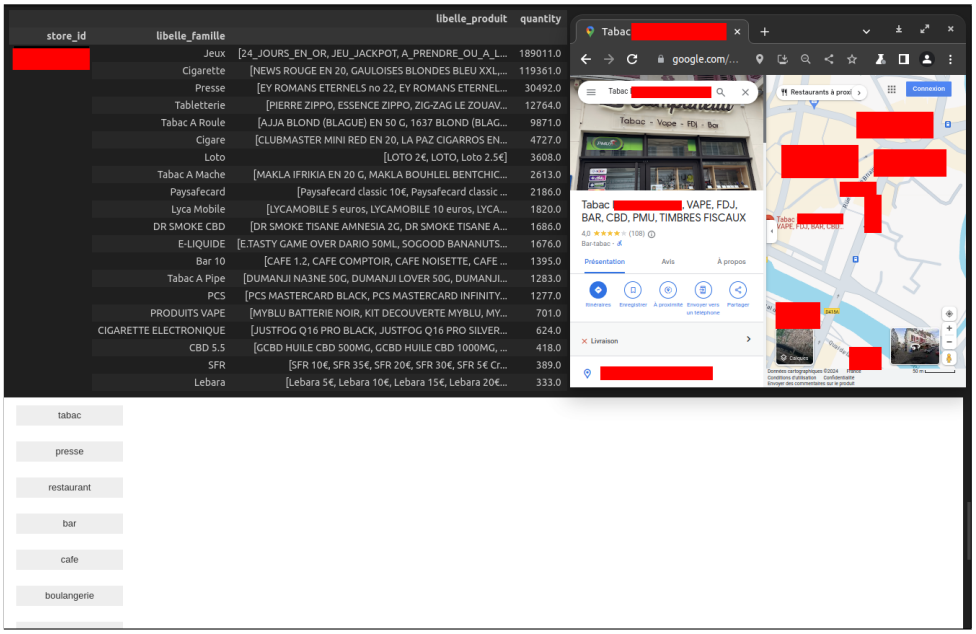


FIGURE 3.2 – Capture d'écran du programme permettant l'étiquetage manuel des activités des commerces

Les experts métiers avaient pour consigne de définir les activités du commerce en se basant uniquement sur le tableau des ventes affiché dans l'interface. La fiche *Google My Business*, incluant des photos du commerce, l'activité enregistrée dans Google Maps, ainsi que diverses autres informations, n'était utile que pour détecter des anomalies. En effet, l'entreprise savait par expérience que certains commerces peuvent posséder plusieurs marques de caisses enregistreuses au sein d'un même établissement afin de séparer physiquement les activités. Par exemple, dans un bar-tabac, une caisse enregistreuse peut servir pour les ventes liées au bar, tandis qu'une autre, d'une autre marque, peut être utilisée pour les ventes de produits liés au tabac. Dans ce cas, nous considérons que nous ne pouvons pleinement définir les activités commerciales de ces commerces, qui sont alors ignorés. Lors du processus de constitution du jeu de données de validation, trois commerces ont été identifiés dans ce cas de figure et ont été écartés.

3.4 Caractérisation des jeux de données

Afin de mieux comprendre la constitution des jeux de données fournis avec ce banc d'essai, nous avons réalisé plusieurs analyses descriptives pour chacun des deux jeux de données. Ces analyses sont présentées dans les sections suivantes, offrant ainsi une vue d'ensemble des caractéristiques principales de chaque jeu de données et permettant une meilleure appréhension des éléments qu'ils contiennent.

3.4.1 Caractérisation du jeu de données des ventes

Le jeu de données des ventes complet se compose de 11 637 397 lignes. Chaque ligne représente les ventes agrégées pour chaque combinaison de code-barres et de commerce sur une période non divulguée pour une durée de 12 mois. Les principales statistiques récapitulatives sont présentées dans le tableau 3.9.

Mesure	Familles fixes	Famille non-fixes	Total
Nombre de commerce	2325	2325	2325
Nombre de codes-barres de produits uniques	456 428	283 480	705 721
Nombre de noms de produits uniques	160 109	680 921	817 385
Nombre d'étiquettes de familles uniques	444	8071	8427
Nombre de combinaisons produit-commerce	10 339 347	1 298 050	11 637 397
Nombre moyen de produit par commerce	4459 ($\sigma \approx 4173$)	562 ($\sigma \approx 1012$)	5005 ($\sigma \approx 4715$)
Nombre de combinaisons famille-commerce	56 328	45 104	101 432
Nombre moyen de familles par commerce	24 ($\sigma \approx 149$)	19 ($\sigma \approx 14$)	44 ($\sigma \approx 18$)
Nombre total de produits vendus	489 879 740 (84%)	96 107 640 (16%)	585 987 380

Table 3.9 – Statistiques résumées des données de vente

Commençons au niveau des produits. Quelques statistiques intéressantes sur la distribution des codes-barres sont présentées dans le tableau 3.10 ainsi que dans les paragraphes suivants.

Mesure	Valeur
Nombre de combinaisons commerce–code-barres	11 637 397
Nombre de combinaisons commerce–code-barres numérique	1 638 282
Nombre de valeurs uniques de code-barres numériques	575 559
Nombre de combinaisons commerce–code-barres alphanumérique	9 998 983
Nombre de valeurs uniques de code-barres alphanumériques	130 042

Table 3.10 – Statistiques résumées sur la distribution des codes-barres

Comme indiqué dans le tableau 3.10, les codes-barres alphanumériques représentent une proportion significative du jeu de données par rapport aux codes-barres uniquement numériques. Cette différence s'explique principalement par la prévalence des commerces ayant une activité liée au tabac ou à la presse. En effet, un grand nombre de codes-barres associés aux produits du tabac commencent par « _ALTA_ » suivi d'une séquence numérique, en moyenne 270.98 produits par commerce ($\sigma \approx 96.50$). De manière similaire, la majorité des codes-barres relatifs à la presse débutent par la lettre « P », en moyenne 4154.80 par commerce ($\sigma \approx 4046.73$). Ces caractéristiques montrent à quel point ces produits sont fréquemment vendus dans de nombreux commerces. Un schéma comparable est observé pour d'autres produits, tels que les cartes prépayées, les jeux d'argent, et d'autres services, dont les codes-barres incluent souvent une abréviation du nom du produit, les rendant également alphanumériques.

Il n'est donc pas surprenant que le nombre de références uniques pour les codes-barres alphanumériques soit nettement inférieur à celui des codes-barres numériques. Les références de cigarettes ou de presse étant largement partagées entre les commerces concernés, le nombre de références uniques reste limité. Par ailleurs, les systèmes de codification standard comme

l'EAN13 ou l'EAN8 sont très répandus pour la majorité des produits de grande consommation et de détail, ce qui explique la diversité des codes-barres numériques, et nous montre la grande diversité d'activités au sein des commerces.

Quelques statistiques sont fournies dans le tableau 3.11. Nous pouvons observer que pour les codes-barres purement numériques présents dans ce jeu de données, 52% (39% + 13%) des quantités vendues concernent des produits avec des codes-barres de 13 chiffres, 22% avec des codes-barres de quatre chiffres, 8% avec des codes-barres de huit chiffres, et 8% également avec des codes-barres de 12 chiffres, les autres longueurs étant moins fréquentes. Pour rappel, les codes-barres à quatre chiffres correspondent principalement aux ventes par famille (également appelées ventes « bouton »). Les codes-barres générés par le système de caisse, composés de 13 chiffres et commençant par « 31000 », représentent 10% des codes-barres EAN13 dans le jeu de données, 76% des quantités vendues pour ce format de code-barres, 39% des quantités vendues pour les codes-barres numériques, et 15% des quantités vendues totales.

Concernant les codes-barres alphanumériques, 56% des quantités vendues concernent des codes-barres commençant par la chaîne « _ALTA_ » (associés au tabac), bien qu'ils ne représentent que 1,1% des codes-barres alphanumériques uniques. De plus, 11% des quantités vendues pour ce type de code-barres concernent des codes-barres commençant par le caractère « P » (souvent lié à la presse), qui constituent 92% des codes-barres alphanumériques uniques. Nous notons également que 19% des quantités totales de produits vendus concernent des codes-barres alphanumériques qui ne sont ni liés à la presse ni au tabac. Cela s'explique principalement par la vente de jeux à gratter et de produits dématérialisés, qui, comme décrit dans la section 3, ont des codes-barres générés avec ces formats.

Cette étude portant sur les codes-barres des produits présents dans le jeu de données montre l'importance prépondérante des règles métier dans la compréhension et l'exploitation des données. En effet, contrairement à un jeu de données plus classique, tel que celui relatif aux ventes de magasins de grande distribution [Andrii Samoshyn, 2013], la prise en compte de ces règles apparaît indispensable pour effectuer un traitement efficace des données.

Type de code barre	Sous type 1	Sous type 2	% Sous type 1	% Type	% Total
Numérique	13 chiffres	31000X	76%	40%	16%
		autre	24%	12%	5%
	4 chiffres			22%	
	8 chiffres			8%	19%
	12 chiffres			6%	
	autre			12%	
Alphanumérique	_ALTA_...			56%	34%
	P...			11%	7%
	autre			33%	19%

Table 3.11 – Statistiques sur les différents formats de codes-barres

Penchons-nous maintenant sur la distribution du nombre de produits par commerce, représentée dans les deux graphiques de la figure 3.3. La distribution est peu symétrique et dépend

fortement du type d'activité des commerces, de leur taille et de leur localisation. Pour ces deux figures, nous avons choisi de retirer toutes les ventes ayant des codes-barres commençant par le caractère 'P', afin d'éliminer une grande partie des ventes liées à la presse. En effet, tandis que le nombre moyen de produits par commerce était initialement de 5005 (avec un écart-type de 4715), ce nombre passe à 1097 (avec un écart-type de 1219) après suppression des produits relatifs à la presse, témoignant ainsi de l'importance de cette activité dans le jeu de données. Sans cette correction, le troisième quartile aurait dépassé les 18 000 produits, rendant le diagramme en boîte difficilement lisible. L'abscisse de l'histogramme a été limitée à 2500 produits.

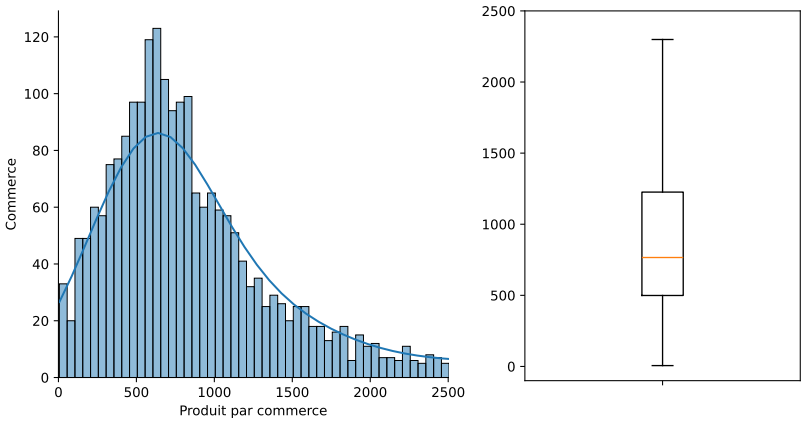


FIGURE 3.3 – Distributions des produits par commerce représentées par un histogramme (à gauche) et une boîte à moustaches (à droite)

À travers ces statistiques sur le nombre de combinaisons commerce/produit, il devient évident qu'il n'est pas trivial d'identifier de manière unique un produit dans ce jeu de données. Cela renforce également ce que nous avons avancé à la fin de l'étude des codes-barres : le contexte métier joue un rôle important dans la constitution du jeu de données (par exemple, l'impact de l'activité liée à la presse sur le nombre de codes-barres).

Comme expliqué dans la section 3.2.2, les produits sont regroupés dans des familles. Les estimations de la répartition des produits dans les familles et les commerces sont illustrées par les deux graphiques de la figure 3.4. Les valeurs aberrantes des boîtes à moustaches ont été ignorées, et l'abscisse de l'histogramme a été limitée à 250 produits.

La distribution des produits montre que les familles de produits non fixes contiennent généralement moins de produits que les familles fixes. En effet, en considérant le nombre de combinaisons commerce/produit vendues au sein des familles non fixes, 1 298 050 combinaisons sont présentes dans le jeu de données des ventes, contre 10 339 347 pour les familles fixes. Cette grande différence s'explique par trois raisons principales.

— Premièrement, les produits relatifs à la presse, classés dans les familles fixes, sont présents en très grand nombre dans le jeu de données, car il existe de nombreuses références, notamment pour les journaux quotidiens où un code-barres différent est attribué chaque jour.

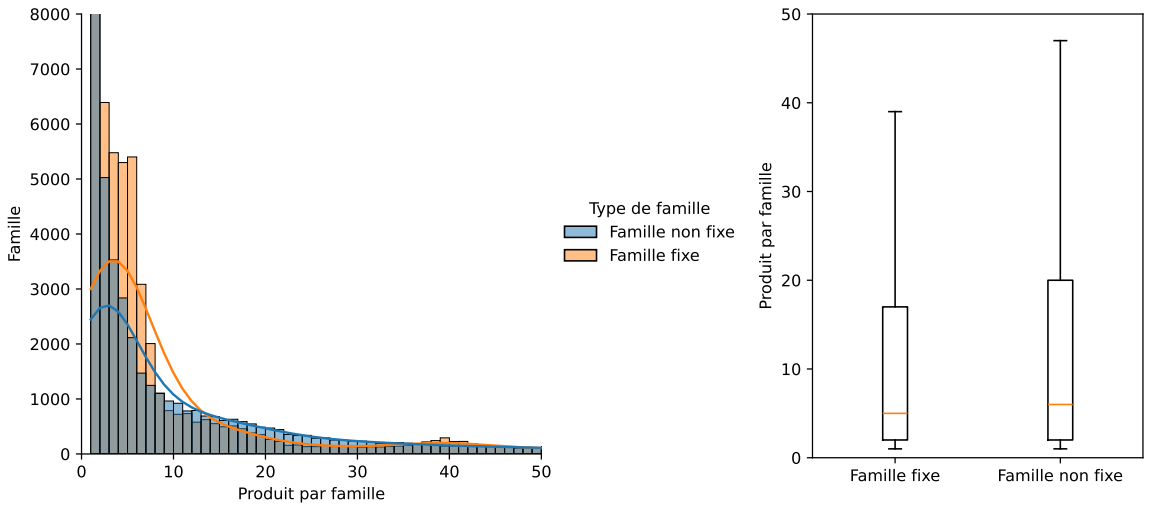


FIGURE 3.4 – Distributions des produits dans les familles représentées par un histogramme (à gauche) et une boîte à moustaches (à droite)

Contrairement à cela, d'autres produits comme les canettes de soda changent rarement de code-barres.

— Deuxièmement, les produits relatifs au tabac, également classés dans les familles fixes, comprennent de nombreuses références. Notre jeu de données étant majoritairement constitué de bureaux de tabac, cela augmente considérablement le nombre de combinaisons commerce/produit dans les familles fixes.

— Troisièmement, les ventes d'objets dématérialisés, comme les services de transfert d'argent, contribuent également à cette différence. Lors de l'exécution d'une vente de ce type de service, le logiciel génère un code-barres unique qui identifie la vente et non le produit. Chaque vente de ce type de service fait alors augmenter le nombre de combinaisons commerce/produit pour les familles fixes.

Cette disparité est également visible dans les sommes des quantités de produits vendus, réparties entre les familles fixes (489 879 740, soit 84%) et les familles non fixes (96 107 640, soit 16%).

Enfin, passons au niveau des familles. Dans la figure 3.5, nous pouvons observer la distribution du nombre de familles fixes et non fixes au sein des commerces. Les valeurs aberrantes ont été ignorées. Comme le montre cette figure, il y a globalement plus de familles fixes que de familles non fixes dans les catalogues produits des commerces. Cela s'explique principalement par l'obligation pour les commerçants d'utiliser ces familles de produits pour tous les articles soumis à une distribution contrôlée par l'État français. Parmi ces familles fixes, on retrouve notamment les produits relatifs au tabac, à la presse, aux jeux d'argent, à divers services bancaires et à d'autres produits réglementés. Comme précisé dans le tableau 3.9, le nombre moyen de familles par commerce est de 44 (avec un écart-type de 18), comprenant en moyenne 24 familles fixes et 20 familles non fixes. Cette étude sur les statistiques descriptives des familles de produit au sein des

commerces nous permet de mieux comprendre l'intérêt pour l'entreprise de ne plus se contenter uniquement d'une maîtrise des produits et activités relatives aux familles fixes (globales), mais de se pencher plus en détail sur les familles non fixes. Une meilleure maîtrise de ces familles locales permettrait alors de mieux comprendre toute une partie importante de l'activité de ses clients.

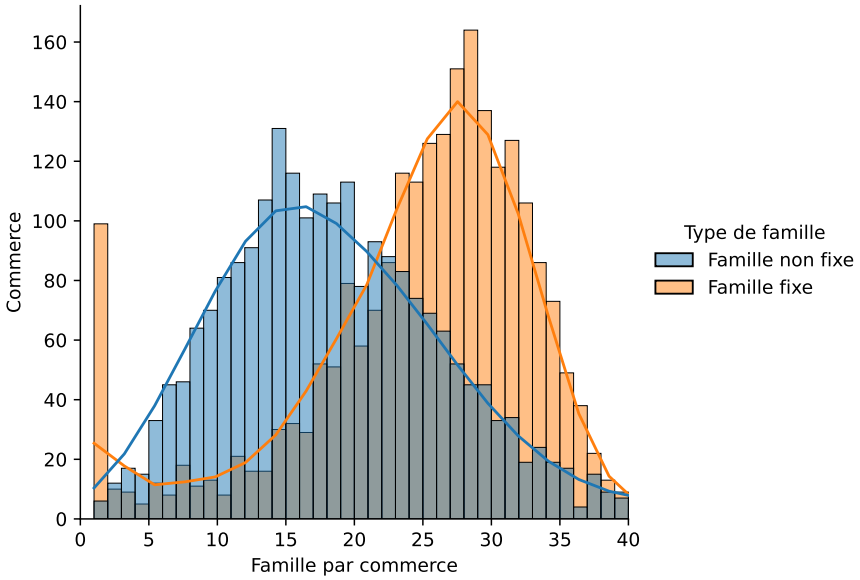


FIGURE 3.5 – Distribution du nombre de familles fixes et non fixes

3.4.2 Caractérisation du jeu de données de validation

Le jeu de données de validation est composé de 400 commerces dont les activités ont été étiquetées manuellement. Parmi ces 400 commerces, 29 combinaisons d'étiquettes d'activités sont présentes. La combinaison la plus fréquente est « Tabac Presse » avec 196 occurrences, suivie de « Tabac Presse Bar Café », présente 82 fois. Si nous extrayons, à partir du jeu de données des ventes, l'échantillon des commerces présents dans le jeu de données de validation, nous retrouvons des distributions très similaires, comme nous pouvons l'observer dans le tableau 3.12 (notamment visible sur les moyennes).

L'étude des codes barres, résumée dans le tableau 3.13 montre des proportions également très similaires a celles du jeux de données des ventes.

L'histogramme de la distribution des produits par commerce peut être retrouver dans la figure 7.22 en annexes. De la même manière, les histogrammes et diagrammes en boîte générés à partir de cet échantillon sur la distribution des produits dans les familles et des familles dans les commerces sont présentés dans les figures 7.20 et 7.21 en annexes.

À travers cette section, nous avons montré la représentativité de notre jeu de données de validation par rapport au jeu de données des ventes fourni dans ce banc d'essai. Nous abordons à présent les critères d'évaluation des performances que nous proposons pour notre banc d'essai.

Mesure	Familles fixes	Famille non-fixes	Total
Nombre de commerce	400	400	400
Nombre de codes-barres de produits uniques	129 868	98 962	220 862
Nombre de noms de produits uniques	38 547	166 059	200 093
Nombre d'étiquettes de familles uniques	171	1 930	2 074
Nombre de combinaisons produit-commerce	1 709 889	245 529	1 955 418
Nombre moyen de produit par commerce	4285 ($\sigma \approx 4270$)	615 ($\sigma \approx 1347$)	4889 ($\sigma \approx 5039$)
Nombre de combinaisons famille-commerce	9772	7728	17 500
Nombre moyen de familles par commerce	24 ($\sigma \approx 8$)	19 ($\sigma \approx 12$)	44 ($\sigma \approx 17$)
Nombre total de produits vendus	85 386 616 (85%)	15 649 282 (15%)	101 035 920

TABLE 3.12 – Statistiques résumées du jeu de données de validation

Type de code barre	Sous type 1	Sous type 2	% Sous type 1	% Type	% Total
Numérique	13 chiffres	31000X	76%	40%	15%
		autre	24%	12%	5%
	4 chiffres			23%	
	8 chiffres			8%	18%
	12 chiffres			6%	
	autre			11%	
Alphanumérique	_ALTA_...			56%	35%
	P...			11%	7%
	autre			33%	21%

TABLE 3.13 – Statistiques sur les différents formats de codes-barres (jeu de données de validation)

4 Critères d'évaluation des performances

Au cœur de notre démarche réside aussi le choix rigoureux des métriques d'évaluation, sélectionnées pour leur aptitude à refléter fidèlement la performance des modèles en termes de précision, de robustesse et d'adéquation avec les besoins réels.

Pour évaluer la performance de la classification des commerces par activité commerciale (étiquetage multiple), de nombreuses métriques peuvent être considérées [Grandini, 2020]. Nous avons choisi deux métriques principales : **l'exactitude** (*accuracy*) et **le score macro $\mathbf{F}_1$**. Ces métriques offrent une évaluation complète des performances des modèles dans ce contexte, et peuvent être calculées à partir des données annotées manuellement issues du jeu de données de validation (uniquement sur la partie qui n'a pas été utilisée pour l'entraînement en cas d'usage d'un modèle d'apprentissage automatique supervisé).

Pour illustrer nos propos, rappelons que les principales métriques pour l'évaluation de la qualité d'une tâche de classification (ou d'étiquetage multiple) sont basées sur les faux positifs (FP), les faux négatifs (FN), les vrais positifs (TP) et les vrais négatifs (TN). L'exactitude est une mesure précieuse à prendre en compte dans cette problématique d'étiquetage multiple, car elle indique la proportion de prédictions correctes. Elle est définie par la formule suivante :

$$\text{Exactitude} = \frac{\text{Nombre de prédictions correctes}}{\text{Nombre total de prédictions}} \tag{3.1}$$

soit

$$\text{Exactitude} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{3.2}$$

L'exactitude offre une bonne indication générale de la performance du modèle. Cependant, dans le contexte de l'étiquetage multiple et des jeux de données déséquilibrés, cette métrique peut être trompeuse. Par exemple, un modèle peut obtenir une haute exactitude en prédisant toujours la classe majoritaire, même s'il échoue à identifier correctement les classes minoritaires. Dans notre contexte, il suffirait alors que le modèle prédise systématiquement les étiquettes « Tabac » et « Presse » pour avoir un score acceptable (supérieur à 70%).

Le score macro F_1 est une métrique plus robuste pour évaluer les performances des modèles dans des contextes de classification déséquilibrée. Il combine la précision et le rappel en une seule métrique, en calculant la moyenne des scores F_1 pour chaque classe. En classification binaire, le score F_1 est défini comme suit :

$$\text{Score F}_1 = 2 \times \frac{\text{Précision} \times \text{Rappel}}{\text{Précision} + \text{Rappel}} \tag{3.3}$$

où

$$\text{Précision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3.4}$$

$$\text{Rappel} = \frac{\text{VP}}{\text{VP} + \text{FN}} \tag{3.5}$$

soit

$$\text{F1} = \frac{2 \times \text{TP}}{2 \times \text{TP} + \text{FP} + \text{FN}} \tag{3.6}$$

Pour un problème multi-classes, le score macro F_1 est ensuite calculé en prenant la moyenne des scores F_1 pour chaque classe :

$$\text{Score macro F}_1 = \frac{1}{N} \sum_{i=1}^{N} \text{Score F}_{1i} \tag{3.7}$$

où N est le nombre total de classes.

Pour illustrer nos propos autour des choix des scores, voici un exemple. En considérant les résultats d'étiquetage présentés (fictifs) dans le tableau 3.14, nous pouvons les reformuler sous un format rendant le calcul de score plus facile pour l'exemple, comme montré dans le tableau 3.15.

En nous basant sur les résultats fictifs présentés dans le tableau 3.15 ainsi que sur les formules présentées plus haut, nous obtenons alors le score global d'exactitude (ou macro exactitude) suivant :

TABLE 3.14 – Exemple fictif de résultats de classification

Identifiant commerce	Prédiction	Activités de référence
1121222	Tabac Presse	Bar Tabac Presse
4658129	Bar	Bar Presse Restaurant
6514278	Bar Presse	Tabac Presse

TABLE 3.15 – Exemple fictif de résultats de classification reformulés

Class	FP	FN	TP	TN
Tabac	1	1	1	0
Presse	0	0	2	1
Bar	1	1	1	0
Restaurant	0	1	0	2

$$\text{Exactitude} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} = \frac{4+3}{4+2+3+3} \approx 0.58$$

et le score global de F_1 (micro-F_1) :

$$\text{F}_1 = \frac{2 \times \text{TP}}{2 \times \text{TP} + \text{FP} + \text{FN}} = \frac{2 \times 4}{2 \times 4 + 2 + 3} \approx 0.62$$

Dans ces formules, *TP* représente la somme (toutes classes confondues) du nombre de vrais positifs, *FP* la somme du nombre de faux positifs, *TN* la somme du nombre de vrais négatifs, et *FN* la somme du nombre de faux négatifs.

L'inconvénient de l'utilisation des scores globaux (*micro-F_1* et *micro-accuracy*) est qu'ils sont fortement influencés par les classes abondantes dans le jeu de données. Ainsi, si le classificateur fonctionne très bien sur les classes majoritaires et mal sur les classes minoritaires, les scores globaux resteront élevés, en particulier pour le score de F_1 global (*micro-F_1*), pour lequel les vrais positifs ont un fort poids (en raison du coefficient multiplicateur). C'est pourquoi nous utilisons le score *macro-F_1* plutôt que le *micro-F_1*. Le modèle est ainsi fortement impacté en cas de mauvaises prédictions, même sur les classes minoritaires.

Ce fort impact s'explique par le fait que le score de *macro-F_1* correspond au score de F_1 moyen obtenu pour chacune des classes à prédire. Il n'y a donc plus de pondération entre les classes, la performance sur chaque classe ayant le même impact sur le score final, quelle que soit son abondance dans le jeu de données. En reprenant notre exemple, le score de macro-F_1 est alors :

$$\text{F}_1 = \frac{\text{F1}_{\text{tabac}} + \text{F1}_{\text{presse}} + \text{F}_{1\text{bar}} + \text{F}_{1\text{restaurant}}}{4} = \frac{0.5 + 1 + 0.5 + 0}{4} = 0.5$$

Nous avons également choisi de rapporter deux métriques concernant des durées : le **temps d'exécution** et le **temps de mise en œuvre**.

— **Le temps d'exécution** doit être inférieur à 12 heures, afin que la solution puisse être exécutée chaque nuit et soit utilisable dans un cadre opérationnel.

— **Le temps de mise en œuvre** est une estimation destinée à évaluer le coût associé au travail humain lors de la mise en place de l'approche. Parmi les solutions considérées, certaines sont semi-automatiques, voire dans un cas extrême, entièrement manuelles. Une solution d'apprentissage automatique non supervisée offrira probablement des performances moindres comparée à une solution spécifique élaborée sur la base d'un jeu de données annoté entièrement par des experts métier, mais cela représenterait un coût important, d'où l'intérêt de prendre en compte cette métrique. Il est important de noter que les solutions proposées dans ce manuscrit ont toutes été développées par la même personne.

La forme attendue des résultats de la caractérisation des activités des commerces est décrite et illustrée dans la section 2 en annexe. Nous présentons dans ce qui suit la mise en œuvre de notre banc d'essai pour évaluer une méthode de base pour la classification des commerces.

5 Évaluation de l'approche initiale de classification des commerces

Avant le début de ce projet, une approche existante était utilisée pour caractériser les activités des commerces. Dans cette section, nous allons décrire succinctement les différentes étapes clés de cette approche ainsi que les résultats obtenus en utilisant les métriques et les conditions expérimentales du banc d'essai. Il est important de noter que cette approche n'avait pas été évaluée de manière précise avant la constitution de ce banc d'essai, en raison de l'absence de jeu de données de validation. Ses résultats n'avaient pas été considérés comme exploitables après une observation empirique des résultats.

L'approche initiale reposait sur l'algorithme de partitionnement K-means. Voici les étapes clés de cette approche :

1. **nettoyage :** les libellés de famille sont nettoyés en corrigeant les erreurs d'encodage, en supprimant les accents, les ponctuations et les espaces superflus. Le résultat de cette étape correspond à la table 1 de la figure 3.6.
2. **transformation :** une table pivot est créée avec pour indice l'identifiant de commerce, pour colonnes les libellés de famille nettoyés, et pour valeurs les quantités de produits vendues pour chaque libellé de famille nettoyé et agrégées. Le résultat de cette étape correspond à la table 2 de la figure 3.6.
3. **étiquetage :** un partitionnement est effectué avec l'algorithme K-means, paramétré pour former cinq clusters, représentant cinq typologies différentes de commerces. Les paramètres de partitionnement, dont le nombre de clusters, ont été optimisés en utilisant le score de silhouette. Le graphique en figure 3.7 montre le coefficient de silhouette des différents clusters, le coefficient de silhouette moyen obtenu pour ce partitionnement, et permet de visualiser les tailles relatives des clusters. Les mêmes graphiques pour d'autres paramètres de partitionnement peuvent être retrouvés dans la section 3 en annexe.

Le résultat de cette approche est un jeu de données dans lequel les 2325 commerces sont catégorisés en cinq typologies différentes. L'objectif était ensuite de sélectionner de manière aléatoire

FIGURE 3.6 – Illustration de l'étape de transformation

Table résultante de l'étape de nettoyage **(1)**

Store ID	Code Barre	Produit	Famille	Quantité	TVA
db...5c2	31...02	pain	pain	2338	5.5
db...5c2	31...51	cookie	patisserie	2378	5.5
bb...49f	11...10	clearomiser q16 pro	divers 20	3	20
bb...49f	31...45	cafe	chaud	83253	7
db...5c2	31...51	tartelette	patisserie	304	5.5

↓

Table résultante de l'étape de transformation **(2)**

Store ID	pain	patisserie	divers 20	chaud
db...5c2	2338	2378	0	0
bb...49f	0	304	3	83253

au moins dix commerces par cluster et d'observer empiriquement les ventes de ces commerces afin de tenter de comprendre à quelle typologie de commerce les clusters correspondaient, dans le but de donner une étiquette à chaque cluster et donc aux commerces qui les composent. Parmi les cinq clusters, deux étaient assez clairement identifiables : les bureaux de tabac (avec ou sans vente de presse) et les magasins de presse. Les autres clusters n'étaient pas clairement identifiables.

Étant donné que cette approche repose sur des modèles de partitionnement et non sur une classification multi-classe, les résultats obtenus par rapport au jeu de données de validation, en utilisant les scores mentionnés dans la section précédente, n'ont pas réellement de sens. Nous avons néanmoins calculé ces scores à titre indicatif, et ils sont présentés dans le tableau 3.16.

Métrique	Valeur
Exactitude	0.23
Macro-F1	0.019
Temps d'exécution	68 secondes

TABLE 3.16 – Résultats de l'approche par partitionnement

Comme nous pouvons l'observer, l'approche affiche de faibles performances pour cette tâche de classification multi-classes des activités des commerces. Ces résultats ne sont pas surprenants, étant donné l'utilisation d'un modèle de partitionnement. Cependant, l'observation empirique des résultats suggère que le modèle avait tout de même correctement capturé la caractérisation de certaines typologies de commerces, bien que la granularité diffère de celle envisagée dans le jeu de données de validation.

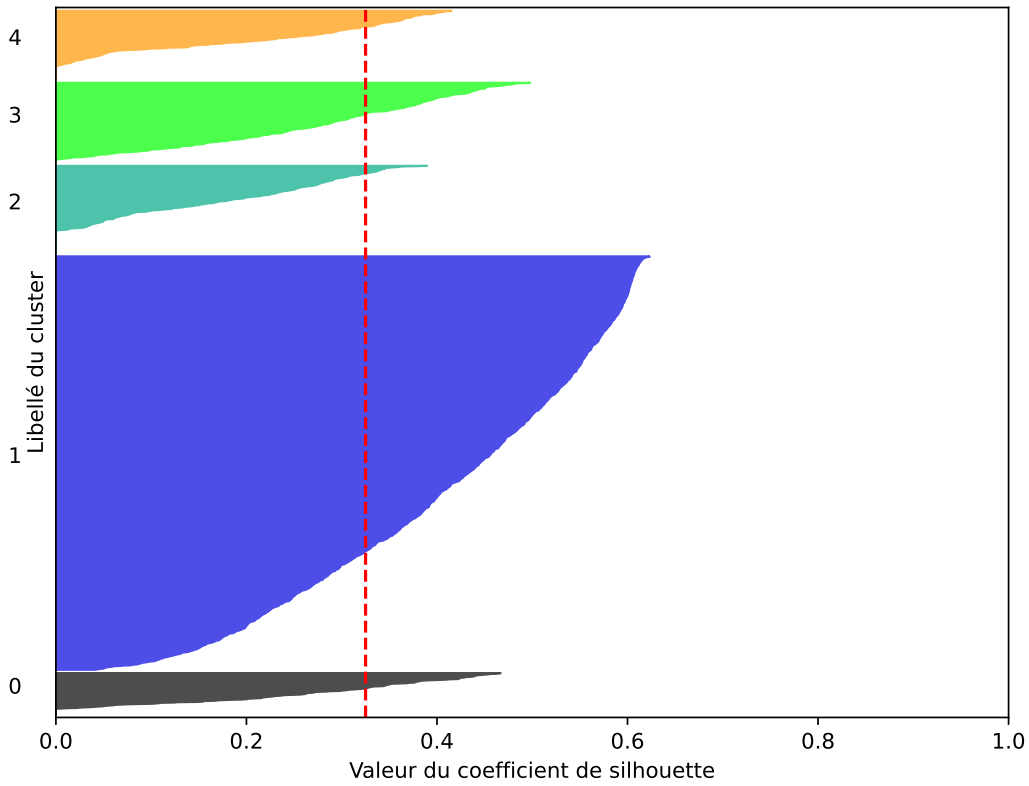


FIGURE 3.7 – Score de silhouette pour 5 clusters (moyenne = 0.33).

6 Conclusion et perspectives

Le banc d'essai que nous avons proposé dans ce chapitre nous permet d'avoir un cadre d'évaluation objectif pour l'étude des solutions potentielles à notre problématique de classification multi-classes des commerces par activités commerciales. Ce cadre nous permet non seulement de structurer nos expérimentations, mais également de garantir la reproductibilité des résultats et de faciliter la comparaison rigoureuse des différentes approches.

Grâce à ce banc d'essai, nous avons pu évaluer la solution existante. Les faibles résultats obtenus confirment l'analyse empirique sur son inadéquation aux nouveaux besoins de l'entreprise. Ce banc d'essai constituera un outil essentiel pour évaluer et comparer les nouvelles solutions que nous identifierons ou implémenterons pour résoudre cette problématique. Il nous permettra également de travailler sur l'optimisation des approches avec des critères précis à améliorer. De plus, le travail réalisé avec la constitution de ce banc d'essai devrait permettre de partager notre problématique et nos jeux de données avec la communauté scientifique, ouvrant ainsi la voie à des contributions extérieures et à des propositions de solutions innovantes. Les jeux de données sont disponibles dans le dépôt sur la forge du laboratoire LIAS [1].

1. https://forge.lias-lab.fr/thesaurusbt

L'une des principales perspectives de ce banc d'essai est d'étendre le jeu de données de validation (étiqueté manuellement) afin de permettre aux solutions basées sur l'apprentissage automatique supervisé de produire des résultats plus pertinents. Pour ce faire, nous envisageons de lancer de nouvelles campagnes d'étiquetage manuel, en impliquant davantage d'experts métiers pour enrichir la base de données et améliorer la représentativité des différentes activités commerciales. Nous prévoyons également de constituer un second banc d'essai, cette fois focalisé uniquement sur la problématique de classification des produits. Ce banc d'essai se concentrera sur les défis spécifiques liés à l'identification et à la catégorisation des produits vendus dans les commerces. Afin de constituer un jeu de données de validation significatif, nous avons l'intention de recourir à des méthodes d'annotation semi-automatiques, réduisant ainsi le besoin d'un étiquetage entièrement manuel, qui serait trop chronophage pour des humains. L'utilisation de référentiels de sources de données ouvertes, tels qu'Open4Good, ainsi que l'application de techniques avancées de traitement du langage naturel, sont envisagées pour accroître l'efficacité du processus d'annotation.

Ce second banc d'essai nous permettra également d'explorer des approches innovantes, telles que l'utilisation de modèles de langage pré-entraînés et de techniques de transfert d'apprentissage, pour améliorer la précision de la classification des produits. Nous espérons que cette initiative contribuera non seulement à résoudre notre problématique spécifique, mais aussi à enrichir le corpus de connaissances dans ce domaine.

Avec ce cadre clairement défini, nous sommes maintenant prêts à présenter notre solution présentée dans le chapitre suivant. Cette solution repose sur les enseignements tirés de nos expérimentations préliminaires et vise à offrir une approche plus robuste et plus précise pour la classification des commerces par activités commerciales. Nous y détaillerons les méthodes employées, les résultats obtenus, ainsi que les perspectives d'amélioration future.

Chapitre 4

Thesaurus-Based Transformation : encodage des produits basé sur un thésaurus

Objectifs

L'objectif de ce chapitre est de présenter une nouvelle méthodologie d'encodage des produits, nommée Thesaurus-Based Transformation (ThesaurusBT), qui s'appuie sur l'utilisation d'un thésaurus pour classifier les produits des commerces équipés de la solution Bimedia, développée par Orisha Retail Shops. Cette approche vise à résoudre la complexité liée à l'analyse des données commerciales hétérogènes et à la classification des produits dans un environnement industriel diversifié. Le chapitre examine les besoins en matière de caractérisation des activités commerciales, les limites des méthodes d'encodage traditionnelles dans ce contexte, ainsi que le développement d'un transformateur sur mesure. La méthodologie est validée à l'aide d'un banc d'essai, puis déployée en production pour optimiser l'efficacité des analyses, notamment dans le cadre de campagnes publicitaires ciblées.

Sommaire

1 Introduction

Comme nous l'avons abordé dans le chapitre précédent, la gestion des informations concernant les activités commerciales des commerces est devenue complexe en raison du volume élevé de commerces équipés de la solution Bimedia. De plus, les commerces ont tendance à diversifier et à modifier leurs activités au fil du temps, ce qui complique davantage la gestion de ces données. Par conséquent, la fiabilité des informations sur les activités stockées dans la base de données est souvent remise en question, réduisant ainsi leur utilité pour des analyses précises. Il est donc crucial pour Orisha Retail Shops de maintenir ces données à jour et fiables.

Cependant, comme mentionné précédemment, la simple tâche de caractérisation des activités des commerces n'est pas triviale dans le contexte industriel de l'entreprise. Chaque client ayant la liberté de gérer son propre catalogue de produits, les données collectées sont hétérogènes en termes de qualité et de quantité. Cette variabilité s'explique notamment par les différences de taille, d'activité et de localisation des commerces. Il est donc difficile de caractériser les activités commerciales sans une compréhension approfondie de leurs catalogues produits.

De plus, l'approche existante de caractérisation des commerces, reposant sur une transformation de type pivot et un modèle d'apprentissage automatique non supervisé basé sur le partitionnement, a donné de faibles résultats au regard des mesures définies dans le banc d'essai (voir section 5 du chapitre 3).

Dans ce chapitre, nous nous concentrons sur l'impact des stratégies de transformation des variables qualitatives du jeu de données des ventes sur la qualité des prédictions des activités commerciales des commerces équipés de la solution. Nous considérons trois grandes catégories de transformations :

— un simple nettoyage des variables catégorielles ;

— un ensemble de traitements issus de méthodes identifiées dans la littérature ;

— une approche sur-mesure basée sur un thésaurus de mots-clés rattachés à des concepts métiers spécifiques à notre contexte industriel.

Nous utilisons le résultat de ces transformations, suivi d'une transformation simple par pivot, en entrée d'un modèle d'apprentissage automatique. Cela permet d'obtenir une caractérisation plus ou moins fine et précise des activités commerciales des commerces, en fonction des différentes approches. Les résultats de prédiction, basés sur les critères définis dans le banc d'essai, servent à évaluer indirectement la qualité de l'étape de transformation.

Une première section de ce chapitre est consacrée à la spécification des besoins de l'entreprise en matière de caractérisation des activités des points de vente (section 2). Ensuite, nous expliquons notre choix de diriger nos travaux vers le développement d'un transformateur pour l'encodage des produits (section 3). Nous détaillons ensuite les différentes étapes du développement de la solution ThesaurusBT (section 4), le plan expérimental s'appuyant sur le banc d'essai détaillé précédemment (section 5), et les résultats obtenus (section 6). Enfin, nous concluons ce chapitre (section 7).

2 Besoin pour spécifiques pour la classification des produits

L'entreprise Orisha Retail Shops a besoin d'accéder à des données à jour concernant la caractérisation des activités commerciales de ses commerces. L'objectif est de pouvoir croiser ces données avec diverses autres afin de réaliser des analyses pour différents services internes ou externes, d'alimenter des tableaux de bord, et plus généralement, d'améliorer la connaissance de son parc de clients ayant la solution Bimedia installée. Cependant, cette tâche de caractérisation des activités des commerces dépend grandement de la transformation efficace des données relatives aux différents catalogues produits des clients et de l'incorporation des ventes. Nous avons donc décidé, pour ce chapitre, de fixer cinq objectifs.

Le premier objectif est la **classification explicite des produits** via une étape de transformation des variables textuelles (libellés et/ou noms de famille des produits). Cela permet de résoudre la problématique d'encodage des catalogues produits. En classifiant les produits dans des catégories généralisées au sein des différents commerces, nous pouvons utiliser les ventes agrégées par classe de produit prédite pour la phase d'encodage et l'incorporation des chiffres de ventes. Ainsi, la taille de la matrice en entrée du modèle d'étiquetage multiple sera de N variables, où N correspondra au nombre de classes définies dans le niveau le plus fin de classification des produits.

Afin de réutiliser ces résultats pour d'autres tâches que l'étiquetage des activités commerciales, nous souhaitons que les classes de produits soient conçues pour être utilisables dans le plus grand nombre de contextes possibles : suffisamment fines pour répondre à divers besoins, mais suffisamment généralistes pour rester compréhensibles et utilisables par le plus grand nombre. Une classification multiniveau avec plusieurs degrés de granularité serait donc idéale

Le second objectif est l'**explicabilité des résultats** de la classification des produits. Les usages envisagés étant multiples (internes et externes), l'entreprise souhaite éviter l'usage de modèles de type « boîte noire » pour cette étape, afin de pouvoir justifier et expliquer les résultats obtenus.

Le troisième objectif est l'**étiquetage multiple des activités commerciales** des commerces de manière fine et précise. Cet objectif est dépendant du succès du premier. Nous souhaitons que le résultat soit fidèle à la vision des experts métier et robuste aux changements transitoires de volume de ventes (par exemple, lors de périodes spécifiques) tout en tenant compte des activités rares, comme les stations-service, relativement peu nombreuses dans le réseau des clients Bimedia.

Le quatrième objectif est la **maintenabilité du modèle**. Nous souhaitons que le flux de traitement soit facile à surveiller et à maintenir, et que son évolution, comme l'ajout d'une nouvelle classe de produit ou d'activité commerciale, soit aisée pour le personnel de l'entreprise.

Le cinquième et dernier objectif est l'**exécution rapide du modèle**. Pour que les résultats soient exploitables (avec des données fraîches), le modèle doit pouvoir être ré-exécuté plusieurs fois par mois, avec un temps d'exécution inférieur à 12 heures. Les résultats de la classification et de l'étiquetage doivent être stockés dans la base de données afin d'alimenter les tableaux de bord (internes et clients).

Nous pouvons citer deux usages principaux de ces résultats : la **segmentation du parc pour le ciblage publicitaire**, et la **prédiction des ventes par types de produits** pour une zone géographique et une période données. La classification des produits permettrait d'améliorer l'efficacité des campagnes publicitaires en ciblant précisément les commerces concernés, et d'effectuer des analyses prédictives des ventes. Cela est aujourd'hui impossible à cause de la disparité des catalogues produits entre commerces.

3 Justification du choix des transformateurs

Nous commençons par rappeler pourquoi il est nécessaire de transformer les données. Les données brutes correspondent aux ventes de 2325 commerces clients de la solution Bimedia, agrégées par produit et par commerce, un extrait est présenté dans la tableau 4.1. Notre problématique principale étant la caractérisation des activités des commerces, une tâche de classification multi-classe également appelée étiquetage multiple. Le jeu de données étant composé de variables qualitatives (nom du produit et famille de produit), celles-ci doivent être encodées/incorporées afin d'être interprétables par un modèle d'apprentissage automatique. Une autre transformation est également nécessaire. Alors que le jeu de données des ventes contient autant de lignes par commerce qu'il y a de produits différents dans leur catalogue, le jeu de données de référence (ou de validation) contient une ou plusieurs étiquettes d'activités par commerce, soit une ligne par commerce. Les données d'entrée doivent donc être transformées pour que les informations d'un commerce soient résumées en une seule ligne.

ID commerce	ID produit	Libellé	Famille	ID Famille	TVA	Qte
aer...134	051131592292	SCOTCH	Papeterie	32	20	9
aer...134	051231549890	CRAYON PAPIER HB	Papeterie	33	20	5
fgh...987	431126592191	coca cola 33CL	Boisson	42	10	32
jkl...321	421786549320	Pain Chocolat	Boulangerie	12	5.5	23
abc...123	123456789012	EAU MINERALE 50CL	Boisson	44	10	15
def...456	987654321098	BIC bleu	Papeterie	35	20	9

Table 4.1 – Extrait du jeu de données des ventes

Pour réaliser la tâche de caractérisation des activités des commerces clients de Bimedia, plusieurs approches sont envisageables. L'approche la plus directe pour résoudre ce problème consiste à incorporer les données à l'aide d'un modèle pré-entraîné destiné à cette tâche, puis à entraîner un modèle d'apprentissage automatique à partir des données transformées. Dans ce type d'approche, l'exploitabilité des données incorporées en dehors du cadre de l'entraînement du modèle de prédiction n'est pas une priorité. Les données incorporées peuvent donc être totalement dénuées de sens pour un humain, ce qui importe peu tant que le modèle est capable de les interpréter et d'apprendre efficacement. C'est souvent le cas, particulièrement avec des transformateurs pré-entraînés tels que Gemini, GPT-4, BERT, etc., qui transforment des variables textuelles en vecteurs plus ou moins complexes, capturant la représentation sémantique dans un espace vectoriel à multiples dimensions. Le modèle de prédiction qui suit cette étape

d'incorporation devient alors encore plus une « boîte noire », tant par sa constitution que par la forme de ses entrées, qui ne sont plus lisibles par un humain. Cette approche ne répond toutefois pas aux attentes de l'entreprise pour trois raisons principales.

Premièrement, comme mentionné plus haut, les transformateurs classiques et autres méthodes d'encodage des variables catégoriques (comme les noms de produits et familles de produits) ne sont pas performants sur notre jeu de données. Certaines mènent à la création de matrices de grandes dimensions et creuses, tandis que d'autres ne parviennent pas à capturer efficacement les relations d'équivalence entre les différents catalogues produits des clients de la solution Bimedia. Dans tous les cas, cela se traduit par de faibles résultats lors de l'entraînement des modèles de prédiction basés sur ces pré-traitements.

Dans le tableau 4.2, nous présentons un exemple dans lequel un modèle de langage pré-entrainés reconnu parmi les plus performants sur la langue française échoue a capturer les différences sémantiques de simples libellés de produits. Dans cet exemple, nous fournissons une matrice de distance entre les vecteurs résultants de la phase d'incorporation des libellés de produits à partir du modèle pré-entrainé CamemBERT. Dans cette matrice, nous pouvons notamment observer que les prédictions du modèle indique que le libellé **Sirop menthe** est sémantiquement plus proche du libellé **Liquide vape** que du libellé **Bière**. Or, cette situation ne répond pas aux attentes, car la bière et le sirop de menthe sont deux boissons couramment présentes dans les commerces liés aux activités de *Bar* ou de *Café*, tandis que le liquide pour cigarette électronique relève d'un secteur d'activité tout à fait distinct. Les libellés présents dans cet exemple sont portant bien moins spécifiques à notre domaine qu'une grande partie des autres libellés de produits contenus dans le jeu de données.

Table 4.2 – Exemple de similarité après incorporation avec le modèle CamemBERT

Libellé de produit/Distance	**Liquide vape**	**Bière**	**Sirop menthe**
Liquide vape	0	13.69	11.9
Bière	13.69	0	13.74
Sirop menthe	11.9	13.74	0

Deuxièmement, nous perdons alors l'explicabilité des résultats ; il est difficile, voire impossible, de comprendre comment le modèle est arrivé à la prédiction faite. Or, l'explicabilité des résultats est un point important pour l'entreprise pour expliquer et ajuster sa stratégie.

Troisièmement, la maintenabilité du traitement est difficile dans le cas d'usage de méthodes d'incorporation basées sur des modèles externes. En effet, en considérant par exemple l'utilisation d'un modèle d'incorporation pré-entraîné comme BERT, l'arrivée d'un nouveau concept attaché à un nouveau vocabulaire provoquera une dérive du modèle. Cette dérive sera alors difficile à corriger en raison de la dépendance de l'approche à un pré-entraînement externe (venu de la communauté qui maintient le modèle).

Pour toutes les raisons précédentes, nous avons décidé de nous orienter vers le développement interne d'un transformateur. Le but de ce transformateur est de répondre à la fois à une tâche de classification des produits et à un encodage efficace des variables textuelles relatives aux produits,

permettant ainsi une caractérisation de l'activité des commerces de manière explicable.

4 Notre proposition : Thesaurus-Based Transformation

Notre proposition repose sur une transformation basée sur un thésaurus, abrégée *ThesaurusBT*, qui regroupe les produits en catégories. La catégorisation des produits est effectuée pour deux raisons. Premièrement, la capacité à catégoriser les produits vendus par les commerces sert à plusieurs fins, dont la génération de tableaux de bord et la publicité ciblée. Cette transformation est par ailleurs explicable comme nous allons le voir, ce qui rend notre méthode supérieure aux solutions de codage non explicatives pour les entreprises recherchant un étiquetage de produits compréhensible. Deuxièmement, comme mentionné précédemment, le défi principal posé par l'étiquetage multiple des activités commerciales des commerces ne réside pas dans le processus d'étiquetage lui-même, mais dans la transformation des données requise pour incorporer les ventes de produits. Cela implique spécifiquement d'encoder des variables catégorielles textuelles « sales » telles que les libellés de produits ou de familles, qui ont souvent une haute cardinalité, des synonymes, des homonymes et d'autres variabilités morphologiques. La classification des produits représente une solution pour cette tâche.

Pour catégoriser les produits, notre approche est basée sur un thésaurus qui relie les termes récurrents apparaissant dans les libellés de familles à des catégories grossières définies par l'entreprise. Dans notre cas d'utilisation, nous avons défini un total de 60 catégories de produits, plus une catégorie *inconnue* pour les types de produits non identifiés. Ces catégories sont par exemple : *restauration à emporter*, *vêtements*, ou *tabac*. Nous supposons qu'en moyenne sur l'ensemble du jeu de données, les étiquettes de familles peuvent être correctement mappées aux catégories prédéfinies. Nous considérons cette hypothèse comme raisonnable lorsque l'entreprise, comme c'est le cas pour Orisha Retail Shops, possède une connaissance significative du type de produits vendus par les commerces équipés de leur solution, et que le nombre de commerces clients de la solution est conséquent. Ainsi, ThesaurusBT nécessite certaines connaissances métier pour être applicable.

Voyons maintenant de plus près comment cela se retranscrit dans les phases de conception du transformateur.

4.1 Conception du thésaurus

Pour le fonctionnement de notre approche par transformation basée sur un thésaurus, il est important de produire un thésaurus de bonne qualité, adapté au contexte, aux données, mais aussi aux besoins. Nous listons dans cette section trois prérequis fondamentaux, qui nécessitent un travail en amont du déploiement de notre transformateur de données (réalisant la tâche de classification des produits et d'encodage en même temps). Dans notre cas, ce travail a été effectué avec une équipe d'experts métier (notamment des commerciaux et d'autres collaborateurs familiers avec les catalogues de produits des clients de la solution) au cours d'ateliers de travail.

Le premier prérequis est de lister les catégories de produits que l'entreprise souhaite identifier. Étant donné qu'une grande variété de types de produits peuvent être vendus, ils peuvent être

organisés en une hiérarchie. Dans notre cas d'utilisation, nous utilisons un arbre à deux niveaux. Un exemple de catégorie de produits de niveau 1 est *alimentation*, avec les niveaux 2 suivants : *boulangerie*, *confiserie*, *épicerie non-boisson*, *restauration sur place*, *restauration à emporter*, *boisson alcoolisée sur place*, *boisson alcoolisée à emporter*, *boisson non-alcoolisée sur place* et *boisson non-alcoolisée à emporter*. Spécifiquement pour le cas de l'entreprise, les familles globales, également appelées « familles fixes » car déjà enregistrées dans les systèmes informatiques des clients à l'installation de la solution, sont directement mappées aux éléments de cette hiérarchie sans autre exigence – les exigences suivantes s'appliquent donc uniquement aux familles locales, c'est-à-dire « sales ».

Le deuxième prérequis est un dictionnaire de mots-clés courants, c'est-à-dire des n-grams qui sont couramment associés à une catégorie lorsque les propriétaires de commerces fournissent des étiquettes textuelles pour les familles de produits. Ces mots-clés sont mappés à la catégorie correspondante dans la hiérarchie. Pour assister la constitution de ce genre de dictionnaire, des traitements d'agrégations et de généralisation de chaînes de caractères, à l'aide d'outils de traitement du langage naturel tels que la lemmatisation, la correction automatique, etc., sur les données brutes ont permis de faire ressortir les mots-clés qui reviennent le plus souvent dans les catalogues clients. De cette manière, les experts métier pouvaient s'appuyer sur ce type d'information afin de fournir un dictionnaire suffisamment généraliste pour convenir au plus grand nombre de commerces possibles, sans devenir trop spécifique pour éviter sa complexification. L'idée est de minimiser le nombre de mots-clés par catégorie de produit, tout en maximisant le nombre de produits qui seront potentiellement étiquetables grâce à ces mots-clés. Ceci évite le surapprentissage. Un extrait du thésaurus utilisé est fourni dans le listing 4.1.

Le troisième prérequis est une liste de mots-clés ambigus, accompagnée de règles de différenciation. Les mots-clés ambigus sont des n-grams couramment associés à deux catégories ou plus. Par exemple, le mot *boisson* pourrait être utilisé pour décrire à la fois des boissons alcoolisées et non alcoolisées. Le jeu de données des ventes comprend des valeurs de TVA (Taxe sur la Valeur Ajoutée) qui, couplées à des règles commerciales, peuvent permettre de résoudre la classification des produits décrits par des mots-clés ambigus. En effet, selon la loi française, la TVA appliquée aux boissons alcoolisées et non alcoolisées diffère (respectivement 20% et 10% ou 5,5%), et peut être utilisée pour discriminer l'une des deux catégories. Un extrait du thésaurus incluant cette notion d'ambiguïté et de TVA est visible dans le listing 4.2, les règles de différenciation sont elles incluses dans le code du transformateur pour le moment.

Dans notre mise en œuvre, nous utilisons des fichiers au format JSON pour stocker cette connaissance métier. Par conséquent, elle peut être facilement mise à jour en ajoutant de nouveaux mots-clés si l'identification d'un nouveau type de produit est nécessaire, ou si la granularité du type de produit doit être modifiée. Dans la section suivante, nous décrivons comment cette connaissance métier est utilisée dans notre méthode.

4.2 Construction automatisée d'un dictionnaire de correspondance

L'objectif de ThesaurusBT est de créer automatiquement un dictionnaire de correspondance entre les produits vendus par les commerces et les catégories fournies par l'entreprise. Notre

```
{
    "Confiserie": {
        "keywords":["confiserie","bonbon","chocolat","carambar","malabar","sucette"
    ↪ ],
        "keywords_plural": ["confiseries","bonbons","chocolats","carambars","
    ↪ malabars","sucettes"],
        "keyword_tuples":[],
        "keyword_tuples_plural":[],
    },
    "Epicerie hors boisson": {
        "keywords":["alimentation", "alimentaire","cremerie","frais","legume","
    ↪ fruit","surgele","conserve","miel","boucherie","biscuit","pate","riz","oeuf"
    ↪ ,"sucre","farine","lait","huile","chips","bio","epice","primeur","confiture"
    ↪ ,"viande","aliment","porc","conserverie","cereale","poisson","potage","
    ↪ nourriture","alim","sodebo","epicerie"],
        "keywords_plural": ["alimentations", "alimentaires","cremeries","legumes","
    ↪ fruits","surgeles","conserves","miels","boucheries","biscuits","pates","
    ↪ oeufs","sucres","farines","laits","huiles","epices","primeurs","confiture","
    ↪ viandes","aliments","conserveries","cereales","poissons","potages","
    ↪ nourritures","epiceries"],
        "keyword_tuples":[["epicerie","alimentaire"]],
        "keyword_tuples_plural":[["produits","regionaux"],["produits","locaux"]],
    }
}
```

FIGURE 4.1 – Thésaurus de mots-clés requis pour la catégorisation des produits

```
{
    "Boisson alcool sur place":{
        "ambigious_words" : ["boisson","bar"],
        "ambigious_words_plural" : ["boissons"],
        "ambigious_tuples":[],
        "ambigious_tuples_plural":[],
        "vat" : [19.6,20.0]
    },
    "Boisson sans alcool sur place":{
        "ambigious_words" : ["boisson","bar"],
        "ambigious_words_plural" : ["boissons"],
        "ambigious_tuples":[],
        "ambigious_tuples_plural":[],
        "vat" : [2.1,5.0,5.5,7.0,10.0]
    },
}
```

FIGURE 4.2 – Thésaurus mots-clés ambigus et taux de TVA requis pour la catégorisation des produits

méthode repose sur le principe suivant. Bien que les propriétaires de commerces conservent la liberté d'attribuer des étiquettes à leurs produits et de les regrouper en familles, en raison du nombre considérable de commerces, il est probable qu'une proportion significative d'entre eux présente certains mots-clés identifiés par les experts métier. De plus, une correspondance réussie d'une famille, et donc des produits sous-jacents, à une catégorie pour certains commerces permet la catégorisation des produits ayant le même identifiant dans tous les autres commerces. Le mappage effectué dans ThesaurusBT suit les étapes de traitement décrites dans le reste de cette section.

La première étape de traitement des données consiste à réaliser un nettoyage partiel des valeurs textuelles des variables catégorielles « sales » devant être encodées. Ce nettoyage inclut plusieurs opérations :

— Conversion en minuscules pour uniformiser la casse.

— Suppression des espaces superflus.

— Élimination des accents et des caractères spéciaux.

— Retrait des mots vides (stopwords), y compris ceux spécifiques au domaine d'activité.

— Homogénéisation des encodages de chaînes de caractères.

— Modification de certaines chaînes pour des cas spécifiques, tels que les unités de mesure (grammes, litres, etc.).

Dans le jeu de données fourni, ce nettoyage est appliqué aux étiquettes des produits et des familles de produits, afin d'améliorer leur traitement dans les étapes suivantes d'encodage. Un exemple sur un sous ensemble de données est visible dans le tableau 4.3.

Table 4.3 – Extrait de l'ensemble de données avec libellés nettoyés

Libellé produit	Libellé produit nettoyé	Libellé famille produit	Libellé famille produit nettoyé
SIXTIZ MANGO HAZE 3G	sixtiz mango haze 3 gramme	CBD	cbd
FRUITBERRY SHOT 50 ML	fruitberry shot 50 millilitre	E LIQUIDE	e liquide
PILOT VBALL BLEU 0.5	pilot vball bleu 05	Papeterie	papeterie
La Poste Mobile 5€	la poste mobile 5	Laposte Mobile	laposte mobile
DORITOS GOUT NATURE 280G	doritos gout nature 280 gramme	Epicerie 5.5	epicerie 55
EPEN 3 SAVEUR CLASSIQUE ICE	epen 3 saveur classique ice	PRODUITS VAPE	produits vape
GRINDER HORNET BARRILLET	grinder hornet barrillet	Articles Fumeurs	articles fumeurs
ASTERIX LE MENHIR D"OR	asterix le menhir d or	Librairie 5.5	librairie 55
GITANES FILTRE EN 20	gitanes filtre en 20	Cigarette	cigarette
Beret Facon Jean	beret facon jean	Textile	textile
Ticket PCS rechargement 150€	ticket pcs rechargement 150	PCS	pcs

La deuxième étape identifie les produits qui peuvent être catégorisés à l'aide de règles simples. Dans le contexte d'Orisha Retail Shops, cela est effectué sur les familles globales, qui sont normalisées et supervisées par l'entreprise. Dans le jeu de données fourni, l'identification et l'étiquetage des produits des familles globales se basent sur le champ identifiant de famille (*ID famille*) et une correspondance fournie entre les identifiants de famille et les catégories. Un exemple de l'ensemble de données après identification des produits de familles fixes et non fixes est visible dans le tableau 4.4. Les étapes de traitement restantes ne s'appliquent qu'aux produits issus de familles aux étiquettes sales — les familles locales dans le cas de Bimedia (*unfixed* dans l'illustration du tableau 4.4).

TABLE 4.4 – Extrait de l'ensemble de données après identifications des produits de familles fixes et non fixes (*les libellés sont nettoyés)

Store ID	Code barre	Libellé produit(*)	Libellé famille produit(*)	ID famille	Type de famille
d6b...739	3770021102065	sixtiz mango haze 3 gramme	cbd	148	unfixed
a97...2f0	1392013009026	fruitberry shot 50 millilitre	e liquide	38	unfixed
831...9d2	4902505085420	pilot vball bleu 05	papeterie	32	unfixed
281...2c4	LAPOM005	la poste mobile 5	laposte mobile	9811	unfixed
1a8...b0f	3168930165231	doritos gout nature 280 gramme	epicerie 5 5	3011	unfixed
957...d66	_S628ead88	epen 3 saveur classique ice	produits vape	851237	vape
502...86e	3100000001278	grinder hornet barillet	articles fumeurs	23	unfixed
561...3bf	9782864973461	asterix le menhir dor	librairie 55	60	unfixed
13d...d23	_ALTA_206	gitanes filtre en 20	cigarette	1101	cigarette
57d...c1c	3760034186183	beret facon jean	textile	80	unfixed
be2...be5	PSCCO150	ticket pcs rechargement 150	pcs	9834	moyen de paiement

La troisième étape crée des identifiants uniques pour les instances de produits. Cette identification n'a pas besoin d'être exacte : certaines instances peuvent être fusionnées incorrectement ou conservées séparément sans impacter significativement l'ensemble du processus. Les meilleurs résultats sont toutefois obtenus lorsque les erreurs d'identification sont minimisées. Dans notre jeu de données, cette étape commence par l'identification des codes-barres de produits internes au commerce et des codes-barres de produits génériques, tels que les codes-barres normalisés EAN13. Pour traiter les codes barres, nous avons défini des règles qui considèrent comme générés tous les codes-barres qui sont alphanumériques ou numériques de moins de 8 caractères, ainsi que ceux de la forme EAN13 commençant par «3100». La création d'un identifiant de produit unique (*UID produit*) dans les cas de codes-barres générés est donc effectuée en générant un hash à partir du nom du produit nettoyé. Quelques exemples sont fournis dans le tableau 4.5, où l'on peut observer que certains identifiants uniques générés permettent une première étape de déduplication des produits, comme c'est le cas en première ligne par exemple.

TABLE 4.5 – Extrait de l'ensemble de données après génération de l'identifiant unique de produit

UID produit	Généré	Code barre	Libellé unique de famille produit	Libellé unique de produit
e23...64f	Oui	['310...773' '310...684' '310...473']	['Bar 10' 'BAR 10']	['SIROP' 'Sirop']
TKPIN025	Non	['TKPIN025']	['Ticket Premium']	['Ticket Premium 25€']
df9...ced	Oui	['310...042' '310...603' '310...530']	['Timbres Poste' 'Timbres Postaux']	['Timbre Rouge' ...]
5fc...9be	Oui	['310...673' '310...424']	['Boissons Chaude' 'Bar 10,0']	['Cafe Allonge' 'Cafe allonge']
743...c24	Oui	['310...196' '310...503']	['Articles Fumeurs' 'Briquets']	['BRIQUET CLIPPER']
3057060366703	Non	['3057060366703']	['Feuilles Filtres' ... 'Articles Fumeurs']	['TUBE OCB 100' ... 'OCB 100 Tubes']
032...dd04	Oui	['310...0448' '310...0387']	['Bar 10']	['NOISETTE' 'Noisette']
42068549	Non	['42068549']	['Papier a rouler'... 'Articles Fumeurs']	['Rizla Micron' ...'RIZLA +Micron']
202...6a1	Oui	['310...110' '310...285']	['Timbres Poste']	['Timbre Vert' 'TIMBRE VERT']
7f9...9d1	Oui	['310...370' '310...810']	['Boissons a emporter 5,5' 'Boisson sans alcool']	['EAU 50CL' 'eau 50cl']
3057060362903	Non	['3057060362903']	['Tabletterie' 'Articles Fumeurs' 'TUBES']	['OCB250' 'TUBES 250' ... 'ocb 250']
80310839	Non	['80310839']	['Confiserie 20' 'Confiserie 20,0']	['Tic Tac Orange' 'TICTAC CITRON']
40111445	Non	['40111445']	['Confiserie 5,5' ... 'VTE 5.5']	['"Mm's sachet 45g" ... 'M and M s']

La quatrième étape liste les libellés de famille et les taux de TVA pour chaque identifiant unique de produit. Elle calcule ensuite le nombre d'occurrences de chaque mot-clé prédéfini (n-gramme) dans cette liste, et pour chaque taux de TVA. Le mot-clé le plus fréquent est utilisé pour identifier la catégorie. Une illustration du résultat de cette étape est visible dans le tableau 4.6. En cas d'ambiguïté, la TVA peut permettre au transformateur de prendre une décision plus éclairée quant à la classification du produit (par exemple, avec les taux de TVA différents des boissons alcoolisées et non alcoolisées).

TABLE 4.6 – Extrait de l'ensemble de données après génération des occurrences de valeurs pour chaque identifiant unique de produit

UID Produit	**Occurrence libellé famille**	**Occurence TVA**
9782290138106	('librairie' : 2)	(5.5 : 2)
8436545615423	('divers' : 2)	(20.0 : 2)
000003987667	('cbd' : 2)	(5.5 : 2)
9791028515508	('librairie' : 2)	(5.5 : 2)
9782758542629	('librairie' : 1, 'cartes' : 1, 'routieres' : 1)	(5.5 : 1, 10.0 : 1)
425...65d7	('viennoiserie' : 1, 'patisserie' : 1)	(5.5 : 2)
4895167903105	('cadeau' : 1, 'jouets' : 1)	(20.0 : 2)
3760344741102	('divers' : 1, 'mj' : 1, 'fumeurs' : 1)	(10.0 : 2, 20.0 : 1)
3560070278046	('pates' : 1, 'alimentaires' : 1, 'alimentation' : 1)	(5.5 : 2)
9782070369652	('librairie' : 2)	(5.5 : 2)

La cinquième et dernière étape correspond au cœur décisionnel de notre transformateur. En se basant sur les compteurs d'occurrences des mots, le thésaurus de mots-clés et un ensemble de règles de traitement, le transformateur va classifier le produit dans l'une des 61 catégories de produit (incluant la catégorie "inconnu"). Un exemple de résultats du transformateur est visible dans le tableau 4.7. Dans cet exemple, nous pouvons notamment noter en 3^{e} ligne un produit pour lequel la classification était ambiguë (le mot clé "bar" peut correspondre autant à des boissons alcoolisées que non alcoolisées), cette ambiguïté a été levée grâce aux occurrences des taux de TVA.

TABLE 4.7 – Extrait de l'ensemble de données après prédiction basée sur les occurrences et le thésaurus

UID Porduit	occurence	occurence vat	predicted type level 2
3256477081036	('epicerie' : 4, 'alimentation' : 1)	(5.5 : 5)	Epicerie hors boisson
3760283580404	('liquidarom' : 2, 'e' : 1, 'liquide' : 1)	(20.0 : 2)	E liquide
561...954	('bar' : 2)	(20.0 : 2)	Boi. alc. sur place
9782266294362	('librairie' : 3, 'lp' : 1)	(5.5 : 3)	Librairie
3662572707014	('e' : 1, 'liquide' : 1, 'ca' : 1, 'eliquides' : 1)	(20.0 : 2)	E liquide
7638900950021	('piles' : 4, 'divers' : 1, 'papeterie' : 1)	(20.0 : 6)	Pile
8000300408942	('confiserie' : 7, 'alimentation' : 1)	(5.5 : 5, 20.0 : 3)	Confiserie
8001348103370	('papeterie' : 2, 'magasin' : 1)	(20.0 : 2)	Papeterie
9782757871799	('librairie' : 5)	(5.5 : 4, 2.1 : 1)	Librairie

Le résultat de ces étapes est une correspondance des produits aux différentes catégories définies par les experts métier. Nous pouvons alors utiliser cette correspondance avec les données de ventes pour identifier les activités des commerces. Dans notre cas d'utilisation, ThesaurusBT catégorise 94% des produits apparaissant dans le jeu de données. Il convient de noter que ce processus ne garantit pas une catégorisation correcte des produits, et son exactitude est difficile à évaluer en raison de l'absence de jeu de données annoté. Cependant, une vérification manuelle a été effectuée par un expert sur un millier de produits, révélant un taux d'erreur très faible (inférieur à 1% pour les données étiquetées). Nous considérons cette catégorisation utilisable en

pratique si, lorsqu'elle est incluse comme étape de transformation, elle améliore l'exactitude d'un autre flux de travail de classification (en l'occurrence la caractérisation des activités commerciales des commerces dans ce cas d'étude). Cette validation est décrite dans nos expériences, où nous avons comparé ThesaurusBT avec des méthodes existantes.

5 Conception d'expérimentations basées sur notre banc d'essai

L'originalité de notre approche réside dans l'étape de transformation des données que nous utilisons pour la tâche d'étiquetage multiple (identification des activités commerciales des commerces). Par conséquent, l'objectif des expériences est de comparer l'efficacité des approches d'apprentissage automatique de base avec notre méthode sur l'étape de transformation des données. Nous supposons que la performance de l'étape de transformation des données a un impact direct sur l'exactitude des résultats de l'étiquetage multiple des activités des commerces.

Cette section offre un rappel des ensembles de données utilisés dans cette étude, ainsi que des mesures d'évaluation employées pour évaluer les performances des méthodes. Nous décrivons également en détail la configuration expérimentale utilisée dans nos expériences. Il est important de préciser que nous nous basons sur le banc d'essai abordé dans le chapitre 3. Plus de détails sur les ensembles de données ainsi que les choix de mesures sont donc disponibles dans le chapitre susmentionné.

5.1 Jeux de données et métriques d'évaluation

Nous utilisons et fournissons deux ensembles de données réelles fournies par l'entreprise Orisha Retail Shops.

Le jeu de données des ventes est composé de données de ventes de 2325 commerces anonymisés sur une période d'un an. La période étudiée n'est pas divulguée pour des raisons de confidentialité. Cet ensemble de données contient sept variables : un identifiant de commerce, un code-barres de produit, un libellé de produit, un identifiant de famille, un libellé de famille, le montant total vendu pendant l'année et le taux de TVA.

Le jeu de données des commerces étiquetés est composé de 400 commerces dont les activités ont été annotées par des experts. Cet ensemble de données contient deux variables : un identifiant de commerce et des étiquettes d'activités. Il existe 12 étiquettes d'activités possibles, seules 9 sont présentes dans le jeu de données de validation (les autres sont anecdotiques). 29 combinaisons uniques de ces 9 étiquettes sont présentes dans cet ensemble de données.

L'évaluation des performances d'étiquetage multiple est effectuée en calculant les scores suivants : l'exactitude (accuracy) et le score macro F_1. Le temps d'exécution et de mise en œuvre sont eux aussi pris en compte.

La description complète de ces deux jeu de données, des statistiques descriptives, des exemples et des explications sur le choix des critères d'évaluations sont fournis dans le chapitre 3.

5.2 Méthodes comparées et protocole expérimental

Nous divisons nos expériences en trois solutions intégrées : une *approche de base* **(1)**, une *approche de la littérature* **(2)** et une *approche spécifique aux besoins de l'entreprise* **(3)**. Ces approches sont illustrées dans la figure 4.3 et décrites plus en détail dans cette section. Pour nos expériences, nous sélectionnons aléatoirement 300 des 400 commerces annotés pour l'ensemble de données d'entraînement et les 100 restants pour l'ensemble de données de test. Les hyper-paramètres de sélection des caractéristiques de CatBOOST, de la forêt d'arbres de décisions (*Random Forest ou RF*) et de la machine à vecteurs de support (*Support Vector Machine ou SVM*) sont optimisés pour chaque flux de travail par un algorithme de recherche en grille (*Grid Search*) avec validation croisée sur l'ensemble d'entraînement (*Cross Validation*).

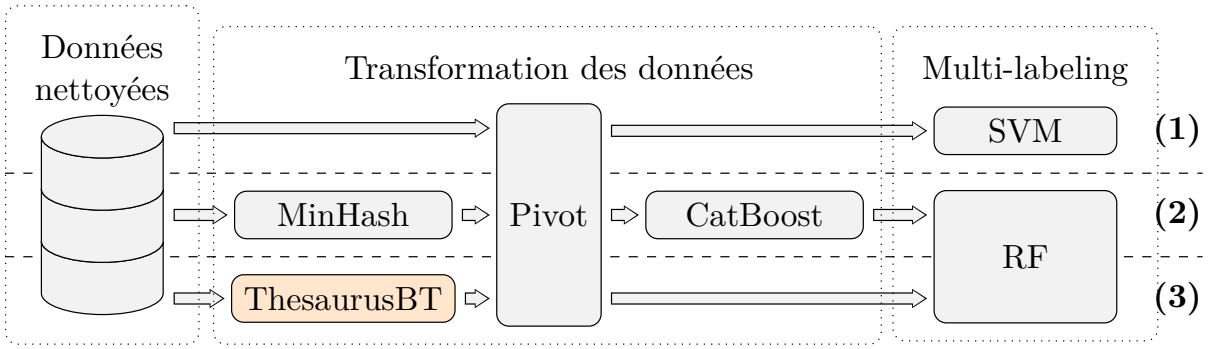


FIGURE 4.3 – Étapes de traitement expérimentées

L'approche de base est un flux de travail en trois étapes utilisant des méthodes de référence.

— La première étape consiste à nettoyer les champs textuels du jeu de données des ventes des commerces (noms des produits et étiquettes des familles de produits). Cette étape de nettoyage comprend la conversion en minuscules, la suppression des espaces supplémentaires, des accents, des caractères spéciaux et des mots inutiles (*stop words* en anglais), ainsi que la lemmatisation (passage au masculin singulier).

— La deuxième étape est la transformation de notre jeu de données en une table pivot, avec l'identifiant du commerce comme clé, les étiquettes des familles de produits comme colonnes et la somme des quantités vendues comme valeurs. En particulier, les valeurs des familles de produits qui ne sont pas vendues dans le commerce correspondant sont remplies de zéro.

— La troisième étape est la tâche d'étiquetage multiple, qui est réalisée par un modèle de Pertinence Binaire (*Binary Relevance*) avec un SVM comme classificateur de base. Pour entraîner ce modèle, nous fusionnons le tableau résultant de la deuxième étape avec le jeu de données d'entraînement (contenant les commerces et les étiquettes qui leur ont été

données par les experts métier), nous utilisons toutes les colonnes des familles de produits comme caractéristiques (nettoyées et agrégées) et l'étiquette d'activité du commerce comme valeurs cibles pour entraîner notre classificateur. Pour évaluer ce modèle, nous effectuons la même tâche que dans le processus d'entraînement précédent avec le jeu de données de validation et laissons notre modèle entraîné pour de l'étiquetage multiple faire des prédictions sur ces données.

L'approche de la littérature se compose de cinq étapes, utilisant les méthodes ayant obtenu les meilleures performances dans la littérature.

— La première étape est le nettoyage des champs textuels du jeu de données des ventes des commerces (identique à l'approche de base).

— La deuxième étape est l'utilisation de la technique d'encodage MinHash sur les libellés des familles de produits, l'objectif étant de réduire le nombre de différentes familles de produits car certaines ne diffèrent que légèrement par l'orthographe.

— La troisième étape est la transformation de notre jeu de données en une table pivot, comme pour l'approche de base, en utilisant l'identifiant du commerce comme clé et les signatures des familles de produits générées par MinHash comme colonnes.

— La quatrième étape est la sélection des caractéristiques, en utilisant le modèle CatBOOST pour sélectionner les caractéristiques les plus importantes (signatures des familles de produits) pour chaque étiquette d'activité de commerce.

— La cinquième étape est la tâche d'étiquetage multiple, qui est réalisée par un modèle de Pertinence Binaire avec une forêt d'arbres décisionnels comme classificateur de base. Cette étape est la même que la troisième étape de l'approche de base, mais dans ce cas, le jeu de données utilisé pour l'entraînement et la validation est beaucoup plus simple, car nous avons conservé uniquement les caractéristiques importantes selon notre modèle CatBOOST.

L'approche spécifique à l'entreprise comprend quatre étapes dont ThesaurusBT.

— La première étape est le nettoyage des champs textuels du jeu de données des ventes des commerces (identique à l'approche de base).

— La deuxième étape est l'application de la méthode ThesaurusBT, résultant en un dictionnaire de produits et de types de produits.

— La troisième étape consiste à transformer le jeu de données résultant de la deuxième étape en une table pivot, comme pour l'approche de la littérature, en utilisant l'identifiant du commerce comme clé et le type de produit prédit (étiqueté par la méthode ThesaurusBT) comme colonne.

— La quatrième étape est la tâche d'étiquetage multiple, qui est réalisée par un modèle de Pertinence Binaire avec RF comme classificateur de base. Cette étape est la même que la troisième étape de l'approche de base, mais dans ce cas, le jeu de données utilisé pour la

TABLE 4.8 – Nomenclature

Transformation	Abréviation	Classification	Abréviation
Pivot seulement	pv	Machine à vecteurs de support	svm
MinHash + pivot	mh	Forêt d'arbres décisionnels	rf
ThesaurusBT + pivot	th		
CatBOOST	cb		

formation et la validation est beaucoup plus simple, car nous avons conservé uniquement les types de produits étiquetés par ThesaurusBT.

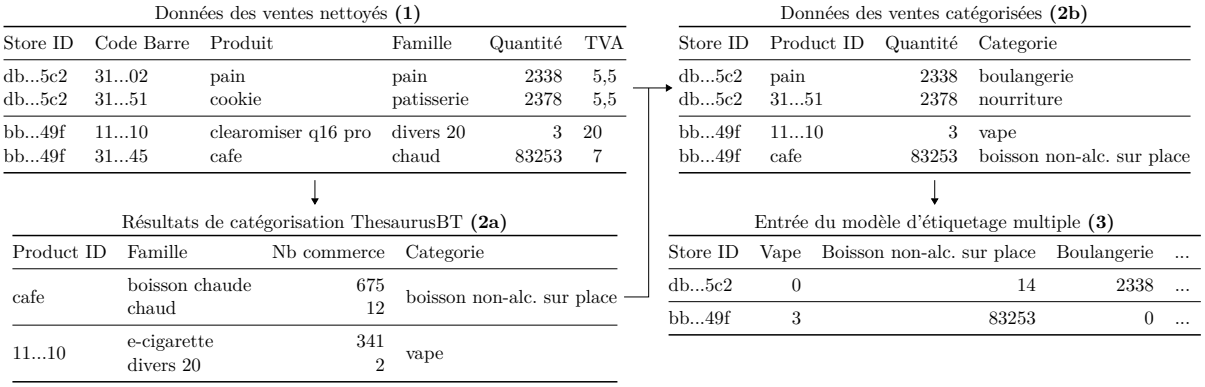


FIGURE 4.4 – Etapes de transformation du flux de travail spécifique à l'entreprise

Des flux de travail supplémentaires sont mis en œuvre afin d'obtenir des résultats plus explicatifs. Ces flux de travail supplémentaires mélangent les étapes des trois approches expliquées plus haut, à l'exception de certaines combinaisons non pertinentes. Nous avons par exemple réalisé l'approche de base mais en utilisant une forêt d'arbres décisionnels à la place d'une machine à vecteurs de support comme classificateur. Une description de la nomenclature des noms de flux de travail peut être trouvée dans le tableau 4.8. Cette nomenclature est utile pour comprendre les résultats de nos expériences.

5.3 Optimisation des paramètres

Afin d'obtenir de meilleures performances, nous avons procédé à l'optimisation de certains paramètres pour les modèles.

Forêt d'arbres décisionnels Le classificateur de base (pour l'étiquetage multiple) a des paramètres optimisés qui sont la profondeur maximale des arbres et le nombre d'arbres de décision du modèle. Ces paramètres ont été optimisés en testant toutes les combinaisons possibles à l'aide d'un algorithme de recherche en grille incluant la validation croisée des résultats.

Sélection de caractéristiques CatBOOST Le modèle inclut un paramètre important qui peut être optimisé : le nombre de caractéristiques sélectionnées pour la prédiction de chaque

étiquette (activités des commerces). Les valeurs testées sont 5, 6, ..., 10, 20, 30, ..., 50, 75, 100, 200, ..., 400 pour les flux de travail avec transformation simple Pivot ou MinHash et Pivot ; et 2, 3, ..., 15, 20, 30, ..., 50 pour les flux de travail avec transformation ThesaurusBT. Les paramètres sélectionnés sont respectivement 8, 7 et 9 [1].

6 Résultats expérimentaux et mise en production

Nous commençons par présenter et analyser les résultats des expérimentations présentées précédemment.

6.1 Présentation et analyse des résultats expérimentaux

La figure 4.5 présente les temps d'exécution les plus élevés des étapes clés des différents flux de travail. Les étapes non listées ont un temps d'exécution comparativement beaucoup plus bas, moins d'une minute. Ces temps ont été calculés en exécutant le flux de données complet, implémenté en Python, sur un CPU i7-11800H à 2.30GHz et 16GB de RAM. Les temps d'exécution incluent à la fois l'entraînement et l'évaluation.

Le temps d'exécution total de chaque flux de travail ne dépasse pas une heure, et est donc bien inférieur à la limite de douze heures. Le flux de travail de la littérature, qui inclut la méthode de codage MinHash, obtient un temps d'exécution significativement plus élevé que les autres en raison de l'étape de calcul de hash de la méthode MinHash (52 min). La méthode ThesaurusBT (9 min) pour l'étape de préparation des données est également plus lente qu'une simple méthode de pivotement (3 s).

Les temps de mise en œuvre sont une autre dimension importante, comme expliqué précédemment. Ces temps ont été mesurés depuis le début de la mise en œuvre des approches jusqu'à la fin (y compris le réglage des hyper-paramètres, etc.), avec une solution clé en main prête à être exécutée sur le jeu de données étiqueté. Les temps eux-mêmes ne peuvent pas vraiment être utilisés (ils dépendent du niveau du développeur, etc.), seuls les temps relatifs ont été considérés (le temps le plus court a été fixé à zéro et les autres sont calculés avec les différences en heures par rapport au temps le plus court). Ces temps sont listés dans la figure 4.6. L'approche la plus longue à mettre en œuvre est notre approche car elle nécessite de définir les connaissances métier requises. La méthode MinHash [Cerda, 2020] a également pris beaucoup de temps à mettre en œuvre car aucune implémentation de niveau professionnel existante n'a été trouvée.

1. Les performances de l'optimisation des paramètres sont évaluées sur la base des résultats des scores d'exactitude et de macro F1 de l'étape d'étiquetage multiple avec RF comme classificateur de base.

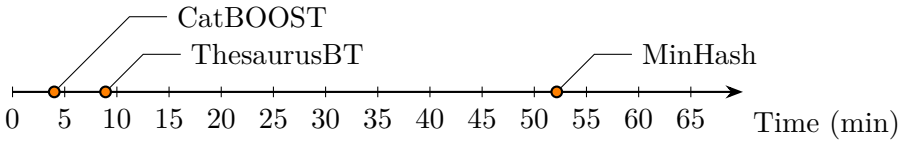


FIGURE 4.5 – Temps d'exécution des fonctions

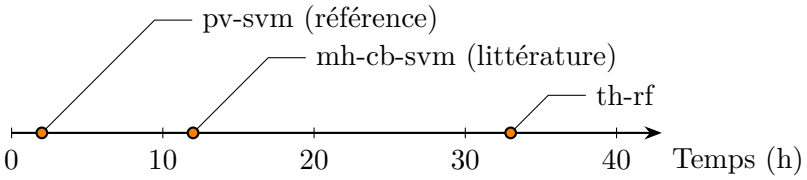


FIGURE 4.6 – Temps d'implémentation estimés totaux

TABLE 4.9 – Résultats de l'étiquetage multiple

Transformation de données	Classifieur	Dimensions des entrées du classifieur	Exactitude	Macro F1	Temps d'exécution (min)
Pivot	RF	6574	0,66	0,41	0,41
	SVM	6574	0,69	0,51	0,23
MinHash–Pivot	RF	5104	0,67	0,42	52,18
	SVM	5104	0,68	0,51	52,16
Pivot–CatBOOST	RF	55	0,78	0,46	5,4
	SVM	55	0,75	0,53	5,23
MinHash–Pivot–CatBOOST	RF	52	0,82	0,49	57,18
	SVM	52	0,79	0,55	57,02
ThesaurusBT–Pivot	RF	61	**0,83**	0,59	9,16
	SVM	61	0,79	**0,63**	9,01
ThesaurusBT–Pivot–CatBOOST	RF	37	0,80	0,60	14,33
	SVM	37	0,79	**0,63**	14,16

L'exactitude et le score macro-F1 de la tâche d'étiquetage multiple sont présentés dans le tableau 4.9. En examinant les résultats des flux de travail qui incluent la méthode MinHash, nous pouvons constater que cette méthode n'améliore pas significativement leur performance, et parfois elle peut même avoir un impact négatif. Cette méthode réduit la complexité du jeu de données de 6574 à 5104 variables. Cette réduction de la complexité peut expliquer la légère amélioration des performances des flux de travail utilisant cette méthode conjointement avec un modèle d'étiquetage multiple utilisant la pertinence binaire et une forêt d'arbres décisionnels comme classificateur de base sans le modèle CatBoost (les performances sont réduites lors de l'utilisation du modèle CatBoost). En revanche, cette méthode a significativement diminué la performance des flux de travail lorsqu'elle est utilisée avec un modèle d'étiquetage multiple basé sur la pertinence binaire et une machine à vecteurs de support comme classificateur de base. Nous pouvons présumer que cette méthode ne réussit pas à améliorer de manière significative les performances de la tâche d'étiquetage multiple en raison de la duplication des noms de produits comportant de multiples variantes morphologiques (fautes de frappe, orthographe incorrecte, etc.) et impacte le temps d'exécution.

Étant donné que notre jeu de données contient non seulement des fautes de frappe et des erreurs d'orthographe, mais aussi diverses appellations pour les mêmes éléments (synonymes), la méthode MinHash peine à traiter les noms de familles de produits. Elle ne permet donc pas d'effectuer efficacement la phase d'encodage des variables textuelles liées aux produits vendus (libellés de produits et de familles de produits), car cette méthode ne capture pas toutes les duplications dans notre jeu de données. De plus, en augmentant le seuil de sensibilité, MinHash

a tendance à inclure davantage de variantes d'écriture pour une même catégorie, ce qui peut également conduire à regrouper des familles de produits sans liens réels entre elles.

En considérant le modèle CatBOOST, nous observons qu'il peut à la fois améliorer de manière significative les performances des flux de travail et les affecter négativement. Les performances sont améliorées en utilisant le modèle CatBOOST en plus de Pivot ou de MinHash et du modèle de pertinence binaire avec des forêts d'arbres décisionnels comme classificateurs de base. Nous constatons que l'exactitude et le score macro-F1 sont améliorés de 25% dans le flux de travail Pivot–CatBOOST–RF par rapport au flux de travail Pivot–RF. Ce sont des résultats positifs, étant donné que cette méthode réduit considérablement la complexité de l'ensemble de données de 6574 à 530 variables. Nous pouvons donc dire que ce type de modèle basé sur le boosting de gradient semble capturer efficacement les variables les plus importantes. Néanmoins, les performances sont réduites lorsqu'elles sont utilisées conjointement avec le modèle de pertinence binaire avec des machines à vecteurs de support comme classificateur de base. Le modèle de pertinence binaire avec SVM semble mieux gérer l'ensemble de données de haute dimension par rapport au modèle de pertinence binaire avec RF comme classificateur de base. Ce résultat pourrait être différent si nous augmentions la profondeur des arbres dans les forêts d'arbres décisionnels, mais nous risquerions alors de provoquer un sur-apprentissage. Finalement, l'utilisation du modèle CatBOOST avec la méthode de prétraitement ThesaurusBT et le modèle de pertinence binaire avec RF comme classificateur de base est un compromis qui augmente le score d'exactitude tout en diminuant le score macro-F1. Cela s'explique par la métrique que le modèle CatBOOST tente d'optimiser lors de son entraînement, l'exactitude.

Une conclusion à tirer de ces résultats est que la méthode ThesaurusBT utilisée avec le modèle de pertinence binaire et RF comme classificateur de base surpasse toutes les autres, en particulier pour le score macro-F1. Sur la base de ces performances, nous pouvons supposer que cette méthode a réussi à étiqueter convenablement une partie significative des produits, résolvant ainsi partiellement le problème de transformation des données. Si nous nous concentrons uniquement sur le score macro-F1, les résultats sont très hétérogènes en raison de notre ensemble de données déséquilibré (il y a de nombreux tabacs et peu d'hôtels ou de restaurants, par exemple). En fait, les modèles sont sur-ajustés pour reconnaître les tabacs ou les journaux, mais sous-ajustés pour reconnaître les hôtels, les restaurants, etc. Cette phase de formation déséquilibrée a directement affecté ce score pour tous les modèles qui n'ont pas pu rapidement apprendre à détecter les cas rares. Les modèles les plus performants si nous considérons uniquement le score de macro-F1 sont ceux qui incluent la méthode ThesaurusBT dans leur flux de travail. Plus généralement, une amélioration significative est obtenue en intégrant la méthode ThesaurusBT dans les flux de travail, comme le montre le tableau 4.9, la solution spécifique à l'entreprise étant la plus efficace.

6.2 Discussion

Dans ces expérimentations, nous avons abordé le problème d'étiquetage multiple des activités des commerces en fonction de leurs ventes. Bien que nous ayons considéré un cas d'utilisation spécifique fourni par l'entreprise, il s'agit d'un problème général qui affecte les fournisseurs de logiciels de différentes industries. Comparé aux cas d'utilisation trouvés dans la littérature,

celui proposé dans ces expérimentations soulève des défis importants car les commerces peuvent utiliser n'importe quel libellé pour nommer leurs produits et leurs familles de produits. Comme base de référence, nous avons proposé et mis en œuvre deux approches : une avec des méthodes de base et une avec les méthodes les plus efficaces connues dans la littérature.

Nous avons réalisé ces implémentations à partir de zéro car elles sont rarement fournies avec les articles correspondants. Tous les ensembles de données et les implémentations sont disponibles en ligne à des fins de reproductibilité dans le dépôt sur la forge du LIAS [2]. Ces approches sont comparées à celle que nous avons proposée, nommée ThesaurusBT, une transformation basée sur un thésaurus. Cette transformation repose sur des connaissances métier qu'une entreprise comme Orisha Retail Shops peut fournir et maintenir dans le temps. Comme nous l'avons montré dans nos expérimentations, cette approche surpasse les approches de base pour la tâche d'étiquetage des activités des commerces utilisant des données de qualité hétérogène.

Comme inconvénient, l'intégration de connaissances métier dans le thésaurus et les règles personnalisées représente environ 35 heures de travail humain. Par conséquent, ThesaurusBT entraîne un coût de développement spécifique à l'entreprise significatif par rapport aux solutions d'apprentissage automatique pures. En rendant cet ensemble de données disponible, Orisha Retail Shops souhaite susciter l'intérêt pour ce type de problème. En effet, disposer d'une solution efficace d'apprentissage automatique avec une implication humaine limitée et qui serait de plus utilisable par une entreprise de taille moyenne, apporterait une amélioration significative à ce domaine.

Comme travail futur, nous prévoyons de tester des approches d'apprentissage profond. Une difficulté est d'avoir suffisamment de données d'entraînement. Ainsi, automatiser la production de telles données est une perspective de notre travail. Un autre défi est d'étendre les connaissances métier utilisées par notre approche. Actuellement, nous utilisons uniquement des ressources lexicales, mais nous sommes convaincus que des modèles plus complexes tels que des ontologies pourraient être utiles pour améliorer notre méthode.

6.3 Mise en production

Un point clé pour une entreprise est la mise en production le flux de travail sélectionné. Dans notre cas, ce flux de travail se décompose en cinq étapes clés :

1. La récupération des données de ventes agrégées en base de données.
2. Le pré-traitement de ces données, en particulier le nettoyage des libellés de produits et de familles de produits.
3. Le traitement des données par notre modèle ThesaurusBT permettant la classification des produits et faisant office d'encodeur pour l'étape suivante.
4. La classification (ou l'étiquetage multiple) des activités commerciales des commerces à partir des ventes de produits encodés et incorporés.
5. La mise à disposition des résultats en base de données.

2. https://forge.lias-lab.fr/thesaurusbt

Étant donné que ce flux de travail sera exécuté non pas sur un sous-ensemble des commerces comme c'était le cas dans nos expérimentations, mais sur l'ensemble des commerces, soit un peu plus de 7000 commerces sur une année, les volumes traités sont plus importants.

De plus, afin que notre approche d'étiquetage des produits (la solution ThesaurusBT permettant l'encodage) soit performante, nous devons l'exécuter sur un volume important de données, et donc sur une période de temps conséquente (au minimum 3 mois pour obtenir plus de 90% de taux d'étiquetage autre que « inconnu »). Nous nous sommes basés sur une exécution sur l'ensemble des commerces, et sur une période de ventes de 6 mois, représentant environ 50 millions de lignes dans le jeu de données des ventes agrégées par commerce et par produit en entrée.

Nous avons alors choisi, pour des contraintes de performance, de temps d'exécution, de consommation de mémoire vive et de robustesse, de déployer ce flux de travail en calcul distribué sur le cloud. Dans notre cas, nous avons choisi la technologie Spark pour la distribution, nécessitant une ré-implémentation du flux de travail en PySpark. Le fournisseur cloud vers lequel nous nous sommes orientés est Amazon Web Services pour des raisons de cohérence par rapport à d'autres traitements au sein de l'entreprise utilisant déjà ce fournisseur.

Le flux de travail, d'un point de vue technique, fonctionne alors en quatre étapes :

1. La récupération des données de ventes agrégées en base de données est effectuée par le service AWS Glue, le service s'exécute pendant environ 45 minutes.
2. Le pré-traitement de ces données, en particulier le nettoyage des libellés de produits et de familles de produits ; ainsi que le traitement des données par notre modèle ThesaurusBT permettant la classification des produits ; et la classification (ou l'étiquetage multiple) des activités commerciales des commerces à partir des ventes de produits encodés et incorporés est effectué par le service AWS EMR. Nous provisionnons ce service avec trois instances EC2 m5.4xl qui sont des instances 16vCore 64Gio RAM, le service s'exécute pendant environ 90 minutes. Les résultats de la classification des produits, donnés par ThesaurusBT, et de la caractérisation des activités des commerces, donnés par le flux de travail complet incluant ThesaurusBT et un modèle d'étiquetage multiple sont sauvegardés dans un conteneur S3.
3. L'export des résultats stockés dans le conteneur S3 est effectué par un service interne. Les résultats sont alors stockés dans une base de données interne à l'entreprise, permettant ainsi d'utiliser ces résultats dans les applicatifs internes à l'entreprise.

Un schéma présenté sur la figure 4.7 permet de visualiser de manière simplifiée l'architecture complète mise en place.

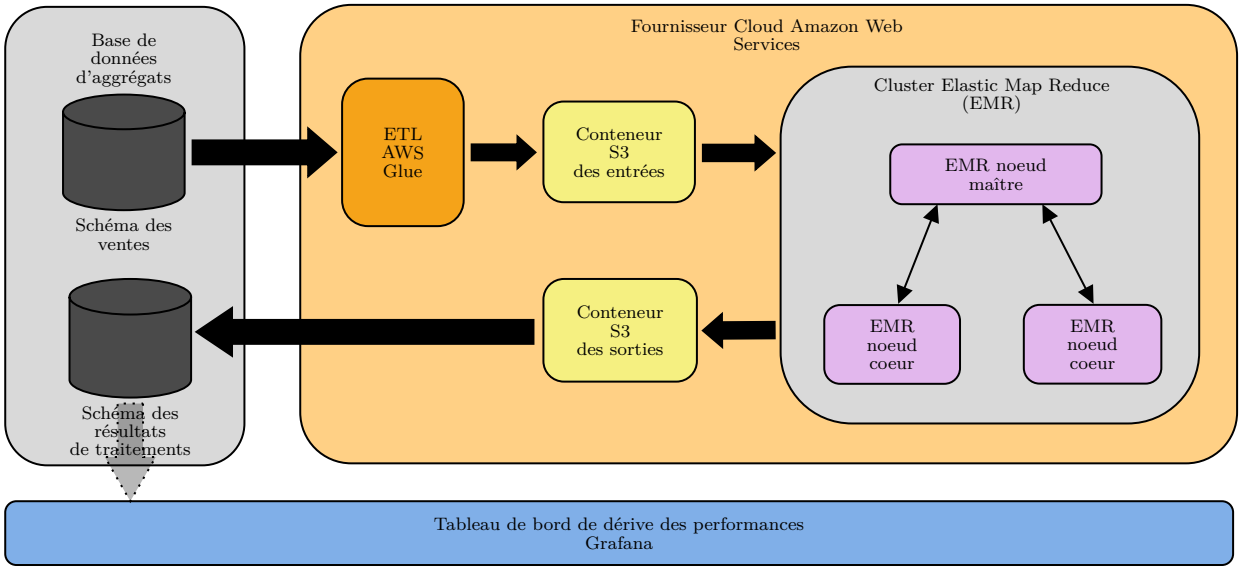


FIGURE 4.7 – Schéma architecture de déploiement

7 Conclusion et Perspectives

Nous pouvons tirer plusieurs enseignements des études menées dans ce chapitre. D'une part, les techniques d'apprentissage automatique généralistes ne sont pas toujours celles qui obtiennent les meilleurs résultats quand on y consacre la même charge de travail pour leur mise en place que pour le développement d'une solution spécifique. C'était particulièrement le cas dans notre contexte avec les solutions considérées. Il est donc important de ne pas considérer systématiquement les approches d'apprentissage automatique comme des solutions répondant à tous les problèmes.

L'approche de développer un transformateur de données qui, d'une part, répond à la problématique d'encodage de manière relativement efficace, et d'autre part, permet de répondre à une tâche de classification des produits utile pour d'autres usages est également importante pour l'entreprise. Orisha Retail Shops y a trouvé des usages tout aussi stratégiques que ceux de la demande initiale de caractérisation des activités des commerces.

L'explicabilité des résultats de l'encodage des produits (et donc de leur classification) est également un aspect crucial pour l'entreprise. En effet, cette donnée étant stratégique, il est souhaitable qu'un certain niveau d'explicabilité des résultats soit possible. Dans notre cas, la traçabilité de notre modèle permet une explicabilité complète des résultats, avec un cheminement complet des valeurs intermédiaires qui ont conduit à classer un produit dans une classe et pas une autre. Pour les mêmes raisons, sa maintenabilité est excellente. L'algorithme ne requiert pas d'expert en apprentissage automatique pour le faire évoluer ; il a été pensé pour pouvoir être mis à jour facilement en ajustant le thésaurus ainsi que la gestion des ambiguïtés.

Ces travaux ont eu un impact positif pour l'entreprise. Nous sommes maintenant en mesure de réaliser des études bien plus fines sur certains types de produits en particulier ou pour certains types d'activités commerciales. Les résultats de ces travaux sont particulièrement utilisés dans le cadre de la valorisation et de la diffusion de publicités ciblées sur les écrans des commerces

équipés de la solution Bimedia.

Jusqu'à présent, nous n'avons pas identifié de solution d'apprentissage automatique généraliste qui réponde à notre besoin. Nous espérions trouver une solution capable de traiter ces données de qualité très hétérogène de manière efficace et facilement. C'était l'objectif que nous pensions pouvoir relever au moment des premiers travaux de transformation à partir des grands modèles de langages pré-entrainés. Nous sommes convaincus que ces modèles ne se sont pas montrés suffisamment efficaces en grande partie à cause d'un entraînement moins conséquent sur la langue française. Nous continuons à explorer les différentes pistes des transformers, sachant que depuis nos travaux datant de l'année 2022, de nombreux modèles pré-entrainés open source ont été publiés, et que c'est un champ de la recherche qui a reçu une attention particulièrement élevée cette année.

Nous envisageons également pour le futur une approche hybride, qui déléguerait par exemple la construction, de tout ou partie, du thésaurus par un modèle de langage. L'objectif final est d'avoir un modèle qui fonctionne très bien pour répondre à notre problématique d'encodage des produits, mais qui pourrait également être appliqué à un autre contexte industriel que le nôtre.

Chapitre 5

Formalisation de la connaissance à travers les ontologies

Objectifs

Ce chapitre a pour objectif de présenter comment les ontologies, issues des technologies du web sémantique, peuvent formaliser et structurer les connaissances liées aux commerces, à leurs activités, ainsi qu'à leur environnement métier selon la perspective d'Orisha Retail Shops. Il propose une méthodologie permettant d'intégrer et de relier les données internes de l'entreprise à des sources externes, facilitant ainsi leur exploitation. En mettant l'accent sur la modélisation des concepts et l'optimisation des requêtes, ce chapitre explore également les performances de différentes architectures de gestion des graphes de connaissances afin d'améliorer les capacités d'analyse, les ressources techniques et la gestion des données, ainsi que d'autres aspects critiques.

Sommaire

1 Introduction

L'approche développée dans le chapitre précédent permet de classifier les produits vendus par les commerces du réseau Orisha Retail Shops. Sur le plan académique, elle se distingue par une classification plus précise que les méthodes de l'état de l'art. Cela est particulièrement vrai lorsqu'il s'agit d'encoder des données caractérisées par des libellés de produits à haute cardinalité, confrontés à des problèmes de synonymie et de variations morphologiques. D'un point de vue industriel, cette approche facilite l'identification des activités commerciales, ce qui permet à Orisha Retail Shops de mieux cibler et personnaliser les offres proposées à ses clients.

Cependant, les besoins d'analyse de l'entreprise dépassent la simple classification des activités commerciales. En particulier, un besoin récurrent consiste à identifier une liste de points de vente en fonction de critères tels que le volume de produits vendus dans une catégorie spécifique, la localisation géographique des points de vente, ou encore les caractéristiques démographiques (âge, répartition des salaires, etc.) dans les environs du magasin. Répondre à ce type de requête pose deux défis majeurs.

Premièrement, cette analyse repose sur des concepts clés comme le volume de ventes ou la catégorisation des commerces, qui ne sont pas explicitement définis dans les systèmes d'information de l'entreprise. Les données associées à ces concepts sont en effet disséminées à travers différentes tables, voire gérées par des systèmes de bases de données distincts. La réalisation de telles études nécessite donc systématiquement un prétraitement pour expliciter ces concepts, ce qui est non seulement chronophage, mais également source potentielle d'erreurs. La modélisation explicite de ces concepts dans un modèle dédié serait donc un atout.

Deuxièmement, ce type d'analyse nécessite l'intégration de données externes à Orisha Retail Shops pour caractériser l'environnement des commerces, en s'appuyant sur des bibliothèques de données ouvertes ou des interfaces de programmation d'applications (API) telles que Google API [1] ou Societe.com [2]. L'intégration de ces données externes, souvent complexes et parfois mal structurées, présente des défis spécifiques, d'autant plus que ces données sont susceptibles d'évoluer indépendamment du système d'information d'Orisha Retail Shops. L'objectif est donc de pouvoir exploiter ces données externes en les liant de manière cohérente à celles de l'entreprise.

Prenons un exemple concret pour illustrer les deux difficultés évoquées précédemment. Parmi les données externes fréquemment utilisées figurent celles de l'INSEE [3] concernant l'appartenance des villes à des aires d'influence. La solution directe consiste à importer ces données sous forme d'une table dans la base de données qui inclut également les ventes et les détails des commerces clients de la solution Bimedia. Pour déterminer le nombre de commerces actifs en 2023 par ville dans une aire d'influence donnée, il serait nécessaire d'effectuer les traitements suivants :

- une requête sur la table des ventes pour récupérer les commerces actifs (ceux ayant réalisé au moins une vente durant une période donnée) ;
- une jointure avec la table des commerces pour obtenir des informations supplémentaires sur ceux-ci ;

1. https://cloud.google.com/apis/docs/overview?hl=fr
2. https://www.societe.com/
3. https://www.insee.fr/fr/information/4803954

- une jointure avec la table des commerces ayant refusé l'utilisation de leurs données, pour les exclure de l'analyse ;
- une jointure avec la table des adresses ;
- une jointure avec la table des villes (les villes étant stockées séparément en raison de leur évolution) pour récupérer le code INSEE correspondant ;
- une jointure avec la nouvelle table des aires d'influence.

Outre la complexité des requêtes, se pose le problème de la mise à jour des données externes, qui évoluent indépendamment du système d'information d'Orisha Retail Shops. Un autre problème réside dans la réutilisation des traitements effectués. La requête précédente permet d'identifier les commerces actifs. Si ce même concept est nécessaire dans une autre étude, il serait pertinent de conserver cette requête dans la base de données, par exemple sous la forme d'une vue. Cependant, c'est rarement le cas en pratique, car les études sont souvent ponctuelles et il est difficile d'identifier les parties des traitements qui pourraient être pertinentes pour des études futures. De plus, les mécanismes des bases de données, tels que les vues, offrent peu de moyens pour définir précisément un concept. En pratique, seul un nom est généralement attribué sans commentaires explicatifs, ce qui nuit à la réutilisabilité. La problématique ne réside donc pas uniquement dans le stockage simple des données externes, mais plutôt dans leur organisation efficace, afin qu'elles soient compréhensibles et exploitables sur le long terme. Ce défi n'est donc pas seulement technique, mais concerne également l'organisation et la gestion des connaissances liées à ces données.

Formaliser les concepts d'un domaine d'étude et les lier à d'autres sources est l'objectif des recherches menées au cours des dernières décennies dans le domaine du web sémantique. Au cœur de ces recherches se trouve le concept d'*ontologie*, une modélisation formelle des connaissances qui articule des entités et leurs relations. Les ontologies permettent de définir les concepts et vocabulaires d'un domaine d'étude avec une expressivité accrue par rapport aux mécanismes traditionnels des bases de données. Le W3C a établi des standards pour la création et l'interrogation des ontologies[4], comme RDF-Schema et SPARQL. Ceux-ci sont mis en œuvre dans des systèmes de gestion de bases de données spécifiques, appelés triplestores. L'émergence de ces standards et technologies a permis le développement d'ontologies non seulement dans le domaine académique, comme *YAGO* [Hoffart, 2013] et *DBpedia* [Lehmann, 2015], mais aussi dans un contexte industriel tels que celles conçus par Google (*Knowledge Vault*) [Dong, 2014] ou Walmart [Deshpande, 2013].

L'utilisation des technologies du web sémantique pour expliciter les concepts impliqués dans les études menées par Orisha Retail Shops semble donc pertinente. Cependant, bien que les fondements théoriques de ces technologies aient été largement explorés, peu d'études ont démontré leur mise en œuvre pratique dans des cas d'utilisation réels et fourni un retour d'expérience sur leur capacité à évoluer à grande échelle. En effet, les recherches sur les performances de ces technologies se concentrent souvent sur elles de manière isolée. Par exemple, des bancs d'essai ont été développés pour évaluer la performance des triplestores [Guo, 2005a ; Aluç, 2014a]. Dans

4. https://www.w3.org/2001/sw/

un contexte industriel, une étude plus globale est nécessaire, prenant en compte l'ensemble de l'architecture utilisée pour la mise en œuvre des technologies sémantiques.

Pour répondre à ce besoin, ce chapitre propose deux contributions principales. La première est la présentation d'une méthodologie pour la mise en œuvre des technologies sémantiques dans un scénario réel. Nous détaillons l'architecture, les outils et les stratégies de mise en œuvre que nous proposons. Bien que cette méthodologie soit illustrée dans le contexte du commerce de détail, nous pensons qu'une approche similaire pourrait être adaptée à d'autres domaines. La deuxième contribution est l'évaluation expérimentale de la performance de différentes implémentations architecturales des technologies du web sémantique. Nous proposons diverses architectures pour la mise en œuvre d'un scénario réel chez Orisha Retail Shops et analysons expérimentalement l'impact de ces choix sur les performances.

Ce chapitre est organisé comme suit. La section 2 définit les concepts fondamentaux du web sémantique nécessaires à la compréhension de la méthodologie de mise en œuvre que nous proposons. La section 3 présente le cas d'étude réel que nous considérons et met en évidence les limites de l'état de l'art pour y répondre. La méthodologie de mise en œuvre des technologies du web sémantique est détaillée dans la section 4, où nous décrivons la conception des ontologies nécessaires, la démarche pour l'intégration des données externes et internes, ainsi que les possibilités d'interrogation offertes par cette méthodologie. La section 5 explore différentes implémentations architecturales possibles et propose une étude de performance de ces architectures. Enfin, la section 6 conclut ce chapitre en synthétisant nos contributions à l'application des technologies du web sémantique dans un contexte industriel.

2 Notions préliminaires sur le web sémantique

Nous présentons dans cette section les concepts du web sémantique nécessaire à la compréhension de ce chapitre.

2.1 Ontologies et Graphes de Connaissances

Gruber a défini une ontologie comme *une spécification explicite d'une conceptualisation* [Gruber, 1993]. Une ontologie présente trois caractéristiques principales :

- *Formelle* : une ontologie définit un ensemble de concepts sur un domaine donné. Ces définitions reposent sur un *langage d'ontologies*. Défini formellement, il permet de vérifier la consistance de l'ontologie conçue et de réaliser des raisonnements automatiques sur ses concepts.
- *Consensuelle* : une ontologie est acceptée et partagée par une communauté. Contrairement à un modèle conceptuel qui est généralement créé pour une application spécifique, une ontologie vise à capturer et structurer les connaissances d'un domaine, répondant ainsi aux besoins communs de la communauté concernée.
- *Identification universelle* : chaque concept au sein d'une ontologie est associé à un identifiant unique et universel, tel qu'une *URI (Uniform Resource Identifier)*. Cet identifiant

permet de référencer un concept et la sémantique associée, indépendamment du contexte d'utilisation ou de l'environnement.

Une ontologie définit un ensemble de concepts qui peuvent être instanciés. Dans ce manuscrit, nous considérons le terme *ontologie* comme désignant exclusivement l'ensemble des concepts qu'elle décrit, c'est-à-dire le schéma. En revanche, lorsqu'il s'agit de faire référence à l'ontologie ainsi qu'à ses instances, nous utilisons le terme *graphe de connaissances* qui est couramment utilisé, en particulier dans l'industrie [Zou, 2020a]. Enfin, nous utiliserons le terme *couche sémantique* pour désigner l'ensemble des modèles, outils et langages qui permettent de créer des graphes de connaissances à partir d'un ensemble de sources de données.

2.2 Principales notions liées aux ontologies et graphes de connaissances

Les principaux éléments qui constituent les ontologies et les graphes de connaissances sont les suivants.

- *Classes :* des catégories ou types d'objets du domaine considéré. Par exemple, dans notre contexte, les classes `Store` et `Product` représentent respectivement les commerces et les produits.
- *Instances :* représentent des entités individuelles ou objets appartenant à une classe. Les instances sont créées lors du peuplement du graphe de connaissances et sont basées sur le schéma défini par l'ontologie. Par exemple, `Petit commerce` pourrait être une instance de la classe `Store`.
- *Relations :* des liens entre les instances ou les classes. Par exemple, la relation `Sell` (vend) peut être établie entre les classes `Store` et `Product`, indiquant qu'un commerce vend un produit.
- *Propriétés :* des attributs ou caractéristiques d'une instance. Pour la classe `Product`, les propriétés pourraient inclure `productName`, `price` et `productDescription`.
- *Axiomes :* des assertions ou règles, considérées comme vraies, qui permettent de déduire de nouvelles connaissances. Les axiomes incluent souvent des notions telles que les sous-classes. Par exemple `SubClassOf( a:TobbacoStore a:Store )` signifie que chaque instance de la classe `TobaccoStore` est également une instance de la classe `Store`.

Ces différents éléments sont définis avec un langage d'ontologies. Nous décrivons dans ce qui suit les langages que nous avons considérés dans nos travaux.

2.3 Langages d'ontologies considérés

RDF-Schema (ou RDFS) [Brickley, 2014] est une extension du langage RDF (Resource Description Framework). RDF permet de formuler des assertions sous forme de triplets *(sujet, prédicat, objet)*, où le sujet est une URI désignant une ressource, le prédicat représente une propriété ou une relation associée à cette ressource, et l'objet correspond à la valeur de cette propriété ou à l'URI d'une autre ressource en cas de relation. RDFS enrichit RDF en introduisant un ensemble de constructeurs permettant de définir des ontologies. Grâce à RDFS, il est possible de spécifier des classes, des propriétés ainsi que leurs hiérarchies (`subClassOf` et `subPropertyOf`).

La définition des instances de classes, des valeurs de propriétés et des relations entre les instances s'appuie sur les triplets RDF. Une particularité de RDF est l'utilisation de nœuds blancs, qui permettent de représenter des URI ou des littéraux non spécifiés. Cette fonctionnalité est particulièrement intéressante dans notre contexte. En effet, certains clients de Bimedia exercent plusieurs activités, chacune pouvant être gérée par des caisses différentes, dont certaines ne sont pas maintenues par Orisha Retail Shops. Les nœuds blancs permettent, par exemple, de capturer le fait qu'un commerce exerce une activité sans avoir à spécifier le volume associé à cette activité.

OWL (Web Ontology Language) [Dean, 2004] étend les capacités expressives de RDFS. Ce langage se décline en trois versions : OWL Lite, OWL DL et OWL Full, chacune offrant un pouvoir d'expression croissant (OWL Lite $\subset$ OWL DL $\subset$ OWL Full), mais au prix de raisonnements de plus en plus coûteux. Ces versions introduisent de nouveaux constructeurs pour la définition d'ontologies. Par exemple, elles permettent de créer des classes définies comme l'union, l'intersection ou le complément d'autres classes. Une classe peut également être spécifiée comme une *restriction* d'une autre classe. Par exemple, la classe `TobaccoStore` peut être définie comme l'ensemble des instances de la classe `Store` qui possèdent au moins une valeur pour la propriété `tobaccoSale`.

Dans RDFS et OWL, les URI permettent d'identifier de manière unique les ressources. Ces identifiants facilitent la liaison des entités au sein des ontologies et des instances au sein des graphes de connaissances, renforçant ainsi l'interopérabilité et l'intégration des données à travers divers systèmes. Les URI permettent également de réaliser des liens entre les instances via des propriétés telles que *owl:sameAs*, ce qui enrichit les graphes de connaissances en connectant des entités similaires ou identiques issues de différentes sources. Les *Internationalized Resource Identifiers* (IRI) sont une extension des URI, permettant l'utilisation de caractères non ASCII pour supporter une plus grande variété de langues et de symboles, ce qui les rend particulièrement utiles dans un contexte mondialisé.

Dans nos travaux, nous avons principalement utilisé les constructeurs du langage RDFS. Cependant, nous avons également utilisé des constructeurs d'OWL DL quand le pouvoir d'expression de RDFS n'était pas suffisant pour le cas d'étude considéré. Nous présentons ci-après les capacités de raisonnement associées aux langages d'ontologie.

2.4 Raisonnement dans les graphes de connaissances

Les langages RDFS et OWL reposent sur l'hypothèse du *monde ouvert*, selon laquelle certains faits peuvent être vrais même s'ils ne sont pas explicitement représentés dans un graphe de connaissances. Cette hypothèse est particulièrement pertinente pour les études menées par Orisha Retail Shops. En effet, le système d'information de l'entreprise ne contient que les données des caisses qu'elle gère, tandis que de nombreux autres commerces utilisent des caisses non maintenues par Orisha Retail Shops. Lorsqu'il s'agit de réaliser des études et des prospections, l'hypothèse du monde ouvert prend donc tout son sens.

Sous l'hypothèse du monde ouvert, des triplets RDF peuvent être *implicites*, c'est-à-dire qu'ils sont considérés comme faisant partie d'un graphe de connaissances même s'ils ne sont pas

explicitement présents. Pour chaque langage d'ontologie, des règles de déduction sont spécifiées afin de déduire ces triplets implicites à partir des triplets explicites.

Exemple. Supposons que nous ayons les deux triplets explicites suivants dans un graphe de connaissance [5] :

```
Tabac-Du-Moulin rdf:type TobbacoStore
TobbacoStore rdfs:SubClassOf Store
```

Il est alors possible de déduire le triplet implicite `Tabac-Du-Moulin rdf:type Store` grâce à la règle de déduction RDFS qui indique chaque instance d'une classe est également une instance de ses superclasses.

Lorsque l'ensemble des triplets implicites a été déduit et intégré au graphe de connaissances, celui-ci est alors qualifié de *saturé*. La saturation d'un graphe RDF est unique (à l'exception du renommage des nœuds blancs) et ne contient plus de triplets implicites, car tous ont été explicités par le processus de saturation.

2.5 Interrogation des graphes de connaissances

Le langage standardisé par le W3C pour l'interrogation des graphes de connaissances définis en RDF est SPARQL. Dans la suite de ce mémoire, nous nous concentrerons sur un sous-ensemble de ce langage. Plus précisément, nous utiliserons les requêtes `SELECT` de SPARQL, qui sont de la forme `SELECT V WHERE P`, où `V` représente un ensemble de variables et `P` un *patron de graphe*. Nous nous intéresserons aux patrons de graphes constitués par la conjonction de *patrons de triplets*, ces derniers étant des triplets RDF dans lesquels une ou plusieurs variables, préfixées par `?`, sont introduites.

Exemple. La requête SPARQL suivante permet de retourner les ventes des clients de Bimedia qui fournissent du tabac.

```
SELECT ?store ?sale WHERE
    ?store rdf:type TobbacoStore .
    ?store tobaccoSale ?sale
```

Cette requête est composée d'une conjonction de deux patrons de triplets. Le premier permet d'identifier les clients de Bimedia vendant du tabac via la variable `?store`. Le second, utilisant également cette variable, permet de récupérer les ventes de ces mêmes clients.

L'exécution d'une requête SPARQL peut se faire sous différents modes de raisonnement (*entailment regimes*). Par défaut, aucun raisonnement n'est effectué (*simple entailment*), c'est-à-dire que seuls les triplets explicites sont utilisés pour répondre à la requête. Dans ce mode, il est donc nécessaire que le graphe de connaissances soit saturé pour obtenir l'ensemble des réponses. Il est également possible d'exécuter la requête en tenant compte non seulement des triplets explicites, mais aussi des triplets implicites pouvant être déduits à partir des règles de déduction d'un langage d'ontologie (par exemple, les modes de raisonnement RDFS ou OWL). Dans ce

5. Pour simplifier, nous utilisons des noms au lieu des URI

cas, le raisonnement s'effectue pendant l'exécution de la requête, ce qui entraîne nécessairement un temps d'exécution plus long. Cependant, la saturation d'un graphe n'est pas toujours la meilleure solution, car toute mise à jour du graphe nécessite de recalculer sa saturation.

Après avoir présenté les principales notions du web sémantique nécessaires à la compréhension de nos travaux, nous abordons dans la section suivante le cas d'étude réel que nous avons choisi d'examiner. Nous montrerons ensuite, à travers une étude de l'état de l'art, que peu de recherches ont proposé une mise en œuvre complète des technologies du web sémantique sur des cas réels, ni réalisé une évaluation globale des performances des différentes architectures possibles pour cette mise en œuvre.

3 Cas d'étude considéré et limite de l'état de l'art

Pour des raisons de confidentialité, l'étude présentée dans cette section est fictive, mais elle reflète la réalité. Nous considérons également un cas d'étude simple pour des raisons de concision.

3.1 Exemple de cas d'étude

Le cas d'étude considéré porte sur l'activité de régie publicitaire d'Orisha Retail Shops. Cette dernière se concentre principalement sur la diffusion de publicités sur le second écran des caisses Bimedia. L'analyse dans ce contexte vise typiquement à extraire une liste de points de vente en fonction de critères spécifiques. Cette approche est souvent utilisée pour cibler des points de vente spécifiques lors de campagnes publicitaires, plutôt que de diffuser la publicité sur l'ensemble du réseau.

Les critères couramment utilisés pour cette sélection incluent : le volume de ventes de produits dans des catégories spécifiques (telles que la *presse* ou la *confiserie*), la localisation géographique des points de vente (y compris des facteurs tels que la proximité de lieux d'intérêt), ainsi que des données démographiques (comme l'âge ou la distribution des revenus dans la zone). Des données additionnelles peuvent être collectées via des services tiers.

Supposons que le département publicitaire d'Orisha Retail Shops souhaite promouvoir une nouvelle édition d'un magazine dédié au rôle des femmes dans l'industrie. L'analyse viserait à cibler les commerces vendant des produits de presse, situés dans des zones où la proportion de femmes cadres est élevée. Actuellement, cette tâche nécessite de consolider des données provenant de multiples sources, notamment les ventes des commerces, les résultats de l'étiquetage automatique des produits via la solution ThesaurusBT (détaillée au Chapitre 4), ainsi que les informations démographiques des zones concernées. Un processus non seulement chronophage, mais également difficilement réutilisable pour de nouvelles études.

L'objectif est de rationaliser ce processus en intégrant toutes les données nécessaires au sein d'une couche sémantique, permettant aux utilisateurs finaux de modifier facilement les paramètres d'analyse en fonction de leurs besoins commerciaux à travers des requêtes sémantiques. La solution doit être conçue pour un usage industriel, capable de gérer de grands volumes de données, et inclure une interface intuitive qui assure l'autonomie des utilisateurs finaux sans nécessiter de compétences techniques.

En tenant compte de ces objectifs, nous présentons dans la section suivante les travaux existants sur le sujet, afin d'identifier des solutions potentielles et de mettre en évidence des lacunes à combler.

3.2 État de l'art sur la mise en œuvre des graphes de connaissance dans un contexte industriel

Les travaux portant sur la mise en œuvre des technologies du web sémantique dans un contexte industriel peuvent être classés en deux catégories : ceux qui démontrent la pertinence de ces technologies dans l'industrie, et ceux qui fournissent un retour d'expérience sur leur usage et leur performance. Nous décrivons ces deux catégories dans les sections suivantes.

La majorité des articles recensés dans la littérature se classe dans la première catégorie, qui examine pourquoi les graphes de connaissances sont bénéfiques dans un contexte industriel. Par exemple, Li et al. [Li, 2021] ont mené une enquête sur 119 articles, résumant les efforts techniques et pratiques liés à l'exploitation des graphes de connaissances en milieu industriel. De même, Yahya et al. [Yahya, 2021], ainsi que Grangel-González [Grangel-González, 2019], ont démontré le potentiel de ces technologies pour la gestion de l'information en maintenance, l'optimisation des ressources, et la production dans des environnements industriels, tout en abordant les défis d'interopérabilité liés aux différentes représentations d'entités et normes.

Par ailleurs, Zou [Zou, 2020b] ansi que Abu-Salih [Abu-Salih, 2021] ont examiné les applications des graphes de connaissances dans divers domaines, discutant de leur évolution, de leur impact académique et industriel, ainsi que de leur contribution à l'amélioration de l'intelligence des machines. Des travaux similaires ont été réalisés sur les Graphes de Connaissances Virtuels (GCV), où les données restent dans leur format et solution de stockage originaux [Chaves-Fraga, 2022 ; Xiao, 2022 ; Mendes de Farias, 2023 ; Vogt, 2023].

Cependant, aucun des articles identifiés dans cette catégorie n'évalue explicitement les architectures possibles pour l'intégration d'une couche de web sémantique, et aucun n'effectue de comparaisons quantitatives des performances des outils ou des architectures techniques. Par conséquent, les articles de cette première catégorie fournissent rarement des perspectives pratiques ou des retours d'expérience sur l'intégration des couches de web sémantique et des graphes de connaissances dans un contexte industriel.

La seconde catégorie de travaux, qui fournit des retours d'expérience sur l'utilisation des technologies sémantiques dans un contexte industriel, est plus proche de nos objectifs. Cependant, bien que des ontologies spécifiques aient été développées pour l'industrie, peu de travaux ont détaillé et évalué leur mise en œuvre. Nous avons identifié deux contributions principales l'ayant fait [Fishkin, 2017 ; Hubauer, 2018].

Hubauer et al. [Hubauer, 2018] présentent la mise en œuvre d'un graphe de connaissances dans le contexte de l'entreprise Siemens. Cependant, cette contribution ne mentionne que les principales étapes de la démarche suivie, sans détailler la méthode et les outils utilisés à chaque étape. De plus, aucune évaluation de performance n'est proposée. Une autre contribution notable est celle de Fishkin [Fishkin, 2017], qui traite également de la mise en œuvre de graphes de connaissances dans l'industrie. Contrairement aux travaux de Hubauer, Fishkin explicite les

outils employés pour mettre en place un graphe de connaissances et décrit les étapes impliquées dans cette mise en œuvre. Toutefois, ces travaux ne proposent ni comparaison entre différentes architectures, en particulier lorsqu'il s'agit d'intégrer des données ouvertes, ni évaluation de leur performance.

Les contributions que nous présentons dans les deux sections suivantes s'inscrivent dans la catégorie des travaux visant à détailler la mise en œuvre des technologies du web sémantique dans un contexte industriel. Notre objectif est de fournir une méthodologie complète, décrivant les étapes de la mise en œuvre de ces technologies ainsi que les outils que nous jugeons les plus adaptés. Étant donné les multiples architectures possibles, nous proposons également une évaluation expérimentale pour démontrer la pertinence de chacune. Bien que ces contributions soient spécifiquement liées à notre domaine d'étude, le secteur de la vente au détail, nous pensons qu'elles peuvent également être pertinentes pour d'autres secteurs.

4 Proposition d'une méthodologie de mise en œuvre des technologies du web sémantique

Les sections suivantes présentent la méthodologie que nous proposons pour l'implantation d'une couche sémantique dans un contexte industriel, tel que celui d'Orisha Retail Shops. Cette méthodologie couvre l'ensemble du processus, depuis la phase initiale de conception jusqu'à la mise en production, en passant par toutes les étapes intermédiaires de réflexion et de développement. Chaque phase est illustrée par des exemples concrets pour illustrer notre méthodologie. Nous abordons également les difficultés rencontrées tout au long du projet, fournissant ainsi des retours d'expérience sur la mise en œuvre des technologies du web sémantique dans un contexte industriel.

4.1 Conception des ontologies

Les ontologies jouent un rôle crucial dans la définition de concepts partagés et de vocabulaires pour intégrer des sources de données diverses. Bien que de nombreuses ontologies soient disponibles sur le Web, elles ne couvrent souvent pas entièrement les besoins spécifiques de certains domaines. Pour Orisha Retail Shops, les concepts relatifs aux données internes sont spécifiques au secteur de la vente au détail et, à notre connaissance, ne sont pas couverts par les ontologies existantes. En revanche, le vocabulaire pour les données démographiques est déjà bien défini dans des ontologies publiques telles que celles de l'INSEE [6]. Il est donc nécessaire pour l'entreprise de développer une nouvelle ontologie centrée sur ses concepts propres et de l'associer à des ontologies ouvertes couvrant des domaines plus larges. Cette stratégie est souvent recommandée pour les cas d'utilisation industriels, qui nécessitent la manipulation de concepts spécifiques et propriétaires [Hubauer, 2018].

6. https://github.com/InseeFr/Ontologies

4.1.1 Choix d'une méthode et d'outils pour la conception d'ontologie

Deux grandes approches pour la conception d'ontologies existent : l'approche automatique, qui vise à extraire les concepts et relations à partir de sources de données existantes [Weikum, 2021], et l'approche manuelle, qui repose sur l'expertise de spécialistes du domaine [Alfaifi, 2022]. Il est également possible de combiner ces deux méthodes. L'approche automatique a été utilisée pour la conception de grandes ontologies telles que *YAGO* [Hoffart, 2013] et *DBpedia* [Lehmann, 2015], à partir de données du Web, notamment issues de Wikipedia. Dans ce contexte, où l'ontologie est très large et couvre de nombreux domaines, une approche manuelle est impraticable.

Dans notre cas, les sources de données sont internes et leur modélisation rend difficile l'extraction automatique des concepts, car ceux-ci sont dispersés sur plusieurs tables et parfois sur différents systèmes de gestion de données. De plus, l'ontologie visée est de taille modeste, puisqu'elle est centrée sur un domaine spécifique et doit rester simple pour être utilisable par différents collaborateurs au sein de l'entreprise. Nous avons donc opté pour une méthode de conception manuelle.

Concernant l'outil utilisé pour la conception de l'ontologie, il existe deux principales approches. D'un côté, les outils graphiques, le plus connu étant Protégé, permettent de construire des ontologies de manière visuelle. De l'autre, des outils textuels sous forme de bibliothèques permettent de définir une ontologie via des scripts, en utilisant des formats comme RDF/XML et Turtle Triplets, ce qui facilite leur intégration dans divers contextes. Nous avons opté pour cette dernière solution, car elle correspond aux pratiques de l'entreprise en matière de conception de modèles de données.

Plus précisément, nous avons utilisé la bibliothèque Python Owlready2 [7], qui offre une manipulation aisée et flexible des structures ontologiques. Cette bibliothèque a été retenue pour sa renommée, sa documentation exhaustive, et sa capacité à générer des ontologies de manière algorithmique, une fonctionnalité précieuse pour l'extension automatique des ontologies. Bien que notre travail se soit concentré sur une approche manuelle, cette option pourrait être intéressante si l'ontologie devait être étendue au-delà d'Orisha Retail Shops, notamment à l'ensemble du groupe Orisha, auquel l'entreprise appartient.

4.1.2 L'ontologie Orisha Retail Shops interne

Atteindre un consensus sur la définition d'une ontologie était un aspect crucial à considérer, une consultation approfondie des experts métiers d'Orisha Retail Shops a été essentielle. Ensemble, nous avons identifié et défini les concepts et les relations à intégrer dans l'ontologie. Dans ce cadre, des classes clés ont été définies et sont présentées sur la figure 5.1.

Dans un premier temps, la définition des classes de l'ontologie d'Orisha Retail Shops s'est concentrée sur les données internes relatives aux commerces. Les classes principales incluent `Store` (commerce), `Activity` (activité commerciale), `Manager` (gérant(e)), `Place` (lieu au sens immobilier), `CashRegister` (caisse enregistreuse), `Module` (modules activés ou non de la solution Bimedia), `Product` (produit) et `Sale` (vente). Chaque nom de classe est écrit en PascalCase pour

7. https://owlready2.readthedocs.io/en/latest/

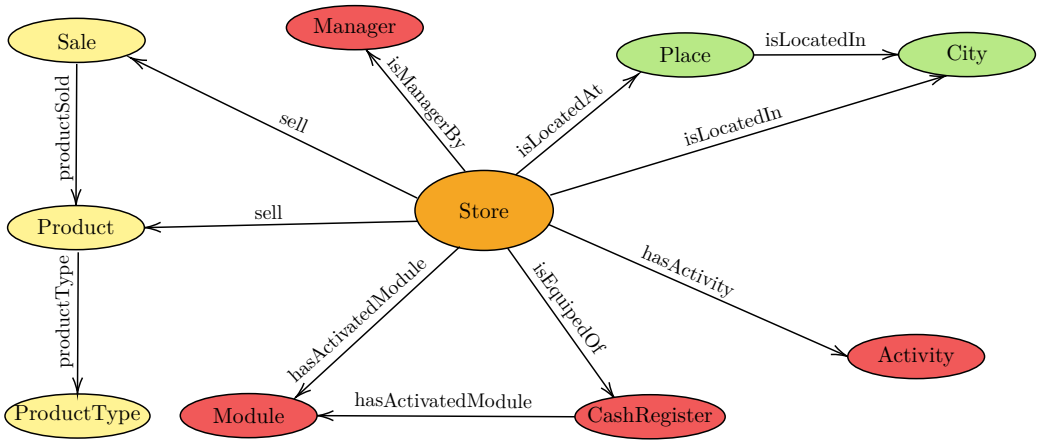


FIGURE 5.1 – Principaux éléments de l'ontologie interne Orisha Retail Shops

respecter les conventions de la nomenclature OWL.

La sélection des classes à inclure a été guidée par leur utilité pour des analyses actuelles et futures, tout en maintenant une cohérence logique dans la représentation des données. Par exemple, inclure la classe *Module* permet d'identifier facilement les commerces qui pourraient bénéficier de l'activation de modules logiciels spécifiques liés à des activités commerciales telles que le transfert d'argent, même si ces modules sont actuellement désactivés.

L'un des défis majeurs a été de déterminer la granularité appropriée des classes pour qu'elles soient suffisamment génériques pour permettre une intégration avec d'autres ontologies tout en restant spécifiques aux besoins de l'entreprise. Par exemple, la classe `Store` pourrait être liée par une relation d'équivalence à une classe similaire dans une autre ontologie qui référence les entreprises françaises. De même, la classe `Place` pourrait être associée à des zones géographiques définies dans une ontologie de données démographiques françaises. Si nous avions choisi pour une granularité trop fine, cela aurait également entraîné une prolifération des classes, risquant ainsi de diminuer l'utilisabilité de l'ontologie.

Pour répondre à ces besoins, toutes nos classes dérivent de `owl:Thing`, la classe racine dans OWL, permettant un niveau d'expressivité supérieur à RDFS. Cela facilite notamment la création d'axiomes complexes, comme le fait que les classes représentant les produits (`Product`) de type (`ProductType`) tabac et presse sont disjointes. Cela garantit que l'ontologie est à la fois précise sur la sémantique des données et adaptée aux spécificités du secteur du commerce de détail.

Après avoir établi les classes principales de notre ontologie, nous nous sommes concentrés sur les relations, appelées `ObjectProperties`. Avec le groupe d'experts métier, nous avons examiné toutes les relations sémantiques possibles entre les différents concepts. Les termes utilisés pour nommer ces relations suivent une convention de nommage précise, généralement en camelCase. Nous avons également pris soin de définir, lorsque possible, les relations inverses (`owl:InverseOf`). Par exemple, la relation `isLocationOf`, qui relie la classe `Place` à la classe `Store`, est l'inverse de `isLocatedAt`. Nous avons appliqué cette approche également pour les relations d'équivalence (`owl:equivalentProperty`), de transitivité (`owl:TransitiveProperty`),

ainsi que pour diverses contraintes comme celles de cardinalité (`owl:FunctionalProperty`).

Lors de l'élaboration de ces propriétés, nous avons choisi de saturer certains faits de l'ontologie, afin de ne pas nécessiter un moteur d'inférence pour cette tâche. Ceci est particulièrement vrai pour les classes liées aux localisations. Nous avons par exemple créé des relations de base entre `Store` et `City`, en dépit de la classe intermédiaire `Place`, comme illustré dans la figure 5.1. Cette approche vise à renforcer la cohérence et à simplifier l'utilisation de l'ontologie, en minimisant la complexité des requêtes nécessaires pour exploiter le graphe de connaissances.

Nous avons intentionnellement conçu notre ontologie pour s'adapter et évoluer en réponse aux exigences changeantes, en intégrant de nouveaux concepts, relations, et autres éléments pertinents. Par exemple, la définition des concepts tels que `Bureau de tabac`, `Boulangerie`, ou `Restaurant` peut évoluer au fil du temps. Un `Bureau de tabac` est généralement défini comme un commerce vendant principalement des produits liés au tabac. Cependant, un établissement comme une boîte de nuit qui vend occasionnellement ces produits ne serait pas classifié comme tel par les experts du domaine.

Des règles spécifiques pourraient définir un `Bureau de tabac` comme un commerce réalisant la vente de plus de 1000 articles liés au tabac par mois, ou un certain pourcentage des ventes totales du commerce, bien que ces seuils puissent varier. Ces critères, susceptibles de changer en fonction des fluctuations du marché, doivent être clairement établis par les experts du domaine. De telles règles commerciales peuvent être implémentées dans l'ontologie en utilisant un langage de règles, le plus connu étant SWRL (Semantic Web Rule Language). La flexibilité dans la modification des paramètres de ces règles permet de s'adapter aux besoins évolutifs. Nous avons également établi d'autres règles, comme des contraintes d'unicité pour les villes ou des règles sur le statut actif d'un commerce, afin d'empêcher qu'un commerce soit incorrectement marqué comme inactif alors qu'il réalise des ventes durant le mois en cours.

Notre cas d'étude requiert également l'utilisation de données externes. Ces dernières évoluant indépendamment du système d'information d'Orisha Retail Shops, nous avons décidé de créer une ontologie spécifique pour ces données. Celle-ci est présentée dans la section suivante.

4.1.3 L'ontologie Orisha Retail Shops externe

Les sources de données externes considérées couvrent des thèmes variés et évoluent indépendamment les unes des autres. Nous aurions donc pu définir plusieurs ontologies spécialisées pour chacune d'entre elles. Cependant, en raison des limitations techniques des outils utilisés, qui ne supportent pas l'emploi simultané de multiples ontologies dans un même graphe de connaissances, et pour éviter la multiplication des graphes hébergés, nous avons opté pour l'intégration de tous les concepts dans une ontologie unique. Les principaux éléments de cette ontologie sont présentés dans la figure 5.2.

Nous avons d'abord élaboré des concepts relatifs aux données juridiques des commerces, extraites de sources gouvernementales ouvertes telles que l'INSEE, data.gouv [8], et datainfogreffe [9]. Cela inclut des classes telles que `LegalUnit` (unité légale enregistrée dans le SIREN),

8. https://www.data.gouv.fr/fr/
9. https://datainfogreffe.fr/

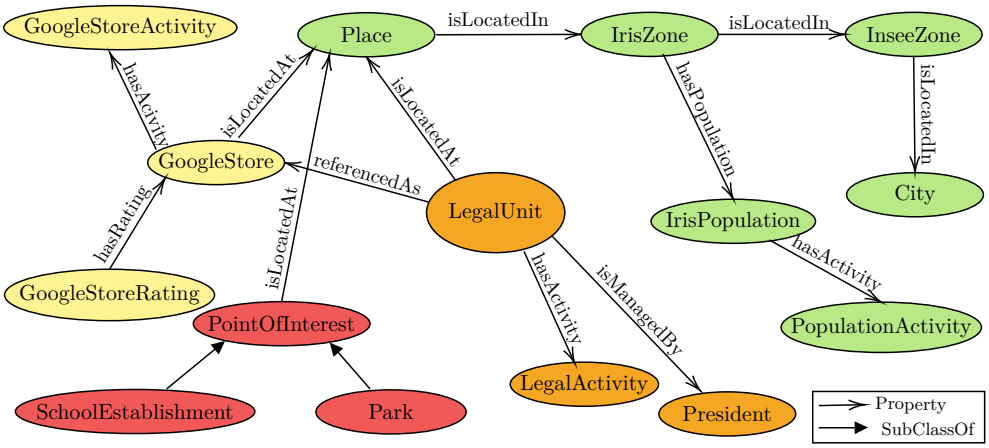


FIGURE 5.2 – Principales classes et relations de l'ontologie externe

`LegalActivity` (activité principale exercée déclarée à l'état), `Establishment` (établissement au sens légal identifié par un SIRET), `President` (président de l'établissement), etc. Ces données sont particulièrement utiles pour réaliser des analyses sur les gérants de multiples commerces ou sur les types d'activité principale exercée déclarés par les commerces du réseau d'Orisha Retail Shops.

Puis, nous avons travaillé sur les concepts relatifs aux données ouvertes concernant les populations. Nous avons intégré des classes telles que `IrisZone`, `InseeZone`, `IrisPopulation`, et `PopulationActivity`. Ces données permettent notamment de réaliser des analyses statistiques sur les corrélations entre les ventes de certains produits et la densité de population ayant certains types d'activités socio-professionnelles, par exemple.

Ensuite, nous avons intégré les concepts des données issues de Google API, en développant des classes comme `GoogleStore` (commerce référencé dans Google), `GoogleStoreActivity` (activité étiquetée par Google), `GoogleStoreRating` (avis laissé sur Google), `GoogleStoreInfo` (informations sur les horaires d'ouverture, descriptions, etc.). Les données obtenues via cette source sont essentielles, notamment pour les analyses liées aux types d'activités des commerces, comme les restaurants, qui sont souvent bien référencés dans Google.

Nous avons ensuite développé des classes et relations pour prendre en charge ce que nous appelons les `Points d'Intérêt`, sous la classe `PointOfInterest`. Ces points d'intérêt représentent des lieux pertinents pour certaines analyses. Par exemple, les collèges et lycées sont pris en compte lorsqu'il s'agit de cibler ou d'exclure des points de vente à proximité de ces établissements pour des campagnes publicitaires concernant des boissons alcoolisées, ou au contraire, pour cibler un public mineur.

A noter que pour chaque instance référençant un établissement ou un lieu, une localisation sera spécifiée à l'aide de la classe `Place`, qui sera déclarée équivalente (`owl:sameAs`) à une instance de la classe `Place` du graphe de connaissances sur les données internes, si une telle localisation existe déjà dans le graphe. Ce type de lien est établi lors du peuplement des données, grâce à un calcul d'alignement d'adresse effectué par un script Python de peuplement. Une relation d'équivalence similaire est également mise en place entre les instances de la classe `Store`

du graphe de connaissances interne et les instances des classes `GoogleStore` ou `LegalUnit`.

Grâce aux relations établies entre les deux graphes de connaissances, il est alors possible d'interroger simultanément les données internes et externes via des requêtes SPARQL, sans contraintes particulières dans la formulation de ces requêtes. Cette intégration assure une fluidité et une cohérence dans l'accès aux informations, facilitant ainsi les analyses complexes et multidimensionnelles. Un exemple de requête est illustré en figure Figure 5.3. Dans cette requête, nous pouvons récupérons pour chaque commerce `Store`, sa zone iris associée ainsi que sa population issue des données ouvertes de l'INSEE (sous la forme d'URIs).

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX bimedia: <http://test.org/products_and_sales_ontology/>
PREFIX open: <http://test.org/open_data_ontology/>

SELECT ?store ?iris ?pop WHERE {
    ?store rdf:type bimedia:Store.
    ?store bimedia:isInIrisZone ?iris.
    ?open_data_iris owl:sameAs ?iris.
    ?open_data_iris open:hasPopulation ?pop
}
```

FIGURE 5.3 – Exemple de requête SPARQL utilisant l'opérateur owl:SameAs

Maintenant que les ontologies internes et externes ont été définies, nous pouvons passer à la seconde étape de notre méthodologie, qui consiste en l'intégration des données provenant des différentes sources sous forme de graphes de connaissances

4.2 Intégration des sources de données internes

L'intégration des données sémantiques implique la création d'un graphe de connaissances, qui englobe l'ontologie définie dans la section 4.1.2, ainsi que ses instances dérivées des sources de données intégrées. Ce graphe de connaissances peut être virtuel, appelé *graphe de connaissances virtuel* (GCV), lorsque les instances demeurent dans leur solution de stockage d'origine. Cette approche permet d'éviter la duplication des données, ce qui est essentiel pour l'entreprise, notamment en raison du volume important des données de ventes, qui font partie d'un processus de gestion soumis à diverses réglementations. Nous démontrons comment ce concept peut être mis en œuvre dans un scénario réel en utilisant l'une des sources de données internes : la base de données relationnelle de ventes PostgreSQL.

Différents outils ont été proposés pour construire un GCV au-dessus de diverses sources de données. Parmi ceux que nous avons identifiés comme les plus matures pour un usage industriel figurent Ontop-VKG [Ontop, 2017] et Virtuoso [OpenLink, 2013]. Nous avons choisi d'utiliser Ontop-VKG en raison de sa nature open-source, ce qui renforce la reproductibilité de notre approche, contrairement à Virtuoso.

La construction d'un GCV avec Ontop-VKG nécessite des règles de correspondance (*mappings*) pour établir des liens entre les concepts de l'ontologie et les données de la base de données.

Cela s'effectue via des règles exprimées dans deux langages distincts : OBDA (Ontology-based Data Access) ou R2RML (RDB to RDF Mapping Language) [W3C, 2012]. Pour ce projet, nous avons choisi le langage R2RML, recommandé par le W3C. Plus précisément, nous utilisons des requêtes SQL pour (1) définir comment récupérer les instances des classes de notre ontologie et (2) créer des relations entre ces instances en spécifiant quelles colonnes utiliser pour les jointures.

Nous illustrons notre approche en intégrant une partie des données commerciales dans notre couche sémantique, en nous concentrant spécifiquement sur les ventes, les produits, les données juridiques et les adresses. Ces données sont distribuées sur cinq tables au sein de notre base de données relationnelle :

— la première est composée des ventes des différents commerces, agrégées par produit et par jour ;

— la seconde contient les données sur les produits (telles que les noms, les prix, les familles de produit) ;

— la troisième contient des informations légales concernant les commerces (telles que le nom commercial et le numéro SIRET) ;

— la quatrième stocke les adresses des commerces ;

— la cinquième contient des informations relatives aux villes.

Au sein de notre ontologie, dont un extrait est présenté sur la figure 5.11, les informations concernant ces données sont représentées sous six classes : `Store`, `Place`, `City`, `ZipZone`, `IrisZone`, et `InseeZone`. Pour peupler notre GCV des commerces virtuellement, nous devons exprimer un mapping entre l'ontologie et la table relationnelle. Ce mapping est basé sur la requête SQL présentée dans la Figure 5.4. Cette requête retourne quatre colonnes. Les figures 5.5 et 5.8 présentent deux exemples de code R2RML, illustrant le processus de mapping de ces données sur notre ontologie pour la construction du GCV.

```
1 SELECT DISTINCT si.store_id, si.business_name, si.siret, addr.iris
2 FROM summary.store_ids si
3 LEFT JOIN summary.address addr ON addr.store_id = si.store_id
```

FIGURE 5.4 – Example of SQL query used in a mapping

Dans l'extrait du script de mapping représenté dans la figure 5.5, nous lions les résultats de cette requête aux concepts définis dans notre ontologie. Dans la section de code commençant par `rr:subjectMap` (lignes 5-7), nous exprimons la clé primaire des instances à peupler et la classe de notre ontologie à laquelle ces instances sont attachées. Définir la clé primaire dans le mapping permet d'éviter des auto-jointures, un phénomène qui se produit lors de la réécriture par Ontop-VKG des requêtes SPARQL en requêtes SQL. Les tables sont alors jointes sur elles-mêmes, allongeant considérablement le temps de réponse. Dans cet exemple, la classe exprimée dans notre ontologie pour les instances à peupler est `Store` et la clé primaire est `store_id`. Ensuite, le `rr:predicateObjectMap` (lignes 8-16) spécifie les valeurs des propriétés des instances en extrayant les données des colonnes du résultat de la requête.

```
@prefix sto: <http://ontologies.orisha.com/products_and_sales_ontology/> .
<#Store>
  a rr:TriplesMap ;
  rr:logicalTable <#StoreTableView>;
  rr:subjectMap [
    rr:template "http://ontologies.orisha.com/products_and_sales_ontology/store/{
    ↪ store_id}" ;
    rr:class sto:Store; ];
  rr:predicateObjectMap [
    rr:predicate rdfs:label;
    rr:objectMap [ rr:column "business_name" ]; ];
  rr:predicateObjectMap [
    rr:predicate sto:business_name;
    rr:objectMap [ rr:column "business_name" ]; ];
  rr:predicateObjectMap [
    rr:predicate sto:siret;
    rr:objectMap [ rr:column "siret" ]; ];
```

FIGURE 5.5 – Example of R2RML Code

Les règles commerciales exprimées en SWRL au sein de l'ontologie (voir la section 4.1.2) ne peuvent pas être interprétées par l'outil Ontop-VKG, car il ne prend pas en charge ce type d'inférence. Par conséquent, ces règles doivent être intégrées directement dans les requêtes SQL au sein du fichier de mapping R2RML. Par exemple, pour définir que les commerces appartenant à la classe des bureaux de tabac (classe `TobbacoStore` dans notre ontologie, une sous-classe de `Store`), sont ceux qui ont vendu pour plus de 1000€ de produits étiquetés « Tabac » le mois précédent, la requête SQL appropriée est présentée dans la Figure 5.6. Par ailleurs, l'identification des produits de type TABAC est basé sur le contenu de la table `summary.product_ontology_mapping`, donc, sur le résultat de l'étiquetage des produits présenté dans le chapitre 4. En associant cette règle de maping à la nouvelle classe `TobbacoStore` dans notre script R2RML, nous pouvons extraire tous les commerces considérés comme bureaux de tabac à un instant donné à travers une requête SPARQL illustrée dans la figure 5.7.

```
SELECT ps.store_id, sum(ps.sum_quantity_product) as monthly_sum_quantity_product
FROM products_summaries ps
LEFT JOIN summary.product_ontology_mapping pom on ps.store_id=pom.store_id and ps.
    ↪ barcode=pom.barcode
WHERE pom.type_product_predicted_1 = 'Fumeur'
AND ps.date >= date_trunc('month', current_date - INTERVAL '1' MONTH)
AND ps.date < date_trunc('month', current_date)
group by ps.store_id
having monthly_sum_quantity_product > 1000
```

FIGURE 5.6 – Exemple de requête SQL de mappage permettant le remplacement d'une règle SWRL

Dans le script de maping entre la base de données relationnelle et notre graphe de connaissance, nous devons également définir des relations entre les classes. Cela peut être réalisé avec

```
REFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX bimedia: <http://test.org/products_and_sales_ontology/>

SELECT ?store WHERE {
    ?store rdf:type bimedia:TobbaccoStore.
 }
```

FIGURE 5.7 – Exemple de requête SPARQL pour la récupération des bureaux de tabac

une certaine forme de `rr:predicateObjectMap`. L'exemple représenté dans la Figure 5.8 correspond à la suite de l'exemple précédent. Dans ce code de mapping, nous exprimons la relation `isInIrisZone`, établissant le lien entre les instances de la classe `Store` et les instances de la classe `IrisZone` (une zone IRIS désigne une petite unité territoriale en France, utilisée dans les statistiques de population). Dans ce mapping, nous lions les instances de ces deux classes par la colonne IRIS, expliquant comment construire l'IRI de l'instance `IrisZone` correspondante.

```
rr:predicateObjectMap [
   rr:predicate sto:isInIrisZone;
   rr:objectMap [ rr:template "http://ontologies.orisha.com/
   ↪ products_and_sales_ontology/IrisZone/{iris}" ]; ];
```

FIGURE 5.8 – Exemple de correspondance exprimée avec R2RML

À ce stade, l'outil Ontop-VKG nous permet de configurer un *SPARQL endpoint*, qui permet de requêter les données à partir de l'ontologie, tout en maintenant les données dans leurs tables relationnelles d'origine

4.3 Intégration des sources de données externes

En plus des données de vente, le cas d'utilisation décrit dans la section 3.1 nécessite également des données externes. Lors de l'intégration de ces données externes dans la couche sémantique d'Orisha Retail Shops, différentes options sont disponibles : stocker les données dans un graphe de connaissance natif (géré par un triplestore) ou les stocker dans une base de données relationnelle, suivie de la virtualisation d'un graphe de connaissance au-dessus de ces données relationnelles.

Lors de l'utilisation d'un GCV, les données externes et internes peuvent être conservées séparément ou stockées par un seul système de gestion de base de données. Ces deux options seront testées dans nos expériences présentées dans la section 5.1. Puisque le processus de mise en œuvre du chargement de différentes sources de données externes dans une base de données relationnelle et de la virtualisation d'un GC au-dessus de cette base de données relationnelle a déjà été abordé, nous allons développer l'autre solution : l'utilisation d'un GC natif.

La première étape pour intégrer les données externes dans un GC à partir de fichiers implique de transformer les données tabulaires en triplets RDF. Ces triplets doivent respecter l'ontologie définie précédemment. Le résultat de cette étape est un nouveau fichier au format Turtle (Terse

RDF Triple Language). Pour faciliter cette transformation, nous avons utilisé les bibliothèques Python Pandas et Owlready2, qui nous permettent respectivement de charger les données brutes et d'exprimer les concepts définis dans notre ontologie (ainsi que d'exporter le résultat sous la forme de triplets RDF). Cela simplifie ainsi le peuplement du graphe de connaissances et son exportation dans un fichier au format Turtle.

Le processus de chargement et de transformation des données est illustré de manière simplifiée dans l'extrait de code Python dans la figure 5.9. Dans cet exemple, nous commençons par déclarer l'IRI de base de notre graphe de connaissances (utilisé pour construire les IRI de nos instances). Ensuite, nous chargeons les données à partir du fichier CSV contenant nos informations sur la population. Puis, en respectant la nomenclature utilisée dans notre ontologie, nous déclarons les classes et les relations que nous souhaitons utiliser dans ce graphe de connaissances, ici `Population`, `IrisZone`, et la relation entre ces deux éléments, nommée `liveIn`. Ensuite, nous itérons à travers notre dataframe pandas contenant les données de population et créons les instances d'intérêt. Finalement, nous exportons ce graphe de connaissances sous forme de triplets RDF dans un fichier avec l'extension ttl (Turtle).

```
from owlready2 import *
import pandas as pd

# Definir l IRI de base pour le graphe de connaissances
population_kg.set_base_iri("<http://test.org/open_data_ontology/>", rename_entities
    ↪ =True)

# Charger les donnees depuis un fichier CSV
df_population = pd.read_csv('PATH_TO_POPULATION_CSV_FILE')

# Definir les classes dans l'ontologie
class Population(Thing):
    pass

class IrisZone(Thing):
    pass

class liveIn(ObjectProperty, FunctionalProperty):
    domain = [Population]
    range = [IrisZone]

# Peupler le graphe de connaissances avec les donnees
for index, pop in df_population.iterrows():
    with population_kg:
        iris = IrisZone(pop['IRIS'])
        population = Population('POP' + pop['IRIS'])
        population.liveIn = iris

# Sauvegarder le graphe de connaissances en format Turtle
population_kg.save(file="augmented_store_data_kg.ttl", format="ntriples")
```

FIGURE 5.9 – Exemple de chargement et transformation de données tabulaires et triplets RDF

La deuxième étape consiste à rendre ces données accessibles via un triplestore, à l'aide d'un

SPARQL endpoint. Plusieurs solutions techniques existent pour héberger les données. L'entreprise Orisha Retail Shops exige un triplestore gratuit, régulièrement mis à jour, efficace pour notre faible volume de données externes, et compatible avec la propriété *owl:sameAs*. Nos tests ont révélé que trois triplestores répondent à ces exigences : Apache Jena, Eclipse RDF4J et GraphDB d'Ontotext. Parmi ces trois options, nous avons choisi GraphDB dans sa version gratuite car il s'aligne mieux avec l'architecture informatique d'Orisha Retail Shops grâce aux connecteurs qu'il fournit.

4.4 Fédération de graphes de connaissances pour l'analyse multi-sources

Pour permettre l'exécution de requêtes à travers les différentes sources de données précédemment intégrées dans différents graphes de connaissances, nous devons mettre en place un point d'accès SPARQL capable d'exécuter des requêtes fédérées et de gérer les relations entre les différents GC.

Les points d'accès SPARQL capables d'exécuter des requêtes fédérées peuvent être facilement identifiés par leur compatibilité avec l'opérateur SERVICE et, plus généralement, leur conformité avec le moteur SPARQL 1.1. Il existe également des outils de fédération transparents tels que FedX, disponible dans GraphDB ou RDF4J. Étant donné que nous avons précédemment choisi le SGBD GraphDB pour héberger les données sur le graphe de connaissances des sources externes, compatible avec l'opérateur SERVICE, nous utilisons ce SPARQL endpoint pour lancer nos requêtes fédérées, évitant ainsi la complexité de l'infrastructure des systèmes.

Comme exprimé dans la section précédente, nous avons utilisé la propriété `owl:sameAs` au moment de l'intégration des sources de données externes pour faire correspondre les instances de notre graphe de connaissances sur les données interne avec celles du graphe de connaissance sur les données externes. Cette étape est réalisée lors du peuplement du graphe de connaissances consacré aux données externes. Par exemple, pour chaque instance de `IrisZone` des données externes, nous établissons une relation d'équivalence avec l'instance correspondante dans le GCV des données internes. Les deux utilisent la même classe `IrisZone` de l'ontologie, et nous les lions en utilisant la propriété `owl:sameAs` comme abordé plus en détails dans les sections 4.1.3 et 4.3.

Un exemple de requête SPARQL fédérée est présenté dans la figure 5.10. Dans cet exemple, nous interrogeons le point d'accès SPARQL associé à notre graphe de connaissances sur les données externes. La requête effectuée sur ce dernier porte sur les zones IRIS et les populations associées à ces dernières. Nous combinons ces résultats avec ceux obtenus par une sous-requête formulée à l'intérieur des balises de l'opérateur SERVICE. Cette association est réalisée grâce aux instances de la classe `IrisZone` liées aux commerces (classe `Store`), permettant ainsi de fusionner les informations sur les commerces et celles sur les populations.

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX bimedia: <http://test.org/products_and_sales_ontology/>
PREFIX open: <http://test.org/open_data_ontology/>

SELECT ?store ?pop WHERE {
    { SERVICE <http://localhost:8080/sparql>
        {
           SELECT ?store ?iris WHERE {
                ?store rdf:type bimedia:Store.
                ?store bimedia:isInIrisZone ?iris.
            }
        }
    }
   ?open_data_iris  owl:sameAs ?iris.
   ?open_data_iris open:hasPopulation ?pop
}
```

FIGURE 5.10 – Exemple de requête SPARQL fédérée basée sur `owl:sameAs` pour la jointure

5 Évaluation expérimentale de différentes implémentations architecturales

Comme nous l'avons vu précédemment, la création de graphes de connaissances à partir de données internes et externes peut s'effectuer selon différentes architectures d'implémentation. Par exemple, il est possible de créer plusieurs graphes de connaissances et d'en virtualiser certains. À notre connaissance, aucune étude n'a proposé de comparaison systématique entre ces différentes architectures possibles. Les évaluations de performance se concentrent généralement sur un élément particulier de l'architecture, tel que le triplestore. Nous proposons donc dans les sections suivantes des expérimentations pour comparer des architectures possibles.

5.1 Protocole d'expérimentation

Les ontologies conçues pour Orisha Retail Shops étant relativement complexes, leur utilisation dans une expérimentation nécessitant un grand volume de données aurait été difficilement réalisable. Pour répondre à ces contraintes, nous avons élaboré une ontologie simplifiée, spécifiquement dédiée à l'évaluation expérimentale. La figure 5.11 présente les principales classes, propriétés et relations de cette ontologie. Elle contient huit classes principales : `Store`, `Sale`, `Product`, `Place`, `City`, `ZipZone`, `IrisZone` et `InseeZone`. Chaque classe possède une ou plusieurs propriétés. Ainsi, un commerce (`Store`) possède une enseigne et un numéro SIRET. Dans cette ontologie, les relations entre les classes sont également exprimées. Par exemple, un commerce (`Store`) a une adresse (`Place`), une adresse est liée à une ville (`City`), une ville est connectée à une zone insee (`InseeZone`), etc.

L'objectif de cette évaluation est de mesurer l'impact de 1) la virtualisation d'un graphe de connaissances par rapport à une solution native, et 2) la séparation des données internes

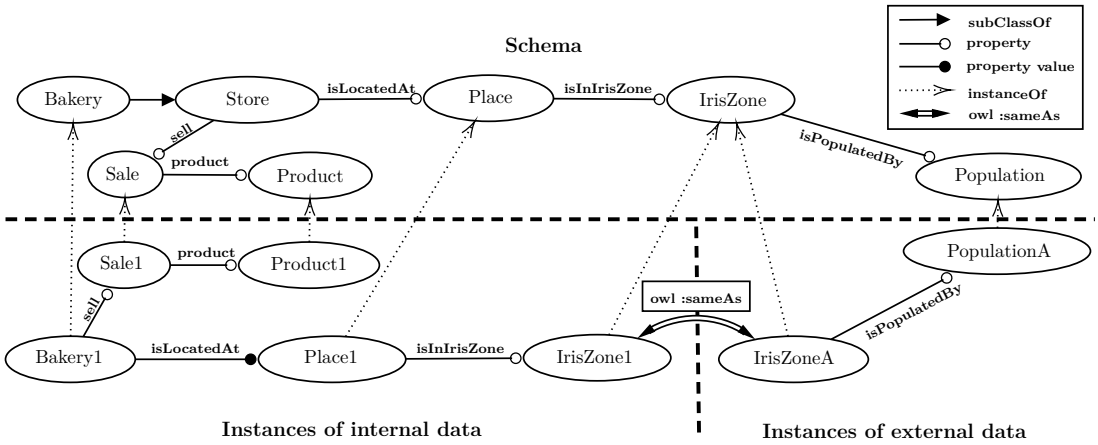


FIGURE 5.11 – Principaux éléments de l'ontologie Orisha Retail Shops dans le cas expérimental

et externes. En combinant ces deux critères, nous avons conçu six architectures, de A1 à A6, qui seront illustrées ultérieurement. Chaque architecture est testée à deux échelles : une avec une petite base de données de 30 millions de lignes (un mois de ventes), et une autre avec une base plus grande de 110 millions de lignes (six mois de ventes). Les données incluent les ventes agrégées par jour, commerce, et produit, ainsi que des informations sur les commerces et produits.

Pour chaque architecture et volume de données, nous exécutons trois requêtes. Bien que les requêtes soient conçues pour être aussi similaires que possible, elles sont adaptées à chaque architecture spécifique. Pour chaque configuration, nous analyserons le temps d'exécution, la complexité de la mise en place de l'architecture, ainsi que la complexité de la préparation des requêtes. La mise en œuvre complète de ces expériences est disponible dans un dépôt sur la forge du LIAS [10]

Les trois requêtes SPARQL considérées ont été conçues pour refléter des cas réels d'exploitation. Elles visent à récupérer les informations suivantes.

— La requête Q1 récupère la liste des numéros SIRET des commerces présents en Charente-Maritime (code de département 17). Pour cela, la requête SPARQL est une simple sélection avec un filtre sur les deux premiers chiffres du code postal des adresses des points de vente. Cette requête permet de mesurer le temps de réponse des différentes architectures sur des requêtes de base accédant à une faible quantité de données (métadonnées des commerces).

— La requête Q2 récupère la liste des numéros SIRET des commerces ayant vendu au moins un produit portant le nom de *coca* dans un mois donné. L'objectif de cette requête est de comparer les performances des architectures sur des volumes de données bien plus importants que le précédent, puisque cette fois, la requête concerne les données de ventes. Avoir une solution qui peut filtrer efficacement les données de vente en fonction des dates est important pour les applications qu'Orisha Retail Shops souhaite faire de ce graphe de connaissances. Des structures d'optimisation sont présentes sur la table des ventes dans les bases de données relationnelles sous-jacentes, nous attendons donc des performances

10. https://forge.lias-lab.fr/retailkgintegration

Graphes de connaissances	Volume
GC interne petit volume	233,178,003
GC interne gros volume	638,033,173
GC externe population	25,183,519
GCV interne petit volume	233,026,330
GCV interne gros volume	638,024,436
GCV externe population	25,167,394
GC interne petit volume + externe population	258,338,227
GC interne gros volume + externe population	663,193,397

TABLE 5.1 – Nombre de triplets des différents graphes de connaissances

adéquates sur les GCV, même avec de grands volumes.

— La requête Q3 récupère les commerces qui vendent plus de 100 articles catégorisés comme *presse* et qui sont situés dans une zone IRIS comptant plus de 300 cadres femmes, selon les données enregistrées par l'État français. Cette requête correspond à l'étude de cas présentée dans la section 3.1. Une seconde version de cette requête (Q3'), destinée uniquement à la première architecture, ne considère pas les données externes et consiste uniquement à récupérer les commerces qui vendent plus de 100 articles catégorisés comme *presse*.

Le tableau 5.1 fournit un résumé des tailles des différents graphes de connaissances considérés. Pour les GCVs, nous comptons tous les triplets du graphe virtuel. Nous pouvons observer de légères variations dans le nombre de triplets entre les graphes de connaissances physiques et leurs homologues virtuels, même lorsque les sources de données sont identiques. Cela peut s'expliquer par les transformations de données nécessaires lors de la construction des graphes de connaissances natifs, qui peuvent conduire à la consolidation de certaines instances dans le graphe, réduisant ainsi le nombre de triplets.

5.2 Résultats expérimentaux

La présentation des résultats se fait de manière systématique en examinant successivement chaque architecture. Pour chaque architecture, nous fournissons un schéma illustratif, accompagné d'un tableau récapitulatif des résultats obtenus.

L'architecture initiale (A1), illustrée dans la figure 5.12, comprend un GCV au-dessus de la base de données relationnelle des ventes et des commerces. Cette architecture sert de référence de base, facilitant l'évaluation des impacts résultant de l'intégration des sources de données externes dans les architectures ultérieures. Les temps de réponse avec cette architecture sont donnés dans le tableau 5.2. Nous observons une influence significative du volume de données de vente sur les temps d'exécution des requêtes liées (Q2 et Q3).

La deuxième architecture (A2), illustrée dans la figure 5.13, consiste en un GCV au-dessus de la base de données relationnelle des ventes et des commerces, avec l'ajout de données externes sur la population. Cette architecture sert de référence pour mesurer l'impact de la séparation des sources de données externes et internes dans les architectures subséquentes.

FIGURE 5.12 – Schéma d'architecture A1

Requête	Petit Volume	Gros Volume
Q1	0.5s	0.5s
Q2	35.9s	2m 21s
Q3' (sans externes)	52.3s	4m 31.2s

TABLE 5.2 – Temps d'exécution des requêtes pour A1

Cette architecture n'est pas pertinente dans le cas d'Orisha Retail Shops, car il est inhabituel d'inclure des sources de données externes dans une base de données construite en agrégeant des données internes issues d'un lac de données, comme c'est le cas avec la base de données relationnelle des ventes et des commerces. De plus, cette base de données est utilisée pour construire des tableaux de bord pour les clients, il est donc interdit de l'utiliser pour stocker des données qui ne sont pas liées à ces fins.

Les résultats obtenus avec cette architecture sont présentés dans le tableau 5.3. Le temps de réponse pour la requête Q3 dans cette architecture est comparable à celui observé dans l'architecture A1 lorsqu'elle est exécutée sur un grand volume de données. Cela s'explique par l'ajout d'un second filtre sur la zone IRIS (données de population), ce qui réduit le nombre de lignes retournées. Les transformations appliquées aux tables de données lors de l'exécution de la requête SQL (ici sur une base de données PostgreSQL locale) pour la virtualisation du graphe sont coûteuses en calcul, ce qui explique l'augmentation du temps d'exécution.

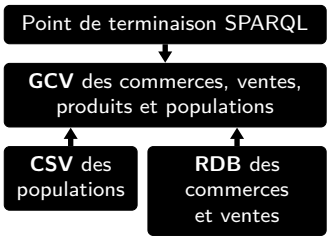


FIGURE 5.13 – Schéma d'architecture A2

Requête	Petit Volume	Gros Volume
Q1	0.5s	0.5s
Q2	35.8s	2m 20s
Q3	1m 0.5s	3m 16.9s

TABLE 5.3 – Temps d'exécution des requêtes pour A2

La troisième architecture (A3), illustrée dans la figure 5.14, se compose de deux GCV : l'un se superpose à la base de données relationnelle des ventes et des commerces, et l'autre à une base de données relationnelle distincte contenant des données démographiques. Pour cette architecture, la requête Q3 a été exécutée à partir d'un autre endpoint SPARQL prenant en charge l'opérateur SERVICE, contrairement au SPARQL endpoint fourni par Ontop-VKG.

La requête utilise l'opérateur SERVICE à deux reprises pour récupérer des données des deux GCV. La première utilisation de SERVICE permet de récupérer les commerces ayant vendu plus de 100 articles catégorisés comme étant de type «`presse`» en un mois, ainsi que l'IRI de la `IrisZone` du commerce et l'IRI correspondant de cette `IrisZone`. La seconde utilisation de SERVICE vise à récupérer les IRI des `IrisZones` comptant plus de 300 cadres féminins dans leur population, ainsi que les IRI correspondants de cette `IrisZone` dans le graphe des ventes et des commerces (grâce à la propriété `owl:sameAs`).

Le moteur SPARQL calcule ensuite l'intersection de ces deux sous-requêtes pour produire les résultats attendus. L'exécution de cette requête nous permet d'évaluer l'impact de l'intégration de plusieurs GCV dans l'architecture technique et de vérifier le bon fonctionnement des jointures dans le cas des GCV. Les résultats obtenus pour cette architecture, présentés dans le tableau 5.4, montrent une diminution significative des performances pour Q3, probablement en raison de l'utilisation de plusieurs opérateurs SERVICE, en plus de la jointure requise par le moteur SPARQL pour retourner les données attendues. Nous observons également une augmentation réelle de la complexité de la requête. En revanche, pour les requêtes ne nécessitant pas de jointure, aucune différence significative par rapport à l'architecture A2 n'est observée, ce qui était attendu.

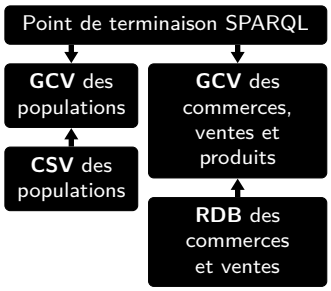


FIGURE 5.14 – Schéma d'architecture A3

Requête	Petit Volume	Gros Volume
Q1	0.5s	0.5s
Q2	38.0s	2m 21s
Q3	76m 0.5s	N/C

TABLE 5.4 – Temps d'exécution des requêtes pour A3

La quatrième architecture (A4), illustrée dans la figure 5.15, se compose d'un GCV au-dessus de la base de données relationnelle des commerces, ainsi que d'un graphe de connaissances natif contenant des données démographiques. Ces dernières ont été préalablement transformées en triplets et importées dans un graphe de connaissances GraphDB. Dans cette configuration, la requête Q3 est exécutée à partir du SPARQL endpoint sur le graphe de population, tirant parti de son support pour l'opérateur SERVICE.

La requête commence par récupérer les `IrisZones` à partir du graphe de population, en filtrant celles qui comptent plus de 300 cadres féminins enregistrés. Ensuite, la requête se poursuit sur le GCV pour récupérer les commerces ayant vendu plus de 100 articles de type "presse" en un mois, avec un filtre supplémentaire sur l'IRI de la `IrisZone` du commerce, correspondant à l'IRI obtenu lors de la première étape.

Le moteur SPARQL évite le calcul d'une intersection entre les sous-requêtes, car il n'y a qu'une seule sous-requête à exécuter, ce qui entraîne une réduction significative du temps d'exécution, même par rapport à l'architecture A1 avec un grand volume de données. Les résultats obtenus pour cette architecture, présentés dans le tableau 5.5, montrent une performance améliorée, notamment par rapport à A3. Nous observons des économies de temps substantielles dans l'exécution de Q3 sur un grand volume de données par rapport à l'architecture A2, soulignant ainsi l'avantage de l'utilisation d'un graphe natif pour les données externes en combinaison avec la propriété `owl:sameAs`.

La cinquième architecture (A5), illustrée dans la Figure 5.16, se compose de deux graphes natifs : l'un pour les données démographiques et l'autre pour les données de ventes et de com-

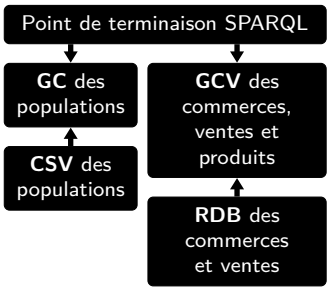


FIGURE 5.15 – Schéma d'architecture A4

Requête	Petit Volume	Gros Volume
Q1	0.5s	0.5s
Q2	36.7s	2m 18.6s
Q3	53.7s	2m 33.5s

TABLE 5.5 – Temps d'exécution des requêtes pour A4

merces. Bien que cette architecture soit peu probable en production en raison de son coût élevé en stockage et en maintenance, elle offre des performances intéressantes et permet d'évaluer l'impact de la virtualisation par rapport à une architecture entièrement native.

Il est particulièrement complexe de transformer un grand volume de données en triplets RDF puis de les importer dans une base de données, une opération qui peut prendre plusieurs heures. Toutefois, les résultats présentés dans le tableau 5.6 montrent une amélioration significative du temps d'exécution de la requête Q3 sur de grands volumes de données. Cette amélioration peut être expliquée par l'utilisation de deux solutions de stockage RDF natif, contrairement à l'architecture A4 où les données internes étaient virtualisées. Cependant, la requête Q2 devient plus lente par rapport à A4, ce qui indique que l'utilisation d'une base de données relationnelle pour interroger exclusivement les données de vente présente des avantages en termes de performances.

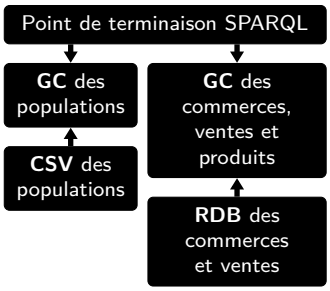


FIGURE 5.16 – Schéma d'architecture A5

Requête	Petit Volume	Gros Volume
Q1	0.5s	2.9s
Q2	1m 19s	4m 16s
Q3	1m 0.4s	1m 32.3s

TABLE 5.6 – Temps d'exécution des requêtes pour A5

La sixième architecture (A6), illustrée dans la figure 5.17, est constituée d'un unique graphe de connaissance intégrant à la fois les données de ventes, de commerces et démographiques. Cette configuration nous permet d'évaluer les performances d'une architecture entièrement native. Bien que cette solution partage avec l'architecture A5 les inconvénients de duplication des données et de complexité de configuration, elle est moins complexe à mettre en place.

Dans cette architecture, la requête Q3 ne nécessite pas de jointure, étant identique à celle de l'architecture A2. Étonnamment, le temps d'exécution de la requête Q3 sur de grands volumes de données est supérieur à celui observé pour les architectures A2, A4 et A5, comme le montre le tableau 5.7.

L'analyse des résultats obtenus en comparant les six architectures permet d'affiner le choix

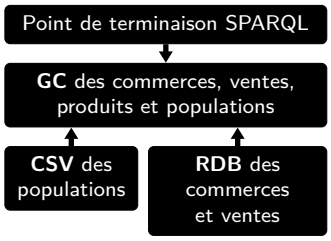


FIGURE 5.17 – Schéma d'architecture A6

Requête	Petit Volume	Gros Volume
Q1	0.5s	0.5s
Q2	1m23.6s	4m07s
Q3	58s	4m11s

TABLE 5.7 – Temps d'exécution des requêtes pour A6

de l'architecture d'implémentation pour Orisha Retail Shops. Dans la section suivante, nous discutons du choix retenu.

5.3 Analyse des résultats : choix de l'architecture

Compte tenu des résultats précédents, ous avons choisi de mettre en œuvre l'architecture technique A4, qui combine un GCV pour les données internes et un graphe natif pour les données externes. Cette architecture présente plusieurs avantages : (1) elle simplifie les requêtes par rapport à l'architecture A3 ; (2) elle sépare les sources de données internes et externes, ce qui la rend plus avantageuse que les architectures A2 et A6 ; (3) en évitant la duplication des données de ventes, elle offre une configuration rapide et une maintenance facilitée par rapport aux architectures A5 et A6. De plus, en termes de temps de réponse cumulé, la solution A4 surpasse toutes les autres architectures, avec une amélioration de 13% par rapport à l'architecture la plus performante suivante, A2.

Lors de l'utilisation de l'outil Ontop-VKG et d'un fichier de mapping R2RML, les performances des GCV peuvent varier considérablement en fonction des facteurs suivants :

- la construction des requêtes SQL définies dans le fichier de mapping ;
- la présence et l'utilisation d'index dans la base de données relationnelle contenant les données à virtualiser ;
- la structure du mapping, incluant l'utilisation de clés primaires et secondaires pour éviter les auto-jointures lors de la traduction des requêtes SPARQL en requêtes SQL par le moteur de réécriture Ontop-VKG.

Le principal défi de la mise en œuvre de cette architecture réside donc dans la construction du script de mapping.

5.4 Amélioration de l'utilisabilité

Faciliter l'accès à des données complexes, issues de sources diverses et ayant parfois subi des traitements algorithmiques, constitue un objectif majeur pour Orisha Retail Shops. Ce projet vise non seulement à optimiser le temps des collaborateurs spécialisés en science des données, mais également à renforcer l'orientation de l'entreprise vers une stratégie centrée sur les données. Cela implique l'intégration systématique des données dans les processus décisionnels et opérationnels de l'entreprise.

À ce stade, l'essentiel du développement nécessaire à l'implémentation d'une interface utilisateur simple et intuitive pour les collaborateurs d'Orisha Retail Shops a été réalisé. Nos graphes de connaissances, déjà élaborés et déployés, sont prêts à être intégrés via une solution d'interface utilisateur.

Nous avons opté pour *Sparnatural*[11], un outil distribué sous licence *LGPL-3.0*. Ce projet facilite l'accès aux graphes de connaissances pour tous les utilisateurs, qu'ils soient initiés ou non aux principes du domaine, grâce à une interface graphique intuitive composée de menus déroulants et d'opérateurs facilement combinables.

Dans la figure 5.18, nous illustrons une application pratique de cette interface à travers une requête destinée à identifier tous les points de vente à Paris qui ont commercialisé plus de 200 articles de papeterie en mars 2024. La figure 5.19 montre la requête SPARQL générée automatiquement par l'outil, démontrant ainsi la facilité de formulation des requêtes complexes.

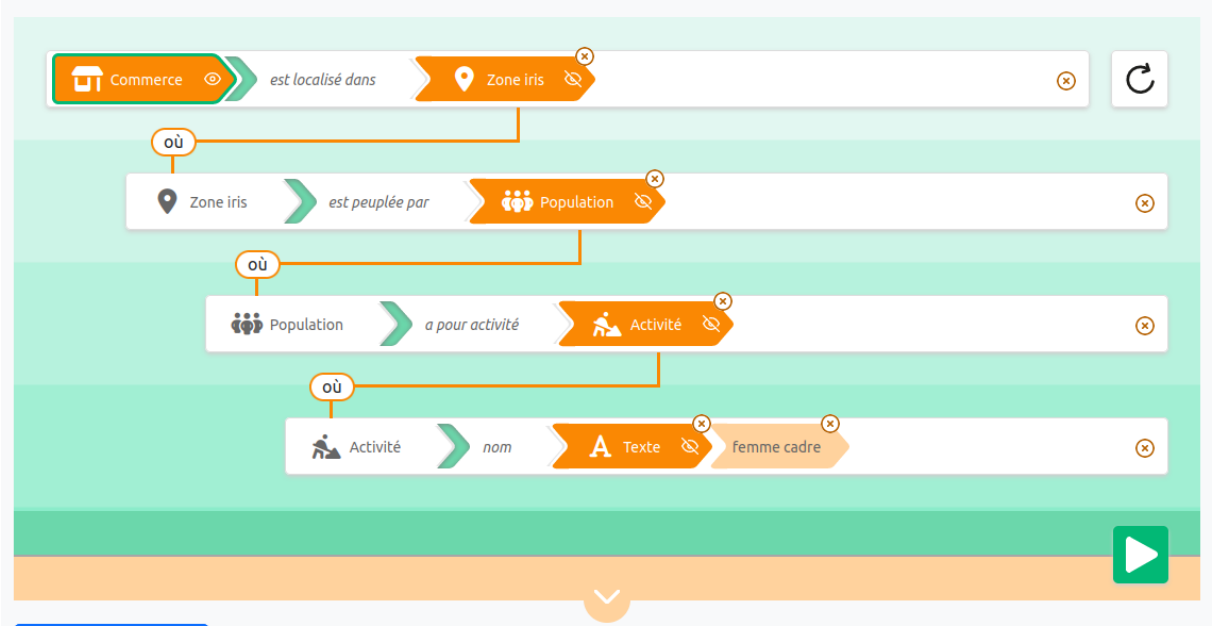


FIGURE 5.18 – Exemple de requête sur un graphe avec l'outil Sparnatural

Nous avons également paramétré le modèle de langage GPT-4 afin qu'il puisse interpréter le schéma spécifique de nos graphes de connaissances. Cette configuration permet aux utilisateurs de communiquer en langage naturel à travers une interface de type chatbot, conçue avec le framework Flask. Le processus commence par la conversion des requêtes textuelles des utilisateurs en requêtes SPARQL. Ces requêtes sont ensuite exécutées sur nos graphes de connaissances, et les résultats sont présentés sous forme de tableaux ou résumés en phrases descriptives, facilitant ainsi l'interprétation et l'interaction.

11. https://sparnatural.eu/ et https://github.com/sparna-git/Sparnatural

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT ?Store_1 ?Store_1_label WHERE {
  ?Store_1 rdf:type <http://test.org/products_and_sales_ontology/Store>.
  OPTIONAL {
    ?Store_1 rdfs:label ?Store_1_label_lang.
    FILTER((LANG(?Store_1_label_lang)) = "fr")
  }
  OPTIONAL {
    ?Store_1 rdfs:label ?Store_1_label_defaultLang.
    FILTER((LANG(?Store_1_label_defaultLang)) = "en")
  }
  BIND(COALESCE(?Store_1_label_lang, ?Store_1_label_defaultLang) AS ?Store_1_label)
  ?Store_1 <http://test.org/products_and_sales_ontology/author> ?IrisZone_2.
  ?IrisZone_2 rdf:type <http://test.org/products_and_sales_ontology/IrisZone>;
    (<http://test.org/products_and_sales_ontology/isPopulatedBy>/<http://test.org/products_and_sales_ontology/Population>) ?
Population_4.
  ?Population_4 rdf:type <http://test.org/products_and_sales_ontology/Population>;
    <http://test.org/products_and_sales_ontology/hasActivity> ?PopulationActivity_6.
  ?PopulationActivity_6 rdf:type <http://test.org/products_and_sales_ontology/PopulationActivity>;
    rdfs:label ?Text_8.
  FILTER((LANG(?Text_8)) = "fr")
  FILTER(REGEX(STR(?Text_8), "femme cadre", "i"))
}
```

FIGURE 5.19 – Exemple de requête SPARQL produite avec l'outil Sparnatural

6 Conclusion et perspectives

Les études menées par Orisha Retail Shops nécessitent non seulement d'être capables de catégoriser l'activité des commerces, mais aussi d'expliciter les concepts nécessaires à cette analyse. C'est cette deuxième tâche que nous avons abordée dans ce chapitre. Les technologies du web sémantique étant conçues à cet effet, nous nous sommes intéressés à leur mise en œuvre dans un cas réel pour l'entreprise Orisha Retail Shops. Bien que ces technologies aient été développées au cours des deux dernières décennies, nous avons constaté que peu d'études proposent une démarche de mise en œuvre dans un contexte réel, ainsi qu'une évaluation du passage à l'échelle des différentes architectures d'implémentation possibles.

Pour combler ces lacunes, nous avons donc proposé une méthodologie de mise en œuvre en quatre étapes : conception des ontologies nécessaires, intégration des données internes, puis externes, et enfin fédération des graphes de connaissances construits afin de les exploiter via des requêtes ou en langage naturel. À chaque étape, nous avons présenté les choix possibles en termes d'outils et de méthodes, et préconisé ce qui nous semblait le plus pertinent. Nous avons également souligné les limites de ces technologies, comme l'impossibilité de combiner différents graphes de connaissances dans de nombreux outils, ou la nécessité de saturer les graphes de connaissances en raison du peu de moteurs d'inférence intégrés dans la plupart des outils.

Dans cette étude, nous avons également montré que différentes architectures d'implémentation étaient possibles, se différenciant essentiellement par la virtualisation ou non du graphe de connaissances, et la séparation ou non des données internes et externes. Nous avons mené des expérimentations pour éclairer ce choix d'implémentation, ce qui nous a conduits à évaluer six architectures sur deux volumes de données réelles et conséquentes. Pour faciliter la reproductibilité de ces expérimentations, l'ensemble des développements est disponible sur la forge du LIAS.

En termes de résultats, nous avons observé qu'une architecture combinant un graphe virtuel

pour les données internes et un graphe natif pour les données externes offrait les meilleures performances, tout en facilitant la maintenance, puisque les données internes ne sont pas dupliquées. La limite de cette approche est la gestion de l'évolution des données externes, car elles sont dupliquées dans un graphe natif. Les sources externes considérées par Orisha Retail Shops évoluent peu, car il s'agit essentiellement de données démographiques. Cependant, dans des cas d'étude nécessitant des données externes évoluant fréquemment, ce choix d'architecture pourrait être remis en cause.

Ces travaux ouvrent plusieurs perspectives pour améliorer la mise en œuvre de solutions basées sur des graphes de connaissances dans des contextes industriels. Parmi elles, l"utilisation des grands modèles de langage (LLM) pourrait faciliter la transformation des requêtes textuelles des utilisateurs en requêtes SPARQL adaptées à l'architecture technique. Par ailleurs, il serait intéressant de développer un moteur de transformation de règles capable de convertir les règles SWRL en règles de mapping R2RML, permettant ainsi d'exprimer certaines règles directement au niveau de l'ontologie plutôt que dans le fichier de mapping.

Conclusion et perspectives générales

Cette thèse a exploré les défis liés à la valorisation des données dans le secteur du commerce de détail, en se concentrant sur la classification précise des commerces par activités et la catégorisation des produits. À travers une collaboration étroite avec Orisha Retail Shops, nous avons mis en lumière les limitations des approches existantes et proposé des solutions pour répondre aux besoins spécifiques de l'entreprise. Plus précisément, nous avons réalisé les contributions suivantes.

Élaboration d'un banc d'essai avec des données réelles

Notre première contribution concerne la catégorisation des produits vendus par les commerçants utilisant les solutions d'Orisha Retail Shops. Ces commerçants définissent librement les libellés et catégories de leurs produits, ce qui leur offre une autonomie et une flexibilité précieuses pour adapter l'organisation de leur inventaire à leurs besoins spécifiques. Cependant, cette liberté complique la consolidation d'une vue globale de l'activité commerciale des commerces, un élément clé pour de nombreuses activités stratégiques d'Orisha Retail Shops.

Bien que diverses approches de classification basées sur des techniques d'apprentissage automatique aient été proposées pour traiter ce type de problème, notre étude de l'état de l'art révèle que les bancs d'essai, définis pour évaluer ces solutions, ne reflètent pas les caractéristiques des données d'Orisha Retail Shops. Ces données se caractérisent par une grande liberté dans les libellés, des problèmes de synonymie, d'homonymie, et une haute cardinalité des catégories, des éléments souvent rencontrés dans les données industrielles. Plusieurs membres du groupe Orisha Retail Shops, traitant des données dans des contextes variés, confirment également que ces particularités ne sont pas uniques aux commerces de tabac-presse et de boulangerie, mais courantes dans d'autres secteurs.

Pour pallier cette lacune, nous avons élaboré un banc d'essai conséquent, basé sur des données réelles, afin d'évaluer les approches de classification et d'identifier leur pertinence pour résoudre le problème spécifique de catégorisation des produits chez Orisha Retail Shops. Ce banc d'essai a été conçu pour représenter de manière fidèle l'ensemble des données d'Orisha tout en restant gérable pour un étiquetage manuel par des experts. En testant les méthodes de classification existantes sur ce banc d'essai, nous avons mis en évidence leurs limites sur des données de qualité variable et complexe, ce qui nous a amené à explorer de nouvelles approches mieux adaptées à ces caractéristiques.

Proposition d'une méthode d'encodage Thesaurus-BT

Face aux limites des solutions de classification identifiées grâce au banc d'essai de notre première contribution, nous avons développé une approche alternative : la méthode Thesaurus-BT. Ce transformateur de données pour l'encodage des produits repose sur un thésaurus regroupant des connaissances métier d'Orisha Retail Shops, permettant ainsi d'améliorer significativement la qualité de l'encodage des libellés de produits. Bien que conçue pour le contexte spécifique de l'entreprise, cette démarche pourrait être transposée à d'autres contextes industriels confrontés à des défis similaires.

Thesaurus-BT présente une caractéristique cruciale en contexte industriel : elle garantit un niveau d'explicabilité en offrant une traçabilité complète des décisions de classification, permettant une interprétation des résultats et un suivi des valeurs intermédiaires utilisées dans le processus de catégorisation. De plus, cette transparence s'accompagne d'une maintenabilité facilitée, le modèle pouvant être mis à jour via des ajustements du thésaurus, sans nécessiter d'expertise en apprentissage automatique.

L'intégration de la méthode Thesaurus-BT dans le système d'information d'Orisha Retail Shops a permis d'affiner les analyses sur les types de produits et les activités commerciales. Cette amélioration a eu un impact positif dans des domaines stratégiques pour l'entreprise, notamment la valorisation des données et la diffusion ciblée de publicités sur les écrans des commerces équipés de la solution Bimedia, contribuant ainsi à l'optimisation des stratégies de vente et de marketing.

Mise en œuvre et évaluation des technologies du Web sémantique

La classification des activités commerciales, permise par nos deux premières contributions, offre de nouvelles possibilités d'études visant à optimiser des domaines comme la prospection et la régie publicitaire. Toutefois, ces études nécessitent, d'une part, une explicitation de concepts qui, bien que présents dans les sources de données, restent non formalisés et, d'autre part, une intégration de multiples sources de données internes et externes. Conçues pour répondre à ces défis, les technologies du Web sémantique ont retenu notre intérêt.

Nous avons constaté qu'il existe peu de travaux traitant de la mise en œuvre de ces technologies dans des environnements industriels, avec notamment un manque de méthodologie de déploiement et d'évaluation des performances des différentes architectures possibles. Notre contribution s'inscrit ainsi dans l'intégration des technologies du web sémantique pour l'analyse des données d'Orisha Retail Shops, afin d'expliciter les concepts clés et de permettre diverses analyses commerciales. Dans un contexte où la diversité des données et des sources d'information est la norme, nous proposons une méthodologie de mise en œuvre en quatre étapes : la conception des ontologies, l'intégration des données internes et externes, et la fédération des graphes de connaissances. Cette approche constitue une feuille de route pour les entreprises cherchant à utiliser les technologies sémantiques pour surmonter les défis de l'hétérogénéité des données.

Enfin, nous avons exploré plusieurs architectures d'implémentation, notamment les choix entre virtualisation des graphes de connaissances et gestion séparée des données internes et

externes. Une évaluation expérimentale de six architectures sur des données réelles a permis d'identifier une solution hybride combinant un graphe virtuel pour les données internes et un graphe natif pour les données externes. Cette approche optimise les performances et simplifie la maintenance des données internes, en évitant leur duplication.

En somme, les contributions abordées ci-dessus ont donné naissance à plusieurs productions scientifiques et techniques. Dans le domaine scientifique, deux publications ont été réalisées :

— *Thesaurus-based Transformation : A Classification Method for Real Dirty Data*, publié à la *European Conference on Advances in Databases and Information Systems (ADBIS)* [Perrot, 2023] ;

— *Knowledge Graphs for Data Integration in Retail*, publié au *International Symposium on Methodologies for Intelligent Systems (ISMIS)* [Perrot, 2024].

Sur le plan technique, plusieurs réalisations ont été accomplies.

— L'algorithme Thesaurus-BT a été adapté pour fonctionner en calcul distribué et a été mis en production ; il est déclenché tous les dimanches à minuit. Sa version originale et tout le nécessaire relatif aux expérimentations sont disponibles dans un dépôt *GitHub* sur la forge du LIAS [12].

— Les graphes de connaissances des données internes et externes ont également été déployés, intégrant notamment les résultats des calculs de Thesaurus-BT. Les informations qu'ils renferment sont accessibles par trois moyens : un outil permettant le requêtage des données de manière graphique, un point de terminaison SPARQL, et un chatbot, également développé pendant cette thèse, qui traduit les demandes textuelles des utilisateurs en requêtes et restitue les résultats correspondants. Le programme développé pour la construction et le peuplement des graphes de connaissances, ainsi que celui relatif aux expérimentations, est disponible dans un dépôt *GitHub* sur la forge du LIAS [13].

Ouverture et perspectives

Les travaux réalisés dans cette thèse ouvrent plusieurs perspectives techniques et fonctionnelles. D'une part pour l'amélioration et l'extension des solutions proposées dans ce manuscrit, d'autres part pour la mise en place de nouvelles applications basées tout ou partie sur les travaux.

Amélioration et enrichissement du banc d'essai

Tout d'abord, l'élargissement du banc d'essai par l'extension du jeu de données de validation est une étape essentielle. Enrichir le jeu de données des commerces étiquetés manuellement permettra aux solutions basées sur l'apprentissage supervisé de produire des résultats plus pertinents. Cela pourrait être réalisé en lançant de nouvelles campagnes d'étiquetage impliquant

12. https://forge.lias-lab.fr/thesaurusbt
13. https://forge.lias-lab.fr/retailkgintegration

davantage d'experts métiers pour améliorer la représentativité des différentes activités commerciales.

Ensuite, la création d'un second banc d'essai focalisé sur la classification des produits offre de nouvelles opportunités. En recourant à des méthodes d'annotation semi-automatiques et en utilisant des sources de données ouvertes, nous pourrions constituer un jeu de données significatif sans nécessiter un étiquetage entièrement manuel. Cela permettrait d'explorer des approches basées sur les modèles de langage préentraînés et les techniques de transfert d'apprentissage pour améliorer la précision de la classification des produits.

Extension de Thesaurus-BT

Plusieurs perspectives se dégagent pour les travaux futurs. Bien que nos essais sur les techniques d'apprentissage automatique généralistes n'aient pas fourni les meilleurs résultats dans notre contexte, nous reconnaissons que le domaine évolue rapidement. Depuis nos premières expérimentations en 2022, de nombreux nouveaux grands modèles de langage (LLM) ont été publiés, en particulier dans la communauté open source, et des avancées significatives ont été réalisées. Il serait donc pertinent d'explorer ces modèles plus récents pour déterminer s'ils peuvent fournir des incorporations (embeddings) suffisamment efficaces pour la classification des produits et des commerces, même avec un jeu de données d'apprentissage restreint.

Par ailleurs, nous envisageons également une amélioration de Thesaurus-BT par le développement d'une approche hybride qui consisterait à construire une partie du thésaurus de manière semi-automatique en s'appuyant sur des modèles de langage avancés. L'utilisation de ces modèles pourrait accélérer le processus de développement du thésaurus, tout en améliorant sa couverture et sa précision. Cela permettrait non seulement de répondre efficacement à notre problématique d'encodage des produits, mais aussi de créer une solution adaptable plus facilement à d'autres contextes industriels.

Construction et exploitation des graphes de connaissances

Par ailleurs, il est primordial d'approfondir les travaux que nous avons entamés sur l'intégration des grands modèles de langage (LLM) pour l'interrogation des graphes de connaissances, afin de rendre cette approche exploitable à l'échelle de l'entreprise. Cette avancée améliorerait considérablement l'accessibilité et l'interaction avec les graphes de connaissances, rendant les technologies du web sémantique plus conviviales pour les utilisateurs non experts. De plus, les premiers résultats obtenus sont très prometteurs.

Enfin, le développement d'un moteur de transformation de règles capable de convertir les règles SWRL en règles de mapping R2RML constituerait une avancée notable. Cela permettrait d'exprimer certaines règles directement au niveau de l'ontologie plutôt que dans les fichiers de mapping, simplifiant ainsi la maintenance et l'évolution des connaissances représentées.

Généralisation des travaux aux autres filiales d'Orisha

Une perspective importante est de généraliser les travaux de cette thèse aux autres filiales d'Orisha, qui, bien que dans des contextes différents, rencontrent des problématiques similaires. Les méthodes développées, telles que le banc d'essai personnalisé, l'algorithme Thesaurus-BT et l'utilisation de graphes de connaissances, peuvent être adaptées pour répondre à leurs besoins spécifiques. En ajustant le thésaurus ou l'ontologie aux particularités de chaque filiale, elles pourraient améliorer la précision de leurs analyses de données et optimiser leurs processus décisionnels, renforçant ainsi la synergie au sein du groupe Orisha.

Valorisation en temps réel des espaces publicitaires

Cette dernière perspective concerne un nouveau type d'application rendu possible par les travaux de cette thèse.

Les enchères en temps réel (Real-Time Bidding, ou RTB) consistent à vendre en temps réel et au plus offrant une impression publicitaire sur un espace publicitaire numérique. L'entreprise Orisha Retail Shops, par le biais de sa filiale de régie publicitaire Orisha ADgency, travaille depuis quelque temps sur la mise en place de cette pratique pour la commercialisation des espaces publicitaires présents dans les commerces équipés de ses solutions (notamment grâce à l'écran de la caisse enregistreuse tourné du côté du client). Aujourd'hui, le RTB ne représente que 4% du chiffre d'affaires publicitaire de l'entreprise, les ventes d'espaces se faisant en majeure partie grâce au travail des chefs de publicité. Cependant, une forte progression de ce processus de vente automatisé est attendue à l'avenir.

Or, les travaux de cette thèse ont permis non seulement une caractérisation fine des activités des commerces ainsi que la classification des produits, mais aussi l'intégration de manière transparente de diverses informations sur les commerces et leur environnement, issues de sources de données internes et externes. L'ensemble de ces informations, désormais centralisé dans des graphes de connaissances interconnectés, est disponible par l'exécution d'une simple requête. Cela permettrait d'augmenter la valeur des espaces publicitaires grâce à l'ajout d'informations importantes pour le ciblage publicitaire recherché par les annonceurs, tout en répondant aux contraintes de performance qu'impose le RTB.

Annexes

1 Statistiques complémentaires

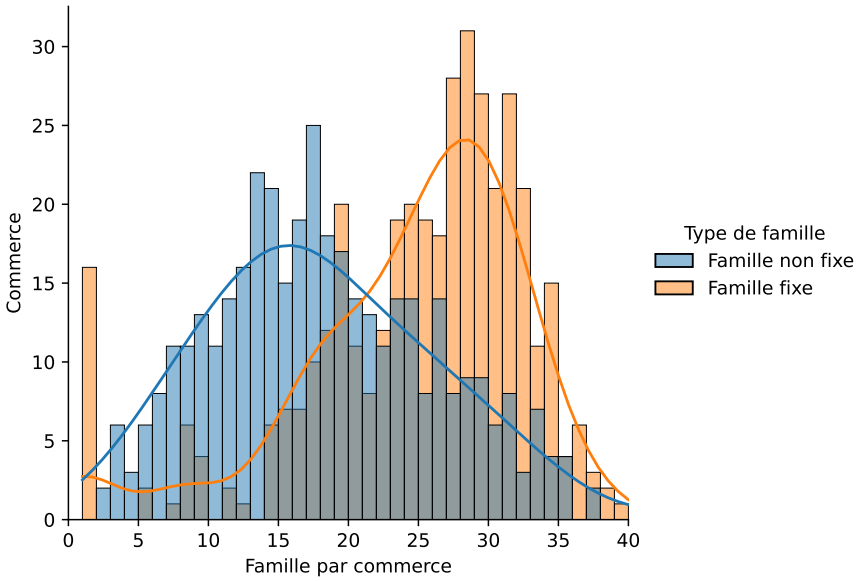


FIGURE 7.20 – Histogramme de la distribution des familles fixes et non fixes

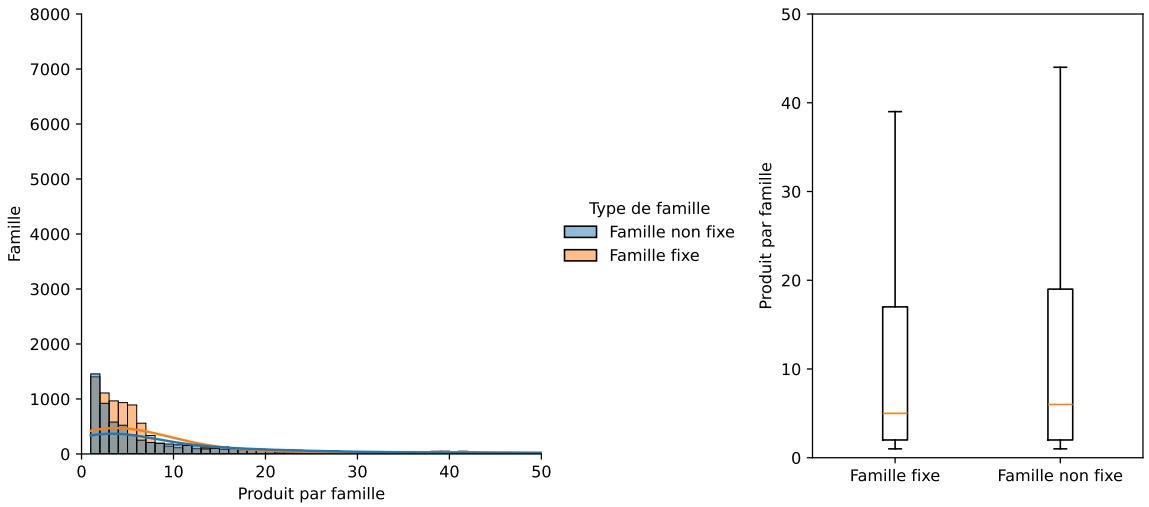


FIGURE 7.21 – Distributions des produits dans les familles représentées par un histogramme (à gauche) et une boîte à moustaches (à droite)

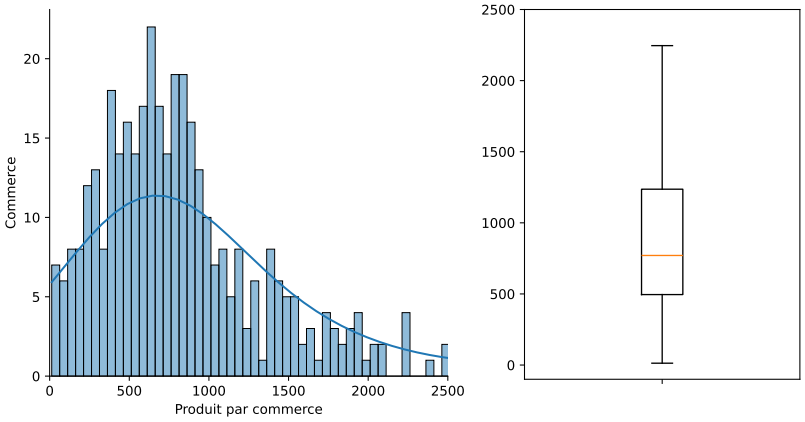


FIGURE 7.22 – Distributions des produits par commerce représentées par un histogramme (à gauche) et une boîte à moustaches (à droite)

2 Normalisation des sorties et traçabilité des résultats

Pour faciliter la comparaison des différents modèles et assurer la traçabilité des résultats, cette section est dédiée aux règles de normalisation des sorties des modèles. Les résultats seront présentés selon les formats suivants :

— Les résultats de l'étiquetage multiple des activités des commerces seront sauvegardés dans un fichier au format CSV, avec le caractère ',' comme séparateur. Ce fichier sera nommé `commerce_labeling_results_XXXX.csv`, où `XXXX` correspond au nom de la solution. La première colonne contiendra les identifiants des commerces, et la seconde colonne les étiquettes prédites. Chaque ligne représentera un commerce unique associé à ses étiquettes.

Les identifiants des commerces seront représentés par leurs *store_id*. Les listes d'étiquettes devront être encadrées par des crochets '[]', chaque étiquette sera entourée de guillemets simples et séparée par une virgule, comme illustré dans le Tableau 7.8.

— Les résultats de l'évaluation des méthodes seront également stockés dans un fichier CSV, avec le caractère ',' comme séparateur. Ce fichier sera nommé `evaluation_metrics_XXXX.csv`, où `XXXX` correspond au nom de la solution. La première colonne contiendra les noms des critères d'évaluation, et la seconde colonne les résultats correspondants. Chaque ligne du fichier représentera un critère d'évaluation spécifique et son résultat associé. Les noms des critères sont : *implementation_time*, *execution_time*, *macro_f1* et *accuracy*.

TABLE 7.8 – Exemple fictif de résultats de classification reformulés

store_id	activity
b25222f7b9c13a99407d72395069a2105453dd17	['tabac', 'presse']
bb8b594b945e52541790de757f3daccf47c1dabb	['bar', 'tabac', 'presse', 'cafe']
7b8f573c39f7484b677771ffdf8ab5e88938d173	['tabac', 'presse', 'bar', 'cafe']
e8bd40f832da6d73679ac68534c06071ded0f0e2	['presse']
1ecfd7b97d95fd8a58777cafd713b5745bea5529	['tabac', 'presse']
1027694455202d133e047693055825d6d63e94b0	['tabac', 'presse']

Ces fichiers normalisés permettent une comparaison facile et systématique des performances des différents modèles, tout en garantissant la traçabilité et la transparence des résultats obtenus.

3 Choix du nombre de cluster pour l'étape de partitionnement avec l'algorithme K-means

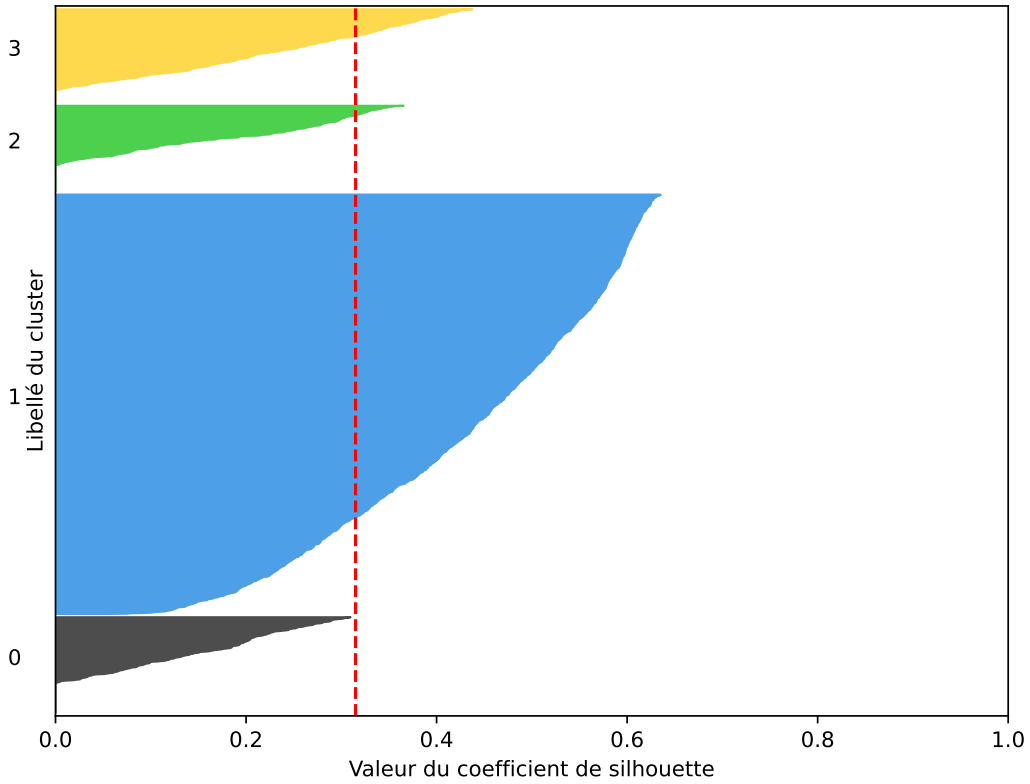


FIGURE 7.23 – Score de silhouette pour 4 clusters (moyenne = 0.31).

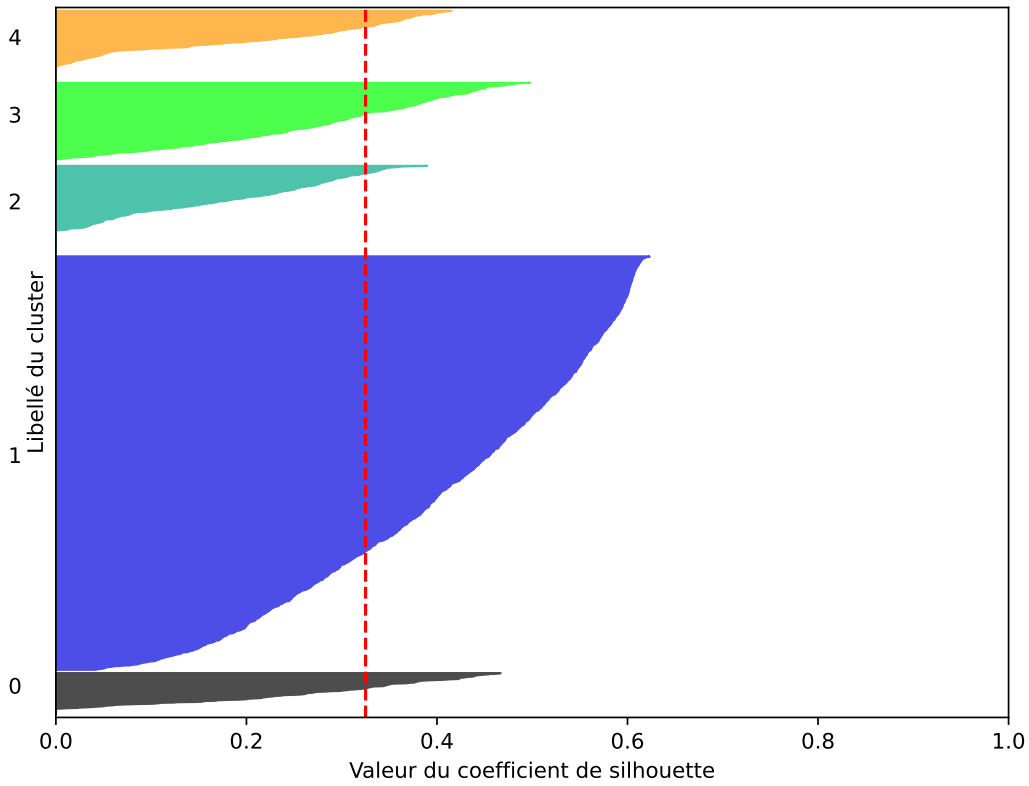


FIGURE 7.24 – Score de silhouette pour 5 clusters (moyenne = 0.33).

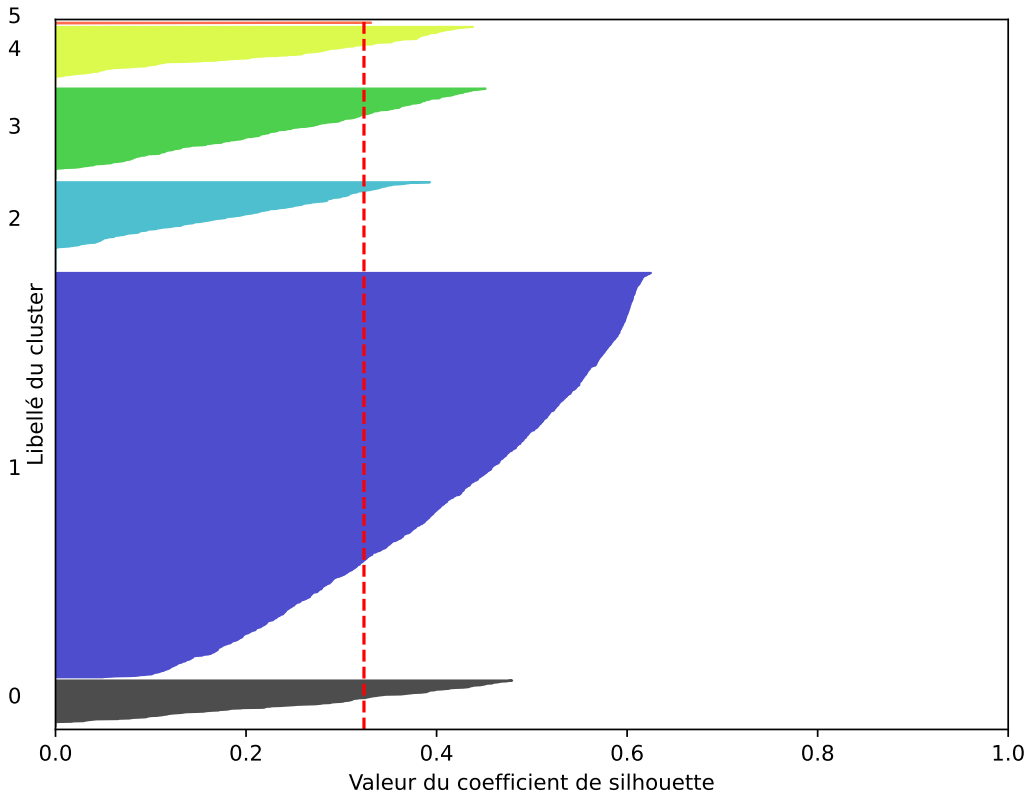


FIGURE 7.25 – Score de silhouette pour 6 clusters (moyenne = 0.32).

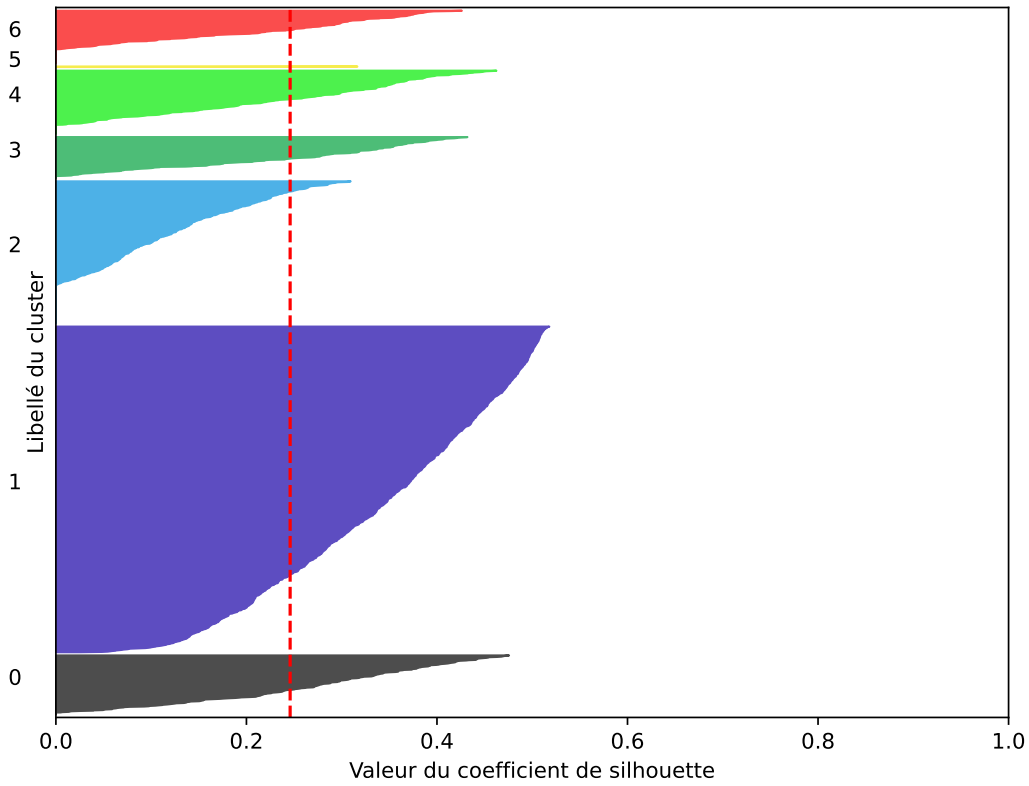


FIGURE 7.26 – Score de silhouette pour 7 clusters (moyenne = 0.25).

Bibliographie

[Abdi, 2010] Hervé Abdi et Lynne J Williams. « Principal component analysis ». *Wiley interdisciplinary reviews : computational statistics* 2.4 (2010), p. 433-459 (cf. p. 33).

[Abu-Salih, 2021] Bilal Abu-Salih. « Domain-specific knowledge graphs : A survey ». *Journal of Network and Computer Applications* 185 (2021), p. 103076 (cf. p. 107).

[Aizawa, 2003] Akiko Aizawa. « An information-theoretic perspective of tf–idf measures ». *Information Processing & Management* 39.1 (2003), p. 45-65 (cf. p. 22).

[Alfaifi, 2022] Yousef Alfaifi. « Ontology Development Methodology : A Systematic Review and Case Study ». *International Conference on Computing and Information Technology (ICCIT'22)*. 2022, p. 446-450 (cf. p. 109).

[Aluç, 2014a] Güneş Aluç, Olaf Hartig, M Tamer Özsu et Khuzaima Daudjee. « Diversified Stress Testing of RDF Data Management Systems ». *Proceedings of the 13th International Semantic Web Conference (ISWC'14)*. 2014, p. 197-212 (cf. p. 101).

[Aluç, 2014b] Güneş Aluç, Olaf Hartig, M Tamer Özsu et Khuzaima Daudjee. « Diversified stress testing of RDF data management systems ». *The Semantic Web–ISWC 2014 : 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I 13*. Springer. 2014, p. 197-212 (cf. p. 3).

[Andrii Samoshyn, 2013] Andrii Samoshyn. *Big Mart Sales Dataset*. https://www.kaggle.com/datasets/mrmorj/big-mart-sales. 2013 (cf. p. 65).

[Bogatinovski, 2022] Jasmin Bogatinovski, Ljupčo Todorovski, Sašo Džeroski et Dragi Kocev. « Comprehensive comparative study of multi-label classification methods ». *Expert Systems with Applications* 203 (2022), p. 117215 (cf. p. 40-43).

[Borgelt, 2012] Christian Borgelt. « Frequent item set mining ». *Wiley interdisciplinary reviews : data mining and knowledge discovery* 2.6 (2012), p. 437-456 (cf. p. 32).

[Borisov, 2022a] Vadim Borisov, Klaus Broelemann, Enkelejda Kasneci et Gjergji Kasneci. « DeepTLF : robust deep neural networks for heterogeneous tabular data ». *International Journal of Data Science and Analytics* (2022), p. 1-16 (cf. p. 29).

[Borisov, 2022b] Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk et Gjergji Kasneci. « Deep Neural Networks and Tabular Data : A Survey ». *IEEE Transactions on Neural Networks and Learning Systems* (2022), p. 1-21 (cf. p. 35).

[Brickley, 2014] Dan Brickley et R V. Guha. *RDF Schema 1.1*. World Wide Web Consortium. 2014 (cf. p. 103).

[Broder, 1997] Andrei Z Broder. « On the resemblance and containment of documents ». *Proceedings. Compression and Complexity of SEQUENCES 1997*. IEEE. 1997, p. 21-29 (cf. p. 30).

[Calvanese, 2017] Diego Calvanese, Benjamin Cogrel, Sarah Komla-Ebri, Roman Kontchakov, Davide Lanti, Martin Rezk et al. « Ontop : Answering SPARQL queries over relational databases ». *Semantic Web* 8.3 (2017), p. 471-487 (cf. p. 3).

[Canny, 2004] John Canny. « GaP : a factor model for discrete data ». *Proceedings of the 27th annual international ACM SIGIR Conference on Research and Development in Information Retrieval*. 2004, p. 122-129 (cf. p. 25, 30).

[Cerda, 2020] Patricio Cerda et Gaël Varoquaux. « Encoding high-cardinality string categorical variables ». *IEEE Transactions on Knowledge and Data Engineering* (2020) (cf. p. 23, 25-27, 30, 45, 91).

[Cerda, 2018] Patricio Cerda, Gaël Varoquaux et Balázs Kégl. « Similarity encoding for learning with dirty categorical variables ». *Machine Learning* 107.8 (2018), p. 1477-1494 (cf. p. viii, 23, 24, 30).

[Chandrashekar, 2014] Girish Chandrashekar et Ferat Sahin. « A survey on feature selection methods ». *Computers & Electrical Engineering* 40.1 (2014), p. 16-28 (cf. p. 33).

[Chaves-Fraga, 2022] David Chaves-Fraga, Oscar Corcho, Francisco Yedro, Roberto Moreno, Juan Olías et Alejandro De La Azuela. « Systematic construction of knowledge graphs for research-performing organizations ». *Information* 13.12 (2022), p. 562 (cf. p. 107).

[Chen, 2016] Tianqi Chen et Carlos Guestrin. « Xgboost : A scalable tree boosting system ». *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016, p. 785-794 (cf. p. 29).

[Cho, 2014] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau et Yoshua Bengio. « On the properties of neural machine translation : Encoder-decoder approaches ». *arXiv preprint arXiv :1409.1259* (2014) (cf. p. 36).

[Church, 2017] Kenneth Ward Church. « Word2Vec ». *Natural Language Engineering* 23.1 (2017), p. 155-162 (cf. p. 26).

[Costa, 2023] Liliane Soares da Costa, Italo L Oliveira et Renato Fileto. « Text classification using embeddings : a survey ». *Knowledge and Information Systems* 65.7 (2023), p. 2761-2803 (cf. p. 29).

[Dan, 2004] Brickley Dan. « RDF vocabulary description language 1.0 : RDF schema ». *http ://www. w3. org/TR/rdf-schema/* (2004) (cf. p. 3).

[Dean, 2004] Mike Dean et Guus Schreiber. *OWL Web Ontology Language Reference*. World Wide Web Consortium. 2004 (cf. p. 104).

[Deshpande, 2013] Omkar Deshpande, Digvijay S. Lamba, Michel Tourn, Sanjib Das, Sri Subramaniam, Anand Rajaraman et al. « Building, Maintaining, and Using Knowledge Bases : A Report from the Trenches ». *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data (SIGMOD'13)*. 2013, p. 1209-1220 (cf. p. 101).

[Dong, 2014] Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy et al. « Knowledge Vault : A Web-scale Approach to Probabilistic Knowledge Fusion ». *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'14)*. 2014, p. 601-610 (cf. p. 101).

[Dwarampudi, 2019] Mahidhar Dwarampudi et NV Reddy. « Effects of padding on LSTMs and CNNs ». *arXiv preprint arXiv :1903.07288* (2019) (cf. p. 33).

[Erling, 2009] Orri Erling et Ivan Mikhailov. « RDF Support in the Virtuoso DBMS ». *Networked Knowledge-Networked Media : Integrating Knowledge Management, New Media Technologies and Semantic Systems*. Springer, 2009, p. 7-24 (cf. p. 3).

[Fishkin, 2017] Alexey Fishkin. *Industrial Knowledge Graph at Siemens CERN Openlab Technical Workshop, Geneva*. 2017. url : https://indico.cern.ch/event/669648/contributions/2838194/attachments/1581790/2499984/CERN_Open_Lab_Technical_Workshop_-_SIEMENS_AG_-_FISHKIN_-_11-01-2018.pdf (visité le 01/11/2018) (cf. p. 107).

[Grandini, 2020] Margherita Grandini, Enrico Bagli et Giorgio Visani. « Metrics for multi-class classification : an overview ». *arXiv preprint arXiv :2008.05756* (2020) (cf. p. 69).

[Grangel-González, 2019] Irlán Grangel-González. « A knowledge graph based integration approach for industry 4.0 ». Thèse de doct. Universitäts-und Landesbibliothek Bonn, 2019 (cf. p. 107).

[Gruber, 1993] Thomas R. Gruber. « A Translation Approach to Portable Ontology Specifications ». *Knowledge Acquisition* 5.2 (1993), p. 199-220 (cf. p. 102).

[Güler, 2005] Nihal Fatma Güler et Sabri Koçer. « Classification of EMG signals using PCA and FFT ». *Journal of medical systems* 29.3 (2005), p. 241-250 (cf. p. 33).

[Guo, 2005a] Yuanbo Guo, Zhengxiang Pan et Jeff Heflin. « LUBM : A Benchmark for OWL Knowledge Base Systems ». *Web Semantics* 3.2-3 (2005), p. 158-182 (cf. p. 101).

[Guo, 2005b] Yuanbo Guo, Zhengxiang Pan et Jeff Heflin. « LUBM : A benchmark for OWL knowledge base systems ». *Journal of Web Semantics* 3.2-3 (2005), p. 158-182 (cf. p. 3).

[Gupta, 2016] Varun Gupta et Monika Mittal. « Respiratory signal analysis using PCA, FFT and ARTFA ». *2016 International Conference on Electrical Power and Energy Systems (ICEPES)*. IEEE. 2016, p. 221-225 (cf. p. 33).

[Hammer, 2000] Barbara Hammer. « On the approximation capability of recurrent neural networks ». *Neurocomputing* 31.1-4 (2000), p. 107-123 (cf. p. 36).

[Hoffart, 2013] Johannes Hoffart, Fabian M. Suchanek, Klaus Berberich et Gerhard Weikum. « YAGO2 : A Spatially and Temporally Enhanced Knowledge Base from Wikipedia ». *Artificial Intelligence* 194 (2013), p. 28-61 (cf. p. 101, 109).

[Hogan, 2020] Aidan Hogan. « The semantic web : two decades on ». *Semantic Web* 11.1 (2020), p. 169-185 (cf. p. 3).

[Hubauer, 2018] Thomas Hubauer, Steffen Lamparter, Peter Haase et Daniel Markus Herzig. « Use Cases of the Industrial Knowledge Graph at Siemens. » *ISWC (P&D/Industry/BlueSky)*. 2018 (cf. p. 107, 108).

[Jordan, 2015] Michael I Jordan et Tom M Mitchell. « Machine learning : Trends, perspectives, and prospects ». *Science* 349.6245 (2015), p. 255-260 (cf. p. 2).

[Kocev, 2013] Dragi Kocev, Celine Vens, Jan Struyf et Sašo Džeroski. « Tree ensembles for predicting structured outputs ». *Pattern Recognition* 46.3 (2013), p. 817-833 (cf. p. 43).

[Lehmann, 2015] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes et al. « DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia ». *Semantic Web* 6.2 (2015), p. 167-195 (cf. p. 101, 109).

[Li, 2021] Xinyu Li, Mengtao Lyu, Zuoxu Wang, Chun-Hsien Chen et Pai Zheng. « Exploiting knowledge graphs in industrial products and services : a survey of key aspects, challenges, and future perspectives ». *Computers in Industry* 129 (2021), p. 103449 (cf. p. 107).

[Liu, 2016] Cong Liu, Chunxue Wu et Linhua Jiang. « Evolutionary clustering framework based on distance matrix for arbitrary-shaped data sets ». *IET Signal Processing* 10.5 (2016), p. 478-485 (cf. p. 34).

[Madjarov, 2012] Gjorgji Madjarov, Dragi Kocev, Dejan Gjorgjevikj et Sašo Džeroski. « An extensive experimental comparison of methods for multi-label learning ». *Pattern recognition* 45.9 (2012), p. 3084-3104 (cf. p. 40, 43).

[McBride, 2002] Brian McBride. « Jena : A semantic web toolkit ». *IEEE Internet computing* 6.6 (2002), p. 55-59 (cf. p. 3).

[McGuinness, 2004] Deborah L McGuinness. « OWL Web Ontology Language Overview ». *W3C Member Submission* (2004) (cf. p. 3).

[Mendes de Farias, 2023] Tarcisio Mendes de Farias, Julien Wollbrett, Marc Robinson-Rechavi et Frederic Bastian. « Lessons learned to boost a bioinformatics knowledge base reusability, the Bgee experience ». *GigaScience* 12 (2023), giad058 (cf. p. 107).

[Mikolov, 2013] Tomas Mikolov. « Efficient estimation of word representations in vector space ». *arXiv preprint arXiv :1301.3781* (2013) (cf. p. 3).

[Moens, 2013] Sandy Moens, Emin Aksehirli et Bart Goethals. « Frequent itemset mining for big data ». *2013 IEEE international conference on big data*. IEEE. 2013, p. 111-118 (cf. p. 32).

[Olson, 2017] Randal S Olson, William La Cava, Patryk Orzechowski, Ryan J Urbanowicz et Jason H Moore. « PMLB : a large benchmark suite for machine learning evaluation and comparison ». *BioData mining* 10 (2017), p. 1-13 (cf. p. 2).

[Ontop, 2017] Ontop. *Ontop-VKG*. 2017. url : https://ontop-vkg.org (visité le 01/01/2023) (cf. p. 113).

[OpenLink, 2013] OpenLink. *OpenLink Virtuoso*. 2013. url : https://virtuoso.openlinksw.com/ (visité le 01/01/2023) (cf. p. 113).

[Pachpande, 2022] Soham Pachpande et Gehan Chopade. « Categorical Data Deduplication ». *unknown* (2022) (cf. p. viii, 23, 24, 30).

[Pargent, 2022] Florian Pargent, Florian Pfisterer, Janek Thomas et Bernd Bischl. « Regularized target encoding outperforms traditional methods in supervised machine learning with high cardinality features ». *Computational Statistics* 37.5 (2022), p. 2671-2692 (cf. p. 45).

[Pennington, 2014] Jeffrey Pennington, Richard Socher et Christopher D Manning. « Glove : Global vectors for word representation ». *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014, p. 1532-1543 (cf. p. 26).

[Perrot, 2023] Maxime Perrot, Mickaël Baron, Brice Chardin et Stéphane Jean. « Thesaurus-Based Transformation : A Classification Method for Real Dirty Data ». *European Conference on Advances in Databases and Information Systems*. Springer. 2023, p. 256-265 (cf. p. 131).

[Perrot, 2024] Maxime Perrot, Mickaël Baron, Brice Chardin et Stéphane Jean. « Knowledge Graphs for Data Integration in Retail ». *International Symposium on Methodologies for Intelligent Systems*. Springer. 2024, p. 231-245 (cf. p. 131).

[Petukhova, 2024] Alina Petukhova, Joao P Matos-Carvalho et Nuno Fachada. « Text clustering with LLM embeddings ». *arXiv preprint arXiv :2403.15112* (2024) (cf. p. 29).

[Popov, 2019] Sergei Popov, Stanislav Morozov et Artem Babenko. « Neural oblivious decision ensembles for deep learning on tabular data ». *arXiv preprint arXiv :1909.06312* (2019) (cf. p. 29).

[Prokhorenkova, 2018] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush et Andrey Gulin. « CatBoost : unbiased boosting with categorical features ». *Advances in neural information processing systems* 31 (2018) (cf. p. 29).

[Pyle, 1999] Dorian Pyle. *Data preparation for data mining*. morgan kaufmann, 1999 (cf. p. 23).

[Rahm, 2000] Erhard Rahm et Hong Hai Do. « Data cleaning : Problems and current approaches ». *IEEE Data Eng. Bull.* 23.4 (2000), p. 3-13 (cf. p. 23).

[Raziff, 2017] Abdul Rafiez Abdul Raziff, Md Nasir Sulaiman, Norwati Mustapha et Thinagaran Perumal. « Single classifier, OvO, OvA and RCC multiclass classification method in handheld based smartphone gait identification ». *AIP Conference Proceedings*. T. 1891. 1. AIP Publishing LLC. 2017, p. 020009 (cf. p. 39).

[Read, 2011] Jesse Read, Bernhard Pfahringer, Geoff Holmes et Eibe Frank. « Classifier chains for multi-label classification ». *Machine learning* 85.3 (2011), p. 333-359 (cf. p. 40, 42).

[Reilly, 2022] Denis Reilly, Mark Taylor, Paul Fergus, Carl Chalmers et Steven Thompson. « The categorical data conundrum : Heuristics for classification problems—A case study on domestic fire injuries ». *IEEE Access* 10 (2022), p. 70113-70125 (cf. p. 23).

[Sparck Jones, 1972] Karen Sparck Jones. « A statistical interpretation of term specificity and its application in retrieval ». *Journal of documentation* 28.1 (1972), p. 11-21 (cf. p. 3).

[Sun, 2019] Baohua Sun, Lin Yang, Wenhan Zhang, Michael Lin, Patrick Dong, Charles Young et al. « Supertml : Two-dimensional word embedding for the precognition on structured tabular data ». *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 2019 (cf. p. 35).

[Tenney, 2019] Ian Tenney, Dipanjan Das et Ellie Pavlick. « BERT Rediscovers the Classical NLP Pipeline ». *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2019, p. 4593-4601 (cf. p. 26, 27).

[Tsoumakas, 2007] Grigorios Tsoumakas et Ioannis Katakis. « Multi-label classification : An overview ». *International Journal of Data Warehousing and Mining (IJDWM)* 3.3 (2007), p. 1-13 (cf. p. 39).

[Tsoumakas, 2008] Grigorios Tsoumakas, Ioannis Katakis et Ioannis Vlahavas. « Effective and efficient multilabel classification in domains with large number of labels ». *Proc. ECML/PKDD 2008 Workshop on Mining Multidimensional Data (MMD'08)*. T. 21. 2008, p. 53-59 (cf. p. 41).

[Tsoumakas, 2009] Grigorios Tsoumakas, Ioannis Katakis et Ioannis Vlahavas. « Mining multi-label data ». *Data mining and knowledge discovery handbook* (2009), p. 667-685 (cf. p. 39).

[Turney, 2006] Peter D Turney. « Similarity of semantic relations ». *Computational Linguistics* 32.3 (2006), p. 379-416 (cf. p. 23).

[Turney, 2008] Peter D Turney. « A uniform approach to analogies, synonyms, antonyms, and associations ». *22nd International Conference on Computational Linguistics (COLING-08)* (2008) (cf. p. 23, 30).

[Visbal-Cadavid, 2020] Delimiro Visbal-Cadavid, Adel Mendoza-Mendoza et Enrique De La Hoz-Dominguez. « Use of Factorial Analysis of Mixed Data (FAMD) and Hierarchical Cluster Analysis on Principal Component (HCPC) for Multivariate Analysis of Academic Performance of Industrial Engineering Programs ». *Journal of Southwest Jiaotong University* 55.5 (2020) (cf. p. 33).

[Vogt, 2023] Lars Vogt, Marcel Konrad et Manuel Prinz. « Knowledge Graph Building Blocks : An easy-to-use Framework for developing FAIREr Knowledge Graphs ». *arXiv preprint arXiv :2304.09029* (2023) (cf. p. 107).

[W3C, 2012] W3C. *R2RML : RDB to RDF Mapping Language W3C Recommendation*. 2012. URL : https://https://www.w3.org/TR/r2rml/ (visité le 27/09/2012) (cf. p. 114).

[Weikum, 2021] Gerhard Weikum, Xin Luna Dong, Simon Razniewski et Fabian Suchanek. « Machine Knowledge : Creation and Curation of Comprehensive Knowledge Basess ». *Foundations and Trends® in Databases* 10.2-4 (2021), p. 108-490 (cf. p. 109).

[Wu, 1992] Sun Wu et Udi Manber. « Fast text searching : allowing errors ». *Communications of the ACM* 35.10 (1992), p. 83-91 (cf. p. 27).

[Xiao, 2022] Guohui Xiao, Emily Pfaff, Eric Prud'hommeaux, David Booth, Deepak K Sharma, Nan Huo et al. « FHIR-Ontop-OMOP : Building clinical knowledge graphs in FHIR RDF with the OMOP Common data Model ». *Journal of Biomedical Informatics* 134 (2022), p. 104201 (cf. p. 107).

[Yahya, 2021] Muhammad Yahya, John G Breslin et Muhammad Intizar Ali. « Semantic web and knowledge graphs for industry 4.0 ». *Applied Sciences* 11.11 (2021), p. 5110 (cf. p. 107).

[Yin, 2020] Pengcheng Yin, Graham Neubig, Wen-tau Yih et Sebastian Riedel. « TaBERT : Pretraining for joint understanding of textual and tabular data ». *arXiv preprint arXiv :2005.08314* (2020) (cf. p. 34).

[Yoon, 2020] Jinsung Yoon, Yao Zhang, James Jordon et Mihaela van der Schaar. « Vime : Extending the success of self-and semi-supervised learning to tabular domain ». *Advances in Neural Information Processing Systems* 33 (2020), p. 11033-11043 (cf. p. 35).

[Zhang, 2018] Min-Ling Zhang, Yu-Kun Li, Xu-Ying Liu et Xin Geng. « Binary relevance for multi-label learning : an overview ». *Frontiers of Computer Science* 12.2 (2018), p. 191-202 (cf. p. 40).

[Zhang, 2013] Min-Ling Zhang et Zhi-Hua Zhou. « A review on multi-label learning algorithms ». *IEEE transactions on knowledge and data engineering* 26.8 (2013), p. 1819-1837 (cf. p. 39).

[Zhang, 2010] Yin Zhang, Rong Jin et Zhi-Hua Zhou. « Understanding bag-of-words model : a statistical framework ». *International journal of machine learning and cybernetics* 1 (2010), p. 43-52 (cf. p. 22).

[Zhu, 2021] Yitan Zhu, Thomas Brettin, Fangfang Xia, Alexander Partin, Maulik Shukla, Hyunseung Yoo et al. « Converting tabular data into images for deep learning with convolutional neural networks ». *Scientific reports* 11.1 (2021), p. 1-11 (cf. p. 35).

[Zou, 2020a] Xiaohan Zou. « A Survey on Application of Knowledge Graph ». *Journal of Physics : Conference Series*. T. 1487. 1. 2020 (cf. p. 103).

[Zou, 2020b] Xiaohan Zou. « A survey on application of knowledge graph ». *Journal of Physics : Conference Series*. T. 1487. 1. IOP Publishing. 2020, p. 012016 (cf. p. 107).