

UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

AMÉLIORATION DES DONNÉES DE MAINTENANCE : INTÉGRATION DE
L'ANALYSE DES MODES DE DÉFAILLANCE ET DE LEURS EFFETS AU
TRAITEMENT DU LANGAGE NATUREL

THÈSE PRÉSENTÉE
COMME EXIGENCE PARTIELLE DU
DOCTORAT EN INGÉNIERIE
CONCENTRATION GÉNIE INDUSTRIEL

PAR
MATHIEU PAYETTE

MARS 2025

Université du Québec à Trois-Rivières

Service de la bibliothèque

Avertissement

L'auteur de ce mémoire, de cette thèse ou de cet essai a autorisé l'Université du Québec à Trois-Rivières à diffuser, à des fins non lucratives, une copie de son mémoire, de sa thèse ou de son essai.

Cette diffusion n'entraîne pas une renonciation de la part de l'auteur à ses droits de propriété intellectuelle, incluant le droit d'auteur, sur ce mémoire, cette thèse ou cet essai. Notamment, la reproduction ou la publication de la totalité ou d'une partie importante de ce mémoire, de cette thèse et de son essai requiert son autorisation.

UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

DOCTORAT EN INGÉNIERIE (Ph. D.)

Direction de recherche :

Georges Abdul-Nour, Ph. D, ing.	UQTR
Prénom et nom	Directeur de recherche

Jury d'évaluation

Professeur Georges Abdul-Nour, Ph. D, ing.	Directeur de recherche
Prénom et nom	Fonction du membre de jury

Professeur Souheil-Antoine Tahan, Ph. D	Membre externe (ETS)
Prénom et nom	Fonction du membre de jury

Mohamed Gaha, Ph. D	Membre externe (Hydro-Québec)
Prénom et nom	Fonction du membre de jury

Professeur François Gauthier, Ph. D, ing.	Président du jury (UQTR)
Prénom et nom	Fonction du membre de jury

Thèse soutenue le 9 janvier 2025

Résumé

Dans le domaine de la gestion des actifs, il est impératif d'assurer la qualité des données de maintenance, afin de garantir le bon fonctionnement du système. Cette réalité est d'autant plus vraie pour les grandes infrastructures, tels les réseaux de télécommunications ou bien les réseaux électriques. Plusieurs de ces organisations, qui dépendent fortement de leurs actifs, ont accumulé des données au fil des années sur l'entretien des équipements, afin d'orienter la prise de décision. Celles-ci offrent un apport d'informations qui mérite d'être valorisé, et les développements majeurs du traitement du langage naturel (TALN) apportent de nouvelles façons d'extraire des informations pertinentes. En effet, de nombreuses études critiquent la qualité des données extraites des bases de données, car elles sont saisies manuellement. Les champs de données structurés sont souvent accompagnés d'explications sous forme de textes courts et non structurés. L'application du TALN permettrait donc de compléter les champs structurés et d'augmenter potentiellement la qualité de ces derniers. Des études récentes traitent de ce sujet, illustrant la nécessité d'adapter les méthodes de TALN aux documents techniques et proposent le traitement automatique du langage technique (TALT) comme moyen de remédier aux limites du TALN. L'un des problèmes posés par l'adaptation du TALN aux textes techniques courts est la nécessité d'une annotation manuelle et d'un prétraitement, un processus souvent très fastidieux. Cette thèse propose une méthodologie complète visant à la correction des données de maintenance par la reclassification des ordres de travail à l'aide du traitement du langage naturel. La recherche est orientée sur l'adaptation des outils d'intelligence artificielle pour faire le traitement des textes techniques. La méthode développée tire profit de l'Analyse des Modes de Défaillance et de leurs Effets (AMDE), une technique d'analyse des risques, pour adapter les modèles préentraînés du TALN au domaine de la maintenance. À l'aide de l'AMDE, une première analyse

est effectuée pour normaliser et nettoyer les textes de maintenance, puis un modèle de reconnaissance d'entité, aussi basé sur l'AMDE, permet l'annotation du texte. Enfin, les textes annotés servent à entraîner un modèle de classification de texte, pour corriger les données de maintenance. La méthodologie développée est testée sur des ordres de maintenance du réseau de transmission d'Hydro-Québec. Les résultats démontrent que cette méthode a un impact significatif dans la précision des modèles de classification. Ce travail propose une solution innovante pour le traitement des données de maintenance et jette les bases d'applications TALN plus avancées dans le domaine de la gestion des actifs.

Abstract

In the field of asset management, it is imperative to ensure the quality of maintenance data, to guarantee the proper functioning of the system. This is particularly true for large infrastructures, such as telecommunications networks or power systems. Many of these organizations, which are heavily asset-dependent, have accumulated data over the years on equipment maintenance to guide decision-making. This data provides a rich source of information that deserves to be capitalized on, and major developments in natural language processing (NLP) are providing new ways of extracting relevant insights. Many studies have criticized data extracted from databases because it has been recorded manually. Structured data fields are often accompanied by explanations in the form of short, unstructured texts. The application of NLP would thus enable structured fields to be completed and potentially increase their quality. Recent studies on this subject illustrate the need to adapt NLP methods to technical documents and propose Technical Language Processing (TLP) as a means of overcoming the limitations of NLP. One of the problems in adapting NLP to short technical texts is the need for manual annotation and pre-processing, which is often a very tedious process.

This thesis proposes a complete methodology for correcting maintenance data by reclassifying work orders using natural language processing. The research focuses on the adaptation of artificial intelligence tools for processing technical texts. The method developed takes advantage of Failure Mode and Effect Analysis (FMEA), a risk analysis technique, to adapt pre-trained NLP models to the area of maintenance. Using FMEA, an initial analysis is carried out to normalize and clean up the maintenance texts, then an entity recognition model, also based on FMEA, is used to annotate the text. Finally, the resulting annotated texts are used to train a text classification model to correct the maintenance data. The methodology developed was tested on maintenance work

orders from Hydro-Québec's transmission network. The results show that this method has a significant impact on the accuracy of classification models. This work proposes an innovative solution for processing maintenance data and lays the foundations for more advanced NLP applications in the field of asset management.

Remerciements

Tout d'abord, je souhaiterais remercier mon directeur de recherche, le Professeur Émérite Georges Abdul-Nour, sans qui ce projet n'aurait pas été possible. Je te remercie profondément des nombreuses opportunités que tu m'as offertes, tout au long de la maîtrise et du doctorat. Tu as su me conseiller et tu m'as poussé à me dépasser tout au long des nombreux projets enrichissants. Je me sens particulièrement chanceux d'avoir pu travailler sous ta direction.

Je tiens aussi à remercier les chercheurs de l'Institut de recherche en électricité du Québec (IREQ), en particulier Alain Côté qui a été d'un grand soutien depuis le début de ma collaboration avec l'équipe de l'IREQ. J'aimerais aussi souligner le travail de tous les collaborateurs du projet PRIAD, qui ont rendu possible cette recherche, notamment Toulith Jean-Marc Meango qui m'a supervisé et épaulé dans l'avancement de ce projet et Miguel Diago Martinez, votre aide m'a été très précieuse.

Je remercie également mon père, Normand Payette, pour ses conseils et sa disponibilité tout au long de mon parcours. C'est à lui que je dois ma passion pour le génie industriel, passion qu'il m'a transmise depuis mon plus jeune âge. J'aimerais aussi souligner tous les encouragements et le soutien dont m'ont témoigné ma mère, Céline, et mes frères, Étienne et Olivier. Merci de me pousser à me dépasser, et surtout, merci d'avoir toujours été présent.

J'aimerais remercier ma femme, Marie-Michèle. Tu es à mes côtés depuis le début, avant même notre périple universitaire. Tu m'as tellement appris, toujours disponible pour me conseiller et m'épauler. Enfin, j'aimerais remercier ma petite fille Roseline, qui, à quelques mois seulement, me motive à avancer et à commencer le travail toujours plus tôt. Je vous aime, merci du fond du cœur !

Table des matières

Résumé	iii
Remerciements	vii
Table des matières	viii
Liste des tableaux	xiii
Liste des figures	xv
Liste des abréviations	xvii
Chapitre 1 - Introduction	1
1.1 Contexte opérationnel	2
1.2 Enjeux et problématique	4
1.3 Questions de recherche	5
1.4 Objectifs de recherche	6
1.4.1 Objectifs généraux	6
1.4.2 Objectifs secondaires	6
1.5 Vue d'ensemble de la méthodologie	7
1.6 Contribution scientifique de l'étude	8
1.7 Considérations et hypothèses	8
1.8 Structure de la thèse	9
Chapitre 2 - Revue de littérature	10
Chapitre 3 - Thèmes de recherche en ingénierie de fiabilité	14

3.1	Thèmes de recherche en ingénierie de la fiabilité	14
3.1.1	Hierarchie de la prise de décision organisationnelle	14
3.1.2	Ingénierie de la gestion des actifs	15
3.1.3	Défaillance, maintenance et fiabilité	17
3.2	Modélisation	20
3.2.1	Bases de la modélisation mathématique	20
3.2.2	Modèles statistiques	21
3.2.3	Modélisation qualitative	23
3.3	Fiabilité, disponibilité, maintenabilité et sécurité	24
3.3.1	Modélisation qualitative en fiabilité	24
3.3.2	Modélisation mathématique dans RAMS	26
3.4	Pronostic et gestion de la santé	33
3.4.1	Modélisation qualitative dans le domaine du PHM	35
3.4.2	Modélisation mathématique en PHM	35
3.5	Discussion	39
Chapitre 4 - Applications de l'apprentissage automatique pour l'ingénierie de la		
	fiabilité	42
4.1	Définition des concepts de l'IA	42
4.1.1	Apprentissage automatique, intelligence artificielle et science des données	42
4.1.2	Histoire de l'intelligence artificielle	43
4.1.3	Apprentissage supervisé	46
4.1.4	Apprentissage non supervisé	49
4.1.5	Apprentissage par renforcement	51
4.1.6	Apprentissage profond	52

4.1.7	Science des données	53
4.2	Processus de modélisation en apprentissage automatique	54
4.2.1	Acquisition de données	55
4.2.2	Nettoyage des données	56
4.2.3	Exploration des données	58
4.2.4	Ingénierie des fonctionnalités	59
4.2.5	Conception du modèle	60
4.3	Analyse des applications ML	61
4.3.1	Exécution et filtrage des résultats	61
4.3.2	Analyse des applications	61
4.3.3	Revue des méthodes d'apprentissage automatique	62
4.3.4	Jeux de données et systèmes à l'étude	68
4.3.5	Travaux connexes	69
4.3.6	Discussion	73
Chapitre 5 - Traitement automatique du langage naturel et applications à la fiabilité		74
5.1	Traitement automatique du langage naturel	74
5.2	Analyse des ordres de travail de maintenance	75
5.3	Principes du traitement du langage naturel	82
5.3.1	Annotation textuelle	82
5.3.2	Préparation du texte	84
5.3.3	Représentation des données textuelles	86
Chapitre 6 - Démarche de conception de la méthode		89
6.1	Préparation des données	89
6.1.1	Préparation des tables	89

6.2	Nettoyage des champs textes	91
6.2.1	Prétraitement des textes	91
6.2.2	Normalisation des termes	93
6.3	Annotations et extraction des connaissances	93
6.4	Classification des ordres de maintenance	94
Chapitre 7 - Méthode proposée pour la correction des données de maintenance		96
7.1	Normalisation du texte	96
7.2	Modélisation des entités	99
7.2.1	Aperçu général de la méthodologie	99
7.2.2	AMDE pour la classification des jetons	100
7.2.3	Création des données d'entraînements	102
7.3	Classification des ordres de maintenance	103
7.3.1	Mise en œuvre et validation	104
Chapitre 8 - Applications et résultats		108
8.1	Étude de cas	108
8.2	Création de dictionnaires de normalisation	110
8.3	Modèle de reconnaissance d'entités	115
8.4	Classification des ordres de maintenances	121
8.4.1	Comparaison des modèles avec et sans annotations	121
8.4.2	Comparaison des techniques d'apprentissage	124
8.4.3	Optimisation des hyperparamètres	126
Chapitre 9 - Discussion		134
9.1	Limites de l'étude	136
9.2	Recherches futures	136

Chapitre 10 -Conclusion	138
10.1 Contribution de la recherche	138
10.2 Travaux futurs	140
Bibliographie	154
Annexe A - Liste des publications	155
A.1 Articles de revue	155
A.2 Articles de conférence	156
Annexe B - Asset management, reliability and prognostics modeling techniques	157
Annexe C - Machine Learning Applications for Reliability Engineering: A Review .	184
Annexe D - Improving Maintenance Data Quality: Application of Natural Language Processing to Asset Management	223

Liste des tableaux

Tableau 1	Liste des abréviations anglaises et françaises	xvii
Tableau 2-1	Paramètre de la recherche bibliographique	10
Tableau 3-1	Fonctions de représentation des distributions de durée de vie	29
Tableau 3-2	Fonctions de représentation pour les systèmes réparables.	31
Tableau 3-3	Comparaison des fonctions d'intensité.	32
Tableau 4-1	Architectures et applications RNA.	54
Tableau 4-2	Résultats du filtrage.	62
Tableau 4-3	Algorithmes associés aux articles examinés.	65
Tableau 4-4	Bibliothèques Python pour l'ingénierie de la fiabilité.	72
Tableau 5-1	Bibliothèques Python pour les applications de traitement du langage naturel	83
Tableau 5-2	Méthodes de Word Embeddings	88
Tableau 6-1	Méthodes de prétraitement du texte	92
Tableau 7-1	Hyperparamètres des algorithmes utilisés	106
Tableau 8-1	Exemple de fichier AMDE	109
Tableau 8-2	Nombre de jetons et de mots au fil des étapes de traitement	111
Tableau 8-3	Les 10 principaux termes et leurs variantes	113
Tableau 8-4	Résultats de la moyenne micro et macro sur les données tests	118
Tableau 8-5	Résultats de l'entraînement pour chaque type d'entité	119
Tableau 8-6	Performance du modèle sans annotations	122

Tableau 8-7	Performance du modèle avec annotations	123
Tableau 8-8	Comparaison des mesures et des améliorations du modèle	124
Tableau 8-9	Paramètres des classificateurs utilisés	125
Tableau 8-10	Résultats comparatifs des méthodes de classifications	126
Tableau 8-11	Résultats des expérimentations en fonction du taux d'apprentissage et de la taille de lot	127
Tableau 8-12	Résultats des secondes expérimentations en fonction du taux d'apprentissage et de la taille de lot	130

Liste des figures

Figure 2.1	Représentation du nombre de publications par sujets	11
Figure 2.2	Structure de la revue de littérature	13
Figure 3.1	Hiérarchie de la prise de décisions.	15
Figure 3.2	Classification des activités de maintenance (adaptée de [1]).	19
Figure 3.3	Cycle de modélisation (adapté de [2]).	21
Figure 3.4	Techniques courantes de modélisation statistique.	23
Figure 3.5	Courbe baignoire (adaptée de [3]).	28
Figure 3.6	Types de systèmes, processus stochastiques et distributions [4].	30
Figure 3.7	Pronostic dans un processus de gestion de la santé [5].	35
Figure 3.8	Méthodes pronostiques (adaptées de [5,6]).	37
Figure 4.1	Développement de l'intelligence artificielle.	45
Figure 4.2	Exemple d'apprentissage supervisé.	47
Figure 4.3	Exemple d'échantillonnage de données pour l'apprentissage supervisé.	48
Figure 4.4	Exemple de courbe d'ajustement.	49
Figure 4.5	Approches et algorithmes d'apprentissage supervisé.	50
Figure 4.6	Algorithme d'apprentissage non supervisé.	51
Figure 4.7	Taxonomie des algorithmes d'apprentissage par renforcement.	52
Figure 4.8	Un modèle mathématique simple pour un neurone.	53
Figure 4.9	Cycle de vie de la science des données, adapté de [7].	55
Figure 4.10	Cercle vertueux pour la gestion de la qualité des données, adapté de [8].	57

Figure 4.11 Publications par an et par thème.	63
Figure 4.12 Fréquence d'utilisation des algorithmes de ML par thème.	64
Figure 4.13 Diagramme à secteurs des types de sources de données.	69
Figure 4.14 Types de systèmes étudiés.	70
Figure 5.1 Préparation du texte	85
Figure 6.1 Cadre méthodologique de la thèse	90
Figure 7.1 Processus de normalisation	98
Figure 7.2 Processus de modélisation	99
Figure 7.3 Extraction d'entités	101
Figure 7.4 Étapes de préparation des données d'entraînement	102
Figure 8.1 Exemples de normalisation du texte	115
Figure 8.2 Séparation des ensembles d'entraînement, de validation et de test ..	116
Figure 8.3 Courbes d'apprentissage du modèle NER	117
Figure 8.4 Exemple de résultat du modèle de reconnaissance d'entités	120
Figure 8.5 Courbe ROC pour les modèles RNA	129
Figure 8.6 Courbe ROC pour les taux d'apprentissage inférieurs à 0.001	131
Figure 8.7 Évolution du rappel du modèle de classification en fonction des cycles d'apprentissage	132
Figure 8.8 Évolution du score F1 du modèle de classification en fonction des cycles d'apprentissage	133

Liste des abréviations

Tableau 1 – Liste des abréviations anglaises et françaises

Abréviation (EN)	Signification	Abréviation (FR)	Signification
AI	Artificial Intelligence	IA	Intelligence Artificielle
AHP	Analytic Hierarchy Process	-	Analyse Hiérarchique des Processus
AM	Asset Management	GDA	Gestion des Actifs
ANOVA	Analysis of Variance	-	Analyse de la Variance
ANN	Artificial Neural Network	RNA	Réseau de Neurones Artificiel
API	Application Programming Interface	-	Interface de Programmation
BERT	Bidirectional Encoder Representations from Transformers	-	-
CBOW	Continuous Bag-of-Words	-	Sac de Mots Continu
CBR	Case-Based Reasoning	RàPC	Raisonnement à partir de Cas
CDF	Cumulative Density Function	FDC	Fonction de Densité Cumulative
CMMS	Computerized Maintenance Management Systems	-	-
CNN	Convolutional Neural Network	RNC	Réseau de Neurones Convolutionnel
CVC	Heating, Ventilation, and Air Conditioning	CVC	Systèmes de Chauffage, de Ventilation et de Climatisation
DB	Database	BD	Base de Données
DL	Deep Learning	-	Apprentissage Profond

Tableau 1 Liste des abréviations anglaises et françaises – (suite)

Abréviation (EN)	Signification	Abréviation (FR)	Signification
EAM	Engineering of Asset Management	GDA	Gestion des Actifs
EDA	Exploratory Data Analysis	AED	Analyse Exploratoire des Données
EPRI	Electric Power Research Institute	-	-
ELMO	Embeddings from Language Model	-	-
ERP	Enterprise Resource Planning	PGI	Progiciel de Gestion Intégré
FDP	Probability Density Function	FDP	Fonction de Densité de Probabilité
FDC	Cumulative Density Function	FDC	Fonction de Densité Cumulative
FMAIRCA	Fuzzy Multi-Attribute Ideal Real Comparative Analysis	-	-
FMECA	Failure Mode, Effect and Criticality Analysis	AMDEC	Analyse des Modes de Défaillance et de leur Criticité
FMEA	Failure Mode and Effect Analysis	AMDE	Analyse des Modes de Défaillance
FTA	Failure Tree Analysis	-	Analyse de l'Arbre des Défaillances
GPT	Generative Pre-Trained Transformer	-	Transformeur Génératif Pré-Entraîné
HAZOP	Hazard and Operability	-	-
HQ	Hydro-Québec	HQ	Hydro-Québec
HPP	Homogeneous Poisson Process	-	Processus de Poisson Homogène

Tableau 1 Liste des abréviations anglaises et françaises – (suite)

Abréviation (EN)	Signification	Abréviation (FR)	Signification
HQT	Hydro-Québec TransÉnergie et Équipement	HQT	Hydro-Québec TransÉnergie et Équipement
IBM	International Business Machines Corporation	-	-
ISO	International Organization for Standardization	-	-
k-NN	K-Nearest Neighbour	-	k-plus proches voisins
LSTM	Long Short-Term Memory	-	Mémoire à Court et Long Terme
ML	Machine Learning	-	Apprentissage Automatique
MTBF	Mean Time Between Failure	-	Temps Moyen Avant Défaillance
MTTF	Mean Time to Failure	-	Temps Moyen Entre Défaillance
NASA	National Aeronautics and Space Administration	-	-
NER	Named Entity Recognition	-	Reconnaissance d'Entités Nommés
NHPP	Non-Homogeneous Poisson Process	-	Processus de Poisson Non-Homogène
NIST	National Institute of Standards and Technology	-	-
NLP	Natural Language Processing	TAL	Traitement Automatique des Langues

Tableau 1 Liste des abréviations anglaises et françaises – (suite)

Abréviation (EN)	Signification	Abréviation (FR)	Signification
NLP	Natural Language Processing	TALN	Traitement Automatique du Langage Naturel
NLTK	Natural Language Toolkit	-	-
PHM	Prognostics and Health Management	-	Pronostique et Gestion de la Santé
PCA	Principal Component Analysis	ACP	Analyse en Composantes Principales
PDF	Probability Density Function	FDP	Fonction de Densité de Probabilité
PRIAD	-	PRIAD	Programme de Robustesse et d'Intégration d'Aide à la Décision pour la Gestion des Actifs
RAMS	Reliability, Availability, Maintainability, and Safety	-	Fiabilité, disponibilité, maintenabilité et sécurité
RCM	Reliability Centered Maintenance	MBF	Maintenance Basée sur la Fiabilité
RNN	Recurrent Neural Network	RNR	Réseau de Neurones Récurrent
ROCOF	Rate of Occurrence of Failures	-	Taux d'Occurrence des Défaillances
RUL	Remaining Useful Life	-	Durée de Vie Utile Restante
SL	Supervised Learning	-	Apprentissage Supervisé

Tableau 1 Liste des abréviations anglaises et françaises – (suite)

Abréviation (EN)	Signification	Abréviation (FR)	Signification
SVM	Support-Vector Machine	-	Machine à Vecteurs de Support
TLP	Technical Language Processing	TALT	Traitement Automatique du Langage Technique
TSDAE	Transformer-Based Sequential Denoising Auto Encoder	-	-
-	-	UQTR	Université du Québec à Trois-Rivières
UWA	University of Western Australia	-	Université d'Australie-Occidentale

Chapitre 1 - Introduction

L'ingénierie de la fiabilité est un domaine de recherche relativement mature. En effet, cette discipline scientifique se développe depuis les années 1950. Ce domaine, basé sur l'application des probabilités et des statistiques, est aujourd'hui omniprésent dans des secteurs tels que le militaire, la consommation, l'énergie, le nucléaire, etc. Au fil du temps, la discipline a beaucoup évolué et a influencé l'émergence de spécialités complémentaires, y compris des approches en gestion d'entreprise. En outre, les progrès technologiques ont aussi largement contribué à la formation de nouveaux sujets dans le domaine de la fiabilité. Ainsi, l'apprentissage automatique (ML) et l'intelligence artificielle attirent l'attention des communautés de chercheurs. De plus en plus de cas d'application émergent dans l'environnement manufacturier, en particulier avec l'avancement de la vision de l'industrie 4.0/5.0. La numérisation de l'environnement par la connectivité et les systèmes cyberphysiques conduit à la génération de mégadonnées, qui présentent plusieurs défis de traitement. On parle désormais de «rupture analytique» en raison du fait qu'en général, les organisations utilisent moins de 10% des données générées pour la modélisation et l'aide à la décision [9]. Les nouvelles techniques d'analyse ML ont fait l'objet de nombreuses publications, bien que les méthodes traditionnelles dans ce domaine soient encore largement appliquées. Les nouvelles techniques et technologies disponibles permettent le développement de nouvelles applications, mais aussi de nouveaux domaines, ce qui rend la sélection des méthodes appropriées de plus en plus complexe [9].

De plus, la modélisation par l'intelligence artificielle nécessite d'importantes quantités de données pour l'entraînement, une limitation expérimentée dans de nombreux cas industriels en raison de la rareté des données de haute qualité. Par conséquent, l'un des principaux défis de la recherche réside dans l'utilisation de ces données pour les initiatives de mise en œuvre de l'industrie 4.0. La plupart des entreprises industrielles disposent

de systèmes de gestion pour enregistrer la production, la maintenance, la comptabilité et d'autres informations opérationnelles. Celles-ci sont souvent saisies manuellement par un opérateur, et leur validité, leur exactitude, leur précision ou leur utilisabilité sont souvent discutables. Dans le domaine de la maintenance, comme dans de nombreux autres domaines, les données sont parfois complétées par des descriptions fournies dans des champs de texte libre. L'extraction de connaissances à partir de textes est l'un des objectifs du traitement automatique du langage naturel (TALN). En bref, le TALN, ou «*Natural Language Processing*» (NLP) est un domaine de l'IA qui se concentre sur l'analyse du langage humain par des machines. Les programmes informatiques perçoivent le texte comme une simple chaîne de caractères, sans signification réelle; ce sont les représentations numériques du langage qui permettent à l'ordinateur de saisir l'essence du texte [10]. Les techniques modernes de TALN tirent parti de l'apprentissage par transfert, en préentraînant les modèles sur de grands corpus de textes. Si les techniques de TALN semblent bien adaptées à l'analyse de données industrielles, la nature technique et la structure des champs textuels nécessitent une analyse plus poussée pour exploiter pleinement les modèles TALN de pointe. En d'autres mots, la plupart des textes de maintenance n'adhèrent pas aux normes de ce qui constitue le langage naturel et nécessitent une standardisation et une normalisation préalables [11]. Les travaux de cette thèse ont pour but de mettre en place une méthodologie basée sur l'intelligence artificielle pour corriger et arrimer les données de maintenance aux modèles de dégradation physique.

1.1 Contexte opérationnel

Il est important de décrire les parties prenantes de ce projet. En effet, il faut bien comprendre le lien entre le contexte de recherche ainsi que les besoins du partenaire industriel. Ce projet est issu d'un partenariat avec le département de génie industriel de

l'Université du Québec à Trois-Rivières (UQTR) et la société d'État Hydro-Québec (HQ), au moyen d'une chaire de recherche. De plus, cette thèse fait suite à un projet de maîtrise : Étude des facteurs affectant la fiabilité des transformateurs de puissance [12].

Hydro-Québec est une société d'État, détenue par le gouvernement du Québec, spécialisée dans la production, le transport et la distribution de l'électricité. Sa mission est de fournir un approvisionnement d'électricité fiable avec un service de qualité pour la province de Québec. De plus, l'entreprise figure parmi l'un des plus grands producteurs d'hydroélectricité au monde. L'entreprise possède son propre centre de recherche afin de mener des projets d'innovation. Le travail de cette thèse est issu d'un projet Centre de Recherche d'Hydro-Québec (CRHQ), le projet PRIAD, mandaté par le groupe TransÉnergie (HQT). Le Programme de Robustesse, d'Intégration et d'Aide à la Décision pour la gestion d'actifs d'Hydro-Québec TransÉnergie (PRIAD) est un projet visant l'amélioration des outils d'aide à la décision en GDA, par l'élaboration de modèles prédictifs, de simulation et d'optimisation de la maintenance. L'équipe du projet PRIAD, volet *Comportement de l'actif*, développe des outils permettant de prédire l'effet des stratégies de maintenance sur le taux de défaillance d'équipements réparables. Les besoins de l'entreprise sont de développer des modèles de fiabilité en tenant compte des contraintes opérationnelles individuelles des équipements, afin d'optimiser les politiques de maintenance. Afin de bien cerner la problématique, il est primordial de considérer les contraintes opérationnelles de l'entreprise. En plus d'exploiter un des plus importants réseaux de transport d'électricité, ce réseau se compose d'équipements divers et parfois âgés. En effet, l'entreprise a été fondée en 1944 et son parc d'équipements est très varié; certains datant d'avant la fondation sont toujours en service. En raison du vaste territoire couvert (1,7 million de kilomètres carrés), on compte plus de 4 types de climats différents : d'un climat continental humide à un climat polaire (Québec, 2019). La grande

majorité des équipements d'HQT sont à l'extérieur et sont donc grandement affectés par ces différences. Dus à l'âge des équipements, les historiques de maintenance sont parfois incomplets et les divers changements de systèmes et de codification rendent difficile de les retracer. Aussi, il existe plusieurs systèmes informatiques indépendants qui servent à différentes fonctions de l'entreprise, et dont les codifications diffèrent entre eux. Tous ces facteurs complexifient la tâche des responsables de la GDA.

1.2 Enjeux et problématique

Les besoins d'entreprise et les contraintes opérationnelles soulèvent plusieurs problématiques de recherche. Les équipements que possède HQT sont généralement réparables et nécessitent des politiques de maintenance adaptées. Toutefois, ces politiques sont directement influencées par la qualité des données de maintenance, sur lesquelles se basent les modèles fiabilistes. Les données historiques sur les entretiens et les défaillances, bien que cruciales, peuvent être incomplètes, mal structurées ou affectées par des erreurs de saisie, ce qui nuit à la précision des modèles de fiabilité.

En effet, les données de maintenance d'Hydro-Québec, historisées dans un système de base de données, sont composées de données structurées et de données non structurées, sous forme de champs de texte libre. Comme ce sont des données d'opérations qui sont entrées manuellement par des techniciens, il arrive qu'il y ait des erreurs de saisie dans les champs structurés, ce qui peut mener à de mauvaises estimations statistiques. Ce cas n'est pas unique à Hydro-Québec. En effet, dans de nombreuses organisations, la qualité des données de maintenance est souvent remise en question [13, 14] et leur validation repose principalement sur l'extraction de champs de texte décrivant les problèmes observés et les réparations effectuées. De plus, le niveau de granularité de l'information peut varier, compte tenu de l'évolution des systèmes informatiques au fil des années. En résumé, il y

a une inadéquation entre la quantité et la qualité des données, en plus de l'intégration des divers systèmes d'entreprise.

En conséquence, il est indispensable de corriger et d'améliorer la qualité des données de maintenance afin d'améliorer les politiques de maintenance de l'entreprise. D'un côté, il serait intéressant d'étudier et d'appliquer les approches préventives pour assurer la qualité des données. Or, il existe déjà une vaste littérature sur le sujet, et ce serait en dehors des limites du projet. Une autre méthode possible serait de corriger les données manuellement, mais la quantité de données historiques rend le processus fastidieux. En contrepartie, cette quantité importante de données offre la possibilité d'exploiter les techniques de l'intelligence artificielle. En effet, les techniques d'apprentissage automatique requièrent d'importantes quantités de données, rendant ce type d'approche particulièrement intéressant. Le traitement automatique du langage naturel (TALN), un ensemble de techniques issu de l'IA, offre une opportunité d'exploiter plus efficacement les données non structurées (texte libre) et d'améliorer la qualité des données.

1.3 Questions de recherche

De ces problématiques découlent plusieurs questions de recherche :

- Comment corriger et améliorer la qualité des données de maintenance ?
 - Peut-on améliorer la qualité des données historiques par la reclassification des types d'ordre de maintenance ?
 - Comment les techniques d'intelligence artificielle, notamment celles du traitement automatique du langage naturel, peuvent-elles être appliquées efficacement à cet effet ?

- Comment l'intégration de connaissances spécifiques au domaine électrique (dictionnaires, annotations, etc.) améliore-t-elle la performance des modèles de TALN ?
- Quel est l'impact de la correction des données de maintenance sur les modèles de fiabilité ?

1.4 Objectifs de recherche

1.4.1 Objectifs généraux

Pour répondre à ces questions, les travaux de cette thèse visent à développer une méthodologie basée sur l'intelligence artificielle pour corriger et arrimer les données de maintenance aux modèles de dégradation physique. Un des points forts d'Hydro-Québec est que l'entreprise historise des données de maintenance depuis plus de 50 ans. De plus, l'information contenue dans les champs non structurés est très détaillée; des champs de textes libres comportent beaucoup d'information qui pourrait être utilisée, mais qui n'est actuellement pas valorisée. Cette importante quantité de données offre l'opportunité d'appliquer et de tester des techniques issues de l'intelligence artificielle afin d'améliorer la classification des données de maintenance. L'hypothèse est que l'information à disposition dans ces champs textes contient suffisamment de renseignements pour augmenter le niveau de granularité et d'arrimer la base de données avec les modèles de risque développés par les experts. Ainsi, l'objectif premier de la thèse est la correction des données de maintenance par l'utilisation de méthodes d'intelligence artificielle.

1.4.2 Objectifs secondaires

Ce projet vise aussi à augmenter la précision des modèles déployés par l'équipe du projet PRIAD, par l'amélioration de la qualité des données historiques de maintenance.

Pour ce faire, les erreurs dans les données doivent être identifiées et corrigées. La littérature démontre que l'application des modèles d'IA sur des textes techniques est très difficile [15]. Certaines études suggèrent qu'il est nécessaire d'adapter les textes, en utilisant divers moyens, pour produire des résultats plus crédibles. L'annotation textuelle et la normalisation sont des techniques utilisées dans le traitement automatique des langues et offrent une solution intéressante à ce problème. Ainsi, pour produire un modèle de correction des données dont le type de maintenance effectué est mal étiqueté, il apparaît nécessaire de développer une méthodologie pour annoter et améliorer la qualité des textes de maintenance. Tout d'abord, la normalisation des textes permettra de réduire la variabilité dans le vocabulaire à analyser. Ensuite, un modèle permettra d'identifier les composants affectés, ainsi que les mécanismes de dégradation en cause. Une fois la qualité des textes augmentée, un modèle de correction des données de maintenance sera appliqué.

1.5 Vue d'ensemble de la méthodologie

L'objectif du travail est de développer une approche fondée sur l'intelligence artificielle pour corriger les données de maintenance. La démarche passe par la reclassification d'ordres de maintenance, tout en produisant des annotations textuelles pour arrimer les textes de maintenance à l'analyse des modes de défaillance et de leurs effets (AMDE). Les descriptions de travaux de maintenance qui sont répertoriées ne répondent généralement pas aux règles syntaxiques usuelles. Le processus de modélisation requiert donc d'importants efforts pour une standardisation et une normalisation préalables des textes. De plus, une procédure est développée pour adapter les modèles de Reconnaissance d'Entités Nommées (NER), afin de reconnaître et de classer les composants et les modes de défaillance dans les textes techniques. Ce travail peut améliorer de manière significative l'automatisation et la précision des processus d'extraction d'informations

qui sont essentiels pour garantir la fiabilité et la sécurité des opérations industrielles. Le modèle NER et les textes normalisés sont ensuite utilisés dans un processus de reclassification des ordres de maintenance. Ce nouveau processus permettra d'augmenter la qualité des données de maintenance, et, par le fait même, améliorer la précision des estimations statistiques de la fiabilité des équipements.

1.6 Contribution scientifique de l'étude

Il existe à ce jour peu de littérature qui étudie l'adaptation du TALN à des données d'opérations, dans un contexte industriel réel. Il existe plusieurs articles qui se concentrent sur la construction d'AMDE ou de structures similaires à partir de la documentation de maintenance en utilisant les outils du TALN [16, 17]. À notre connaissance, il n'existe pas de littérature qui, à l'inverse, combine les AMDE existants et le TALN pour analyser des textes techniques, tels que les ordres de travail de maintenance. Ainsi, les méthodes développées pour la normalisation des textes et l'extraction d'entités représentent une contribution scientifique notable à la littérature. De plus, cette étude se distingue aussi par son contexte d'application; d'une part, par l'analyse de données de maintenance d'équipements de transmission d'électricité, d'autre part, puisque les textes analysés sont en français.

1.7 Considérations et hypothèses

Selon les experts de l'entreprise, la qualité des données contenues dans le système informatique est parfois discutable. Les causes suspectées sont liées aux erreurs humaines, ainsi qu'aux différents changements de système de saisie au fil du temps. De plus, les travaux de recherche faits précédemment ont soulevé des problématiques, notamment dans l'identification du type de maintenance effectuée [12]. Cette thèse vise à proposer

une méthodologie pour l'identification et la correction des différentes erreurs de saisie, dans le but d'améliorer la qualité des données qui seront exploitées dans des études de fiabilité. Cette méthodologie sera appliquée aux données d'Hydro-Québec TransÉnergie, à titre d'étude de cas. Pour ce faire, l'étude de cas touchera un seul type d'équipement de transport d'énergie, les transformateurs de puissance.

1.8 Structure de la thèse

Le chapitre 2 propose un portrait préliminaire de la littérature scientifique, focalisant sur les applications du TALN à la fiabilité des équipements de transport électrique. Ensuite, le chapitre 3 établit les bases du domaine, par une revue de la littérature générale, visant la définition et la distinction des concepts de fiabilité et maintenance, de gestion des actifs et de pronostic. Puis, la revue fait un survol de l'application de l'apprentissage automatique dans le domaine de la fiabilité dans le chapitre 4. Enfin, elle présente l'état de l'art, en ce qui a trait à l'application du traitement automatique du langage naturel en fiabilité et maintenance, au chapitre 5. Le chapitre 6 expose les différentes étapes de la méthodologie globale qui ont mené à l'élaboration du processus décrit au chapitre 7. Le chapitre 8 sert à présenter les résultats détaillés, en plus de détailler l'étude de cas, qui sert de preuve de concept à la méthode proposée. La discussion de ces résultats pour chaque étape de modélisation se fait au chapitre 9 et le chapitre 10 servira à présenter les conclusions et les leçons tirées tout au long de ce travail. Les chapitres 4 et 5 sont adaptés d'articles scientifiques publiés dans des revues internationales reconnues et ont déjà plusieurs citations [18, 19]. Enfin, les chapitres 6, 8 et 9 sont inspirés d'articles scientifiques qui ont été soumis et acceptés à la conférence «*51st International Conference on Computers and Industrial Engineering (CIE51)*» et «*71st Annual Reliability and Maintainability Symposium*».

Chapitre 2 - Revue de littérature

Ce chapitre a pour but de dresser un portrait de la littérature, afin d'identifier les opportunités de contributions à la recherche. Pour ce faire, une recherche méticuleuse par mot clé a été effectuée en fonction des paramètres suivants :

Tableau 2-1 Paramètre de la recherche bibliographique

Ressource Bibliographique	Base de Données	Date de Début	Date de Fin
EBSCOhost	Academic Search Complete	2013	2024
	Applied Science & Technology Source		
	Computers & Applied Sciences Complete		
	MLA Directory of Periodicals		
	MLA International Bibliography		

Les mots clés suivants ont été utilisés afin de cibler les articles pertinents au projet :

- Termes liés au traitement automatique des langues :
 - «*Natural Language Processing*» OR/OU
 - «*NLP*» OR/OU
 - «*Technical Language Processing*» NOT/PAS
 - «*nonlinear programming*» AND/ET
- Termes liés à la fiabilité :
 - «*maintenance*» OR/OU
 - «*reliability*» AND/ET
- Termes liés au service d'électricité :
 - «*power system*» OR/OU
 - «*power systems*» OR/OU
 - «*energy systems*»

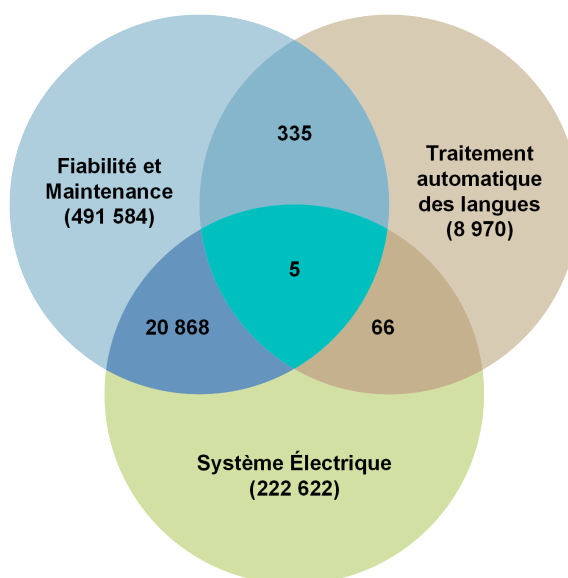


Figure 2.1 Représentation du nombre de publications par sujets

De cette recherche, seulement cinq résultats d'articles sont ressortis, dont deux qui ne correspondent pas au sujet du traitement automatique des langues. La figure 2.1 met en évidence le nombre de publications par sujets, pour la période observée de 2013 à 2024. Dans les publications retenues, une étude propose d'utiliser l'analyse des modes de défaillance, de leurs effets et de leur criticité, combinés au TALN. L'auteur démontre l'applicabilité de la méthode par une étude de cas sur les données de maintenance d'un transformateur de puissance d'une centrale hydroélectrique. À l'aide de l'outil Power BI, la démarche permet d'effectuer des analyses statistiques appuyées par des visualisations des données. Or, en analysant la méthodologie présentée, l'auteur propose en fait une méthode de recherche par mot clé, afin d'identifier les textes qui contiennent des termes liés aux AMDEC. L'algorithme est en fait une requête SQL pour sélectionner les composants ou les modes de défaillance, et compter le nombre de résultats retournés pour faire une analyse de fréquence [20]. Un autre article présente une application TALN, par l'analyse de rapports de maintenance d'un distributeur électrique. L'article tire profit des données publiques d'EPRI, le «*Electric Power Research Institute*», pour la construction

du modèle. La première étape consiste à créer des vecteurs de textes en utilisant la méthode d'allocation de Dirichlet latente. Cette technique de «*topic modeling*» permet de créer une représentation des phrases et de les associer à un sujet. Cette représentation vectorielle est ensuite utilisée pour entraîner un modèle de classification. L'auteur teste plusieurs méthodes populaires de classification, mais c'est la méthode du plus proche voisin qui obtient le meilleur score [21]. Enfin, la dernière étude porte sur les textes de maintenance d'un parc éolien. Trois méthodes d'étiquetage de texte sont utilisées afin de calculer le temps moyen entre les défaillances (MTBF). La première méthode consiste à classer manuellement les données, à l'aide d'experts en maintenance. Au total, cela représente plus de 230 heures pour classer un peu moins de 4 000 textes. Le MTBF obtenu par ces données offre la véritable mesure pour comparer les deux autres méthodes. La seconde méthode consiste à utiliser les données étiquetées par les experts pour élaborer deux modèles de classification, soit avec la régression logistique et la classification naïve de Bayes. Enfin, ces méthodes sont comparées avec le logiciel NESTOR, un outil d'étiquetage semi-automatique. Les résultats montrent que les modèles de classification arrivent à une estimation plus près du MTBF, comparativement aux résultats avec NESTOR. Cependant, l'étiquetage des données avec NESTOR est beaucoup plus rapide, avec un temps de 28 heures, une réduction de 88 %. Qui plus est, les modèles de classifications ont bénéficié d'un jeu de données rééchantillonnées, ce qui n'est pas le cas avec le logiciel. Cela peut donc expliquer une partie de l'écart observé avec la méthode d'étiquetage semi-automatique [22].

Comme le démontre cette revue de littérature préliminaire, il existe de plus en plus de littérature sur le traitement automatique du langage naturel appliqué à la maintenance. Par contre, le nombre de publications appliquées spécifiquement à la maintenance d'actifs de systèmes électriques reste limité. Enfin, aucune des études appliquées aux réseaux

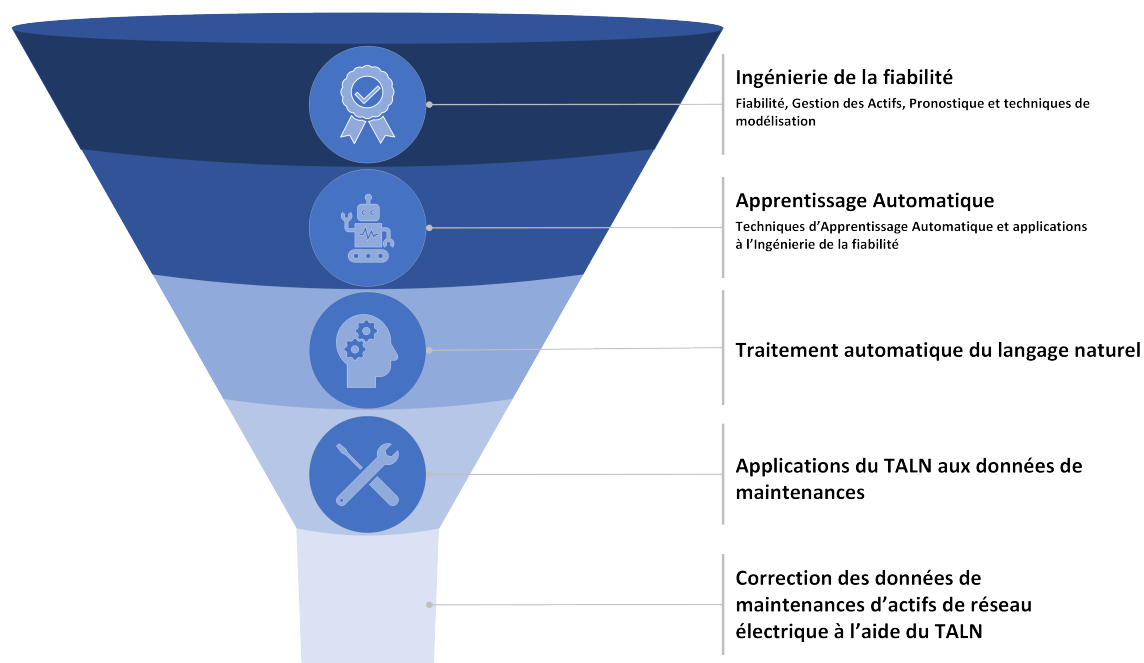


Figure 2.2 Structure de la revue de littérature

électriques ne concerne la correction des ordres de maintenance, soulignant le caractère unique de la présente recherche. Cependant, il est évident que pour dresser un portrait complet de la littérature, il est indispensable d'élargir la recherche documentaire. La figure 2.2 présente la structure de la revue de littérature des chapitres suivants. Tout d'abord, une revue générale de chacun des thèmes est faite, en débutant par l'ingénierie de la fiabilité, qui aborde les principes de la maintenance, de la gestion des actifs et du pronostic. Ensuite, le thème de l'apprentissage automatique est examiné afin de comprendre les bases des techniques du TALN, en plus de faire un survol des applications de l'IA au domaine de la fiabilité. Ces deux chapitres sont tirés d'une publication dans la revue «*Sustainability*» [18, 19]. Enfin, le chapitre 5 décrit le traitement automatique du langage naturel, ainsi que les techniques typiquement utilisées, puis un survol des publications d'applications dans le domaine de la maintenance démontre la pertinence de l'étude. Ces trois chapitres font un état global des connaissances et présentent les bases sur lesquelles s'appuie ce travail.

Chapitre 3 - Thèmes de recherche en ingénierie de fiabilité

Ce chapitre propose une revue des concepts de l'ingénierie de la fiabilité, dans le but d'identifier les différentes communautés scientifiques, ce qui les différencie et ce qui les relie. Il aborde également les techniques de modélisation RAMS et PHM et souligne l'importance de ces disciplines pour assurer le fonctionnement des systèmes complexes. En bref, il vise à clarifier l'interrelation entre les sujets de l'ingénierie de la fiabilité, afin de simplifier la recherche et la sélection des méthodes de modélisation. En outre, ce chapitre provient d'un article portant sur le sujet, publié en 2023 [18]

3.1 Thèmes de recherche en ingénierie de la fiabilité

3.1.1 Hiérarchie de la prise de décision organisationnelle

Toute entreprise doit avoir une mission, des objectifs et un plan d'action pour atteindre ses buts. Cette section présente un modèle hiérarchique de prise de décision dans une organisation. À ce titre, il est possible de mettre en perspective ce modèle avec la gestion des actifs, la RAMS et le PHM. La figure 3.1, traduite de [23], illustre la hiérarchie entre la mission et les stratégies commerciales. Selon leur définition, la déclaration de mission est un guide pour la formulation des stratégies d'une entreprise. L'énoncé de la mission vise à décrire la raison d'être de l'entreprise. Les objectifs de l'entreprise sont définis en fonction de la mission et donnent une orientation claire à l'organisation. Les stratégies organisationnelles sont des plans visant à atteindre les objectifs de l'entreprise. Il s'agit de stratégies globales qui guident l'ensemble de l'organisation. Les stratégies fonctionnelles, qui sont liées aux cinq fonctions de l'entreprise (finances, marketing, ressources humaines, opérations et production), soutiennent les stratégies organisationnelles; les stratégies tactiques regroupent les méthodes et les plans d'action permettant de les réaliser [23]. La hiérarchie de ces stratégies est liée à l'horizon temporel dans lequel elles sont déployées,

ainsi qu'à leur portée et à leur niveau de détail. En effet, la mission est établie à un niveau hiérarchique élevé et sert de guide à long terme pour une organisation. Sa portée est très large, mais son niveau de détail est faible. En revanche, les stratégies tactiques ont un niveau hiérarchique très bas et un horizon temporel court, mais leur niveau de détail est élevé.



Figure 3.1 Hiérarchie de la prise de décisions.

3.1.2 Ingénierie de la gestion des actifs

Il est important de définir la gestion des actifs, qui représente un niveau supérieur de prise de décision. Par définition, un actif est un objet ou une entité à laquelle une valeur est attribuée. Un actif technique a une valeur réelle pour une organisation; il peut s'agir d'un équipement, d'un stock, d'un bâtiment, etc. [24, 25]. L'ingénierie de la gestion des

actifs (GDA) est le domaine qui englobe toutes les activités organisationnelles visant à réaliser la valeur d'un actif. Il ne faut pas la confondre avec la gestion des actifs financiers, qui concerne les investissements et les actifs bancaires et non les actifs physiques. Le système de gestion des actifs permet de coordonner les activités dans le but d'atteindre les objectifs de la GDA. Les objectifs du système GDA doivent s'aligner sur les objectifs stratégiques de l'entreprise [26]. Cela implique d'équilibrer les coûts, les risques, les opportunités et les avantages de la performance [25]. La norme ISO établit également les principes fondamentaux de la gestion des actifs. Un concept central de la GDA est la valeur et la manière dont l'actif la génère. Selon l'ISO, cette valeur doit être définie par l'organisation et ses parties prenantes, sur la base des objectifs de l'entreprise. En outre, la valeur varie d'une organisation à l'autre et dépend principalement du contexte et des besoins de l'entreprise [25]. La notion d'alignement est donc importante; la GDA doit être alignée sur les objectifs de l'entreprise, sur la base de cette notion de valeur. Les objectifs doivent être traduits en politiques et stratégies de gestion [27]. L'impact de la gestion des actifs ne se limite pas à l'aspect financier. Elle joue un rôle important dans le développement durable; la vision à long terme de l'optimisation du cycle de vie des actifs réduit les déchets en optant pour une conception plus robuste et plus durable. En outre, l'optimisation des remplacements réduit la maintenance inutile et le remplacement de composants encore en bon état, réduisant ainsi l'empreinte carbone des actifs tout au long de leur durée de vie. Dans la perspective de la section précédente, l'ingénierie de la gestion des actifs est établie au niveau de la stratégie organisationnelle, étant donné que son champ d'application s'étend à toutes les fonctions de l'entreprise. Des exemples sont donnés dans [28], où l'auteur examine les concepts de gestion des actifs appliqués aux grands réseaux électriques. Il présente divers cas d'application à différents niveaux de gestion de l'entreprise pour mettre en évidence les avantages d'une approche intégrée de

la gestion des actifs. Dans [29], il étend les concepts de gestion des actifs à l'utilisation de la connectivité et des mégadonnées. En outre, il aborde les avancées en matière de surveillance, de technologies des capteurs et de science des données, ainsi que la manière dont elles seront appliquées à l'appui de la gestion des actifs. En outre, dans [30], l'auteur propose un examen des concepts de l'industrie 4.0 en relation avec la gestion des actifs. L'étude se concentre sur les applications pour l'industrie électrique.

3.1.3 Défaillance, maintenance et fiabilité

Les prochaines sous-sections servent à définir les différents concepts de base liant les défaillances, la maintenance et la fiabilité.

3.1.3.1 Défaillance

Du point de vue de la GDA, une défaillance signifie qu'un actif perd ses fonctions de manière temporaire ou permanente et ne génère plus de valeur pour son organisation. La défaillance est un événement inévitable pour les produits et les systèmes [31]. La défaillance d'un système peut être causée par des facteurs internes tels que l'usure, la corrosion, etc., ou par un facteur externe, par exemple la chute d'un arbre sur la ligne de transport d'électricité. L'étude des mécanismes de dégradation et des modes de défaillance est donc essentielle pour optimiser le cycle de vie des actifs. Savoir quand et pourquoi un système perd ses fonctions permet aux gestionnaires de mettre en œuvre des solutions appropriées. Afin de maintenir ses actifs, une organisation peut employer une stratégie de maintenance différente, comme décrite dans la section suivante.

3.1.3.2 Maintenance

Une défaillance est un événement qui empêche le système de remplir ses fonctions prévues [4]. La maintenance est la combinaison des moyens utilisés pour s'assurer que

le système puisse continuer à remplir ces fonctions. Elle comprend toutes les activités liées à la planification, à la gestion et à l'exécution de la maintenance des équipements afin d'atteindre ces objectifs opérationnels. La maintenance préventive vise à maintenir le système en état de fonctionnement, la réparation étant effectuée avant qu'une défaillance ne se produise. En cas de défaillance, la maintenance corrective permet de remettre le système en état de fonctionnement. Au cours du cycle de vie d'un bien, plusieurs facteurs détériorent l'état des composants. Ces mécanismes de dégradation peuvent être liés à l'utilisation ou à des phénomènes physiques et environnementaux. Lorsque la dégradation est cyclique, la maintenance préventive systématique permet de stopper ces mécanismes de dégradation avant qu'il n'y ait une défaillance. La maintenance systématique consiste à effectuer des opérations de maintenance à intervalles fixes, la périodicité pouvant dépendre du nombre de cycles (par exemple, le nombre de kilomètres parcourus) ou du nombre d'heures de service. Dans ce cas, les modes de défaillance sont connus, prévisibles et faciles à planifier. Dans certains cas, la dégradation des composants n'est pas cyclique, mais des signes de vieillissement peuvent être observés ou mesurés lors de l'inspection. On parle alors de maintenance conditionnelle. Elle consiste à réparer le système lorsqu'une inspection ou une surveillance révèle des signes de vieillissement, indiquant la présence de mécanismes de défaillance. Généralement, la maintenance conditionnelle intervient lorsque les modes de défaillance sont aléatoires. La maintenance prédictive intervient également lorsque des signes d'usure apparaissent. Dans ce cas, les modes de défaillance sont connus et l'occurrence peut être modélisée à l'aide de données historiques et de données de capteurs en temps réel. La maintenance corrective correspond à une réparation lorsqu'une défaillance se produit. La plupart du temps, elle concerne des événements non désirés et problématiques. Cependant, dans un programme de maintenance basée sur la fiabilité (MBF), une organisation peut choisir d'opter pour une telle stratégie pour certains

systèmes redondants ou peu coûteux à remplacer [1]. La figure 3.2, adaptée de [1], montre les différents types de maintenance et leurs caractéristiques, comme décrit ci-dessus.

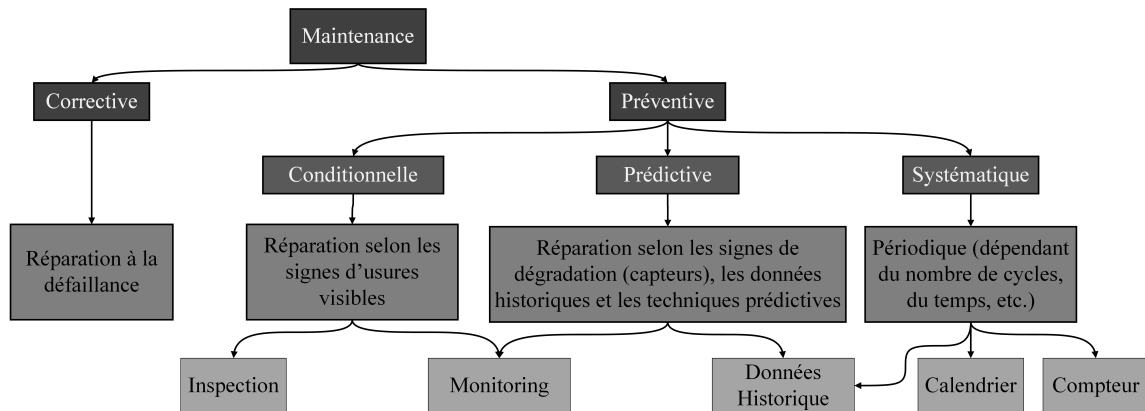


Figure 3.2 Classification des activités de maintenance (adaptée de [1]).

3.1.3.3 Statistiques des défaillances

Une défaillance est considérée comme un événement aléatoire. En maintenance, la fiabilité est la capacité du bien à effectuer les actions prévues dans un intervalle de temps donné. Ainsi, la fiabilité peut être exprimée comme la probabilité que le système fonctionne correctement pendant cet intervalle de temps et dans des conditions prédéfinies [32,33]. L'objectif de l'étude de fiabilité est de prévenir et/ou de réduire la fréquence de ces défaillances en appliquant des techniques d'ingénierie et des techniques statistiques pour identifier et quantifier les sources de défaillance. La fiabilité est souvent exprimée sous la forme d'un modèle mathématique ou d'un descripteur statistique, tel que le temps moyen avant défaillance (MTTF) ou le temps moyen entre défaillances (MTBF). Deux branches de la recherche en ingénierie s'intéressent à l'étude de la gestion de la santé des systèmes : la fiabilité, la disponibilité, la maintenabilité et la sécurité (RAMS) et le pronostic et la gestion de la santé (PHM). Dans la littérature, ces disciplines sont souvent difficiles à distinguer l'une de l'autre. Les deux disciplines cherchent à optimiser la valeur dérivée

des actifs tout au long de leur cycle de vie. Cependant, les modèles et les outils utilisés par chaque approche sont très différents. Selon Pierre Dersin, la RAMS se concentre sur l'étude des propriétés générales d'une population, tandis que le PHM se concentre sur un seul actif à la fois [34].

3.2 Modélisation

Cette section décrit les différentes approches de modélisation utilisées dans le cadre de la GDA, de la RAMS et du PHM. Tout d'abord, les bases de la modélisation mathématique sont présentées. Ensuite, les approches fondamentales de modélisation statistique sont comparées. Enfin, la modélisation qualitative est examinée.

3.2.1 Bases de la modélisation mathématique

La modélisation, d'une manière très générale, est utilisée pour représenter un objet ou une situation, la simplifier, la comprendre et l'analyser. La modélisation mathématique consiste à représenter une situation du monde réel à l'aide d'équations mathématiques. Le cycle de modélisation mathématique est présenté dans la Figure 3.3, et est adapté de [2]. Cela représente bien la tâche de modélisation, ainsi que la boucle de validation; les hypothèses de recherche permettent de simplifier une situation existante et de la décrire à l'aide des mathématiques. La validation entre les résultats du modèle et la situation réelle permet de savoir si le modèle, bien que simplifié, représente bien la situation en question. Ce processus de modélisation est répété jusqu'à ce que les résultats produits soient satisfaisants [35].

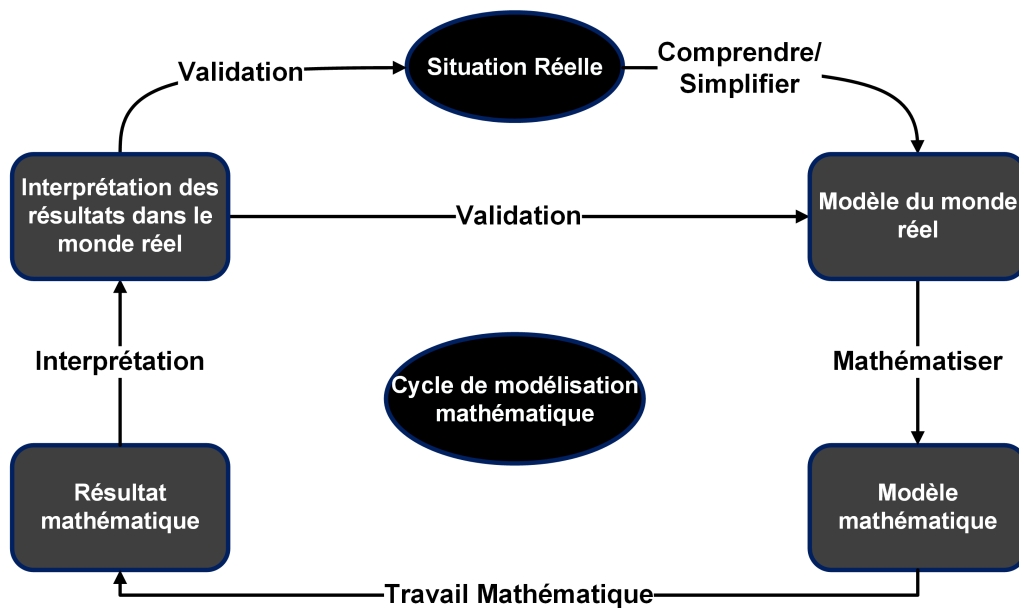


Figure 3.3 Cycle de modélisation (adapté de [2]).

3.2.2 Modèles statistiques

En matière de modélisation statistique, on peut distinguer deux types d'approches. Les statistiques descriptives visent à décrire et à résumer les observations d'un échantillon à l'aide d'indicateurs, de graphiques, etc. [36]. Les statistiques inférentielles, quant à elles, visent à déduire les caractéristiques d'une population à partir d'un échantillon [37]. La figure 3.4 présente les approches les plus courantes des statistiques descriptives et inférentielles. En statistique inférentielle, les distributions de probabilité sont utilisées pour décrire les variables aléatoires d'un échantillon et en extraire les caractéristiques. En connaissant le type d'expérience aléatoire, il est généralement facile d'associer la distribution de probabilité à l'échantillon et de déterminer les paramètres de cette distribution à partir de ce dernier. Ce processus est souvent connu sous le nom d'analyse paramétrique. Dans certains cas, il est facile de définir la distribution des données, compte tenu du contexte opérationnel et aléatoire des phénomènes étudiés [38].

En revanche, l'analyse descriptive, également connue sous le nom d'analyse non paramétrique, est utilisée pour déduire les caractéristiques d'un échantillon sans utiliser de distribution statistique. Les mesures de tendance centrale et de dispersion sont généralement employées pour décrire les paramètres d'une population étudiée. La construction d'histogrammes, de diagrammes de dispersion et de diagrammes en boîte est largement répandue pour étudier le comportement des systèmes en termes de fiabilité. Les tableaux de fréquence peuvent également être utilisés pour estimer la fonction de densité de probabilité, sans qu'une loi de distribution spécifique ne soit nécessaire [38]. Ces méthodes statistiques sont mises en œuvre dans de nombreux domaines et ne se limitent pas à la fiabilité. Par exemple, [39] présente une simulation d'une usine de fabrication pour tester la mise en œuvre de la technologie de l'industrie 4.0. À cette fin, les tests d'hypothèse et l'analyse de la variance (ANOVA) sont respectivement appliqués à des fins de validation et de sélection des paramètres. Dans [40], ils utilisent des métriques statistiques, en particulier la matrice de confusion, pour évaluer les résultats d'un modèle de classification de texte. Dans [41], ils ont utilisé des tests d'hypothèse (tests du chi-carré de Pearson) pour valider le choix et l'estimation des paramètres d'une distribution de Weibull. L'étude consiste en un banc d'essai pour étudier la fiabilité des composants électroniques afin de déterminer l'effet du rayonnement gamma sur la tension de claquage des parafoudres remplis de gaz.

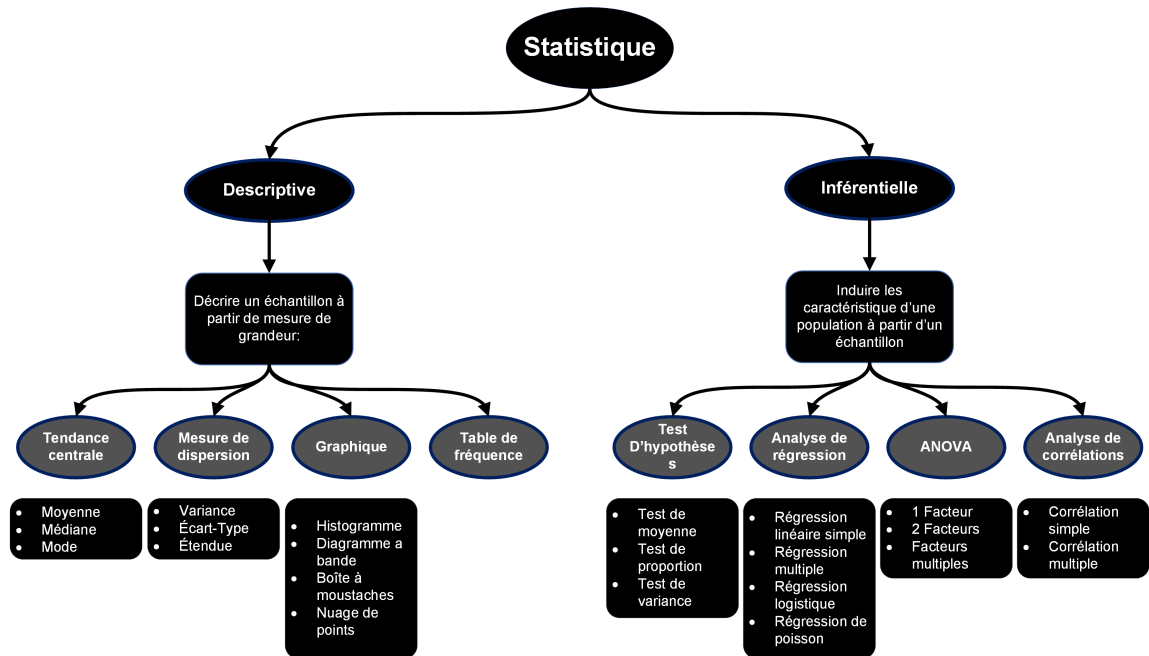


Figure 3.4 Techniques courantes de modélisation statistique.

3.2.3 Modélisation qualitative

Les méthodes de modélisation présentées précédemment sont des méthodes basées sur des données. Toutefois, dans le domaine de l'ingénierie, il existe également des méthodes basées sur des modèles qualitatifs. Cette brève section servira à décrire ce qu'est un modèle qualitatif et ce qui le distingue des modèles basés sur des données. Les modèles qualitatifs visent à représenter les systèmes et les raisonnements de manière symbolique. Le système et ses contraintes sont représentés de manière discrète, plutôt qu'à l'aide de données et d'équations mathématiques. L'objectif est de déduire autant de comportements que possible à partir de peu d'informations. Les modèles qualitatifs sont basés sur l'expérience du milieu et l'expertise professionnelle. Ces expériences sont utilisées pour décider de ce qui doit être inclus en fonction de la situation, des phénomènes physiques importants et des simplifications pertinentes [42].

3.3 Fiabilité, disponibilité, maintenabilité et sécurité

La RAMS est le domaine de l'ingénierie qui traite de l'étude de la fiabilité d'un ensemble de systèmes et/ou d'équipements [34]. L'objectif est de déterminer le comportement général d'une population afin de prendre des décisions sur les politiques de maintenance, les remplacements, etc. Pour ce faire, les ingénieurs utilisent différentes méthodes de modélisation statistiques et qualitatives. La première étape de l'établissement d'une politique de maintenance pour un système consiste à identifier le risque associé à son fonctionnement [31]. À cette fin, la section suivante présentera les méthodes qualitatives utilisées pour identifier ces risques, puis les méthodes permettant de quantifier ces risques.

3.3.1 Modélisation qualitative en fiabilité

Lors de l'analyse de la fiabilité d'un bien, il est essentiel d'identifier les risques associés à son fonctionnement. La section 3.3.1 est utilisée pour présenter deux des méthodes qualitatives les plus populaires pour développer de tels modèles. Ces méthodes s'appuient fortement sur l'éllicitation d'experts, généralement ceux qui ont conçu, exploité ou entretenu le système.

3.3.1.1 Analyse des modes de défaillance et de leurs effets

L'analyse des modes de défaillance et de leurs effets (AMDE) est une méthode d'analyse permettant d'identifier les modes de défaillance des composants. Le mode de défaillance est une cause observable de dysfonctionnement du système. L'AMDE consiste à décomposer le système en sous-systèmes fonctionnellement indépendants. Dans un premier temps, l'analyste doit identifier les différents modes de fonctionnement et les configurations du système dans ces modes de fonctionnement. Ensuite, il compile ces informations dans un tableau (par mode de fonctionnement) et indique les modes de

défaillance associés et l'effet qu'ils ont sur d'autres entités du système [31]. En outre, cette méthode simple permet d'examiner les défaillances potentielles du système et de déterminer les mesures préventives à prendre pour éviter les problèmes identifiés [1]. À titre d'exemple, les auteurs de [43] ont développé une nouvelle méthode d'analyse multicritère, afin de classer les modes de défaillance en fonction du niveau de risque et des actions d'atténuation et d'élimination des risques. La nouvelle méthode combine le processus de hiérarchie analytique floue (Fuzzy AHP) et la «*Fuzzy Multi-Attribute Ideal Real Comparative Analysis*» (FMAIRCA). De même, la référence [44] propose une méthode basée sur la théorie de Dempster–Schaffer pour améliorer la méthode traditionnelle de hiérarchisation des risques de l'AMDE. Une étude de cas est présentée pour l'analyse d'un terminal de gaz naturel liquéfié. L'AMDE est souvent utilisée comme point de départ pour la construction d'arbres de défaillance, qui seront examinés dans la section suivante, et pour les programmes de maintenance basés sur la fiabilité [31].

3.3.1.2 Analyse de l'arbre des défaillances

Parfois, aucune donnée n'est collectée pour effectuer une analyse statistique de la défaillance d'un système. Dans ce cas, il peut être utile d'examiner les causes sous-jacentes et la combinaison d'événements qui peuvent conduire à une défaillance particulière. L'une des méthodes utilisées est l'analyse de l'arbre des défaillances (FTA), où la logique booléenne est appliquée pour relier les événements par des relations logiques simples (liens de cause à effet). La méthode commence par l'identification d'un mode de défaillance (un à la fois) et la détermination de tous les événements élémentaires qui y sont liés. Comme cette méthode qualitative exploite l'algèbre de Boole, l'arbre peut être décrit par un ensemble d'équations et de variables aléatoires. En utilisant les règles de l'algèbre de Boole, il est possible de calculer les probabilités de défaillance du système.

Il existe de nombreuses autres méthodes qualitatives pour modéliser les défaillances d'un système. Le diagramme des causes et des effets (Ishikawa), les listes de contrôle, le réseau bayésien, l'analyse de l'arbre des événements, l'analyse des risques et de l'exploitabilité (HAZOP) et la méthode de l'indice de risque sont des exemples de méthodes couramment utilisées en matière de fiabilité [4, 31].

3.3.2 Modélisation mathématique dans RAMS

La fiabilité peut être mesurée à l'aide de la fonction de survie ($S(t)$ ou $R(t)$), de la fonction de densité de probabilité, du taux de défaillance $z(t)$, etc. [4]. Ces fonctions probabilistes sont des représentations de la distribution statistique de la durée de vie [33]. $F(t)$ désigne généralement la fonction de distribution cumulative, c'est-à-dire la probabilité que le dispositif tombe en panne au cours d'une période donnée. La fonction de densité de probabilité (FDP ou $f(t)$) est la dérivée de la fonction de distribution. La fonction de survie/fiabilité est la probabilité qu'un élément ne connaisse pas de défaillance dans un intervalle de temps donné. L'équation est représentée par la formule (3.1) :

$$R(t) = S(t) = 1 - F(t) = Pr(T < t) \quad \text{pour } t \in \mathbb{R}^+ \quad (3.1)$$

D'autre part, le taux de défaillance $\lambda(t)$, souvent appelé taux de risque $h(t)$, modélise la relation entre la FDP et la fonction de survie [33]. Cette relation est représentée par l'équation (3.2) :

$$\lambda(t) = h(t) = \frac{f(t)}{R(t)} = \frac{f(t)}{1 - F(t)} \quad (3.2)$$

La fonction de risque cumulatif $H(t)$ est l'intégrale de la fonction de risque et est utilisée principalement pour la génération de données en simulation [33]. En ce qui concerne la fiabilité, il existe deux types de systèmes/composants : les systèmes réparables et les systèmes non réparables. Un système non réparable, comme son nom l'indique, ne peut être réparé et doit être remplacé en cas de défaillance. En d'autres termes, cela signifie qu'une défaillance entraîne la fin de la vie du système. En revanche, un système réparable peut être remis en état de fonctionnement à la suite d'actions de maintenance. Les mêmes fonctions s'appliquent aux deux types de systèmes, mais différentes distributions sont utilisées pour modéliser leur comportement.

3.3.2.1 Système non réparable

Dans le cas des systèmes non réparables, la probabilité de survie ou le taux de risque sont calculés pour déterminer l'occurrence des événements de défaillance. Le temps moyen avant défaillance exprime la durée de vie du système concerné [45]. Le taux de risque est caractérisé par trois états : décroissant au début de la durée de vie, constant pendant la durée de vie et croissant à la fin de la durée de vie. Ces comportements sont généralement présentés sous la forme d'une courbe en baignoire, comme le montre la figure 3.5.

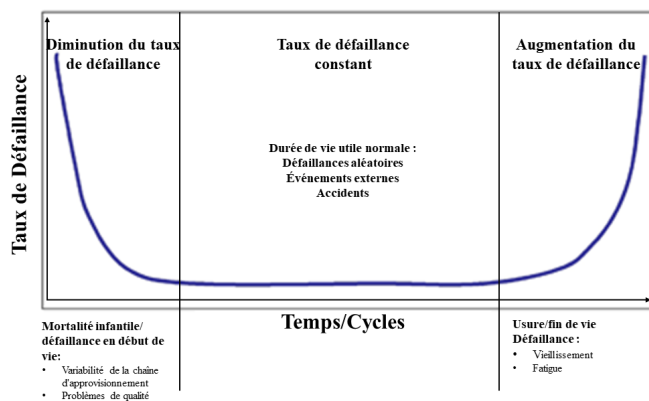


Figure 3.5 Courbe baignoire (adaptée de [3]).

Les défaillances en début de vie sont souvent associées à des défauts dus à la production, à l'assemblage, etc. Divers contrôles sont effectués pour éviter que ces produits n'atteignent le client. En ce qui concerne les défaillances en cours de vie, elles sont souvent considérées comme des défaillances aléatoires, ce qui signifie qu'un niveau inhabituel de stress sur le système dépasse sa capacité et provoque la défaillance. Enfin, lorsque le nombre de défaillances augmente (fin de vie), elles sont généralement associées au vieillissement et à la dégradation physique des composants. En utilisant les courbes en baignoire, les praticiens peuvent facilement savoir à quel stade du processus de vieillissement se trouve le système et appliquer le programme de maintenance en conséquence. Comme indiqué ci-dessus, la fiabilité est mesurée à l'aide de fonctions probabilistes, qui sont dérivées des distributions de durée de vie. En statistiques inférentielles, on parle d'analyse paramétrique; les distributions appropriées sont d'abord sélectionnées, puis les paramètres sont extraits des données. Les distributions les plus couramment utilisées dans les RAMS sont la distribution exponentielle, la distribution de Weibull et la distribution gamma [33,38]. Le tableau 3-1 montre les différentes fonctions

de représentation de ces trois distributions de durée de vie. Ces distributions utilisent toutes trois des paramètres similaires : un paramètre d'échelle, un paramètre de localisation et un paramètre de forme. Le paramètre de forme (k ou m) affecte la forme de la distribution de probabilité. Le paramètre de localisation indique si la distribution est décalée vers la droite ou vers la gauche sur l'axe du temps. Enfin, le paramètre d'échelle (λ ou θ) indique si la distribution est plus compacte ou plus étendue le long de l'axe du temps [33, 38].

Tableau 3-1 Fonctions de représentation des distributions de durée de vie

Distribution	Cumulative $F(t)$ or FDC	Densité de Probabilité $f(t)$ or FDP	Survie $S(t)$	Risque $h(t) = z(t)$	Risque Cumulatif $H(t)$
Exponentiel	$1 - e^{(-\lambda t)}$	$\lambda e^{(-\lambda t)}$	$e^{(-\lambda t)}$	λ	λt
Weibull	$1 - e^{(-\lambda t)^k}$	$k\lambda^k t^{k-1} e^{(-\lambda t)^k}$	$e^{(-\lambda t)^k}$	$k\lambda^k t^{k-1}$	λt^k
Gamma	$I(k, \lambda t)$	$\frac{\lambda(\lambda t)^{k-1} e^{-\lambda t}}{\Gamma k}$	$1 - I(k, \lambda t)$	$\frac{\lambda(\lambda t)^{k-1} e^{-\lambda t}}{\Gamma k [1 - I(k, \lambda t)]}$	$-\log(1 - I(k, \lambda t))$

3.3.2.2 Système réparable

Dans le cas d'un système réparable, la fiabilité est caractérisée par le fait que lorsqu'une défaillance se produit, le système peut être remis en état de fonctionnement. La fiabilité est alors exprimée par le taux d'occurrence des défaillances (ROCOF ou taux de réparation) ou par le temps moyen entre les défaillances (MTBF). Contrairement au système non réparable, l'événement qui nous intéresse est récurrent, et les techniques de modélisation sont choisies en conséquence. En général, le processus stochastique des défaillances récurrentes est représenté par un processus de comptage. Étant donné qu'un bien peut

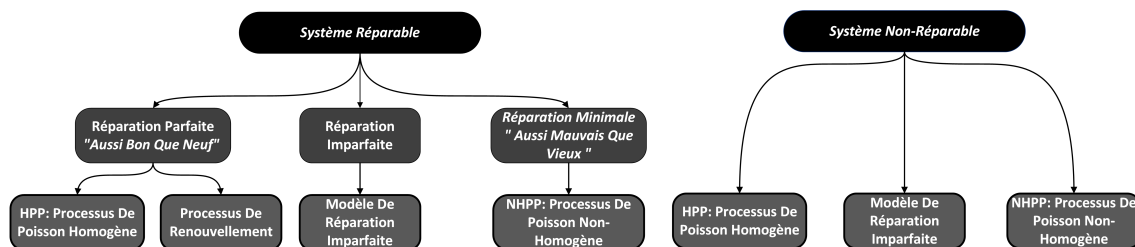


Figure 3.6 Types de systèmes, processus stochastiques et distributions [4].

connaître plusieurs défaillances au cours de sa vie utile, la fréquence d'occurrence peut augmenter, diminuer ou rester constante.

Lorsque l'actif est réparé, trois hypothèses peuvent être formulées quant à la qualité de l'intervention. La maintenance peut être considérée comme parfaite, c'est-à-dire que l'appareil est considéré comme neuf après la réparation. Dans le cas d'une réparation minimale, le système est considéré comme «aussi mauvais que précédemment» après sa réparation, ce qui signifie que le système, au lieu de revenir aux conditions initiales, est ramené à son état d'avant la défaillance [46]. Enfin, la maintenance imparfaite se situe entre ces deux extrêmes. La figure 3.6, adaptée de [4], présente les différentes approches. Pour un système réparable, le taux de défaillance est appelé fonction d'intensité, au lieu de la fonction de risque.

Les modèles de Poisson sont très courants dans la modélisation du processus de comptage. Le modèle HPP «*Homogeneous Poisson Process*» est une forme simple et populaire pour les calculs probabilistes des systèmes réparables. Ce modèle considère un taux de défaillance constant, c'est-à-dire qu'il suit une distribution exponentielle de λ et que ces événements sont indépendants et identiquement distribués [33]. Comme le taux de réparation est constant, ce modèle est le plus souvent utilisé lorsque le système est dans sa période de vie utile (voir Figure 3.5). En outre, il implique qu'une réparation ramène le système à son état initial, c'est-à-dire «comme neuf» [47]. Contrairement au modèle

HPP, le modèle NHPP «*Non-Homogeneous Poisson Process*» ne prend pas en compte la maintenance parfaite. Le taux de défaillance n'est pas nécessairement constant, il peut évoluer avec l'âge de l'actif et est appelé processus de loi de puissance. En d'autres termes, les réparations antérieures n'affecteront pas les performances futures du système. En outre, le modèle HPP est un cas particulier du modèle NHPP, lorsque les paramètres de forme sont égaux à 1. Les fonctions de représentation des deux modèles sont présentées dans le tableau 3-2. Ce modèle offre une grande flexibilité, car il est en mesure de représenter des taux de défaillance croissants (détérioration du système), décroissants (régénération du système) et constants. En complément, la fonction d'intensité de la loi de puissance a la même forme que la fonction de risque d'une distribution de Weibull, présentée dans le tableau suivant. Dans le cas du NHPP, le système est considéré comme «aussi vieux que neuf» après réparation [33,47].

Tableau 3-2 Fonctions de représentation pour les systèmes réparables.

Distribution	Cumulatif $F(t)$ or FDC	Densité de Probabilité $f(t)$ or FDP	Survie $S(t)$ or $R(t)$	Risque $h(t)$ or $\lambda(t)$	Risque Cumulatif $H(t)$
HPP	$1 - e^{-\lambda t}$	$\lambda e^{-\lambda t}$	$e^{-\lambda t}$	λ	λt
NHPP	$1 - e^{(-\lambda t)^k}$	$k\lambda^k t^{k-1} e^{(-\lambda t)^k}$	$e^{(-\lambda t)^k}$	$k\lambda^k t^{k-1}$	λt^k

3.3.2.3 Analyse de survie

Les analyses paramétriques sont essentielles pour estimer le taux de défaillance d'un groupe d'actifs. Ces analyses sont très courantes dans le domaine de la fiabilité. Cependant, elles ne considèrent que les événements de défaillance, sans tenir compte de l'influence d'autres facteurs sur le système. À cet égard, les modèles de survie permettent

de déterminer une relation entre ces événements de défaillance et certaines variables [46, 48]. Comme pour l'analyse paramétrique, il existe des techniques spécifiques pour les systèmes réparables et non réparables. Les modèles les plus connus, pour traiter les cas non réparables, sont le modèle de risque proportionnel (modèle de Cox), le modèle additif d'Aalen et le modèle de durée de vie accélérée. Plusieurs chercheurs se sont efforcés de développer des extensions capables de traiter les cas d'événements récurrents. Les variations les plus courantes des modèles de Cox, connues sous le nom de modèles d'intensité proportionnelle, sont les modèles d'Andersen-Gill, de Wei-Lin et Weissfield, et de Prentice-Williams-Peterson [49, 50]. Comme le montre le tableau 3-3, la fonction d'intensité du modèle de survie comprend un paramètre β , qui est un vecteur de coefficient, multiplié par X_i , une matrice de covariables. De cette manière, la fonction de risque de base $\lambda_0(t)$ est ajustée en fonction des covariables, ce qui n'est pas le cas avec le NHPP. Ainsi, l'analyse de survie peut être utilisée en fiabilité pour obtenir une estimation plus précise du taux de défaillance, en fonction des paramètres d'un actif spécifique.

Tableau 3-3 Comparaison des fonctions d'intensité.

NHPP	$k\lambda t^{k-1}$
Anderson-Gill	$\lambda_0(t)e^{\beta X_i(t)}$

Il existe de nombreuses applications de la modélisation mathématique pour la RAMS. Par exemple, les auteurs de [51] présentent une analyse de fiabilité complète pour différents types de transformateurs de puissance d'une société d'électricité australienne. Ils comparent les résultats obtenus par l'analyse paramétrique avec Weibull à l'analyse non paramétrique avec les courbes de Kaplan-Meier. En outre, une analyse des modes de

défaillance est utilisée pour déterminer les principales causes de défaillance en fonction de l'âge des actifs. Dans [52], l'auteur propose une méthodologie pour quantifier l'incertitude liée à la qualité des données dans l'application d'outils statistiques pour estimer la fiabilité. Une étude de cas sur les actifs des services publics d'électricité au Canada est présentée, et les méthodes HPP et NHPP sont utilisées pour tester l'approche. En outre, une étude de cas appliquée au système électrique néerlandais est présentée par [53]. Les statistiques de fiabilité sont obtenues à l'aide de l'analyse de survie, en utilisant les courbes de Kaplan-Meier, et les paramètres sont estimés à l'aide d'une distribution de Weibull. Ces résultats sont ensuite utilisés dans des scénarios de simulation pour optimiser les politiques de maintenance et de remplacement.

3.4 Pronostic et gestion de la santé

Comme son nom l'indique, le pronostic et la gestion de la santé des actifs (PHM) est une discipline qui se concentre sur la gestion de l'état de santé des systèmes, les pronostics et les techniques de diagnostic améliorées [54]. Alors que la RAMS se concentre sur l'étude des propriétés globales d'une population d'actifs, le PHM propose des approches beaucoup plus ciblées, en suivant des actifs individuels. Un facteur important est que cela implique que les actifs doivent être surveillés en continu afin de garantir les fonctions du système. Cette discipline s'appuie donc sur des données générées par des capteurs, ce qui diffère de la RAMS, où les données historiques sont généralement utilisées pour la modélisation. Le système de mesure permet des diagnostics en temps réel ainsi que la détection et l'isolation des défaillances. Les techniques de pronostic permettent de passer d'un mode de décision réactif à un mode de décision préventif, notamment dans l'exécution des tâches de maintenance. L'objectif général de cette discipline est d'optimiser l'utilisation d'un bien tout au long de sa durée de vie utile. En d'autres

termes, les objectifs du pronostic s'alignent sur ceux de la gestion des actifs et du RAMS. L'estimation de la durée de vie résiduelle du système est la base du domaine du pronostic. Pour tenir compte des différents modes de défaillance, le pronostic implique de faire plusieurs estimations de la durée de vie résiduelle. Ces estimations sont réalisées à l'aide de données historiques, de scénarios prévisionnels, de modèles qualitatifs (AMDEC), de données de fabrication, etc. En outre, le pronostic vise à prédire l'état de santé d'un système au fil du temps, afin d'adapter les actions de maintenance et la prise de décision [6]. Étant donné que la PHM consiste à effectuer la maintenance sur la base de l'état prédit obtenu par la surveillance, elle est appelée maintenance conditionnelle, et plus précisément maintenance prédictive.

La figure 3.7 montre qu'avant les pronostics, les décisions étaient prises sur la base de données et de diagnostics. Avec le pronostic dans le processus, la prise de décision est influencée par la modélisation prédictive; avec l'estimation de la durée de vie utile restante, le temps jusqu'à la défaillance peut être prédit, et la maintenance effectuée sur la base de l'état réel du système. Les méthodes PHM sont divisées en approches basées sur la physique, approches basées sur les données et approches hybrides [6]. Parfois, certains articles incluent des approches basées sur la fiabilité dans la modélisation PHM, mais les prochaines sous-sections décrivent les trois premières approches, étant donné que les approches basées sur la fiabilité ont déjà été discutées.

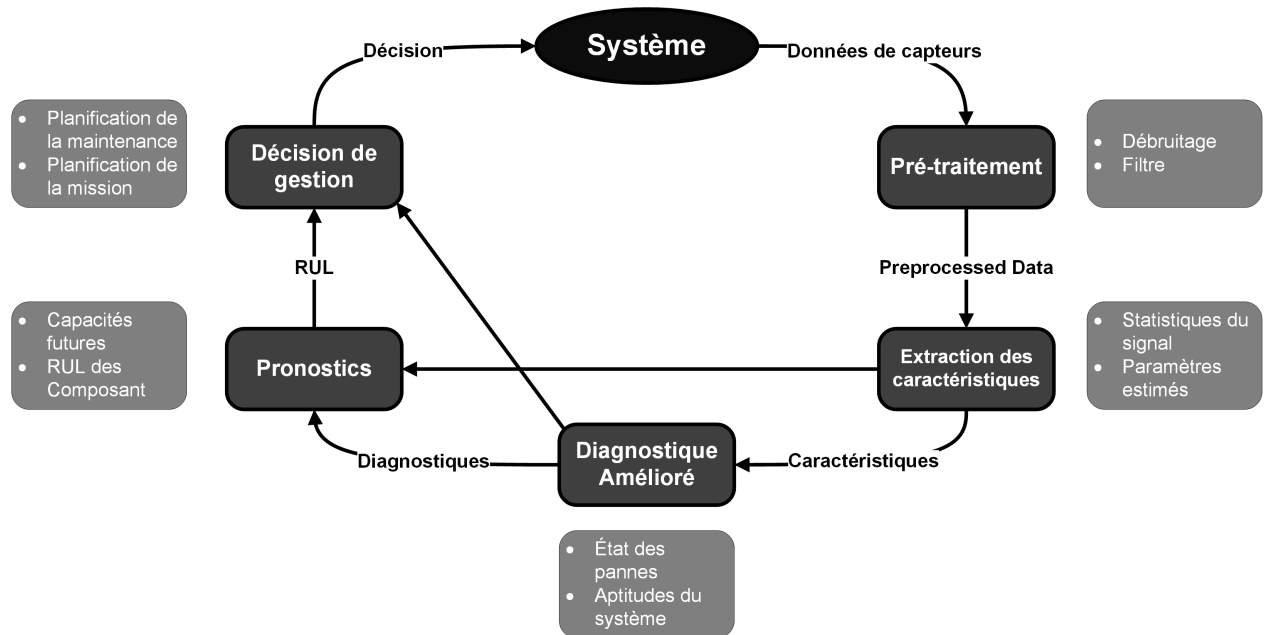


Figure 3.7 Pronostic dans un processus de gestion de la santé [5].

3.4.1 Modélisation qualitative dans le domaine du PHM

Comme pour la fiabilité, les praticiens du pronostic s'appuient sur des modèles qualitatifs pour établir une base de connaissances sur les biens et leur dégradation. En effet, il est courant d'utiliser l'analyse des modes de défaillance, de leurs effets et de leur criticité (AMDE/AMDEC), les arbres de défaillance et d'autres méthodes pour identifier les risques [5].

3.4.2 Modélisation mathématique en PHM

Il existe plusieurs méthodes mathématiques pour représenter les systèmes dans le domaine du pronostic. Les types de méthodes les plus courantes sont décrites dans les prochaines sections.

3.4.2.1 Modèle basé sur la physique

Dans le domaine du pronostic, la modélisation basée sur la physique, ou simplement modèle physique, est l'une des méthodologies les plus précises pour l'estimation des paramètres du système. Pour développer ce type de modèle, il est nécessaire d'avoir une connaissance approfondie du système, des interactions entre les composants et l'environnement, etc. En outre, il requiert généralement une connaissance approfondie des mathématiques et des concepts de la physique des dégradations. Comme le montre la figure 3.8, les modèles physiques peuvent être développés à l'aide de la méthode des éléments finis, en se concentrant sur la physique des matériaux d'un composant, ou même par simulation du système et de ses facteurs de stress. Ils peuvent également être établis à partir des lois fondamentales de la physique à l'aide d'équations différentielles. Dans cette approche, un modèle physique est développé pour un système, un sous-système ou un composant, décrivant le système sur la base d'équations mathématiques. Ces équations représentent les modes de défaillance et les phénomènes de dégradation, sur la base des lois physiques applicables. Une fois le modèle établi, il permet d'établir un diagnostic tout en surveillant le bien en fonctionnement. Compte tenu de l'état du système, il est possible de prédire son comportement futur. Les modèles basés sur la physique peuvent être extrêmement précis et très utiles pour le pronostic. Toutefois, cela implique que le modèle reste fidèle à la réalité. Ainsi, le développement peut être difficile et nécessiter des connaissances approfondies. En conséquence, la complexité des modèles et la fidélité du modèle peuvent entraîner des difficultés de calcul [5, 6].

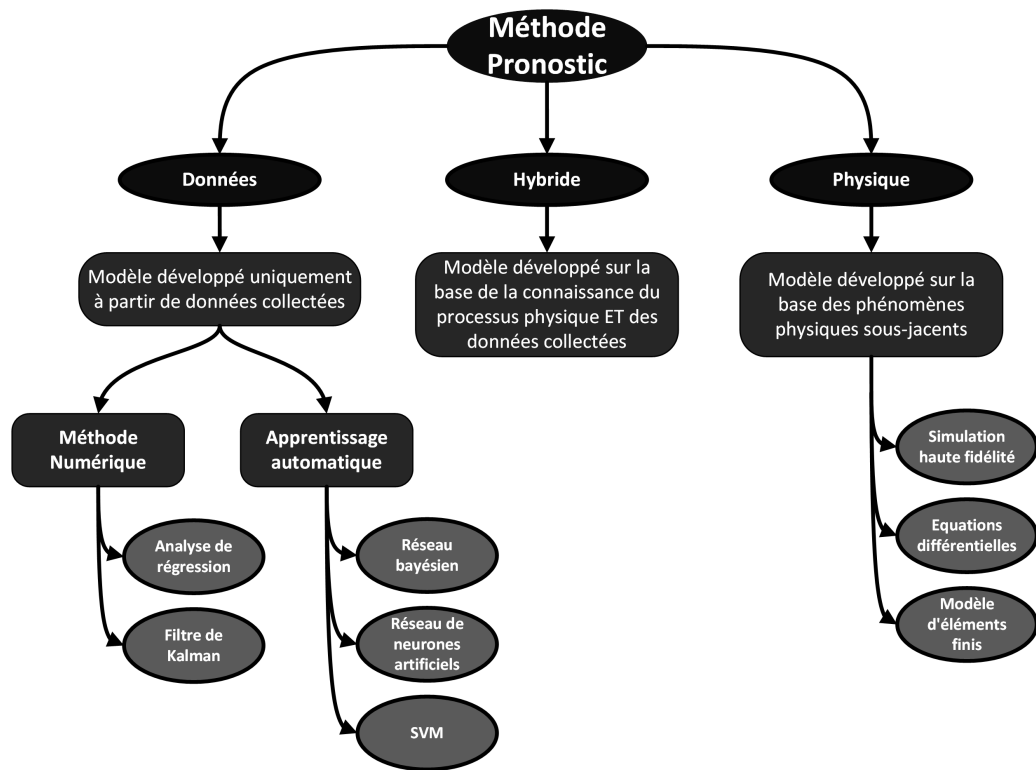


Figure 3.8 Méthodes pronostiques (adaptées de [5, 6]).

3.4.2.2 Modèle basé sur les données

La modélisation fondée sur les données est, en théorie, uniquement basée sur les données collectées à partir du système [6]. Les approches fondées sur la physique sont généralement plus précises que celles qui reposent sur les données [5]. Cependant, les approches par les données nécessitent généralement moins de connaissances sur le système et les phénomènes de dégradation, ce qui les rend très attrayantes [6]. C'est encore plus vrai lorsqu'il est trop difficile de modéliser le système avec des approches basées sur la physique. En effet, celles-ci permettent un développement plus rapide et sont donc plus répandues dans le domaine du pronostic. De manière simplifiée, ces méthodes consistent à utiliser des données historiques ou des données de capteurs pour déterminer

les tendances du système. Lorsque le système se dégrade, c'est parce que les variables observées s'écartent de ces tendances et dépassent un certain seuil préétabli. Ces modèles peuvent ensuite être utilisés pour estimer ou prédire la durée de vie utile restante du système [6]. Les techniques basées sur les données comprennent les méthodes statistiques et l'apprentissage automatique. Les inconvénients de ces techniques sont les mêmes que pour les projets de science des données : les modèles nécessitent une grande quantité de données pour décrire le phénomène, ce qui peut s'avérer difficile lors de l'acquisition de données sur les défaillances. En outre, le choix des bonnes méthodes et la prévention du sur/sous-apprentissage sont également des défis importants. Les applications basées sur les données et l'apprentissage automatique sont abondantes dans la littérature. Dans [55], les auteurs présentent le développement d'un modèle de maintenance prédictive prenant en compte les covariables et les interactions possibles, appliqué à une centrale thermique solaire. Le modèle développé combine l'analyse de survie avec des réseaux neuronaux artificiels. L'objectif est de l'intégrer dans un système de surveillance d'état dynamique de la pompe de transfert de chaleur et de prédire les défaillances en fonction de cet état. [56] propose une étude pour la sélection des variables pertinentes afin de faciliter le déploiement d'un programme de maintenance prédictive et de surveillance en temps réel. Dans la même optique, la Réf. [57] propose un cadre et une méthodologie pour réduire la dimensionnalité des données sur le cycle de vie des actifs et minimiser la perte d'informations.

3.4.2.3 Modèle hybride

De toute évidence, la modélisation requiert une compréhension fondamentale du système ainsi que des modes de défaillance et des données disponibles (capteurs ou historique), y compris pour les techniques appuyées sur les données. La plupart des

applications en PHM sont développées à partir des données et de la physique du système. Par exemple, comme pour tout projet de science des données, une connaissance préalable du système et de son fonctionnement est nécessaire pour sélectionner les méthodes appropriées (régression, classification, regroupement, etc.), les variables pertinentes et les mesures de performance. C'est là que les modèles hybrides entrent en jeu. Les techniques hybrides combinent des approches de modélisation basées sur les données et sur la physique des défaillances. De cette manière, les techniques basées sur la physique sont appliquées lorsque c'est possible, et si ce n'est pas le cas, les données sont utilisées pour pallier le manque de connaissances. Le pire scénario est celui où la modélisation par la physique est impossible et où il n'y a pas assez de données. C'est l'un des principaux problèmes que peut poser cette méthode. Comme elle présente les avantages des deux types de techniques, elle peut aussi en présenter les inconvénients.

3.5 Discussion

Comme ce fut démontré, les deux domaines de recherche sont complémentaires, bien qu'ils diffèrent à certains égards. La fiabilité se concentre sur la santé d'un groupe de systèmes, tandis que le PHM étudie un seul système de plus près. Un parallèle peut être établi avec le domaine de la médecine, où la santé publique se concentre sur l'étude des questions de santé à l'échelle nationale, provinciale ou régionale, tandis que les médecins traitent les patients individuellement, sur la base de leurs antécédents, de leurs problèmes actuels et des données collectées lors des examens médicaux. Dans la perspective du modèle hiérarchique de prise de décision, les stratégies RAMS se situent principalement au niveau tactique. En effet, les modèles permettent des décisions à moyen terme sur la durée de vie des actifs et le niveau de détail fourni est modéré. Quant au PHM, il s'agit généralement de stratégies opérationnelles; la relation avec le bien est directe, et

le niveau de détails de l'information est très élevé et permet des décisions à court terme, voire en temps réel. Néanmoins, les deux domaines ont généralement une influence à plusieurs niveaux de l'entreprise, jusqu'au choix de la conception du système. Quatre types d'approches de modélisation ont été présentés : qualitative, basée sur la physique, basée sur les données et hybrides. Les approches qualitatives sont utilisées pour établir les connaissances de base sur le système, sa conception, les interactions entre les composants et les risques de défaillance associés. Les approches qualitatives sont utilisées à la fois pour le RAMS et le PHM, et les approches les plus couramment mentionnées dans la littérature sont l'analyse des modes de défaillance, des effets et de la criticité, et l'analyse de l'arbre de défaillance. Les techniques basées sur la physique des défaillances sont davantage associées au domaine de la PHM. Les approches basées sur les données sont applicables aux deux domaines et certaines techniques sont similaires. Le RAMS est un domaine plus ancien, et les méthodes mathématiques découlent généralement des statistiques inférentielles et de l'analyse de régression. Le PHM est un domaine de recherche plus récent qui tend à utiliser des technologies de capteurs et de connectivité, et la modélisation est souvent réalisée avec des techniques d'apprentissage automatique. Les techniques de fiabilité, telles que les modèles exponentiels (Weibull, Gamma, etc.), sont parfois utilisées dans les PHM, mais ces techniques nécessitent de nombreuses données de défaillance, qui peuvent être difficiles à obtenir lors de l'étude d'un seul système. Le temps nécessaire pour accumuler suffisamment de données de défaillance seraient prohibitifs. Au lieu de cela, la PHM s'appuie sur les données des capteurs pour développer des modèles de maintenance prédictive et de détection des anomalies. Lorsque les modèles sont construits à partir de données, la procédure est similaire à celle utilisée dans la science des données, que ce soit pour RAMS, PHM ou toute autre application. En outre, l'ingénierie de la gestion des actifs vise à coordonner les efforts pour optimiser le cycle de vie des actifs. Son application est

d'une importance cruciale pour les organisations à forte intensité d'actifs, telle que les secteurs de l'énergie, de l'aéronautique ou de l'aérospatiale. Le cadre de la GDA fournit des stratégies à long, moyen et court terme dans la gestion des activités. Ainsi, il intègre le RAMS et le PHM dans la planification et l'optimisation du cycle de vie, le RAMS ayant un objectif général sur le long et le moyen terme, et le PHM sur un horizon décisionnel à moyen et court terme. Il est également important de reconnaître l'impact de la gestion des actifs, non seulement d'un point de vue économique, mais aussi d'un point de vue social et environnemental. Les objectifs de la gestion des actifs contribuent activement au développement durable. En effet, l'optimisation du cycle de vie des actifs vise à réduire les déchets grâce à une vision à long terme, depuis une conception plus durable jusqu'à la mise au rebut du système, en passant par une maintenance optimale. Comme l'ont montré de nombreuses applications, les stratégies de gestion et de maintenance des actifs sont largement utilisées dans les secteurs public et énergétique, où les ressources doivent être soigneusement allouées.

Chapitre 4 - Applications de l'apprentissage automatique pour l'ingénierie de la fiabilité

L'objectif de ce chapitre est de fournir une vue d'ensemble des différentes techniques d'apprentissage machine (ML) du point de vue des techniques de modélisation traditionnelles. Il s'agit d'une adaptation d'un article publié sur le sujet en 2023 [19]. En outre, il présente une méthodologie pour l'application de la science des données et la manière dont l'apprentissage automatique peut être appliqué à chaque étape. Enfin, il démontrera comment les techniques d'apprentissage automatique peuvent être complémentaires aux approches traditionnelles.

4.1 Définition des concepts de l'IA

La première partie de ce chapitre sert à établir les concepts de base de l'intelligence artificielle, en apportant des définitions claires, et en présentant les différentes méthodes d'apprentissage couramment utilisées.

4.1.1 Apprentissage automatique, intelligence artificielle et science des données

L'apprentissage automatique est une forme de modélisation mathématique qui permet à un système d'apprendre à partir de données et non par la programmation explicite des contraintes et de l'environnement d'un système [58]. Cette définition de l'apprentissage automatique, élaborée par IBM (International Business Machines Corporation), met en évidence les différentes composantes importantes de l'apprentissage automatique. Tout d'abord, il s'agit d'un système, d'une machine qui apprend. L'apprentissage est un ensemble de processus qui cherchent à développer ou à modifier le comportement par l'expérience ou l'interaction avec l'environnement. Un autre point important est que l'apprentissage se fait sur des données, et non à partir d'une programmation explicite,

par exemple en recherche opérationnelle, où les contraintes d'un système doivent être spécifiées par des équations mathématiques. Cela signifie qu'un modèle mathématique est généré par l'expérience acquise à partir des données envoyées à l'algorithme. En général, les approches de ML sont divisées en trois catégories : les méthodes d'apprentissage supervisé, les méthodes d'apprentissage non supervisé et les méthodes d'apprentissage par renforcement. Il existe aussi l'apprentissage profond, dont les méthodes peuvent être utilisées pour faire de l'apprentissage supervisé ou non, ainsi que par renforcement.

4.1.2 Histoire de l'intelligence artificielle

À l'ère du numérique, il est indéniable que l'intelligence artificielle est un domaine scientifique en pleine effervescence. Étonnamment, les concepts de l'intelligence artificielle sont développés depuis près de 80 ans. On peut donc se demander pourquoi cette recrudescence se produit aujourd'hui. Cette section, complétée par la figure 4.1, décrit un bref historique du développement de l'IA et tente de mettre en évidence les raisons de cet intérêt. Dès 1943, une étude a présenté le premier concept de neurones artificiels capables d'effectuer des opérations logiques. En 1950, le mathématicien anglais Alan Turing propose un test, le jeu d'imitation ou test de Turing, pour tester l'intelligence des machines [59]. En 1955, le terme «intelligence artificielle» a été introduit par John McCarthy, Marvin Minsky, Nathaniel Rochester et Claude Shannon dans le cadre du projet de recherche estival de Dartmouth sur l'intelligence artificielle (1956). Cet événement est parfois considéré comme la naissance de l'intelligence artificielle en tant que domaine d'étude. En 1958, Frank Rosenblatt a réalisé la première mise en œuvre de l'algorithme du perceptron, basé sur les travaux de McCulloch et Pitt sur les neurones artificiels. En 1959, le concept d'apprentissage automatique a été présenté par Arthur Samuel. En 1965, Edward Feigenbaum et son équipe de l'Université de Stanford développent DENDRAL,

le premier système expert capable d'automatiser la prise de décision et la résolution de problèmes [60]. La même année, le scientifique américain Gordon Moore prédit que, selon ses observations, le nombre de composants dans les circuits électroniques doublera chaque année. Cette prédiction, connue aujourd'hui sous le nom de loi de Moore, a été corrigée en 1975, proposant plutôt que la capacité de calcul double tous les deux ans et cette prédiction s'est réalisée jusqu'à aujourd'hui. Dans leur article de 1969, Marvin Minsky et Seymour Papert décrivent certaines limites des réseaux neuronaux, notamment le manque de puissance de calcul des ordinateurs de l'époque [60]. Cet article a ralenti la recherche sur l'apprentissage profond, et le succès des systèmes experts a poussé la recherche vers ce domaine. Au cours des années 1970 et 1980, la prolifération des systèmes experts a eu lieu : MYCIN (1972), XCON (expert Configurer) (1978), Deep Blue (1997), etc. Cependant, la manière dont les systèmes experts sont construits limite leur capacité : il s'agit d'une collection de règles représentées par une séquence d'énoncés «si» et «alors» permettant de résoudre des problèmes [59]. En revanche, les algorithmes de machine learning et de deep learning apprennent un modèle avec les données, l'interaction avec son environnement, etc. En 1989, le chercheur français Yan LeCun a appliqué l'algorithme de rétropropagation à un réseau de neurones pour reconnaître des codes postaux manuscrits. En 1998, avec Léon Bottou, Yoshua Bengio et Patrick Haffner, LeCun proposaient un réseau neuronal convolutif (CNN) pour la reconnaissance de caractères manuscrits [60]. Ces succès ont progressivement ravivé l'intérêt pour l'apprentissage profond et l'apprentissage automatique, la capacité de calcul devenant de moins en moins un problème. En 2009, une équipe de chercheurs de Stanford proposait d'utiliser des processeurs graphiques plutôt que des unités centrales pour effectuer l'apprentissage. Leur projet détaille également une architecture permettant de paralléliser les calculs.

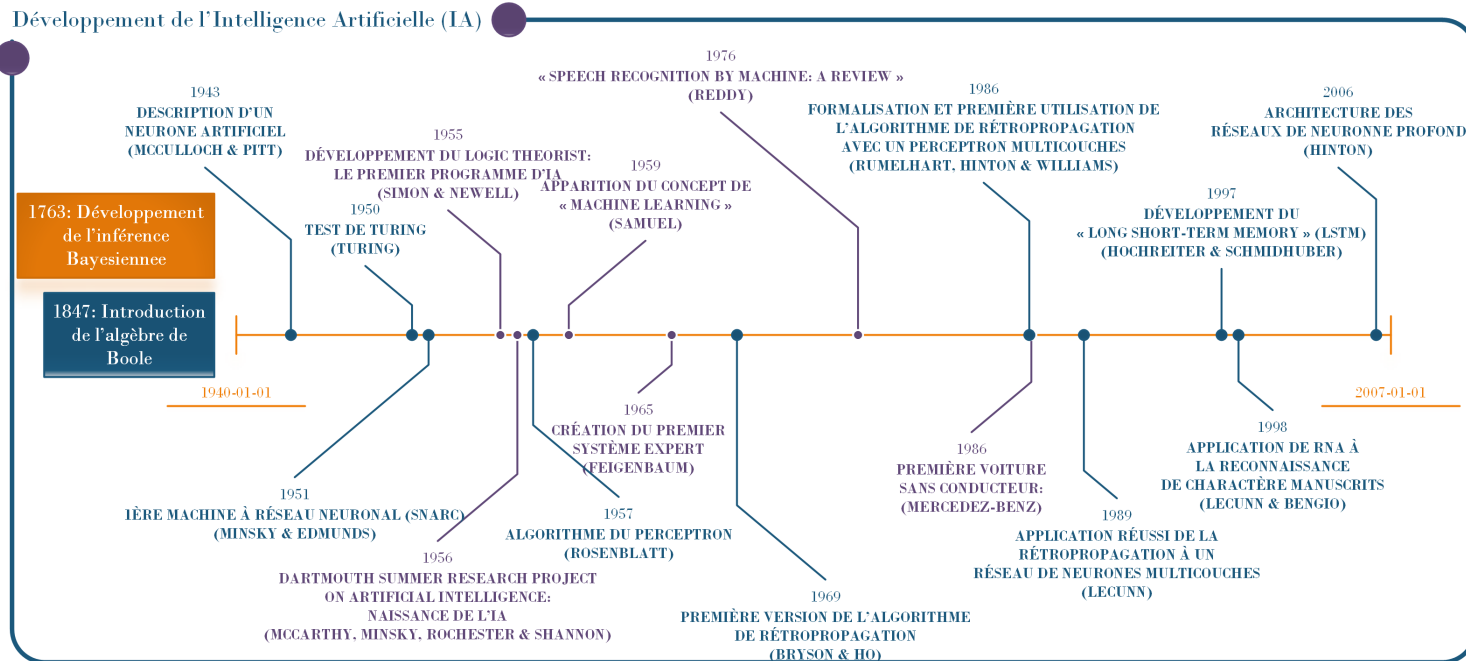


Figure 4.1 Développement de l'intelligence artificielle.

Au tournant du siècle, la connectivité s'est rapidement accrue avec le développement des téléphones intelligents et des réseaux sociaux. La démocratisation de ces technologies a entraîné une explosion des données générées. L'augmentation exponentielle du volume de données peut également être attribuée à la présence croissante des technologies de capteurs et à l'émergence de l'Internet des objets (IoT) [61]. Les capacités informatiques sont de plus en plus sophistiquées et les coûts associés aux technologies sont de moins en moins élevés. En outre, d'importantes quantités variées de données (big data) sont facilement accessibles aux organisations. En d'autres termes, ce qui était auparavant manquant pour appliquer l'intelligence artificielle est aujourd'hui largement disponible.

4.1.3 Apprentissage supervisé

L'apprentissage supervisé (SL) est le processus dans lequel la machine observe des exemples de données sous la forme de paires d'entrées et de sorties X_i, y_i [62]. La première phase de l'apprentissage est appelée entraînement et les paires X_i, y_i sont appelées données étiquetées. La figure 4.2 est un exemple visuel d'un processus d'apprentissage pour classer des images de chats et de chiens.

L'algorithme reçoit les variables \mathbf{X} (images) et fait une prédiction (chat ou chien). Connaissant la valeur à prédire, l'algorithme peut modifier son comportement (ses paramètres) à chaque exemple qu'il reçoit. L'hypothèse est qu'au fil des itérations, l'erreur de prédiction diminuera suffisamment pour que le modèle résultant soit capable de prédire la variable y , avec de nouveaux exemples X qu'il n'a jamais observés. C'est ce que l'on appelle la généralisation. Pour évaluer la capacité prédictive du modèle, les données sont divisées en deux parties : l'échantillon d'apprentissage et l'échantillon de validation. Le modèle est entraîné sur un échantillon de données, puis l'ensemble de validation est envoyé pour évaluer les erreurs de prédiction moyennes du modèle et pour améliorer ses

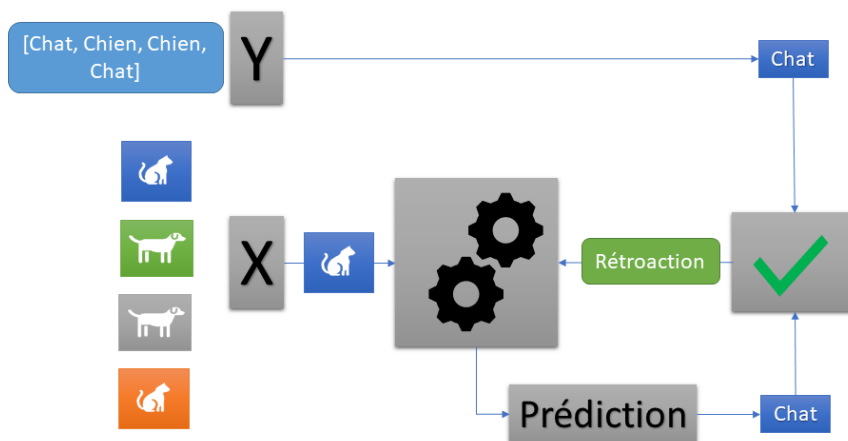


Figure 4.2 Exemple d'apprentissage supervisé.

performances en optimisant les hyperparamètres. Une fois les hyperparamètres optimisés, la validation donne un premier indicateur de la performance du modèle, en indiquant si le modèle s'ajuste bien aux données. Pour savoir si un modèle a une bonne capacité de généralisation, les données sont divisées en deux parties : l'échantillon d'entraînement et l'échantillon de test, comme le montre la figure 4.3. L'évaluation du modèle final avec l'échantillon de test donne un deuxième indicateur de performance, sur la capacité à prédire de nouvelles données. En résumé, les données sont divisées en trois parties : l'échantillon d'apprentissage, l'échantillon de validation et l'échantillon de test. Cette procédure d'échantillonnage peut être répétée plusieurs fois, afin de générer différents modèles sur le même jeu de données. Colliger les résultats à chaque itération permet une meilleure évaluation de la performance du modèle. Ce type de méthode est appelé validation croisée.

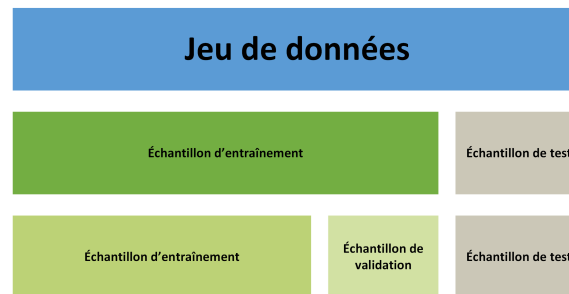


Figure 4.3 Exemple d'échantillonnage de données pour l'apprentissage supervisé.

La figure 4.4 montre deux ajustements de courbe différents, où le coefficient de détermination (r^2) et le coefficient de détermination ajusté ($r^2_{\text{ajusté}}$) nous permettent d'évaluer le degré d'ajustement aux données, c'est-à-dire la qualité de l'ajustement. La courbe du haut a un coefficient de 0,71 et celle du bas de 0,85.

Bien que la courbe du haut soit plus fidèle aux données, il est peu probable que ce modèle atteigne de bonnes performances en matière de prédiction sur de nouvelles données. Ce phénomène est souvent appelé surajustement ou surapprentissage. De la même manière, un modèle qui n'est pas performant sur les données d'apprentissage peut ne pas avoir une bonne capacité de prédiction. C'est ce qu'on appelle le compromis biais/variance. En procédant à deux évaluations des performances du modèle, on s'assure que le modèle est bien équilibré. En général, les méthodes d'apprentissage supervisé sont divisées en deux familles en fonction de la variable à prédire. Il y a tout d'abord la classification, où la variable à prédire est une variable discrète. Souvent, il s'agit de prédire une classe, une étiquette, etc. Ensuite, il y a la régression, où la variable à prédire est une variable continue. La figure 4.5 présente quelques méthodes bien connues d'apprentissage supervisé en régression et en classification.

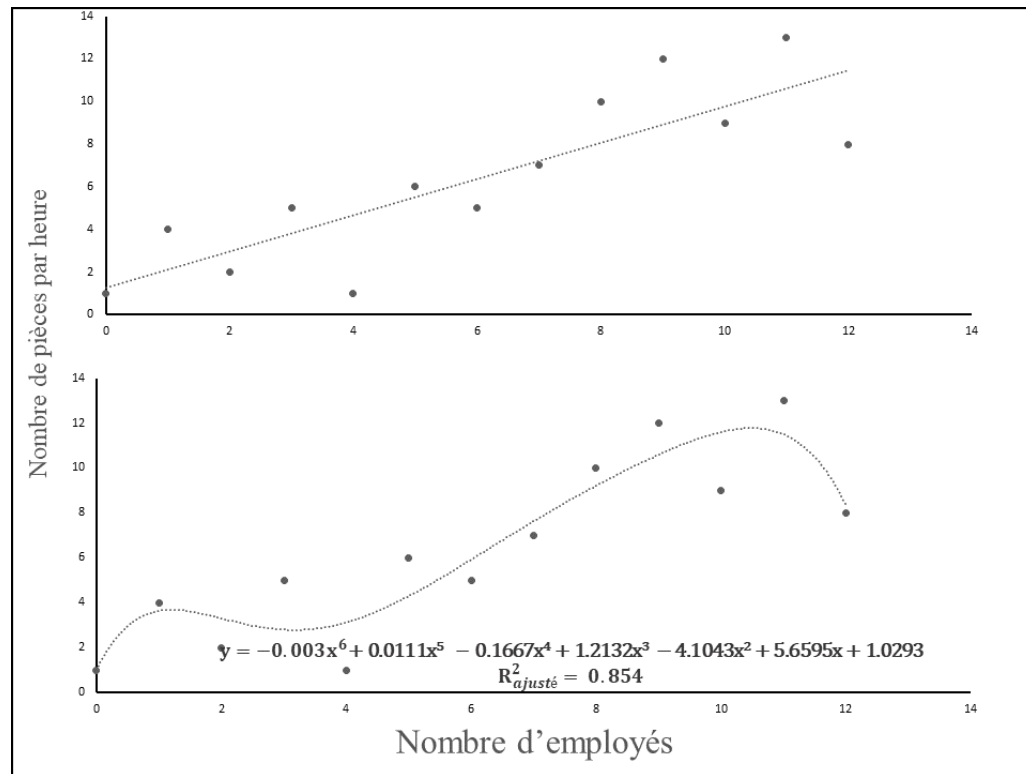


Figure 4.4 Exemple de courbe d'ajustement.

4.1.4 Apprentissage non supervisé

L'apprentissage non supervisé, par opposition à l'apprentissage supervisé, est le processus de ML dans lequel l'apprentissage est effectué à l'aide de données non étiquetées [58]. L'idée est de déterminer les relations entre les variables sans avoir de variable à prédire. Une utilisation classique de ce type d'apprentissage est le partitionnement de données ou «*Data Clustering*». L'objectif du «*Clustering*» est de catégoriser les données en sous-groupes déterminés par la similarité entre les données [63]. L'apprentissage non supervisé est souvent utilisé dans le traitement des mégadonnées, où la modélisation peut prendre beaucoup de temps, surtout si toutes les variables sont incluses. Le partitionnement peut par exemple être utilisé pour réduire le nombre de variables dans un ensemble de données en regroupant certaines variables sur la base de

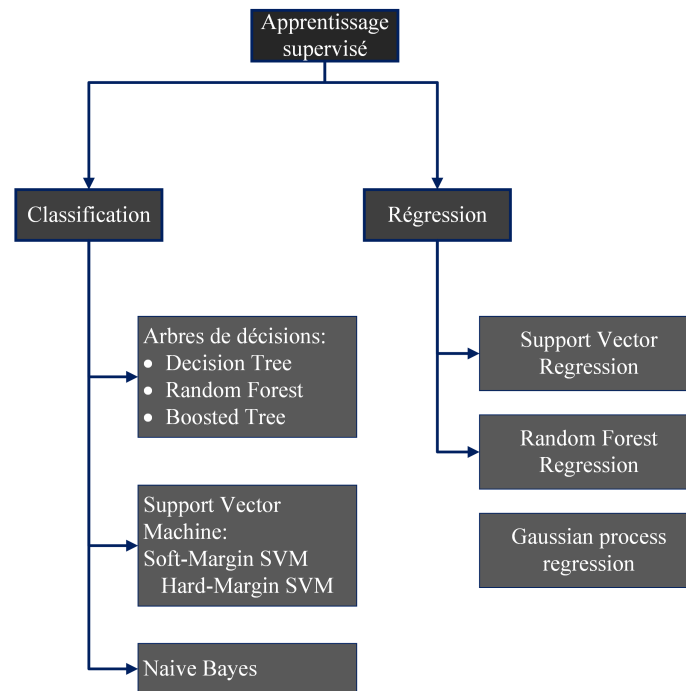


Figure 4.5 Approches et algorithmes d'apprentissage supervisé.

caractéristiques communes. Un bon exemple de regroupement est la classification des espèces animales. En classant les animaux par espèces (mammifères, poissons, oiseaux, etc.), une grande partie de leurs caractéristiques est résumée dans une seule variable. Le même type de traitement est appliqué dans l'apprentissage supervisé. Une autre catégorie d'approches d'apprentissage non supervisé est la réduction de la dimensionnalité. Comme leur nom l'indique, ces méthodes prennent un ensemble de données et en réduisent la dimensionnalité, c'est-à-dire le nombre de variables. Par exemple, l'analyse en composantes principales (ACP) consiste à prendre les données et à essayer de construire de nouvelles variables en modifiant les points de référence (axes). Les données sont projetées dans un nouveau système de représentation simplifié (moins de variables) en minimisant la perte d'information. De manière simplifiée, la méthode consiste à trouver une ligne qui minimise la somme des distances dans un nuage de points, comme dans la régression. Ensuite, les points sont projetés (projection orthogonale) sur cette ligne, qui

devient un nouveau système de référence, appelé composante principale. La figure 4.6 est adaptée de la taxonomie développée dans [63] pour inclure l'algorithme de réduction de la dimensionnalité. Elle présente une classification des différentes méthodes d'apprentissage non supervisé couramment utilisées par les professionnels.

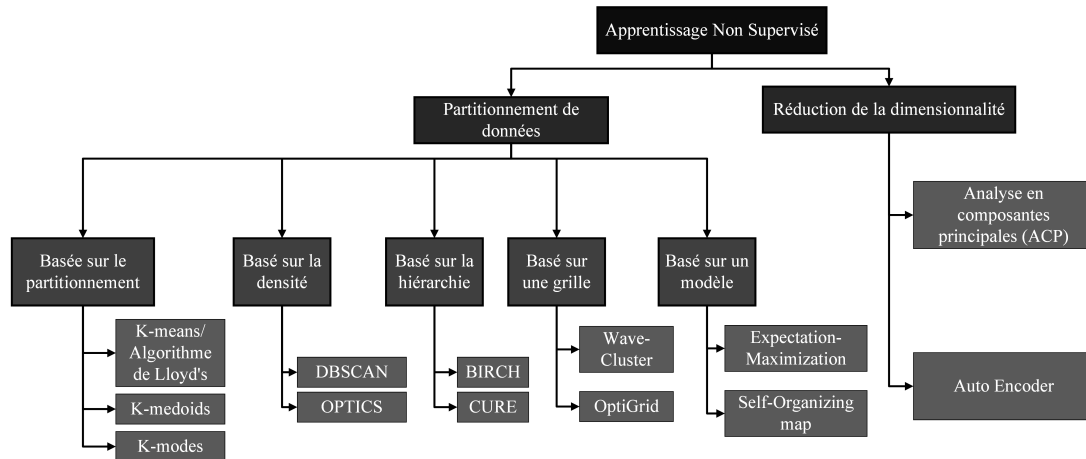


Figure 4.6 Algorithme d'apprentissage non supervisé.

4.1.5 Apprentissage par renforcement

L'apprentissage par renforcement est un autre type d'algorithme d'apprentissage automatique. Ce type d'apprentissage est assez similaire au processus étudié en psychologie comportementale, où l'on tente d'induire des comportements par le biais d'un renforcement positif ou négatif du sujet. Ces méthodes sont fréquemment appliquées en robotique et dans le domaine des jeux vidéo, par exemple [58]. Dans de nombreux cas, les données sont générées par l'interaction entre un agent intelligent (une machine) et son environnement [64]. Ces données proviennent de capteurs, qui font le lien entre la physique et le numérique, comme dans le cas des robots. D'un point de vue algorithmique, l'apprentissage se fait à travers ces interactions avec l'environnement et une fonction de pénalité/récompense qui guide les décisions de l'agent. L'objectif de l'agent intelligent est de maximiser les récompenses de ses actions. Au fur et à mesure que l'agent prend des

décisions et reçoit des informations en retour, il devient de plus en plus compétent dans l'exécution des tâches pour lesquelles il a été formé [64]. La figure 4.7 présente un résumé de la taxonomie des algorithmes d'apprentissage par renforcement développée par Zhang et Yu [65].

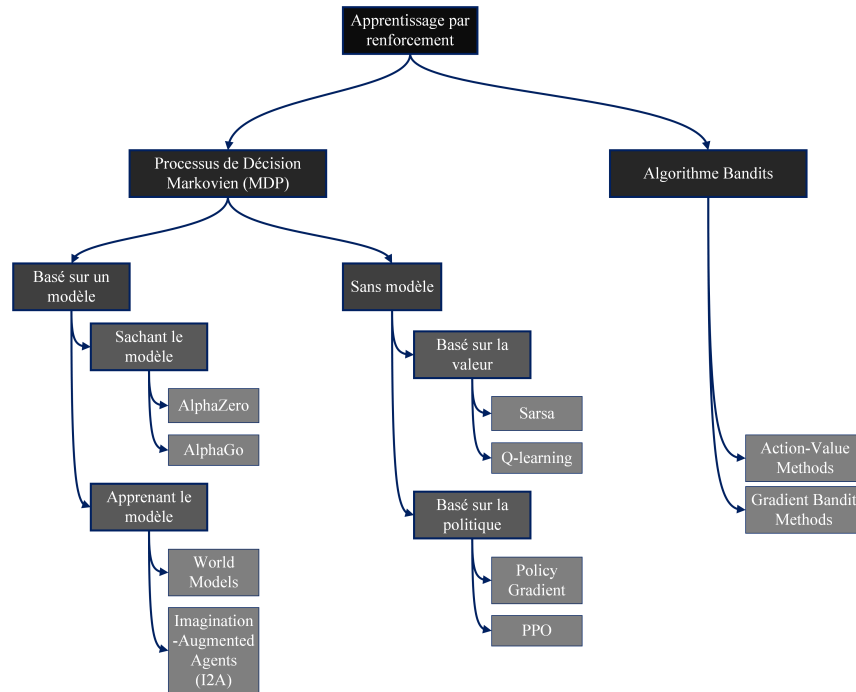


Figure 4.7 Taxonomie des algorithmes d'apprentissage par renforcement.

4.1.6 Apprentissage profond

L'apprentissage profond est une forme d'intelligence artificielle. Ces méthodes tirent leur nom de l'architecture des algorithmes de réseaux neuronaux. Le neurone est l'unité de base qui compose un réseau neuronal artificiel (RNA), comme le montre la figure 4.8 [62]. Le neurone simple, plus formellement l'algorithme du perceptron, prend un vecteur en entrée, comme le montre l'exemple de l'apprentissage supervisé. Ces valeurs sont multipliées par leurs poids respectifs et agrégées par les fonctions d'entrée, puis une fonction d'activation est appliquée pour produire un résultat de sortie.

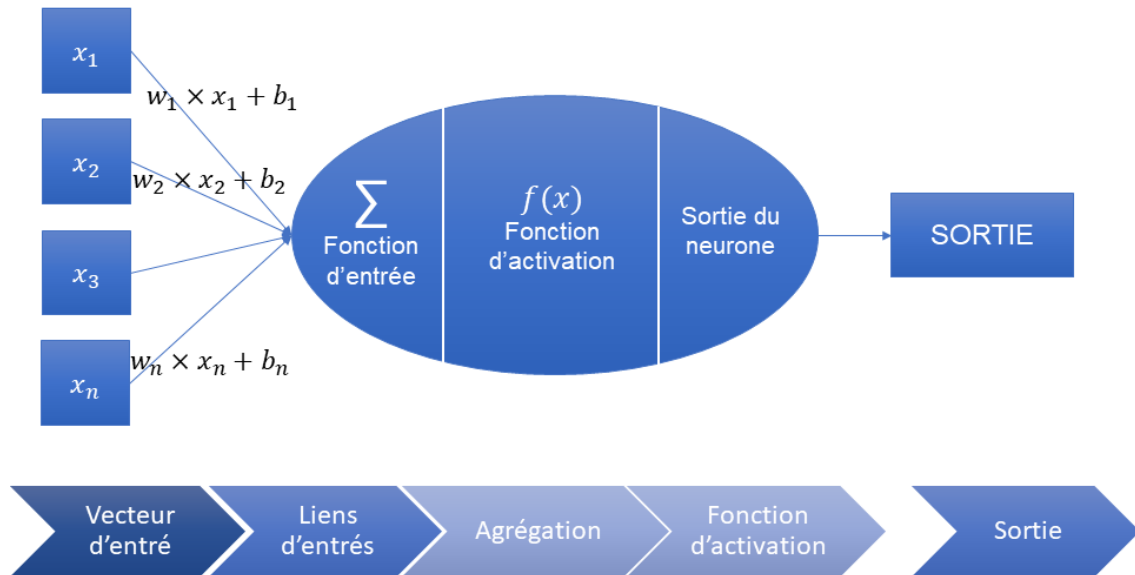


Figure 4.8 Un modèle mathématique simple pour un neurone.

Un réseau neuronal, comme son nom l'indique, est composé de couches successives de neurones, disposées selon une architecture en réseau. L'apprentissage en profondeur se produit lorsqu'un réseau comporte trois couches ou plus [58]. Le tableau 4-1 présente les trois architectures RNA les plus courantes, ainsi que leurs applications, comme indiqué dans [66].

4.1.7 Science des données

La science des données est une discipline émergente. Son objectif est d'utiliser les données pour obtenir des informations et de transformer ces données en valeur pour une organisation [67]. Les applications générales de la science des données comprennent le reporting, le diagnostic, la prédiction et la recommandation. Ce domaine combine plusieurs autres disciplines telles que l'apprentissage automatique, l'exploration des données, les statistiques, la visualisation des données et l'analyse prédictive [67]. La

Tableau 4-1 Architectures et applications RNA.

Type	Structure	Applications
Réseau de neurones artificiels	Couche d'entrée Couches cachées Couche de sortie	Classification Régression Reconnaissance des formes
Réseau neuronal convolutif	Couche d'entrée Couche de convolution Couche de «pooling» Couche entièrement connectée Couche de sortie	TALN Traitement d'images
Réseau neuronal récurrent	Couche d'entrée Couches cachées Couche de sortie	Analyse des séries chronologiques TALN Analyse des sentiments

figure 4.9 présente le cycle de vie de la science des données, qui décrit le processus général de modélisation utilisé par le professionnel. La première étape d'un projet de science des données, comme dans la recherche appliquée, consiste à définir le problème et ses objectifs en fonction de la perspective et du contexte de l'entreprise. Les données sont ensuite collectées, nettoyées et préparées pour la modélisation. La modélisation est souvent réalisée à l'aide de méthodes d'apprentissage automatique. La méthode et les mesures de performance sont sélectionnées en fonction des objectifs définis et du type de problème à résoudre.

4.2 Processus de modélisation en apprentissage automatique

Cette section vise à décrire, étape par étape, le processus de modélisation tel que couramment employé en science des données, tel que détaillé à la figure 4.9.

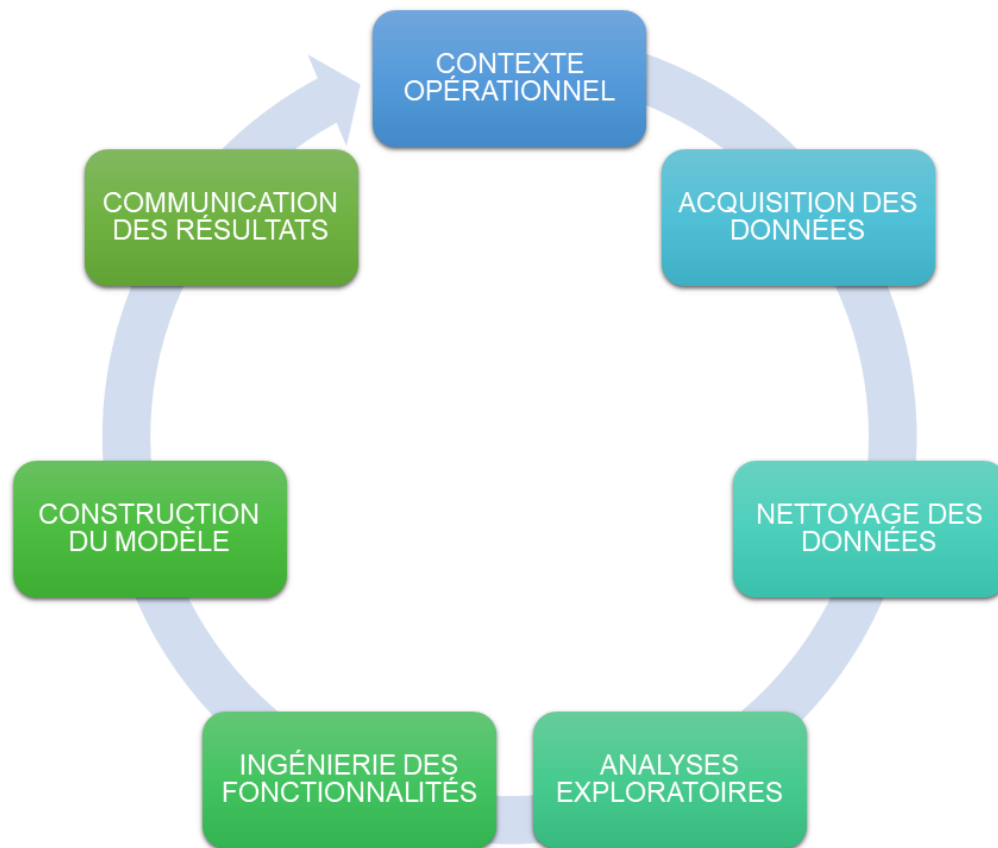


Figure 4.9 Cycle de vie de la science des données, adapté de [7].

4.2.1 Acquisition de données

La première partie de la modélisation à partir de données est le processus d'acquisition des données. Elles peuvent être générées automatiquement par la technologie de détection ou entrées manuellement par un technicien via un progiciel, par exemple. Les données sont ensuite chargées dans un environnement de stockage. Elles peuvent être structurées et stockées dans une base de données relationnelle ou non structurées et stockées dans un lac de données. Une fois que ces données sont disponibles dans un environnement de stockage, elles peuvent être préparées pour la modélisation. Il arrive que l'information

pertinente soit enregistrée dans plusieurs sources et doit être rassemblée à des fins d'analyse.

4.2.2 Nettoyage des données

La qualité des données est une préoccupation importante pour la modélisation orientée vers les données. Par exemple, une étude de cas appliquée à la fiabilité des équipements miniers montre que les données brutes sont souvent erronées, manquent de détails et de précision et ne conviennent donc pas à la prise de décision. L'étude révèle que différents champs des bases de données relationnelles présentent des erreurs dans l'affectation des tâches de maintenance au bon sous-système, dans l'attribution de codes pour décrire le type de travail, dans l'association du bon type de maintenance (conditionnelle, préventive, corrective), etc. Le nettoyage doit être effectué de manière à ce que les données soient exploitées pour produire des informations précieuses pour les décideurs. Dans un contexte idéal, il est préférable de prévenir les erreurs de saisie plutôt que de les corriger en aval. En fait, étant donné que la variance de la qualité des données est fortement influencée par l'utilisateur qui les saisit, il est essentiel de développer une meilleure gestion des travailleurs qui interagissent avec la base de données. Les organisations doivent tenir compte de l'effet de la pression du temps sur la saisie des données et fournir un retour d'information du personnel de supervision aux opérateurs. En outre, il est nécessaire d'encourager la participation des opérateurs et de valoriser leur travail de saisie des données, afin d'en améliorer la qualité [68]. Néanmoins, les scientifiques des données doivent garantir la qualité tout au long de la modélisation et le nettoyage doit tout de même être exécuté. Le processus de nettoyage implique la détection des erreurs et leur suppression ou le remplacement dans l'ensemble de données [8,69]. La figure 4.10 résume

une méthodologie permettant de quantifier la qualité des données (diagnostic) et de gérer sa qualité (correction) par le biais d'un processus continu [8].

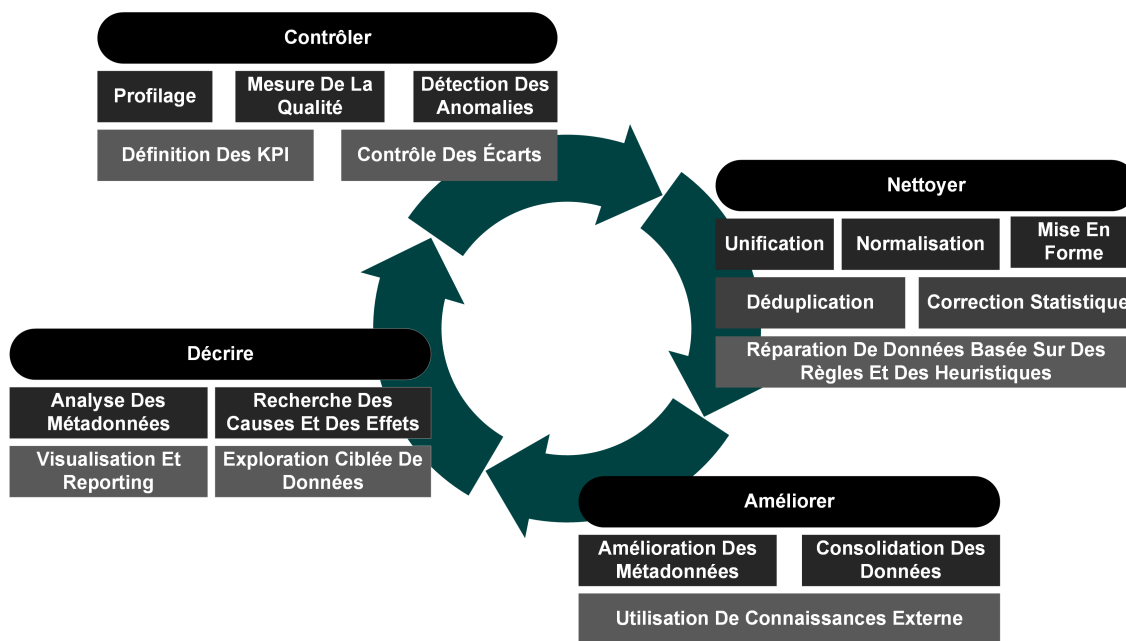


Figure 4.10 Cercle vertueux pour la gestion de la qualité des données, adapté de [8].

Le nettoyage de l'ensemble de données comprend le traitement des valeurs manquantes, des valeurs aberrantes et des mauvaises données. L'utilisateur peut décider de corriger les données à l'aide de techniques d'imputation, de les supprimer ou de les laisser en place [70]. Pour supprimer les données inadéquates, l'utilisateur peut soit retirer les entrées (lignes), soit éliminer un champ entier (colonne), en fonction de la complétude (% de données manquantes) d'une caractéristique. Pour l'imputation des valeurs manquantes, les méthodes les plus simples sont basées sur les statistiques descriptives. La mesure de la tendance centrale est un bon choix : pour les variables continues, il y a la moyenne, le mode et la médiane; pour les variables catégorielles, il y a la valeur la plus fréquente. L'interpolation est souvent utilisée pour l'imputation dans le cas d'une série temporelle. Enfin, certains utilisateurs peuvent recourir à des techniques d'apprentissage automatique

pour remplacer les valeurs manquantes. L'algorithme des k-plus proches voisins (k-NN) et d'autres méthodes de régression sont fréquemment utilisés à cette fin [69].

4.2.3 Exploration des données

Une fois les données recueillies, l'exploration des données ou l'analyse exploratoire des données est effectuée à l'aide de méthodes graphiques ou statistiques. L'analyse exploratoire des données (AED) est une pratique largement promue par le statisticien John Wilder Tukey [71]. L'idée sous-jacente à l'AED est de procéder à un examen initial des données sans aucune hypothèse. L'exploration sert à découvrir des modèles ou des anomalies, puis à formuler une hypothèse sur les données [71]. Les méthodes couramment utilisées dans l'AED sont les diagrammes en boîte, les histogrammes, les nuages de points, les cartes thermiques, etc. Le diagramme en boîte, développé par Tukey, est utilisé pour représenter graphiquement les valeurs minimales et maximales, la médiane et le quartile d'un ensemble de données. Ce graphique est très utile pour détecter les valeurs aberrantes, en plus de montrer la dispersion et l'asymétrie d'une distribution. Pour observer graphiquement la forme d'une distribution, la représentation préférée est l'histogramme, ce qui en fait un excellent outil pour l'AED. Le nuage de points est utilisé pour représenter le point de deux variables sous forme de coordonnées. Les cartes thermiques, dans le contexte de l'analyse des données, sont utilisées pour représenter les covariables les unes par rapport aux autres et démontrer les relations, souvent à l'aide d'une matrice de corrélation. L'exploration des données commence généralement au début du projet de modélisation, mais elle est utilisée tout au long du processus. Par exemple, un diagramme en boîte est une bonne méthode pour détecter les valeurs aberrantes et peut aider au nettoyage des données. En outre, les cartes thermiques constituent un bon point

de départ pour sélectionner ou éliminer des caractéristiques lors de l'étape de l'ingénierie des fonctionnalités.

4.2.4 Ingénierie des fonctionnalités

Lorsque l'on traite des données réelles, il peut y avoir des centaines de caractéristiques et il est nécessaire de sélectionner les plus pertinentes dans un ensemble de données. L'ingénierie des fonctionnalités est un processus qui comprend la sélection, la transformation, la création et l'extraction des fonctionnalités. L'objectif de ce processus est de réduire la taille de l'ensemble de données en sélectionnant et en transformant les caractéristiques afin d'optimiser l'apprentissage d'un modèle. La sélection des fonctionnalités est le processus de sélection des variables les plus pertinentes pour effectuer la modélisation. Certaines variables peuvent ne pas être pertinentes pour le phénomène étudié, tandis que d'autres peuvent avoir des effets indésirables sur le modèle. Par exemple, une caractéristique peut être redondante, car elle est fortement corrélée avec une autre variable explicative. Lorsqu'on a affaire à un grand nombre de variables, il est important de sélectionner la caractéristique qui permettra de mieux expliquer le phénomène sans pour autant nécessiter trop de calculs. Les coefficients de corrélation (Pearson) et les cartes thermiques sont de bonnes méthodes de sélection des caractéristiques, de même que les tests d'analyse de la variance (ANOVA) et les tests d'hypothèse. Parfois, des algorithmes d'apprentissage automatique tels que des modèles arborescents (forêt aléatoire, arbre de décision, etc.) sont utilisés pour la sélection des caractéristiques [72]. La transformation des fonctionnalités comprend la normalisation et la linéarisation des paramètres. La normalisation consiste à mettre à l'échelle les valeurs d'une série de manières que toutes les données contribuent de la même manière au modèle. De nombreuses techniques d'apprentissage utilisent la distance euclidienne pour calculer

la distance entre les points. Si les caractéristiques numériques ne sont pas proportionnelles, l'estimation peut être biaisée en faveur des variables les plus grandes [69, 70]. La linéarisation est une technique permettant de transformer les points d'une distribution de manière à ce qu'ils puissent être représentés par une fonction linéaire. Cette méthode est largement utilisée pour exprimer la fiabilité, lorsque les points sont représentés par une distribution exponentielle. Avec une transformation logarithmique, l'ajustement de la courbe des données donne l'équation de la distribution exponentielle ou de Weibull, par exemple [38]. L'extraction de caractéristiques est associée aux techniques de réduction de la dimensionnalité. Le concept consiste à réduire le nombre de variables en combinant les descripteurs avec une projection linéaire dans un espace de dimensionnalité inférieure. La création ou la construction de variables consiste à utiliser des variables existantes pour en créer de nouvelles, plus appropriées pour la modélisation. Parmi les exemples de création de variables, on peut citer les techniques d'encodage (encodage à une touche, encodage d'étiquettes) et le «*Binning*». Les descripteurs peuvent également être créés à l'aide de méthodes de regroupement, où les nouvelles variables représentent des groupes de points présentant une similarité [63].

4.2.5 Conception du modèle

D'après ce qui a été décrit dans les sections précédentes, les méthodes d'apprentissage automatique sont utilisées bien avant la conception du modèle pour décrire le comportement d'actifs. Cependant, c'est lors de la conception de ce modèle que les méthodes divergent entre la RAMS et le pronostic. Bien que certaines techniques soient similaires, l'objectif des modèles et leur contexte sont différents, comme l'a démontré le chapitre précédent. La figure 4.9 fait référence à la modélisation prédictive, mais elle peut être interprétée dans un contexte plus général comme une modélisation de tout

type (classification, prédiction, regroupement, ajustement de courbes, etc.) Dans le cas d'un modèle d'apprentissage automatique, la méthode consiste à entraîner le modèle et à valider ses performances, par exemple par validation croisée, puis à l'utiliser pour générer de nouvelles connaissances à partir des données. Le même processus s'applique à la modélisation statistique : le modèle est ajusté aux données, puis la qualité de l'ajustement est évaluée à l'aide de différentes mesures de performance.

4.3 Analyse des applications ML

Cette section vise à analyser la littérature sur l'ingénierie de la fiabilité, en particulier les méthodes d'apprentissage automatique qui ont été utilisées par les chercheurs. Une analyse des applications dans les domaines de la RAMS et de la PHM donnera une image claire du pourquoi et du comment de la modélisation ML et identifiera les lacunes entre les applications théoriques et les cas d'utilisation dans l'industrie.

4.3.1 Exécution et filtrage des résultats

La recherche par mots-clés sur les deux bases de données disponibles sur EBSCO a donné des centaines de résultats. Pour réduire le nombre de publications, la recherche se concentrera sur les articles des cinq dernières années qui traitent de RAMS ou de PHM. En outre, l'article complet doit pouvoir être téléchargé, car il est important d'examiner le travail. Le tableau 4-2 présente les différentes règles de sélection et d'exclusion, ainsi que le nombre de publications qui correspondent à ce critère.

4.3.2 Analyse des applications

Comme il a été mentionné, RAMS est un cadre d'évaluation et d'optimisation des performances d'un système, qui se concentre sur les caractéristiques générales d'une population, tandis que PHM est une approche plus proactive qui implique la surveillance

Tableau 4-2 Résultats du filtrage.

Règle	Nombre de Publications
Mots clés	547
De 2017 à 2023	362
Travaux non liés à l'ingénierie de la fiabilité	253
Anglais ou français uniquement	251
Texte intégral disponible	38
Révision par les pairs	34
Sélectionné à la main	19

continue d'un système pour prédire et prévenir les défaillances potentielles. Chaque article est classé en fonction de son application et de ces définitions. La figure 4.11 montre la distribution des publications chaque année, par sujet d'application. Il est observé que le nombre de publications par sujet est assez proche entre RAMS et PHM, avec quelques variations mineures au fil des années. Une forte augmentation du nombre total de publications en 2020 est remarquée, ce qui correspond à l'apparition de la pandémie, puis une baisse significative l'année suivante.

4.3.3 Revue des méthodes d'apprentissage automatique

La figure 4.12 montre les différentes méthodes d'apprentissage automatique utilisées dans les articles. Le réseau de neurones artificiels (RNA) est la méthode la plus utilisée de toutes, puisqu'il est employé dans plus de 30 % des études. En outre, si l'on considère les différentes architectures de réseaux neuronaux (réseaux convolutifs, auto-encodeurs et réseaux neuronaux récurrents (RNR) de type LSTM (mémoire à long et court terme)), plus de la moitié des méthodes relèvent de l'apprentissage profond. La figure 4.12 montre le nombre total d'utilisations de l'algorithme utilisé dans RAMS par rapport au total. Bien que les méthodes d'apprentissage profond soient les plus populaires,

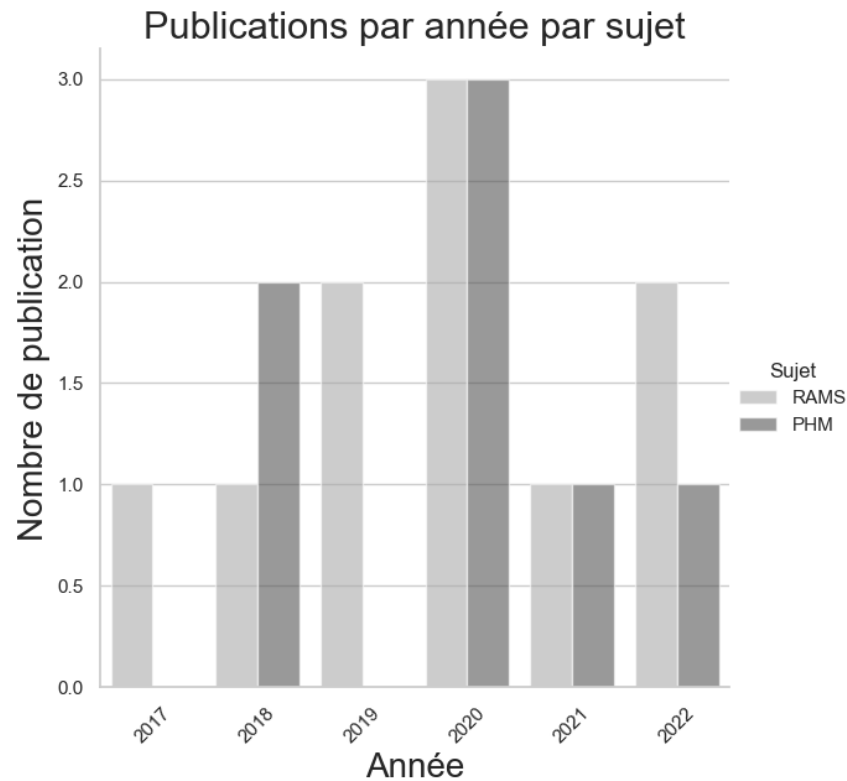


Figure 4.11 Publications par an et par thème.

la figure montre qu'elles sont généralement plus utilisées dans la recherche sur les PHM que dans les RAMS. Lorsque l'on examine les objectifs de publication dans le domaine du pronostic, il semble que la recherche soit davantage axée sur l'estimation de la durée de vie utile restante (RUL) ou sur la surveillance et le diagnostic en ligne. Ces deux applications nécessitent une grande quantité de données pour construire un modèle de prédiction supervisé, il n'est donc pas surprenant que l'apprentissage profond soit l'approche ML préférée. En outre, les méthodes de réseaux neuronaux récurrents tels que LSTM peuvent prendre en compte les dépendances temporelles grâce à leur architecture, qui contient des liaisons de rétroaction entre les couches, ce qui en fait une excellente solution pour l'estimation de la durée de vie résiduelle (RUL). Dans la RAMS, les méthodes de ML et les objectifs de modélisation sont plus variés. Par

exemple, [73] a testé des approches d'apprentissage profond pour résoudre le problème des réseaux de fabrication à flux stochastiques afin de prédire la fiabilité globale d'une chaîne de production manufacturière. Une autre étude utilise un algorithme basé sur une machine à vecteurs de support (SVM) pour résoudre un problème d'optimisation de la fiabilité d'une structure. D'autres applications RAMS appliquent l'apprentissage automatique pour simuler des scénarios possibles et évaluer la fiabilité du système [74,75]. La grande majorité des techniques, tant pour le RAMS que pour le PHM, sont des méthodes d'apprentissage supervisé; certaines études ont également utilisé l'apprentissage par transfert et l'apprentissage autosupervisé. Le tableau 4-3 indique dans quels articles chacune des techniques a été utilisée.

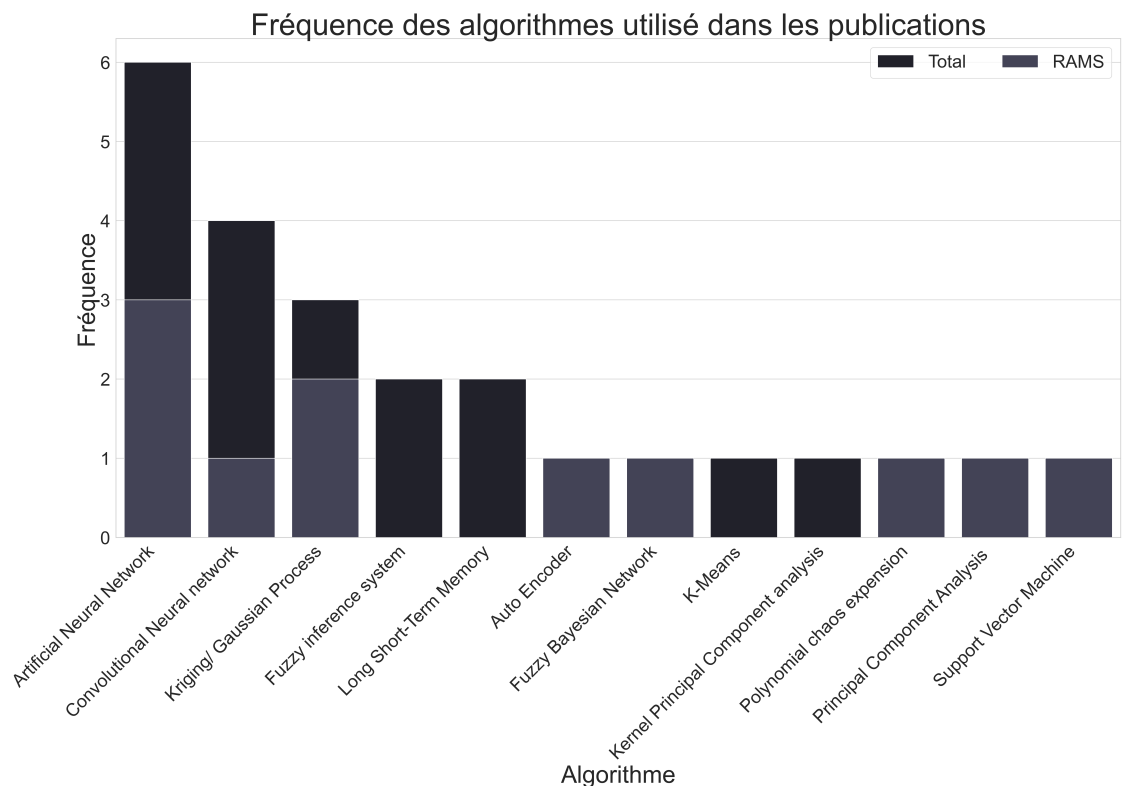


Figure 4.12 Fréquence d'utilisation des algorithmes de ML par thème.

Tableau 4-3 Algorithmes associés aux articles examinés.

Titre	Auteurs	Type de données	Algorithme
Effective software fault localization using predicted execution results	Gao, Wong et al., 2017	Operational	K-Means
An Intelligent Reliability Assessment technique for Bipolar Junction Transistor using Artificial Intelligence Techniques	Bhargava and Handa 2018	Experimental testing	Artificial Neural Network, Fuzzy Inference System
Active fault tolerant control based on a neuro fuzzy inference system applied to a two shafts gas turbine	Hadroug, Hafaifa et al., 2018	Operational	Artificial Neural Network, Fuzzy inference system
Deep Learning for Accelerated Seismic Reliability Analysis of Transportation Networks	Nabian and Meidani 2018	Simulated/ Generated randomly	Artificial Neural Network
Gaussian Process-Based Response Surface Method for Slope Reliability Analysis	Hu, Su et al., 2019	Simulated/ Generated randomly	Kriging/Gaussian Process
Fault diagnosis of multi-state gas monitoring network based on fuzzy Bayesian net	Xue, Li et al., 2019	Operational	Fuzzy Bayesian Network
Active learning polynomial chaos expansion for reliability analysis by maximizing expected indicator function prediction error	Cheng and Lu 2020	Simulated/ Generated randomly	Polynomial Chaos Expansion, Kriging/Gaussian Process
A Reliability Management System for Network Systems using Deep Learning and Model Driven Approaches	Min, Jiasheng et al., 2020	Simulated/ Generated randomly	Artificial Neural Network

Tableau 4-3 – (suite)

Titre	Auteurs	Type de données	Algorithme
Integration of Dimension Reduction and Uncertainty Quantification in Designing Stretchable Strain Gauge Sensor	Sungkun, Gorguluarslan et al., 2020	Simulated/ Generated randomly	Auto Encoder, Artificial Neural Network, Principal Component Analysis
Data-driven prognostic method based on self-supervised learning approaches for fault detection	Wang, Qiao et al., 2020	Public dataset	Kernel Principal Component Analysis
Transfer Learning Strategies for Deep Learning-based PHM Algorithms	Yang, Zhang et al., 2020	Public dataset	Convolutional Neural Network
Bounds approximation of limit-state surface based on active learning Kriging model with truncated candidate region for random-interval hybrid reliability analysis	Yang, Wang et al., 2020	Simulated/ Generated randomly	Kriging/Gaussian Process
A data-driven approach based on deep neural networks for lithium-ion battery prognostics	Kara 2021	Public dataset	Convolutional Neural Network, Long Short-Term Memory
Using Deep Neural Networks to Evaluate the System Reliability of Manufacturing Networks	Yi-Fan, Yi-Kuei et al., 2021	Simulated/ Generated randomly	Artificial Neural Network
Convolutional And Long Short-Term Memory Neural Networks Based Models For Remaining Useful Life Prediction	Gritsyuk and Gritsyuk 2022	Public dataset	Convolutional Neural Network, Long Short-Term Memory

Tableau 4-3 – (suite)

Titre	Auteurs	Type de données	Algorithme
DL-RSIM: A Reliability and Deployment Strategy Simulation Framework for ReRAM-based CNN Accelerators	Wei-Ting, Hsiang-Yun et al., 2022	Public dataset	Convolutional Neural Network
A Novel Support-Vector-Machine-Based Grasshopper Optimization Algorithm for Structural Reliability Analysis	Yang, Sun et al., 2022	Simulated/ Generated randomly	Support Vector Machine

4.3.4 Jeux de données et systèmes à l'étude

Dans cette section sont présentés les différents types d'ensembles de données utilisés dans la littérature, ainsi que les différents systèmes d'où proviennent ces données.

Comme le montre la figure 4.13, 47 % des recherches utilisent des données simulées ou des données générées aléatoirement par des fonctions mathématiques théoriques. Ensuite, 29 % des publications ont utilisé des ensembles de données publiques; ces données sont librement accessibles au public pour différentes utilisations. Une grande partie des ensembles de données publiques sont des données opérationnelles réelles. Toutefois, leur traitement est simplifié par rapport à une étude de cas réelle, et leur utilisation est principalement destinée à tester et à comparer de nouvelles approches avec des méthodes existantes. De nombreuses organisations mettent leurs données à la disposition du public sur une plateforme telle que Google Dataset Search ou Kaggle. Par exemple, dans [76], ils ont utilisé les données de la simulation de dégradation du moteur Turbofan de la NASA [77] pour comparer la nouvelle architecture RNR pour l'estimation de la durée de vie résiduelle. Dans [78], ils ont utilisé les données expérimentales de la NASA sur les batteries lithium-ion pour tester une nouvelle architecture RNC-LSTM afin d'améliorer la précision de la prédiction de la durée de vie utile restante [79]. Moins d'un quart de l'étude s'intéresse à l'analyse d'études de cas industriels (6 %) par le biais d'essais expérimentaux et de données opérationnelles (18 %).

Les types de systèmes étudiés sont très divers, comme le montre la figure 4.14. La fiabilité des réseaux informatiques et des logiciels est un sujet qui a été largement étudié, ce qui se reflète dans le graphique. Comme le montre la figure 4.13, les études théoriques utilisant diverses fonctions mathématiques sont très populaires, au lieu d'utiliser des systèmes réels. Un autre sujet relativement populaire est l'étude de la fiabilité des systèmes

de surveillance et de détection. En effet, il est important d'envisager la possibilité que les systèmes de surveillance produisent des signaux trompeurs et souffrent eux-mêmes de défaillances pour améliorer la prise de décision au sein d'une organisation.

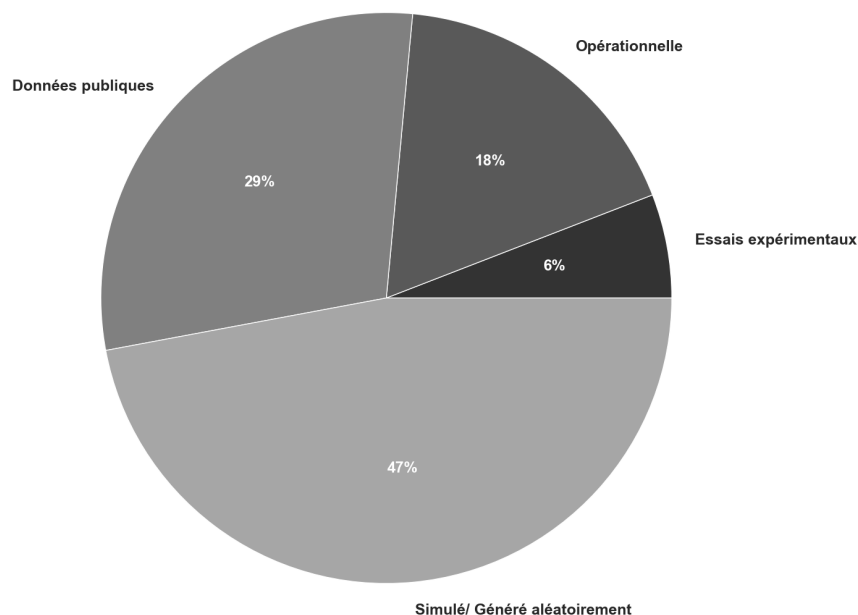


Figure 4.13 Diagramme à secteurs des types de sources de données.

4.3.5 Travaux connexes

L'étude systématique présentée dans la section précédente fournit une vue d'ensemble des données probantes sur le thème de l'apprentissage automatique appliqué à la fiabilité. Cependant, l'étude est basée sur deux bases de données, ce qui peut exclure des articles intéressants sur le sujet. Cette section vise donc à compléter l'analyse en présentant quelques applications supplémentaires ainsi que des outils de programmation pour les praticiens. Dans [80], les auteurs proposent une méthode de modélisation basée sur des agents pour la simulation afin d'étudier l'équilibrage des réseaux intelligents. L'objectif est d'utiliser ce modèle pour tester l'effet de l'équilibrage sur les réseaux électriques

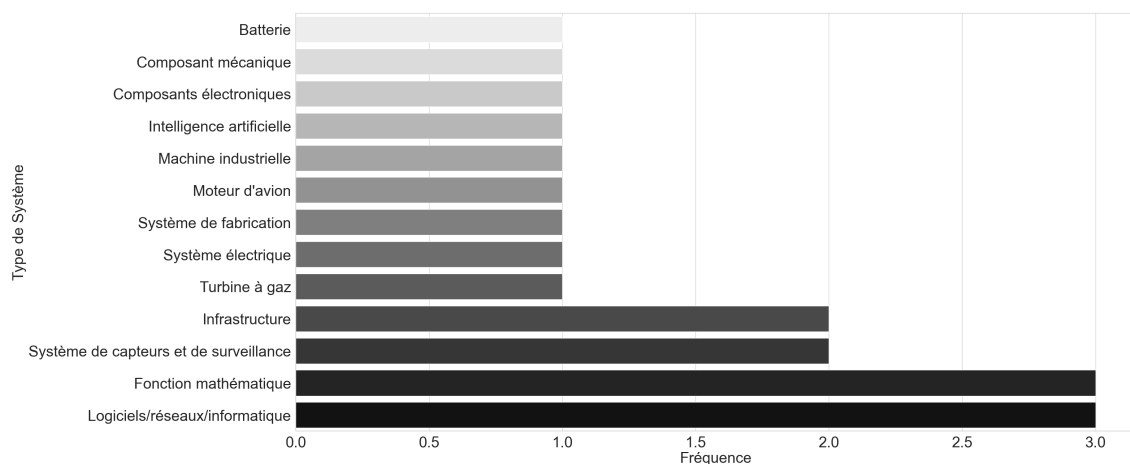


Figure 4.14 Types de systèmes étudiés.

et de télécommunications. Dans [55], on tente de développer un modèle pour prédire les défaillances, en prenant en compte plusieurs covariables tout en considérant les interactions possibles. Une approche combinant un réseau neuronal, en particulier un perceptron monocouche, avec la méthode du processus de Weibull «*general renewal process*» pour l'ajustement des courbes est présentée. Enfin, l'approche est testée à travers une étude de cas sur les centrales solaires en analysant la fiabilité des pompes thermiques. Dans [57], une structure et une méthode sont développées pour réduire la dimensionnalité des données du cycle de vie des actifs tout en minimisant les pertes d'information. Cette application PHM se concentre également sur l'ingénierie des fonctionnalités en introduisant une méthode de transformation des données pour préparer les modèles ML pour la maintenance prédictive. Le cadre proposé semble prometteur, mais l'article manque d'une application pour démontrer l'applicabilité et la pertinence de la méthode. D'autre part, dans [56], une méthodologie similaire est proposée dans le cadre d'une étude de cas sur un centre d'usinage. Les auteurs utilisent une technique d'apprentissage supervisé pour la sélection des caractéristiques nommée redondance minimale, pertinence maximale. Les analyses démontrent que la méthode a permis d'éliminer une dizaine de

variables redondantes. Ensuite, un modèle est construit à partir de données historiques et les résultats sont utilisés pour produire un outil de surveillance pour la gestion de l'entreprise. Un modèle basé sur des règles est ensuite appliqué à la maintenance prédictive. Dans [81], un modèle de maintenance préventive est développé pour améliorer les politiques, en plus de présenter une architecture infonuagique pour un programme de maintenance prédictive et corrective avec détection en temps réel. Une étude de cas est réalisée pour analyser la durée de vie résiduelle des équipements d'un centre d'usinage. Un réseau neuronal artificiel est formé à partir de données historiques, puis utilisé pour la surveillance en temps réel. L'article [82] décrit une étude de fiabilité dans le contexte d'un fabricant de semi-conducteurs. L'objectif est de concevoir un modèle capable d'apprendre et d'associer des indicateurs à des défaillances potentielles et de déterminer des règles ou des modèles d'indicateurs ou de zones critiques. Les auteurs utilisent des réseaux bayésiens pour déterminer les distributions de probabilité dans la phase d'apprentissage, puis utilisent le réseau résultant pour apprendre les tendances menant à la défaillance. Les résultats suggèrent que le modèle pourrait être étendu à des applications de prédiction en temps réel.

Le tableau 4-4 présente divers outils de programmation et bibliothèques Python, largement utilisés par la communauté scientifique et dans le domaine de l'ingénierie RAMS. En particulier, les bibliothèques Reliability [83], Lifelines [84] et Scikit-survival [85] permettent plusieurs analyses statistiques pertinentes pour le RAMS, y compris des analyses paramétriques avec des distributions connues telles que Weibull, Gamma et Exponentielle, ainsi que des analyses de survie. La bibliothèque ProgPy [86] est une bibliothèque récemment développée par la NASA pour les applications PHM. La bibliothèque statsmodels [87] est une bibliothèque statistique générale, tandis que Scikit-Learn [88] est une bibliothèque ML générale contenant une multitude de méthodes.

TensorFlow [89] et Keras [90] sont parmi les bibliothèques les plus utilisées pour le développement de l'apprentissage profond. Enfin, NLTK [91] et Spacy [88] sont deux bibliothèques en libre accès qui permettent d'intégrer facilement et rapidement des outils de traitement du langage naturel.

Tableau 4-4 Bibliothèques Python pour l'ingénierie de la fiabilité.

Nom de la bibliothèque	Application	Fonction de la bibliothèque
Reliability	Fiabilité et analyse de survie	Modèles paramétriques (Weibull, exponentiel, lognormal, etc.), modèles non paramétriques (Kaplan–Meier, Nelson–Aalen, etc.), essais de durée de vie accélérés.
Lifelines	Analyse de Survie	Modèle des Risques proportionnels de Cox, Estimateur de Kaplan-Meier, Modèles Paramétriques de Survie
Scikit-survival	Analyse de survie	Arbre de survie, modèle d'ensemble pour l'analyse de survie, modèle des risques proportionnels de Cox, estimateur de Kaplan-Meier
ProgPy Python Prognostics Packages	Prognostics	
Statsmodels	Statistics	Linear Regression, Generalized Linear Model, ANOVA
Scikit-Learn	Machine Learning	Classification, Regressions, Clustering, Cross-Validation/Model Selection Methods
TensorFlow	Deep Learning	Artificial Neural Network, Recurrent Neural Network, Convolutional Neural Network
Keras	Deep Learning	High-level API built on top of Tensor Flow for easy use
NLTK	Natural Language Processing	Tokenization, Sentiment Analysis, Stemming, Part-of-Speech Tagging
Spacy	Natural Language Processing	Text Classifier, Transformer Models, Custom Trainable Pipeline, Named Entity Recognition

4.3.6 Discussion

Cette section permet de faire plusieurs observations concernant l'utilisation de l'apprentissage automatique dans le domaine de la fiabilité. Bien que cette revue ne répertorie que les résultats de quelques bases de données, elle permet d'avoir une vue d'ensemble de la situation. Le nombre de publications trouvées sur le sujet a été assez faible au cours des 5 dernières années (19 articles), bien qu'il y ait une tendance croissante pour les applications de l'intelligence artificielle. L'examen des sources de données donne une idée de la raison pour laquelle il n'y a pas plus de publications dans ce domaine. En effet, la plupart des publications appliquent l'intelligence artificielle à des données fictives, notamment dans le but de développer de nouvelles méthodologies. Les études de cas sont assez rares : moins d'un quart des publications. La difficulté d'obtenir des données opérationnelles de bonne qualité fait probablement partie du problème. Cependant, on peut s'attendre à ce que le développement de nouvelles méthodologies conduise à des études de cas sur des données réelles. Les données utilisées en fiabilité sont parfois difficiles à exploiter, compte tenu de leur complexité (données saisies manuellement, champs de texte libre, etc.). De plus, l'apprentissage automatique nécessite une grande quantité de données, ce qui signifie qu'il est nécessaire de disposer d'une grande base de données sur l'historique des travaux de maintenance, ce qui constitue en soi une contrainte considérable. Compte tenu de ces facteurs, cela justifie également pourquoi les approches d'apprentissage profond sont généralement préférées dans le domaine.

Chapitre 5 - Traitement automatique du langage naturel et applications à la fiabilité

Ce chapitre donne un aperçu des techniques de traitement du langage naturel et explore leur application pour améliorer l'analyse de fiabilité, en particulier dans les contextes de maintenance. En passant en revue les avancées récentes et les adaptations spécifiques au domaine, il met en évidence le potentiel croissant du TALN pour améliorer la prise de décision et la gestion des actifs.

5.1 Traitement automatique du langage naturel

Le traitement du langage naturel (TALN) est un sous-domaine de l'intelligence artificielle (IA) qui s'intéresse à l'interaction entre les ordinateurs et le langage humain. Les ordinateurs perçoivent le texte comme une série de caractères sans signification inhérente; il est donc nécessaire de convertir le texte en une représentation numérique qui capture les mots et leur signification. C'est l'objectif du TALN : permettre aux machines de comprendre le langage d'une manière similaire à celle des humains [10]. L'histoire du traitement du langage naturel (TALN) remonte aux années 1940, avec le développement des premiers systèmes de traduction linguistique [92]. Au cours de la dernière décennie, l'essor des modèles d'apprentissage profond a été étendu aux applications de TALN, avec des techniques telles que les réseaux neuronaux récurrents et les modèles de transformer. Il existe plusieurs types de tâches gérées par le TALN.

La classification de textes, comme son nom l'indique, est avant tout un problème de classification, tel que défini dans le cadre de l'apprentissage supervisé. Ainsi, un texte peut être classé dans des catégories prédéterminées, par exemple dans la détection de pourriels (spam/pas spam), la classification des sujets, également connue sous le nom de

modélisation des thèmes, etc. L'analyse des sentiments est un problème de classification en soi. En effet, elle vise à analyser un texte et, en fonction du vocabulaire utilisé (polarité), à déterminer si le texte est positif, négatif ou neutre. Cette application est très utile pour l'analyse du service client ou des médias sociaux. La reconnaissance d'entités nommées vise à identifier des éléments dans un texte, par exemple des noms d'entreprises, des dates, des noms de personnes, etc. Une autre tâche importante du TALN est la génération de textes. La capacité d'un système à générer des textes est extrêmement utile, notamment pour la traduction de documents (traduction automatique), le résumé de texte et la correction grammaticale. La génération de texte est également utilisée pour répondre à des questions, notamment dans la création de «*Chatbots*», d'assistants virtuels, etc. [93].

5.2 Analyse des ordres de travail de maintenance

Cette section se concentre sur l'analyse de diverses publications dans le domaine de la fiabilité et de la maintenance qui exploitent les techniques de TALN. Tout d'abord, une étude démontre comment les approches TALN de base peuvent aider à traiter les données textuelles. Elle présente une étude de cas qui analyse les données d'un système de transport ferroviaire. Dans ce cas, l'auteur utilise une approche statistique, en comptant l'occurrence des mots, pour classer rapidement le type de défaut lié à une demande d'inspection d'urgence [94]. Dans un autre ouvrage, une approche fondée sur des règles est proposée pour nettoyer les données de l'historique de maintenance, en tirant parti des données textuelles contenues dans les ordres de travail [95]. Une approche originale propose de combiner les connaissances des experts en maintenance avec des techniques d'IA avancées, en particulier des techniques de traitement du langage naturel [96]. En utilisant des techniques d'annotation textuelle et des outils de traitement du langage

naturel, la méthode permet d'extraire des informations importantes pour la gestion de la maintenance, en intégrant les humains dans le processus d'étiquetage des données. De même, cette méthodologie est utilisée pour développer des indicateurs de performance clé à partir du texte libre des ordres de travail de maintenance [97]. Une autre étude compare deux méthodes d'extraction de mots-clés à partir de bons de travail, dans le cas d'équipements miniers. Une méthode d'extraction est basée sur des règles spécifiées par des experts, tandis que l'autre méthode est axée sur les données et utilise une technique d'étiquetage issue de l'intelligence artificielle. Une analyse de survie compare ensuite la durée médiane de défaillance des actifs pour chaque méthode d'extraction [98]. Une méthode d'exploration de texte est proposée pour analyser la maintenance des systèmes de chauffage, de ventilation et de climatisation (CVC). Les données analysées proviennent de la maintenance des systèmes de deux bâtiments d'un campus. L'auteur propose d'encoder les textes à l'aide de la méthode «*Document Term-Matrix*» et d'extraire, par regroupement hiérarchique, les mots-clés liés aux événements de défaillance. Le lexique développé par le regroupement est ensuite utilisé pour développer un ensemble de règles d'association en vue d'une future analyse des requêtes [99].

Un processus d'évaluation est proposé pour extraire les termes et concepts importants des textes de maintenance. Les termes sont ensuite associés à des indicateurs de performances clés, tels que la durée des travaux de réparation [100, 101]. Un premier article décrit un modèle d'inférence de la durée des activités de maintenance, basé sur les descriptions des tâches de maintenance par les techniciens. L'auteur utilise un logiciel (NESTOR [102]) pour annoter les données textuelles et extraire les informations qui s'y rapportent (solution, problème, objets). Ces trois variables sont ensuite utilisées pour prédire le temps de maintenance, sur une échelle de temps catégorielle [100].

Dans un article des mêmes auteurs, trois études de cas appliquées respectivement aux données d'actifs de l'industrie automobile, d'un fournisseur de composants automobiles et du domaine de l'éclairage sont présentées pour tester la méthodologie. Les auteurs utilisent également NESTOR pour l'annotation des données textuelles, mais utilisent «*word2vec*» pour les encodages. Une fois le texte préparé dans un pipeline TALN, un modèle de prédiction des indicateurs de performance est développé [101]. Dans les deux cas, ils comparent les arbres de décision aux réseaux neuronaux pour la prédiction [100, 101].

Une autre étude vise à automatiser le processus de répartition des tâches de maintenance, sur la base des demandes/problèmes rencontrés par les utilisateurs. Elle se concentre sur la maintenance des établissements de soins de santé. À partir des demandes des utilisateurs, un modèle de classification est construit pour trier les types d'intervention et attribuer automatiquement les tâches aux agents d'entretien. En utilisant des techniques TALN pour la classification de textes, l'auteur teste plusieurs architectures de réseaux neuronaux convolutifs (RNC), en utilisant «*FastText*» pour la représentation vectorielle des mots. Le modèle atteint une précision de 78 % [103]. D'un point de vue différent, il est suggéré que, bien que le traitement du langage naturel progresse rapidement, les techniques actuelles ne doivent pas être appliquées aveuglément; les modèles doivent être adaptés à l'analyse technique [11, 15]. Les auteurs s'efforcent donc à démontrer la nécessité d'adapter le TALN, soulignant les différences entre les textes utilisés pour l'entraînement des modèles disponibles et les textes techniques. À cette fin, ils proposent le traitement du langage technique, une méthodologie permettant d'adapter le TALN [15]. Dans le contexte de la maintenance, par exemple, de nombreuses informations sont saisies dans des champs de texte libre, mais elles sont sous-utilisées en raison de leur forme. Il s'agit donc d'un bon contexte pour tester cette approche. Au travers de ces publications, ils décrivent un cadre pour adapter les modèles de TALN au contexte des textes techniques.

Ils démontrent comment ce nouveau cadre peut être adapté à différents contextes et surmonter le problème de la quantité et de la qualité des données. Ils utilisent ensuite les textes contenus dans les ordres de travail de maintenance comme exemple. Les résultats suggèrent que l'adaptation du TALN dépend fortement de l'inclusion humaine dans la démarche, en particulier pour l'annotation manuelle des textes [11].

Bikaun & Hodkiewicz [104] ont développé une méthodologie pour détecter les événements de fin de vie en utilisant le traitement du langage naturel. Le pipeline adapte un modèle de traitement du langage naturel au domaine de la maintenance. En utilisant les ordres de travail et l'éllicitation des experts, la première étape consiste à extraire des mots-clés liés aux événements de fin de vie. Ensuite, un second pipeline intégrant ces mots-clés est utilisé pour classer les ordres de travail afin de déterminer s'il s'agit d'événements de fin de vie. Les résultats sont utilisés pour produire des estimations statistiques du temps moyen entre pannes (MTBF) pour les actifs en service, en particulier les équipements miniers [104].

Dans [105], une analyse des données de maintenance des systèmes de chauffage, de ventilation et de climatisation (CVC). L'objectif de l'article est de présenter des stratégies de détection des problèmes de qualité des données. Des techniques d'analyse exploratoire des données (AED) sont utilisées pour examiner les données. Une tentative est faite pour déterminer l'impact de la qualité/non-qualité des données sur les indicateurs de performance de la maintenance, en développant une technique comparative basée sur l'analyse de survie.

Dans [106], ils analysent les données de maintenance pour identifier les actifs immobiliers problématiques. Les auteurs proposent une méthode complète de prétraitement et de classification des données textuelles de maintenance. L'étude vise

également à évaluer la performance des actifs et à développer des cartes de visualisation pour les décideurs. La méthodologie s'appuie sur des outils TALN pour nettoyer et préparer le texte. Elle utilise également la classification de texte pour prédire l'étiquette d'un ordre de travail. Pour ce faire, ils utilisent une approche basée sur l'occurrence des mots.

Un autre article compare les méthodes de traitement du langage naturel à l'aide d'une étude de cas portant sur une entreprise de fabrication [107]. À l'aide de textes contenus dans des bons de travail de maintenance, ils tentent de déterminer le type de problème en cause. L'objectif est de comparer les méthodes traditionnelles de traitement du langage naturel avec les modèles de «*Transformer*», en particulier CamemBERT, un modèle de langage développé pour les textes français. Les résultats des modèles de classification de textes montrent que l'architecture de «*Transformer*» est beaucoup plus performante que les méthodes de sacs de mots.

Dans une autre étude, les chercheurs tentent de développer une technique de ré-entraînement du modèle BERT à un domaine spécifique, soit la maintenance, en utilisant le «*Transformer-Based Sequential Denoising Auto Encoder*» (TSDAE), une technique d'apprentissage non supervisée. En s'entraînant sur des corpus spécifiques à un domaine, le modèle peut trouver des similitudes sémantiques entre les mots. Les données utilisées sont des ordres de travail de maintenance provenant d'un ensemble de données publiques sur les excavateurs miniers [108]. Les mêmes auteurs ont combiné le TALN avec des techniques de raisonnement à partir de cas (RàPC), ou «*Case-Based Reasoning*» (CBR), pour la recherche d'informations sur l'historique de la maintenance. Une étude de cas appliquée au domaine de l'aviation compare les performances des architectures de transformeur, avant et après l'application du raisonnement à partir de cas [109]. Comme

dans leurs travaux précédents, un modèle BERT est affiné à l'aide de la technique TSDAE. Dans ce cas, le RàPC est utilisé pour affiner la connaissance du modèle, en stockant des cas similaires pour améliorer les encodages de mots. En générant des requêtes, les modèles sont testés sur leur capacité à retrouver des cas similaires (sémantiques) à la requête. La méthode augmente la précision du modèle affiné de 37 % par rapport au modèle préentraîné standard. Comme dans [108], un modèle BERT est affiné à l'aide de la technique TSDAE. Dans ce cas, le RàPC est utilisé pour affiner la connaissance du modèle, en stockant les cas similaires pour améliorer l'encodage des mots. En générant des requêtes, les modèles sont testés sur leur capacité à retrouver des cas similaires (sémantiques) à cette requête. La méthode augmente la précision du modèle affiné de 37 % par rapport au modèle préentraîné disponible.

L'amélioration de la qualité des données de maintenance est également cruciale pour garantir la fiabilité des modèles. Une approche utilisant RoBERTa et des techniques avancées de NLP a permis d'automatiser le nettoyage des données textuelles en réduisant les erreurs et les incohérences dans les historiques de maintenance [110]. Une autre approche récente combine la classification à l'aide de BERT et un modèle SpERT (*Span-based Joint Entity and Relation Extraction with Transformer Pre-training*) pour l'extraction d'informations. Dans un premier temps, la classification sert à éliminer les textes non pertinents à l'analyse, puis, un second modèle sert à extraire les entités ainsi que les relations causales pertinentes dans les actions de maintenance [111].

Ensuite, l'intégration d'un modèle d'Allocation Latente de Dirichlet à un modèle d'attention sujet-contexte (TCAM) permet d'extraire et de structurer les caractéristiques des textes de pannes. L'objectif est de faciliter le diagnostic automatique des défaillances par la récupération de cas similaires [112]. Dans un contexte hospitalier, l'affectation des

ordres de travail repose souvent sur des processus manuels inefficaces. Une approche combinant l'apprentissage supervisé et non supervisé a permis l'automatisation de cette tâche. Dans un premier temps, un algorithme de modélisation des sujets (LDA) identifie automatiquement les objets de maintenance courants et quatre approches de classification sont testées pour attribuer les interventions aux techniciens [113]. Toujours dans le contexte de la gestion d'infrastructure publique, une étude évalue l'application de classificateurs pour prédire la priorité des demandes d'entretien envoyées à la maintenance. En outre, l'effet du prétraitement du texte est étudié pour quantifier ses impacts sur la performance des classificateurs [114].

D'autres travaux proposent l'utilisation de l'IA générative afin de faire une classification automatique des modes de défaillance. Cette étude explore l'impact des « prompts » sur la performance et l'efficacité des grands modèles de langage (LLM) pour détecter et classer les modes de défaillance [115]. L'hétérogénéité et l'imprécision des descriptions d'actions de maintenance compliquent leur analyse. Dans cette perspective, une approche visant à extraire l'information en utilisant la reconnaissance d'entités nommées est appliquée au domaine automobile. Le travail propose d'identifier les entités à l'aide de l'étiquetage morphosyntaxique, pour identifier les termes liés aux verbes d'actions et l'objet visé par la réparation [116].

Les études analysées soulèvent d'importants défis quant à l'utilisation du TALN pour la maintenance. La variabilité des textes, tant dans les formats que dans l'utilisation de langages techniques, rend difficile l'application de modèles génériques, nécessitant le développement de dictionnaires spécialisés et de techniques d'adaptation de modèles [116,117]. Il est donc impératif d'améliorer la qualité des textes avant leur exploitation par des techniques d'IA. Bien que ce soient des défis majeurs, les travaux récents montrent des

avancées significatives dans l'extraction et la classification des informations, l'affectation des tâches et la recherche sémantique.

5.3 Principes du traitement du langage naturel

Cette section illustre les principes de base du TALN, s'intéressant particulièrement à l'annotation, la préparation et la représentation numérique des textes.

5.3.1 Annotation textuelle

Tout d'abord, il est essentiel de définir ce qu'est l'annotation textuelle et comment elle est couramment utilisée dans le traitement du langage naturel. L'annotation est une pratique de traitement du langage naturel qui consiste à ajouter automatiquement ou manuellement des informations en attribuant des étiquettes ou des attributs spécifiques à différents éléments du texte (mots, expressions, phrases, etc.). L'annotation joue un rôle crucial dans le TALN, car elle permet de créer des ensembles de données étiquetés qui sont utilisés pour créer des modèles d'apprentissage supervisés. L'ajout d'annotations permet aussi aux algorithmes d'apprentissage automatique de traiter les données plus efficacement [118]. En outre, selon [11, 15], l'annotation manuelle, c'est-à-dire avec des humains supervisant le processus, est la clé de l'adaptation du TALN à des domaines spécifiques tels que l'analyse des ordres de travail de maintenance. Heureusement, les progrès dans le domaine sont tels que de nombreuses tâches d'annotation peuvent être effectuées automatiquement, en utilisant l'apprentissage par transfert. En bref, l'apprentissage par transfert est une technique d'IA qui utilise des modèles préentraînés (modèle construit sur un grand ensemble de données pour une tâche spécifique) afin d'accélérer le processus de traitement. En utilisant un réseau neuronal préentraîné, les analystes n'ont plus qu'à affiner le modèle avec des données pour leur application spécifique [119]. En fait, plusieurs

groupes de travail se sont concentrés sur l’annotation de grands corpus publics, par exemple la «*Penn Treebank*» qui comprend des annotations syntaxiques et de parties de discours [5], ou le Universal Dependencies Project qui développe des annotations de banques d’arbres multilingues [120, 121]. Aujourd’hui, plusieurs modèles sont disponibles gratuitement pour les tâches d’annotation, telles que l’étiquetage morphosyntaxique et la reconnaissance des entités nommées, directement disponibles dans des bibliothèques de code prêtes à l’emploi. Le tableau 5-1 présente différents programmes Python qui offrent une variété d’outils TALN, y compris des composants d’annotations préentraînés.

Tableau 5-1 Bibliothèques Python pour les applications de traitement du langage naturel

Bibliothèque	Outils	Auteurs/Organisations
NLTK	Tokenisation, Lemmatisation, Étiquetage, Analyse syntaxique, Analyse de sentiment, Corpus	Steven Bird, Edward Loper et Ewan Klein [91]
SpaCy	Tokenisation, Étiquetage POS, Reconnaissance d’entités nommées, Analyse de dépendance, Classification de texte, Détection de similarité	Explosion AI [122]
Gensim	Modélisation de thématique, Représentation Vectorielle, Analyse de similarité de documents	Radim Řehůřek [123]
TextBlob	Analyse de sentiment, Étiquetage POS, Extraction de syntagmes nominaux	Siddharth Roy [124]
Stanford CoreNLP	Tokenisation, Étiquetage POS, Reconnaissance d’entités nommées, Analyse de dépendance, Analyse de sentiment	Stanford NLP Group [125]
AllenNLP	Classification de texte, Reconnaissance d’entités nommées, Analyse de dépendance	Allen Institute for AI [126]
Transformers	Modélisation du langage, Classification de texte, Réponse à des questions	Hugging Face [127]

5.3.1.1 Étiquetage Morphosyntaxique

Cette sous-section définit deux méthodologies d'annotation couramment employées pour le TALN. L'étiquetage morphosyntaxique, ou «*Part-of-Speech Tagging*» (POS) est une technique d'annotation qui catégorise les jetons d'un corpus en fonction de leur fonction grammaticale, par exemple s'il s'agit de noms, de verbes, d'adverbes, d'adjectifs, etc. Cette tâche permet la désambiguïsation du texte, c'est-à-dire que dans certains cas, un mot peut avoir plusieurs significations [93, 128, 129]. En attribuant la bonne étiquette au bon contexte, les ambiguïtés lexicales peuvent être éliminées. De cette manière, les informations textuelles peuvent être enrichies à différentes fins de modélisation.

Tout comme le POS, la reconnaissance des entités nommées (NER) est une technique d'annotation de texte qui repose sur la classification du texte. Bien que la NER ne soit pas limitée à la classification des jetons, elle s'applique également aux groupes de mots (phrases, blocs, etc.). Des modèles de NER préentraînés sont disponibles pour la classification selon des classes communes afin d'identifier, par exemple, une personne, un lieu (ville, pays, etc.) ou des organisations (par exemple, Apple, Google, Microsoft, etc.). La NER peut simplifier l'analyse des textes en reliant des mots ou des groupes de mots à des concepts connus [10, 15, 129, 130].

5.3.2 Préparation du texte

Une tâche importante du TALN consiste à décomposer le texte en éléments plus petits à traiter pour en extraire le maximum de connaissances. La délimitation de phrase ou «Sentence Boundary Detection» (SBD) consiste à identifier les limites des phrases dans un texte donné afin de les segmenter en phrases individuelles. La segmentation de texte «*Chunking*», également connue sous le nom d'analyse syntaxique, implique

l'identification des groupes de mots à l'aide de l'étiquetage morphosyntaxique. L'objectif est d'identifier et de regrouper les mots qui forment une unité grammaticale cohérente, par exemple en groupe nominal, en groupe verbal, etc [130]. La tokenisation est le processus de décomposition d'un texte en unités plus petites, généralement des mots ou même des ponctuations, appelées jetons «*Tokens*» [131]. En d'autres termes, la délimitation de phrase sépare le texte en phrases, la segmentation sépare les phrases en structures grammaticales et la tokenisation sépare les mots, les ponctuations, etc. en jetons.

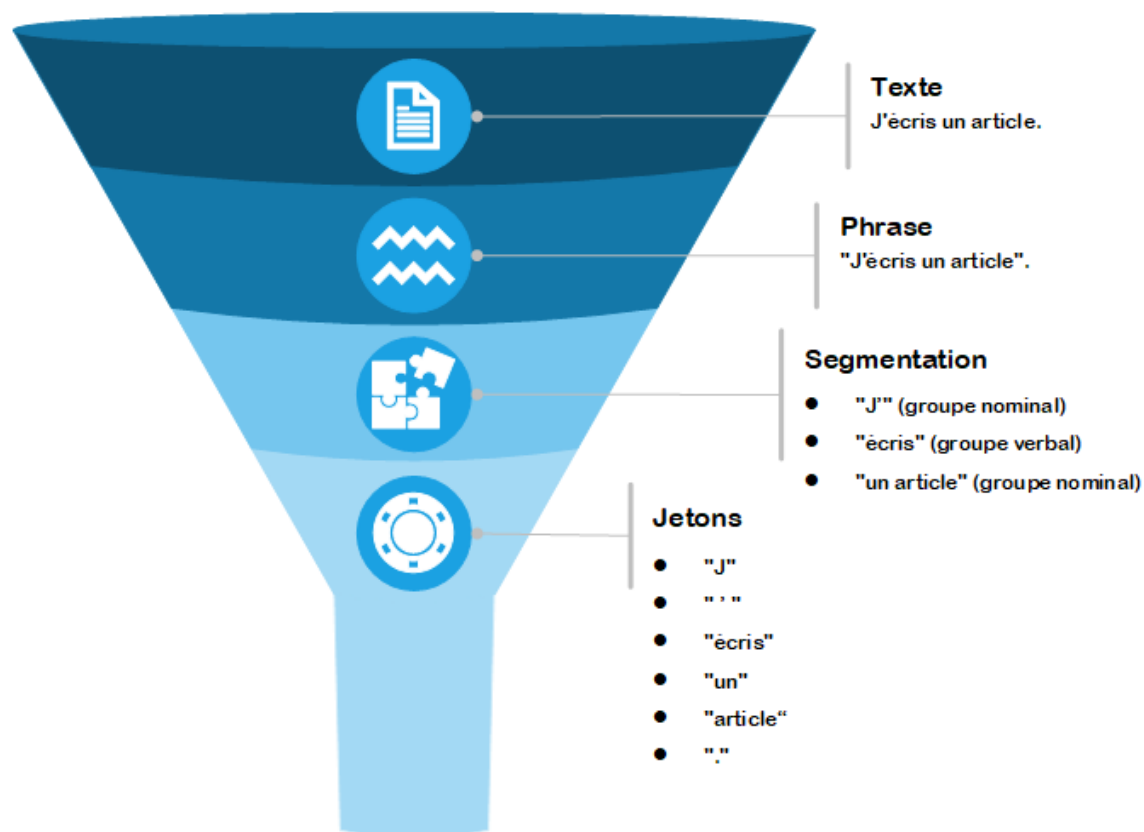


Figure 5.1 Préparation du texte

La plupart des textes comportent à la fois des majuscules et des minuscules. Pour simplifier l'analyse des textes, il est utile de normaliser la casse de tous les mots en majuscules ou en minuscules [128, 131]. Par commodité, le texte est converti en minuscules la plupart du temps. Une autre étape du prétraitement consiste à supprimer

les mots vides. Les mots vides sont des mots très fréquents dans une langue donnée, souvent des articles et des conjonctions (un, une, le, la, et, etc.) qui n'ajoutent que peu de sens à un texte. Les listes de mots vides sont souvent incluses dans les bibliothèques de TALN, et il est courant pour de nombreuses applications de supprimer ces mots [10, 128]. Cependant, dans certains contextes, la suppression de tous les mots vides peut altérer le sens d'une phrase [11, 15], et dans certains cas, elle n'améliore pas vraiment les performances du modèle [93]. Il serait donc judicieux de sélectionner manuellement les mots vides, en particulier pour les applications d'adaptation à un domaine. La racinisation et la lemmatisation sont des techniques de TALN qui permettent de réduire les mots à une forme simplifiée, soit à la forme racine ou canonique. Bien que ces deux techniques aient le même objectif, elles diffèrent par la manière dont elles parviennent à la forme de base du mot. La racinisation est plus simple, puisque cela consiste à supprimer le suffixe ou le préfixe des mots. Par exemple, chanteur, chanteuse, chanteuses peuvent être convertis en leur forme racine chant. D'autre part, la lemmatisation est plus complexe et met en correspondance les mots en utilisant l'analyse morphologique ou un dictionnaire pour obtenir la forme canonique, par exemple, chanteur, chanteuse et chanteuses, deviendraient chanteur [15, 93, 128].

5.3.3 Représentation des données textuelles

L'un des principaux défis du traitement du langage naturel consiste à représenter le texte sous une forme numérique pouvant être interprétée par une machine. Les approches les plus simples sont l'encodage par sac de mots et l'encodage à chaud. L'encodage à chaud repose sur un encodage binaire d'un vecteur de longueur équivalente à la taille du vocabulaire. L'encodage par sac de mots, quant à lui, consiste à créer un vecteur dans lequel chaque texte est représenté par l'occurrence de mots dans le

vocabulaire [10,131]. Ces représentations de données sont parfois appelées représentations locales. En revanche, les représentations distribuées constituent un moyen plus efficace de représenter les données en produisant des vecteurs de moindre dimension grâce à un processus d'apprentissage [129]. Ces méthodes de représentation vectorielle des mots sont fortement basées sur des algorithmes d'apprentissage profond. Parmi les méthodes les plus répandues, on peut citer «*word2vec*», «*GloVe*», «*FastText*», etc. Les développements impressionnants de la recherche en IA sont largement dus au développement du TALN, et plus particulièrement aux progrès réalisés dans l'application de l'apprentissage profond à l'apprentissage de représentations. L'article «*Attention is all you need* » [132] a présenté l'architecture Transformer, surpassant les modèles de réseaux neuronaux convolutifs (RNC) et de réseaux neuronaux récurrents (RNR) avec la capacité de capturer les relations entre les mots et une plus grande capacité à s'adapter aux contextes de ceux-ci. En outre, les Transformers sont plus puissants, car leur structure permet des calculs parallèles [129, 133]. Le tableau 5-2 présente les méthodes d'encodage de textes les plus courantes.

Tableau 5-2 Méthodes de Word Embeddings

Méthode	Description	Année	Organisation
Word2Vec	Un type de modèles (Skip-gram et CBOW) qui apprennent des représentations en prédisant le contexte des mots	2013	Google [134]
GloVe (Global Vectors for Word Representation)	Combine les statistiques de cooccurrence avec la factorisation globale des matrices pour apprendre les représentations des mots	2014	Université de Stanford [135]
fastText	Étend Word2Vec en représentant les mots sous forme de n-grams de caractères, capturant ainsi l'information de sous-mot	2016	Facebook AI Research [136]
BERT (Bidirectional Encoder Representations from Transformers)	Un modèle de langage basé sur des Transformers qui apprend des représentations contextualisées	2018	Google [137]
ELMO (Embeddings from Language Model)	Représentations contextualisées générées par un modèle de langage bidirectionnel	2018	Allen Institute for Artificial Intelligence [138]
GPT (Generative Pre-trained Transformer)	Un modèle de langage génératif basé sur des Transformers qui apprend des représentations contextualisées	2018	OpenAI [139]
Gecko	Un modèle compact de représentations contextualisées, basé sur les Grands Modèles de Langues	2024	Google [140]
LLM2Vec	Un modèle de représentations contextualisées, basé sur les Grands Modèles de Langues	2024	MILA [141]

Chapitre 6 - Démarche de conception de la méthode

Ce chapitre présente la méthodologie générale employée dans ce projet, qui a pour but d'améliorer la qualité des données de maintenance, par l'utilisation de l'intelligence artificielle et en analysant les descriptions des travaux d'entretien. La figure 6.1 illustre les étapes de la méthodologie utilisée pour ce projet. Pour chacune de ces étapes, les procédures utilisées dans la littérature sont explorées, afin de cibler les techniques applicables au présent contexte. De plus, certains manques ont été identifiés, ce qui justifie l'élaboration de nouvelles méthodes, en particulier concernant le nettoyage du texte.

6.1 Préparation des données

Dans un premier temps, les données brutes issues des systèmes d'entreprises doivent subir un prétraitement avant la modélisation.

6.1.1 Préparation des tables

La première étape est de croiser les tables d'ordres de maintenance, qui contiennent les informations sur les travaux effectués, avec les avis de maintenance, qui contiennent les informations et les descriptions des inspections et autres activités qui ont mené à l'émission d'un ordre de travail. Un filtre est appliqué afin de sélectionner une seule famille d'équipement, pour qu'il y ait concordance entre les données de maintenance et l'extraction des AMDE aux prochaines étapes. De plus, les équipements rebutés et hors service sont aussi filtrés. Les colonnes qui contiennent plus de 50 % de données manquantes sont exclues des analyses. Ensuite, les entrées qui ne contiennent pas de champs texte sont supprimées, n'apportant aucune valeur à la modélisation. Enfin, il existe plusieurs champs de texte pour consigner l'information; certains issus des ordres

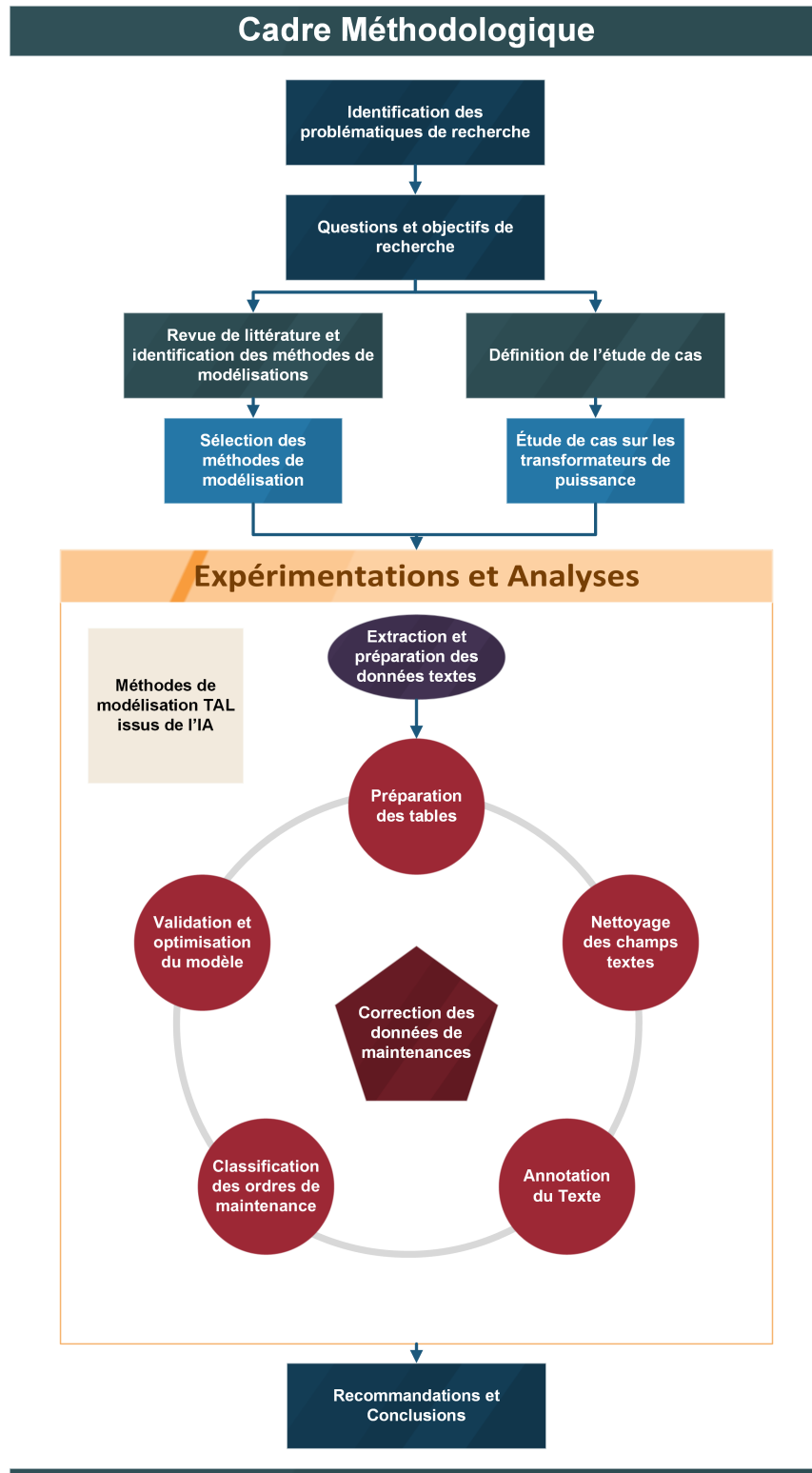


Figure 6.1 Cadre méthodologique de la thèse

et d'autres des avis de maintenance. Ils sont donc concaténés afin de maximiser la complétude d'information tirée des textes.

6.2 Nettoyage des champs textes

Une fois les tables de données de maintenance prêtes à l'analyse, il faut tout de même effectuer une certaine préparation des champs textes. En effet, les textes contiennent parfois des caractères spéciaux, des codifications et d'autres éléments non désirables pour la modélisation du texte, qui devront être corrigés. La section 8.2 présentera les différentes fonctions appliquées et l'effet sur les textes de maintenance.

6.2.1 Prétraitement des textes

La première étape de nettoyage du texte est le prétraitement. Il s'agit d'opérations simples pour éliminer le bruit inutile contenu dans le texte, qui peut nuire aux algorithmes de TALN. Le tableau 6-1 présente les approches couramment utilisées dans des applications similaires. Dans le cadre du projet, plusieurs de ces méthodes sont testées afin d'optimiser l'analyse des données. Tout d'abord, une normalisation de la casse est appliquée en convertissant l'ensemble du texte en minuscules. Cette approche facilite le traitement linguistique, notamment lors de l'utilisation des techniques de vectorisation. Étant donné que le texte présente peu de dates et d'heures, la normalisation n'est pas requise. Les symboles et ponctuations superflus sont éliminés pour minimiser le bruit dans les données. L'élimination des mots vides ou «*stop words*» est effectuée de manière sélective. Par exemple, les négations sont conservées, car elles jouent un rôle crucial dans l'interprétation du contexte. De plus, les codes numériques hérités d'anciens systèmes sont supprimés ou normalisés en fonction de leur signification, comme la conversion de

Tableau 6-1 Méthodes de prétraitement du texte

	Normalisation		Suppression				Autres
Réf	Casse	Dates et heures	Symboles spéciaux	Ponctuation	Mots vides	Chiffres	Correction des fautes
[116]	*						distance vectorielle
[113]			*	*	*		
[114]			*	*	*	*	
[117]	*						
[111]							python
[112]					*		
[108]	*						
[109]	*						
[104]	*	*	*				
[106]	*			*	*		python
[101]							logiciel
[11]				*		*	python
[15]				*	*	*	
[100]							logiciel
[98]							manuelle
[99]				*	*	*	
[96]							
Total (17)	6	1	3	6	6	4	7
%	35 %	6 %	18 %	35 %	35 %	24 %	41 %

«T5» en «transformateur 5». Enfin, différentes techniques de normalisation des mots sont testées, afin de trouver la meilleure façon de diminuer la variabilité des mots dans le texte.

6.2.2 Normalisation des termes

En premier lieu, les approches de distance vectorielle pour trouver des synonymes potentiels sont appliquées et différentes bibliothèques python sont testées pour la correction des fautes d'orthographe. Dans le contexte du projet, l'emploi de logiciels dédiés au TALN pour l'annotation et la correction des fautes n'est pas envisageable. De la même façon, la correction des fautes et la recherche de synonymes manuelle sont écartées, puisqu'il serait difficile d'appliquer cette méthode de façon efficace sur l'ensemble des actifs d'HQ. L'échec dans l'application de ces techniques de normalisation, telles que présentées au chapitre 8, force à développer une technique de normalisation, qui sera présentée au chapitre 7. En outre, un dictionnaire de normalisation est élaboré et intégré dans les pipelines afin de standardiser certaines expressions ou abréviations récurrentes.

6.3 Annotations et extraction des connaissances

Il existe plusieurs techniques d'annotations du texte, telles que présentées à la section 5.3.1. Ici, un modèle de reconnaissance d'entités nommées est développé. Dans une démarche de conception d'un modèle d'IA, il est nécessaire d'avoir au préalable des données d'entraînement. Il existe plusieurs façons de générer ces données. Par exemple, il est possible d'extraire les verbes d'actions de maintenance et les objets impliqués dans le travail en utilisant l'analyse morphosyntaxique comme méthode d'étiquetage [116]. Sinon, les autres techniques observées dans la littérature sont les méthodes manuelles ou l'étiquetage à l'aide d'un logiciel spécialisé [96, 100, 111, 142]. La première option est envisageable, mais les catégories ne sont pas assez précises pour le contexte actuel.

Tout comme pour la normalisation des termes, les méthodes manuelles ou assistées par un logiciel sont écartées. La méthode choisie, détaillée à la section 7.2, consiste à employer les fonctionnalités de correspondance de texte disponibles dans les librairies python pour le TALN.

6.4 Classification des ordres de maintenance

Dans ce travail, il est proposé de corriger les données de maintenance par la reclassification des ordres de maintenance. Pour ce faire, il est convenu de concevoir un modèle de classification à partir du texte, pour assigner une étiquette selon le type de maintenance préventive ou corrective. Pour ce faire, il faut convertir le texte en représentation vectorielle, permettant ensuite d'utiliser les techniques d'apprentissage supervisé sur ces vecteurs. Des méthodes de représentation basées sur la fréquence ont été appliquées avec succès [98, 99, 107, 114], mais des méthodes plus efficaces sont disponibles. De même, les méthodes de représentation statique qui permettent tout de même de bons résultats [96, 98, 101, 103, 104, 112, 142] sont maintenant remplacées par des méthodes contextuelles basées sur les architectures *Transformers* [107–109, 111, 116, 117, 142].

Ensuite, plusieurs types de méthodes de classification sont employées dans le domaine de la maintenance, notamment des méthodes de base d'apprentissage supervisées. Parmi celles-ci, on retrouve des techniques par arbres de décision, les machines à vecteurs de support, la régression logistique, etc. [107, 113, 114]. Étonnamment, il n'y a que quelques applications utilisant des approches d'apprentissage profond [111, 114]. Les méthodes manuelles et/ou basées sur des règles sont très répandues [96, 98, 100, 104, 106], de même que celles tirant profit de logiciels dédiés au TALN [96, 101, 111].

Pour ce projet, il a été convenu d'utiliser les méthodes de représentations contextuelles, qui sont spécifiquement entraînées sur du texte français. Pour la classification, des approches d'apprentissage supervisé seront comparées à des méthodes d'apprentissage profond. Encore une fois, les techniques manuelles et l'utilisation de logiciels sont des solutions qui sont écartées.

En résumé, ce chapitre a servi à recenser les techniques potentiellement applicables à chaque étape du projet. Les méthodes qui ne sont pas faisables et celles qui sont aujourd'hui dépassées par des algorithmes plus avancés ont été écartées. Ainsi, les techniques appropriées ont été sélectionnées. Par contre, pour l'étape de la normalisation du texte, aucune méthode trouvée ne convient au contexte, soulignant le besoin de développer un nouveau processus. Le prochain chapitre décrit en détail les modalités implémentées pour chaque étape de modélisation.

Chapitre 7 - Méthode proposée pour la correction des données de maintenance

L'analyse de la littérature a montré la nécessité d'un travail préliminaire approfondi, afin de pouvoir utiliser les méthodes les plus récentes en matière d'intelligence artificielle. Ce qui semble également commun à de nombreux travaux étudiés est la nécessité de consulter des experts, soit pour établir des règles d'extraction, soit pour annoter et identifier les termes pertinents pour la maintenance. L'un des avantages de ce travail est la réduction de l'effort lié à ces étapes de prétraitement en réutilisant la documentation produite dans le cadre d'une approche d'analyse des modes de défaillance et de leurs effets (AMDE). Cette méthode, largement utilisée dans le domaine de la fiabilité, consiste à subdiviser le système puis à identifier ses modes de fonctionnement et les configurations correspondantes. Ces informations sont ensuite compilées dans un tableau (organisé par mode de fonctionnement) qui décrit les modes de défaillance associés et leurs effets sur les autres entités du système [1, 31, 43]. Les informations sur le système, ses composants, ses modes de défaillance, etc. sont ainsi accessibles de manière claire et concise, ce qui facilite leur réutilisation dans une approche TALN. Cette section permettra de décrire les différentes étapes du processus de modélisation, de la préparation des données jusqu'au développement des modèles TALN.

7.1 Normalisation du texte

Tel que décrit à la section 6.1, il convient d'établir une méthodologie de normalisation adaptée au contexte du projet. En effet, les techniques manuelles et/ou supportées par un logiciel automatique sont écartées. Ce faisant, cette section décrit la démarche employée pour réduire la variabilité du texte.

La démarche appliquée est relativement simple. Tout d’abord, le texte des données de maintenance doit être extrait et prétraité, tel que défini au chapitre précédent. Ensuite, à partir des fiches de l’AMDE, il faut procéder à l’extraction des données pour les :

- Équipements
- Composants Primaires
- Composants Secondaires
- Mécanisme de dégradation
- Modes de défaillances
- Etc.

En dressant une liste de tous ces éléments, cela permet d’extraire les mots les plus importants du corpus. À partir de cela, un lexique est construit à partir de mots ou de groupes de mots issus de ces termes. Certains groupes de mots peuvent être décomposés de façon à avoir le plus de configurations de mots possible. Par exemple, ‘transformateur’ et ‘transformateur de puissance’ désignent potentiellement la même chose, il est donc important de conserver les deux configurations dans le lexique. Une fois ces étapes terminées, le processus de normalisation du texte des AMDE se déroule par itération, comme le montre la figure 7.1

La première étape du processus de normalisation est d’identifier les synonymes, à partir de la liste extraite des AMDE. Un élément crucial d’un pipeline TALN typique est la tokenisation, qui consiste à diviser le texte en éléments discrets, généralement des mots, des termes, des phrases, etc. Dans le contexte des données techniques, un mot peut être désigné de plusieurs façons différentes, ce qui génère autant de jetons que de variantes. Dans ce cas, un modèle de ML percevrait chaque instance comme étant différente, ce qui rendrait des tâches simples telles que la classification et le regroupement extrêmement difficiles. La méthode développée ici remédie rapidement à ce problème, en identifiant

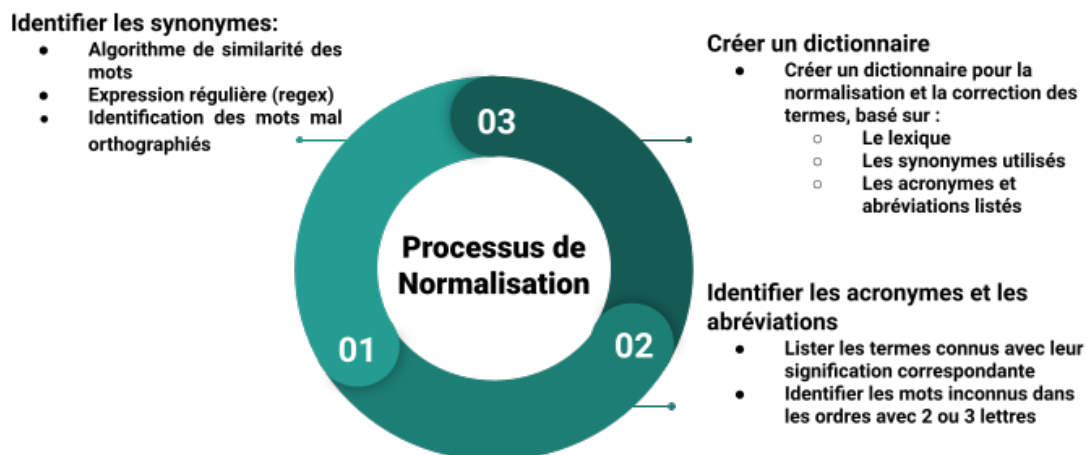


Figure 7.1 Processus de normalisation

les différentes variantes de mots, en se concentrant sur les éléments les plus essentiels des textes de maintenance. Le lexique basé sur l'AMDE met en évidence les mots-clés et les phrases liés aux principaux éléments des systèmes et des sous-systèmes, ainsi que les mots-clés et les phrases liés aux signes potentiels de dégradation et de défaillance. Les techniques qui peuvent être utilisées sont la correspondance par expression régulière, qui permet de faire une recherche avec une partie (suffixe par exemple) d'un mot, la recherche de synonymes par algorithme d'intelligence artificielle, ou la recherche par algorithme de correction orthographique. Ensuite, les acronymes et les abréviations dans le texte sont répertoriés. Si une documentation est disponible dans l'organisation, cela peut simplifier l'analyse; sinon, en excluant les mots vides et en sélectionnant tous les mots contenant entre deux et trois caractères. En analysant les phrases dans lesquelles sont employées ces abréviations/acronymes et à l'aide d'experts en maintenance, il est possible de rattacher ces termes à leur signification dans un dictionnaire. Enfin, toutes ces informations sont consignées dans un dictionnaire, afin de normaliser le texte à travers le processus de traitement automatique de la langue.

7.2 Modélisation des entités

Dans cette section, la mise en œuvre et la validation d'un modèle de reconnaissance d'entités seront d'abord détaillées. Ensuite, le processus de développement d'un modèle de traitement du langage naturel personnalisé sera décrit étape par étape. Enfin, cette section détaillera les avantages de l'utilisation des documents d'AMDE dans la méthodologie développée.

7.2.1 Aperçu général de la méthodologie

Le développement d'un modèle personnalisé de reconnaissance d'entités nommées est très similaire au développement d'un modèle standard d'apprentissage supervisé. La première étape consiste à rassembler un ensemble de données étiquetées. Dans le cas présent, il s'agit d'un ensemble de données annotées, c'est-à-dire contenant du texte dont les entités ont déjà été correctement identifiées. Pour être compatibles avec «*Spark NLP*», les données doivent être au format CoNLL-2003. Ensuite, les données sont séparées en échantillons d'entraînement et de validation. La figure 7.2 détaille l'approche générale :

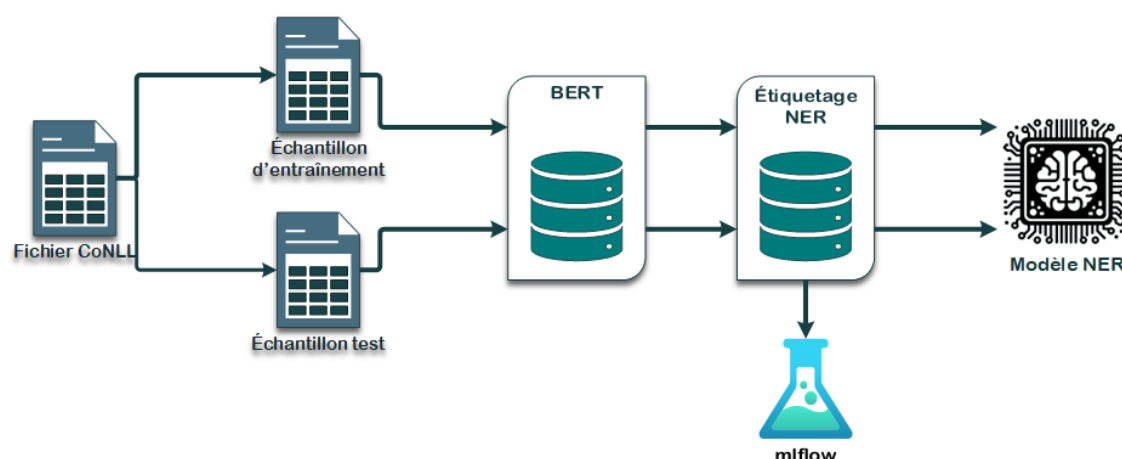


Figure 7.2 Processus de modélisation

Les ensembles d'entraînement et de validation passent par un module BERT, qui permet d'effectuer directement tous les prétraitements TALN (tokenisation, encodage de mots, etc.), puis les vecteurs sont poussés dans un étiqueteur NER. Dans «*Spark NLP*», on parle d'annotateur NER, et celui utilisé dans ce projet est appelé «*NerDL Approach*». Cet annotateur a une architecture de réseau neuronal de Char CNN - BiLSTM-CRF, qui signifie «*Character-level Convolutional Neural Networks, Bidirectional Long Short-Term Memory and Conditional Random Field*» (réseaux neuronaux convolutifs au niveau des caractères, mémoire bidirectionnelle à long terme et champ aléatoire conditionnel). Il combine plusieurs structures de réseaux neuronaux, ce qui facilite l'apprentissage dans des tâches telles que la reconnaissance d'entités. En résumé, l'annotateur NER sert à identifier et à classifier les entités (Équipements, composants, modes de défaillance, etc.) dans le texte, et utilise une structure d'apprentissage profond pour apprendre à partir des exemples envoyés. Ces étapes simples permettent de définir et d'évaluer facilement un modèle de NER qui peut être réutilisé pour d'autres applications TALN, telles que l'extraction d'informations ou la classification de textes. La prochaine sous-section décrit les étapes afin de construire un jeu de données d'entraînement, pour adapter le modèle de reconnaissance au contexte des équipements de poste électrique.

7.2.2 AMDE pour la classification des jetons

La modélisation nécessite un ensemble de données étiquetées. Il existe plusieurs sources pour trouver ce type de données, mais elles sont généralement adaptées à des types d'entités génériques (par exemple, le lieu, le nom de l'entreprise, la date, etc.) Pour ce projet, il était donc nécessaire de générer un ensemble de données d'entraînement, à partir des données de l'entreprise. La première étape a consisté à collecter les données de

l'analyse des modes de défaillance et de leurs effets et à extraire la structure générale de l'AMDE pour l'équipement cible, comme le montre la figure 7.3.

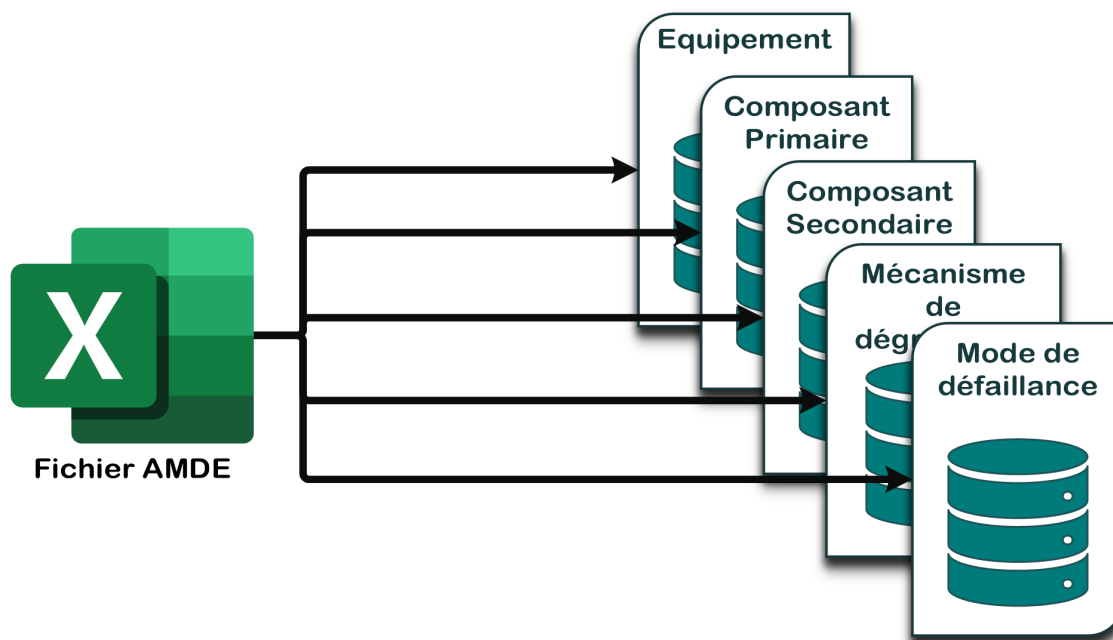


Figure 7.3 Extraction d'entités

L'AMDE fournie comprend les noms de tous les composants ainsi que leurs modes de défaillance et mécanismes de dégradation respectifs. Grâce à des opérations simples, ces données peuvent être utilisées pour construire un tableau qui relie tous les termes liés à la maintenance à une liste d'entités. Les textes des ordres de travail de maintenance peuvent maintenant être annotés à l'aide de techniques de correspondance des motifs («*Pattern-Matching*») en utilisant ce tableau. En d'autres termes, on retrouve les termes correspondants, par exemple «transformateur» et on lui attribue une étiquette, dans ce cas-ci «EQUIPEMENT». Ainsi, les termes sont identifiés et classifiés selon les mêmes regroupements que dans l'AMDE.

7.2.3 Création des données d'entraînements

Cette sous-section décrit la préparation de l'ensemble de données, comme indiqué dans la figure 7.4 :

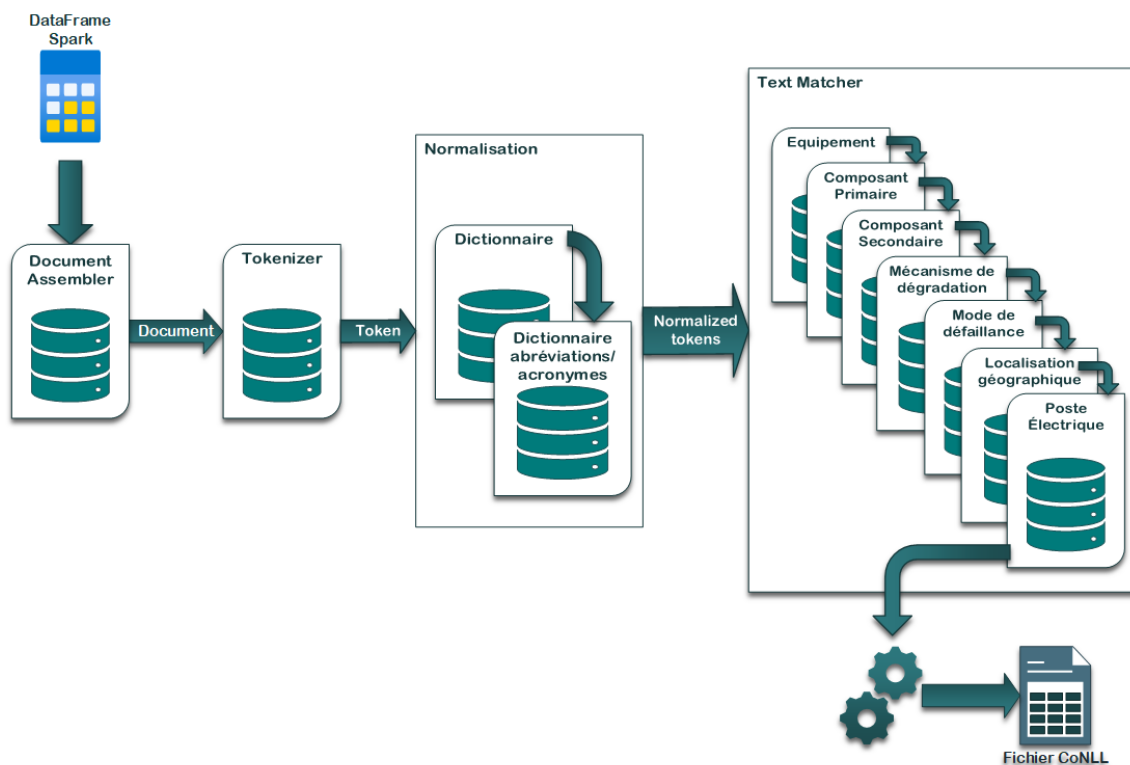


Figure 7.4 Étapes de préparation des données d'entraînement

Tout d'abord, les données textuelles de maintenance sont prétraitées à l'aide de règles de nettoyage simples. Ensuite, la colonne du «*DataFrame Spark*» contenant le texte est envoyée dans un pipeline «*Spark NLP*», conçu pour annoter le texte. Le premier élément du pipeline est le «*Document Assembler*», qui est généralement le point de départ des applications «*Spark NLP*». Son rôle consiste essentiellement à lire une chaîne de caractères et à la convertir dans un format qui peut être traité par les autres éléments du pipeline. Ensuite, les données, désormais sous forme de Document, sont envoyées au tokenizer qui, comme décrit plus haut, divise les éléments de texte en jetons (tokens),

l'unité élémentaire du processus TALN. Ces jetons sont ensuite traités lors d'une étape de normalisation. Le «*Normalizer*» de «*Spark NLP*» prend les jetons en entrée et nettoie le texte, en fonction de sa configuration. Le «*Normalizer*» peut être programmé avec des expressions régulières ou transformer les jetons à l'aide d'un dictionnaire (appelé «*Slang Dictionary*»). Dans ce projet, deux dictionnaires distincts ont été créés : l'un pour la correction des mots et l'autre pour les abréviations connues et le remplacement des acronymes, tel que décrit plus tôt.

Enfin, la dernière étape du pipeline est le «*Text Matcher*». Cet élément établit une correspondance entre le texte de maintenance et les entités extraites de l'analyse des modes de défaillance et de leurs effets. Contrairement à d'autres outils d'annotation d'entités qui utilisent l'apprentissage automatique, «*Text Matcher*» établit une correspondance exacte entre les textes et l'expression de l'entité, ce qui en fait un outil idéal pour la constitution d'un ensemble de formation. En ayant établi des dictionnaires de normalisation, le programme comprend ainsi toutes les variantes et les acronymes d'un mot, permettant que ces éléments soient aussi identifiés par le «*Text Matcher*». En outre, la correspondance exacte serait problématique, étant donné que peu de mots sont parfaitement écrits dans les textes de maintenance. Une fois le texte entièrement annoté par le pipeline, le jeu de données résultant est transformé dans le format de fichier CoNLL-2003 afin qu'il puisse être utilisé pour entraîner le modèle de reconnaissance des entités nommées.

7.3 Classification des ordres de maintenance

La classification des ordres de maintenance est l'objectif principal de ce travail. Les étapes de modélisation subséquente auront permis d'assurer que les modèles de traitement automatique du langage soient en mesure de «comprendre» l'essence des textes analysés. En effet, l'annotation du texte est essentielle afin d'adapter les approches

d'apprentissage profond du TALN. L'élaboration de dictionnaires permet de réduire la taille de l'échantillon de mots ou d'expressions de façon à ce que les termes ayant la même signification soient tous compris de la même manière par un ordinateur. Le développement d'un modèle de reconnaissance d'entité ajoute aussi une couche d'intelligence, favorisant la compréhension de certains termes importants du texte. De plus, ces différents outils peuvent être réutilisés dans d'autres applications TALN.

7.3.1 Mise en œuvre et validation

Le cadre expérimental décrit ici comprend les outils utilisés, la sélection des mesures pour l'évaluation et les procédures de validation. En définissant soigneusement le cadre expérimental, l'objectif est de fournir des informations reproductibles et significatives sur les capacités et les limites de l'approche. Le développement a été effectué sur la plateforme «*Azure DataBricks*», une solution analytique basée sur le cloud qui combine le traitement basé sur «*Apache Spark*» avec les services cloud Azure. La programmation est réalisée à l'aide de «*PySpark*», une interface de programmation Python pour «*Spark*». La bibliothèque utilisée pour la modélisation est «*Spark NLP*», et les bibliothèques «*Spacy*» et «*NLTK*» ont été utilisées principalement pour le prétraitement et l'analyse exploratoire. Les résultats de la modélisation et de la validation ont été enregistrés à l'aide de ML Flow. L'entraînement et la validation des modèles sont générés directement dans le pipeline «*Spark NLP*». Les hyperparamètres contrôlés sont les suivants, pour l'entraînement du modèle de reconnaissance d'entités :

- Nombre d'itérations : 10
- Taux d'apprentissage : 0.003
- Entraînement/Validation : 80 % - 20 %

Le registre d'entraînement et de validation recense tant pour la reconnaissance d'entités que pour la classification des ordres de maintenance :

- Nombre de vrais positifs (tp)
- Nombre de faux positifs (fp)
- Nombre de faux négatifs (fn)
- Précision (prec) :
 - Le rapport entre les observations positives correctement prédites et le total des observations positives prédites.

$$\text{Précision} = \frac{tp}{tp + fp} \quad (7.1)$$

- Rappel (rec) :
 - Le rapport entre les observations positives correctement prédites et les observations positives réelles.

$$\text{Rappel} = \frac{tp}{tp + fn} \quad (7.2)$$

- Score F1 (f1) :
 - La moyenne harmonique de la précision et du rappel, qui fournit une mesure équilibrée.

$$F1 = 2 \cdot \frac{\text{Précision} \cdot \text{Rappel}}{\text{Précision} + \text{Rappel}} \quad (7.3)$$

Le programme enregistre également les valeurs globales et les moyennes macro et micro. Ces mesures sont disponibles pour l'échantillon d'entraînement et l'échantillon de validation pour chaque itération.

7.3.1.1 Algorithmes et hyperparamètres d'apprentissage

Il existe une panoplie d'algorithmes disponibles pour traiter les problèmes de classification, tels que discutés au chapitre 4. De plus, la performance des algorithmes est relative à chaque cas d'application. Dans cette optique, il est important de s'intéresser tout autant à l'annotation du texte et à son effet qu'au choix des techniques de classification employées. Un premier classificateur est construit directement avec «*Spark NLP*» et son module «*ClassifierDL*», qui fait l'entraînement d'un réseau de neurones artificiels. Celui-ci est ensuite comparé à deux autres modèles, en utilisant la régression logistique et la méthode ensembliste «*Random Forest*», de la librairie Spark ML. Le tableau suivant présente les paramètres utilisés pour chaque algorithme lors de la comparaison ainsi qu'une description de chaque paramètre :

Tableau 7-1 Hyperparamètres des algorithmes utilisés

Algorithme	Paramètre
Random Forest	Nombre d'arbres
	Profondeur maximale
Régression Logistique	Itérations maximales
	Paramètre de régularisation
	Paramètre Elastic Net
Réseau de Neurones Artificiel	Taux d'apprentissage
	Taille de lots

Dans le cas de la forêt d'arbres de décision, le nombre d'arbres permet de spécifier le nombre de classificateurs faibles pour effectuer le vote majoritaire. La construction d'un ensemble de classificateurs permet de faire un compromis entre le biais et la variance du modèle [143]. Le paramètre de profondeur d'arbre contrôle la taille de chaque arbre et donc limite la recherche pour éviter le surajustement. Pour la régression logistique, le paramètre du nombre maximum d'itérations encadre la minimisation de la fonction de coût, et limite la durée d'exécution de l'algorithme. Les paramètres de régularisation

et «Elastic Net» servent tous deux à prévenir le surajustement; le premier sert à ajuster l'impact des coefficients de pénalité à la fonction de coût, et le second détermine la méthode de pénalisation. Enfin, le réseau de neurones utilise le taux d'apprentissage et la taille des lots comme hyperparamètres. Les poids du réseau neuronal sont ajustés en fonction du taux d'apprentissage et la taille de lots contrôle le nombre d'exemples d'entraînement envoyés avant la mise à jour du réseau. Une fois les trois techniques comparées, une attention particulière est portée pour l'optimisation des hyperparamètres de l'algorithme le plus performant.

Chapitre 8 - Applications et résultats

Ce chapitre a pour but de présenter l'application de chaque étape de la méthodologie à travers une étude de cas industriel. Les résultats sont présentés en fonction de chacune de ces étapes.

8.1 Étude de cas

Cette recherche est menée dans une compagnie d'électricité canadienne, Hydro-Québec (HQ). Les données utilisées proviennent des actifs du réseau de transport d'électricité. Hydro-Québec TransÉnergie (HQT) est chargée d'assurer la fiabilité du réseau de transport d'électricité pour la province de Québec. Son réseau comprend 533 postes et 34 000 km de lignes de transport à haute tension. L'analyse porte sur les transformateurs de puissance, mais la méthodologie sera étendue à un plus grand nombre d'actifs. L'avantage d'appliquer cette méthodologie dans un tel contexte est que les infrastructures des services publics d'électricité doivent être bien entretenues et que ces entreprises disposent souvent de bases de données d'entretien avec des enregistrements détaillés des travaux effectués. Dans le cas présent, l'entreprise dispose de plus de 50 ans d'historique de maintenance, ce qui offre de grandes possibilités d'applications en matière de NLP et de ML. Le tableau 8-1, traduit de [144] présente un exemple simplifié d'un tableau d'Analyse des Modes de Défaillance et de leurs Effets. Cette représentation permet de se familiariser avec la structure d'un AMDE sur les transformateurs de puissance, sans toutefois divulguer des informations confidentielles du partenaire d'affaires. Les AMDE des transformateurs de puissance, sont fournies par Hydro-Québec. Ces analyses ont été conduites par des groupes d'expert et de chercheurs d'HQ, et l'AMDE utilisé pour ce travail a récemment été révisé par ces groupes.

Tableau 8-1 Exemple de fichier AMDE

ID	Composants	Fonction	Modes de défaillances	Cause(s) de défaillance	...
1	Isolation solide	Isolation des enroulements	Chimie physique	Humidité excessive	...
2	Isolation d'huile	Isoler et refroidir la partie active du transformateur	Chimie physique	Contamination par des particules	...
3	Enroulements	Conduire le courant	Mécanique	Serrage insuffisant	...
4	Cuve	Enclore l'huile/protéger la partie active	Chimique/Physique	Entretien insuffisant	...
5	Traversées	Connecter les enroulements au réseau, isoler cuve et enroulements	Chimie physique	Manque d'entretien	...
6	Noyau	Supporter le champ magnétique	Thermique	Courants de circulation vers la terre	...
7	Interrupteur de dérivation	Maintenir un courant cohérent	Électrique	Contact usé	...

8.2 Création de dictionnaires de normalisation

Une fois les données textuelles extraites, les étapes de prétraitement mentionnées à la section 6.1 ont été appliquées. Le processus de nettoyage a également été l'occasion de compter les occurrences de mots avant et après ces opérations. Les fonctions de nettoyage comprenaient :

- Supprimer les doublons;
- Supprimer les caractères spéciaux;
- Séparer la ponctuation des mots;
- Supprimer les espaces inutiles;
- Convertir les codes d'équipement en fonction de leur signification;
- Supprimer sélectivement les mots vides.

La première opération consiste à supprimer les données entrées en double. Il arrive effectivement qu'il y ait des doublons, parfois issus de la jonction de plusieurs tables, par exemple. Il est donc préférable d'éliminer ceux-ci pour simplifier l'analyse. Il existe une multitude de caractères spéciaux, mais dans le cadre d'une analyse NLP, la plupart de ces caractères n'ajoutent pas de contexte important, ils sont donc aussi éliminés. Ensuite, les algorithmes NLP peuvent créer inutilement des jetons. Si un mot est collé à une ponctuation, le programme peut créer deux jetons différents, par exemple, mot? et mot serait représenté comme des éléments différents. En séparant systématiquement les mots de la ponctuation par une espace, on peut éviter ce problème. Il arrive aussi qu'il y ait des doubles espaces dans le texte qui doivent être éliminés. Certains codes alphanumériques ont une signification qui peut être traduite. Par exemple, un *T* suivi d'un chiffre désigne un transformateur particulier dans un poste électrique. Il est donc possible de convertir ces codes en conséquence. D'autres codes hérités d'ancien système qui ne peuvent être

convertis sont quant à eux éliminés. Finalement, les mots vides, en anglais «Stopwords» , sont principalement des pronoms, prépositions, etc. qui se retrouvent en grandes quantités, sans nécessairement avoir un impact significatif pour la compréhension d'une phrase. Une pratique courante dans le TALN est d'éliminer complètement les mots vides. Cependant, dans un contexte de texte technique, certains de ces termes peuvent avoir un impact important dans l'interprétation, par exemple . Le prétraitement a permis d'éliminer 72 % des jetons et 80 % du nombre total de mots. Le tableau 8-2 présente l'effet des étapes de traitement sur le nombre de jetons et sur la taille du vocabulaire.

Tableau 8-2 Nombre de jetons et de mots au fil des étapes de traitement

Étape	Nombre de jetons	Pourcentage	Nombre de mots (Vocabulaire)	Pourcentage
Corpus Original	5 695 598	-	157 068	-
Prétraitement	1 596 470	72,0 %	31 721	79,8 %
Normalisation	741 997	87,0 %	20 726	86,8 %

L'un des principaux défis liés à l'application du TALN aux textes de maintenance réside dans la complexité linguistique, caractérisée par un vocabulaire technique spécialisé, une abondance d'acronymes et d'abréviations, ainsi que la variabilité introduite par des synonymes et des erreurs orthographiques récurrentes. Il est primordial de diminuer cette variabilité à l'aide de la normalisation lexicale. Il existe plusieurs façons de faire, et les approches les plus populaires ont été testées.

Dans un premier temps, un correcteur orthographique français, disponible dans une librairie «*Python*» a été testé. «*Pyspellchecker*» est un outil populaire, qui utilise l'algorithme de distance de «*Levenshtein*» en comparant les permutations possibles d'un

mot inconnu à une liste de mots connus. Les résultats de ce test se sont montrés inconcluants, cette approche a donc été écartée pour le projet. Une approche alternative testée, bien qu'encore inefficace dans ce contexte, repose sur la vectorisation complète du vocabulaire. Dans ce cas-ci, la représentation a été créée en utilisant «*Word2vec*». Cette approche évalue la proximité sémantique en calculant la distance vectorielle entre un mot cible et l'ensemble des autres mots du vocabulaire. Les mots les plus proches dans l'espace vectoriel sont ensuite identifiés comme des synonymes potentiels. Les résultats produits par cette méthode ont été tout aussi erronés qu'avec le correcteur orthographique. Cela démontre la difficulté encourue dans l'analyse des textes techniques et par la grande variabilité dans la terminologie employée. Enfin, il existe aussi des logiciels conçus spécifiquement pour le TALN qui permettent de faire automatiquement la normalisation. Cette méthode a aussi été écartée, puisque ce type d'outil n'était pas disponible dans l'entreprise. Par conséquent, une approche alternative a dû être développée pour le contexte.

Tel que décrit dans la section méthodologie, il a été convenu de se concentrer sur les termes les plus importants quant à la maintenance d'un actif. Par conséquent, un lexique de base est établi à l'aide de l'AMDE des transformateurs de puissance. Pour chaque terme identifié, une expression régulière sert à retrouver les synonymes potentiels. En validant le terme inconnu dans la phrase où il est employé, il est ajouté à un dictionnaire de normalisation s'il y a une correspondance. En outre, c'est plus de 4000 mots inconnus qui ont pu être rattachés à une centaine de termes issus de l'AMDE. Le tableau 8-3 présente les cas les plus importants en termes de nombre de variantes associées à un seul terme. Comme le montre le tableau, certains mots peuvent être désignés par plus d'une centaine de formes différentes. Les termes du lexique ont contribué à identifier des variantes dans

un rapport d'environ 4:1. En outre, ce dictionnaire a permis de réduire de 35 % additionnel le nombre de mots dans le corpus.

Tableau 8-3 Les 10 principaux termes et leurs variantes

Mot	Nombre de variants
dessiccant	175
défectueux	65
alarme	64
ventilateur	63
transformateur	48
traversée	45
température	41
refroidissement	37
échantillon	37

Suite à la première itération, une nouvelle liste de termes inconnus est extraite. L'emploi de la fonction «*has_vector*» de la bibliothèque «*spacy*» sert à vérifier si un terme est présent dans la définition du modèle. Dans le cas où un terme n'est pas répertorié dans ce vocabulaire ni dans le dictionnaire de normalisation, il devient nécessaire de lui attribuer une définition. En analysant la liste, il est évident que les termes peuvent être divisés en catégories et traités rapidement.

La première catégorie de termes réfère aux emplacements géographiques. En utilisant les données gouvernementales, une liste préliminaire de localisation a été construite. Plusieurs localisations portent des noms religieux, par exemple Sainte-Foy, qui peuvent être écrits de multiples façons (Sainte-Foy, Ste-Foy, Sainte Foy, Ste Foy, etc.). Ainsi, un dictionnaire de localisation a été rédigé pour tenir compte des diverses configurations. Le même exercice a été effectué en utilisant la documentation d'Hydro-Québec. Les postes électriques sont généralement nommées en fonction de la position géographique, ceux-ci ont donc été ajoutés au dictionnaire.

La seconde catégorie représente les abréviations et les acronymes, qui sont fréquemment utilisés par les opérateurs. Comme présenté dans la méthodologie, les termes de 2 à 3 lettres, excluant les mots vides, sont alors analysés. De la même manière que pour les termes de l'AMDE, des expressions régulières ont été utilisées pour extraire les mots courts, et les termes ont été associés manuellement à leur forme normale connue. La documentation a facilité la définition d'une bonne partie des acronymes, et la consultation d'experts interne a permis de confirmer ces éléments du dictionnaire et définir une bonne partie des autres acronymes et abréviations inconnus. Au total, cela représente plus de 200 abréviations et acronymes qui ont été identifiés et normalisés, et qui ont été utilisés à plus de 50 000 reprises dans l'échantillon analysé.

La troisième catégorie de termes inconnus réfère aux noms propres. Ce travail a été entamé; une liste a été établie à partir des prénoms et noms de famille du recensement du gouvernement du Québec. En revanche, ce dictionnaire n'a pas encore été implémenté.

Enfin, il reste trois autres catégories qui peuvent être observées. Il y a quelques termes qui n'ont pas été associés en utilisant les expressions régulières, mais dont il est facile de définir la forme normale. Ils ont donc été facilement ajoutés au dictionnaire. Il y a aussi les termes où il a été impossible de trouver une définition, et, finalement, il y a des termes qui sont ambigus. Par exemple, «*HT*» est un acronyme fréquemment employé, mais il peut désigner entre autres «*haute tension*» et «*hors tension*». Il est donc préférable que les termes de ces deux dernières catégories soient ignorés, dans le cadre de cet exercice. Finalement, ces différentes étapes pour compiler un dictionnaire de normalisation a permis de réduire considérablement la taille du corpus. La figure 8.1 présente un exemple fictif de textes de maintenance traités par les règles de normalisation

Description	Description normalisée
T4=Ventillateur, moteur semble faible.	transformateur 4 ventilateur moteur semble faible.
#####:ventillateur T3 phase 2 défectu	ventilateur transformateur 3 phase 2 défectueux
#####:remplacer récipient gel de sili	remplacer récipient dessiccant
Cond anorm sys refr	condition anormale système refroidissement

Figure 8.1 Exemples de normalisation du texte

L'expérience acquise montre que la méthodologie a permis d'améliorer considérablement la qualité des données textuelles, tout en minimisant le recours aux experts en équipement, aux techniciens de maintenance et à d'autres spécialistes. En effet, bien que ces étapes puissent sembler fastidieuses, elles peuvent facilement être réalisées par un commis aux données bien formé et avec un peu de soutien, plutôt que par des professionnels. En outre, une fois la première itération terminée, il est possible d'effectuer d'autres cycles de nettoyage; plus la qualité du texte est élevée, plus il est facile d'utiliser le NLP pour retrouver des synonymes, ou pour la correction orthographique, par exemple.

8.3 Modèle de reconnaissance d'entités

Cette section présente les résultats obtenus par le développement du modèle de reconnaissance d'entités. Les données d'une classe spécifique d'équipement de poste, dans ce cas-ci, les transformateurs de puissance, et l'AMDE correspondante, sont utilisées comme point de départ. Les résultats d'entraînement et de validation seront analysés et un exemple appliqué de reconnaissance d'entités sur des données de maintenance sera utilisé pour démontrer l'utilité et la robustesse des modèles. Comme mentionné précédemment, l'atteinte des objectifs de ce travail passe par le développement d'une méthodologie d'annotation des données textuelles, basée sur l'analyse des modes de défaillance et de

leurs effets. Les résultats qui suivent démontrent la pertinence et l'efficacité de cette étape cruciale.

Dans un premier temps, le modèle a été entraîné en utilisant les données des ordres de travail de 2010 à 2017. Cette période a été choisie parce que les données postérieures à 2017 seront analysées à l'aide du modèle dans des applications futures. La période antérieure à 2010 a été exclue afin de conserver les données les plus récentes; la plage d'analyse pourrait être élargie, mais les résultats du modèle sont néanmoins satisfaisants avec la période choisie. Les ordres ont également été filtrés afin d'éviter les redondances et de s'assurer que chaque ordre est accompagné d'un texte. En outre, l'ensemble de données contient plus de 15 000 ordres de travail, dont 30 % ont été sélectionnés pour tester le modèle. Au total, cela représente plus de 50 000 exemples d'entraînement, dont 20 % ont été utilisés pour valider le modèle. La figure 8.2 montre la séparation de l'ensemble de données en échantillons d'entraînement, de validation et de test.

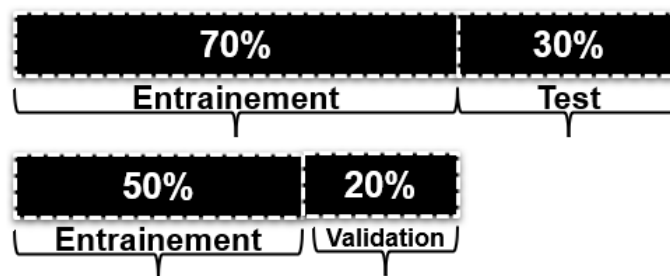


Figure 8.2 Séparation des ensembles d'entraînement, de validation et de test

La figure 8.3 montre l'évolution de la moyenne macro du rappel au fil des cycles d'entraînements, pour l'échantillon de validation et de test. L'analyse des courbes montre que la moyenne du rappel augmente rapidement au cours des premiers cycles d'apprentissage, et, après environ cinq cycles, les performances commencent à se stabiliser. Cela indique qu'il y a une convergence du modèle. La performance observée sur

l'échantillon de test indique une excellente aptitude à généraliser aux nouvelles données. Dans le cas d'un surapprentissage, une diminution de la performance sur l'ensemble de validation aurait été constatée, ce qui n'est pas observable ici. En outre, ce graphique démontre la bonne performance du modèle, autant à l'entraînement que sur de nouvelles données.

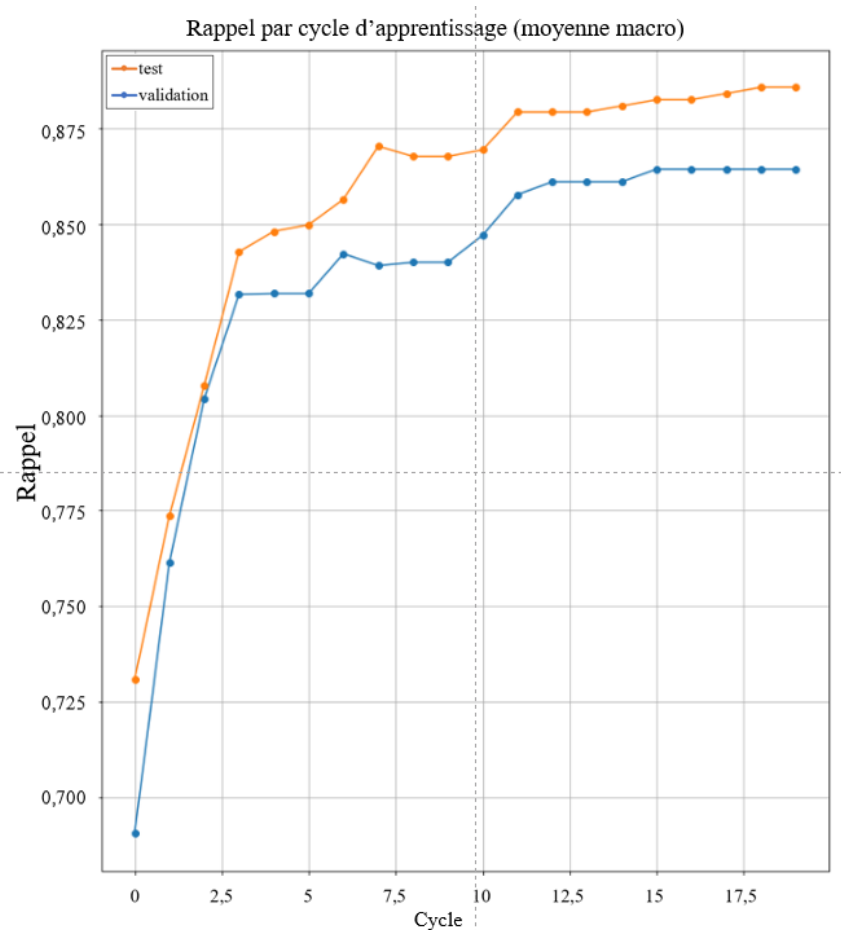


Figure 8.3 Courbes d'apprentissage du modèle NER

Une fois l'entraînement, la validation et les tests terminés, les résultats ont été analysés pour déterminer si le modèle avait une bonne capacité à identifier et à classer les entités dans le texte. Le tableau 8-4 présente les résultats globaux des performances du modèle après 10 cycles.

Tableau 8-4 Résultats de la moyenne micro et macro sur les données tests

Métrique	Précision	Rappel	Score F1
Moyenne Macro	0,96	0,82	0,88
Moyenne Micro	0,97	0,97	0,97

Tout d'abord, le modèle de reconnaissance d'entités présente généralement d'excellentes performances, comme le révèlent les statistiques micromoyennes élevées pour le score F1, le rappel et la précision. Cependant, la macro-moyenne montre qu'il semble y avoir un déséquilibre entre la précision et le rappel, ce qui indique qu'il peut y avoir une certaine variabilité dans les performances entre les classes (entités). En examinant les statistiques détaillées pour chaque classe, on constate que les entités correspondant à la localisation géographique et aux postes électriques (localisation), ainsi que les mécanismes de dégradation, sont les catégories qui obtiennent les scores les plus faibles. Cependant, les résultats concernant la classe de localisation montrent que le modèle est très performant, avec un score F1 de 0,86, et un bon équilibre entre la précision et le rappel. Le tableau 8-5 présente les statistiques détaillées pour l'échantillon de test et pour chaque type d'entité.

Pour les mécanismes de dégradation, la précision est relativement faible, tout comme le rappel. Cette classe est également sous-représentée par rapport aux autres entités dans les échantillons d'apprentissage et de test, ce qui explique probablement sa faible performance. Ce résultat suggère que la précision du modèle pourrait être améliorée en travaillant sur cette classe. Plusieurs moyens pourraient être utilisés pour obtenir

Tableau 8-5 Résultats de l'entraînement pour chaque type d'entité

Entité	Score F-1	Précision	Rappel
Composant Primaire	0,90	0,91	0,89
Composant Secondaire	0,96	0,95	0,97
Équipement	0,99	0,98	1,00
Localisation	0,86	0,88	0,84
Mécanisme de Dégradation	0,37	0,50	0,29
Mode de Défaillance	0,98	0,99	0,98

de meilleurs résultats; l'ajout de synonymes ou de différentes formulations utilisées pour définir les mécanismes de dégradation serait un bon point de départ. Une autre itération d'amélioration des dictionnaires, avec un accent particulier sur ces termes, serait également une piste intéressante. Une autre solution consisterait à rééchantillonner l'ensemble de données de manière à ce qu'il contienne une plus grande proportion de textes contenant ce type d'entités, ou simplement à étendre la portée de l'analyse sur quelques années supplémentaires. Si ces options ne fonctionnaient pas, cette classe pourrait également être supprimée du modèle de NER, car sa précision est très aléatoire à 50 %. Hormis les mauvais résultats de la classe des mécanismes de dégradation, le reste des entités est généralement bien classé et les performances du modèle sont excellentes.

Il est donc clair qu'avec cette méthode d'annotation, la capacité des modèles de TALN à comprendre les textes techniques est améliorée. Comme mentionné dans la littérature, appliquer des techniques de TALN sans les adapter au contexte conduit à des résultats douteux. La méthodologie développée permet d'extraire des informations de manière fiable et rigoureuse, en réutilisant des sources de connaissances déjà disponibles au sein de l'organisation. L'un des défis de l'application des techniques de ML à de telles données est la qualité et la quantité des données disponibles. Grâce à cette nouvelle technique d'annotation, le processus de création d'un ensemble de données, et donc de modèles NLP, est simplifié. L'utilisation de l'expertise existante ne se limite pas aux AMDE, mais

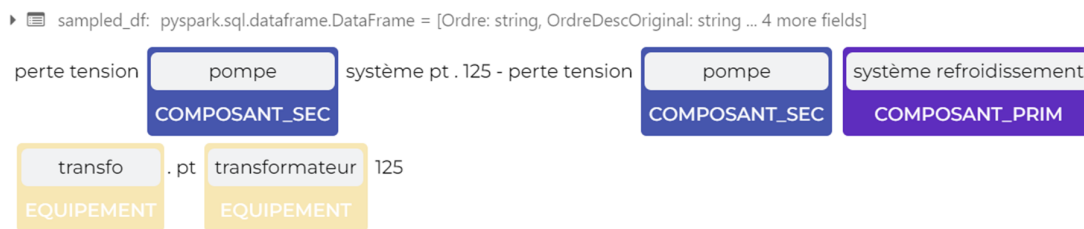


Figure 8.4 Exemple de résultat du modèle de reconnaissance d'entités

la méthodologie peut être étendue à d'autres techniques d'analyse visant à décomposer un système et ses relations. La figure 8.4 présente l'annotation fournie par le modèle de reconnaissance d'entités nommées sur un exemple de texte de maintenance fictif.

Toutefois, il existe certaines limites; comme l'a montré l'étude de cas, certaines entités peuvent ne pas être suffisamment référencées dans le texte, ce qui rend difficile l'obtention d'un modèle fiable permettant de les distinguer. En outre, lors de l'extraction des données, une différence de notation et d'écriture a été constatée, notamment lorsque des changements ont été apportés aux systèmes informatiques de l'entreprise; cela peut se traduire par l'utilisation d'acronymes qui ont changé au fil du temps, ou par des références à une codification qui n'est répertoriée dans aucune documentation accessible.

En fin de compte, les modèles générés par la méthodologie permettront d'inclure ces éléments dans des pipelines NLP beaucoup plus complexes. En fait, les modèles NER sont des éléments essentiels à la compréhension du texte pour de nombreuses applications NLP. Ces modèles gagneraient donc à être utilisés pour effectuer la classification de textes, par exemple pour classer les types d'activités qui ont été effectuées sur un équipement particulier, pour extraire des connaissances du texte en énumérant les composants et l'équipement impliqués dans un travail de maintenance, et bien d'autres choses encore.

8.4 Classification des ordres de maintenances

Cette section détaille les résultats de modélisation, pour la classification des ordres de maintenance. En premier lieu, deux modèles sont comparés, un avec et l'autre sans annotations. En second, différentes techniques d'apprentissage supervisé sont comparées afin de déterminer la plus adaptée au projet. Enfin, les hyperparamètres de la méthode choisie sont optimisés.

8.4.1 Comparaison des modèles avec et sans annotations

L'analyse est effectuée à l'aide de données provenant d'Hydro-Québec et quantifie les effets des efforts d'annotation sur la capacité de classification des modèles. Le modèle de classification présenté montre une grande capacité de prédiction, malgré la faible quantité de données étiquetées disponibles. Les résultats démontrent une amélioration significative des modèles annotés par rapport aux autres modèles. La construction du modèle de classification de texte nécessite une grande quantité de données étiquetées pour produire des résultats significatifs. Actuellement, seule une partie des données a été étiquetée à cette fin spécifique. Toutefois, les résultats de cette première preuve de concept sont très encourageants. L'ensemble de données contient plus de 3 000 enregistrements de maintenance, ce qui représente tout de même un nombre important. D'autre part, les classes sont déséquilibrées, avec une répartition d'environ 94 % à 6 %. Il est donc préférable de rééchantillonner les données pour produire un ensemble synthétique, mais plus équilibré. En conséquence, l'ensemble de données utilisé contient plus de 2 000 entrées, qui sont maintenant réparties à 50/50. Les données envoyées aux modèles sont réparties à 70/30 % en échantillon d'entraînement et de test. Dans l'échantillon d'entraînement, un 20 % est retenu pour la validation du modèle.

Les deux modèles sont évalués sur la base des scores de précision, de rappel et de F1. Dans le cas suivant, ces paramètres seront mesurés sur les moyennes micro et macro des modèles. Enfin, les deux modèles sont comparés sur la base des mêmes mesures. Pour chaque indicateur, la moyenne micro évalue la performance moyenne envers chaque classe. Dans ce processus, les catégories ayant le plus grand nombre d'entrées auront un impact plus important sur cette métrique. La moyenne macro, quant à elle, est utilisée pour déterminer une moyenne afin que chaque classe ait le même poids, quel que soit le nombre d'instances par classe. Le tableau 8-6 présente l'évaluation du modèle sans annotation, et le tableau 8-7 présente l'évaluation du modèle avec annotation.

Tableau 8-6 Performance du modèle sans annotations

Ensemble	Métrique	Précision	Rappel	F1
Validation	Macro	25,9 %	50,0 %	34,1 %
Test	Macro	24,5 %	50,0 %	32,8 %
Validation	Micro	51,7 %	51,7 %	51,7 %
Test	Micro	48,9 %	48,9 %	48,9 %

En commençant par l'analyse de la macro-moyenne, la précision du modèle est très faible (environ 25 %), ce qui indique que le modèle a un taux élevé de faux positifs dans les différentes classes. En outre, un rappel de 50 % suggère que le modèle identifie la moitié des vrais positifs dans les différentes catégories. Le score F1, qui reflète l'équilibre entre la précision et le rappel, est faible (environ 33-34 %), ce qui témoigne d'une mauvaise performance en matière de distinction entre les classes. Si l'on considère la micromoyenne, le résultat global du modèle est meilleur que celui de la macro-moyenne, bien que la performance soit discutable. La cohérence entre la précision, le rappel et le score F1 suggère que le modèle est cohérent dans la classification sans être particulièrement bon. La comparaison des résultats avec l'échantillon de validation et l'échantillon de test ne fait apparaître qu'une légère variation. La performance globale

du modèle reste insatisfaisante, ce qui est normal lorsque les textes d'entrée n'ont pas fait l'objet d'un prétraitement rigoureux et d'une annotation manuelle. Les résultats suggèrent une grande variabilité dans les données, ce qui confirme l'hypothèse selon laquelle il est difficile pour les algorithmes de TALN de traiter directement les données d'entretien, sans annotation ni adaptation contextuelle.

Tableau 8-7 Performance du modèle avec annotations

Ensemble	Métrique	Précision	Rappel	F1
Validation	Macro	91.0 %	90.9 %	90.97 %
Test	Macro	87.8 %	87.8 %	87.8 %
Validation	Micro	91.0 %	91.0 %	91.0 %
Test	Micro	87.7 %	87.7 %	87.7 %

Comme le montre le tableau 8-7, le modèle a obtenu des scores élevés pour la précision, le rappel et le F1 dans l'ensemble de validation, avec un pourcentage supérieur à 90 % pour chaque mesure. Ces résultats indiquent d'excellentes performances au niveau de l'apprentissage. En outre, les résultats des mesures sur l'échantillon de test indiquent une performance de plus de 87 %, ce qui suggère que le modèle fasse preuve d'une excellente capacité de généralisation à de nouvelles données. Les scores de précision, de rappel et de F1 sont très stables lors de la validation et du test. Cela suggère que les performances soient cohérentes pour l'ensemble de ces mesures. La macro-moyenne est élevée pour tous les indicateurs, ce qui signifie que le modèle est performant en ce qui concerne la classification pour les deux classes. Enfin, la micromoyenne confirme les performances élevées du modèle pour la classification des données.

En comparant les deux modèles, il est clair que les performances sont améliorées par le prétraitement et l'annotation du texte. La performance du modèle sans annotation (Modèle S-A) démontre la difficulté d'utiliser des modèles NLP préentraînés pour traiter des données techniques, telles que les données de maintenance. La comparaison suivante

illustre les différences entre les deux modèles sur les échantillons de test pour chaque mesure décrite ci-dessus.

Tableau 8-8 Comparaison des mesures et des améliorations du modèle

Moyenne	Modèle	Métrique	Résultats	Amélioration
Micro	Modèle S-A	Précision	48.9 %	
		Rappel	48.9 %	
		F1	48.9 %	
	Modèle A	Précision	87.8 %	79.3 %
		Rappel	87.8 %	79.3 %
		F1	87.8 %	79.3 %
Macro	Modèle S-A	Précision	24.5 %	
		Rappel	50.0 %	
		F1	32.8 %	
	Modèle A	Précision	87.8 %	258.9 %
		Rappel	87.8 %	75.5 %
		F1	87.8 %	167.2 %

En outre, le tableau montre une amélioration d'environ 80 % pour tous les indicateurs de la moyenne micro et, pour la moyenne macro, une augmentation d'environ 258 %, 76 % et 167 % pour la précision, le rappel et le score F1. Le modèle annoté (Modèle A) présente des performances beaucoup plus équilibrées entre les classes et dans l'ensemble. Cela démontre sans aucun doute la pertinence et l'efficacité des méthodes d'annotation mises en œuvre sur les performances des modèles de NLP.

8.4.2 Comparaison des techniques d'apprentissage

Cette section présente une comparaison de différentes techniques d'apprentissages supervisés utilisées pour la classification, afin de déterminer l'algorithme le plus approprié à ce cas d'application.

Tableau 8-9 Paramètres des classificateurs utilisés

Algorithme	Paramètre	Valeur
Random Forest	Nombre d'arbres	100
	Profondeur maximale	4
Régression Logistique	Itérations maximales	10
	Paramètre de régularisation	0.3
	Paramètre Elastic Net	0.0
Réseau de Neurones Artificiel	Taux d'apprentissage	0.001
	Taille de lots	64

Dans le cas de la forêt d'arbres de décision, les paramètres utilisés sont les paramètres par défaut dans «*Spark ML*», mis à part le nombre d'arbres qui est fixé à 100. En laissant le paramètre de profondeur maximale des arbres à 4, cela permet de prévenir un surajustement aux données. Pour la régression logistique, en fixant le paramètre «*Elastic Net*» à 0, cela permet d'utiliser la méthode de la régression de Ridge. Le paramètre de régularisation à 0.3 aide à prévenir le surajustement et favorise le développement d'un modèle plus simple et plus robuste aux données bruitées. Enfin, le réseau de neurones utilise un taux d'apprentissage de 0.001, ce qui permet de maintenir une bonne efficacité et une convergence graduelle de l'entraînement. En optant pour des lots de 64 exemples d'entraînement, cela donne une bonne vitesse de calcul tout en maintenant une bonne stochasticité. Le tableau 8-10 présente les résultats de tests de chaque modèle, entraînés, validés et testés sur le même jeu de données.

La régression logistique démontre un modèle bien équilibré et a une performance similaire entre les classes. Il en va de même pour les deux autres méthodes, qui présentent

Tableau 8-10 Résultats comparatifs des méthodes de classifications

Moyenne	Précision	Rappel	F1	Algorithme
Macro	78,0 %	78,0 %	77,0 %	Régression Logistique
Micro	77,0 %	77,0 %	77,0 %	Régression Logistique
Pondérée	78,0 %	77,0 %	77,0 %	Régression Logistique
Macro	87,0 %	86,0 %	86,0 %	Random Forest
Micro	87,7 %	87,7 %	87,7 %	Random Forest
Pondérée	88,0 %	86,0 %	86,0 %	Random Forest
Macro	91,1 %	91,1 %	91,1 %	Réseau de Neurones (après 20 itérations)
Micro	91,1 %	91,1 %	91,1 %	Réseau de Neurones (après 20 itérations)

des résultats assez équilibrés entre la précision, le rappel et le score F1. Cependant, avec des métriques inférieures à 80 %, la performance de la régression logistique n'est pas impressionnante. Les performances de l'algorithme de la forêt d'arbres aléatoires sont beaucoup plus robustes, mais c'est le réseau de neurones qui présente les meilleurs résultats, dépassant 90 % pour chacun des critères mesurés. À partir de ce constat, les hyperparamètres du réseau de neurones seront optimisés afin de produire le modèle le plus performant possible.

8.4.3 Optimisation des hyperparamètres

Cette sous-section présente l'optimisation des paramètres internes du réseau de neurones. Il existe plusieurs outils pour effectuer l'optimisation des hyperparamètres, mais dû à des enjeux de compatibilités entre les bibliothèques, cette étape a été réalisée manuellement. Ainsi, différents niveaux des deux hyperparamètres sont testés dans un processus itératif. Un premier test a été fait pour tester les combinaisons de paramètres à un taux d'apprentissage de 0.001, 0.01 et 0.1 et des lots de taille de 8, 16, 32 et 64 exemples, tels que présenté au tableau 8-11 :

Tableau 8-11 Résultats des expérimentations en fonction du taux d'apprentissage et de la taille de lot

Taux d'apprentissage	Taille de lot	Moyenne	Précision	Rappel	F1
0,001	8	Macro	0,850	0,849	0,849
		Micro	0,848	0,848	0,848
0,001	16	Macro	0,892	0,892	0,892
		Micro	0,892	0,892	0,892
0,001	32	Macro	0,865	0,865	0,865
		Micro	0,865	0,865	0,865
0,001	64	Macro	0,894	0,891	0,892
		Micro	0,890	0,890	0,890
0,01	8	Macro	0,257	0,500	0,339
		Micro	0,513	0,513	0,513
0,01	16	Macro	0,257	0,500	0,339
		Micro	0,513	0,513	0,513
0,01	32	Macro	0,243	0,500	0,327
		Micro	0,487	0,487	0,487
0,01	64	Macro	0,257	0,500	0,339
		Micro	0,513	0,513	0,513
0,1	8	Macro	0,257	0,500	0,339
		Micro	0,513	0,513	0,513
0,1	16	Macro	0,243	0,500	0,327
		Micro	0,487	0,487	0,487
0,1	32	Macro	0,257	0,500	0,339
		Micro	0,513	0,513	0,513
0,1	64	Macro	0,257	0,500	0,339
		Micro	0,513	0,513	0,513

Ces résultats démontrent sans équivoque que plus le taux d'apprentissage est bas, plus la performance du modèle est élevée. En ce qui a trait à la performance du modèle en fonction de la taille de lots, le changement dans la performance est plus subtil. Cependant, les combinaisons d'hyperparamètres qui produisent les meilleurs résultats sont avec des tailles de lots de 16 et de 64 exemples d'entraînements. La fonction d'efficacité du récepteur, aussi référée comme la courbe ROC («*Receiver Operating Characteristic*») est un bon outil visuel pour comparer ce type de modèle. En mettant en relation le taux de vrais positifs en fonction du taux de faux positifs, la courbe représente la capacité de classification du modèle, en variant le seuil du classificateur. L'aire sous la courbe permet aussi de comparer deux modèles; plus la valeur se rapproche de 1, plus sa performance est élevée. La figure 8.5 suivante présente les courbes pour les deux combinaisons de paramètres les plus performants.

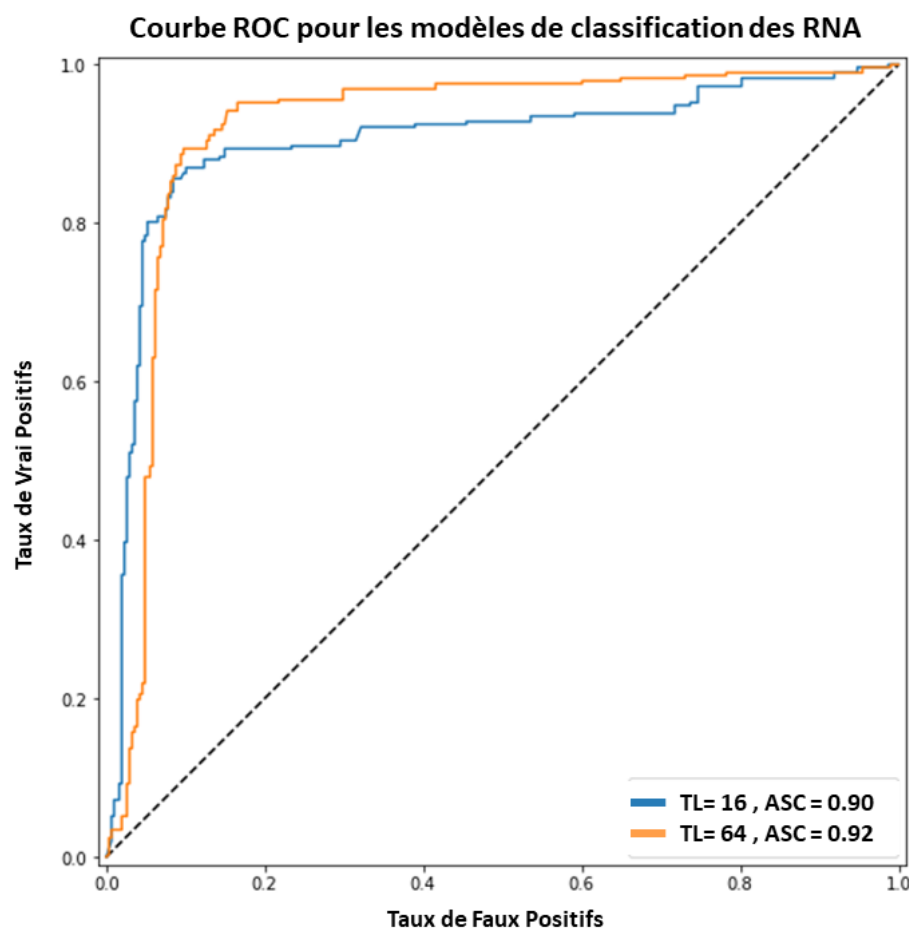


Figure 8.5 Courbe ROC pour les modèles RNA

L'aire sous la courbe des deux modèles permet de démontrer que la meilleure combinaison d'hyperparamètres est avec un taux d'apprentissage de 0.001 et une taille des lots de 64 avec un score de 0.92. Cependant, les résultats démontrent que plus le taux d'apprentissage est bas, plus la performance est élevée, ce qui pousse donc à tester des valeurs inférieures à 0.001, afin de vérifier s'il n'y a pas des taux qui offrent une meilleure performance. Des taux de 0.001, de 0.0005 et de 0.0001 sont donc testés en combinaisons à des tailles de lots de 16, 32 et 64 et les résultats sont illustrés dans le tableau 8-12.

Tableau 8-12 Résultats des secondes expérimentations en fonction du taux d'apprentissage et de la taille de lot

Taux d'apprentissage	Taille de lot	Moyenne	Précision	Rappel	F1
0,001	16	Macro	0,857	0,857	0,857
		Micro	0,857	0,857	0,857
0,001	32	Macro	0,873	0,870	0,871
		Micro	0,868	0,868	0,868
0,001	64	Macro	0,903	0,904	0,904
		Micro	0,903	0,903	0,903
0,0005	16	Macro	0,902	0,902	0,902
		Micro	0,902	0,902	0,902
0,0005	32	Macro	0,903	0,903	0,903
		Micro	0,902	0,902	0,902
0,0005	64	Macro	0,875	0,875	0,875
		Micro	0,875	0,875	0,875
0,0001	16	Macro	0,821	0,821	0,821
		Micro	0,820	0,820	0,820
0,0001	32	Macro	0,815	0,815	0,815
		Micro	0,815	0,815	0,815
0,0001	64	Macro	0,794	0,794	0,794
		Micro	0,793	0,793	0,793

En observant les résultats de la deuxième expérience, la constatation est que l'amélioration du modèle est inversement proportionnelle au taux d'apprentissage, atteint un plateau à 0.0005 et la qualité du modèle recommence à se dégrader à 0.0001. Tout comme pour l'autre expérimentation, les meilleurs modèles sont sélectionnés et la courbe ROC est employée pour déterminer les paramètres optimaux à la figure 8.6.

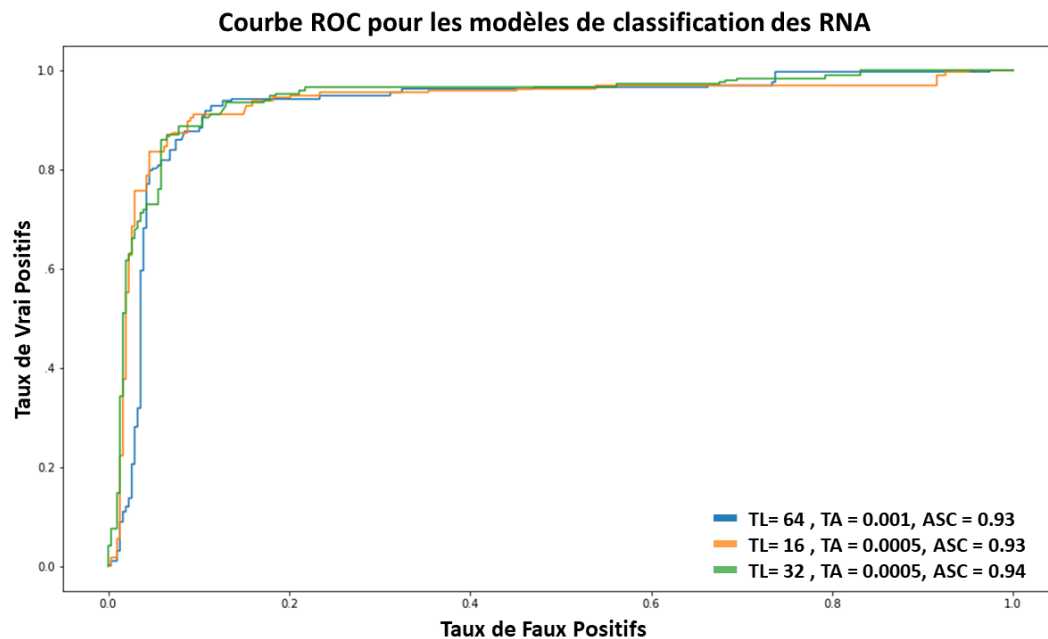


Figure 8.6 Courbe ROC pour les taux d'apprentissage inférieurs à 0.001

Le graphique présente les courbes, qui sont confondues l'une à l'autre, indiquant que la qualité des modèles est assez similaire. C'est en regardant l'aire sous la courbe que la distinction est faite, sans être non plus très marquante. En effet, les trois modèles ont une aire sous la courbe d'au-delà de 90 %, mais la combinaison d'hyperparamètres qui offre la meilleure performance, d'un point de plus que les autres, est avec une taille de lot de 32 et un taux d'apprentissage de 0.0005, avec un score de 94 %. En termes pratiques, une aire sous la courbe de 94 % reflète une grande précision de classification. Le modèle est très efficace pour classer les instances de sorte que les cas positifs soient mieux classés que les cas négatifs. Le modèle démontre aussi une performance élevée et il est très efficace pour faire la distinction entre les deux classes, soit préventif et correctif. Enfin, les figures 8.7 et 8.8 illustrent l'évolution de la performance du modèle en fonction du nombre de cycles d'entraînement, pour les métriques du rappel et du score F1 respectivement. En observant les courbes, une augmentation rapide de la qualité du modèle

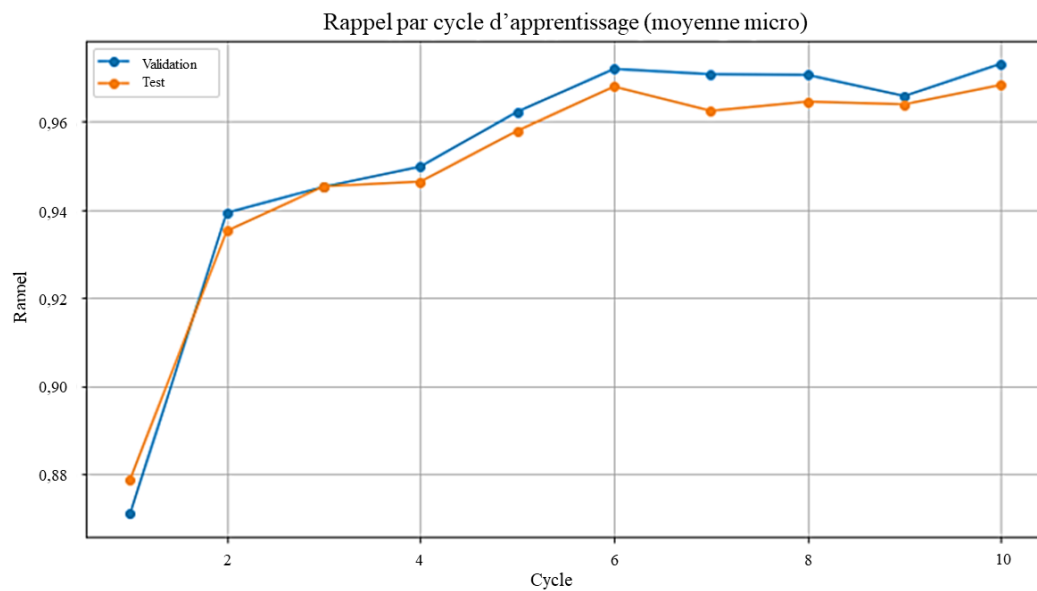


Figure 8.7 Évolution du rappel du modèle de classification en fonction des cycles d'apprentissage

est observée, indiquant que le modèle apprend efficacement à classer les instances. L'écart de performance entre les données tests et les données de validation suggère que le modèle a une bonne capacité de généralisation. Les légères différences de performance peuvent s'expliquer par la variation dans les données échantillonnées. En outre, un plateau est rapidement observé, ce qui laisse croire que le modèle converge sans toutefois démontrer des signes de surapprentissage.

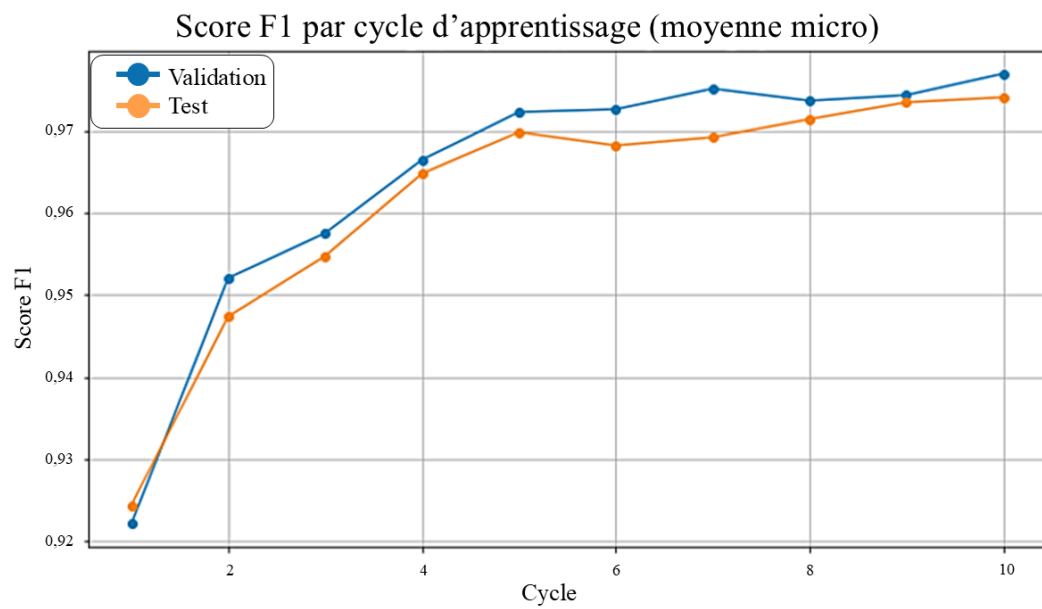


Figure 8.8 Évolution du score F1 du modèle de classification en fonction des cycles d'apprentissage

Chapitre 9 - Discussion

Cette étude visait l'amélioration de la qualité des données de maintenance par l'utilisation des techniques de l'intelligence artificielle, en particulier par la combinaison de l'analyse des modes de défaillance aux méthodes du TALN. Les résultats fournissent une perspective importante sur le rôle que joue l'adaptation spécifique au domaine du TALN et le potentiel d'amélioration du traitement des données de maintenance dans la gestion des actifs. Tout d'abord, l'approche de normalisation du texte tirant profit de dictionnaires élaborés à partir des AMDE a amélioré considérablement la qualité des textes à traiter. En effet, après le prétraitement et la normalisation, le vocabulaire a été réduit d'environ 87 %, diminuant ainsi grandement la variabilité du texte. Une fois les textes nettoyés, un modèle de reconnaissance d'entités a été développé, en tirant profit encore une fois de la structure des AMDE pour définir les catégories d'entités. Les résultats sont excellents pour la majorité des classes, autant pour les moyennes micro que macro. En effet, la précision globale du modèle excède 90 % ce qui suggère que d'utiliser l'AMDE comme structure pour l'annotation offre une performance exceptionnelle. Enfin, les résultats indiquent que l'intégration de l'AMDE dans le processus d'annotation a considérablement amélioré les performances des modèles de classification. En enrichissant les ordres de travail de maintenance avec des étiquettes spécifiques au domaine, le modèle a pu mieux identifier et catégoriser les activités de maintenance. Le modèle entraîné sur les données annotées a surpassé celui basé uniquement sur des modèles génériques préentraînés, avec une amélioration d'environ 80 % sur les métriques de micro moyennes. Cela met en évidence la valeur de l'incorporation des connaissances d'experts dans la préparation des données pour les tâches d'apprentissage automatique. En outre, la capacité du modèle à distinguer les activités de maintenance préventive et corrective a été améliorée, ce qui a permis d'obtenir

des prévisions de maintenance plus précises. Aussi, différentes méthodes de classification ont été comparées; la régression logistique, les réseaux de neurones et les forêts d'arbres de décisions ont été utilisés pour déterminer la technique la plus appropriée pour cette étude. Les résultats suggèrent que le modèle de réseau de neurones est le plus performant, c'est donc celui qui a été choisi, puis ses hyperparamètres ont été optimisés pour assurer les meilleurs résultats possible. Ces résultats contribuent au nombre croissant de recherches sur l'application du traitement automatique du langage naturel, dans le domaine de la fiabilité et de la maintenance. Alors que l'apprentissage par transfert a été largement adopté pour sa capacité à généraliser à travers les domaines, cette étude montre que l'adaptation spécifique au domaine, en particulier dans les domaines techniques, est cruciale pour atteindre une performance optimale. Ainsi, cela vient appuyer ce qui est suggéré dans la littérature, remettant en question la notion selon laquelle les modèles génériques préentraînés sont universellement applicables. Les résultats de chaque étape de la classification démontrent un fait important : la plus grande amélioration observée entre les modèles est lors de la comparaison du modèle annoté et non annoté. Le choix de l'algorithme a aussi un impact significatif, si l'on compare la régression logistique et le RNA, il y a une amélioration d'un peu plus de 16 %. Cependant, l'impact est bien moindre que celui de l'annotation du texte. L'optimisation des hyperparamètres est une étape de modélisation importante, mais son effet est moins notable. Cela démontre aussi que l'utilisation de l'AMDE pour l'adaptation du texte donne d'excellents résultats et qu'il s'agit d'un bon moyen de réutiliser les connaissances d'experts pour affiner les modèles d'IA préentraînés. D'un point de vue pratique, la méthodologie développée dans cette étude offre aux organisations un moyen efficace d'améliorer la qualité de leurs données de maintenance. Par le fait même, cela augmente la qualité des modèles statistiques et des indicateurs de fiabilité. En développant des annotations basées sur l'AMDE, les

responsables de la maintenance peuvent aussi extraire de l'information des données existantes pour la prise de décision. Cette approche simplifie également le processus d'annotation, réduisant la nécessité de mobiliser des ressources pour faire de l'étiquetage manuel en réutilisant les connaissances antérieures. Dans un environnement industriel, cela peut se traduire par une meilleure fiabilité des actifs et une prise de décision mieux éclairée.

9.1 Limites de l'étude

Bien que les résultats soient prometteurs, l'étude présente certaines limites. Le développement des modèles de reconnaissance d'entités démontre que, pour certaines classes, il y a un manque de données d'entraînement suffisantes pour représenter adéquatement celles-ci. Une autre limite est la spécificité des données au réseau de transmission d'Hydro-Québec, qui peut limiter la généralisation des résultats à d'autres industries ou types de données de maintenance, sans un effort d'adaptation.

9.2 Recherches futures

Les limitations mentionnées plus haut offrent cependant des opportunités pour de futures recherches. Bien que l'AMDE ait fourni une base solide pour l'annotation des textes, les performances du modèle pourraient être améliorées en élargissant l'ensemble de données, particulièrement pour les classes sous-représentées. De la même façon, il pourrait être bénéfique de travailler sur le vocabulaire entourant ces classes, afin d'identifier des synonymes potentiels et d'améliorer les dictionnaires. En outre, il serait intéressant d'étendre la méthodologie à d'autres actifs du réseau de transport, et même aux données des actifs de production et de distribution d'Hydro-Québec. Des recherches pourraient aussi être étendues à d'autres secteurs industriels.

Finalement, ce projet démontre la valeur de l'utilisation de l'analyse des modes de défaillance et de leurs effets afin d'annoter et de normaliser les textes de maintenance. De plus, il souligne l'importance d'adapter les techniques TALN aux contextes spécifiques de l'entreprise, afin d'obtenir des résultats adéquats. En effet, la comparaison des modèles démontre sans équivoque la nécessité de l'annotation, notamment à l'aide de l'AMDE, pour le traitement des données de maintenance. Ce travail jette les bases de futurs travaux d'application du TALN aux domaines de la gestion des actifs et de la maintenance, avec des implications pratiques pour l'amélioration de la qualité des données de maintenance et de la prise de décisions.

Chapitre 10 - Conclusion

En conclusion, les travaux présentés dans cette thèse visent à créer une méthodologie basée sur l'IA pour corriger les données de maintenance avec l'aide des modèles de dégradation physique. Le travail est appuyé par une étude de cas, particulièrement par l'analyse des données de maintenance fournies par Hydro-Québec, sur un actif central de son réseau de transport d'électricité. Comme dans plusieurs organisations, l'entreprise possède une foule de données, mais dont la qualité est remise en question. Les données de maintenance sont saisies manuellement par des techniciens, qui peuvent involontairement entrer des informations erronées dans la base de données. De nouvelles techniques du TALN issues de l'apprentissage profond permettent désormais d'utiliser des modèles préentraînés pour la classification du texte. Par le fait même, ces techniques peuvent être adaptées pour reclassifier les ordres de maintenance en tirant profit des descriptions provenant des textes libres. Pour ce faire, la méthodologie développée tire profit des analyses des modes de défaillance et de leurs effets pour l'annotation textuelle. Dans un premier temps, l'AMDE est employé pour la normalisation et la correction des champs textes. Ensuite, il sert de base pour l'annotation et l'extraction des connaissances de ces textes par l'élaboration d'un modèle de reconnaissance d'entités nommées. Enfin, les données annotées ont été utilisées pour faire la reclassification des ordres de maintenance.

10.1 Contribution de la recherche

Cette recherche contribue à la littérature de plusieurs façons. Les publications qui traitent du TALN appliqué à la maintenance sont de plus en plus nombreuses, mais cela reste tout de même marginal. C'est encore plus vrai lorsque l'on considère le traitement de texte en français. Comme le démontre la revue de littérature préliminaire, il y a peu d'applications dans le domaine électrique. Ce projet contribue donc à la littérature, autant

par le fait que ce soit un cas d'application concret sur les transformateurs de puissance que par les nombreux enjeux posés par une application sur des textes techniques francophones.

De plus, la normalisation et l'annotation du texte à partir des AMDE sont aussi des distinctions notables de ce projet. Pour la normalisation du texte, la plupart des méthodes observées dans la littérature ont été difficiles, voire impossibles à appliquer, autres que les méthodes manuelles et le recours aux experts. En utilisant des connaissances déjà structurées, le processus de normalisation a été facilité, notamment par la construction de dictionnaires qui ciblent les mots importants contenus dans les AMDE. Ensuite, cette structure de données a été réutilisée pour construire les modèles d'extraction d'information (modèle NER), ce qui simplifie la création de classes et d'étiquettes. Dans bien des cas, les modèles de reconnaissance d'entités sont construits en utilisant des logiciels d'annotation, ce qui n'était pas possible dans le cadre de ce projet [96, 100, 111]. La méthode développée permet donc de faire l'entraînement d'une manière semi-automatisée, qui s'est avérée très efficace.

Également, les résultats démontrent avec certitude que l'annotation et la normalisation des textes sont des étapes indispensables à l'application des techniques de TALN. La comparaison des classificateurs illustre des résultats impressionnants pour corriger les données, lorsque des textes annotés sont utilisés pour l'entraînement du modèle. Bien que le besoin de prétraiter le texte et d'adapter les méthodes TALN soit fréquemment discuté, aucune étude comparant ces aspects n'a été identifiée à ce jour dans la littérature. Cela souligne aussi que l'incorporation des connaissances issues de l'AMDE, préexistantes dans l'entreprise, permet une adaptation rapide et efficace des modèles préentraînés.

Enfin, l'étude contribue à la littérature sur l'utilisation du TALN au domaine de la maintenance et identifie des solutions pour surmonter les limites d'applicabilité des

modèles d'IA génériques. D'un point de vue pratique, ce travail fournit de l'information pertinente pour les chercheurs et les spécialistes du domaine industriel, notamment dans le domaine de l'énergie, où des données de qualité sont essentielles à une prise de décision rationnelle. La méthodologie offre une solution efficace pour l'amélioration de la qualité des données, qui peut s'étendre au-delà du domaine de la maintenance.

Qui plus est, la démarche présentée dans cette thèse a inspiré plusieurs publications scientifiques. Une première publication servant de preuve de concept a été présentée dans le cadre de la conférence «*World Congress on Engineering Asset Management*» en 2022 et publiée dans le «*WCEAM Proceedings*» [40]. Ensuite, une partie de la revue de littérature a fait l'objet de deux publications dans la revue «*Sustainability*» [18, 19]. Puis, trois autres articles ont été soumis; deux ont été acceptés, respectivement à la conférence «*The 51st International Conference on Computers and Industrial Engineering (CIE51)*» et «*71st Annual Reliability and Maintainability Symposium*». Le troisième article est soumis à une revue et en attente d'une décision.

10.2 Travaux futurs

Comme l'ont démontré les résultats du modèle NER, il est difficile d'identifier et d'extraire de l'information sur certaines classes qui sont sous-représentées dans les données. Cela peut s'avérer une limitation notable si l'on essaie d'identifier des événements rares. De plus, la méthodologie est validée par une étude de cas, mais est limitée à une catégorie d'actifs. De ce fait, les travaux futurs devraient se focaliser sur la bonification du jeu de données, en élargissant la plage temporelle étudiée. De plus, la méthodologie devrait être appliquée à d'autres actifs, afin de voir si elle est généralisable. Pour finir, cette recherche établit une base solide pour les applications futures du TALN dans le traitement des données de maintenance. L'intégration de connaissances spécifiques

au domaine dans les modèles de TALN offre une voie prometteuse pour l'amélioration de la qualité des données dans les environnements industriels, occasionnant ainsi une gestion des actifs plus performante.

Bibliographie

- [1] H. Islam, “Reliability-centered maintenance methodology and application: a case study,” *Engineering*, vol. 2010, 2010.
- [2] G. A. Stillman, G. Kaiser, W. Blum, and J. P. Brown, *Teaching mathematical modelling: Connecting to research and practice*. Springer Science & Business Media, 2013.
- [3] J. P. King and W. S. Jewett, *Robustness development and reliability growth: Value adding strategies for new products and processes*. Pearson Education, 2010.
- [4] M. Rausand and A. Høyland, *System reliability theory: models, statistical methods, and applications*. John Wiley & Sons, 2003, vol. 396.
- [5] K. Goebel, A. Saxena, M. Daigle, J. Celaya, I. Roychoudhury, and S. Clements, “Introduction to prognostics,” in *European PHM conference*, 2012, Conference Proceedings.
- [6] H. M. Elattar, H. K. Elminir, and A. M. Riad, “Prognostics: a literature review,” *Complex & Intelligent Systems*, vol. 2, no. 2, pp. 125–154, 2016. [Online]. Available: <https://doi.org/10.1007/s40747-016-0019-3>
- [7] S. Agarwal, “Understanding the data science lifecycle,” 2018. [Online]. Available: <https://www.sudeep.co/data-science/2018/02/09/Understanding-the-Data-Science-Lifecycle.html>
- [8] L. Berti-équille, “Qualité des données,” *Techniques de l’ingénieur Bases de données*, vol. base documentaire : TIB309DUO, no. ref. article : h3700, 2018. [Online]. Available: <https://www.techniques-ingenieur.fr/base-documentaire/technologies-de-l-information-th9/bases-de-donnees-42309210/qualite-des-donnees-h3700/>
- [9] M. Sage and Y. F. Zhao, “Is machine learning suitable to improve my process?” *McGill University: Montreal, QC, Canada*, 2020.
- [10] T. Beysolow II, *Applied Natural Language Processing with Python: Implementing Machine Learning and Deep Learning Algorithms for Natural Language Processing*. Apress, 2018.
- [11] A. Dima, S. Lukens, M. Hodkiewicz, T. Sexton, and M. P. Brundage, “Adapting natural language processing for technical text,” *Applied AI Letters*, vol. 2, no. 3, 2021. [Online]. Available: <http://dx.doi.org/10.1002/AIL2.33>
- [12] M. Payette, “Étude des facteurs affectant la fiabilité des transformateurs de puissance,” Master’s thesis, Université du Québec à Trois-Rivières, 2020.
- [13] K. Unsworth, E. Adriasola, A. Johnston-Billings, A. Dmitrieva, and M. Hodkiewicz, “Goal hierarchy: Improving asset data quality by improving motivation,” *Reliability Engineering & System Safety*, vol. 96, no. 11, pp. 1474–1481, 2011.

- [14] T. Sexton, M. Hodkiewicz, M. P. Brundage *et al.*, “Categorization errors for data entry in maintenance work-orders,” in *Proceedings of the Annual Conference of the PHM Society*, vol. 11, 2019.
- [15] M. P. Brundage, T. Sexton, M. Hodkiewicz, A. Dima, and S. Lukens, “Technical language processing: Unlocking maintenance knowledge,” *Manufacturing Letters*, vol. 27, 2021. [Online]. Available: <http://dx.doi.org/10.1016/j.mfglet.2020.11.001>
- [16] Y. Xia, N. Jazdi, M. Weyrich, and S. Germany, “Enhance fmea with large language models for assisted risk management in technical processes and products,” *Authorea Preprints*, 2024.
- [17] S. R. Andrade and H. S. Walsh, “Machine learning framework for hazard extraction and analysis of trends (heat) in wildfire response,” *Safety science*, vol. 167, p. 106252, 2023.
- [18] M. Payette and G. Abdul-Nour, “Asset management, reliability and prognostics modeling techniques,” *Sustainability*, vol. 15, no. 9, p. 7493, 2023.
- [19] —, “Machine learning applications for reliability engineering: A review,” *Sustainability*, vol. 15, no. 7, p. 6270, 2023.
- [20] V. M. D. Toni, L. A. G. da Costa, M. E. T. Botelho, M. M. Belinelli, M. Rodrigues, and E. Rigoni, “The natural language processing (nlp) algorithm integrated with fmeca analysis applied in the maintenance plans development to power generation system assets,” *Proceedings of the International Conference on Industrial Engineering & Operations Management*, pp. 170–179, 2022. [Online]. Available: <https://biblioproxy.uqtr.ca/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=aci&AN=161074803&site=ehost-live>
- [21] B. Stephen, X. Jiang, and S. D. J. McArthur, “Extracting distribution network fault semantic labels from free text incident tickets,” *IEEE Transactions on Power Delivery*, vol. 35, no. 3, pp. 1610–1613, 2020.
- [22] M.-A. Lutz, B. Schäfermeier, R. Sexton, M. Sharp, A. Dima, S. Faulstich, and J. M. Aluri, “Kpi extraction from maintenance work orders—a comparison of expert labeling, text classification and ai-assisted tagging for computing failure rates of wind turbines,” *Energies (19961073)*, vol. 16, no. 24, p. 7937, 2023.
- [23] W. J. Stevenson, C. Benedetti, and Y. A. Youssef, *La gestion des opérations: produits et services*. Chenelière McGraw-Hill, 2012.
- [24] J. E. Amadi-Echendu, K. Brown, R. Willett, and J. Mathew, *Definitions, Concepts and Scope of Engineering Asset Management*. London: Springer-Verlag London, 2011.
- [25] I. O. for Standardization, “Iso 55000, 55001 and 55002 asset management standards,” ISO, Standard, 2015.
- [26] I. of Public Works Engineering Australasia IPWEA, “International infrastructure management manual (iimm),” IPWEA, Standard, 2015.
- [27] IAM, “Asset management - an anatomy,” *Institute of Asset Management*, vol. V3, 2015.

- [28] S. R. Khuntia, J. L. Rueda, S. Bouwman, and M. A. van der Meijden, "A literature survey on asset management in electrical power [transmission and distribution] system," *International Transactions on Electrical Energy Systems*, vol. 26, no. 10, pp. 2123–2133, 2016.
- [29] S. R. Khuntia, J. L. Rueda, and M. A. van der Meijden, "Smart asset management for electric utilities: Big data and future," in *Asset Intelligence through Integration and Interoperability and Contemporary Vibration Engineering Technologies: Proceedings of the 12th World Congress on Engineering Asset Management and the 13th International Conference on Vibration Engineering and Technology of Machinery*. Springer, 2017, Conference Proceedings, pp. 311–322.
- [30] G. Biard and G. A. Nour, "Industry 4.0 contribution to asset management in the electrical industry," *Sustainability*, vol. 13, no. 18, p. 10369, 2021.
- [31] E. Zio, *Introduction To The Basics Of Reliability And Risk Analysis*, An, ser. Series on Quality, Reliability and Engineering Statistics. Singapore: World Scientific, 2007.
- [32] M. Lazzaroni, L. Cristaldi, L. Peretto, P. Rinaldi, and M. Catelani, *Reliability Engineering: Basic Concepts and Applications in ICT*. Berlin, Heidelberg: Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.
- [33] L. M. Leemis, *Reliability: Probabilistic Models and Statistical Methods*, 2nd ed. Lawrence M. Leemis, 2009. [Online]. Available: <https://books.google.ca/books?id=ER5IPwAACAAJ>
- [34] P. Dersin, "Prognostics & health management for railways: Experience, opportunities, challenges," 2020.
- [35] P. Stender, "The use of heuristic strategies in modelling activities," *ZDM*, vol. 50, no. 1, pp. 315–326, 2018.
- [36] G. Biau, M. Herzlich, and J. Droniou, *Mathématiques et statistique pour les sciences de la nature : : modéliser, comprendre et appliquer*, ser. Collection enseignement sup. Mathématiques. Les Ulis: EDP Sciences, 2010.
- [37] C. R. Hicks and K. V. Turner, *Fundamental concepts in the design of experiments*, 5th ed. New York: Oxford University Press, 1999.
- [38] E. E. Lewis, *Introduction to reliability engineering*, 2nd ed., ser. Reliability engineering. New York, N.Y.: J. Wiley and Sons, 1996.
- [39] S. Abdalnour, C. Baril, G. Abdalnour, and S. Gamache, "Implementation of industry 4.0 principles and tools: Simulation and case study in a manufacturing sme," *Sustainability*, vol. 14, no. 10, p. 6336, 2022.
- [40] M. Payette, G. Abdul-Nour, T. J.-M. Meango, and A. Côté, *Improving Maintenance Data Quality: Application of Natural Language Processing to Asset Management*. Springer, 2023, pp. 582–589.
- [41] E. Živanović, M. Živković, and S. Veljković, "Study of breakdown voltage stability of gas-filled surge arresters in the presence of gamma radiation," *Electronics*, vol. 11, no. 15, p. 2447, 2022.

- [42] F. Van Harmelen, V. Lifschitz, and B. Porter, *Handbook of knowledge representation*. Elsevier, 2008.
- [43] S. Boral, I. Howard, S. K. Chaturvedi, K. McKee, and V. N. A. Naikan, “An integrated approach for fuzzy failure modes and effects analysis using fuzzy ahp and fuzzy mairca,” *Engineering Failure Analysis*, vol. 108, p. 104195, 2020.
- [44] M. J. Kalathil, V. Renjith, and N. R. Augustine, “Failure mode effect and criticality analysis using dempster shafer theory and its comparison with fuzzy failure mode effect and criticality analysis: A case study applied to lng storage facility,” *Process Safety and Environmental Protection*, vol. 138, pp. 337–348, 2020.
- [45] P. D. T. O’Connor and A. Kleyner, *Practical Reliability Engineering*. Chichester, UK: Chichester, UK: John Wiley & Sons, Ltd, 2011.
- [46] M. L. Gámiz, K. B. Kulasekera, N. Limnios, and B. H. Lindqvist, *Applied Nonparametric Statistics in Reliability*. London: Springer London, London, 2011.
- [47] N. I. of Standards and Technology, “Nist/sematech e-handbook of statistical methods,” 2003 2003. [Online]. Available: <http://www.itl.nist.gov/div898/handbook/index.htm>
- [48] N. Gorjian, L. Ma, M. Mittinty, P. Yarlagadda, and Y. Sun, *A review on reliability models with covariates*. Springer, 2010, pp. 385–397.
- [49] T. Therneau and P. Grambsch, *Modeling Survival Data: Extending The Cox Model*, ser. Statistics for Biology and Health. New York: Springer, 2000, vol. 48.
- [50] L. D. Amorim and J. Cai, “Modelling recurrent events: a tutorial for analysis in epidemiology,” *International Journal of Epidemiology*, vol. 44, no. 1, pp. 324–333, 2014. [Online]. Available: <https://doi.org/10.1093/ije/dyu222>
- [51] D. Martin, J. Marks, T. K. Saha, O. Krause, and N. Mahmoudi, “Investigation into modeling australian power transformer failure and retirement statistics,” *IEEE Transactions on Power Delivery*, vol. 33, no. 4, pp. 2011–2019, 2018.
- [52] O. Blancke, G. McCarthy, M. Payette, L. Bibeau, J.-F. Boudreau, A. Côté, D. Komljenovic, D. Messaoudi, and B. Souilah, “Performance of reliability models for repairable systems: dealing with information uncertainty,” in *2020 Annual Reliability and Maintainability Symposium (RAMS)*. IEEE, 2020, Conference Proceedings, pp. 1–6.
- [53] S. R. Khuntia, F. Zghal, R. Bhuyan, E. Schenkel, P. Duvivier, O. Blancke, and W. Krasny, “Use of survival analysis and simulation to improve maintenance planning of high voltage instrument transformers in the dutch transmission system,” *arXiv preprint arXiv:2301.01239*, 2023.
- [54] M. Pecht and S. Kumar, “Data analysis approach for system reliability, diagnostics and prognostics,” in *Pan Pacific Microelectronics Symposium*, vol. 795, 2008, Conference Proceedings, pp. 1–9.
- [55] J. G. Fernández, F. Olivencia, J. Ferrero, A. C. Márquez, and G. C. García, “Analysis of dynamic reliability surveillance: a case study,” *IMA Journal of management Mathematics*, vol. 29, no. 1, pp. 53–67, 2018.

- [56] M. Fernandes, A. Canito, V. Bolón-Canedo, L. Conceição, I. Praça, and G. Marreiros, "Data analysis and feature selection for predictive maintenance: A case-study in the metallurgic industry," *International Journal of Information Management*, vol. 46, pp. 252–262, 2019, international Journal of Information Management.
- [57] O. O. Aremu, A. S. Palau, A. K. Parlikad, D. Hyland-Wood, and P. R. McAree, "Structuring data for intelligent predictive maintenance in asset management," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 514–519, 2018. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S2405896318314952>
- [58] J. Hurwitz and D. Kirsch, *Machine Learning for dummies*. John Wiley & Sons, Inc., 2018.
- [59] M. Haenlein and A. Kaplan, "A brief history of artificial intelligence: On the past, present, and future of artificial intelligence," *California management review*, vol. 61, no. 4, pp. 5–14, 2019.
- [60] G. Press, "A very short history of artificial intelligence (ai)," 2016. [Online]. Available: <https://www.forbes.com/sites/gilpress/2016/12/30/a-very-short-history-of-artificial-intelligence-ai/?sh=77517a5e6fba>
- [61] IBM, "Big data for the intelligence community," IBM, Report, 2013.
- [62] S. J. Russell and P. Norvig, *Artificial intelligence: a modern approach*. pearson, 2016.
- [63] A. Fahad, N. Alshatri, Z. Tari, A. Alamri, I. Khalil, A. Y. Zomaya, S. Foufou, and A. Bouras, "A survey of clustering algorithms for big data: Taxonomy and empirical analysis," *IEEE transactions on emerging topics in computing*, vol. 2, no. 3, pp. 267–279, 2014.
- [64] N. Akalin and A. Loutfi, "Reinforcement learning approaches in social robotics," *Sensors*, vol. 21, no. 4, p. 1292, 2021.
- [65] H. Zhang and T. Yu, *Taxonomy of reinforcement learning algorithms*. Springer, 2020, pp. 125–133.
- [66] N. Zhu, X. Liu, Z. Liu, K. Hu, Y. Wang, J. Tan, M. Huang, Q. Zhu, X. Ji, and Y. Jiang, "Deep learning for smart agriculture: Concepts, tools, applications, and opportunities," *International Journal of Agricultural and Biological Engineering*, vol. 11, no. 4, pp. 32–44, 2018.
- [67] W. M. Van der Aalst, *Process mining: data science in action*. Springer, 2016.
- [68] G. D. Murphy, "Improving the quality of manually acquired data: Applying the theory of planned behaviour to data quality," *Reliability Engineering & System Safety*, vol. 94, no. 12, pp. 1881–1886, 2009. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0951832009001124>
- [69] M. G. Pecht and M. Kang, "Prognostics and health management of electronics," 2018. [Online]. Available: <https://ieeexplore.ieee.org/servlet/opac?bknumber=8470990>

- [70] J. Grus, *Data science from scratch: first principles with python*. O'Reilly Media, 2019.
- [71] W. L. Martinez, A. R. Martinez, and J. L. Solka, *Exploratory data analysis with MATLAB®*. Chapman and Hall/CRC, 2017.
- [72] S. Ozdemir and D. Susarla, "Feature engineering made easy : identify unique features from your dataset in order to build powerful machine learning systems," 2018.
- [73] C. Yi-Fan, L. Yi-Kuei, and H. Cheng-Fu, "Using deep neural networks to evaluate the system reliability of manufacturing networks," *International Journal of Performability Engineering*, vol. 17, no. 7, pp. 600–608, 2021.
- [74] M. A. Nabian and H. Meidani, "Deep learning for accelerated seismic reliability analysis of transportation networks," *Computer-Aided Civil & Infrastructure Engineering*, vol. 33, no. 6, pp. 443–458, 2018.
- [75] L. I. N. Wei-Ting, C. Hsiang-Yun, Y. Chia-Lin, L. I. N. Meng-Yao, L. Kai, H. U. Han-Wen, C. Hung-Sheng, L. I. Hsiang-Pang, C. Meng-Fan, T. Yen-Ting, and N. Chin-Fu, "Dl-rsim: A reliability and deployment strategy simulation framework for reram-based cnn accelerators," *ACM Transactions on Embedded Computing Systems*, vol. 21, no. 3, pp. 1–29, 2022.
- [76] K. M. Gritsyuk and V. I. Gritsyuk, "Convolutional and long short-term memory neural networks based models for remaining useful life prediction," *International Journal on Information Technologies & Security*, vol. 14, no. 1, pp. 61–76, 2022.
- [77] A. Saxena and K. Goebel, "Turbofan engine degradation simulation data set," 2008.
- [78] A. Kara, "A data-driven approach based on deep neural networks for lithium-ion battery prognostics," *Neural Computing & Applications*, vol. 33, no. 20, pp. 13 525–13 538, 2021.
- [79] B. Saha and K. Goebel, "Battery data set," *NASA AMES prognostics data repository*, 2007.
- [80] O. Kilkki, A. Kangasrääsio, R. Nikkilä, A. Alahäivälä, and I. Seilonen, "Agent-based modeling and simulation of a smart grid: A case study of communication effects on frequency control," *Engineering Applications of Artificial Intelligence*, vol. 33, pp. 91–98, 2014.
- [81] J. Wan, S. Tang, D. Li, S. Wang, C. Liu, H. Abbas, and A. V. Vasilakos, "A manufacturing big data solution for active preventive maintenance," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 2039–2047, 2017.
- [82] A. Abu-Samah, M. K. Shahzad, E. Zamai, and A. B. Said, "Failure prediction methodology for improved proactive maintenance using bayesian approach," *IFAC-PapersOnLine*, vol. 48, no. 21, pp. 844–851, 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S2405896315017619>
- [83] M. Reid, "Reliability—a python library for reliability engineering," *Version 0.5*, vol. 1, 2020.

- [84] C. Davidson-Pilon, “lifelines: survival analysis in python,” *Journal of Open Source Software*, vol. 4, no. 40, p. 1317, 2019. [Online]. Available: <https://doi.org/10.21105/joss.01317>
- [85] S. Pölsterl, “scikit-survival: A library for time-to-event analysis built on top of scikit-learn,” *Journal of Machine Learning Research*, vol. 21, no. 212, pp. 1–6, 2020. [Online]. Available: <http://jmlr.org/papers/v21/20-729.html>
- [86] C. Teubert, K. Jarvis, M. Corbetta, C. Kulkarni, and M. Daigle, “Progpy python prognostics packages,” October 2022. [Online]. Available: <https://nasa.github.io/progpy>
- [87] S. Seabold and J. Perktold, “statsmodels: Econometric and statistical modeling with python,” in *9th Python in Science Conference*, 2010.
- [88] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [89] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015, software available from tensorflow.org. [Online]. Available: <https://www.tensorflow.org/>
- [90] F. Chollet *et al.*, “Keras,” <https://keras.io>, 2015.
- [91] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.
- [92] S. R. Joseph, H. Hlomani, K. Letsholo, F. Kaniwa, and K. Sedimo, “Natural language processing: A review,” *International Journal of Research in Engineering and Applied Sciences*, vol. 6, no. 3, pp. 207–210, 2016.
- [93] D. Jurafsky, *Speech & language processing*. Pearson Education India, 2000.
- [94] C. Stenström, M. Aljumaili, and A. Parida, “Natural language processing of maintenance records data,” *International Journal of COMADEM*, 2015.
- [95] M. Hodkiewicz and M. T.-W. Ho, “Cleaning historical maintenance work order data for reliability analysis,” *Journal of Quality in Maintenance Engineering*, vol. 22, no. 2, pp. 146–163, 2016.
- [96] T. Sexton, M. P. Brundage, M. Hoffman, and K. C. Morris, “Hybrid datafication of maintenance logs from ai-assisted human tags,” in *2017 IEEE International Conference on Big Data*. IEEE, 2017, Conference Proceedings. [Online].

Available: <http://dx.doi.org/10.1109/BigData.2017.8258120https://ieeexplore.ieee.org/document/8258120/>

- [97] M. P. Brundage, K. C. Morris, T. Sexton, S. Mocozet, and M. Hoffman, "Developing maintenance key performance indicators from maintenance work order data," in *Proceedings of the ASME 2018 International Manufacturing Science and Engineering Conference*. American Society of Mechanical Engineers, 2018, Conference Proceedings. [Online]. Available: <http://dx.doi.org/10.1115/MSEC2018-6492https://asmedigitalcollection.asme.org/MSEC/proceedings-abstract/MSEC2018/51371/V003T02A027/274075>
- [98] T. Sexton, M. Hodkiewicz, M. P. Brundage, and T. Smoker, "Benchmarking for keyword extraction methodologies in maintenance work orders," *Annual Conference of the PHM Society*, vol. 10, no. 1, 2018. [Online]. Available: http://dx.doi.org/10.36001/PHMCONF.2018.V10I1.541https://papers.phmsociety.org/index.php/phmconf/article/download/541/phmc_18_541
- [99] H. B. Gunay, W. Shen, and C. Yang, "Text-mining building maintenance work orders for component fault frequency," *Building Research & Information*, vol. 47, no. 5, pp. 518–533, 2018. [Online]. Available: <http://dx.doi.org/10.1080/09613218.2018.1459004>
- [100] M. Navinchandran, M. E. Sharp, M. P. Brundage, and T. B. Sexton, "Studies to predict maintenance time duration and important factors from maintenance workorder data," *Annual Conference of the PHM Society*, vol. 11, no. 1, 2019. [Online]. Available: http://dx.doi.org/10.36001/PHMCONF.2019.V11I1.792https://papers.phmsociety.org/index.php/phmconf/article/download/792/phmc_19_792
- [101] M. Navinchandran, M. E. Sharp, M. P. Brundage, and T. Sexton, "Discovering critical kpi factors from natural language in maintenance work orders," *Journal of Intelligent Manufacturing*, vol. 33, no. 6, pp. 1859–1877, 2021. [Online]. Available: <http://dx.doi.org/10.1007/S10845-021-01772-5https://link.springer.com/article/10.1007/s10845-021-01772-5>
- [102] R. Sexton, "Nestor," National Institute of Standards and Technology (NIST), 2019, last updated 05/23/2019. [Online]. Available: <https://www.nist.gov/services-resources/software/nestor>
- [103] Y. Bouabdallaoui, Z. Lafhaj, P. Yim, L. Ducoulombier, and B. Bennadji, "Natural language processing model for managing maintenance requests in buildings," *Buildings*, vol. 10, no. 9, p. 160, 2020. [Online]. Available: http://dx.doi.org/10.3390/buildings10090160https://mdpi-res.com/d_attachment/buildings/buildings-10-00160/article_deploy/buildings-10-00160.pdf?version=1599487176
- [104] B. Tyler and M. Hodkiewicz, "Semi-automated estimation of reliability measures from maintenancework order records," in *Proceedings of the 6th European Conference of the Prognostics and Health Management Society 2021*, 2021, Conference Proceedings.

- [105] C. Annalisa, B. Coline, P. Lynn, B. Michael, and S. Thurston, "The impact of data quality on maintenance work order analysis: A case study in hvac work durations," 2021.
- [106] P. Nojedehe, W. O'brien, and H. B. Gunay, "Benchmarking and visualization of building portfolios by applying text analytics to maintenance work order logs," *Science and Technology for the Built Environment*, vol. 27, no. 6, pp. 756–775, 2021. [Online]. Available: <http://dx.doi.org/10.1080/23744731.2021.1913957>
- [107] S. M. R. Naqvi, C. Varnier, J.-M. Nicod, N. Zerhouni, and M. Ghufan, "Leveraging free-form text in maintenance logs through bert transfer learning," in *Progresses in Artificial Intelligence & Robotics: Algorithms & Applications: Proceedings of 3rd International Conference on Deep Learning, Artificial Intelligence and Robotics,(ICDLAIR) 2021*. Springer, 2022, Conference Proceedings, pp. 63–75.
- [108] S. M. R. Naqvi, M. Ghufan, S. Meraghni, C. Varnier, J.-M. Nicod, and N. Zerhouni, "Generating semantic matches between maintenance work orders for diagnostic decision support," *Annual Conference of the PHM Society*, vol. 14, no. 1, 2022. [Online]. Available: <http://dx.doi.org/10.36001/phmconf.2022.v14i1.3241>https://papers.phmsociety.org/index.php/phmconf/article/download/3241/phmc_22_3241
- [109] —, "Cbr-based decision support system for maintenance text using nlp for an aviation case study," in *2022 Prognostics and Health Management Conference*. IEEE, 2022, Conference Proceedings. [Online]. Available: <http://dx.doi.org/10.1109/PHM2022-London52454.2022.00067><https://ieeexplore.ieee.org/document/9808428/>
- [110] A. Deloose, G. Gysels, B. De Baets, and J. Verwaeren, "Combining natural language processing and multidimensional classifiers to predict and correct CMMS metadata," *Computers in Industry*, vol. 145, 2023, publisher: Elsevier B.V. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85144614540&doi=10.1016%2fj.compind.2022.103830&partnerID=40&md5=b56cda1dff4acd006891db2d5d7e34e>
- [111] B. Hershowitz, M. Hodkiewicz, T. Bikaun, M. Stewart, and W. Liu, "Causal knowledge extraction from long text maintenance documents," *Computers in Industry*, vol. 161, p. 104110, Oct. 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0166361524000381>
- [112] Z. Hu, X. Zhang, and H. Xiong, "Two-stage attention network for fault diagnosis and retrieval of fault logs," *Expert Systems with Applications*, vol. 249, p. 123365, Sep. 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0957417424002306>
- [113] Y. Li, Y. Liu, J. Zhang, L. Cao, and Q. Wang, "Automated analysis and assignment of maintenance work orders using natural language processing," *Automation in Construction*, vol. 165, p. 105501, Sep. 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0926580524002371>

- [114] M. D’Orazio, G. Bernardini, and E. D. Giuseppe, “Influence of pre-processing methods on the automatic priority prediction of native-language end-users’ maintenance requests through machine learning methods,” *Journal of Information Technology in Construction (ITcon)*, vol. 29, no. 6, pp. 99–116, Mar. 2024. [Online]. Available: <http://itcon.org/paper/2024/6>
- [115] C. Woods, M. Selway, T. Bikaun, M. Stumptner, and M. Hodkiewicz, “An ontology for maintenance activities and its application to data quality,” *Semantic Web*, vol. 15, no. 2, pp. 319–352, Jan. 2024, publisher: IOS Press. [Online]. Available: <https://content.iospress.com/articles/semantic-web/sw233299>
- [116] V. Giordano and G. Fantoni, “Decomposing maintenance actions into sub-tasks using natural language processing: A case study in an Italian automotive company,” *Computers in Industry*, vol. 164, p. 104186, Jan. 2025. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0166361524001143>
- [117] S. M. R. Naqvi, M. Ghufuran, C. Varnier, J.-M. Nicod, K. Javed, and N. Zerhouni, “Unlocking maintenance insights in industrial text through semantic search,” *Computers in Industry*, vol. 157-158, p. 104083, May 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0166361524000113>
- [118] K. Fort, “Collaborative annotation for reliable natural language processing : technical and sociological aspects,” 2016.
- [119] X. Yang, X. He, Y. Liang, Y. Yang, S. Zhang, and P. Xie, “Transfer learning or self-supervised learning? a tale of two pretraining paradigms,” *arXiv preprint arXiv:2007.04234*, 2020.
- [120] M.-C. De Marneffe, C. D. Manning, J. Nivre, and D. Zeman, “Universal dependencies,” *Computational linguistics*, vol. 47, no. 2, pp. 255–308, 2021.
- [121] J. Nivre, M.-C. De Marneffe, F. Ginter, Y. Goldberg, J. Hajic, C. D. Manning, R. McDonald, S. Petrov, S. Pyysalo, N. Silveira *et al.*, “Universal dependencies v1: A multilingual treebank collection,” in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, 2016, pp. 1659–1666.
- [122] M. Honnibal and M. Johnson, “An improved non-monotonic transition system for dependency parsing,” in *Proceedings of the 2015 conference on empirical methods in natural language processing*, 2015, pp. 1373–1378.
- [123] R. Řehůřek and P. Sojka, “Software Framework for Topic Modelling with Large Corpora,” in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Valletta, Malta: ELRA, May 2010, pp. 45–50, <http://is.muni.cz/publication/884893/en>.
- [124] S. Loria *et al.*, “textblob documentation,” *Release 0.15*, vol. 2, no. 8, 2018.
- [125] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, “The stanford corenlp natural language processing toolkit,” in *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, 2014, pp. 55–60.

- [126] M. Gardner, J. Grus, M. Neumann, O. Tafjord, P. Dasigi, N. Liu, M. Peters, M. Schmitz, and L. Zettlemoyer, “Allennlp: A deep semantic natural language processing platform,” *arXiv preprint arXiv:1803.07640*, 2018.
- [127] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz *et al.*, “Transformers: State-of-the-art natural language processing,” in *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, 2020, pp. 38–45.
- [128] A. Bansal, “Advanced natural language processing with tensorflow 2 build real-world effective nlp... applications using ner, rnns, seq2seq models, tran,” 2021. [Online]. Available: <https://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=2746412>
- [129] Z. Liu, Y. Lin, and M. Sun, “Representation learning for natural language processing,” 2020.
- [130] P. M. Nadkarni, L. Ohno-Machado, and W. W. Chapman, “Natural language processing: an introduction,” *Journal of the American Medical Informatics Association*, vol. 18, no. 5, pp. 544–551, 2011.
- [131] J. Thanaki, *Python natural language processing*. Packt Publishing Ltd, 2017.
- [132] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.
- [133] S. Ravichandiran, “Getting started with google bert : build and train state-of-the-art natural language processing models using bert,” 2021. [Online]. Available: <http://public.eblib.com/choice/PublicFullRecord.aspx?p=6463518https://ebookcentral.proquest.com/lib/washington/detail.action?docID=6463518http://www.vlebooks.com/vleweb/product/openreader?id=none&isbn=9781838826239http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=2736061>
- [134] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” *Advances in neural information processing systems*, vol. 26, 2013.
- [135] J. Pennington, R. Socher, and C. D. Manning, “Glove: Global vectors for word representation,” in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.
- [136] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching word vectors with subword information,” *Transactions of the association for computational linguistics*, vol. 5, pp. 135–146, 2017.
- [137] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.

- [138] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, “Deep contextualized word representations,” in *North American Chapter of the Association for Computational Linguistics*, 2018.
- [139] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, “Improving language understanding by generative pre-training,” 2018.
- [140] J. Lee, Z. Dai, X. Ren, B. Chen, D. Cer, J. R. Cole, K. Hui, M. Boratko, R. Kapadia, W. Ding, Y. Luan, S. M. K. Duddu, G. H. Abrego, W. Shi, N. Gupta, A. Kusupati, P. Jain, S. R. Jonnalagadda, M.-W. Chang, and I. Naim, “Gecko: Versatile text embeddings distilled from large language models,” 2024. [Online]. Available: <https://arxiv.org/abs/2403.20327>
- [141] P. BehnamGhader, V. Adlakha, M. Mosbach, D. Bahdanau, N. Chapados, and S. Reddy, “Llm2vec: Large language models are secretly powerful text encoders,” 2024. [Online]. Available: <https://arxiv.org/abs/2404.05961>
- [142] H. Li, F. Deng, J. Lu, T. Zhang, and H. Li, “An Application of Automatic Text Revision for Power Defect Log,” *Journal of Physics: Conference Series*, vol. 1757, no. 1, p. 012027, Jan. 2021, publisher: IOP Publishing. [Online]. Available: <https://dx.doi.org/10.1088/1742-6596/1757/1/012027>
- [143] G. Aurélien, “Hands-on machine learning with scikit-learn & tensorflow,” *Geron Aurelien*, vol. 134, pp. 145–150, 2017.
- [144] M. Akbari, P. Khazaei, I. Sabetghadam, and P. Karimifard, “Failure modes and effects analysis (fmea) for power transformers,” in *28th International power system conference*, vol. 2013, 2013, pp. 1–7.
- [145] F. Iafrate, *Artificial intelligence and big data: The birth of a new intelligence*. John Wiley & Sons, 2018.
- [146] M. Ho, M. R. Hodkiewicz, C. F. Pun, J. Petchey, and Z. Li, “Asset data quality—a case study on mobile mining assets,” in *Engineering Asset Management-Systems, Professional Practices and Certification: Proceedings of the 8th World Congress on Engineering Asset Management (WCEAM 2013) & the 3rd International Conference on Utility Management & Safety (ICUMAS)*, ser. Engineering Asset Management - Systems, Professional Practices and Certification, Springer. Springer International Publishing, 2015, Conference Proceedings, pp. 335–349.
- [147] R. Krishnan, “Reliability analysis of k-out-of-n: G system: A short review,” *International Journal of Engineering and Applied Sciences (IJEAS)*, vol. 7, no. 2, pp. 21–24, 2020.
- [148] P. Odeyar, D. B. Apel, R. Hall, B. Zon, and K. Skrzypkowski, “A review of reliability and fault analysis methods for heavy equipment and their components used in mining,” *Energies*, vol. 15, no. 17, p. 6263, 2022.
- [149] R. K. Perrons and J. W. Jensen, “Data as an asset: What the oil and gas sector can learn from other industries about “big data”,” *Energy Policy*, vol. 81, pp. 117–121, 2015.

- [150] A. Jimenez-Cortadi, I. Irigoien, F. Boto, B. Sierra, and G. Rodriguez, “Predictive maintenance on the machining process and machine tool,” *Applied Sciences*, vol. 10, no. 1, p. 224, 2019.
- [151] Y. LeCun and I. Misra, “Self-supervised learning: The dark matter of intelligence,” March 4 2021, research blog post. [Online]. Available: <https://ai.meta.com/blog/self-supervised-learning-the-dark-matter-of-intelligence/>
- [152] Y. Liang, O. Blancke, M. Gaha, A. Côté, G. St-Jean, and E. Aïmeur, “Automatic database alignment method to improve failure data quality,” in *2022 Annual Reliability and Maintainability Symposium (RAMS)*. IEEE, 2022, pp. 1–6.

Annexe A - Liste des publications

A.1 Articles de revue

1. Payette, M., Abdul-Nour, G., Côté, A., Meango, T., & Diago, M. *Integrating Failure Modes and Effect Analysis for Technical Language Processing*. Available at SSRN 5007553.
— Soumis (pre-print)
2. Payette, M., & Abdul-Nour, G. (2023). *Machine learning applications for reliability engineering: A review*. Sustainability, 15(7), 6270.
— Facteur d'impact (5 ans) : 3.6
— Cité par : 17
3. Payette, M., & Abdul-Nour, G. (2023). *Asset management, reliability and prognostics modeling techniques*. Sustainability, 15(9), 7493.
— Facteur d'impact (5 ans) : 3.6
— Cité par : 9

A.2 Articles de conférence

1. Payette, M., Abdul-Nour, G., Meango, T., & Diago, M., Côté, A. (2025). *Reclassification des Données de Maintenance par les LLMs : Impact sur l'Évaluation Statistique de la Fiabilité des Systèmes*.
— Soumis à la conférence CIGI-QUALITA-MOSIM
2. Payette, M., Abdul-Nour, G., Meango, T., & Diago, M., Côté, A. (2025). *Enhancing Maintenance Data Analytics: A Novel FMEA-NLP Integration*. In 2025 Annual Reliability and Maintainability Symposium (RAMS). IEEE.
— Proceedings à venir
— Cité par : 0
3. Payette, M., Abdul-Nour, G., Meango, T., & Diago, M., Côté, A. (2025). *Leveraging Natural Language Processing for Enhanced Maintenance Data in Power System Management*. The 51st International Conference on Computers and Industrial Engineering, Sydney, Australia, October 9th to 11th December 2024.
— Proceedings à venir
— Cité par : (à venir)
4. Payette, M., Abdul-Nour, G., Meango, T. J. M., & Côté, A. (2022, October). *Improving maintenance data quality: Application of natural language processing to asset management*. In World Congress on Engineering Asset Management (pp. 582-589). Cham: Springer International Publishing.
— Cité par : 5

Annexe B - Asset management, reliability and prognostics modeling techniques

M. Payette and G. Abdul-Nour, Sustainability, vol. 15, no. 9, p. 7493, 2023.

Ce chapitre correspond à l'article publié dans la revue Sustainability, publié le 3 mai 2023.

Abstract

In recent years, reliability engineering has seen significant growth in data-driven modeling, mainly due to the democratization of sensing technologies, big data processing, and computing capabilities. It has also seen a paradigm shift, with Engineering of Asset Management (EAM) becoming widely accepted as a high-level framework to support corporate policies and strategies. The rapid evolution of research leads to the development of multiple research communities, making it difficult for the uninitiated to navigate the literature. Indeed, system reliability encompasses several research subfields that focus on maximizing the life cycle of assets, including Reliability, Availability, Maintainability, and Safety (RAMS), Prognostics and Health Management (PHM), and Engineering of Asset Management. This article proposes a review of these concepts with the aim of identifying the different scientific communities, what differentiates them, and what connects them. It also addresses RAMS and PHM modeling techniques and highlights the significance of these disciplines in ensuring the functioning of complex systems. In summary, this article aims to clarify the interrelationship between the topics of reliability engineering, to simplify the search and selection for modeling methods.

Mots clés: Engineering of asset management; prognostic and health management; reliability; modeling.

Introduction

Reliability engineering is a relatively old research field. In fact, this scientific discipline has been in development since the 1950s. This area, based on the application of probabilities and statistics, is now omnipresent in sectors such as military, consumer, energy, etc. The success of the Japanese car manufacturers is a good example that reliability is one of the most important criteria for consumers. Over time, the field has evolved greatly and has influenced the emergence of complementary disciplines, including business management approaches. In addition, technological advances have also greatly contributed to the formation of new topics in reliability. As a result, there are now several domains and denominations, which makes research tedious. These include the domain of RAMS (reliability, availability, maintainability, and safety), PHM (prognostic and health management), AM (asset management), etc. There is little literature that defines the RAMS ecosystem in relation to PHM and asset management and the techniques inherent to these domains. The objective of this work is to review the concepts of reliability engineering and identify the different scientific communities related to the topic. It also highlights the importance of an integrated approach in the functioning of complex systems through the coordination of RAMS and PHM modeling techniques. By clarifying the interrelationships between these topics, the article seeks to simplify the search and selection of modeling methods for researchers and practitioners. The article is structured as follows: Section B presents the three different research topics and covers a model for the hierarchy of organizational decision-making, engineering of asset management, and defining the basics of maintenance and reliability. Section B defines different types of modeling methods, namely mathematical statistical and qualitative modeling. Sections B and B focus on the classical modeling methods used, respectively, in the field

of RAMS and PHM. Section B illustrates the differences and complementarities in the modeling techniques and in the philosophy of the respective fields.

Research Topics in Reliability Engineering

Hierarchy of Organizational Decision-Making

Every enterprise must have a mission, objectives, and an action plan to achieve its goals. This section presents a hierarchical model of decision-making in an organization. In that respect, it is possible to put this model into perspective with asset management, RAMS and PHM. Figure B.1, translated from [23], illustrates the hierarchy between mission and business strategies. According to their definition, the mission statement is a guide for formulating a company's strategies. It is intended to ensure that everyone in the company and at every level of management knows the purpose of the company. Business objectives are defined according to the mission and provide a clear direction for the organization. Organizational strategies are plans to achieve the company's objectives. They are global strategies that guide the entire organization. The functional strategies, which are related to the five functions of the company (finance, marketing, human resources, operations, and production) support the organizational strategies, the tactical strategies regroup the methods, and the action plans to achieve them [23].

The hierarchy of these strategies is related to the time frame in which they are deployed, in addition to their scope and level of detail. Indeed, the mission is established at a high hierarchical level and serves as a long-term guide for an organization. Its scope is very broad, but its level of detail is low. In contrast, tactical strategies have a very low hierarchical level and a short time frame, but the level of detail is high.



FIGURE B.1 – Hierarchy of decisions.

Engineering of Asset Management

It is important to define asset management, which represents a higher level of decision-making. By definition, an asset is an object or entity to which a value is assigned. An engineering asset has real value to an organization; it can be equipment, inventory, buildings, etc. [24, 25]. Engineering of Asset Management (EAM) is the field that encompasses all organizational activities aimed at realizing the value of an asset. It is not to be confused with Financial Asset Management, which concerns investment and banking assets and not physical assets. The asset management system allows for the coordination of activities with the goal of achieving EAM objectives. The objectives of the EAM system must align with the strategic objectives of the company [26]. This involves balancing the costs, risks, opportunities, and benefits of performance [25]. The ISO standard also establishes the fundamental principles

of asset management. A central concept in EAM is value and how the asset generates it. According to the ISO, this should be defined by the organization and its stakeholders, based on business objectives. Furthermore, value varies from one organization to another and is mainly driven by the business context and needs [25]. Therefore, the notion of alignment is important; the EAM must be aligned with the business objectives, based on this notion of value. The objectives must be translated into management policies and strategies [27]. The impact of asset management is not limited to the financial aspect. It plays a significant role in sustainable development; the long-term vision of AM life cycle optimization reduces waste by opting for a more robust and durable design. In addition, optimizing replacements reduces unnecessary maintenance and replacement of components that are still in good condition, reducing the carbon footprint of assets throughout their lifespan. In the perspective of the previous section, asset management engineering is established at an organizational strategy level, given that its scope extends to all business functions. Concrete examples are given in [28], where the author reviews the concepts of asset management applied to large power systems. He presents various application cases at different levels of enterprise management to highlight the benefits of an integrated approach to AM. In [29], he extends the concepts of asset management to the use of connectivity and big data. Furthermore, he discusses advances in monitoring, sensor technologies, and data science, and how they will be applied in support of asset management. Additionally, in [30], the author proposes a review of the concepts of Industry 4.0 in relation to asset management. The article focuses on applications for the electrical industry.

Failure, Maintenance, and Reliability

Failure

From the point of view of EAM, a failure means that an asset loses its functions temporarily or permanently and no longer generates value for its organization. Failure is an inevitable event for products and systems [31]. The failure of a system can be caused by internal factors such as wear, corrosion, etc., or by an external factor, for example, a tree falling on the power transmission line. Therefore, the study of degradation mechanisms and failure modes is essential to optimize the asset life cycle. Knowing when and why a system loses its functions allows managers to implement appropriate solutions. In order to maintain its assets, an organization might employ a different maintenance strategy, as described in the following section.

Maintenance

A failure is an event that prevents the system from performing its intended functions [4]. Maintenance is the combination of means used to ensure that the system can continue to perform these functions. It includes all the activities involved in planning, managing, and executing the maintenance of equipment in order to achieve these business objectives.

Preventive maintenance aims to keep the system in a functioning state, with maintenance performed before a failure occurs. If a failure occurs, corrective maintenance allows the system to be restored to operation. During an asset's life cycle, several factors deteriorate the condition of the components. These degradation mechanisms can be related to the use or to physical and environmental phenomena. When degradation is cyclic, systematic preventive maintenance will stop these degradation mechanisms before there is a failure. Systematic maintenance consists of performing maintenance at fixed intervals, and the

periodicity can depend on the number of cycles (e.g., number of kilometers traveled) or the number of hours of service. In this case, the failure modes are known, predictable, and easy to plan. In some cases, component degradation is not cyclic, but signs of aging can be observed or measured during inspection. In this case, it is referred to as condition-based maintenance. This consists of repairing the system when an inspection or monitoring reveals signs of aging, indicating the presence of failure mechanisms. Generally, condition-based maintenance occurs when the failure modes are random. Predictive maintenance also occurs when signs of wear appear. In this case, the failure modes are known, and the occurrence can be modeled using historical data and real-time sensor data. Corrective maintenance corresponds to a repair when a failure occurs. Most of the time, it concerns events that are unwanted and problematic. However, in a reliability-centered maintenance (RCM) program, an organization can choose to opt for such a strategy for certain systems that are redundant or inexpensive to replace [1]. Figure B.2, adapted from [1], shows the different types of maintenance and their characteristics, as described above.

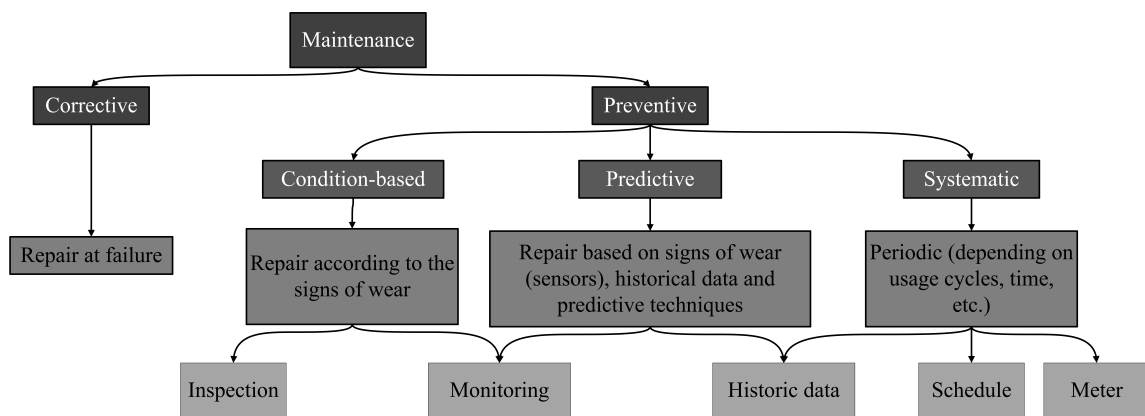


FIGURE B.2 – Maintenance classification (adapted from [1]).

Statistics of Failure

A failure is considered a random event. When it comes to maintenance, reliability is the ability of the asset to perform its intended actions within a given time interval. Thus,

reliability can be expressed as the probability that the system operates correctly during that time interval and under predefined conditions [32, 33]. The goal of the reliability study is to prevent and/or reduce the frequency of those failures by applying engineering and statistical techniques to identify and quantify the sources of faults. Reliability is often expressed as a mathematical model or as a statistical estimator, such as the mean time to failure (MTTF) or the mean time between failures (MTBF). There are two branches of engineering research that are interested in the study of system health management; reliability, availability, maintainability, and safety (RAMS) and prognostic and health management (PHM). In the literature, these disciplines are often hard to distinguish from each other. Both disciplines seek to optimize the value derived from assets throughout their life cycle. However, the models and tools used by each approach are very different. According to Pierre Dersin, RAMS focuses on the study of the general properties of a population, while PHM focuses on a single asset at a time [34].

Modeling

This section describes different modeling approaches used in EAM, RAMS, and PHM. First, the basics of mathematical modeling are presented. Then, the fundamental statistical modeling approaches are compared. Finally, qualitative modeling is discussed.

Basics of Mathematical Modeling

Modeling, in a very general way, is used to represent an object or situation, to simplify it, to understand it, and to analyze it. Mathematical modeling consists of representing a real-world situation by using mathematical equations. The mathematical modeling cycle is presented in Figure B.3, and is adapted from [2].

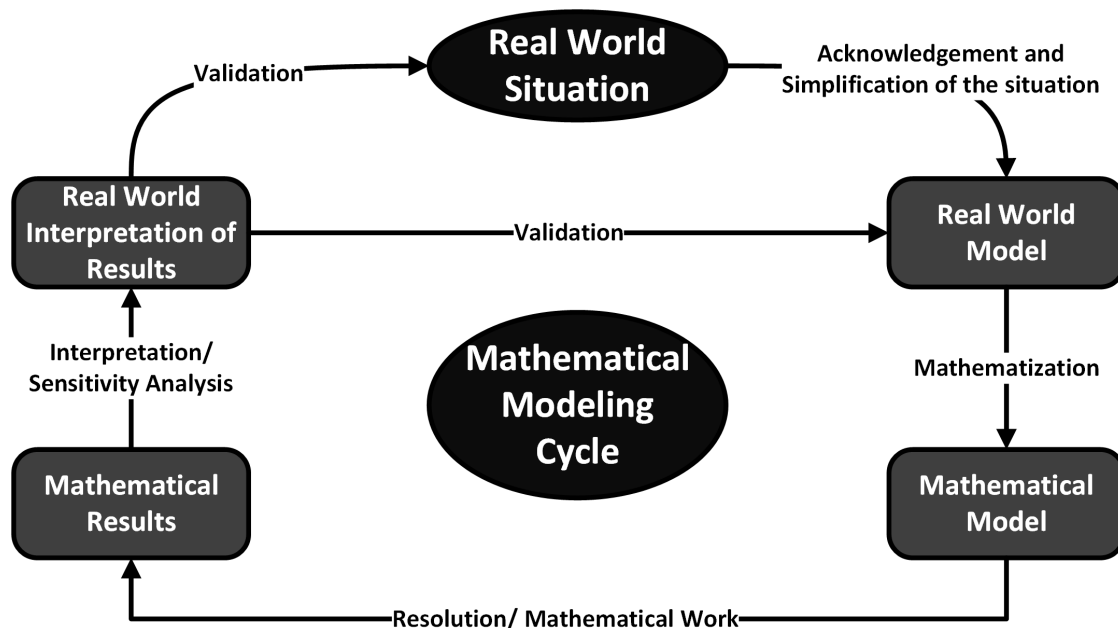


FIGURE B.3 – Modeling cycle (adapted from [2]).

This represents well the task of modeling, as well as the validation loop; the research hypotheses allow for simplifying a real situation and describing it using mathematics. The validation between the results of the model and the real situation allows for knowing if the model, although simplified, represents well the situation in question. This modeling process is repeated until the results produced are adequate [35].

Statistical Models

In statistical modeling, two types of approaches can be distinguished. Descriptive statistics aim to describe and summarize the observations of a sample using indicators, graphical construction, etc. [36]. Inferential statistics, on the other hand, aim to infer the characteristics of a population from a sample [37]. Figure B.4 presents the most common approaches from both descriptive and inferential statistics. In inferential statistics, probability distributions are used to describe the random variables in a sample and to extract their characteristics.

By knowing the type of random experiment, it is generally easy to associate the probability distribution to the sample, and to determine the parameters of this distribution from the sample. This process is often known as parametric analysis. In some cases, it is easy to define the distribution of the data, considering the operational and random context of the phenomena under study [38].

In contrast, descriptive analysis, also known as nonparametric analysis, is used to deduce the characteristics of a sample without the use of a statistical distribution. Measures of central tendency and dispersion are generally used to describe the parameters of a population under study. The construction of histograms, scatter plots, and box plots are widely used to study the behavior of systems in terms of reliability. Frequency tables can also be used to estimate the probability density function, without the need for a specific distribution law [38]. These statistical methods are used in many fields and are not limited to reliability. For example, Reference [39] presents a simulation of a manufacturing plant to test the implementation of Industry 4.0 technology. For this purpose, hypothesis testing and analysis of variance (ANOVA) are used for validation and parameter selection purposes, respectively. In [40], they use statistical metrics, particularly the confusion matrix, to evaluate the results of a text classification model. In [41], they used hypothesis testing (Pearson's Chi-square tests) to validate the choice and estimation of the parameters of a Weibull distribution. The study consists of a test bench to study the reliability of electronic components to determine the effect of gamma radiation on the breakdown voltage of gas-filled surge arresters.

Qualitative Modeling

The modeling methods presented previously are data-driven methods. However, in engineering, there are still several that are based on qualitative models. This brief section

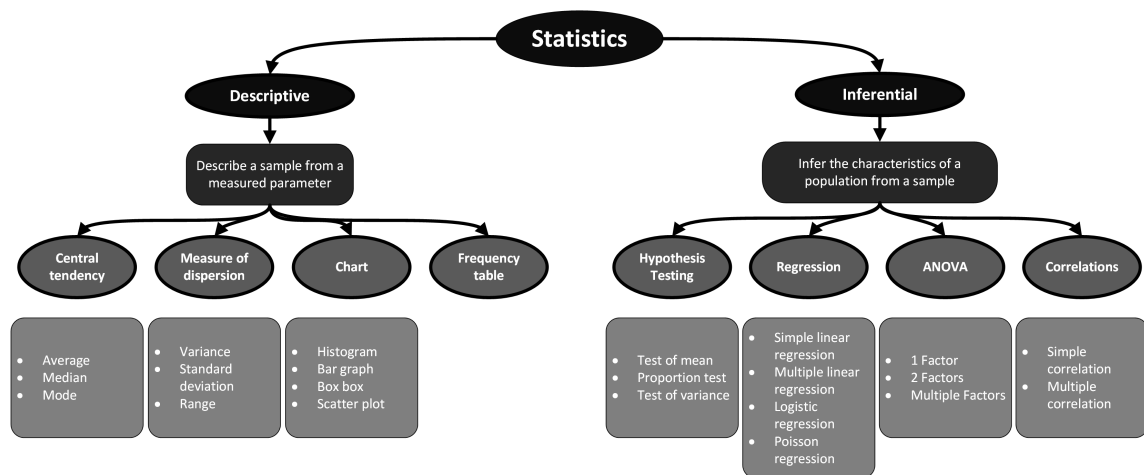


FIGURE B.4 – Common statistical modeling techniques.

will serve to describe what a qualitative model is and what distinguishes them from data-driven models. Qualitative models aim to represent systems and reasoning in a symbolic way. The system and its constraints are represented in a discrete manner, rather than with data and mathematical equations. The objective is to infer as much behavior as possible from little information. Qualitative models are based on world experience and professional expertise. These experiences are used to decide what should be included depending on the situation, the important physical phenomena, and the simplifications that are relevant [42].

Reliability, Availability, Maintainability, and Security

RAMS Basics

RAMS is the field of engineering that deals with the study of the reliability of a set of systems and/or equipment [34]. The objective is to determine the general behavior of a population to make decisions on maintenance policies, replacements, etc. For this, engineers use different statistical and qualitative modeling methods. The first step in establishing a maintenance policy for a system is to identify the risk associated with its operation [31].

To this end, the next section will present qualitative methods used to identify those risks, then, methods to quantify those risks will be presented.

Qualitative Modeling in RAMS

When analyzing the reliability of an asset, it is essential to identify the hazards associated with its operation. Section B is used to present two of the most popular qualitative methods to develop such models. These methods rely heavily on the elicitation of experts, typically those who have designed, operated, or maintained the system.

Failure Modes and Effect Analysis

The Failure Modes and Effect Analysis (FMEA) is an analysis method used to identify the failure modes of components. Failure mode is an observable cause of system malfunction. FMEA consists of the decomposition of the system into functionally independent subsystems. First, analysts must identify the different operating modes and the system's configurations of these operating modes. Then they compile this information into a table (by operating mode) and indicate the related failure modes and the effect they have on other entities in the system [31]. In addition, this simple method allows for the examination of potential system failures and the determination of preventive measures to avoid the identified problems [1]. As an example, the authors of [43] have developed a new method of multi-criteria analysis, in order to classify the failure modes according to the level of risk and the actions of mitigation and elimination of risks. The new method combines the Fuzzy Analytical Hierarchy Process (Fuzzy AHP) method and Fuzzy Multi-Attribute Ideal Real Comparative Analysis (FMAIRCA). Similarly, Reference [44] proposes a method based on Dempster–Schaffer theory to improve the traditional FMEA risk prioritization method. A case study is presented for the analysis of a liquefied natural gas terminal. FMEA is often used as a starting point

for the construction of fault trees, which will be discussed in the next section, and for reliability-based maintenance programs [31].

Fault Tree Analysis

Sometimes, no data are collected to perform a statistical analysis of a system failure. In this case, it may be relevant to look at the underlying causes and the combination of events that may lead to a particular failure. One of the methods used is the Fault Tree Analysis (FTA), where Boolean logic is applied to link events by simple logical relationships (cause and effect links). The method starts by identifying a failure mode (one at a time) and determining all the elementary events that are related to it. As this qualitative method takes advantage of Boolean algebra, the tree can be described by a set of equations and random variables. Using the rules of Boolean algebra, it is possible to calculate the probabilities of failure of the system [31]. There are many other qualitative methods for modeling the failures of a system. Cause and effect (Ishikawa) diagram, check lists, Bayesian network, event tree analysis, hazard and operability analysis (HAZOP), and the hazard index method are example of commonly used methods in reliability [4,31].

Mathematical Modeling in RAMS

Reliability can be measured with the survivor function ($S(t)$ or $R(t)$), the probability density function, the failure rate $z(t)$, etc. [4]. These probabilistic functions are representations of statistical lifetime distribution [33]. $F(t)$ commonly refers to the cumulative distribution function, that is, the probability that the item will fail over a period of time. The probability density function (PDF or $f(t)$) is the derivative of the distribution function. The survivor/reliability function is the probability that an item does not experience a failure in a specified time interval. The equation is represented in Equation (1):

$$R(t) = S(t) = 1 - F(t) = Pr(T > t) \text{ for } t \geq 0 \quad (\text{B.1})$$

On the other hand, the failure rate $z(t)$, often called the hazard rate $h(t)$, models the relationship of PDF with the survivor function [33]. The equation is represented in Equation (2):

$$z(t) = h(t) = \frac{f(t)}{R(t)} \quad (\text{B.2})$$

The cumulative hazard function $H(t)$ is the integral of the hazard function and is used primarily for data generation in simulation [33]. With regard to reliability, there are two types of systems/components: repairable and non-repairable. A non-repairable system, as the name suggests, cannot be repaired and needs to be replaced when a failure occurs. In other words, it means that a failure leads to the end of the system's life. In contrast, a repairable system can be restored to a working state as a result of maintenance actions. The same functions apply to both types of system, but different distributions are used to model their behavior.

Non-Repairable System

In the case of non-repairable systems, the survival probability or the hazard rate are computed to determine the occurrence of failure events. The mean time to failure expresses the lifetime of the system involved [45]. The hazard rate is characterized by three states, decreasing at the beginning of the life, constant during the service life, and increasing at the end of the life. These behaviors are typically presented in the form of the bathtub curve as shown in Figure B.5.

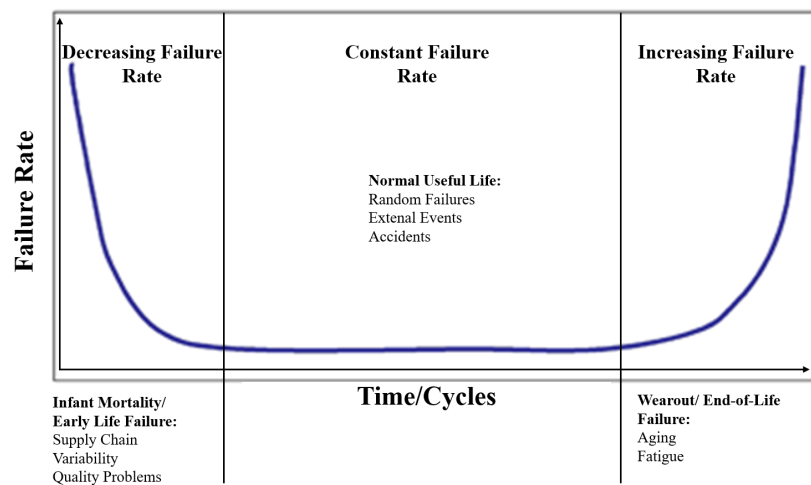


FIGURE B.5 – Bathtub curve (adapted from [3]).

Early life failures are often associated with defects due to production, assembly, etc. Various controls are used to prevent these products from reaching the customer. Regarding lifetime failures, they are often considered random failures, which means that an unusual level of stress on the system exceeds its capacity and causes the failure. Finally, when the number of failures is increasing (end-of-life), they are typically associated with aging and physical degradation of components [3]. By using the bathtub curves, it is easy for practitioners to know at what stage of the aging process the system is, and apply the maintenance program accordingly. As mentioned above, reliability is measured with probabilistic functions, which are derived from lifetime distributions. In inferential statistics, this is referred to as parametric analysis; first, the appropriate distributions are selected, and then the parameters are extracted from the data. The most commonly used distributions used in RAMS are the exponential distribution, the Weibull distribution, and the gamma distribution [33, 38]. Table B.1 shows the different representation function from these three lifetime distribution. These distributions all used three similar parameters: a scale parameter, a location parameter, and a shape parameter. The shape parameter (k or m) affects the shape of the probability distribution. The location indicates if the distribution

is shifted to right or left on the time axis. Finally, the scale parameter (λ or θ) indicates whether the distribution is more compact or expanded along the time axis [33,38].

Tableau B.1 – Representation functions from lifetime distributions.

Distribution	Cumulative $F(t)$ or CDF	Probability Density $f(t)$ or PDF	Survival $S(t)$	Hazard $h(t) = z(t)$	Cumulative Hazard $H(t)$
Exponential	$1 - e^{(-\lambda t)}$	$\lambda e^{(-\lambda t)}$	$e^{(-\lambda t)}$	λ	λt
Weibull	$1 - e^{(-\lambda t)^k}$	$k\lambda^k t^{k-1} e^{(-\lambda t)^k}$	$e^{(-\lambda t)^k}$	$k\lambda^k t^{k-1}$	λt^k
Gamma	$I(k, \lambda t)$	$\frac{\lambda(\lambda t)^{k-1} e^{-\lambda t}}{\Gamma k}$	$1 - I(k, \lambda t)$	$\frac{\lambda(\lambda t)^{k-1} e^{-\lambda t}}{\Gamma k [1 - I(k, \lambda t)]}$	$-\log(1 - I(k, \lambda t))$

Repairable System

In the case of a repairable system, it is characterized by the fact that when a failure event occurs, the system can be returned to its operating state. Reliability is then expressed as the rate of occurrence of failure (ROCOF or repair rate) or as the mean time between failures (MTBF). In contrast with the non-repairable system, the event of interest is recurrent, and the modeling techniques are chosen accordingly. In general, the stochastic process of recurrent failure events is represented by a counting process. Since an asset may experience several failures during its useful life, the frequency of occurrence may increase, decrease, or remain constant.

When the asset is repaired, three assumptions can be made about the quality of the intervention. Maintenance can be considered perfect, i.e., the device is considered as new after repair. In the case of minimal repair, the system is considered “as bad as old” after its repair, meaning that the system, rather than returning to the initial conditions, is returned to its pre-failure state [46]. Finally, imperfect maintenance lies between these two extremes.

Figure B.6, adapted from [4], presents the different approaches. For a repairable system, the failure rate is referred to as the intensity function, instead of the hazard function.

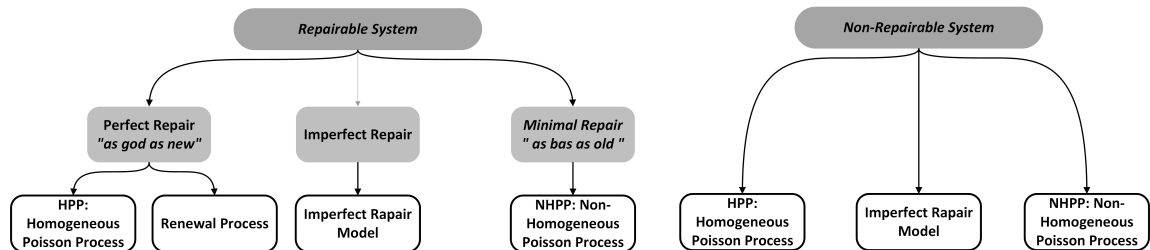


FIGURE B.6 – Types of system, stochastic point processes, and distributions [4].

Poisson's models are very common to the model counting process. The HPP model (Homogeneous Poisson Process) is a simple and popular form for probabilistic calculations of repairable systems. This model considers a constant time between failures, i.e., it follows an exponential distribution of λ and these events are independent and identically distributed [33]. Since the repair rate is constant, this model is most often used when the system is in its useful life period (see Figure B.5). Furthermore, it implies that a repair returns the system to its original state of "as good as new" [47]. In contrast to the HPP, the NHPP model (Non-Homogeneous Poisson Process) does not consider perfect maintenance. The failure rate is not necessarily constant, it can evolve with the age of the asset and is referred to as a power law process. In other words, previous repairs will not affect future system performance [47]. Furthermore, the HPP model is a special case of the NHPP, when the shape parameters are equal to 1. The representation functions of both models are presented in Table B.2. This model offers much flexibility because it can model increasing (system deterioration), decreasing (system regeneration), and constant failure rates. In addition, the intensity function of the power law has the same form as the hazard function of a Weibull distribution, presented in the following table. In the case of the NHPP, the system is considered "as old as new" after repair [33, 47].

Tableau B.2 – Representation functions for repairable systems.

Distribution	Cumulative $F(t)$ or CDF	Probability Density $f(t)$ or PDF	Survival $S(t)$ or $R(t)$	Hazard $h(t)$ or $\lambda(t)$	Cumulative Hazard $H(t)$
HPP	$1 - e^{(-\lambda t)}$	$\lambda e^{(-\lambda t)}$	$e^{(-\lambda t)}$	λ	λt
NHPP	$1 - e^{(-\lambda t)^k}$	$k\lambda^k t^{k-1} e^{(-\lambda t)^k}$	$e^{(-\lambda t)^k}$	$k\lambda^k t^{k-1}$	λt^k

Survival Analysis

Parametric analyses are essential to estimate the failure rate of a group of assets. These analyses are very common in the field of reliability. However, they only consider failure events, without considering the influence of other factors on the system. In this regard, survival models allow determining a relationship between these failure events and some variables [46,48]. As with parametric analysis, there are specific techniques for repairable and non-repairable systems. The most well-known models, for dealing with non-repairable cases, are the proportional hazard model (Cox model), the additive Aalen model, and the accelerated life model. Several researchers have focused on developing extensions capable of handling the case of recurrent events. The most common variations of Cox models, known as proportional intensity models, are the Andersen–Gill, Wei–Lin and Weissfield, and Prentice–Williams–Peterson models [49, 50]. As shown in Table B.3, the intensity function of the survival model includes a β parameter, which is a coefficient vector, that is multiplied by X_i , a covariate matrix. In this way, the baseline hazard function $\lambda_0(t)$ is adjusted depending on the covariates, which is not the case with the NHPP. Thus, survival analysis can be used in reliability to obtain a more precise estimate of the failure rate, with respect to the parameters of a specific asset.

Tableau B.3 – Intensity function comparison.

NHPP	$k\lambda t^{k-1}$
Anderson–Gill	$\lambda_0(t)e^{\beta X_i(t)}$

There are numerous applications of mathematical modeling for RAMS. As an example, the authors of [51] present a complete reliability analysis for different types of power transformers from an Australian electrical utility. They compare the results obtained by parametric analysis with Weibull to nonparametric analysis with Kaplan–Meier curves. In addition, a failure mode analysis is used to determine the prevalent causes of failure in relation to the age of the assets. In [52], the author proposes a methodology to quantify the uncertainty related to data quality in the application of statistical tools to estimate reliability. A case study on electrical utility assets in Canada is presented, and the HPP and NHPP methods are used to test the approach. In addition, a case study applied to the Dutch power system is presented by [53]. Reliability statistics are obtained with survival analysis, using Kaplan–Meier curves, and the parameters are estimated using a Weibull distribution. These results are then used in simulation scenarios to optimize maintenance and replacement policies.

Prognostic and Health Management

PHM Basics

As the name suggests, Prognostics and Health Management (PHM) is a discipline that focuses on system health management, prognostics, and enhanced diagnostic techniques [54],

while RAMS focuses on studying the overall properties of a population, PHM proposes much more targeted approaches, by tracking individual assets. An important factor is that it implies that the assets must be continuously monitored in order to ensure the functions of the system. Thus, this discipline relies on data generated by sensors, which is different from RAMS, where historical data are generally used for modeling. The sensing system allows for real-time diagnostics as well as fault detection and isolation. The prognostic techniques allow for moving from a reactive to a preventive decision mode, especially in the execution of maintenance tasks. The general objective of this discipline is to optimally use an asset throughout its useful life. In other words, the prognostic goals align with those of asset management and RAMS. Estimation of the residual life of the system is the basis of the prognostic domain. To account for different failure modes, prognostics involves making multiple estimates of residual life. These estimates are made using historical data, projected scenarios, qualitative models (FMECA), manufacturing data, etc. In addition, prognostics aims at predicting the health of a system over time, to adapt maintenance actions and decision-making [6]. As PHM is about performing maintenance based on the predicted state obtained from monitoring, it is called condition-based maintenance, specifically, predictive maintenance.

Figure B.7 shows that, before prognostics, decisions were taken based on data and diagnostics. With prognostics in the process, the decision-making is influenced by predictive modeling; with the estimation of the remaining useful life, the time to failure can be predicted, and the maintenance performed based on the actual system state. PHM methods are divided into physics-based approaches, data-driven approaches, and hybrid approaches [6]. Sometimes, some papers include reliability-based approaches in PHM modeling, but this section will describe the first three approaches, since reliability-based approaches have already been discussed.

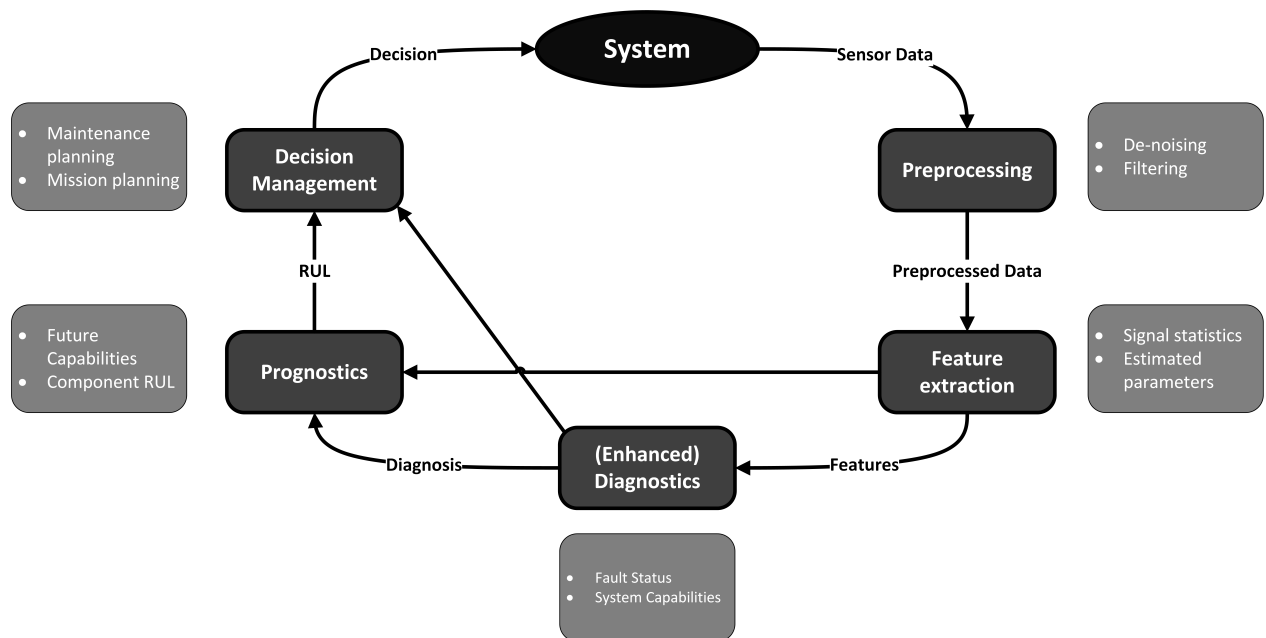


FIGURE B.7 – Prognostic in a health management process [5].

Qualitative Modeling in PHM

As with rams, PHM practitioners rely on qualitative models to establish a knowledge base on assets and their degradation. Indeed, it is common to use failure mode, effect, and criticality analysis (FMEA/FMECA), fault trees, and other methods to identify risks [5].

Mathematical Modeling in PHM

Physics-Based Model

In prognostics, physics-based modeling is one of the most accurate methodologies for the estimation of system parameters. To develop this type of model, it is necessary to have a thorough knowledge of the system, the interactions between the components and the environment, etc. In addition, it usually requires significant knowledge of mathematics and concepts of the physics of degradation. As shown in Figure B.8, physical models

can be developed using the finite element method, focusing on the physics of materials of a component, or even by simulation of the system and its stressors. They can also be established from the fundamental laws of physics using differential equations. In this approach, a physical model is developed for a system, a subsystem, or a component, describing the system based on mathematical equations. These equations represent the failure modes and degradation phenomena, based on the laws of physics that are applicable. Once the model is established, it allows for a diagnosis while monitoring the asset in operation. Given the state of the system, it is possible to predict future behavior. Physics-based models can be extremely accurate and very useful in prognostics. However, this implies that the model remains true to reality. As such, development can be difficult and require significant knowledge. Furthermore, the complexity of the models and the fidelity of the model can lead to computational difficulties [5, 6].

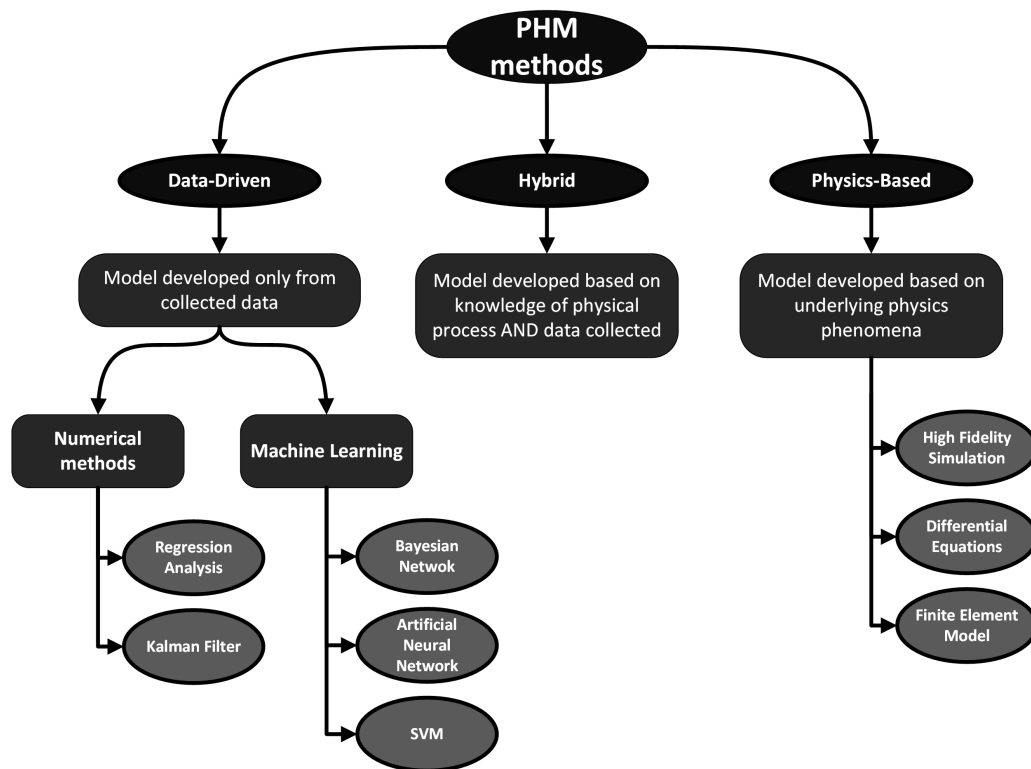


FIGURE B.8 – Prognostic methods (adapted from [5, 6]).

Data-Driven Model

Data-driven modeling is, in theory, only based on data collected from the system [6]. Physics-based approaches are generally more accurate than data-based approaches [5]. However, data-based approaches generally require less knowledge of the system and the degradation phenomena, which makes them very attractive [6]. This is even more true when it is too difficult to model the system with physics-based approaches. Indeed, these allow a faster development and are therefore more widespread in the prognostic domain. In a simplified way, these methods consist of using historical or sensor data to determine system trends. When the system degrades, it is because the observed variables deviate from these trends and exceed a certain pre-established threshold. These models can then be used to estimate or predict the system's remaining useful life [6]. Data-driven techniques include statistical methods and machine learning (ML). The drawbacks of these techniques are the same as for data science projects; the models require a large amount of data to describe the phenomenon, which can be difficult when acquiring run-to-failure data. Moreover, choosing the right methods and avoiding overfitting and/or underfitting are also important challenges. Data-driven and machine learning applications are abundant in the literature. In [55], the authors present the development of a predictive maintenance model taking into account covariates and possible interactions, applied to a solar thermal power plant. The model developed combines survival analysis with artificial neural networks, with the aim of integrating it into a system for monitoring the state of the heat transfer pump and predicting failures according to the dynamic state of the system. Ref. [56] proposes a study for the selection of relevant variables to facilitate the deployment of a predictive maintenance and real-time monitoring program. From the same perspective, Ref. [57] proposes a framework and a methodology to decrease the dimensionality of asset life cycle data and minimize information loss.

Hybrid Model

It is evident that, in practice, modeling still requires basic knowledge of the system and, in this case, of the failure modes and data that are available (sensors or history), even for data-driven techniques. Most applications in PHM are developed from both the data and the physics of the system [5]. For example, as for any data science project, prior knowledge of the system and its functioning is required to select appropriate methods (regression, classification, clustering, etc.), relevant variables, and performance metrics. This is where hybrid models come in. Hybrid techniques combine data-driven and physics of failure approaches for modeling. In this way, physics-based techniques are applied when possible, and if not, data are used to overcome the lack of knowledge. The worst-case scenario is where modeling by physics is impossible and there is not enough data. This is one of the main problems that can occur with this method. As it has advantages from the two types of techniques, it can also have disadvantages from both [5, 6].

Discussion

As demonstrated, the two research areas are complementary, although they differ in some respects. RAMS focuses on the health of a group of systems, whereas PHM studies a single system in a closer manner. A parallel can be drawn with the field of medicine, where public health focuses on studying health issues on a national, provincial, or regional scale, while physicians treat patients individually, based on their history, current problems, and data collected during medical examinations. In the perspective of the hierarchical model of decision-making, the RAMS strategies are mainly at the tactical level. Indeed, the models allow for medium-term decisions over the life of the assets, and the level of detail provided is moderate. As for the PHM, it is generally operational strategies; the relationship with

the asset is direct, and the level of detail of the information is very high and allows for short-term decisions, or even real-time decisions. Nevertheless, the two fields generally have influence at several levels of the company, up to the choice of system design. Four types of modeling approaches were presented: qualitative, physics-based, data-based, and hybrid. Qualitative approaches are used to establish basic knowledge about the system, its design, components' interactions, and associated failure risks. Qualitative approaches are used for both RAMS and PHM, and the most commonly referred to approaches in the literature are failure mode, effects and criticality analysis, and fault tree analysis. The techniques based on the physics of failures are more associated with the PHM domain. Data-driven approaches are applicable to both domains, and some techniques are similar. RAMS is an older field, and the mathematical methods generally derive from inferential statistics and regression analysis. PHM is a more recent research domain that tends to use sensor and connectivity technologies, and modeling is often performed with machine learning techniques. Reliability techniques, such as exponential models (Weibull, Gamma, etc.), are sometimes used in PHM, but these techniques require much failure data, which can be difficult to obtain when studying a single system. The time to accumulate sufficient failure data would be prohibitive. Instead, PHM relies on sensor data to develop predictive maintenance and anomaly detection models. When models are built from data, the procedure is similar to that used in data science, whether it is for RAMS, PHM, or any other application. In addition, the Engineering of Asset Management aims to coordinate efforts to optimize the life cycle of assets. Its application is of crucial importance for asset-intensive organizations, such as the energy, aeronautics, or aerospace sectors. The EAM framework provides long-, medium-, and short-term strategies in the management of activities. Thus, it integrates RAMS and PHM in the planning and optimization of the life cycle, the RAMS having a general aim on the long and medium terms, and the PHM on a medium- and short-term

horizon of decision-making. It is also important to acknowledge the impact of asset management, not only from an economical perspective, but also from a social and environmental point of view. Asset management objectives actively contribute to sustainable development. Indeed, the optimization of the asset life cycle aims to reduce waste through a long-term vision, from a more durable design, through optimal maintenance, to the disposal of the system. It has been shown by numerous applications, asset management and maintenance strategies are largely employed in the public and energy sector, where resources must be carefully allocated. In summary, this work helps researchers in the field to correctly identify the different theoretical approaches to reliability engineering, while giving some concrete examples of applications. Moreover, this text allows distinguishing the different modeling methods related to each discipline and to identify the similarities and limitations of these disciplines. However, this work is limited to asset management in the perspective of reliability engineering. As stated before, EAM is an integrated framework, therefore it goes beyond the scope of maintenance activity. Furthermore, the article covers the basic mathematical modeling methods of RAMS and PHM. It is intended to be a starting point for research rather than an in-depth review of all available methods.

Conclusions

The field of reliability engineering, like many other fields, is evolving rapidly thanks to the democratization of data acquisition and processing technologies. In this paper, we have tried to provide a unified perspective of the different research fields, namely RAMS, PHM, and Engineering of Asset Management. This review also addressed RAMS and PHM modeling techniques and highlighted the importance of these disciplines in ensuring the reliability, availability, maintainability, and security of complex systems. It also presented how they are defined in a management and decision-making model

by including the hierarchy of organizational decision-making in management. The first sections have defined the basic concepts of maintenance, failure, and aging, in addition to defining the different modeling concepts. The importance of understanding the basics of mathematical modeling and statistical models was also emphasized. Furthermore, the most common analysis methods were associated with their respective fields, from qualitative modeling to data-driven techniques. This review highlighted the different modeling approaches used in RAMS, including qualitative modeling techniques such as failure modes, effect analysis, and fault tree analysis, and statistical modeling techniques for repairable systems and non-repairable systems. In PHM, the review explored qualitative modeling and different mathematical modeling techniques, including physics-based models, data-driven models, and hybrid models. From this, the distinction between the different subjects has been clarified, in accordance with their analysis objective. In conclusion, this review of the literature provides a comprehensive overview of the different aspects of RAMS and PHM modeling techniques, emphasizing their importance and complementarities in an asset management program. This allows new researchers to become familiar with the field of reliability and bridge the gap between traditional analytical techniques and the latest developments. Indeed, artificial intelligence techniques, although of increasing interest, are still not widely applied in practical contexts with operational data. This reflects an opportunity to link reliability and machine learning for industrial applications.

Annexe C - Machine Learning Applications for Reliability

Engineering: A Review

M. Payette and G. Abdul-Nour, Sustainability, vol. 15, no. 7, p. 6270, 2023.

Ce chapitre correspond à l'article publié dans la revue Sustainability, publié le 27 mars 2023.

Abstract

The treatment of big data as well as the rapid improvement in the speed of data processing are facilitated by the parallelization of computations, cloud computing as well as the increasing number of artificial intelligence techniques. These developments lead to the multiplication of applications and modeling techniques. Reliability engineering includes several research areas such as reliability, availability, maintainability, and safety (RAMS); prognostics and health management (PHM); and asset management (AM), aiming at the realization of the life cycle value. The expansion of artificial intelligence (AI) modeling techniques combined with the various research topics increases the difficulty of practitioners in identifying the appropriate methodologies and techniques applicable. The objective of this publication is to provide an overview of the different machine learning (ML) techniques from the perspective of traditional modeling techniques. Furthermore, it presents a methodology for data science application and how machine learning can be applied in each step. Then, it will demonstrate how ML techniques can be complementary to traditional approaches, and cases from the literature will be presented.

Mots clés: artificial intelligence; engineering of asset management; machine learning; prognostic and health management; reliability

Introduction

For the past few years, machine learning and artificial intelligence have been attracting the research community's attention. More and more application cases are emerging in the manufacturing environment, especially with the advancement of the Industry 4.0 vision. The digitization of the environment through connectivity and cyber-physical systems is leading to the generation of big data, which has several processing challenges. This is now referred to as the "analytics disruption" due to the fact that, in general, organizations use less than 10% of the data generated for modeling and decision support [145]. This phenomenon is not escaping the reliability domain either. New ML analysis techniques have been the subject of many publications, although traditional methods in the field are still widely applied. The new techniques and technologies available allow the development of new applications, but also new domains, making the selection of appropriate methods more and more complex [9]. The goal of this work is to make it easier to understand the difference between traditional and ML modeling techniques, as well as how they can be applied in reliability applications. It seeks to provide a summary analysis of the different topics so that researchers interested in the application of machine learning in reliability engineering can become familiar with these different topics.

Section C defines the different types of modeling methods. First, a definition of mathematical modeling is presented, then the branches of statistical modeling and machine learning are defined. Furthermore, Section C presents a history of the development of artificial intelligence methods in order to show recent developments in the field. Although the discipline has been investigated for a long time, the success of its applications is quite recent, especially in the field of reliability. Section C formalizes the modeling process used in data science, a process common to both RAMS and PHM for building data-driven

models. Finally, a systematic review presents recent ML applications in reliability engineering.

The analysis of the literature will be structured as follows:

Research questions:

1. Is the application in RAMS or in PHM ?
2. What machine learning methods are used ?
3. What system is under study ?
4. Does the dataset come from real data, simulated data, or a public benchmark dataset ?

Databases used for the review: The research was done using the EBSCO library, precisely with the following databases:

1. Applied Science & Technology Source
2. Applied science and computing database
3. Computers & Applied Sciences Complete
4. Applied science and computing database

Keywords

title -abstr-key(("reliability engineering") AND ("machine learning" OR "deep learning"))

Exclusion criteria

1. Works not related to reliability engineering, i.e., not related to either RAMS or PHM
2. Works dated before 2017
3. Works that are not in English or French

Selection Criteria

1. Full text available
2. Peer-reviewed journal

Modeling

This section explains the various modeling techniques used in RAMS and PHM. First, reliability engineering is defined in relation to the RAMS and PHM topics. Following that, the fundamentals of mathematical and statistical modeling are introduced. Then, the respective methods for both fields are highlighted, from qualitative models to physics-based methods. Finally, methods for machine learning and artificial intelligence are discussed.

Reliability Engineering

Reliability engineering is an engineering field that focuses on ensuring the reliability and maintainability of systems. It uses various tools, techniques, and methods to identify, analyze, and mitigate potential failures that could affect the performance and safety of assets. This field has been in development since the 1950s and is used in various industries, such as the military, consumer, and energy. Prognostics and health management (PHM) and reliability, availability, maintainability, and safety (RAMS) are subfields of reliability engineering. PHM focuses on the management of system health, the prediction of future performance, and the implementation of advanced diagnostic techniques [54]. Unlike RAMS, which examines the general characteristics of a group, PHM takes a more specific approach by monitoring individual components [34].

Mathematical Modeling

In a broad sense, modeling is used to represent a simplified version of an object or situation, to understand it, and to analyze it. Mathematical modeling is the use of mathematical techniques to represent the true conditions of a specific scenario. Kaiser and Stender's modeling process describes a cycle of modeling and validation in order to obtain a model

that accurately depicts a real-world problem [2,35]. Two approaches to statistical modeling can be differentiated. Descriptive statistics seek to describe and summarize the observations of a sample using indicators, graphical representations, etc. [36]. Inferential statistics, in contrast, intends to infer the characteristics of a group based on a sample [37]. Figure C.1 summarizes the most common approaches from both descriptive and inferential statistics. Probability distributions are used in inferential statistics to describe and extract the characteristics of random variables in a sample. It is generally simple to identify the probability distribution and determine the parameters of this distribution from the sample, knowing the type of random experiment. This is commonly referred to as parametric analysis. In some cases, the distribution of data can be easily defined by factoring in the operational and random context of the phenomenon under study [38].

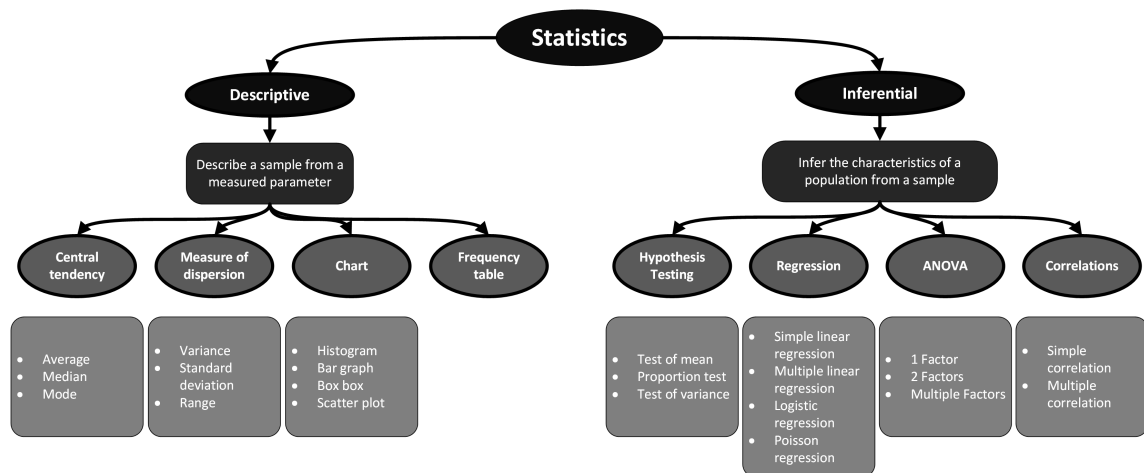


FIGURE C.1 – Common statistical modeling techniques.

Descriptive analysis, also known as non-parametric analysis, is used to determine the characteristics of a sample without using a statistical distribution. Measures of central tendency (mean, median, modes) and dispersion (range, variance, standard deviation, etc.) are commonly used to describe the characteristics of a population under investigation. Histograms, scatter plots, and box plots are commonly used to study the behavior of

systems in reliability. Without the need for a specific distribution law, frequency tables can also be used to estimate the probability density function [38].

RAMS and PHM Approaches

There are three main categories of models used in reliability engineering: data-driven, physics-based, and qualitative modeling. Data-driven modeling is a method of modeling that relies primarily on data to make predictions or inferences about a system or process. This approach involves using statistical techniques (Section C) and machine learning algorithms to analyze data and create a model that can be used to make predictions. Physics-based modeling, on the other hand, is a method of modeling that relies on the laws of physics and the fundamental properties of the system or process being modeled. This approach involves using mathematical equations and simulations to understand and predict the behavior of a system based on its physical properties. Finally, qualitative modeling is used to deal with non-numerical or non-quantitative information. It is used to understand complex systems, processes, or phenomena that are difficult to quantify or measure [42]. These models can take the form of diagrams, flowcharts, or other visual representations that can be used to understand the complex relationships between different components of the system.

As stated before, PHM seeks to provide a personalized follow-up of assets. This means that monitoring the components is a continuous process to guarantee the system's performance. Therefore, this field heavily relies on data obtained from sensors, unlike RAMS, which typically uses historical data for modeling. RAMS mainly uses qualitative and statistical (data-driven) modeling to determine the characteristics of assets. PHM relies on qualitative and data-driven modeling, as well as physics-based and hybrid (data and physics-based) modeling. The use of sensor data makes it easier to apply machine learning and deep

learning techniques in the fields of PHM. Qualitative modeling techniques include failure modes and effects analysis (FMEA) and fault trees and are generally used both in PHM and RAMS.

Machine Learning, Artificial Intelligence and Data Science

Machine learning, also known as artificial intelligence, is a form of mathematical modeling that allows a system to learn from data and not through the explicit programming of a system's constraints and environment [58]. This definition of machine learning, developed by IBM (International Business Machines Corporation), raises the different important components of ML. First, it is about a system, a machine that learns. Learning is a set of processes that seek to develop or modify behavior through experience or interaction with the environment. Another important point is that learning is done on data, and not from explicit programming, for example in operational research, where the constraints of a system must be specified by mathematical equations. This means that a mathematical model is generated by the experience gained from the data that are sent to the algorithm. Generally, ML approaches are divided into supervised, unsupervised, and reinforcement learning methods.

History of Artificial Intelligence

In this digital age, it is undeniable that artificial intelligence is a scientific field in effervescence. Surprisingly, the concepts of artificial intelligence have been developed for almost 80 years. This raises the question as to why this recrudescence is happening now. This section, complemented by Figure C.2, describes a brief history of AI development and attempts to highlight the reasons for this interest. As early as 1943, a study presented the first concept of artificial neurons capable of performing logical operations. In 1950,

the English mathematician Alan Turing proposed a test, the imitation game or Turing test, to test machine intelligence [59]. In 1955, the term artificial intelligence was introduced by John McCarthy, Marvin Minsky, Nathaniel Rochester, and Claude Shannon for the Dartmouth Summer Research Project on Artificial Intelligence (1956). This event is sometimes considered the birth of artificial intelligence as a field of study. In 1958, Frank Rosenblatt made the first implementation of the perceptron algorithm, based on the work of McCulloch and Pitt on artificial neurons. In 1959, the concept of machine learning was presented by Arthur Samuel. In 1965, Edward Feigenbaum and his team at Stanford University developed DENDRAL, the first expert system capable of automating decision-making and problem-solving [60]. In the same year, the American scientist Gordon Moore predicted that according to his observations, the number of components in electronic circuits would double every year. This prediction, known today as Moore's Law, was adjusted in 1975, proposing instead that the computational capacity would double every two years, and this prediction has been realized until today. In their 1969 article, Marvin Minsky and Seymour Papert describe some limitations of neural networks, including the lack of computational power of computers at the time [60]. This article slowed down the research on deep learning, and the success of expert systems pushed the research toward this field. During the 1970s and 1980s, we saw the proliferation of expert systems: MYCIN (1972), XCON (expert Configurer) (1978), Deep Blue (1997), etc. However, the way in which expert systems are built limits their capacity: it is a collection of rules represented by a sequence of if-then statements allowing problem-solving [59]. On the other hand, machine learning and deep learning algorithms learn a model with the data, the interaction with its environment, etc. In 1989, the French researcher Yan LeCun applied the backpropagation algorithm to a neural network to recognize handwritten postal codes. In 1998, with Léon Bottou, Yoshua Bengio, and Patrick Haffner, LeCun proposed a convolutional neural network (CNN) for

handwritten character recognition [60]. These successes gradually revived interest in deep learning and machine learning, with computational capacity becoming less and less of an issue. In 2009, a Stanford research team proposed using graphics processors rather than CPUs to do the learning. Their project also details an architecture to parallelize the computations.

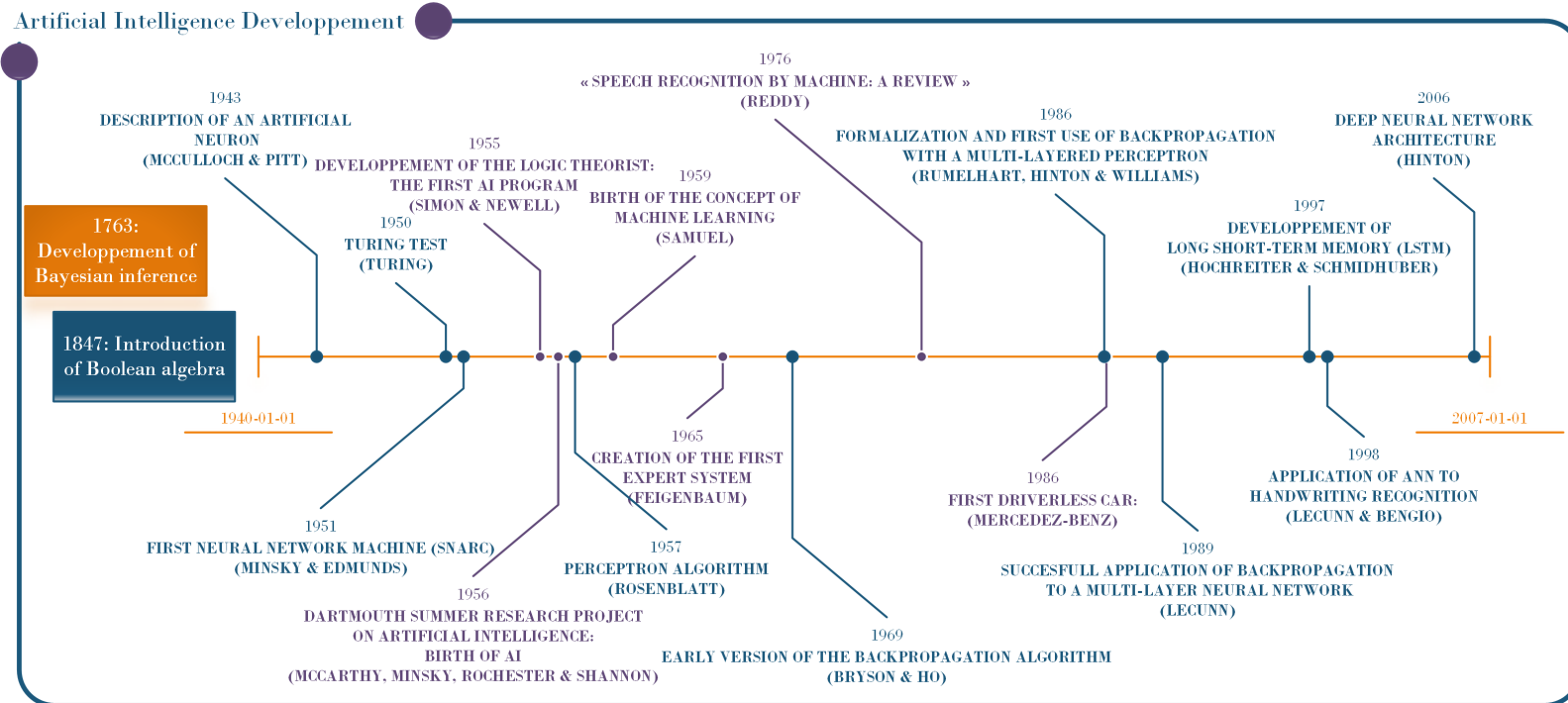


FIGURE C.2 – Artificial intelligence development.

At the turn of the century, there was a rapid increase in connectivity with the development of smartphones and social networks. The democratization of these technologies leads to an explosion of generated data. The exponential increase in the volume of data can also be attributed to the growing presence of sensor technologies and the emergence of the Internet of Things (IoT) [61]. Computing capabilities are becoming increasingly sophisticated, and the costs associated with technologies are becoming inexpensive. In addition, large and varied amounts of data (big data) are easily available to organizations. In other words, what was missing in the past to apply artificial intelligence is now widely available.

Supervised Learning

Supervised learning (SL) is the process in which the machine observes examples of data in the form of input and output pairs X_i, y_i [62]. The first phase of learning is called training and the X_i, y_i pairs are called labeled data. Figure C.3 is a visual example of a learning process where one would try to classify pictures of cats and dogs.

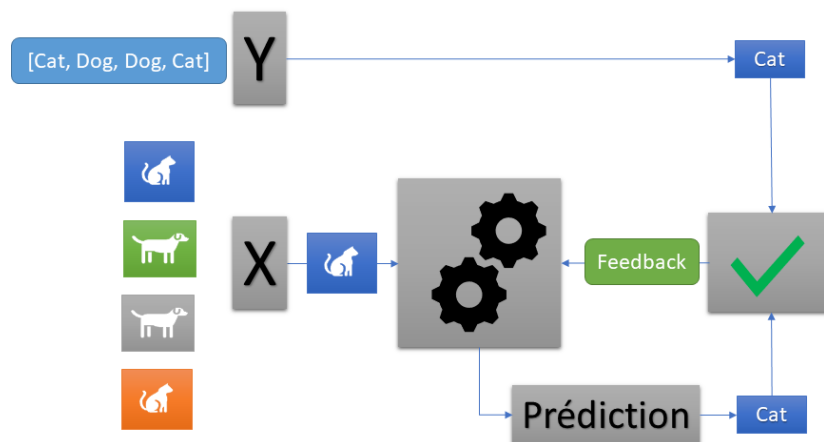


FIGURE C.3 – Example of supervised learning.

The algorithm receives the X variables (pictures) and makes a prediction (cat or dog). Knowing the value to predict, the algorithm can modify its behavior (its parameters) with each example it receives. The assumption is that over the course of iterations, the prediction error will decrease sufficiently so that the resulting model is able to predict the variable y , with new examples X that it has never observed. This is referred to as generalization. To evaluate the predictive ability of the model, the data are divided into two parts: the training sample and the validation sample. The model is trained on a sample of the data, and then, the validation set is sent to evaluate the average prediction errors of the model and to improve its performance by optimizing the hyperparameters. Once the hyperparameters have been optimized, the validation gives a first indicator of the model's performance, indicating whether the model fits the data well. To know if a model has a good ability to generalize, the data are divided into two parts: the training sample and the test sample, as shown in Figure C.4. The evaluation of the final model with the test sample gives a second performance indicator, on the ability to predict new data. In summary, the data are divided into three parts: the training sample, the validation sample, and the test sample. This procedure is called cross-validation.

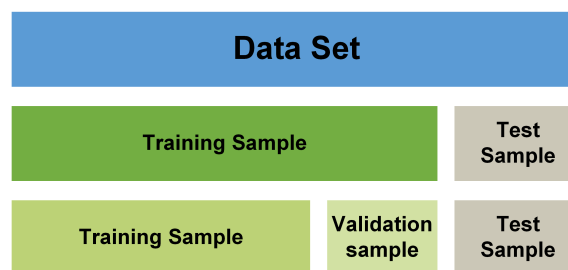


FIGURE C.4 – Example of data sampling for SL.

Figure C.5 shows two different curve fittings, where the coefficient of determination (R^2) allows us to evaluate how well it fits the data, that is, the goodness of fit. The curve on the left has a coefficient of 0.71 and the one on the right of 0.85.

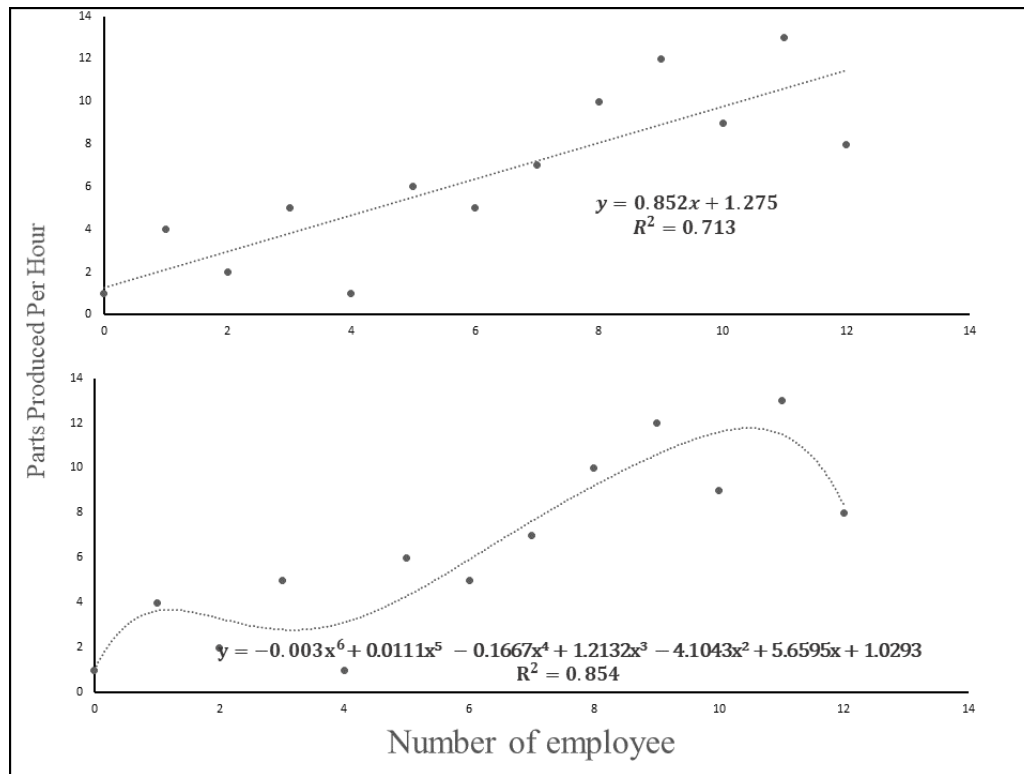


FIGURE C.5 – Example of curve fitting.

Although the curve on the right shows a better fit to the data, it is unlikely that this model will achieve good performance for prediction on new data. This phenomenon is often referred to as over-fitting. In the same way, a model that does not perform well on the training data might not have good predictive ability. This is what is called the bias/variance trade-off. By doing two evaluations of the performance of the model, it ensures that the model is well-balanced. Generally, supervised learning methods are divided into two families according to the variable to predict. First, there is classification, where the variable to predict is a discrete variable. Often, it is about predicting a class, a label, etc. Then, there is regression, where the variable to predict is a continuous variable. Figure C.6 shows some well-known methods in supervised learning in regression and classification.

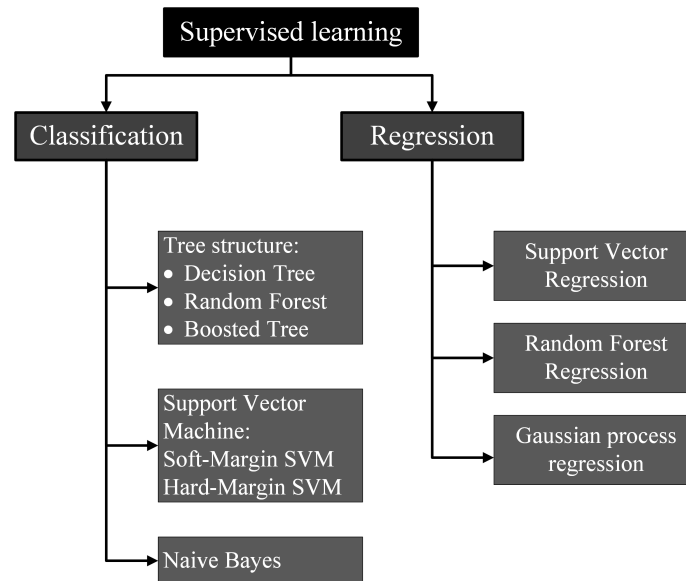


FIGURE C.6 – Supervised learning approaches and algorithm.

Unsupervised Learning

Unsupervised learning, as opposed to supervised learning, is the ML process where learning is done using unlabeled data [58]. The idea is to determine the relationships between variables without having a variable to predict. A classic use of this type of learning is data clustering. The objective of clustering is to categorize the data into subgroups that are determined by the similarity between the data [63]. Unsupervised learning is often used in big data where modeling can be very time-consuming, especially if all variables are included. Clustering, for example, can be used to reduce the number of variables in a dataset by grouping certain variables together based on common characteristics. A good example of clustering is the classification of animal species. By classifying animals by species (mammals, fish, birds, etc.), a large part of their characteristics is encapsulated in a single variable. The same kind of treatment is applied in supervised learning. Another class of unsupervised learning approach is dimensionality reduction. As the name implies, these methods take a dataset and reduce the dimensionality, i.e., the number of variables.

For example, principal component analysis (PCA) consists of taking the data and trying to construct new variables by making changes to reference points (axes). The data are projected into a new simplified representation system (fewer variables) minimizing the loss of information. In a simplified way, the method consists in finding a line that minimizes the sum of the distances in a scatter plot, as in regression. Then, the points are projected (orthogonal projection) on this line, which becomes a new reference system, named the principal component. Figure C.7 is adapted from the taxonomy developed in [63] to include the dimensionality reduction algorithm. It shows a classification of different unsupervised learning methods that are commonly used by practitioners.

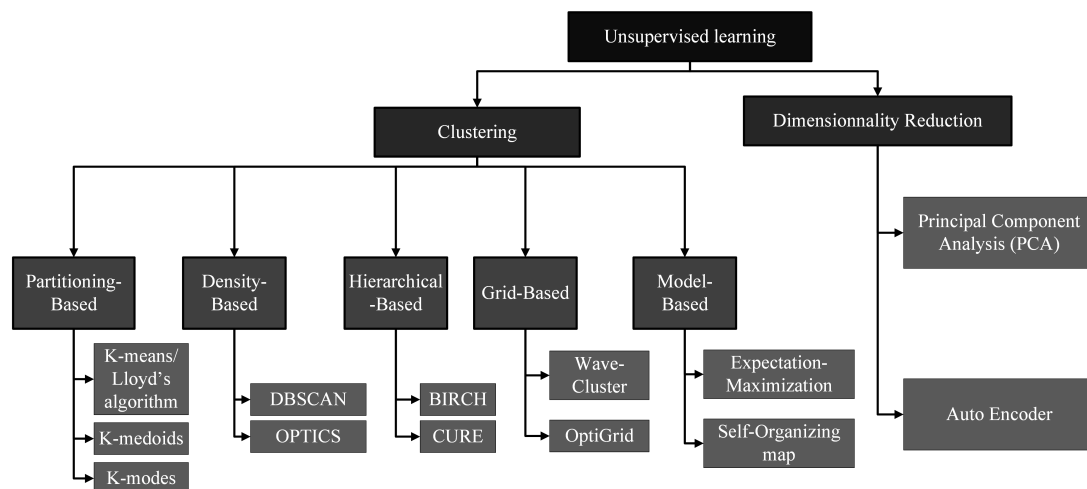


FIGURE C.7 – Unsupervised learning algorithm.

Reinforcement Learning

Reinforcement learning is another type of machine learning algorithm. This type of learning is quite similar to the process studied in behavioral psychology, where one tries to induce behaviors through positive or negative reinforcement of the subject. These methods are frequently applied in robotics and in the field of video games, for instance [58]. In many cases, data are generated by the interaction of an intelligent agent (a machine) and

its environment [64]. These data come from sensors, which bridge the gap between the physical and the computational, in the case of robots. From an algorithmic point of view, learning is done through these interactions with the environment and a penalty/reward function that guides the agent's decisions. The goal of the intelligent agent is to maximize the rewards of its actions. As the agent makes decisions and receives feedback, it will become increasingly competent at performing the tasks it is trained to do [64]. Figure C.8 presents a summary of the taxonomy of reinforcement learning algorithms developed by Zhang and Yu [65].

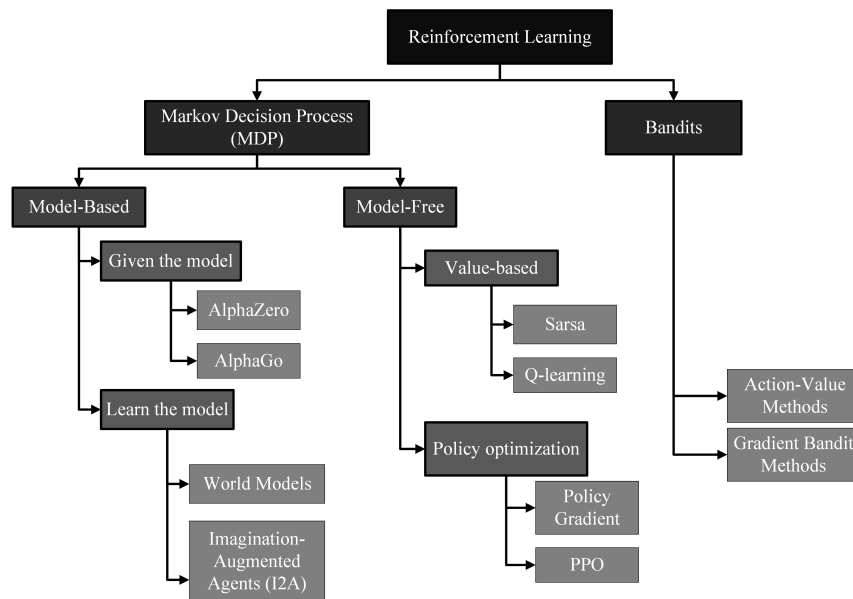


FIGURE C.8 – Taxonomy of reinforcement learning algorithm.

Deep Learning

Deep learning (DL) is a form of artificial intelligence. These methods take their name from the architecture of neural network algorithms. The neuron is the basic unit that makes up an artificial neural network (ANN), as demonstrated in Figure C.9 [62]. The simple neuron, more formally the perceptron algorithm, takes a vector as input, as shown in the supervised learning example. These values are multiplied by their respective weights and

aggregated by the input functions, then an activation function is applied to produce an output result.

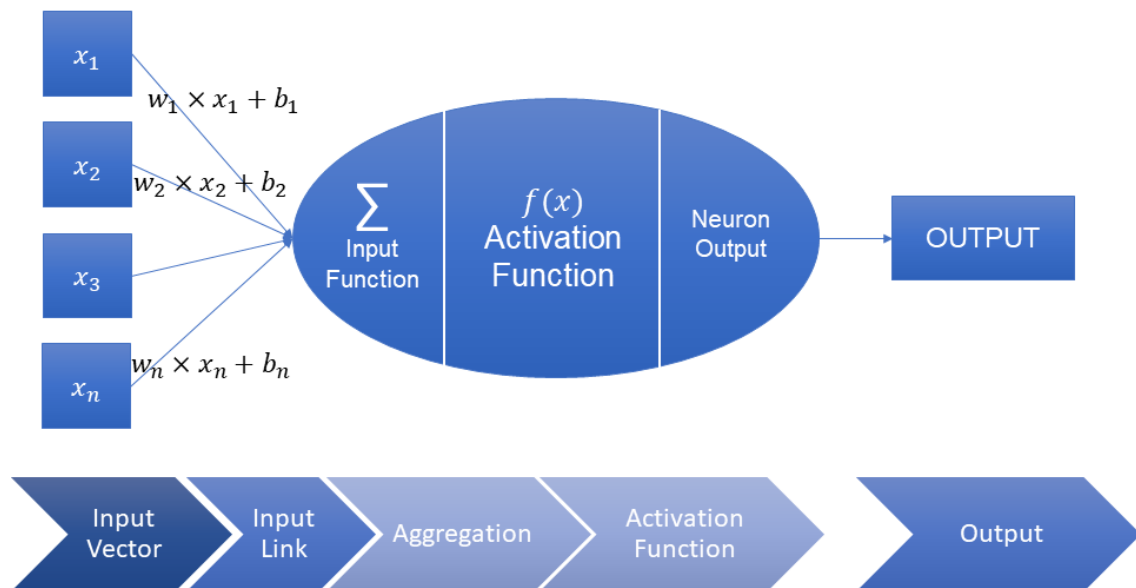


FIGURE C.9 – A simple mathematical model for a neuron.

A neural network, as its name suggests, is composed of successive layers of neurons, arranged in a network architecture. Deep learning occurs when a network has three or more layers [58]. Table C.1 shows the three most common ANN architectures, along with their applications, as presented in [66].

Data Science

Data science is an emerging discipline. Its objective is to use the data to gain insight and turn those data into value for an organization [67]. General applications for data science include reporting, diagnosis, prediction, and recommendation. The field combines multiple other disciplines such as machine learning, data mining, statistics, data visualization, and predictive analytics [67]. Figure C.10 presents the data science life cycle, which describes the general modeling process used by the practitioner. The first step of a data

Tableau C.1 – ANN architectures and applications.

Type	Structure	Applications
Artificial Neural Network	Input layer Hidden layers Output layer	Classification Regression Pattern recognition
Convolutional Neural Network	Input layer Convolution layers Pooling layer Fully connected layer Output layer	Natural language processing Image processing
Recurrent Neural Network	Input layer Hidden layers Output layer	Time series analysis Sentiment Analysis Natural language processing

science project, like in applied research, is to define the problem and its objectives according to the business perspective and context. The data are then collected, cleaned, and prepared for modeling. Modeling is often performed using machine learning methods. The method and the performance metrics are selected according to the objectives defined and the type of problem at hand.

Machine Learning

Data Science Modeling Process

Data Acquisition

The first part of data-driven modeling is the data acquisition process. Data are generated automatically by sensing technology or by a technician through an ERP system, for example. The data are then loaded into a storage environment. The data can be structured and stored in a relational database or unstructured and stored in a data lake. Once the data are

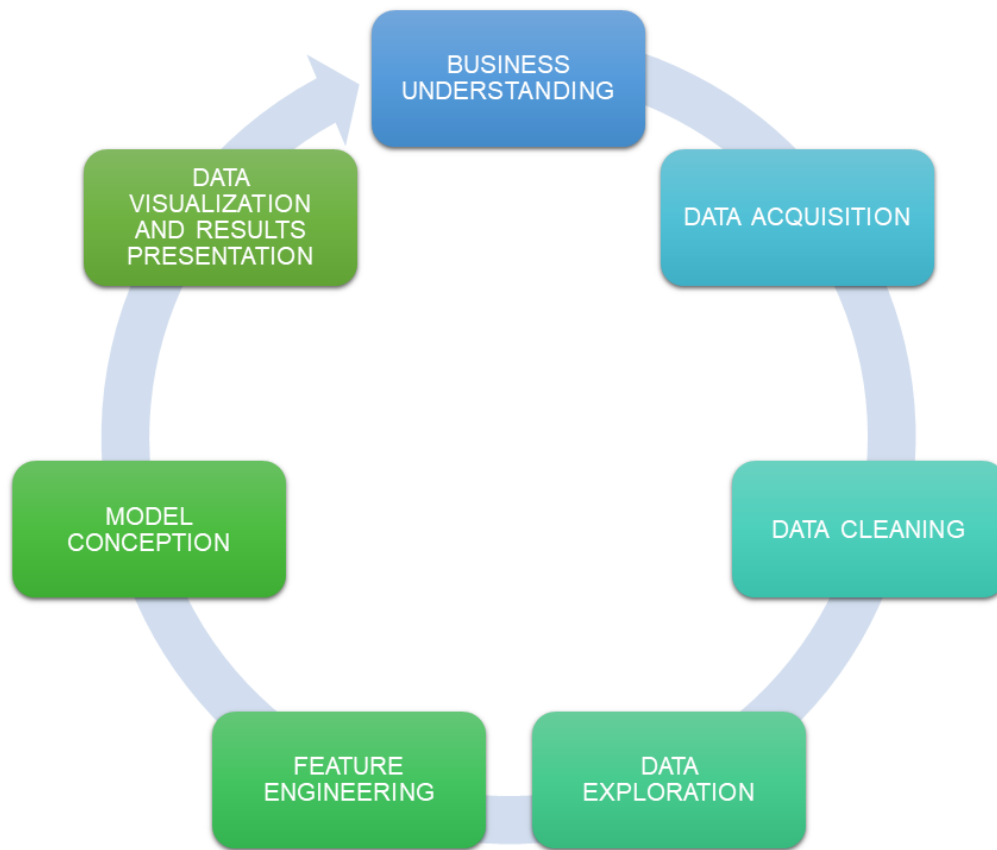


FIGURE C.10 – Data science life cycle adapted from [7].

available in a stored environment, it can be prepared for modeling. Sometimes, relevant data may be stored across multiple sources and needs to be gathered for analytics.

Data Cleaning

Data quality is an important issue for data-driven modeling. For example, a case study applied to the reliability of mining equipment shows that raw data are often erroneous, lacking detail and accuracy, and, therefore, not suitable for decision-making. The study reveals that different fields of relational databases show errors in assigning maintenance tasks to the right subsystem, assigning codes to describe the type of work, associating the right type of maintenance (condition-based, preventive, corrective), etc. [146]. Cleaning

must be performed so that the data are exploited to produce valuable insights for decision makers. In an ideal context, it is preferable that entry errors be prevented rather than corrected downstream. In fact, since the variance in data quality is greatly influenced by the user who enters it, it is essential to develop better management of the workers who interact with the database. Organizations must consider the effect of time pressure on data entry and provide feedback from supervisory staff to operators. In addition, it is necessary to encourage the participation of operators and to value their work toward data entry, in order to improve its quality [68]. Nonetheless, data scientists must ensure quality throughout the modeling and cleaning still needs to be performed. The data cleaning process involves the detection of errors and the removal or replacement in the dataset [8, 69]. Figure C.11 summarizes a methodology to quantify data quality (diagnostic) and manage its quality (correction) through a continuous process [8].

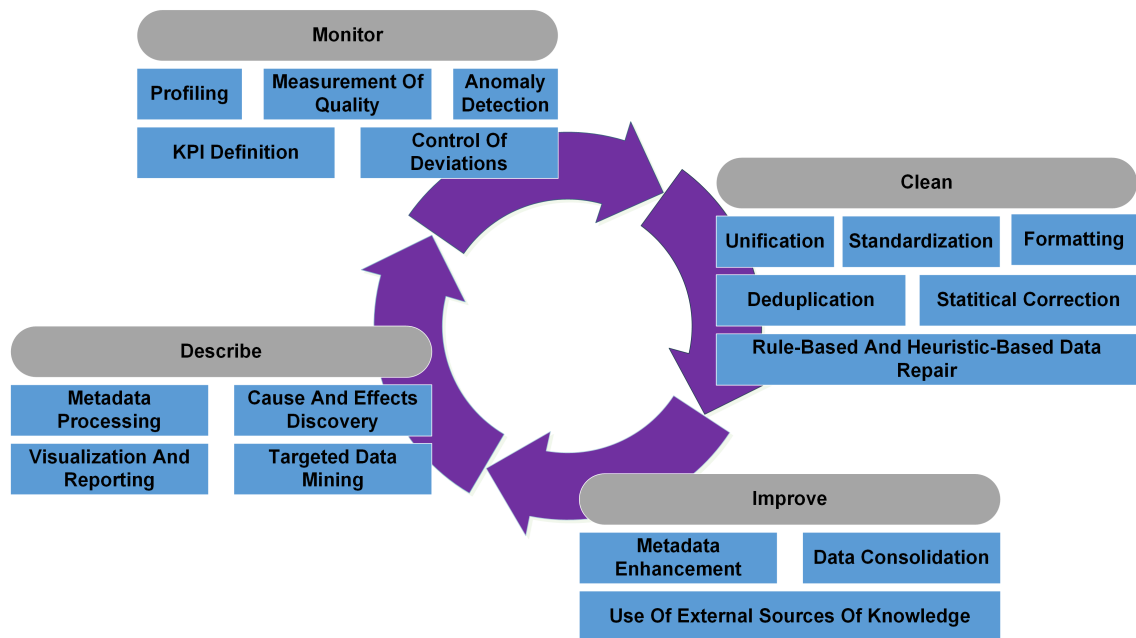


FIGURE C.11 – Virtuous circle for data quality management adapted from [8].

Cleaning of the dataset includes handling missing values, outliers, and bad data. The user can decide to correct the data with imputation techniques, get rid of the data, or

leave it as it is [70]. To get rid of inadequate data, the user can either remove the entries (row) or delete a whole field (column), depending on the completeness (% of missing data) of a feature. For the imputation of missing values, the simplest methods are based on descriptive statistics. Good choices are central tendency measurement: for continuous variables, there is mean, mode, and median; for categorical variables, there is the most frequent value. Interpolation is often used for imputation in the case of a time series. Finally, some users may use machine learning techniques to replace missing values. The k-nearest neighbors' algorithm (k-NN) and other regression methods are frequently used for this purpose [69].

Data Exploration

Once the data are gathered, data exploration or exploratory data analysis is done using graphical or statistical methods. Exploratory data analysis (EDA) is a practice that has been widely promoted by statistician John Wilder Tukey [71]. The idea behind EDA is to perform an initial examination of the data without any assumptions. Exploration is used to discover patterns or anomalies and then form a hypothesis on the data [71]. Common methods used in EDA are box plots, histograms, scatter plots, heatmaps, etc. The box plot, developed by Tukey, is used to represent graphically the minimum and maximum value, the median, and the quartile of a dataset. This graphic is very useful to detect outliers, in addition to showing the dispersion and skewness of a distribution. To graphically observe the shape of a distribution, the preferred representation is the histogram, which makes it an excellent tool for EDA. The scatter plot is used to plot the point of two variables as coordinates. Heatmaps, in the context of data analysis, are used to plot covariates against each other's and demonstrate the relationships, often using a correlation matrix. Data exploration generally starts at the beginning of the modeling project but is used throughout

the entire process. For example, a box plot is a good method for detecting outliers and can help with data cleaning. Additionally, heatmaps are a good starting point for selecting or eliminating features in the step of feature engineering.

Feature Engineering

When dealing with real-world data, there can be hundreds of features, and it is necessary to select the most relevant one from a dataset. Feature engineering is a process that includes feature selection, feature transformation, feature creation/feature construction, and feature extraction. The goal of this process is to reduce the size of the dataset by selecting and transforming features to optimize the learning of a model. Feature selection is the process of selecting the most relevant variables to perform modeling. Some variables may be irrelevant to the phenomenon studied, but some variables that are relevant may have unwanted effects on the model. For example, a feature may be redundant since it is highly correlated with another explanatory variable. When dealing with lots of variables, it is important to select the feature that will better explain the phenomenon without being too computationally intensive. Correlation coefficients (Pearson) and heatmaps are good methods for feature selection, as well as analysis of variance (ANOVA) tests and hypothesis testing. Sometimes, machine learning algorithms such as tree-based models (random forest (RF), decision tree (DT), etc.) are used in feature selection [72]. Feature transformation includes feature normalization and linearization. Feature normalization consists of scaling the values of a feature so that all features have the same contribution to the model. Many ML techniques use Euclidean distances to compute the distance between points. If the numerical features are not proportional, the estimation might be biased towards the largest variables [69, 70]. Linearization is a technique to transform the points of a distribution so that they can be represented by a linear function. This method is widely used in reliability

for the exponential distribution. With a logarithmic transformation, a curve fitting of the data gives the equation of exponential or Weibull distribution, for example [38]. Feature extraction is associated with dimensionality reduction techniques. The concept is to reduce the number of features by combining features with a linear projection in a lower dimensionality space. Feature creation or feature construction consists of using existing variables to create new features that are more appropriate for modeling. Examples of feature creation include encoding techniques (one-hot encoding, label encoding) and binning. Features can also be created with clustering methods, where the new variables represent groups of points with similarity [63].

Model Conception

According to what has been described in the previous sections, machine learning methods are used well before the design of the asset model. However, it is during the design of this model that the methods diverge between RAMS and PHM. Although some techniques are similar, the purpose of the models and their context is different, as presented in Sections C and C. Figure C.10 refers to predictive modeling, but it can be interpreted in a more general context as modeling of any kind (classification, prediction, clustering, curve fitting, etc.). In the case of a machine learning model, the method consists of training the model and validating its performance, for example, by cross-validation, and then using it to generate new knowledge from data. The same process applies to statistical modeling: the model is fitted to the data, and then the goodness of fit is evaluated with different performance metrics.

ML Applications Analysis

This section of the paper aims to analyze the literature on reliability engineering, particularly machine learning methods that have been used by practitioners. An analysis of applications in RAMS and PHM will give a clear picture of why and how ML modeling is used and identify gaps between theoretical applications and industry use cases.

Results

Execution and Filtering of Results

The keyword search on the two databases available on EBSCO gave hundreds of results. To reduce the number of publications, the search will focus on articles from the last 5 years that deal with either RAMS or PHM. Additionally, the complete article must be available for download, as it is important to review the work. The EBSCO filter tool allows making these selections quickly, especially with the publication date filter, the source type (academic journals), and the help of the thesaurus, to select publication by subject. Finally, of these results, only a few articles correspond to the subject in question, and the last filtering is done manually. Table C.2 presents the different selection and exclusion rules, as well as the number of publications that match this criterion.

Applications Analysis

As mentioned earlier, RAMS is a framework for evaluating and optimizing the performance of a system, focusing on the general characteristics of a population, while PHM is a more proactive approach that involves continuously monitoring a system to predict and prevent potential failures. We classified each article and its application according to these definitions. Figure C.12 shows the distribution of publications each year, by the subject

Tableau C.2 – Results filtering.

Rule	Numbers of Publications
Keywords	547
From 2017 to 2023	362
Works not related to reliability engineering	253
English or French only	251
Full-text available	38
Peer-reviewed	34
Hand-selected	19

of application. We notice that the number of publications per subject is quite similar for RAMS and PHM, with small variations over the years. We also notice that there is a strong increase in the overall number of publications in 2020, which corresponds to the outbreak of the pandemic, and then a significant decrease the following year.

Machine Learning Methods Review

Figure C.13 shows the different machine learning methods that were used in the articles. The artificial neural network is the most used method of all, being used in more than 30% of the studies. Furthermore, considering the different architectures of neural networks (convolutional networks, auto-encoders, and recurrent neural networks (RNNs) of type LSTM (long-short-term memory)), more than half of the methods are deep learning. Figure C.13 shows the total number of uses of the algorithm used in RAMS versus in total. Although deep learning methods are the most popular, the figure demonstrates that they are generally more used in PHM research than in RAMS. When examining publication objectives in prognostics, it seems that the research is more focused either on remaining useful life (RUL) estimation or online monitoring and diagnostics. Both applications require a large amount of data to build a supervised prediction model, so it is not surprising that deep

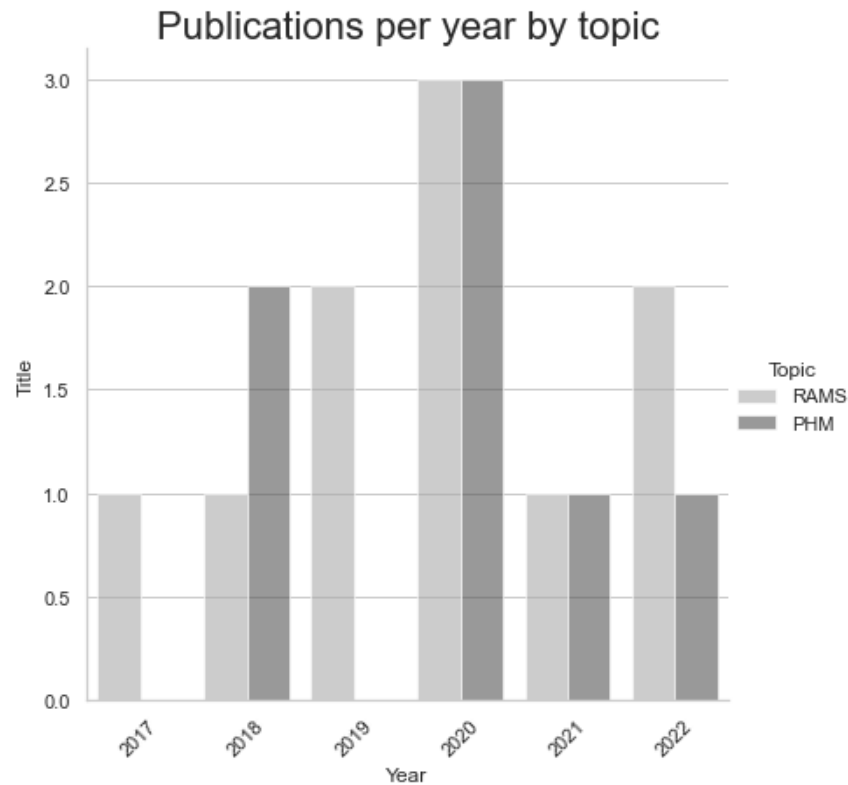


FIGURE C.12 – Publications per year by topic.

learning is the preferred ML approach. Furthermore, recurrent neural network methods such as LSTM can account for time dependencies because of their architecture, which contains feedback connections between layers, making it an excellent solution for RUL estimation. In RAMS, ML methods and modeling objectives are more diverse. For example, [73] tested deep learning approaches to solve the problem of stochastic flow manufacturing networks to predict the overall reliability of a manufacturing production line. Another study uses a support vector machine (SVM)-based algorithm to solve an optimization problem of structure reliability. Other RAMS applications apply machine learning to simulate possible scenarios and evaluate system reliability [74,75]. The vast majority of techniques, for both RAMS and PHM, are supervised learning methods; some studies have also used

transfer learning and self-supervised learning. Table C.3 shows in which articles each of the techniques were used.

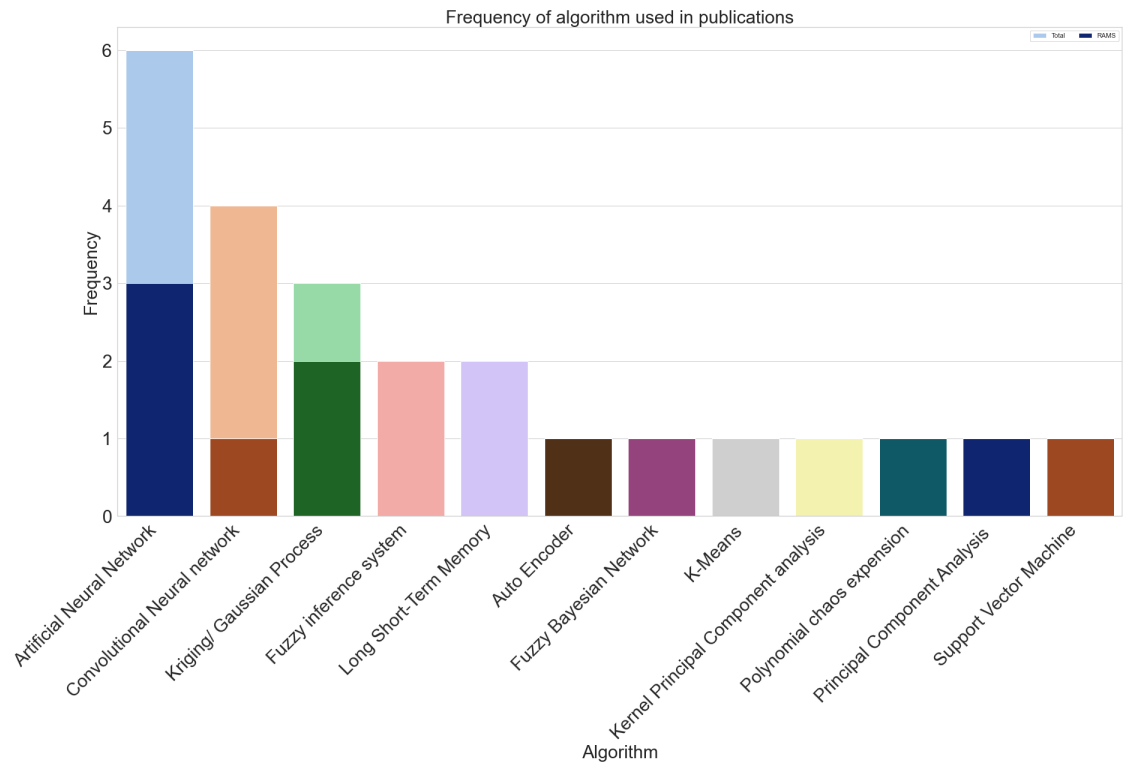


FIGURE C.13 – Frequency of use of ML algorithms by topic.

Tableau C.3 – Algorithms associated with reviewed articles.

Title	Authors	Data Type	Algorithm
Effective software fault localization using predicted execution results	Gao, Wong et al., 2017	Operational	K-Means
An Intelligent Reliability Assessment technique for Bipolar Junction Transistor using Artificial Intelligence Techniques	Bhargava and Handa 2018	Experimental testing	Artificial Neural Network, Fuzzy Inference System
Active fault tolerant control based on a neuro fuzzy inference system applied to a two shafts gas turbine	Hadroug, Hafaifa et al., 2018	Operational	Artificial Neural Network, Fuzzy inference system
Deep Learning for Accelerated Seismic Reliability Analysis of Transportation Networks	Nabian and Meidani 2018	Simulated/ Generated randomly	Artificial Neural Network
Gaussian Process-Based Response Surface Method for Slope Reliability Analysis	Hu, Su et al., 2019	Simulated/ Generated randomly	Kriging/Gaussian Process
Fault diagnosis of multi-state gas monitoring network based on fuzzy Bayesian net	Xue, Li et al., 2019	Operational	Fuzzy Bayesian Network

Tableau C.3 – (suite)

Title	Authors	Data Types	Algorithm
Active learning polynomial chaos expansion for reliability analysis by maximizing expected indicator function prediction error	Cheng and Lu 2020	Simulated/ Generated randomly	Polynomial Chaos Expansion, Kriging/Gaussian Process
A Reliability Management System for Network Systems using Deep Learning and Model Driven Approaches	Min, Jiasheng et al., 2020	Simulated/ Generated randomly	Artificial Neural Network
Integration of Dimension Reduction and Uncertainty Quantification in Designing Stretchable Strain Gauge Sensor	Sungkun, Gorguluarslan et al., 2020	Simulated/ Generated randomly	Auto Encoder, Artificial Neural Network, Principal Component Analysis
Data-driven prognostic method based on self-supervised learning approaches for fault detection	Wang, Qiao et al., 2020	Public dataset	Kernel Principal Component Analysis

Tableau C.3 – (suite)

Title	Authors	Data Types	Algorithm
Transfer Learning Strategies for Deep Learning-based PHM Algorithms	Yang, Zhang et al., 2020	Public dataset	Convolutional Neural Network
Bounds approximation of limit-state surface based on active learning Kriging model with truncated candidate region for random-interval hybrid reliability analysis	Yang, Wang et al., 2020	Simulated/ Generated randomly	Kriging/Gaussian Process
A data-driven approach based on deep neural networks for lithium-ion battery prognostics	Kara 2021	Public dataset	Convolutional Neural Network, Long Short-Term Memory
Using Deep Neural Networks to Evaluate the System Reliability of Manufacturing Networks	Yi-Fan, Yi-Kuei et al., 2021	Simulated/ Generated randomly	Artificial Neural Network
Convolutional And Long Short-Term Memory Neural Networks Based Models For Remaining Useful Life Prediction	Gritsyuk and Gritsyuk 2022	Public dataset	Convolutional Neural Network, Long Short-Term Memory

Tableau C.3 – (suite)

Title	Authors	Data Types	Algorithm
DL-RSIM: A Reliability and Deployment Strategy Simulation Framework for ReRAM-based CNN Accelerators	Wei-Ting, Hsiang-Yun et al., 2022	Public dataset	Convolutional Neural Network
A Novel Support-Vector-Machine-Based Grasshopper Optimization Algorithm for Structural Reliability Analysis	Yang, Sun et al., 2022	Simulated/ Generated randomly	Support Vector Machine

Datasets and Systems under Study

In this section, we present the different types of datasets that are used in the literature, as well as the different systems that these data come from.

As shown in Figure C.14, 47% of the studies use simulated data or data that are generated randomly by theoretical mathematical functions. Then, 29% of the publications used public datasets; these data are freely available to the public for different uses. A large proportion of public datasets are actual operational data. However, their treatment is simplified compared to a real case study, and their use is mainly intended to test and compare new approaches with existing methods. Many organizations make their data available to the public on a platform such as Google Dataset Search or Kaggle. For example, in [76], they used data from NASA's Turbofan engine degradation simulation [77] to compare the new RNN architecture for the estimation of the remaining useful life. In [78], they used NASA's experimental data on lithium-ion batteries to test a new CNN-LSTM architecture to improve the precision of the prediction of remaining useful life [79]. Less than a quarter of the study are interested in the analysis of industrial case studies (6%) through experimental testing and operational data (18%).

The types of systems studied are very diverse, as shown in Figure C.15. The reliability of computer networks and software is a subject that has been studied extensively, and this is reflected in the graph. As shown in Figure C.14, theoretical studies using various mathematical functions are very popular, instead of using real-life systems. Another topic that is relatively popular is the reliability of the monitoring and sensing systems themselves. Indeed, it is important to consider the possibility that monitoring systems produce misleading signals and themselves suffer from failures to improve decision-making in an organization.

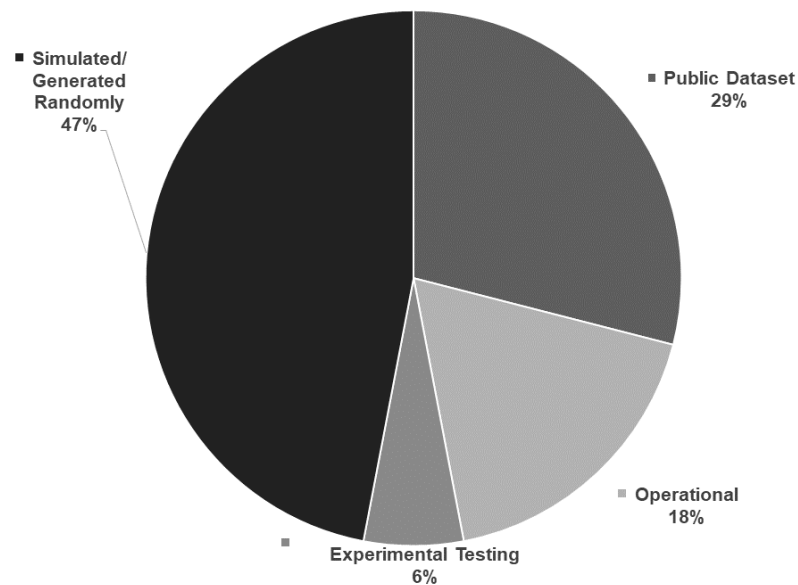


FIGURE C.14 – Pie chart of types of data sources.

Related Works

The systematic review presented in the previous section provides a comprehensive and reliable overview of the evidence on the topic of machine learning applied to reliability. However, the review is based on two databases, potentially excluding interesting articles on the subject. Therefore, this section aims to supplement the review by presenting some additional applications as well as programming tools for practitioners. To begin with, in [80], the authors present an agent-based modeling method for simulation to study the balancing of smart grids. The objective is to use this model to test the effect of balancing on electrical and telecommunications networks, among others. In [55], an attempt is made to develop a model to predict failures, taking into account several covariates while considering possible interactions. An approach to combining a neural network, specifically a single-layer perceptron, with the method of general renewal Weibull process for curve fitting is presented. Finally, the approach is tested through a case study on solar power plants and presented,

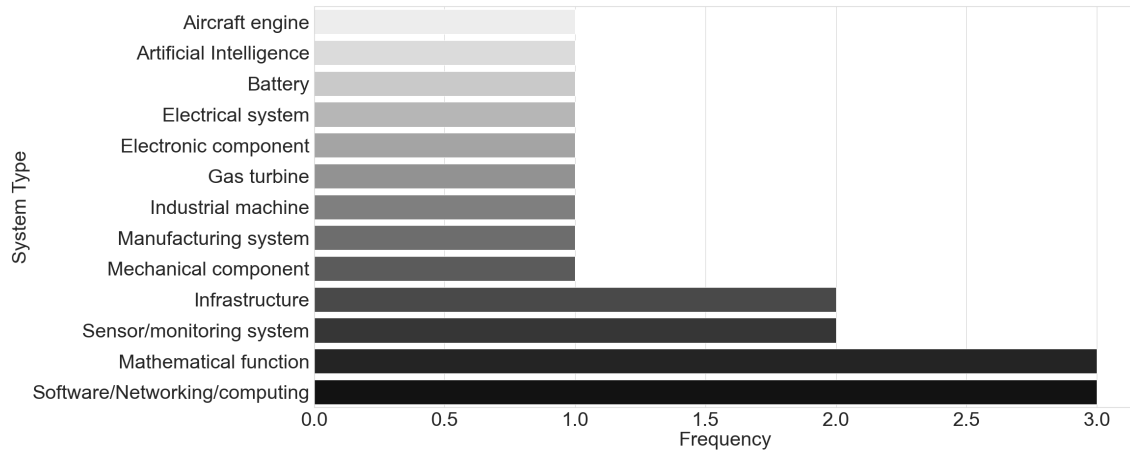


FIGURE C.15 – Types of systems under study.

particularly by analyzing the reliability of thermal pumps. In [57], a structure and method are developed to reduce the dimensionality of asset lifecycle data while minimizing information losses. This PHM application also focuses on feature engineering by introducing a data transformation method to prepare ML models for predictive maintenance. The proposed framework appears promising, but the article lacks an application to demonstrate the applicability and relevance of the method. On the other hand, in [56], a similar methodology is attempted in a case study in a machining center. The authors use a supervised learning technique of feature selection: minimum redundancy maximum relevance. The analyses demonstrate that the method has eliminated about ten redundant variables. Then, a model is built from historical data, and the results are used to produce a monitoring tool for enterprise management. A rule-based model is then used for predictive maintenance. In [81], a preventive maintenance model is developed to improve policies, in addition to presenting a cloud architecture for a predictive maintenance and corrective maintenance program with real-time detection. A case study is presented to analyze the residual life of equipment in a machining center. An artificial neural network is trained from historical data and then used for real-time monitoring. The article [82] presents a reliability study in the context of a semiconductor manufacturer. The objective is to present a model that can learn and

associate indicators with potential failures and determine rules or patterns of indicators or critical areas. They use Bayesian networks to determine probability distributions in the learning phase, then use the resulting network to learn patterns leading to failure. The results suggest that the model could be extended to real-time prediction applications. A growing area of research in reliability is natural language processing (NLP), particularly for leveraging historical data in the form of free text. In [15], the authors discuss the need to develop a new methodology, technical language processing, to adapt NLP to the context of short technical text analysis. Indeed, traditional NLP tools are not suitable for processing technical language contained in engineering databases; the texts are generally short and contain abbreviations or domain-specific language for example. The article proposes a new framework to address the reality of short texts contained in maintenance work orders. In [109], a case study is presented for the classification of maintenance data in a manufacturing company. CamemBERT, a pre-trained transformer architecture, is used for French language processing. In [40], a classification model is developed based on a pre-trained model for reclassifying maintenance orders in the context of an electrical utility. Some reviews of the literature also present interesting perspectives on reliability. The authors of [147] propose a summary of the literature on the application of the k-out-of-n: G system method for evaluating the reliability of the system. This method aims to analyze the reliability of a system consisting of n components, which can operate as long as k components function. This method is particularly useful in the analysis of complex systems. Finally, Ref. [148] provides a summary of the literature on statistical techniques in reliability, particularly for predicting failures and applications of heavy equipment in the mining industry. In addition, they propose to compare traditional methods with machine learning methods by analyzing case studies presented in the literature.

Table C.4 presents various programming tools, and Python libraries, which are widely used by the scientific community and in the field of RAMS engineering. In particular, the Reliability [83], Lifelines [84], and Scikit-survival [85] libraries allow for several statistical analyses relevant to RAMS, including parametric analyses with known distributions such as Weibull, Gamma, and Exponential, as well as survival analyses. The ProgPy [86] library is a recently developed library by NASA for PHM applications. The statsmodels [87] library is a general statistical library, while Scikit-Learn [88] is a general ML library containing a multitude of methods. TensorFlow [89] and Keras [90] are among the most widely used libraries for deep learning development. Finally, NLTK [91] and Spacy [88] are two open-access libraries offering easy and quick integration of tools for natural language processing.

Discussion

This section allows us to make several observations regarding the use of machine learning in the field of reliability. Although this review only lists the results of a few databases, it still allows us to have an insightful overview of the situation. The number of publications found on the subject has been quite low for the last 5 years (19 articles), although there is a growing trend for artificial intelligence applications. When you look at the data sources, this gives a hint as to why there are no more publications in the field. Indeed, most of the publications apply ML to fictive data, in particular with the aim of developing new methodologies. Case studies are quite rare: less than a quarter of publications. The difficulty to obtain good-quality operational data is probably part of the problem. However, the development of new methodologies can be expected to lead to case studies on real data. The data used in reliability are sometimes difficult to exploit, given the complexity of the data (manually entered data, free text fields, etc.). Moreover, machine learning requires a large amount of data, which means that it is necessary to have a large

database of historical maintenance work, which is a considerable constraint in itself. Given these factors, this also justifies why deep learning approaches are generally preferred in the field.

Future Work

This research introduces the concepts of machine learning and reliability engineering. Learning these concepts is essential to start applying advanced techniques in the industry. Although the general concepts may be perceived as well mastered by the community, the systematic review shows that there are still few applications that are made in real industrial contexts. With the rapid development of Industry 4.0 and the various enablers, it is becoming clear that it is only a matter of time before these methods spread throughout the industry. To do so, the research community needs to prove that machine learning can be applied to real, often imperfect data.

Conclusions

The field of reliability engineering has seen tremendous growth in recent years due to advancements in data acquisition and processing technologies. However, the integration of artificial intelligence (AI) techniques into the domain remains a complex task. The goal of this publication is to summarize the basic techniques used in reliability and machine learning in order to demonstrate how they can be applied in an industrial context. The first sections focused on providing a summary of the techniques by presenting the basics of modeling and statistical techniques, followed by a comprehensive overview of machine learning, artificial intelligence, and data science. These sections provided a solid foundation to understand the different types of techniques used for modeling. Then a systematic review of machine learning applied in reliability engineering was presented. An analysis

was done by linking the analysis domain (RAMS or PHM), the type of algorithm, and the type of system in which it is involved. Furthermore, the review compared the types of data used, whether synthetic or operational. Finally, an overview of related works was presented, introducing different machine learning applications and tools used in real-world applications. This provided a good overview of different use cases of machine learning for reliability engineering.

The findings of this review suggest that machine learning techniques are still not widely used in the reliability engineering field. The results also showed that most of the studies did not use operational data as input for their models, but rather used synthetic data or publicly available datasets. In addition, deep learning techniques (deep neural network) are the most widely used machine learning method in reliability.

In conclusion, the application of machine learning techniques in RAMS and PHM gives new opportunities for researchers and practitioners to optimize the decision-making processes and improve the reliability and performance of systems. This work provides a good foundation for researchers with an understanding of the field of machine learning applied to reliability engineering. However, there is still great difficulty in working with operational data, opening the way for applied research in data mining and natural language processing, particularly for the analysis of maintenance data.

Tableau C.4 – Python libraries for reliability engineering.

Library Name	Application	Library Function
Reliability	Reliability and Survival Analysis	Parametric Reliability Models (Weibull, Exponential, Lognormal, etc.), Non-Parametric Models (Kaplan–Meier, Nelson–Aalen, etc.), Accelerated Life Testing
Lifelines	Survival Analysis	Cox Proportional Hazard Model, Kaplan–Meier Estimator, Parametric Survival Models
Scikit-survival	Survival Analysis	Survival Tree, Ensemble Model for Survival Analysis, Cox Proportional Hazard Model, Kaplan–Meier Estimator
ProgPy Python Prognostics Packages	Prognostics	
Statsmodels	Statistics	Linear Regression, Generalized Linear Model, ANOVA
Scikit-Learn	Machine Learning	Classification, Regressions, Clustering, Cross-Validation/Model Selection Methods
TensorFlow	Deep Learning	Artificial Neural Network, Recurrent Neural Network, Convolutional Neural Network
Keras	Deep Learning	High-level API built on top of Tensor Flow for easy use
NLTK	Natural Language Processing	Tokenization, Sentiment Analysis, Stemming, Part-of-Speech Tagging
Spacy	Natural Language Processing	Text Classifier, Transformer Models, Custom Trainable Pipeline, Named Entity Recognition

Annexe D - Improving Maintenance Data Quality: Application of Natural Language Processing to Asset Management

Payette, M., Abdul-Nour, G., Meango, T.JM., Côté, A. (2023). In: Crespo Márquez, A., Gómez Fernández, J.F., González-Prida Díaz, V., Amadi-Echendu, J. (eds) 16th WCEAM Proceedings. WCEAM 2022. Lecture Notes in Mechanical Engineering. Springer, Cham.

Ce chapitre correspond à l'article publié dans le cadre de l'édition 2022 de la conférence «*World Congress on Engineering Asset Management*», présenté à Séville en Espagne.

Abstract

Artificial intelligence techniques are increasingly used for asset management. The abundance of data available in large electrical utility offers many application opportunities. The use of data-driven models can address some of the biases of physical models traditionally used in reliability engineering. However, in this context, as in many other fields of operation, the quality of data is often questioned by domain experts. Operational data are entered manually by maintenance technicians, and data entry errors are common. One of the errors that is observed is mislabeling of maintenance types, which can lead to poor statistical estimates of failure rate. This paper aims to improve the quality of historical maintenance data, to increase the accuracy of deployed models. To this end, the text fields available in the maintenance history is analyzed to predict the type of maintenance performed. Natural language processing (NLP) techniques are applied to solve this text classification problem. The models are applied to Hydro-Québec TransÉnergie's power transmission assets. The application of such techniques allows the enrichment of databases and thus reduces uncertainty in decisionmaking for asset management.

Introduction

The increasing complexity of industrial systems requires a high level of expertise, especially in the field of reliability engineering and asset management. Recent machine learning techniques now makes it possible to process Big Data and is a good opportunity for large organization to improve decision-making. The goal of Engineering of Asset Management (EAM) is to realize the value of company's assets. The asset management system (EAMS) aims to implement strategies to maximize the value of the asset throughout its life cycle [24, 25]. Thus, the EAM decision-making process is strongly reliant on information availability. Maintenance data is therefore crucial to the delivery of the asset's life cycle. Furthermore, some argue that data itself is an asset that can generate value for an organization [149]. Consequently, historical data should be maintained and improved, to extract the maximum value from them and for the physical asset's it describes. Hydro-Quebec operates one of the most important transmission systems in America and is one of the world's largest producers of hydroelectricity. The company was founded in 1944 and its assets are very diverse; some dating back to before the foundation are still in service. Maintenance histories are sometimes incomplete due to the age of the equipment and the various changes in data acquisition systems and coding makes it difficult to trace them. To improve the company's maintenance policies, it is necessary to model the reliability of equipment. In many cases, reliability is represented by qualitative models using methods such as the Failure Mode, Effects and Criticality Analysis (FMECA) or statistical lifetime distribution. However, this approach has its drawbacks, since it requires a deep knowledge of the equipment, its maintenance history and its operational environment. On the other hand, data-driven models can eliminate some of these disadvantages and provide an estimate of failure rates from the recorded history. Since data are manually entered by technicians,

the quality of data is often questioned by experts. It has been reported that mislabelling of data is common, leading to biased estimation of the failure statistics. Indeed, it would be relevant to address the problem at the source, by improving data entry methods for example. This would not change the quality of the data that is already available. As mentioned above, these historical data are assets, and should be treated as such. To increase the accuracy of the models deployed by Hydro-Quebec experts, this research seeks to improve the quality of historical maintenance data. One of the strengths of Hydro-Quebec is that the company has been historicizing maintenance data for more than 50 years. The information contained in the databases is very detailed and free text fields contain a significant amount of information that could be used to correct the labelling of maintenance types.

Literature review

Types of Maintenance

To understand the problem, it is imperative to define the different types of maintenance performed. Corrective or reactive maintenance is where the system is repaired when there is a failure that stops the system functions [38]. Preventive maintenance (PM) is when actions are planned and performed to avoid or reduce the occurrence of failures. It can be divided into several types. Systematic or interval maintenance is a predetermined periodic maintenance (ex: change of oil on a car). The interval is often specified by the system's manufacturer. Condition-based maintenance is performed when the system shows signs of wear that could lead to a failure [1]. This type of maintenance is triggered by an inspection, or anomaly detected by a set of sensors. Finally, predictive maintenance is where maintenance is performed when mathematical models predict an imminent risk of

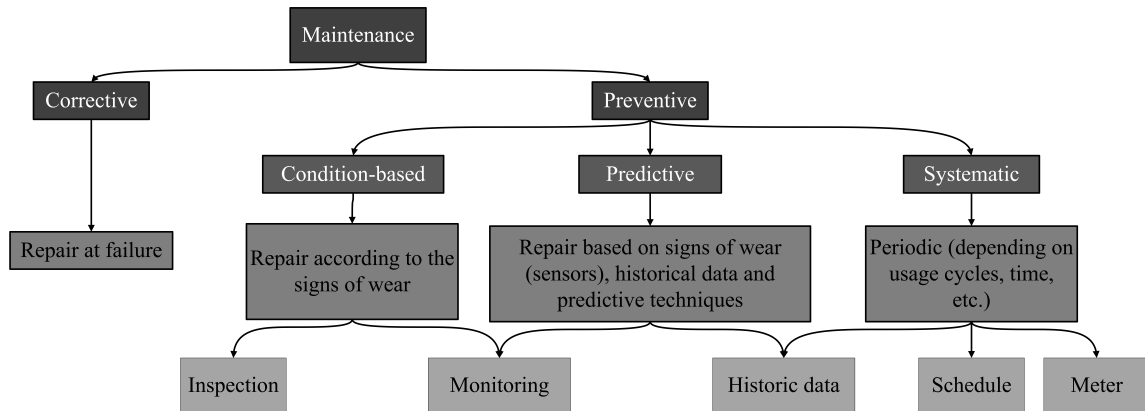


FIGURE D.1 – Maintenance classification

failure [150]. Figure D.1, adapted from [1], summarizes each maintenance type presented above.

Machine learning

Machine learning (ML) is a set of mathematical modelling techniques that allow a computer to learn the structure of a system from data and not by explicitly programming its structure [58]. There is a multitude of machine learning techniques, which are often grouped according to the way the computer learns from the data. The most commonly known learning methods are supervised learning and unsupervised learning. In supervised learning, the input data are labelled data, meaning that the dataset contains both the predictors and the variable to be predicted. The learning process is divided in three steps: training, validation and tests. In training, a sample of the data is sent, and the algorithm makes predictions on the predictors. Knowing the expected value, the algorithm parameters are modified so that the model is perfected at each iteration. Another sample of the data is used to find the best hyperparameters and the test sample evaluate the ability of the model to make predictions on new data [62]. In general, supervised learning is applied to classification problems, where the variable to be predicted is a class, or to regression where

the variable to be predicted is continuous. Unsupervised learning, on the other hand, uses unlabelled data as input. The goal is to find relationships between the variables themselves. In clustering, for example, the objective is to group data points into meaningful clusters, according to similarity metrics [63]. Supervised learning requires a large amount of labelled data, and even more for complex modelling tasks. As such, the amount of data required to build the models is very difficult to generate. To overcome this problem, two new branches of artificial intelligence (AI) have emerged: transfer learning and self-supervised learning. In both approaches, a model is pre-trained, with large public datasets for example, to perform a predictive task. Then the model is fine-tuned using the application specific dataset. In transfer learning, the model is pre-trained with labelled data as oppose to self-supervised learning where it is trained on unlabelled data [119, 151].

Text Pre-Processing

Text classification encompasses two problems in itself. Indeed, to perform classification with common ML methods (decision trees, vector support machines, etc.), the data must be understandable for a machine. A computer recognizes text as a sequence of characters with no particular meaning, so the text must be converted in a numerical form that encapsulates the words and their meaning. Natural language processing (NLP) is the field of computer science whose goal is for machines to interpret language in the same way as a human [10]. Tokenization is the process of separating each element of a sentence into a single fragment. Fragmentation can be done at the sentence level (cutting paragraphs into individual sentences) or at the word level. Lemmatization is the process of transforming words into their canonical forms (Ex: “I am being” converted to “I be”). It also includes the meaning of the word in its transformation [131]. Part-of-Speech tagging consists of evaluating words and their function in a sentence. The first step is to identify the category of the word (noun, verb,

adjective, etc.) and then to identify the gender and number of the word. For verbs, it is a matter of identifying the mode and tense of the verb. Then, it is a question of analysing the function of the words and their links between them.

Word Representation

As mentioned above, the goal of NLP is to represent text in a way that can be interpreted by a computer. There are several techniques for digitally representing text. The simplest technique is the bag-of-words. The principle is to establish a vocabulary from a corpus (set of texts) and, for each sentence, create a vector that counts each word. The bag-of-words can also be a binary representation, similar to one-hot-encoding. A more advanced technique is "word embedding", which consists in making a vector representation of words in several dimensions. The closer the words are, in the sense of the corpus, the closer they will be in the vector space. Vectors are generated using neural networks (NN), by taking each word and trying to predict its neighbourhood. The best-known approaches are the continuous-bag-of-words and the skip gram model. The drawback of these methods is that they do not take into account the meaning of sentences and words in their contexts. Each word has its own vector, even if the same word can have several meanings [131]. To address this problem, recurrent neural networks (RNR), in particular long short-term memory models (LSTM) allow generating a vector considering the words and their context in a sentence. The transformers, a recent neural network architecture, can also be used to create such a representation. The advantage of the transformers is that, unlike LSTMs and RNRs which operate sequentially, the architecture of the transformers allows for parallel operations, which greatly reduces the computation time. In addition, the attention mechanisms reduce the effect of vanishing gradient, compared to other RNRs, and thus allow for better understanding of dependencies in text [132]. Finally, the advantage of

using LSTM and transformers is that there are pre-trained networks easily available, and it is possible to use them for different machine learning tasks. Popular models based on Google’s Bidirectional Encoder Representations from Transformers (BERT) are generally trained in a self-supervised manner [151].

Methodology

Case Study and Problem Formalization

This section provides a formalization of the problem as well as the methodology that is used to solve it. As previously mentioned, experience with enterprise data has shown that historical maintenance data is often misclassified. When examining failure data, many data labelled as failures (corrective maintenance orders) turn out to be preventive maintenance, and vice versa. The goal of this work is to improve the quality of the data by correcting the type of maintenance recorded in the history. The dependent variable is the type of maintenance performed (preventive vs. corrective) and the predictors (independent variables) are extracted from the text fields contained in the maintenance orders and notices. This case study will only focus on those two categories, without distinction for the different types of preventive maintenance. From section 2.2, it’s clear that this problem can be defined as supervised learning (learning from labelled data), particularly a classification problem (categorical predictor) [62]. Document classification is an application of NLP and since there are only two categories, it is a binary classification problem.

Data Acquisition and Preprocessing

At Hydro-Quebec, there are two types of documents that detail maintenance work; the maintenance notice and maintenance order. The first is created following the detection of

a problem by a technician and describes the problems of the faulty device. A maintenance order is issued if it is deemed that the device needs to be repaired and contains the details of the work done. Another type of data is called forced equipment downtime. The electrical network is under constant surveillance, since electricity is an essential commodity. When there is a power outage, the equipment in the affected area are all considered unavailable. The employees in charge of monitoring the network will enter this unavailability data into the acquisition system. In theory, by cross-referencing work notices and orders with unavailability data, corrective maintenance should be easily identified. However, identifying correctly which equipment is failing, which is collateral and labelling the cause of failure is highly time consuming, considering the number of events and components involved. Furthermore, the equipment identification code differs from one database to the other. A lot of effort has been put to manually cross reference the two data sources by Hydro-Quebec experts, and a year of pre-processed failure data is now available. A cross-referencing method is developed to automate this process by [152], to unify the two data sources. The first step of the project is to extract all the maintenance notices and orders, and to examine the text fields. Then, the data cross-referenced by the experts is used as ground-truth to build the model.

Modelling, Validation and Testing

Once the training data is ready, different classification models will be tested. The results of two pre-trained models are used for the case study as a proof of concepts. The validation of the models is done through cross validation; a validation sample will allow comparing the two models and a test sample will allow to make sure of their quality. From the confusion matrix, the accuracy and recall will be calculated as performance metrics for the models. The models are built with the Python library Spacy [122]. Figure D.2 shows the

general pipeline for NLP. The raw text is the input and the Doc object is a representation of the text that is usable for modeling.

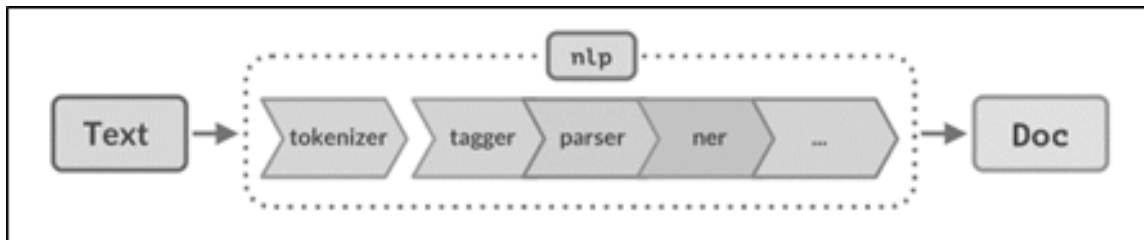


FIGURE D.2 – Spacy pipeline for NLP

For this case study, the general models cannot be used, since the text fields analysed are in French. Specific models designed for this purpose are applied. Figure D.3 shows the architecture of the French model:

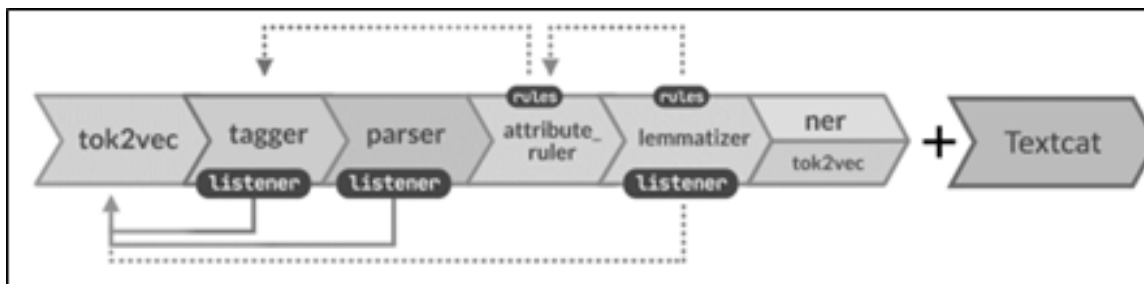


FIGURE D.3 – Text classification pipeline

The pipeline consists of a pre-trained network to which the components can be modified or added. In this case, the pipeline is modified to add text classification.

Results and Discussion

After extracting data from different databases, work orders are examined for a one-year period. Corrective and preventive maintenance are labelled according to previous studies

made on equipment unavailability. Due to class imbalances, the sample from PM is reduced, to have equal samples from corrective maintenance. Both models are trained on a sample of 120 entries. The first model implemented is the small model from the Spacy French model package and the second is the large model. Respective results are shown below.

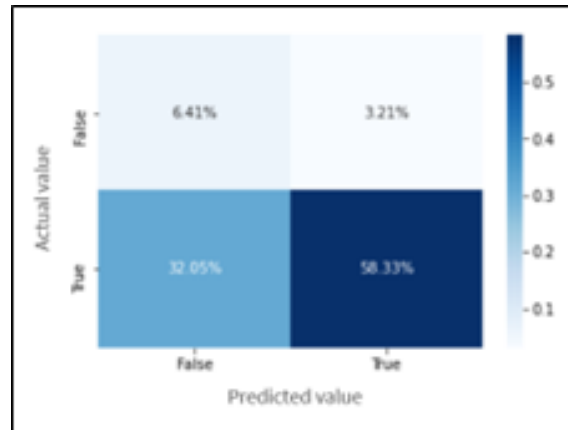


FIGURE D.4 – Confusion Matrix (small model)



FIGURE D.5 – Confusion Matrix (large model)

As results show, there is a slight improvement from the small to the large model, without a major impact on the runtime. The precision metrics indicate the proportion of positive identifications of corrective maintenance, which exceed 90% for both models.

Tableau D.1 – Comparison of Models

	Metric	
	fr_core_news_sm	fr_core_news_lg
Precision	94.79%	97.05%
Recall	64.54%	70.21%
Accuracy	64.74%	71.15%
Runtime (sec)	2.56	5.60

Recall indicates the percentage of overall correctly classified entry. At 64.54%, the small model is slightly better than a weak classifier. Recall increases at 70.21% for the large model.

Conclusion

The goal of this study was to create and compare text classification models to improve data quality in the context of an asset management program. Two models were trained to use a few text samples from two databases. Impressive results are shown from both the small and the large classification models built with Spacy pipelines. Indeed, the two models were trained with Spacy French models and when considering sample size, the prediction accuracy is relatively high. This case study proves nonetheless, improvement needs to be made on the dataset as well as the models. The next steps of this work are to build better models by increasing the size of the input data, using data from 5 years of maintenance work orders. In addition, the Spacy pipeline were mostly used as is; an implementation of the models with an optimization of the pipeline's hyperparameters will yield better results. In the next study, we also plan to test transformer architectures, which are the most advanced models for NLP.

Acknowledgement

This research was supported by Hydro-Québec, the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Université du Québec à Trois-Rivières through the Hydro-Québec Asset Management research Chair.