Apprentissage multimodal pour le diagnostic de fautes sur données séquentielles non alignées et arbitrairement longues

Victor Pellegrain[1,2], Myriam Tami[2], Michel Batteux[1], Céline Hudelot[2]

[1] IRT SystemX
[2] Université Paris Saclay, CentraleSupélec, MICS

victor.pellegrain@irt-systemx.fr

Résumé

La complexité toujours grandissante des systèmes industriels amène de nouveaux verrous scientifiques pour les tâches liées à la maintenance prévisionnelle. Dans cet article nous présentons une revue des méthodes utilisées pour réaliser un diagnostic de fautes, et pointons leurs limites pour gérer des données multi-sources et hétérogènes, propres à l'industrie 4.0. Nous formalisons théoriquement ce nouveau cadre et proposons StreaMulT, une architecture permettant de gérer des séquences multimodales au fil de l'eau, non alignées et arbitrairement longues.

Mots-clés

Apprentissage multimodal, Diagnostic, Dépendances à long terme, Données non alignées, Séquences arbitrairement longues

Abstract

The industry 4.0 era brings more and more complexity to industrial systems, resulting in new challenges for predictive maintenance strategies. This work presents a review of the methods developed to perform fault diagnosis, along with their limitations to handle heterogeneous multi-sources data. We theoretically formalize this new applicative setting and propose StreaMulT, a Streaming Multimodal Transformer able to manage heterogeneous, unaligned and arbitrary long input sequences in a streaming fashion.

Keywords

Multimodal learning, Fault diagnosis, Long-term dependencies, Unaligned data, Arbitrary long sequences

1 Introduction

Un système industriel peut rencontrer des *défaillances*, se définissant par l'incapacité de ce système à réaliser au moins une de ses fonctions requises [17]. Ces défaillances peuvent mener à l'occurrence d'événements indésirables et redoutés, avec des conséquences plus ou moins importantes selon la criticité du système. Les occurrences de défaillances font souvent suite à la présence de *fautes*, à savoir une condition anormale du système caractérisée par la déviation d'une de ses caractéristiques par rapport à une valeur de référence acceptable. Afin d'éviter de telles occurrences, il est souvent nécessaire de considérer les techniques de diagnostic de fautes (DF) : leur détection puis leur isolation et identification, consistant à classifier le type de faute. Ainsi, les termes "isolation et identification" et "classification" sont interchangeables comme indiqué par [32]. Ces étapes sont essentielles dans une politique de maintenance préventive. De nombreuses approches de DF ont été utilisées, historiquement catégorisées entre les méthodes dites "basées modèle" et les méthodes dites "basées données".

Jusqu'à encore récemment, l'immense majorité des données acquises sur les systèmes était composée de séries temporelles, décrivant des grandeurs physiques locales comme la température, la pression, la vibration, etc. Aujourd'hui, l'ère de l'industrie 4.0 place l'interconnectivité et l'automatisation intelligente au centre du schéma de production industrielle. Cela se traduit essentiellement par l'intégration d'une multitude de capteurs connectés dans les machines ou systèmes industriels, dans le but de créer des systèmes de contrôle global appelés SCADA (Supervisory Control And Data Acquisition). Ces nombreux capteurs permettent ainsi l'acquisition d'une grande quantité de données issues d'une multitude de sources. En conséquence, ils fournissent plus d'information pour guider les modèles d'apprentissage et ainsi améliorer leurs performances. Cependant, ces flux de données multi-sources sont de nature hétérogène : séries temporelles issues de mesures de capteurs, textes issus de rapports d'intervention, images issues de prises de vue d'éléments du système. Ces différentes sources de données peuvent également présenter une hétérogénéité dans leur fréquence d'acquisition. En effet, si les grandeurs physiques sont mesurées régulièrement à une période de l'ordre de la seconde, des images sont acquises de façon plus éparse dans le temps, lorsque des rapports textuels d'intervention sont enregistrés encore plus rarement et à une fréquence sporadique. Cette double hétérogénéité entre les données multi-sources constitue un véritable verrou scientifique pour les modèles d'apprentissage usuels, qui sont généralement conçus pour exploiter la structure d'un type de données particulier, commune à tous les exemples d'entraînement. Ce verrou explique l'absence d'approches exploitant simultanément différents types de données dans la littérature de la surveillance et du diagnostic, pourtant es-

sentielles pour exploiter ces nouveaux jeux de données dans leur entièreté.
La prise en compte de données de natures hétérogènes est justement le problème auquel s'attaque la communauté de l'apprentissage multimodal [6, 13], et ce à différentes fins comme par exemple la fusion de modalités. En revanche, les méthodes actuelles de l'état de l'art se concentrent essentiellement sur des données statiques (une image et sa description textuelle par exemple), ou sur des données temporelles de taille fixe et généralement courte (clips vidéos de quelques secondes par exemple) et ne sont donc pas applicables en l'état sur des flux de données de capteurs industriels arbitrairement longs. Or, ce type de données est classique pour le cadre applicatif du DF. Notre contribution est donc triple, et dresse le plan de cet article :

— Dans la section 2 nous dressons un état de l'art des approches utilisées pour le DF, ainsi que pour l'apprentissage multimodal. Nous soulignons les forces et les limites de ces approches dans notre cadre applicatif.
— Nous formalisons ensuite dans la section 3 un nouveau cadre théorique modélisant la tâche de détection et classification simultanées de fautes à partir de données multimodales et arbitrairement longues. Ce cadre est essentiel pour exploiter de tels jeux de données propres à l'industrie 4.0, mais pas exclusivement.
— Enfin, dans la section 4 nous présentons StreaMulT, un modèle d'apprentissage profond basé sur une architecture Transformer [50], et capable de gérer des données multimodales, de fréquences d'acquisition hétérogènes, non alignées et arbitrairement longues. Nous validons également cette architecture expérimentalement.

2 Travaux antérieurs

2.1 Diagnostic

Un des premiers travaux à avoir listé et ordonné les différentes méthodes de DF est la série de trois articles de Venkatasubramanian et al. [51]. Cette revue, qui constitue le point de départ de notre étude, classifie les approches de DF selon la connaissance a priori que le concepteur a sur les différentes fautes pouvant survenir, ainsi que sur leur expression à travers les données acquises du système (symptômes de faute). Les stratégies utilisant cette connaissance a priori en représentant le système par un modèle physique sont dites "basées modèle", et différenciées entre qualitatives et quantitatives selon les représentations mathématiques utilisées ; les approches exploitant l'historique des données sont logiquement dites "basées données". Si les méthodes basées modèle fonctionnent bien lorsque le concepteur possède une bonne compréhension a priori des lois physiques régissant le système, elles sont difficilement exploitables dans le cas contraire. Ainsi, à un certain stade de complexité du système considéré, les interactions entre les composants sont difficilement modélisables. Dans ce cas, les méthodes basées données sont une alternative adéquate : le modèle utilisé apprend ces relations à partir de l'historique des données. Nous nous concentrerons sur les approches basées données dans cette étude, et plus précisément les approches de machine learning (ML).

Machine Learning. De nombreuses revues (présentées ci-après) ont répertorié les approches de ML appliquées au DF. Notre but n'est donc pas ici de donner une liste exhaustive des différents modèles utilisés, mais plutôt de dresser un état des lieux des différentes positions de ces articles, des challenges auxquels ils répondent, et leurs limites par rapport aux nouveaux défis de l'industrie 4.0.
Certaines revues de la littérature adoptent une position propre à un domaine applicatif industriel. C'est notamment le cas des articles [32, 72, 38], qui font part des méthodes de DF appliquées respectivement aux systèmes de traitements chimiques, aux roulements, ou aux systèmes d'air conditionné. Ces études motivent ainsi principalement leur démarche par les conséquences liées à l'apparition de fautes dans leurs domaines respectifs, comme la surconsommation d'électricité et des coûts économiques importants [38]. Les approches évoquées sont également présentées comme adaptées aux jeux de données propres à ces domaines : Zhang et al. [72] considèrent essentiellement des données de vibration et de courant de moteur, comme sur le jeu de données Paderborn [*] ; tandis que Rogers et al. [38] présentent des modèles utilisant essentiellement des données de température et d'humidité, et incitent la communauté à travailler sur des systèmes de thermostat. A l'opposé, d'autres travaux adoptent une posture plus méthodologique dans leur présentation de l'état de l'art des méthodes de DF [33]. Les plus récents [3, 37, 23] motivent leur démarche par l'apparition de nouveaux challenges pratiques liés à l'arrivée de l'industrie 4.0, comme notamment la capacité à gérer des quantités massives de données multi-sources en temps rapide.
Ces études présentent les approches de ML comme plus adaptées lorsque les profils de fautes sont complexes. Ainsi, Zhang et al. [72] font part de la limite des approches basées modèles à détecter précocement des fautes en raison de symptômes non traçables par ce type de modèles, ou à correctement démêler la présence de plusieurs fautes simultanément. Si certains travaux cités ne traitent que de la détection de fautes [28, 57], la majorité considère également la partie isolation et identification, bien que les auteurs de [3] utilisent parfois le terme de diagnostic pour évoquer la détection seule. En revanche, comme souligné par Reis et al. [37], en pratique deux méthodologies différentes coexistent : si la communauté de la maîtrise statistique des procédés (MSP) traite les tâches de détection et d'isolation et identification de fautes de manière séquentielle, la communauté du ML les traite parfois simultanément, sous la forme d'une classification en $C + 1$ classes, décomposées en une classe de fonctionnement normal, et C classes de fautes distinctes.
Comme présenté dans [23], les modèles de ML pour le DF

∗. Available online : https ://mb.uni-Paderborn.de/kat/forschung/datacenter/bearing-datacenter

sont généralement composés d'un module d'extraction de caractéristiques, fournissant à la suite du modèle des éléments pertinents depuis les données brutes, et d'un module de diagnostic. Certains modules d'extraction de caractéristiques se concentrent sur le domaine temporel pour capturer et caractériser l'information présente dans les séries temporelles fournies par les capteurs du système, par exemple via l'utilisation de réseaux de neurones [71]. Il est également d'usage d'avoir recours à des outils de traitement du signal, pour exploiter les caractéristiques du domaine fréquentiel des séries acquises : [26, 48] utilisent respectivement des transformées de Fourier et de Laplace. Enfin, d'autres approches travaillent dans le domaine temps-fréquence, par exemple via l'utilisation de transformée en ondelettes [73]. Ce choix de module d'extraction de caractéristiques est fortement influencé par l'application et le type de données d'entrée, et donc par la connaissance a priori du concepteur.

Le module de diagnostic est ensuite composé au choix :

- soit d'un premier sous-module de détection permettant de réaliser la surveillance, suivi par un second sous-module de classification effectuant ensuite la tâche d'isolation et identification de fautes.
- soit d'un unique modèle de classification assurant simultanément la détection et l'isolation et identification de fautes.

Dans le cas où le jeu de données est étiqueté, le module unique de classification recevant ces caractéristiques est libre d'utiliser le modèle de ML de son choix : SVM [21], forêt aléatoire [62], réseau de neurones peu profond [18], réseau de neurones récurrents [61], etc. Cette approche de détection et classification simultanées est cependant parfois critiquée [37] car pouvant mener à des problèmes pratiques :

- Les occurrences des fautes pouvant conduire à des défaillances et des événements redoutés sont souvent assez rares dans les jeux de données réelles. Cela amène un problème de déséquilibre entre les classes, amplifié si on considère une classification multi-classes.
- Pour ce type de tâche, une erreur de classification du modèle aura le même poids durant la phase d'apprentissage, quelle que soit cette mauvaise classification. Cependant selon la criticité du système, la performance de la détection d'une faute peut être bien plus importante que sa classification.

Pour pallier ces difficultés, une première tâche de surveillance peut être effectuée via des méthodes de détection d'anomalies [10]. De façon semblable au schéma utilisé dans les travaux issus de la communauté de la MSP, ces méthodes semi-supervisées modélisent le comportement normal du système dans la phase d'apprentissage, et classifient comme faute les points effectuant une déviation significative de ce modèle lors de la phase d'inférence. Ces modèles sont plus robustes au déséquilibre présent dans les jeux de données, et pourront être suivis par un modèle de classification pour réaliser l'isolation et identification de la faute.

Enfin, si les conditions de fonctionnement normal du système ne sont pas connues, il est également possible de concevoir le modèle de diagnostic en utilisant des approches de clustering. C'est ce que font Diaz-Rozo et al. dans [9], comparant les performances des algorithmes de mélanges de lois gaussiennes, de clustering hiérarchique agglomératif, et K-means.

Deep Learning. De la même façon que les méthodes basées modèle, les approches de ML classiques se retrouvent aujourd'hui assez limitées face aux données plus complexes de l'industrie 4.0. Ainsi, comme décrit par [72, 23, 35], les méthodes d'extraction de caractéristiques basées sur une connaissance des données acquises peuvent ne plus suffire à effectuer un diagnostic correct. Pour répondre à ces challenges, des modèles de Deep Learning (DL) sont ainsi utilisés, intégrant une phase d'apprentissage de représentation des données dans les premières couches, afin d'extraire automatiquement les caractéristiques les plus saillantes pour une tâche subsidiaire [7, 22], ici le diagnostic. Ainsi, de nombreux travaux ont montré la supériorité des modèles de DL pour le DF, utilisant aussi bien comme algorithme d'apprentissage de représentation des modèles discriminatifs (réseaux convolutifs [59, 56, 34], réseaux récurrents profonds [1, 12], Transformers [58], etc.) que des modèles génératifs (modèles probabilistes graphiques [65, 24], Autoencoders [20, 46, 39], GANs [60, 25]).

Diagnostic à partir de données multimodales. La complexité des données acquises s'intensifie encore de nos jours, avec des capteurs mettant à disposition des données multimodales. Si certains travaux s'attaquent au DF à partir d'images thermiques [8, 19, 47], de rayons X [36], de photographies [55, 54], ou de rapports textuels de maintenance [52, 42], l'application à des données multimodales (de natures hétérogènes) en est à son balbutiement. La majorité des travaux relatifs à la tâche de DF et mentionnant des données "multimodales" fait en fait référence à des modes de fonctionnement différents de l'appareil (comme un climatiseur fonctionnant en mode économique) [43]. Pour Zhou et al. [74], le terme "multimodal" fait référence aux différentes dérivées de leurs séries numériques. A notre connaissance, seuls deux travaux considèrent des données multimodales (au sens "hétérogènes") dans une optique de maintenance. Mian et al. [29] fusionnent des données numériques de signaux vibratoires avec des images thermiques du système afin d'améliorer les performances de classification. Ils utilisent une approche de ML classique, réalisant l'extraction de caractéristiques grâce à une transformée de Hilbert, et utilisant la concaténation comme technique de fusion. Malheureusement, leur jeu de données n'est pas mis à disposition de la communauté, empêchant de se comparer à leur approche. Yang et al. [63] appliquent un modèle multimodal à une tâche connexe de la notre : le pronostic de défaillances. Leur objectif est ainsi de prévoir le temps restant avant l'occurrence d'une défaillance du système. En ce sens, la tâche finale est une régression, mais leur cadre d'étude peut se transposer à celui que nous considérons. Leur approche traite trois modalités (données numériques de capteurs, images et textes) sous la forme de trois

branches distinctes, apprenant une représentation propre à chaque modalité (à l'aide de couches convolutives pour les images et le texte, et d'une couche linéaire pour la modalité numérique). Ils adoptent une approche de fusion tardive, par concaténation de chaque sortie de branche, avant d'appliquer une dernière couche de régression. Cet article est le travail s'appuyant sur un jeu de données public le plus proche de notre problème considéré. Cependant, ce jeu de données comporte quelques points négatifs pour notre cadre. Premièrement, les images considérées ne sont en réalité que des graphiques correspondant aux courbes acquises dans la modalité numérique. Elles ne représentent donc pas réellement des images issues d'une prise de vue du système, qui ont une structure bien différente à l'échelle locale, et n'apportent de surcroît pas d'information supplémentaire sur l'état du système. Deuxièmement, le jeu de données est simulé. Cela implique un manque de richesse et de diversité pour la modalité textuelle. On retrouve beaucoup de fois les mêmes phrases au mot près dans les exemples et on perd ainsi une partie de la nature non-structurée du texte brut.
Les mécanismes de fusion de ces deux seules contributions existantes sur cette application sont relativement simples (concaténation). Nous passons en revue dans la section suivante les enjeux et avancées de l'apprentissage multimodal, afin de tirer parti des meilleures architectures existantes pour notre problème.
Par ailleurs, devant l'absence de jeu public de données multimodales et réelles dans les communautés liées aux systèmes industriels, nous nous sommes tournés vers des jeux de données issus d'autres domaines (voir section 5). Par conséquent, nous invitons la communauté industrielle à mettre à disposition un jeu de données représentatif de ce problème afin d'encourager à la réalisation de futurs travaux sur cette tâche à forts enjeux.

2.2 Apprentissage multimodal

L'accès à différentes sources d'observation d'un même phénomène nous donne de l'information complémentaire et/ou supplémentaire (parmi d'autres types de relation entre modalités [11]). Ce gain d'information est en général bénéfique pour les performances du modèle utilisé pour une tâche considérée [16]. Cependant, cette hétérogénéité de nature entre les différentes sources de données se traduit par des espaces de définition et des propriétés hétérogènes : données structurées et continues (séries temporelles de grandeurs physiques) ou non structurées, discrètes et parcimonieuses (one-hot encodings de texte libre) par exemple. Ainsi, un même concept aura des représentations vectorielles également très différentes dans chaque espace propre à une modalité, ce qui implique une difficulté à mesurer une similarité entre des points de modalités différentes. Cette difficulté est définie sous le nom de fossé d'hétérogénéité [13]. De ce verrou scientifique découlent plusieurs enjeux comme la transduction ou l'alignement entre modalités, tous décrits dans différentes revues [13, 6]. Le challenge nous intéressant ici est celui de la fusion entre modalités, souvent lié à l'apprentissage de représentation jointe multimodale. Cette tâche a pour but de projeter des représentations unimodales dans un sous-espace sémantique joint, afin de combler ce fossé d'hétérogénéité.
Pour ce faire, les premières approches ont adopté des stratégies de fusion précoce en concaténant [30] ou multipliant [69] les caractéristiques de chacune des modalités ; ou de fusion tardive, combinant les décisions de modèles unimodaux par système de vote [53].
A l'opposé de ces méthodes agnostiques à un type de modèle, certaines architectures de DL modélisent les interactions inter-modalités et intra-modalité afin d'apprendre les représentations jointes les plus pertinentes. Des approches génératives utilisent par exemple des variantes multimodales de machines de Boltzmann [44], ou des réseaux de croyance profonds [45], apprenant une distribution jointe sur les deux modalités d'entrée. Ces architectures entraînables de façon non supervisée peuvent ainsi générer des modalités à partir d'une autre, et sont donc plus robustes aux modalités manquantes. Cependant, le coût élevé d'approximation d'inférence des algorithmes est souvent rédhibitoire à leurs usages.
L'autre grande famille de modèles génératifs utilisés est celle des autoencoders. Ils visent à apprendre une représentation condensée, qui capture les éléments essentiels à la reconstruction de l'objet initial. Des adaptations multimodales ont été développées [31], dans lesquelles la couche intermédiaire commune prend en entrée les deux modalités, et essaye de les reconstruire à partir de ce vecteur intermédiaire commun. Cependant, ces architectures ne se basant que sur la reconstruction des données d'entrée, les représentations apprises sont agnostiques à une tâche précise et donc génériques. Cela peut impliquer une baisse de performances si l'apprentissage n'est pas guidé par des contraintes supplémentaires [41].
Par ailleurs, des modèles discriminatifs utilisant des mécanismes d'attention [5] ont également été utilisés à la fois dans un but d'amélioration de performances mais aussi de gain d'interprétabilité. En effet, à une échelle intra-modalité, le mécanisme d'attention est utilisé pour sélectionner les composantes les plus pertinentes de chaque modalité, dépendant du contexte donné par les autres modalités, comme sur une tâche de Visual Question Answering [64]. A une échelle inter-modalités, ce type de mécanismes permet de pondérer la contribution de chacune des modalités dans la prise de décision finale [27].
Plus récemment, ce mécanisme d'attention cross-modal a été étendu aux architectures de Transformers, avec pour but l'apprentissage de représentations contextuelles [70, 49, 2]. Ces approches ont également l'avantage de pouvoir gérer des modalités non alignées, comme le mettent en avant Tsai et al. [49]. Effectivement, les données que nous considérons dans le cadre du DF ne sont pas alignées temporellement. Par exemple, une faute apparaissant à un temps spécifique t pourra être corrélée à la fois à des points récents de données de capteurs (de l'ordre des secondes précédentes), ainsi qu'à des points de données bien plus antérieurs pour la modalité textuelle (rapport de maintenance de la semaine précédente). Les approches multimodales classiques utilisées pour des données séquentielles (basées sur des ré-

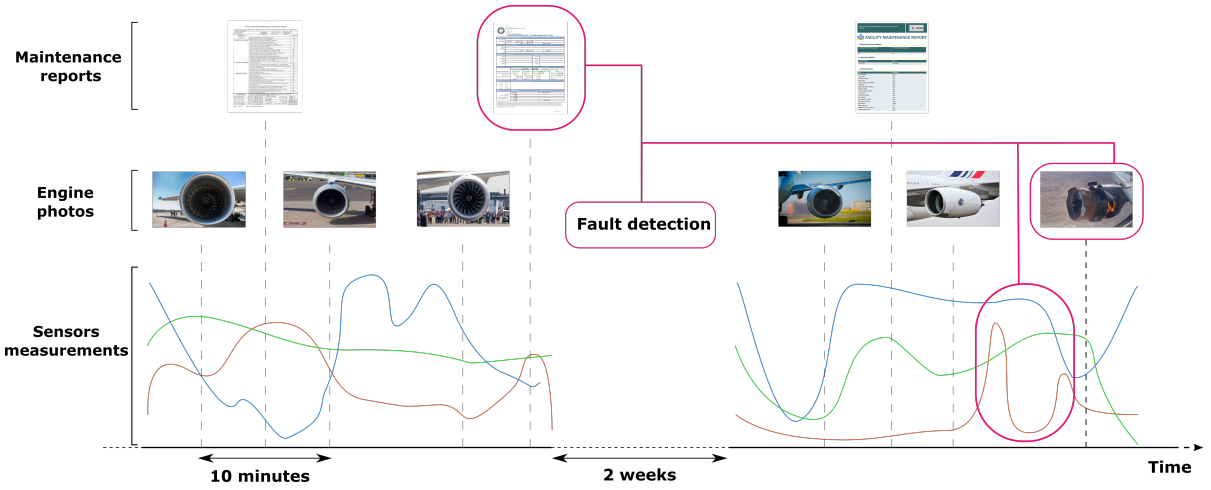

FIGURE 1 – Exemple d'un système industriel produisant des données hétérogènes, non alignées, arbitrairement longues, dans le cadre d'une tâche de diagnostic de faute.

seaux récurrents par exemple), ne gèrent pas ce problème de non-alignement [67, 68]. L'approche utilisée par Tsai et al. [49] s'attaque à ce challenge grâce à son module de représentation cross-modale, et plus précisément au produit matriciel requêtes-clés, modélisant toutes les corrélations entre deux séquences de modalités différentes. L'architecture StreaMulT que nous proposons en section 4 s'inspire de ce modèle. Elle présente l'avantage de s'adapter au cadre de données multimodales arbitrairement longues, que nous introduisons dans la prochaine section.

3 Cadre théorique

Dans cette section, nous définissons le problème auquel s'attaque notre méthode. Nous nous plaçons dans le contexte applicatif du DF et considérons des flux de données hétérogènes à la fois par leur nature (séries numériques, texte brut, images, son, etc.) et par leurs fréquences d'acquisition. Nous supposons que ces différents flux sont a priori non alignés et que l'historique des données peut être arbitrairement long. Enfin, nous considérons le cas où un système industriel peut ne jamais s'arrêter de fonctionner et nécessite donc que les séquences d'entrée soient traitées au fil de l'eau (en "streaming"). Cet exemple est illustré dans la Fig. 1.

Pour des besoins de clarté, et sans perte de généralité, nous considérons trois modalités notées α, β, γ. Soient, trois séries temporelles $(X_\alpha, X_\beta, X_\gamma)$ de différentes modalités. Chaque série temporelle est indicée par le temps, possède ses propres temps d'acquisition et son propre espace de définition. Ainsi, pour la modalité α,

$$X_\alpha := (X_\alpha(t))_{t\in\mathcal{T}_\alpha} \text{ et } \forall t \in \mathcal{T}_\alpha,\ \ X_\alpha(t) \in \mathbb{R}^{d_\alpha}$$

où $\mathcal{T}_\alpha$ et d_α sont respectivement les ensembles dénombrables contenant les temps d'acquisition de la modalité α et sa dimension de caractéristiques associée.

Notre objectif est de réaliser une tâche de prédiction au cours du temps. Soit $\mathcal{X}$ l'ensemble d'entrée défini par : $\mathcal{X} := \left\{ [X(s)]_{s\leq t}, t \in \mathbb{R} \right\}$, où $[X(s)]_{s\leq t} = \bigcup_{j\in\{\alpha,\beta,\gamma\}} \{X_j(s), s \leq t\}$ sont les données de toutes modalités acquises avant le pas de temps t. Formellement, étant donné un espace de labels $\mathcal{Y}$ commun à toutes les modalités, notre but est de trouver la fonction de prédiction optimale $h^* : \mathcal{X} \mapsto \mathcal{Y}$ minimisant une fonction de perte L sur un espace d'hypothèse $\mathcal{H}$:

$$h^* = \arg\min_{h\in\mathcal{H}} L(h)$$

avec $\quad L(h) := \frac{1}{|\mathcal{T}_y|} \sum_{t\in\mathcal{T}_y} l\left(h\left([X(s)]_{s\leq t}\right), y_t\right)$

où l est une fonction de score mesurant l'erreur entre la prédiction de h au temps t et la vérité terrain y_t, et où $\mathcal{T}_y$ représente l'ensemble dénombrable des temps d'acquisition des labels, dont la définition dépend de la tâche. Par exemple, pour une tâche de DF, $\mathcal{T}_y := \mathcal{T}_\alpha \cup \mathcal{T}_\beta \cup \mathcal{T}_\gamma$ car l'objectif est de détecter et classifier une faute à toute nouvelle acquisition de données.

A notre connaissance, ce cadre de données multimodales, possédant des temps d'acquisition différents, potentiellement non alignées et arbitrairement longues n'a jamais été introduit et traité auparavant.

4 Modèle proposé

Nous proposons StreaMulT (Streaming Multimodal Transformer), un modèle prenant avantage des architectures de Multimodal Transformer [49] et Emformer [40]. La longueur arbitraire des séquences d'entrée est contrôlée par un mécanisme de traitement par blocs (voir Fig. 2), et la multimodalité est gérée par des modules de Transformers cross-modaux fonctionnant en streaming (voir Fig. 3).

Transformer cross-modal. Le module d'attention cross-modale, défini dans [49], traite le fossé d'hétérogénéité des données d'entrée [14] en exprimant une modalité cible α avec les caractéristiques brutes d'une modalité source β. Formellement, en considérant nos séquences d'entrée X_α et X_β, l'attention cross-modale exprimant X_α à partir de

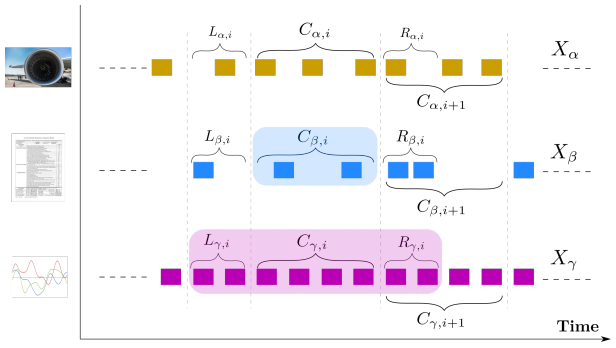

FIGURE 2 – Mécanisme de traitement par blocs pour l'apprentissage multimodal. Pour la modalité $\alpha : X_\alpha, C_{\alpha,i}, L_{\alpha,i}$ et $R_{\alpha,i}$ correspondent respectivement à la séquence d'entrée entière, au i-ème segment central initial et aux contextes gauche et droit associé à ce segment central, afin de former le i-ème segment contextuel. La zone bleue représente un segment central pour la modalité β et la zone rose représente un segment contextuel pour la modalité γ.

X_β , notée $X_{\beta\to\alpha}$ se calcule comme suit :

$$X_{\beta\to\alpha} := Attn(Q_\alpha, K_\beta, V_\beta) = \text{softmax}\left(\frac{Q_\alpha K_\beta^T}{\sqrt{d_k}}\right) V_\beta$$

$$= \text{softmax}\left(\frac{X_\alpha W_{Q_\alpha} W_{K_\beta}^T X_\beta^T}{\sqrt{d_k}}\right) X_\beta W_{V_\beta}$$

avec Q_α la matrice de requêtes de la modalité α, K_β, V_β les matrices de clés et valeurs de la modalité β et $W_{Q_\alpha}, W_{K_\beta}, W_{V_\beta}$ des poids appris. Cette "scaled dot-product attention", inspirée par le mécanisme de self-attention du Transformer original [50], modélise les dépendances à long terme à travers son produit matriciel et gère ainsi des données non alignées de la même façon [49].

Mécanisme de traitement par blocs. Cependant, la longueur arbitraire des séquences d'entrée de notre cadre implique deux verrous majeurs. Premièrement, l'entraînement du modèle est insoluble en raison de la complexité quadratique de l'architecture Transformer, et deuxièmement l'inférence ne peut s'effectuer au fil de l'eau, l'architecture Transformer ayant besoin de la séquence d'entrée complète pour effectuer le produit matriciel. Pour résoudre ces problèmes, nous adoptons un mécanisme de traitement par blocs, découpant les séquences d'entrée en plus petits segments disjoints $(C_i)_{i\geq 0}$ (voir Fig. 2). Nous calculons ensuite l'attention sur ces segments et réduisons ainsi la complexité du modèle durant le calcul de l'attention crossmodale. Pour éviter les effets de bords, nous ajoutons à ces segments disjoints des blocs de contextes gauche et droit, concaténés aux segments initiaux afin de former des segments contextuels $X_i = [L_i : C_i : R_i]$. Enfin, pour véhiculer l'information entre les segments, nous utilisons une banque de mémoire, à la manière d'Emformer [40].

Architecture globale. Notre architecture globale *end-to-end* combine ainsi les avantages des deux architectures précédentes [49, 40] et est illustrée dans la Fig. 4.

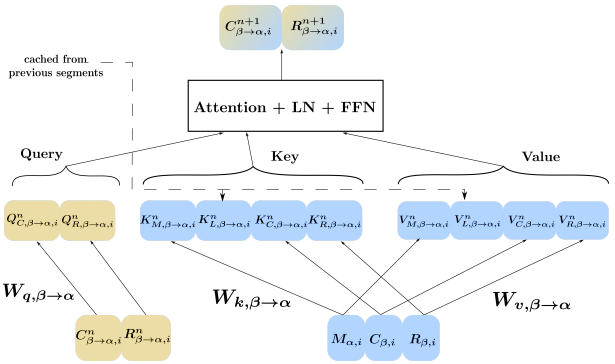

FIGURE 3 – Module de Streaming Crossmodal Transformer

Nous décrivons ici le traitement de la modalité α.
X_α passe d'abord par une couche convolutive 1D afin de modéliser une première structure locale temporelle, et de projeter l'ensemble des modalités dans un espace commun de dimension d. Les bornes des segments sont ensuite délimitées, et en suivant l'approche de traitement par blocs, tous les segments contextuels $X_{\alpha,i}$ sont traités en parallèle. Ils passent premièrement à travers un module Emformer unimodal afin d'initialiser la banque de mémoire propre à cette modalité. Ensuite, chaque paire de modalités source/cible (β / α) est traitée par son propre module Streaming Crossmodal Transformer (SCT), dont le fonctionnement est illustré dans la Fig. 3. Plus spécifiquement, chaque segment de la modalité cible $X_{\alpha,i}$ est exprimé en utilisant le segment temporel correspondant de la modalité source $X_{\beta,i}$ ainsi que la banque de mémoire de cette même modalité source $M_{\beta,i}$, contenant de l'information compressée des segments précédents.
Ainsi, pour chaque couche n du module SCT $\beta \to \alpha$:

$$\left[\hat{C}^n_{\alpha,i}, \hat{R}^n_{\alpha,i}\right] = \text{LN}(\left[C^n_{\alpha,i}, R^n_{\alpha,i}\right])$$
$$\left[\hat{C}^n_{\beta,i}, \hat{R}^n_{\beta,i}\right] = \text{LN}(\left[C^n_{\beta,i}, R^n_{\beta,i}\right])$$
$$K^n_{\beta,i} = \left[K^n_{M,\beta\to\alpha,i}, K^n_{L,\beta\to\alpha,i}, K^n_{C,\beta\to\alpha,i}, K^n_{R,\beta\to\alpha,i}\right]$$
$$V^n_{\beta,i} = \left[V^n_{M,\beta\to\alpha,i}, V^n_{L,\beta\to\alpha,i}, V^n_{C,\beta\to\alpha,i}, V^n_{R,\beta\to\alpha,i}\right]$$
$$Z^n_{C,\beta\to\alpha,i} = Attn(Q^n_{C,\beta\to\alpha,i}, K^n_{\beta,i}, V^n_{\beta,i}) + C^n_{\beta\to\alpha,i}$$
$$Z^n_{R,\beta\to\alpha,i} = Attn(Q^n_{R,\beta\to\alpha,i}, K^n_{\beta,i}, V^n_{\beta,i}) + R^n_{\beta\to\alpha,i}$$
$$\left[\hat{C}^{n+1}_{\alpha,i}, \hat{R}^{n+1}_{\alpha,i}\right] = \text{FFN}(\text{LN}([Z^n_{C,\beta\to\alpha,i}, Z^n_{R,\beta\to\alpha,i}]))$$
$$\left[C^{n+1}_{\alpha,i}, R^{n+1}_{\alpha,i}\right] = \text{LN}(\left[\hat{C}^{n+1}_{\alpha,i}, \hat{R}^{n+1}_{\alpha,i}\right] + [Z^n_{C,\beta\to\alpha,i}, Z^n_{R,\beta\to\alpha,i}])$$

où,

$$\left[K^n_{M,\beta\to\alpha,i}, K^n_{C,\beta\to\alpha,i}, K^n_{R,\beta\to\alpha,i}\right] = W_{k,\beta\to\alpha}\left[M_{\beta,i}, \hat{C}^n_{\beta,i}, \hat{R}^n_{\beta,i}\right]$$
$$\left[V^n_{M,\beta\to\alpha,i}, V^n_{C,\beta\to\alpha,i}, V^n_{R,\beta\to\alpha,i}\right] = W_{v,\beta\to\alpha}\left[M_{\beta,i}, \hat{C}^n_{\beta,i}, \hat{R}^n_{\beta,i}\right]$$
$$\left[Q^n_{C,\beta\to\alpha,i}, Q^n_{R,\beta\to\alpha,i}\right] = W_{q,\beta\to\alpha}\left[C^n_{\beta\to\alpha,i}, R^n_{\beta\to\alpha,i}\right]$$

et $\left(K^n_{L,\beta\to\alpha,i}, V^n_{L,\beta\to\alpha,i}\right)$ sont les copies des clés et valeurs (mises en cache) correspondant aux segments précédents, dont la taille est fixée par la taille du contexte gauche. LN, FFN, Attn correspondent respectivement à des

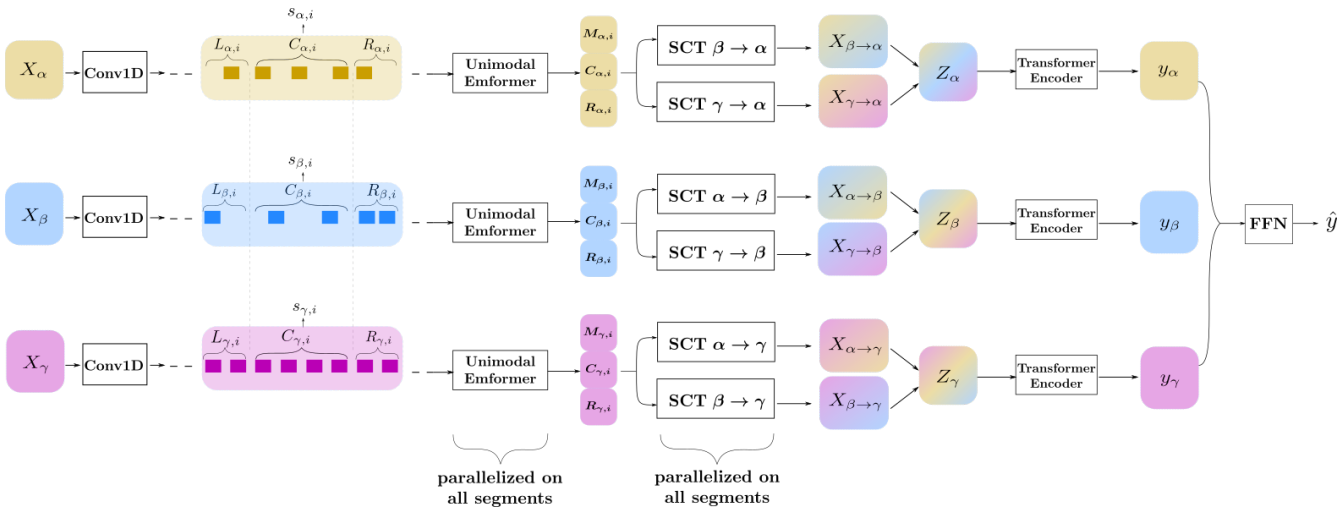

FIGURE 4 – Architecture globale du Streaming Multimodal Transformer. SCT signifie Streaming Crossmodal Transformer. Les couleurs différentes représentent les natures hétérogènes des différentes modalités, et les dégradés expriment les caractéristiques cross-modales.

couches "Layer Normalization", "Feed-Forward" et "scaled dot-product Attention".

Après la dernière couche N, les représentations des contextes droits $(R^N_{\beta\to\alpha,i})_{i>0}$ sont défaussées. Les segments $(C^N_{\beta\to\alpha,i})_{i>0}$ sont concaténés pour former la représentation cross-modale finale $X_{\beta\to\alpha}$. Les représentations cross-modales correspondant à la même modalité cible α sont alors concaténées selon la dimension des caractéristiques dans un vecteur $Z_\alpha := \begin{pmatrix} X_{\beta\to\alpha} \\ X_{\gamma\to\alpha} \end{pmatrix}$, qui est à son tour traité par un encoder de Transformer classique afin d'exploiter la nature séquentielle des données. La sortie y_α de cette couche est finalement concaténée avec celles des autres modalités, et une couche linéaire résulte en la prédiction $\hat{y}_t$.

5 Expériences et résultats

Comme abordé en section 2, à notre connaissance il n'existe pas de jeu de données public représentatif de notre cadre. Nous décidons donc de conduire nos expériences sur le jeu de données CMU-MOSEI [4], afin d'évaluer empiriquement notre architecture StreaMulT et la comparer avec des approches existantes sur un cadre proche de celui recherché, à savoir des données séquentielles réelles, multimodales et non alignées. Le jeu de données CMU-MOSEI est composé de 23,454 clips vidéos témoignant de l'opinion de plus de 1000 orateurs sur plus de 250 sujets divers. De ces clips sont extraits des caractéristiques audio, vidéo et textuelles, utilisées pour créer une version brute non-alignée du jeu de données ainsi qu'une version réalignée entre les trois modalités. La longueur des phrases alignées est fixée à 50 tokens, utilisant un éventuel padding.

La tâche associée à ce jeu de données est une analyse de sentiments sur ces clips vidéos, étiquetés par des annotateurs humains avec un score de sentiment allant de -3 (sentiment très négatif) à 3 (sentiment très positif). Comme dans les travaux précédents [49], nous évaluons les performances de notre modèle suivant 5 métriques : l'accuracy d'une classification sur 7 classes, l'accuracy binaire (sentiment positif ou négatif), le score F1, l'erreur moyenne absolue et la corrélation entre les prédictions du modèle et les étiquettes.

Pour souligner la valeur ajoutée de StreaMulT, nous conduisons nos expériences dans deux cadres différents. (1) Nous considérons d'abord les clips vidéo comme nos séquences d'entrée complètes, et observons les performances de StreaMulT lorsque nous divisons ces clips en segments plus courts. Pour pouvoir définir les mêmes bornes de segments entre les modalités, nous réalisons ces expériences sur la version alignée de CMU-MOSEI. Nous décidons de diviser chaque séquence en 5 segments de 10 pas de temps. (2) Nous concaténons ensuite tous les clips vidéos d'un même orateur et considérons cette suite de clips comme notre séquence d'entrée longue, afin de construire artificiellement des séries arbitrairement longues. Dans cette configuration, nous choisissons comme segments les clips initiaux et pouvons donc utiliser la version non-alignée.

StreaMulT n'a pas pour objectif de battre les architectures les plus performantes sur la tâche d'analyse de sentiments multimodal [66, 15], sa valeur ajoutée étant sa capacité à gérer des séquences multimodales non alignées **arbitrairement longues**. Ainsi nous ne reportons ici que les métriques concernant le Transformer multimodal données dans [49], pour une comparaison équitable avec une architecture similaire. Nous avons également utilisé le code officiel de cette approche mis à disposition [†], en gardant les valeurs d'hyperparamètres données dans [49]. Nous ne sommes cependant pas parvenus à reproduire exactement les résultats communiqués dans l'article, et nous présentons donc ceux que nous avons obtenus. Toutes les métriques sont moyennées sur 5 trajectoires d'entraînement.

†. https ://github.com/yaohungt/Multimodal-Transformer

Métrique	Acc_7^h	Acc_2^h	F1^h	MAE^l	Corr^h
MulT [49]	**51.8**	**82.5**	**82.3**	**0.580**	**0.703**
MulT‡(1)	49.32	81.05	81.42*	0.615	0.666
StreaMulT‡ (1)	50, 08*	81.08*	81.01	0.608*	0.671*
MulT‡(2)	-	-	-	-	-
StreaMulT‡ (2)	49.25	80.55	80.84	0.621	0.665

TABLE 1 – Résultats sur CMU-MOSEI. Les meilleurs résultats sont en gras. ‡ : notre implémentation ou reproduction depuis le code officiel, avec les valeurs d'hyperparamètres fournies. * : meilleur score parmi la catégorie ‡. (1) et (2) font références aux deux environnements d'expérimentation définis plus haut.

Le tableau 1 montre que notre architecture reproduit les résultats du Transformer Multimodal dans l'environnement (1) (fait même un peu mieux sur 4 des 5 métriques), ce qui démontre la capacité de la banque de mémoire à véhiculer l'information pertinente à la classification entre les différents segments de taille 10, tandis que MulT a accès à la séquence de taille 50 dans son intégralité. Pour l'environnement (2), les performances baissent légèrement. Cependant, ce cadre d'évaluation artificiel permet de mettre en évidence la plus-value de notre architecture qui est sa capacité à traiter au fil de l'eau des données arbitrairement longues. A l'inverse, on observe que MulT rencontre une erreur de mémoire et n'est donc pas capable de gérer ces longues séquences en raison de sa complexité.

Pour valider qualitativement notre modèle, nous affichons la carte de chaleur des différents poids d'attention du modèle dans la Fig. 5. Cette carte représente les différents poids d'attention du module SCT associé aux modalités images (requêtes) / texte (clés), pour une séquence d'entrée de taille 50.

Cette carte de chaleur nous rappelle premièrement que les séquences de langage sont non alignées entre les modalités : à l'inverse d'une diagonale monotone, nous observons différentes activations sur des lignes verticales, correspondant à certains embeddings textuels corrélés à plusieurs images. Si certains non-alignements restent dans le champ d'un même segment, comme représenté dans le quatrième segment par le rectangle vert, l'accès à la banque de mémoire permet au modèle d'accéder à des données à plus longue portée, comme illustré dans le troisième segment par les deux rectangles jaunes. Celui de droite indique des dépendances non alignées au sein du troisième segment, tandis que celui de gauche met en lumière l'activation de certaines images de ce segment par des caractéristiques textuelles venant du passé, sauvegardées dans la banque de mémoire.

Ces comportements différents témoignent de la capacité de l'architecture StreaMulT à adapter sa stratégie selon le contexte, accédant à des données non-alignées du passé via la banque de mémoire lorsque nécessaire.

6 Conclusion

Notre état des lieux des méthodes utilisées pour le diagnostic de fautes dresse leurs limites pour répondre aux défis posés par l'industrie 4.0 : gérer des séquences hétérogènes, non alignées et arbitrairement longues. Notre architecture StreaMulT combine l'attention cross-modale et le mécanisme de traitement par blocs parallélisé afin de traiter ces séquences multimodales en streaming. Les expériences conduites sur le jeu de données CMU-MOSEI ont montré des résultats prometteurs : une conservation des performances couplée à une capacité à gérer des séquences arbitrairement longues durant la phase d'entraînement, et à traiter ces flux de données en streaming à l'inférence.

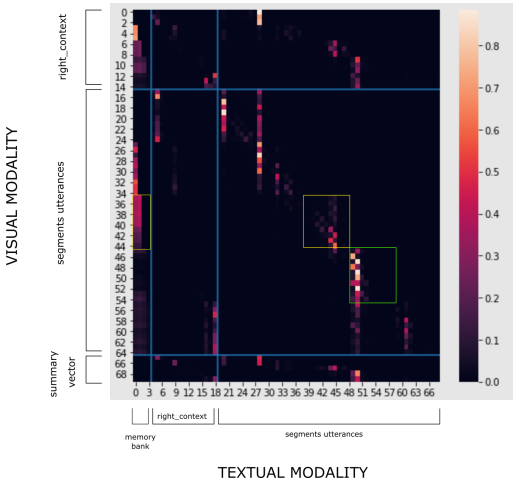

FIGURE 5 – Carte de chaleur des poids d'attention de StreaMulT pour le module cross-modal images (requêtes) / texte (clés). La séquence de taille 50 est découpée en segments de taille 10, avec des contextes gauche et droit de tailles respectives 10 et 3. Les lignes bleues délimitent les interactions entre les différents composants des séquences (contextes, segments, mémoire).

Remerciements

Victor Pellegrain est financé par l'IRT SystemX en collaboration avec CentraleSupélec. Ce travail a été réalisé grâce aux ressources du centre de calcul Mésocentre de CentraleSupélec et de l'ENS Paris-Saclay, soutenu par le CNRS et la Région Ile-de-France.

Références

[1] Abed, W. : A robust bearing fault detection and diagnosis technique for brushless dc motors under non-stationary operating conditions. JCAES (2015)

[2] Akbari, H., et al. : VATT : transformers for multimodal self-supervised learning from raw video, audio and text. CoRR abs/2104.11178 (2021)

[3] Angelopoulos, A., et al. : Tackling faults in the industry 4.0 era—a survey of machine-learning solutions and key aspects. Sensors 20(1), 109 (2019)

[4] Bagher Zadeh, A., et al. : Multimodal language analysis in the wild : CMU-MOSEI dataset and interpretable dynamic fusion graph. In : ACL 2018. pp. 2236–2246

[5] Bahdanau, D., et al. : Neural machine translation by jointly learning to align and translate. In : ICLR 2015

[6] Baltrusaitis, T., et al. : Multimodal Machine Learning : A Survey and Taxonomy. IEEE TPAMI 41(2), 423–443 (2019)

[7] Bengio, Y., et al. : Representation learning : A review and new perspectives. IEEE TPAMI 35, 1798–1828 (2013)

[8] Choudhary, A., et al. : Bearing fault diagnosis of induction motor using thermal imaging. pp. 950–955 (2018)

[9] Diaz Rozo, J., et al. : Machine learning-based cps for clustering high throughput machining cycle conditions. Procedia Manufacturing 10, 997–1008 (2017)

[10] Goldstein, M., Uchida, S. : A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. PloS one 11, e0152173 (2016)

[11] Grifoni, P. : Multimodal human computer interaction and pervasive services (2009)

[12] Guo, L., et al. : A recurrent neural network based health indicator for remaining useful life prediction of bearings. Neurocomputing 240 (2017)

[13] Guo, W., et al. : Deep Multimodal Representation Learning : A Survey. IEEE Access 7, 63373–63394 (2019)

[14] Guo, W., et al. : Deep multimodal representation learning : A survey. IEEE Access 7, 63373–63394 (2019)

[15] Han, W., et al. : Improving multimodal fusion with hierarchical mutual information maximization for multimodal sentiment analysis. In : EMNLP 2021. pp. 9180–9192. ACL (2021)

[16] Huang, Y., et al. : What makes multimodal learning better than single (provably). ArXiv abs/2106.04538 (2021)

[17] Isermann, R. : Fault-Diagnosis Systems From Fault Detection to Fault Tolerance, vol. 28 (2006)

[18] Jafar, R., et al. : Application of artificial neural networks (ann) to model the failure of urban water mains. Mathematical and Computer Modelling 51, 1170–1180 (2010)

[19] Janssens, O., et al. : Thermal image based fault diagnosis for rotating machinery. Infrared Physics Technology 73, 78–87 (2015)

[20] Jia, F., et al. : Deep neural networks : A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data. Mechanical Systems and Signal Processing 72-73 (2015)

[21] Konar, P., et al. : Bearing fault detection of induction motor using wavelet and neural networks. pp. 798–809 (2009)

[22] LeCun, Y., et al. : Deep learning. Nature 521, 436–44 (2015)

[23] Li, Z. : Deep learning driven approaches for predictive maintenance : A framework of intelligent fault diagnosis and prognosis in the industry 4.0 era (2018)

[24] Liang, T., et al. : Bearing fault diagnosis based on improved ensemble learning and deep belief network. Journal of Physics 1074, 012154 (2018)

[25] Liu, H., et al. : Unsupervised fault diagnosis of rolling bearings using a deep neural network based on generative adversarial networks. Neurocomputing 315 (2018)

[26] Liu, Y., et al. : Application to induction motor faults diagnosis of the amplitude recovery method combined with fft. Mechanical Systems and Signal Processing 24, 2961–2971 (2010)

[27] Long, X., et al. : Multimodal keyless attention fusion for video classification. AAAI 2018 pp. 7202–7209

[28] Luo, B., et al. : Early fault detection of machine tools based on deep learning and dynamic identification. IEEE TIE 66(1), 509–518 (2019)

[29] Mian, T., et al. : A sensor fusion based approach for bearing fault diagnosis of rotating machine. Journal of Risk and Reliability 0(0), 1748006X211044843 (0)

[30] Morency, L.P., et al. : Towards multimodal sentiment analysis : Harvesting opinions from the web. In : ICMI 2011. p. 169–176. ACM

[31] Ngiam, J., et al. : Multimodal Deep Learning. ICML 3(3), 194–203 (2011)

[32] Nor, N., et al. : A review of data-driven fault detection and diagnosis methods : Applications in chemical process systems. Reviews in Chemical Engineering 36 (2019)

[33] Palade, V., et al. : Computational Intelligence in Fault Diagnosis (2006)

[34] Pan, J., et al. : Liftingnet : A novel deep learning network with layerwise feature learning from noisy mechanical data for fault classification. IEEE TIE PP, 1–1 (2017)

[35] Peng, Y., et al. : Current status of machine prognostics in condition-based maintenance : A review. International Journal of Advanced Manufacturing Technology 50, 297–313 (2010)

[36] Reid, A., et al. : Fault location and diagnosis in a medium voltage epr power cable. IEEE TDEI 20, 10 – 18 (2013)

[37] Reis, M.S., Gins, G. : Industrial process monitoring in the big data/industry 4.0 era : from detection, to diagnosis, to prognosis. Processes 5(3) (2017)

[38] Rogers, A., et al. : A review of fault detection and diagnosis methods for residential air conditioning systems. Building and Environment 161, 106236 (2019)

[39] Shao, H., et al. : A novel method for intelligent fault diagnosis of rolling bearings using ensemble deep auto-encoders. MSSP 102, 278–297 (2018)

[40] Shi, Y., et al. : Emformer : Efficient memory transformer based acoustic model for low latency streaming speech recognition (2020)

[41] Silberer, C., et al. : Learning grounded meaning representations with autoencoders. ACL 2014 1, 721–732

[42] Sipos, R., et al. : Log-based predictive maintenance. In : ACM SIGKDD 2014. p. 1867–1876

[43] Sipple, J. : Interpretable, multidimensional, multimodal anomaly detection with negative sampling for detection of device failure. In : ICML 2020. vol. 119, pp. 9016–9025. PMLR (2020)

[44] Srivastava, N. : Multimodal Learning with Deep Boltzmann Machines 15, 2949–2980 (2014)

[45] Srivastava, N., et al. : Learning representations for multimodal data with deep belief nets. ICML Workshop (2012)

[46] Sun, J., et al. : Intelligent bearing fault diagnosis method combining compressed data acquisition and deep learning. IEEE TIM PP, 1–11 (2017)

[47] Taheri-Garavand, A., et al. : An intelligent approach for cooling radiator fault diagnosis based on infrared thermal image processing technique. Applied Thermal Engineering 87, 434–443 (2015)

[48] Taneja, G., et al. : Reliability modelling and analysis of a single machine subsystem of a cable plant (2017)

[49] Tsai, Y.H.H., et al. : Multimodal transformer for unaligned multimodal language sequences. ACL 2019 pp. 6558–6569

[50] Vaswani, A., et al. : Attention is all you need. In : NIPS 2017. vol. 30

[51] Venkatasubramanian, V., et al. : A review of process fault detection and diagnosis. part i : Quantitative model-based methods 27(3), 293–311. part ii : Qualitative models and search strategies 27(3), 313–32. part iii : Process history based methods 27(3), 327–346. Computers Chemical Engineering (2003)

[52] Wang, F., et al. : Bilevel feature extraction-based text mining for fault diagnosis of railway systems. IEEE TITS 18(1), 49–58 (2016)

[53] Wang, H., et al. : Select-additive learning : Improving cross-individual generalization in multimodal sentiment analysis (2016)

[54] Wang, J., et al. : Machine vision intelligence for product defect inspection based on deep learning and hough transform. Journal of Manufacturing Systems 51, 52–60 (2019)

[55] Wang, S., et al. : Panoramic crack detection for steel beam based on structured random forests. IEEE Access 6, 16432–16444 (2018)

[56] Wen, L., et al. : A new convolutional neural network based data-driven fault diagnosis method. IEEE TIE PP, 1–1 (2017)

[57] Wen, L., et al. : A new snapshot ensemble convolutional neural network for fault diagnosis. IEEE Access 7, 32037–32047 (2019)

[58] Wu, B., et al. : Simultaneous-fault diagnosis considering time series with a deep learning transformer architecture for air handling units. Energy and Buildings 257, 111608 (2021)

[59] Xia, M., et al. : Fault diagnosis for rotating machinery using multiple sensors and convolutional neural networks. IEEE/ASME Transactions on Mechatronics PP, 1–1 (2017)

[60] Xie, Y., Zhang, T. : Imbalanced learning for fault diagnosis problem of rotating machinery based on generative adversarial networks. pp. 6017–6022 (2018)

[61] Yam, R., et al. : Intelligent predictive decision support system for condition-based maintenance. IJAMT 17, 383–391 (2001)

[62] Yang, B.S., et al. : Random forests classifier for machine fault diagnosis. JMST 22, 1716–1725 (2008)

[63] Yang, Z., et al. : A multi-branch deep neural network model for failure prognostics based on multimodal data. Journal of Manufacturing Systems 59, 42–50 (2021)

[64] Yang, Z., et al. : Stacked attention networks for image question answering. IEEE CVPR 2016(1), 21–29

[65] Yu, K., et al. : A bearing fault and severity diagnostic technique using adaptive deep belief networks and dempster–shafer theory. Structural Health Monitoring (2019)

[66] Yu, W., et al. : Learning modality-specific representations with self-supervised multi-task learning for multimodal sentiment analysis. arXiv (2021)

[67] Zadeh, A., et al. : Memory fusion network for multi-view sequential learning. AAAI 2018 pp. 5634–5641

[68] Zadeh, A., et al. : Multimodal language analysis in the wild : Cmu-mosei dataset and interpretable dynamic fusion graph. ACL 2018 1, 2236–2246 (2018)

[69] Zadeh, A., et al. : Tensor Fusion Network for Multimodal Sentiment Analysis pp. 1103–1114 (2018)

[70] Zadeh, A., et al. : Factorized Multimodal Transformer for Multimodal Sequential Learning pp. 1–13 (2019)

[71] Zarei, J., et al. : Vibration analysis for bearing fault detection and classification using an intelligent filter. Mechatronics 24 (2014)

[72] Zhang, S., et al. : Deep learning algorithms for bearing fault diagnostics—a comprehensive review. IEEE Access 8, 29857–29881 (2020)

[73] Zhang, Z., et al. : Fault diagnosis and prognosis using wavelet packet decomposition, fourier transform and artificial neural network. Journal of Intelligent Manufacturing 24 (2013)

[74] Zhou, F., et al. : A multimodal feature fusion-based deep learning method for online fault diagnosis of rotating machinery. Sensors 18, 3521 (2018)