Quels jeux de données pour la prédiction d'anomalies dans l'industrie 4.0 ?

Mouhamadou Saliou Diallo*, Sid Ahmed Mokeddem*, Agnès Braud*, Gabriel Frey*, Nicolas Lachiche*

* ICube, Université de Strasbourg
300 Bd Sébastien Brant
67400 Illkirch-Graffenstaden
{ms.diallo, mokeddem, agnes.braud, g.frey, nicolas.lachiche}@unistra.fr

Résumé. L'industrie du futur est celle où les machines sont équipées de capteurs afin de suivre le bon déroulement des opérations. La prédiction d'anomalies, en particulier de pannes, est un enjeu important pour réduire les coûts liés aux arrêts des chaînes de production. Il faut des jeux de données variés pour mettre au point et tester des algorithmes d'apprentissage adéquats. Cet article établit les caractéristiques requises pour ces jeux de données et donne des exemples de jeux de données adaptés mais aussi de jeux de données inadaptés. Nous montrons un exemple de mise en oeuvre et proposons plusieurs perspectives de recherche.

1 Introduction

Le succès des usines dépend fortement de la fiabilité et de la qualité de leurs machines et de leurs produits. Les pannes imprévues des machines dans les processus de production entraînent des maintenances coûteuses et des retards de production (Borges et al., 2017). Par conséquent, comprendre et prévoir les situations critiques avant qu'elles ne se produisent peut être une source précieuse pour éviter les pannes imprévues et économiser les coûts associés à la défaillance. Les technologies de l'information et de la communication connaissent un développement rapide et de nombreuses technologies, telles que le cloud computing, l'internet des objets, les techniques d'analyse de gros volumes de données et l'intelligence artificielle, ont vu le jour. De nos jours, ces technologies sont omniprésentes notamment dans l'industrie où elles permettent entre autres la fusion des univers physique et virtuel grâce aux systèmes cyberphysiques (SCP), qui représentent la quatrième phase du développement industriel (c'est l'industrie 4.0). Grâce à ces technologies émergentes, les équipements des lignes de production industrielles sont de plus en plus connectés et produisent donc de gros volumes de données. Ce gros volume de données produites peut être analysé et transformé en connaissances afin de permettre aux décideurs de mieux gérer les maintenances.

Les approches de la gestion de la maintenance peuvent être regroupées en trois grandes catégories avec des niveaux de complexité et d'efficacité différents (Susto et al., 2015).

— La maintenance réactive : c'est l'approche la plus simple et la moins efficace. La maintenance n'est effectuée qu'après l'apparition de la panne entraînant donc un temps

d'intervention et temps d'arrêt des équipement beaucoup plus importants que ceux associées aux mesures correctives planifiées à l'avance ;
— La maintenance préventive : où les actions de maintenance sont effectuées selon un calendrier fixé à l'avance, basé sur des itérations de temps ou de processus. Avec cette approche, les défaillances sont généralement évitées, mais des actions correctives inutiles sont souvent effectuées, ce qui conduit à une utilisation inefficace des ressources et à une augmentation des coûts d'exploitation ;
— La maintenance prédictive : Dans cette approche, la maintenance est effectuée sur la base d'une estimation de l'état d'un équipement (Krishnamurthy et al., 2005). Les systèmes de maintenance prédictive permettent, grâce à des outils de prédiction basés sur des données historisées, de détecter à l'avance les anomalies à venir et d'intervenir en temps utile avant une défaillance.

Dans cet article, nous nous intéressons à la maintenance prédictive, et pour cela à la prédiction d'anomalies. Nous préférons le terme "prédiction d'anomalies" pour souligner que nous voulons prédire à un instant t qu'une anomalie va se produire dans le futur. Nous évitons le terme "détection d'anomalies" qui couvre aussi des cas où l'on veut constater à un instant t qu'il y a une anomalie en cours à ce même instant. Par exemple, le concours Kaggle sur la prédiction de performances de lignes de production (Mangal et Kumar, 2017) est initialement formulé comme un problème de détection d'anomalies, ainsi que nous le verrons en section 5. La prédiction d'anomalies tient explicitement compte du fait que les données sont séquentielles, ordonnées dans le temps.

On doit faire face à de nombreux défis :

1. les masses de données : les capteurs génèrent automatiquement une quantité de données qui atteint rapidement l'ordre du Go ;
2. le déséquilibre des données : les cas d'anomalies sont beaucoup plus rares que les cas normaux ;
3. la diversité des données : on doit souvent apprendre avec peu d'exemplaires d'une même machine, voire d'une même famille (mais de puissances différentes, par exemple).

Ces défis ne sont pas spécifiques aux données séquentielles. Cependant ils sont quasiment toujours présents, et plus prononcés, que lorsque l'on s'intéresse à d'autres domaines, en particulier sans données temporelles.

Des jeux de données sont indispensables pour concevoir, mettre au point et évaluer des algorithmes dédiés à la prédiction d'anomalies dans l'industrie 4.0. La section 2 établit les caractéristiques que doivent remplir des jeux de données pour l'entraînement et le test de tels algorithmes. La section 3 donne des exemples de jeux de données appropriés mais aussi des contre-exemples. Dans la section 4, nous montrons un exemple d'utilisation d'apprentissage profond pour prédire la durée de vie restante d'un turboréacteur. La section 5 présente un jeu de données massif réel proposé par Bosch et les défis associés. La section 6 conclut et liste quelques perspectives de recherche.

2 Caractéristiques nécessaires

Dans cet article, nous ne faisons pas de différence entre flux de données, données séquentielles et données temporelles. Nous supposons que les données arrivent en continu. Cela est

émulé quand nous travaillons à partir de jeux de données existants. Les données ont une estampille temporelle qui permet de les ordonner en données séquentielles. Nous n'imposons pas que la fréquence d'arrivée des données soit constante. L'ordre ou plutôt la séquence de ces données est la caractéristique importante. Certains algorithmes sont prévus pour apprendre à partir de séquences, par exemple les approches d'apprentissage profond comme LSTM (Ding et al., 2019; Zhang et al., 2018; Malhotra et al., 2015). À défaut, l'utilisation d'une fenêtre glissante de largeur fixe permet de transformer une séquence en des données attributs-valeurs classiques et ainsi d'accéder à des algorithmes qui ne sont pas dédiés aux séquences initialement, et donc à toute la panoplie disponible dans les librairies usuelles d'apprentissage.

En apprentissage artificiel, il est important d'identifier l'individu (ou, de façon complémentaire, la population) sur lequel on généralise. Dans le cas des données séquentielles, nous considérons que chaque instant correspond à un individu. Par exemple, si nous nous intéressons à une machine dont nous voulons prédire les pannes, nous collectons des données, avec une certaine fréquence. Chaque instant où les données sont collectées est un individu. Nous nous plaçons dans le cadre d'un apprentissage supervisé, c'est-à-dire celui où une étiquette est associée à chaque individu. L'objectif est d'apprendre un modèle qui prédit la valeur de cette étiquette pour un nouvel individu, qui est un autre instant dans notre cas. Par exemple, nous devons savoir pour chaque instant s'il est à moins de 3 jours de la prochaine panne, si nous imaginons que 3 jours est le délai raisonnable pour organiser la maintenance.

Suivant que l'étiquette est qualitative/catégorielle ou quantitative/numérique, nous ferons appel à des techniques de classification (supervisée) dans le premier cas ou de régression dans le second cas. Puisque nous nous intéressons à la prédiction d'anomalies, nous supposons que la collecte des données s'arrête lorsqu'une panne/anomalie survient. En effet, on peut supposer que la collecte des données reprendra lorsque la panne/anomalie aura été corrigée, et constituera une nouvelle séquence, de fait. Ainsi chaque séquence d'apprentissage se termine par une panne. Dans ce cas, si l'on adopte le point de vue de la classification supervisée, on peut définir l'étiquette comme un booléen indiquant si la panne va se produire dans moins de temps qu'un seuil donné, ce seuil correspondant au temps nécessaire pour planifier la maintenance et réorganiser la production. Alternativement, au lieu de faire de la classification en fixant un seuil, on peut adopter le point de vue de la régression et définir l'étiquette à prédire comme le temps restant avant la panne qui est appelé : Remaining Useful Life (RUL) en anglais.

Lorsque l'on parle de classification de séries temporelles, la tâche d'apprentissage est parfois présentée comme la classification d'une série entière (Bondu et al., 2019; Appice et al., 2014; Grabocka et al., 2014; Ye et Keogh, 2009), c'est-à-dire que l'étiquette n'est disponible que pour le dernier instant de la séquence. De fait, c'est un cas particulier par rapport au contexte que nous nous sommes fixés où chaque instant de la séquence a une étiquette.

La question de classer seulement le dernier instant ou chaque instant d'une séquence soulève plutôt la question du lien entre les séquences qui servent à l'apprentissage du modèle et des séquences sur lesquelles des prédictions seront faites à l'aide de ce modèle. Un cas particulier est celui d'un flux de données, donc une seule séquence de données, dont la partie déjà vue peut servir de données d'entraînement pour construire un modèle que l'on applique aux données qui suivent.

Dans de nombreux cas, on dispose de plusieurs séquences distinctes. La question est de savoir si ces séquences concernent la même machine. Si les séquences concernent différentes machines, les différences peuvent être de plusieurs natures :

— on peut avoir plusieurs exemplaires, physiques, d'un même modèle de machine ;
— les machines peuvent être d'une même famille, mais être de puissances différentes par exemple ;
— et on peut le généraliser à des machines "similaires", donc comparables pour l'apprentissage et pour la prédiction, mais éventuellement de familles différentes.

Dans tous les cas, la construction puis l'application d'un modèle impose de trouver une représentation commune, "comparable", des données, par exemple en intégrant la puissance, le modèle et la famille des machines considérées. Les données d'entraînement et de test sont dites représentatives si le modèle appris à l'aide des données d'entraînement peut être appliqué, avec pertinence, aux données de test. (Gay et Lemaire, 2019) remarquent que beaucoup de jeux de données sont fournis sous la forme d'un jeu d'entraînement et d'un jeu de test séparés. Dans ces jeux de données,les fréquences de la classe qui représente pour nous la présence d'une anomalie, sont très différentes. Ces jeux d'entraînement et de test ne sont donc pas issus de la même population et par conséquent pas représentatifs.

En conclusion,nous recommandons qu'un jeu de données soit constitué de séquences :

— toutes comparables entre elles, c'est-à-dire ayant une même représentation incluant une description de leurs points communs et de leurs différences, et permettant d'apprendre un modèle à partir de n'importe quel sous-ensemble de ces séquences et de pouvoir l'appliquer aux séquences restantes ;
— où chaque séquence se termine par une panne/anomalie, permettant de déterminer la valeur de l'étiquette pour chacun des instants de la séquence.

3 Exemples et contre-exemples

Dans cette section, nous examinons 3 jeux de données disponibles sur internet traitant de la prédiction de pannes. Nous décrivons chacun et vérifions s'il satisfait les caractéristiques requises plus haut.

3.1 Secom

Le jeu de données SECOM [1] concerne un processus de fabrication de semi-conducteurs. Il y a 591 attributs, pour un jeu de données comprenant 1567 exemples. Seulement 104 exemples se terminent par une panne. On ne peut pas utiliser les 1463 autres "exemples" dans une approche supervisée car on ne peut pas leur attribuer une étiquette puisque l'on ne sait pas quand la panne va se produire (en réalité, quand la panne s'est produite pour les données d'entraînement et de test). De plus ces séquences sont courtes, 18 estampilles temporelles en moyenne, voire 3 pour la plus courte. Cela ne laisse pas le temps d'organiser une maintenance et il n'est donc pas pertinent d'essayer de prédire une panne dans ce contexte.

3.2 Li-ion Battery Aging Datasets

Le jeu de données sur le vieillissement de batteries lithium-ion [2] est destiné à la prédiction de la charge restante d'une part, et de la durée de vie restante (avant que la capacité de la bat-

1. https ://archive.ics.uci.edu/ml/datasets/secom
2. https ://data.nasa.gov/dataset/Li-ion-Battery-Aging-Datasets/uj5r-zjdb

terie n'ait diminué de plus de 30%), d'autre part. Chaque séquence est composée de cycles de charge et de décharge. 5 à 7 attributs sont mesurés en fonction de l'état courant du cycle. Mais seulement 4 batteries sont testées. Cela fournit trop peu d'exemples sur lesquels apprendre et tester.

3.3 Durée de vie des disques durs

Le site de backblaze fournit chaque trimestre des statistiques et des données sur la durée de vie de ses disques durs. [3] Sur la Figure 1, on voit que fin septembre 2019 l'entreprise a collecté des données sur 112 864 disques durs. 6 078 ont rendu l'âme. On ne peut pas utiliser les 106 786 autres disques qui n'ont pas encore failli. Les disques proviennent de 4 fabricants et sont de plusieurs modèles. Si l'on veut se servir de tous pour apprendre et tester, il faut s'assurer qu'ils ont tous les mêmes descripteurs et que les descripteurs incluent les caractéristiques des disques (fabricant, modèle, capacité).souvent,les études s'intéressent au modèle le plus fréquent (Basak et al., 2019; Anantharaman et al., 2018; Basak et al., 2018; Su et Li, 2019). Il y a 3724 exemplaires du modèle ST4000DM000 de Seagate qui sont tombés en panne. Cela fournit un jeu de données homogène, dans lequel les disques sont décrits par les mêmes attributs. En fait les données contiennent la date (puisqu'elles sont collectées par jour), le numéro de série et le modèle du disque, sa capacité, un indicateur de panne, et 45 attributs S.M.A.R.T. (Self-Monitoring, Analysis and Reporting Technology) avec leurs valeurs brutes et normalisées. Un prétraitement est nécessaire pour générer les séquences, avec un fichier par numéro de série contenant une ligne par date, puisque les données sont fournies par date avec une ligne par numéro de série. Ainsi les parties de ce jeu de données qui concernent des disques qui sont tombés en panne et qui ont les mêmes descripteurs peuvent être utilisées pour entraîner et tester des modèles de prédiction de pannes.

Nous allons examiner de plus près deux autres jeux de données.

4 Apprentissage profond sur turbofan

Plusieurs jeux de données de simulation sur l'usure de turboréacteurs ont été publiés par la NASA [4] (Saxena et al., 2008). Nous considérons ici le premier jeu de données pour lequel 100 séquences se terminant par une panne sont disponibles. Ces séquences ont des longueurs comprises entre 128 et 362, avec une moyenne de 206 instants. On dispose des valeurs relevées par 24 capteurs à chaque instant. Chaque instant a été étiqueté par la durée de vie restante avant la panne.

70 séquences ont été utilisées pour construire un modèle à l'aide de LSTM. La Figure 2 illustre l'architecture utilisée. Les 30 séquences restantes servent de jeu de test. La Figure 3 contient une capture d'écran du logiciel de prédiction que nous avons développé.

Les 5 premiers graphes, dans l'ordre de lecture de gauche à droite et de haut en bas, représentent en temps réel les valeurs des 24 capteurs. Le graphe en bas à droite représente les prédictions effectuées par notre modèle (courbe en rouge) et la ligne bleue est la durée de vie restante réelle. La capture d'écran a été faite à la fin de l'exécution du programme, c'est-à-dire

3. https ://www.backblaze.com/b2/hard-drive-test-data.html
4. https ://data.nasa.gov/dataset/Turbofan-engine-degradation-simulation-data-set/vrks-gjie

Jeux de données pour la prédiction d'anomalies

Backblaze Lifetime Hard Drive Annualized Failure Rates

For hard drive models in service as of September 30, 2019
Reporting period April 2013 - September 2019 inclusive

MFG	Model	Drive Size	Drive Count	Avg. Age	Drive Days	Drive Failures	AFR*
HGST	HMS5C4040ALE640	4TB	2,707	42.0	11,420,392	161	0.51%
HGST	HMS5C4040BLE640	4TB	12,641	35.6	18,409,871	233	0.46%
HGST	HUH728080ALE600	8TB	1,001	22.3	746,311	16	0.78%
HGST	HUH721212ALE600	12TB	1,560	4.8	183,560	4	0.80%
HGST	HUH721212ALN604	12TB	10,849	6.1	1,923,518	25	0.47%
Seagate	ST4000DM000	4TB	19,330	47.3	50,839,992	3,724	2.67%
Seagate	ST6000DX000	6TB	886	53.9	2,821,207	83	1.07%
Seagate	ST8000DM002	8TB	9,839	36.3	10,910,157	316	1.06%
Seagate	ST8000NM0055	8TB	14,416	26.8	11,856,443	386	1.19%
Seagate	ST10000NM0086	10TB	1,200	24.3	897,426	14	0.57%
Seagate	ST12000NM0007	12TB	37,116	15.4	17,458,380	1,102	2.30%
Toshiba	MD04ABA400V	4TB	99	52.3	225,739	5	0.81%
Toshiba	MG07ACA14TA	14TB	1,220	11.9	441,195	9	0.74%
		Totals	112,864		128,134,191	6,078	1.73%

* AFR - Annualized Failure Rate

BACKBLAZE

FIG. 1 – *Statistiques de Backblaze en septembre 2019*

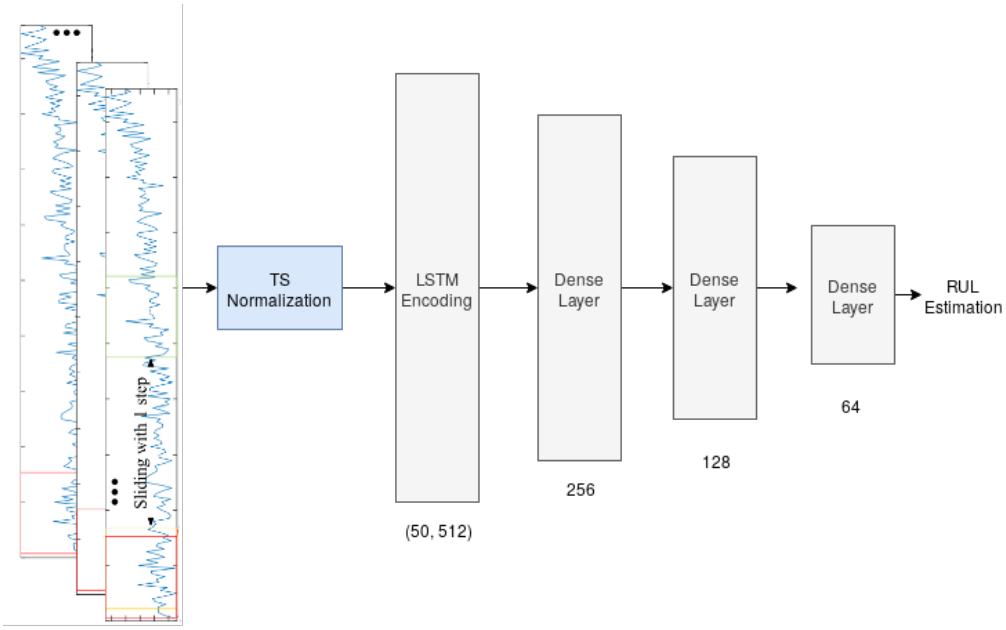

FIG. 2 – *Architecture LSTM*

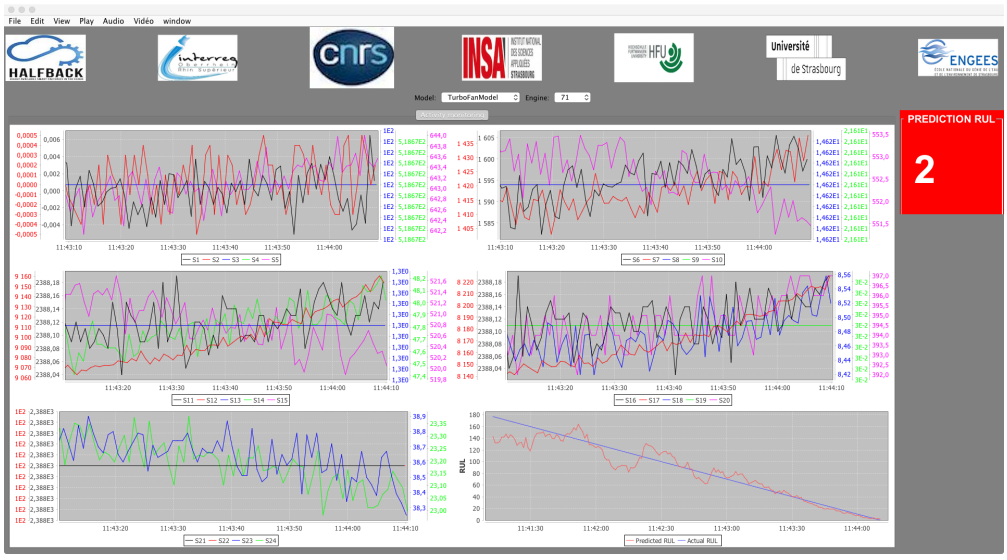

FIG. 3 – *Notre logiciel de prédiction de la durée de vie restante*

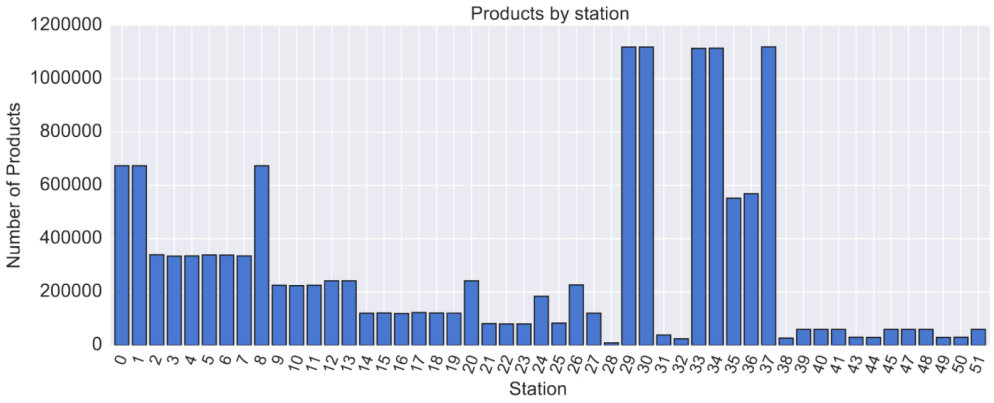

FIG. 4 – *Products count passing through each station*

avec les prédictions du début à la fin de la séquence testée. Une vidéo annotée [5] est également disponible pour mieux visualiser ce travail.

5 Les défis du jeu de données proposé par Bosch

L'entreprise Bosch a proposé un concours sur le site Kaggle [6]. Ce concours concerne la prédiction de la qualité des pièces produites. Le jeu de données d'entraînement est relatif à 1 184 687 produits, décrits par 968 valeurs numériques des différents capteurs et 1156 estampilles temporelles correspondant aux instants de passage par les différents capteurs. Les données ne sont pas plus détaillées par l'entreprise. On sait seulement qu'il y a 51 stations sur un total de 4 lignes de production.

On peut déjà remarquer que :

— Le jeux de données proposé est de grande taille (14,3 Go). Chaque opération sur une telle masse de données est difficile ;

— Il n'y a que 6 879 produits défectueux, soit 0,58% des produits. Les données sont extrêmement déséquilibrées ;

— De nombreuses valeurs sont manquantes. Comme on peut le voir sur la Figure 4 présentant le nombre de pièces passant sur chacune des 51 stations, la totalité des pièces ne passent pas par toutes les stations, donc elles n'ont pas de valeurs pour les attributs relatifs aux stations où elles ne passent pas. Nous reviendrons sur cette difficulté.

Pour ces raisons, le concours proposé sur Kaggle est difficile. Cependant, sous cette forme, ce n'est pas à proprement parler un problème de maintenance prédictive. En effet le concours demande de "prédire" la qualité de la pièce décrite par les valeurs mesurées lors de sa production, donc une fois qu'elle est déjà produite. On utilise les valeurs des descripteurs à l'instant t pour prédire la classe à ce même instant t. Il n'y a pas prédiction de la classe à partir de données collectées avant le début de la fabrication de la pièce. Pourtant c'est ainsi que l'on

5. http ://icube-sdc.unistra.fr/en/index.php/HALFBACK
6. https://www.kaggle.com/c/bosch-production-line-performance

pourrait éviter de produire des pièces défectueuses au lieu de constater leurs mauvaise qualité a posteriori.

Nous proposons d'utiliser les estampilles temporelles afin de reconstruire la séquence dans laquelle les pièces ont été produites et d'utiliser cette séquence pour prédire la qualité des prochaines pièces à partir des pièces déjà produites. De plus, nous supposons que chaque fois qu'une pièce est défectueuse, une correction a été apportée sur la chaîne de production et donc que la pièce défectueuse indique la fin d'une séquence et qu'après elle commence une nouvelle séquence. Les pièces deviennent les instants sur lesquels on travaille. Il reste plusieurs défis à relever :

— Les données restent massives et rendent chaque opération très longue ;
— Le nombre de pièces défectueuses devient le nombre de séquences. Le ratio du nombre d'instants où une pièce est défectueuse ne change pas, et de même si l'on introduit un seuil sur la durée avant la pièce défectueuse. En revanche, si l'on veut estimer la durée de vie restante, Remaining Useful Life (RUL) en anglais, chaque instant/pièce de la séquence peut être étiquetée par le nombre de pièces le/la séparant de la fin de la séquence ;
— La principale difficulté vient des valeurs "manquantes". Les valeurs des attributs des stations où les pièces ne passent pas ne sont pas manquantes. On pourrait leur attribuer une valeur explicite, "ne passe pas par cette station". Cependant tous les attributs sont numériques et il est difficile de donner une valeur numérique à "ne passe pas par cette station". De plus, les pièces qui ne passent pas une station rallonge artificiellement la séquence des pièces. Prenons l'exemple d'une fraiseuse. On comprend bien que l'outil de coupe s'use en fonction des pièces qui y sont réellement usinées. Mais entre les estampilles temporelles de deux pièces qui sont usinées par cette fraiseuse, il peut y avoir les estampilles temporelles d'autres pièces qui "ne passe(nt) pas par cette station" mais sont produites à la même période. Ces autres pièces vont s'intercaler, de façon indépendante de cette fraiseuse. Cela implique alors que mesurer la durée de vie de la fraise par le nombre de pièces passées dans l'usine est incorrect. En fait il ne faudrait considérer que les pièces passées par cette fraiseuse pour calculer la durée de vie restante. Mais l'information sur la cause de la pièce défectueuse n'est pas communiquée. La machine-outil défectueuse est portant connue par l'entreprise a posteriori, puisque le défaut a été corrigé afin de reprendre la production, mais cette information n'est pas fournie. C'est la vraie valeur manquante dans ce jeu de données.

6 Conclusion et perspectives

Pour apprendre à prédire des pannes, il faut des séquences en nombre suffisant et de durée suffisante. Il est nécessaire aussi que les séquences se terminent par une panne afin de pouvoir étiqueter chacun des instants par rapport à la fin de la séquence.

On peut imaginer des séquences de test, voire d'entraînement, qui ne se terminent pas par une panne, à condition que chaque instant soit étiqueté, donc que l'on connaisse la fin de la séquence. Les séquences de test du jeu de données sur la dégradation des turboréacteurs sont dans ce cas : elles se terminent par un instant étiqueté par la durée avant la panne. Les instants suivants ne sont pas fournis. Pourrait-on pour autant apprendre et tester qu'avec des séquences se terminant avant la fin, par exemple 30 unités de temps (jours dans le cas des disques durs) ?

Ce serait acceptable si on sait que 30 unités de temps avant la fin est beaucoup trop tard pour intervenir, planifier une maintenance, et que c'est avant 30 unités de temps que notre prédiction est utile et pour cela il est préférable qu'elle soit précise. Cette question est liée à une question plus générale sur la période où l'on souhaite prédire la panne.

La période où l'on souhaite prédire la panne est évidemment celle où il faut planifier la maintenance. Sa valeur exacte est indiquée par les experts, en prenant en compte le temps qu'il faut pour organiser la maintenance. Du point de vue de l'apprentissage artificiel, cela veut dire que l'on s'intéresse plus particulièrement à une plage de valeur. Si l'on prend l'exemple des prédictions sur le graphe en bas à droite de la Figure 3, on peut imaginer que les erreurs pour des valeurs supérieures à 80 sont moins cruciales que celles pour des valeurs plus proches de la panne. C'est un aspect considéré en apprentissage artificiel, par exemple lorsque l'on privilégie une partie de l'aire sous la courbe ROC (Narasimhan et Agarwal, 2013a,b; Dodd et Pepe, 2003; Wang et Chang, 2011; Ye et al., 2019), mais à notre connaissance cela n'a pas été étudié spécifiquement pour les séries temporelles.

En classification supervisée, il faut distinguer les faux positifs des faux négatifs. En régression, on peut distinguer une surestimation d'une sous-estimation. En effet, il est plus grave de prédire et donc planifier une maintenance trop tard que plus tôt. Cet aspect a été étudié dans le cas général par exemple par (Hernández-Orallo, 2013) mais pas dans le cas particulier des séries temporelles.

Remerciements

Ces travaux ont été financés par INTERREG Haut Rhin (Fonds Européen de Développement Régional) et les Ministères de la Recherche du Bade-Württemberg, de Rheinland- Pfalz (Allemagne) et du Grand Est (France) dans le cadre du projet Offensive Science HALFBACK.

Références

Anantharaman, P., M. Qiao, et D. Jadav (2018). Large scale predictive analytics for hard disk remaining useful life estimation. In *2018 IEEE International Congress on Big Data (BigData Congress)*, pp. 251–254.

Appice, A., M. Ceci, C. Loglisci, G. Manco, E. Masciari, et Z. W. Ras (Eds.) (2014). *New Frontiers in Mining Complex Patterns - Second International Workshop, NFMCP 2013, Held in Conjunction with ECML-PKDD 2013, Prague, Czech Republic, September 27, 2013, Revised Selected Papers*, Volume 8399 of *Lecture Notes in Computer Science*. Springer.

Basak, S., S. Sengupta, et A. Dubey (2018). A data-driven prognostic architecture for online monitoring of hard disks using deep LSTM networks. *CoRR abs/1810.08985*.

Basak, S., S. Sengupta, et A. Dubey (2019). Mechanisms for integrated feature normalization and remaining useful life estimation using lstms applied to hard-disks. In *2019 IEEE International Conference on Smart Computing (SMARTCOMP)*, pp. 208–216.

Bondu, A., D. Gay, V. Lemaire, M. Boullé, et E. Cervenka (2019). FEARS : a feature and representation selection approach for time series classification. In W. S. Lee et T. Suzuki (Eds.), *Proceedings of The 11th Asian Conference on Machine Learning, ACML 2019, 17-19 No-*

vember 2019, Nagoya, Japan, Volume 101 of *Proceedings of Machine Learning Research*, pp. 379–394. PMLR.

Borges, J., M. A. Neumann, C. Bauer, Y. Ding, T. Riedel, et M. Beigl (2017). Predicting target events in industrial domains. In P. Perner (Ed.), *Machine Learning and Data Mining in Pattern Recognition*, Cham, pp. 17–31. Springer International Publishing.

Ding, N., H. Ma, H. Gao, Y. Ma, et G. Tan (2019). Real-time anomaly detection based on long short-term memory and gaussian mixture model. *Computers & Electrical Engineering 79*.

Dodd, L. E. et M. S. Pepe (2003). Partial auc estimation and regression. *Biometrics 59 3*, 614–23.

Gay, D. et V. Lemaire (2019). Should we reload time series classification performance evaluation ? (a position paper). *CoRR abs/1903.03300*.

Grabocka, J., N. Schilling, M. Wistuba, et L. Schmidt-Thieme (2014). Learning time-series shapelets. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, New York, NY, USA, pp. 392–401. ACM.

Hernández-Orallo, J. (2013). ROC curves for regression. *Pattern Recognition 46*(12), 3395–3411.

Krishnamurthy, L., R. Adler, P. Buonadonna, J. Chhabra, M. Flanigan, N. Kushalnagar, L. Nachman, et M. Yarvis (2005). Design and deployment of industrial sensor networks : Experiences from a semiconductor plant and the north sea. In *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems*, SenSys '05, New York, NY, USA, pp. 64–75. ACM.

Malhotra, P., L. Vig, G. Shroff, et P. Agarwal (2015). Long short term memory networks for anomaly detection in time series. In *23rd European Symposium on Artificial Neural Networks, ESANN 2015, Bruges, Belgium, April 22-24, 2015*.

Mangal, A. et N. Kumar (2017). Using big data to enhance the bosch production line performance : A kaggle challenge. *CoRR abs/1701.00705*.

Narasimhan, H. et S. Agarwal (2013a). A structural SVM based approach for optimizing partial AUC. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, Volume 28 of *JMLR Workshop and Conference Proceedings*, pp. 516–524. JMLR.org.

Narasimhan, H. et S. Agarwal (2013b). $\text{Svm}_{\text{pauc}}^{\text{tight}}$: a new support vector method for optimizing partial AUC based on a tight convex upper bound. In I. S. Dhillon, Y. Koren, R. Ghani, T. E. Senator, P. Bradley, R. Parekh, J. He, R. L. Grossman, et R. Uthurusamy (Eds.), *The 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2013, Chicago, IL, USA, August 11-14, 2013*, pp. 167–175. ACM.

Saxena, A., K. Goebel, D. Simon, et N. Eklund (2008). Damage propagation modeling for aircraft engine run-to-failure simulation. In *2008 international conference on prognostics and health management*, pp. 1–9. IEEE.

Su, C.-J. et Y. Li (2019). Recurrent neural network based real-time failure detection of storage devices. *Microsystem Technologies*.

Susto, G. A., A. Schirru, S. Pampuri, S. McLoone, et A. Beghi (2015). Machine learning for predictive maintenance : A multiple classifier approach. *IEEE Transactions on Industrial*

Informatics 11(3), 812–820.

Wang, Z. et Y.-C. I. Chang (2011). Marker selection via maximizing the partial area under the roc curve of linear risk scores. *Biostatistics 12 2*, 369–85.

Ye, L. et E. Keogh (2009). Time series shapelets : A new primitive for data mining. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, New York, NY, USA, pp. 947–956. ACM.

Ye, W., Y. Lin, M. Li, Q. Liu, et D. Z. Pan (2019). Lithoroc : lithography hotspot detection with explicit roc optimization. In *ASP-DAC*.

Zhang, Y., R. Xiong, H. He, et M. G. Pecht (2018). Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries. *IEEE Transactions on Vehicular Technology 67*(7), 5695–5705.

Summary

Industry 4.0 is characterized by the availability of sensors to operate the so-called smart factory. Anomaly prediction, in particular failure prediction, is an important issue to cut the costs associated to production breaks. Various datasets are needed to design and test dedicated learning algorithms. This article sets up the requirements on such datasets and gives examples of appropriate and inappropriate datasets. We highlight one example of application and raise several perspectives for research.