Atelier GAST

Gestion et Analyse de données Spatiales et Temporelles

Organisateurs :

Aurélie LEBORGNE (ICube, Université de Strasbourg, France)
Loïc SALMON (LABISEN, Yncrea Ouest, France)
Nida MEDDOURI (LRE, EPITA, Kremlin-Bicêtre, Paris, France)

PRÉFACE

Le huitième atelier « Gestion et Analyse des données Spatiales et Temporelles » (GAST) est associé à EGC'2023. Cet atelier, s'appuyant sur le Groupe de Travail GAST, regroupe des chercheurs, du domaine académique et de l'industrie, qui s'intéressent aux problématiques liées à la prise en compte de l'information temporelle ou spatiale (quantitative ou qualitative) dans leurs processus de gestion et d'analyse de données (méthodes et application d'extraction, de gestion, de représentation, d'analyse et de visualisation d'informations).

Ces actes regroupent six soumissions présentées à l'atelier GAST'2023. Ces articles montrent une large étendue des recherches actuelles à des fins de modélisation, d'extraction, d'analyse, ou de visualisation d'information, basées sur les dimensions temporelles et spatiales associées. Nous espérons que les orateurs, les auditeurs et les lecteurs pourront interagir autour de ces sujets, que les questions et les défis associés à l'information temporelle et spatiale continueront à animer les débats.

Nous tenons à remercier tous les auteurs pour leurs propositions d'articles ainsi que les membres du comité de lecture dont les retours ont été de qualité pour l'ensemble des articles.

En espérant que ces articles vous apporteront de nouvelles perspectives autour de la Gestion et l'Analyse des données Spatiales et Temporelles, nous vous souhaitons une bonne lecture.

Aurélie Leborgne	Loïc Salmon	Nida Meddouri
ICube, UNISTRA	LABISEN, Yncrea Ouest	LRE, EPITA

Membres du comité de lecture

Ahmed Samet (ICUBE, INSA Strasbourg, France)
Amedeo Napoli (LORIA Nancy, Université de Lorraine, France)
Antoine Vacavant (Institut Pascal, Université Clermont Auvergne, France)
Aurelie Leborgne (ICube, Université de Strasbourg, France)
Bruno Cremilleux (GREYC, Université de Caen Normandie, France)
Chemesse Ennehar Bencheriet (Université 8 Mai 1945 de Guelma, Algérie)
Cyril de Runz (LIFAT, Université de Tours, France)
Dino Ienco (INRAE, UMR TETIS - LIRMM, France)
Florence Le Ber (ICube, Université de Strasbourg/ENGEES, France)
François Rioult (GREYC, Université de Caen Normandie, France)
Jérôme Gensel (Laboratoire d'Informatique de Grenoble, France)
Joseph Chazalon (LRE, EPITA, Kremlin-Bicêtre, Paris, France)
Loïc Salmon (LABISEN, Yncrea Ouest, France)
Ludovic Moncla (LIRIS, INSA Lyon, France)
Marie-Noëlle Bessagnet (LIUPPA, Université de Pau et des pays de l'Adour, France)
Nicolas Boutry (LRE, EPITA, Kremlin-Bicêtre, Paris, France)
Nida Meddouri (LRE, EPITA, Kremlin-Bicêtre, Paris, France)
Roberto Interdonato (CIRAD, UMR TETIS, France)
Thomas Guyet (Inria, AIstroSight, France)
Thomas Lampert (ICube, Université de Strasbourg, France)

TABLE DES MATIÈRES

Le temps, un challenge à prendre en considération dans l'attrition des employés

Youssef Oubelmouh*,**, Frédéric Fargon*, Cyril de Runz**, Arnaud Soulet**, Cyril Veillon*

*Devoteam, Levallois-Perret, prenom.nom@devoteam.com
** BDTLN, LIFAT, Université de Tours, Blois, prenom.nom@univ-tours.fr

Résumé. Réduire le taux d'attrition des employés est devenu un objectif majeur au sein des entreprises. Il existe une grande quantité de travaux sur la prédiction de l'attrition des clients mais beaucoup moins sur l'attrition des employés bien que ce phénomène soit de plus en plus important. Nous montrons dans cet article que le temps a un impact sur l'attrition alors que la majorité des variables utilisées pour prédire les démissions dans la littérature ne prennent pas en compte l'aspect temporel. Pour cela, nous proposons d'identifier si une variable temporelle est significativement différente entre les employés partis et ceux restés en nous appuyant sur les tests statistiques de Mann-Whitney U. Nous appliquons notre approche à des données internes à une entreprise de services du numérique permettant d'isoler des variables comme par exemple la durée des différentes missions effectuées. Notre étude conclut qu'il est nécessaire d'avoir la trajectoire temporelle des employés pour mieux appréhender le phénomène de l'attrition.

1 Introduction

L'attrition des employés est un problème qui s'accentue de plus en plus dans les entreprises technologiques à travers le monde et plus particulièrement dans les entreprises de services du numérique (ESN). Selon la DARES [1] (Adrien Lagouge, 2022), la France a enregistré près de 520 000 démissions par trimestre, à la fin de l'année 2021 et au début de l'année 2022, dont 470 000 démissions de contrats de travail à durée indéterminée (CDI), ce qui représente un taux de démissions de 2.7%. Ce taux est nettement plus élevé dans les ESN, il atteint par exemple 19% chez Accenture, 16% chez Atos et même 25% dans le secteur de la technologie en Inde (Standard, 2022). En effet, les consultants restent en moyenne entre 3 et 8 ans dans leur ESN (Morrison, 2021). Les démissions spontanées impliquent une baisse de productivité, d'autant plus si l'employé était spécialisé dans un domaine ou s'il avait de l'ancienneté. L'acquisition de nouveaux employés engendre de nombreux coûts et de la perte de temps que ce soit pour la recherche lors de la phase de recrutement ou la formation et l'adaptation

1. DARES : Direction de l'Animation de la Recherche, des Études et des Statistiques

à un nouvel environnement après l'embauche. Le coût moyen du désengagement et de la non-disponibilité en France est de 14 580 euros par an et par salarié dont 9 185 euros de maîtrisables selon les conclusions de l'IBET [2] présenté par le groupe APICIL (2019). Il est donc préférable pour les entreprises d'identifier et d'activer les leviers leur permettant de retenir ses employés qui souhaitent partir.

Beaucoup d'études basées sur des approches issues des sciences humaines (Mobley, 1977) ont été effectuées depuis plus de 30 ans pour essayer de déterminer quels sont les facteurs qui poussent les employés à démissionner. Plus récemment, surtout depuis la diffusion par IBM d'un jeu de données fictives en 2016 [3], des travaux se sont penchés sur l'utilisation d'approches de fouille de données et d'apprentissage automatique dans un but principal de prédiction, car ce jeu représente un snapshot donnant très peu d'informations sur les dynamiques latentes. Par exemple, Zhao et al. (2018) sont partis de deux bases de données de ressources humaines, dont celle d'IBM, de tailles et de nombre d'attributs différents pour en déduire 1) que la taille des jeux d'apprentissage a un effet important sur la qualité des modèles et 2) que les algorithmes d'ensemble eXtreme Gradiant Boosting, Gradient Boosting Tree et Random Forest semblent donner les meilleurs résultats. De leur côté, Yiğit et Shourabizadeh (2017) concluent que les machines à vecteur de support seraient le meilleur type de modèle pour les données d'IBM. Kane-Sellers (2007) indique que la régression logistique fonctionnerait le mieux si l'on se concentre principalement sur trois types de variables : caractéristiques personnelles, caractéristiques du travail, développement des ressources humaines. Brockett et al. (2019) sont les seuls qui ont introduit artificiellement de la temporalité dans leurs données. Pour cela, ils ont décidé aléatoirement pour chaque personne partie, s'il était parti en milieu d'année ou alors en fin d'année sur les données IBM. Ainsi, malgré le nombre grandissant de recherches menées sur la prédiction de l'attrition des employés, nous observons que les dynamiques temporelles dans les trajectoires des employés ne sont que peu ou pas considérées, et cela du fait même de la nature des jeux de données (snapshot). Par exemple, les rares variables liées au temps dans les données IBM ne sont pas assez exhaustives et/ou précises, e.g. « Years In Current Role », « Years With Current Manager ». Il est à noter que si les données fictives d'IBM se sont imposées dans ces recherches, c'est parce que la sensibilité des informations personnelles pouvant y être contenues ne permet pas de proposer des jeux de données ouverts à la communauté (contraintes du RGPD).

Dans cet article, contrairement à la littérature actuelle, nous cherchons à étudier les liens entre l'attrition et les dynamiques temporelles des trajectoires des employés. Nous cherchons à montrer que le nombre de changements entre états, et les durées des états dans ces trajectoires sont liées significativement à l'attrition sur un cas d'étude réel. Ce premier travail semble montrer que les jeux de données exploités dans la littérature sont peu adéquats car ils ne permettent pas de capturer des informations fortement significatives.

Nous commencerons par exposer notre cas d'étude dans la section 2, puis notre méthodologie présentant les variables que nous construisons et les tests statistiques

2. IBET : Indice du Bien-Être au Travail
3. `https://www.kaggle.com/datasets/pavansubhasht/ibm-hr-analytics-attrition-dataset`

(section 3), nous continuerons par l'analyse des résultats obtenus (section 4), puis nous conclurons (section 5).

2 Présentation du cas d'étude

Dans ce travail, nous nous sommes restreints aux employés qui sont ou qui ont été en CDI. Afin d'étudier leur attrition, nous nous intéressons à 4 tables obtenues à travers 3 sources de données différentes issues d'une ESN dont deux logiciels de gestion de ressources humaines. La constitution de notre entrepôt de données a un coût non négligeable en raison des différents processus *Extract-Transform-Load* à gérer tout en respectant le règlement général sur la protection des données (RGPD) particulièrement sensible à ce type de données personnelles.

Les 4 tables utilisées dans cet article sont **Job History**, **Mission History**, **Pricing Profil History** et **Compensation History**. Job History contient les changements liés au poste c'est-à-dire la date d'embauche, et/ou le nom des nouveaux postes (e.g. « Business Analyst », « Contrôleur de Gestion », « Product Owner »). Mission History contient les différentes missions effectuées avec la date de début et la date de fin. Pricing Profil History possède les différents changements liés aux types de profils vendus au client (« Consultant Junior », « Consultant Senior », « Expert Director », etc.). Enfin, Compensation History contient l'historique des salaires. Le tableau 1 décrit succinctement ces différentes tables en indiquant le nombre d'enregistrements, le nombre d'employés renseignés et le nombre d'employés ayant démissionné. Dans la suite de l'article, la variable « Active » vaut 0 pour les démissionnaires, 1 sinon.

Nom	Contenu	Nbr. de lignes	Nbr. d'employés *Active* = 1	Nbr. d'employés *Active* = 0
Job History	Changements liés au poste (embauche, nom du poste)	34 111	8 899	3 669
Job History (2020)	Changements liés au poste (embauche après 2020, nom du poste)	5 491	2 941	388
Mission History	Différentes missions effectuées	36 392	1 743	7 603
Pricing Profil History	Changements du type de profil de l'employé vendu aux clients	20 790	3 764	2 185
Compensation History	Historique des changements de salaire	15 549	8 025	3 214

TAB. 1 – *Description des jeux de données (employé actif dénoté par Active = 1 / employé démissionnaire dénoté par Active = 0).*

De manière générale, nous observons dans le tableau 1 que le nombre d'enregistrements par employé n'est pas très élevé avec une médiane au plus égale à 3 pour la table Mission History. Notons que la table Mission History contient des données subjectives puisque ce sont les employés qui ajoutent eux-mêmes ces données dans leur profil. Par conséquent, les données ne concernent pas exclusivement des missions chez les clients,

mais également des formations, des séminaires ou encore des sous-missions ponctuelles. Les durées peuvent donc être très brèves de l'ordre de quelques jours. L'employé en faisant l'effort de renseigner ces informations dans son cursus traduit une forme d'engagement. Cette table capte donc des signaux informationnels sur la perception de sa situation dans l'entreprise. À l'inverse, les autres tables contiennent des données plus objectives puisqu'elles ne sont pas renseignées par les employés eux-mêmes.

Les tables Job History et Compensation History sont issues du même logiciel qui a été mis en place entre fin 2019 et début 2020. Par conséquent, les employés ayant quitté l'entreprise avant 2020 ne sont pas inclus dans ces tables. De même, les changements intervenus entre la date d'embauche et début 2020 (pour ceux toujours actifs ou ceux ayant quitté l'entreprise après 2020) ne sont pas inscrits dans ces deux tables. Cela a pour conséquence de générer un biais dans nos données. Nous pouvons remarquer que ces données correspondent à la période de la pandémie de la COVID.

Pour chaque mission de la table Mission History, nous disposons de la date de début et la date de fin permettant de calculer la durée de chaque mission pour tous les employés. Cependant, pour la table Pricing Profil History, nous avons seulement la date de changement effective. Sachant que nous n'avons pas à notre disposition les dates de fin de contrat des employés ayant quitté l'entreprise avant 2020, nous ne pouvons pas calculer la durée du dernier état qui serait pourtant une valeur importante. Nous avons donc décidé de nous restreindre à la même catégorie de personnes contenue dans les données de Job History et Compensation History sauf que cette fois nous avons les changements de profil entre la date d'embauche et l'année 2020.

Enfin, la table Compensation History, nous avons également omis les primes pour se concentrer uniquement sur les changements de salaire annuel fixe.

3 Méthodologie

Cette section présente notre méthode pour déterminer si le temps a un impact sur l'attrition. Nous cherchons dans un premier temps à construire des variables traduisant les dynamiques temporelles des trajectoires des employés, appelées *variables temporelles*. Ces variables ont pour objectif de fournir des informations sur le nombre de changements d'états et la durée des états de ces trajectoires. Nous cherchons ensuite à identifier les possibles dépendances significatives, au sens statistique, entre ces variables temporelles et l'attrition des employés.

Construction des variables temporelles Pour chacune des 4 tables de notre cas d'étude, nous créons 1 variable comptant le nombre de changements ayant eu lieu (après suppression ou correction des anomalies trouvées) et 4 variables de durées à partir des durées entre chaque changement qui sont *First duration*, *Last duration*, *Min duration*, *Max duration* qui correspondent respectivement à la durée du premier état, la durée du dernier état, la durée minimale et celle maximale parmi tous les états. Par exemple, pour la table Job History où chaque enregistrement concerne un changement de poste pour un employé, le nombre d'états indique le nombre de changements de poste ; la variable *First duration* indique la durée du premier poste ; etc.

Identification de dépendances temporelles significatives Afin d'identifier des possibles dépendances significatives entre les variables temporelles précédemment construites et l'attrition, nous séparons en deux populations les employés selon l'attrition (partis ou encore actif). Dans l'ensemble de notre étude, nous faisons l'hypothèse de l'indépendance entre les données de chaque employé. Nous cherchons à savoir si ces deux populations sont significativement différentes à la vue de nos variables temporelles.

Pour cela, en statistique, on utilise des tests paramétriques si la distribution de nos données suit une loi (le plus souvent la loi normale), non-paramétrique sinon. La loi normale ou la distribution gaussienne définit une représentation de données selon laquelle la plupart des valeurs sont regroupées autour de la moyenne et les autres s'en écartent symétriquement des deux côtés. Dans notre cas, nous commençons par déterminer si la distribution de nos données est gaussienne avec un test de normalité. Le test de normalité est un test statistique qui permet de déterminer si un échantillon de données suit une distribution normale. Pour cela, nous utilisons à la fois une méthode visuelle avec les boîtes à moustaches (box-plot) et une méthode statistique. Les deux méthodes statistiques les plus utilisées pour tester la normalité des données sont le test de Kolmogorov-Smirnov (Massey Jr, 1951) et le test de Shapiro-Wilk (Shapiro et Wilk, 1965). Le test de Shapiro-Wilk est considéré comme plus approprié pour les échantillons de petite taille (Razali et al., 2011) (<50 échantillons) alors que le test de Kolmogorov-Smirnov est utilisé pour un nombre d'échantillons plus grand que 50, ce qui est notre cas. Puisque nos données sont de grandes tailles nous avons choisi d'utiliser le test de Kolmogorov-Smirnov. Sur nos variables, les résultats des tests nous montrent que les distributions des données ne suivent pas une loi normale. Nous devons alors recourir à un test non-paramétrique. Pour comparer deux populations, l'une des solutions lorsque la distribution des données ne suit pas une loi normale est le test non-paramétrique de Mann-Whitney U (MWU) aussi connu sous le nom de Wilcoxon-Mann-Whitney test ou encore Rank Sum Test (Mann et Whitney, 1947). Nous avons donc choisi d'utiliser ce test pour comparer nos deux groupes.

Le test de **Mann-Whitney U** (MWU-test) ne suppose aucune distribution spécifique (telle qu'une distribution normale des échantillons) pour calculer les statistiques du test et les p-valeurs. La p-valeur est une probabilité mesurant l'importance d'un résultat statistique. En revanche, le MWU-test suppose que les observations soient indépendantes. Basé sur les rangs, son principe est de déterminer s'il est probable qu'une observation dans un groupe soit supérieure à une observation dans l'autre groupe. En d'autres termes, il s'agit de savoir si un échantillon a une dominance stochastique par rapport à l'autre. L'hypothèse nulle et l'hypothèse alternative de ce test sont les suivantes :

- H0 : Les deux groupes sont échantillonnés à partir de populations ayant des distributions identiques. Généralement, les populations échantillonnées présentent une égalité stochastique.
- H1 : Les deux groupes sont échantillonnés à partir de populations ayant des distributions différentes. La plupart du temps, cela signifie que l'une des populations échantillonnées présente une dominance stochastique.

Nous détaillons maintenant les 4 étapes pour appliquer le MWU-test :

1. Classer les valeurs des deux échantillons de la plus basse à la plus haute, quel que soit le groupe auquel chaque valeur appartient.
2. Additionner les rangs des deux échantillons séparément, R_1 et R_2 (expliquant la dénomination du test par *rank-sum*).
3. Calculer les statistiques du test : $U = min(U1, U2)$ avec $U_1 = n_1 n_2 - n_1(n_1+1)/2 - R_1$ et $U_2 = n_1 n_2 - n_2(n_2+1)/2 - R_2$)
4. Déterminer la p-valeur.

Si la p-valeur est inférieure à 5% le test est statistiquement significatif et l'hypothèse nulle est rejetée. Dans ce cas, nous pouvons conclure que les deux populations sont différentes.

Le test qui a été décrit dans cette section est un test bilatéral, il peut être légèrement modifier pour le transformer en test unilatéral. Le test bilatéral permet de rejeter ou non l'hypothèse nulle énoncée précédemment. Les tests unilatéraux permettent de savoir quelle population a une dominance stochastique sur l'autre. Nous allons donc utiliser le test bilatéral ainsi que les deux tests unilatéraux sur l'ensemble de nos tables.

4 Analyse des résultats

En suivant notre méthodologie, pour chaque table, nous analysons nos variables temporelles construites afin de déterminer leur importance dans la compréhension du phénomène d'attrition. Comme indiqué précédemment, tous les tests de Kolmogorov-Smirnov sur nos variables ont significativement montré qu'aucun des groupes des 4 tables ne suit une loi normale. Aussi, l'ensemble des résultats des tests statistiques donnés dans cette section sont issus des tests de Mann-Whitney U.

4.1 Analyse de la table Job History

Nous commençons par analyser les 5 variables temporelles (le nombre et les 4 durées) construites au regard de la table Job History en considérant dans un premier temps l'ensemble des employés, puis en se restreignant à ceux embauchés après 2020 (pour lesquels les données sont complètes). La figure 1 représente les box-plots pour les 5 variables en séparant les deux populations à savoir employés partis (en bleu) et ceux restés (en orange). Le tableau 2 présente les différents tests de Mann-Whitney U bilatéral puis unilatéraux. « $0 > 1$ » et « $0 < 1$ » signifient que la valeur de la variable considérée est respectivement supérieure et inférieure pour les employés partis par rapport à ceux qui sont restés.

Nous pouvons voir que pour le Job History les deux populations sont statistiquement différentes (rejet de l'hypothèse nulle) concernant toutes les variables (nombre de jobs et durées des jobs). Les tests unilatéraux nous montrent que le nombre de jobs ainsi que la durée du dernier job sont inférieurs chez ceux qui sont partis et que les autres durées sont supérieures chez les employés toujours actifs. Nous pouvons en déduire que l'attrition survient plus fréquemment chez les employés qui restent longtemps sur un même job et n'en changent pas assez. Ces derniers décident donc de partir au bout de quelques mois après le début de leur dernier job. Cependant, ces résultats

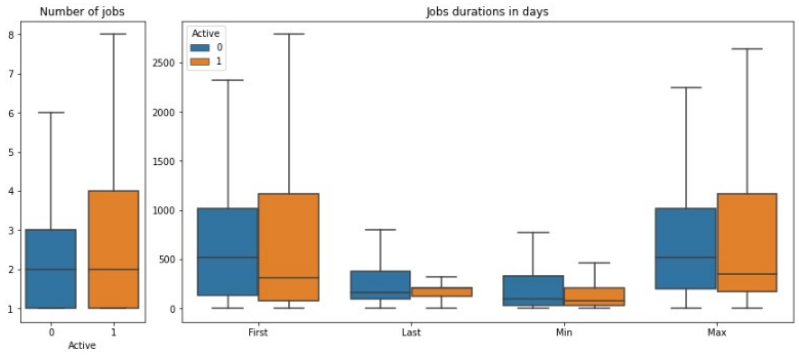


FIG. 1 – *Box-plots des variables pour la table job history.*

	bilatéral	unilatéral 0 > 1	unilatéral 0 < 1
Count	<0.001	1.0	<0.001
First duration	<0.001	<0.001	1.0
Last duration	0.014	0.992	0.007
Min duration	<0.001	<0.001	1.0
Max duration	<0.001	<0.001	1.0

TAB. 2 – *P-valeurs des variables pour la table Job History.*

sont à pondérer car, comme mentionné dans la description des données, cette table ne contient pas les changements survenus avant 2020. Par conséquent, cette absence biaise les données des personnes ayant été embauchées dans l'entreprise avant 2020. Nous allons donc maintenant nous focaliser uniquement sur les employés arrivés après 2020 afin de confirmer ou d'infirmer notre analyse précédente.

	bilatéral	unilatéral 0 > 1	unilatéral 0 < 1
Count	<0.001	1.0	<0.001
First duration	0.003	0.002	0.998
Last duration	<0.001	1.0	<0.001
Min duration	0.032	0.016	0.984
Max duration	0.042	0.979	0.021

TAB. 3 – *P-valeurs des variables pour la table Job History (depuis 2020).*

Si on se concentre uniquement sur les employés ayant intégré l'entreprise après 2020, nous retrouvons toujours des différences statistiquement significatives entre les deux populations pour toutes les variables temporelles (voir figure 2 et tableau 3). On remarque par contre que la variable max est maintenant inférieure chez les personnes ayant démissionné contrairement à notre analyse précédente. Cela peut être expliqué

Importance du temps dans l'attrition des employés

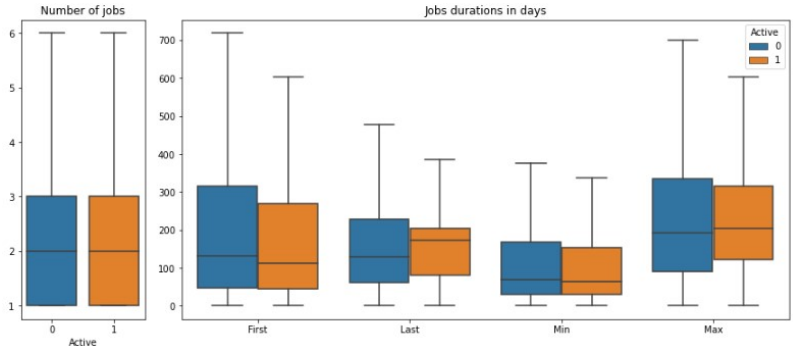


FIG. 2 – *Box-plots des variables pour la table Job History (depuis 2020).*

par le fait qu'avec moins de 3 ans de contrat, la durée maximale d'un job sera forcément plus petite que la durée de ceux qui sont toujours en poste à l'heure actuelle.

4.2 Analyse de la table Mission History

Nous appliquons la même méthode d'analyse sur les variables temporelles de la table des missions en nous appuyant sur la figure 3 et sur le tableau 4.

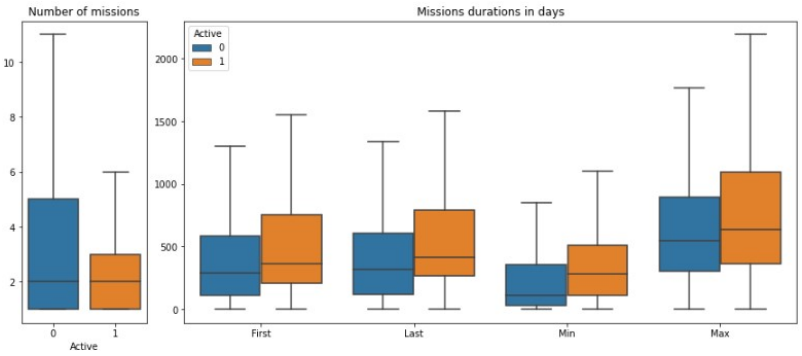


FIG. 3 – *Box-plots des variables pour la table Mission History.*

Concernant les missions, nous observons également que toutes les variables sont significativement différentes entre les deux populations. Le nombre de missions effectuées par les employés qui démissionnent est plus important que le nombre de missions de ceux qui restent. Cependant, les valeurs de toutes les variables de durées sont inférieures. Cela signifie donc que les employés ayant eu beaucoup de missions et ce, d'autant plus avec des courtes durées, ont tendance à partir.

	bilatéral	unilatéral 0 > 1	unilatéral 0 < 1
Count	<0.001	<0.001	1.0
First duration	<0.001	1.0	<0.001
Last duration	<0.001	1.0	<0.001
Min duration	<0.001	1.0	<0.001
Max duration	<0.001	1.0	<0.001

TAB. 4 – *P-valeurs des variables pour la table Mission History.*

4.3 Analyse de la table Pricing Profile History

Nous considérons maintenant les variables temporelles de la table du profil de coût des employés en nous appuyant sur la figure 4 et sur le tableau 5.

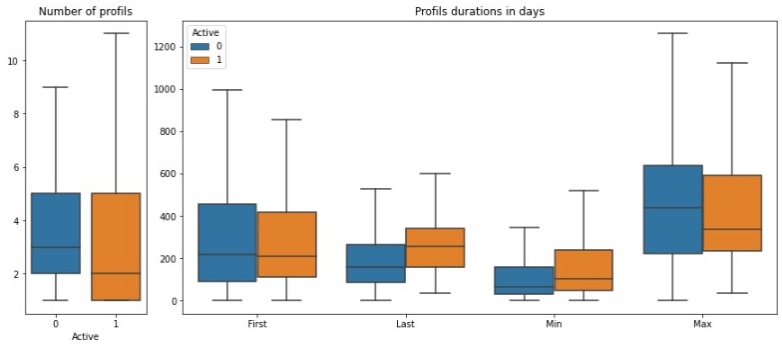


FIG. 4 – *Box-plots des variables pour la table Pricing Profile History.*

	bilatéral	unilatéral 0 > 1	unilatéral 0 < 1
Count	<0.001	<0.001	1.0
First duration	0.388	0.805	0.194
Last duration	<0.001	1.0	<0.001
Min duration	<0.001	1.0	<0.001
Max duration	0.006	0.003	0.997

TAB. 5 – *P-valeurs des variables pour la table Pricing Profil History.*

Pour la table Pricing Profil History, le comportement des deux populations ne se différencie pas significativement suivant toutes les variables temporelles. En effet, d'après les *p*-valeurs du test de Mann-Whitney U nous avons la première durée qui ne rejette pas l'hypothèse nulle selon laquelle les deux populations sont similaires. Nous avons donc la première durée de profil qui est similaire pour les deux populations. Les tests unilatéraux nous montrent que ceux qui ne sont plus en poste ont subi plus de modifications de profil et que la durée max d'un de leur profil est plus élevé par rapport

à ceux qui sont toujours sous contrat. Tandis que la durée minimale parmi toutes les durées ainsi que la dernière durée sont plus élevées pour les personnes encore actives.

4.4 Analyse de la table Compensation History

Cette section porte sur les variables temporelles de la table des salaires des employés en nous appuyant sur la figure 5 et sur le tableau 6.

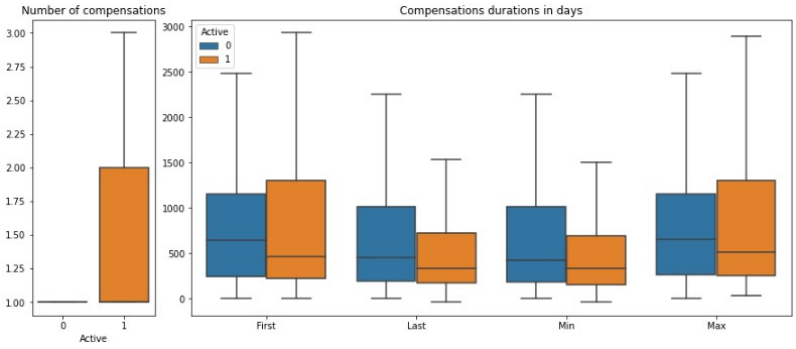


FIG. 5 – *Box-plots des variables pour la table Compensation History.*

	bilatéral	unilatéral 0 > 1	unilatéral 0 < 1
Count	<0.001	1.0	<0.01
First duration	0.055	0.027	0.973
Last duration	<0.001	<0.001	1.0
Min duration	<0.001	<0.001	1.0
Max duration	0.262	0.131	0.869

TAB. 6 – *P-valeurs des variables pour la table Compensation History.*

Dans la table Compensation History, le test bilatéral sur la donnée First duration ne montre pas de différence significative entre les deux populations. Pourtant, le test unilatéral nous montre la supériorité significative de cette variable pour les personnes ayant démissionné. Cela est confirmé visuellement par la boîte à moustache de cette variable qui présente une médiane bien plus élevée que celle des personnes qui sont restées et ce, malgré que les troisièmes quartiles soient inversés pour les deux groupes. La durée maximale avec le même salaire n'est pas différente. Quant aux autres variables temporelles, elles ont une valeur plus élevée pour les non actifs. Comme attendu, nous pouvons déduire de ces résultats que les employés qui démissionnent ont moins de changement de salaire et que leur salaire reste fixe durant une plus longue durée.

4.5 Comparaison des résultats

Afin d'identifier des informations intéressantes quant à l'impact du temps sur l'attrition, nous comparons les résultats des tests unilatéraux entre les différentes tables. Le tableau 7 permet de comparer à l'aide de flèches les employés ayant démissionné par rapport à ceux toujours en CDI. Une flèche dirigée vers le haut signifie que la valeur de la variable concernée est supérieure chez les employés inactifs, une flèche vers le bas signifie l'inverse et un tiret signifie qu'il n'y pas de différences entre les deux populations.

	Job	Mission	Pricing	Compensation
Count	↓	↑	↑	↓
First duration	↑	↓	-	-
Last duration	↓	↓	↓	↑
Min duration	↑	↓	↓	↑
Max duration	↑	↓	↑	-

TAB. 7 – *Supériorité et infériorité des employés ayant quitté l'entreprise.*

À partir de ce tableau, nous pouvons en déduire les observations et possibles explications présentées dans le tableau 8. Ces informations nous semblent utiles pour les ressources humaines et surtout indique clairement l'apport que peut représenter une analyse des dynamiques temporelles pour expliquer et prédire l'attrition des employés. Attention, du fait des biais de nos données, les comportements observés peuvent être différents de la réalité.

Tables d'origine	Observations groupe démission	Explications possibles
Mission	Nombre élevé de missions de courtes durées.	Les employés pourraient chercher une certaine stabilité dans leur travail.
Job	Nombre bas de jobs et en plus de longues durées.	Les employés semblent vouloir évoluer dans l'entreprise et ne pas rester au même niveau.
Compensation	Nombre d'augmentations de salaire faible et salaire fixe durant une plus longue période.	Les augmentations seraient perçues comme peu fréquentes.
Job/Mission	Beaucoup de missions effectuées avec des durées plus courtes mais résultats opposés pour job.	Les employés veulent sans doute de l'évolution dans leur job mais une stabilité sur les lieux de missions.
Pricing/Compensation	Nombre élevé de pricings de courte durée et opposé pour le salaire.	Le pricing profile évolue rapidement mais le salaire ne semble pas suivre.
Mission/Pricing	Nombre élevé de changements de pricing profil et de missions et durée plus courte (sauf pour le max).	Les changements de missions entraînent des changements de clients permettant à l'entreprise de modifier le pricing profile de l'employé.

TAB. 8 – *Observations et tentatives d'explications des démissions des employés.*

5 Conclusion

Nous avons vu à travers le test statistique de Mann-Whitney U appliqué aux 4 tables que nos variables temporelles sont souvent significativement différentes entre les employés toujours en CDI et ceux qui avaient démissionnés. Cela nous conforte dans l'idée que le temps a un impact sur l'attrition. Mais il s'avère dans la littérature existante que ce facteur temps n'est pratiquement jamais utilisé pour prédire l'attrition des employés. Le fait d'avoir des évolutions et des changements est a priori un facteur sur lequel les entreprises pourraient agir pour remédier à l'attrition, ce qui n'est pas traité dans la littérature orientée informatique, et représente donc une faille dans les outils numériques actuels à disposition des RH.

Pour renforcer notre étude, l'une des pistes est de faire une distinction entre jeunes et moins jeunes salariés car d'après l'INSEE [4] (Picart, 2014), le taux de rotation chez les jeunes est beaucoup plus élevé que chez les seniors. Comme la période d'étude des données chevauche la période de la pandémie de la COVID, il faudra vérifier nos hypothèses sur les années antérieures. De plus, une perspective découlant de cette étude est de construire des séquences qui caractérisent les trajectoires des employés dans l'entreprise, depuis l'embauche jusqu'à la démission. Sur ces séquences d'événements, nous chercherons à la fois à prédire mais aussi à expliquer l'attrition à l'aide d'extractions de motifs séquentiels significatifs facilement interprétables.

Références

Adrien Lagouge, Ismaël Ramajo, V. B. (2022). La france vit-elle une "grande démission". Technical report, DARES : Direction de l'animation de la recherche, des études et des statistiques.

APICIL, G. (2019). Ibet – désengagement des salariés : un coût de 14580€/an/salarié. Technical report, Groupe APICIL.

Brockett, N., C. Clarke, M. Berlingerio, et S. Dutta (2019). A system for analysis and remediation of attrition. In *2019 IEEE International Conference on Big Data (Big Data)*, pp. 2016–2019.

Kane-Sellers, M. L. (2007). *Predictive models of employee voluntary turnover in a North American professional sales force using data-mining analysis*. Ph. D. thesis, Texas University.

Mann, H. B. et D. R. Whitney (1947). On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics*, 50–60.

Massey Jr, F. J. (1951). The kolmogorov-smirnov test for goodness of fit. *Journal of the American Statistical Association 46*(253), 68–78.

Mobley, W. H. (1977). Intermediate linkages in the relationship between job satisfaction and employee turnover. *Journal of applied psychology 62*(2), 237.

Morrison, G. (2021). How long do people stay at their firms ? Technical report, Consulting Point.

Picart, C. (2014). Une rotation de la main d'œuvre presque quintuplée en 30 ans : plus qu'un essor des formes particulières d'emploi, un profond changement de leur usage. Technical report, INSEE : Institut National de la Statistique et des Études Économiques.

4. INSEE : Institut National de la Statistique et des Études Économiques

Razali, N. M., Y. B. Wah, et al. (2011). Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests. *Journal of statistical modeling and analytics 2*(1), 21–33.

Shapiro, S. S. et M. B. Wilk (1965). An analysis of variance test for normality (complete samples). *Biometrika 52*(3/4), 591–611.

Standard, B. (2022). It industry logged 25% attrition in fy22, trend to continue : Report. Technical report, Business Standard.

Yiğit, İ. O. et H. Shourabizadeh (2017). An approach for predicting employee churn by using data mining. In *2017 International IDAP*, pp. 1–4. IEEE.

Zhao, Y., M. K. Hryniewicki, F. Cheng, B. Fu, et X. Zhu (2018). Employee turnover prediction with machine learning : A reliable approach. In *Proc. of SAI intelligent systems conference*, pp. 737–758. Springer.

Summary

Reducing employee attrition has become a major objective within companies. There is a large amount of scientific literature on the prediction of customer attrition but much less on employee attrition, although there is a growing body of literature. We want to show in our paper that time has an impact on attrition, since most of the variables used to predict resignations in the literature do not take into account the temporal aspect. To do so, we propose to identify whether a temporal variable is significantly different between employees who left and those who stayed using Mann-Whitney U statistical tests. We apply our approach to internal data of an information technology consulting company allowing us to isolate variables such as the duration of the different missions performed. Our study concludes that it is necessary to have the temporal trajectory of employees to better understand the attrition phenomenon.

Vers un outil d'évaluation comparative pour la maintenance prédictive : comment comparer différentes approches ?

Antoine Guillaume*, Christel Vrain*, Elloumi Wael**

*LIFO, Université d'Orléans, 6 Rue Léonard de Vinci, 45067 Orléans
christel.vrain@univ-orleans.fr
antoine.guillaume@ensta-paris.fr
**Worldline, 19 Rue de la Vallée Maillard, 41000 Blois
wael.elloumi@worldline.com

Résumé. L'industrie 4.0 et la maintenance prédictive sont des domaines très actuels, les applications industrielles sont aujourd'hui nombreuses, ainsi que les articles présentant des modèles prédictifs pour résoudre des cas d'applications. Il y a cependant dans la littérature un manque d'articles traitant des problématiques liées à l'évaluation et la comparaison de ces modèles. Cela peut conduire à l'utilisation de protocoles expérimentaux et de métriques non adaptés à l'application. Également, la comparaison entre différentes familles de modèle prédictif (p. ex. régression et classification) n'est pas systématique, car les métriques utilisées sont différentes. Dans cet article de positionnement, nous souhaitons attirer l'attention de la communauté sur les problématiques liées à l'évaluation des modèles de maintenance prédictive. Dans un premier temps, nous introduisons le domaine de la maintenance prédictive ainsi qu'une formulation communément utilisée pour la tâche d'apprentissage. Ensuite, nous montrons les failles des protocoles expérimentaux classiques, et proposons une alternative permettant d'évaluer et de comparer des modèles en fonction du processus de maintenance associé à l'application.

1 Introduction

L'objectif de la maintenance prédictive est d'optimiser les coûts de production. Cela passe par la réduction de certains coûts liés aux opérations de maintenance et par la maximisation du temps pendant lequel les machines sont productives. En effet, le système ayant la capacité de planifier des maintenances à l'avance, les opérations peuvent être effectuées au moment où le coût d'arrêt de la machine est minimum (p. ex. lorsqu'elle n'est pas utilisée).

Cependant, la politique de maintenance peut être différente en fonction des cas d'utilisation. Par exemple, dans le cas d'un système impactant la sécurité des personnes, on jugera qu'il vaut mieux effectuer une maintenance avant tout risque d'incident, au détriment de la productivité de la machine. Chaque application défini donc sa propre politique de maintenance, située entre une fiabilité maximale et une réduction des coûts maximale.

Dans une revue récente de Esteban et al. (2022), on constate une grande diversité dans le choix des modèles et des prétraitements utilisés pour résoudre des cas pratiques. De nombreux

articles présentent des modèles sur des jeux de données publics, tel que le "NASA Turbofan Jet Engine Data Set" publié par Saxena et al. (2008). Cependant, Zschech et al. (2019) montre que sur ce même jeu de données, de nombreuses modélisations du problème sont possibles. Cela rend la comparaison entre les approches difficiles, et à notre connaissance, aucun outil n'a pour le moment été proposé pour comparer ces modèles de manière équitable.

Dans le domaine de l'ingénierie de fiabilité (O'Connor et Kleyner, 2012), la performance d'un système de maintenance est définie par une notion de coût. Celui-ci dépend des machines étudiées, du processus de maintenance et du moment où la maintenance est effectuée. C'est là qu'une première fracture apparaît entre le champ de l'ingénierie de fiabilité et celui de l'apprentissage automatique. Cette notion de coût d'un système de maintenance fait partie des notions de base dans le cadre de l'ingénierie de fiabilité, alors qu'elle n'apparaît que rarement dans les articles présentant des modèles d'apprentissage automatique. De plus, l'utilisation de métrique comme l'erreur moyenne absolue ou le F1-score peut mener à une mauvaise interprétation des performances d'un modèle pour les applications de maintenance prédictive.

Dans cet article, notre objectif est de proposer une méthodologie pour évaluer le coût d'un système de maintenance, indépendamment du type de modèle utilisé. Pour cela, nous définissons un protocole expérimental s'adaptant aux contraintes applicatives et aux différentes formulations du problème. Dans un premier temps, nous introduisons le domaine de la maintenance prédictive (types de données utilisées, extraction d'un jeu de données) et formalisons la tâche d'apprentissage. Ensuite, nous présentons un protocole de base, souvent utilisé dans la littérature, et montrons les problèmes qu'il peut causer. Enfin, pour répondre aux problématiques identifiées, nous définissons notre protocole et notre métrique d'évaluation, basée sur les coûts du processus de maintenance.

2 Formalisation du problème

Dans cette section, nous introduisons plus formellement l'application d'un modèle d'apprentissage à un cas de maintenance prédictive. Par souci de simplicité, nous considérons que nous disposons de données émises par une ou plusieurs machines et que notre objectif est simplement de prédire les défaillances, indépendamment des parties de la machine les provoquant.

Il existe cependant des cas où l'on souhaite identifier la partie de la machine qui va causer la panne, par exemple, pour apporter des informations complémentaires lors d'un diagnostic effectué par l'équipe de maintenance, ou simplement pour ne pas avoir à réaliser de diagnostic. La formulation du problème peut facilement s'adapter à ce cas d'usage, par exemple, en considérant un problème multiclasse plutôt que binaire (c.-à-d. émettre une alerte ou non à un temps t) pour une tâche de classification. Il est aussi possible de découper le problème en sous problèmes, avec un modèle par partie à monitorer.

Notre objectif dans les sections suivantes n'est pas de fournir une revue des modélisations possibles pour les problèmes de maintenance prédictive, déjà effectuée par Esteban et al. (2022), mais d'introduire une modélisation communément utilisée dans la littérature, afin de pouvoir définir formellement notre protocole.

2.1 Type de données pour la maintenance prédictive

On peut distinguer plusieurs cas concernant l'origine des données utilisées pour apprendre un modèle prédictif :

— Le premier cas, qui est le plus courant, est celui de données issues de capteurs qui mesurent des phénomènes physiques, tels que l'amplitude d'une vibration, comme dans Orhan et al. (2006), ou la température. Ces données sont représentées sous forme de séries temporelles, qui représentent l'évolution des valeurs capturées au fil du temps.

— Le deuxième cas est celui de données issues de journaux d'événement, qui sont produits par un système informatique. Par exemple, Wang et al. (2017) présente un cas d'utilisation de données catégorielles, avec des journaux contenant des codes d'événements caractérisant une action du système ou un problème de fonctionnement. On peut distinguer les journaux d'événements spécifiquement générés pour la maintenance prédictive des journaux d'événements destinés à la maintenance logicielle.

— Plus rarement, on trouve aussi des séries d'images, chacune pouvant permettre d'évaluer l'état d'usure d'une partie du système. Par exemple, Schlagenhauf et Burghardt (2021) propose un système basé sur l'identification des marques d'usures sur des parties d'une machine.

Un point commun existe entre toutes ces sources de données : la composante temporelle. Dans la suite, nous considérerons donc que nous disposons de séries temporelles, indépendamment de leur contenu.

2.2 Extraire un jeu de donnée de maintenance prédictive

Considérons une série temporelle $X = \{x_1, \ldots, x_m\}$ (ici univariée), contenant les données générées par une machine dans un intervalle de temps $[1, m]$. À partir d'un historique des défaillances sur cette période, contenant la date de début et de fin de chaque défaillance, on souhaite extraire des sous-séquences adaptées pour apprendre un modèle de maintenance prédictive. Si on a plusieurs machines, l'ensemble des sous-séquences extraites de chaque machine formera le jeu de données. Pour définir un processus d'extraction adapté au cas d'usage, on peut poser les questions suivantes :

— Doit-on déterminer la partie de la machine responsable de la défaillance ?

— Dans le cas de séries multivariées, les variables sont elles indépendantes les unes des autres ?

— L'historique des défaillances contient il la cause des pannes et les opérations effectuées pour les résoudre ?

Nous définissons les sous-séquences extraites de X comme des cycles de vie. Chaque cycle de vie contient les données émises par la machine pendant un intervalle de fonctionnement normal. Cet intervalle commence après la résolution d'une panne et prend fin au moment de la prochaine panne. Un cycle de vie est donc défini par une machine, depuis laquelle il est extrait, un intervalle de temps, et si la cause de la panne est connue, une partie de la machine.

En fonction des caractéristiques des séries temporelles, le processus d'extraction des cycles de vie peut changer. Par exemple, si on dispose de capteurs qui mesurent la température de plusieurs parties d'une machine, il est possible qu'une augmentation de la température causée par une partie affectera également les autres capteurs. Dans ce cas, créer des cycles de vie indépendants pour chaque partie de la machine, qui contiendrait donc uniquement les données émises

par le capteur de la partie concernée, pourrait mener à des faux positifs dus aux changements de température causés par d'autres parties.

Plus formellement, considérons une série $X = \{x_1, \ldots, x_m\}$ représentant les données émises par une machine. Supposons qu'on connaisse l'existence de deux pannes, chacune définie par une date de début et de fin (c.-à-d. le moment où la panne se produit et la fin de la maintenance). On définira ces dates par $[t_1, t_2[$ pour la première panne, et $[t_3, t_4[$ pour la seconde. À partir de ces deux pannes, on peut extraire trois cycles de vie de X. Le premier serait $X_1 = \{x_1, \ldots, x_{t_1-1}\}$, le second $X_2 = \{x_{t_2}, \ldots, x_{t_3-1}\}$ et le troisième $X_3 = \{x_{t_4}, \ldots, x_m\}$. Dans ce cas, X_3 ne se termine pas par une panne, et peut ne pas être utilisable par certaines approches, dû à l'incertitude sur la date de la panne. Par exemple, un modèle de régression devrait connaitre le temps restant avant la panne à chaque point du temps pour pouvoir utiliser X_3 durant la phase d'entraînement. Enfin, si les parties responsables des pannes sont connues, on les affectera aux cycles de vie correspondants.

À partir de maintenant, on notera $\mathcal{X} = \{X_1, \ldots, X_n\}$ l'ensemble des cycles de vie extrait des données brutes émises par les machines, avec n le nombre de cycles de vie. On peut également disposer d'un vecteur $Y = \{y_1, \ldots, y_n\}$ qui contient la classe (c.-à-d. la partie causant la défaillance) de chaque cycle de vie, si celle-ci est connue. Comme nous nous concentrons sur les modèles prédisant les pannes indépendamment des parties responsables, la classe y_i ne sera pas directement utilisée pour l'apprentissage d'un modèle. Il est cependant simple, par exemple, pour le cas d'un modèle de classification, de passer du cas binaire (panne/non panne) à un cas multiclasse où la cause de la panne est également prédite.

2.3 Problèmes de classification

Dans un contexte de classification, l'objectif est de déterminer si, au temps t, une alerte de maintenance devrait être lancée. Idéalement, une alerte permettra de planifier une intervention au moment où le coût de non-fonctionnement de la machine est minimum.

Pour créer un classifieur capable de prédire les pannes avec suffisamment d'avance, on peut définir des variables qui caractérisent le processus de maintenance. Bien que cette notion de variables soit présente dans la majeure partie de la littérature, il ne semble pas exister de consensus pour leur définition. Nous définissons ces variables en nous basant sur les définitions de Sipos et al. (2014), qui nous semblent les plus générales :

— **Temps de Réponse** (tr) : une durée correspondant au temps nécessaire pour effectuer la maintenance depuis le moment où une alerte est lancée. Cela comprend le temps nécessaire à l'équipe de maintenance pour arriver sur site et pour réaliser la maintenance.
— **Intervalle Prédictif** (ip) : une durée caractérisant l'avance avec laquelle il est acceptable de prédire une panne. Une alerte levée dans cet intervalle permet de réaliser une maintenance dans des conditions optimales.

L'approche la plus répandue pour intégrer ces contraintes est d'extraire des fenêtres glissantes de chaque cycle de vie, puis d'étiqueter chaque fenêtre grâce aux valeurs de tr et ip. Soit $\mathcal{W}_i = \{W_1, \ldots, W_{m-(l-1)}\}$ l'ensemble des fenêtres glissantes de taille l extraites d'un cycle de vie $X_i = \{x_1, \ldots, x_m\}$ qui subit une panne au temps t_{m+1}. Dans un contexte de classification binaire, où on souhaite prédire les pannes indépendamment des parties responsables, on définit la classe 1 comme les fenêtres dans lesquelles on souhaite identifier des signatures

de panne, la classe z_i d'une fenêtre $W_i = \{x_i, \ldots, x_{i+(l-1)}\}$ est alors définie telle que :

$$z_i = \begin{cases} 1 & \text{si } i + (l-1) \geq (m+1) - (ip + tr) \\ 0 & sinon \end{cases} \quad (1)$$

La Figure 1 illustre ce processus pour un cycle de vie. Les fenêtres se terminant dans l'intervalle $[(m+1) - (ip + tr), (m+1) - tr]$ peuvent être utilisées pour entrainer le modèle si elles contiennent des signatures de panne qui pourraient survenir avant $(m+1) - (ip + tr)$. Sinon, il est possible de créer une classe spécifique pour ces fenêtres, caractérisant une panne imminente.

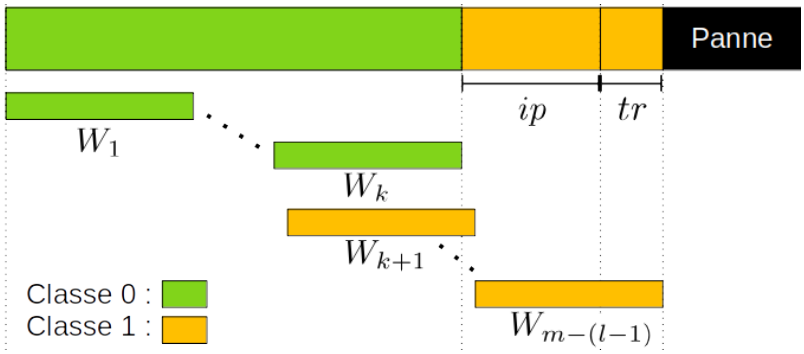


FIG. 1 – *Illustration de la formulation du problème pour les tâches de classification avec une approche utilisant des fenêtres glissantes. W_i est la i^{eme} fenêtre de taille l extraite du cycle de vie, tr représente le temps nécessaire pour effectuer la maintenance, et ip l'intervalle dans lequel il est acceptable de prédire la panne.*

À partir d'un ensemble de fenêtres étiquetées, on peut alors apprendre un classifieur. À ce stade, une étape d'ingénierie de caractéristiques est souvent mise en œuvre pour créer des descripteurs capturant le processus de dégradation de la machine. On appliquera ensuite le modèle de classification choisi, qui, à partir de la fenêtre de taille l la plus récente d'un nouveau cycle de vie, prédira si une alerte doit être levée.

2.4 Problèmes de régression

Dans un contexte de régression, le but est d'estimer au temps t le temps restant avant la prochaine défaillance. Cette quantité est souvent désignée par le terme "remaining useful life" (RUL). Pour lancer une alerte de maintenance, un système utilisant un modèle de régression devra définir un seuil, en dessous duquel une alerte devra être lancée. Ce seuil peut être exprimé en utilisant l'intervalle prédictif (ip) et le temps de réponse (tr) définis dans la section précédente, tel que si $RUL \leq (ip + tr)$, une alerte sera lancée.

Comme pour les modèles de classification, pour apprendre un modèle de régression à partir d'un ensemble de cycles de vie, il est commun d'extraire des fenêtres glissantes. À partir d'un cycle de vie $X_i = \{x_1, \ldots, x_m\}$ et d'une taille de fenêtre l, on extrait un ensemble de fenêtres

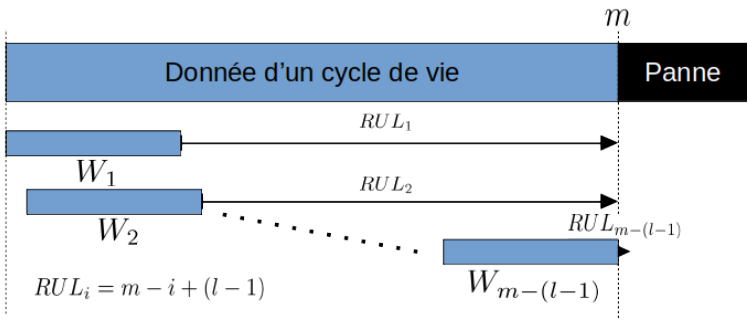


FIG. 2 – *Illustration de la formulation du problème pour les tâches de régression avec une approche utilisant des fenêtres glissantes.*

$\mathcal{W}_\rangle = \{W_1, \ldots, W_{m-(l-1)}\}$ avec $W_j = \{x_j, \ldots, x_{j+(l-1)}\}$. On peut ensuite calculer le RUL de chaque fenêtre par $RUL_j = m - j + (l-1)$. Ces opérations sont illustrées par la Figure 2.

À partir des fenêtres glissantes et de leur RUL, comme pour le cas de la classification, on peut construire un ensemble de descripteurs qui caractérisent le processus de dégradation de la machine, et utiliser un modèle de régression pour estimer les coefficients à affecter à chaque descripteur. On pourra ainsi utiliser les l derniers points de données émis par une machine pour avoir une estimation du temps restant avant la prochaine défaillance.

3 Protocoles existants

Considérons un modèle de classification ou de régression quelconque dont on souhaite évaluer la performance. Pour cela, il est commun d'utiliser un processus de validation où on divise les données en entrée en deux ensemble disjoints, l'un pour l'entrainement du modèle et l'autre pour l'évaluation de ses performances. On trouve dans la littérature l'utilisation de validation simple, où on spécifie seulement la taille des deux ensembles, comme dans Gutschi et al. (2019), mais aussi de validation croisée où cette étape est répétée pour que toutes les données soient apparues dans l'ensemble de test, comme dans le framework décrit par Vallim Filho et al. (2022).

L'évaluation des performances s'effectue ensuite avec des métriques adaptées au type modèle choisi. Par exemple, Bampoula et al. (2021) utilise le F1-score et l'exactitude pour évaluer un modèle de classification. On retrouve aussi ce type de métrique pour des modèles de régression, comme dans Gutschi et al. (2019), où à partir d'une prédiction du temps restant avant la panne, et d'un seuil de déclenchement d'une alerte, on peut calculer une mesure d'exactitude. On pourra noter que plusieurs points d'importance sont parfois occultés dans les expérimentations :

— Si une seule étape de validation est effectuée (et non une validation croisée), pourquoi ce choix ? Cela peut être justifier par des temps de calcul important, mais en l'absence d'argument, cela peut cacher un choix intentionnel des échantillons pour maximiser les métriques (c.-à-d. le phénomène de "cherry picking").

— La création des ensembles d'entrainement et de test est-elle faite sur les cycles de vie ou sur les fenêtres glissantes ? Dans ce dernier cas, garantit-on que toutes les fenêtres issues d'un cycle de vie X_i soit dans un seul des deux ensembles à chaque étape de validation ?

Dû à l'absence d'articles traitant des protocoles expérimentaux à utiliser pour les applications de maintenance prédictive, nous présentons un protocole "de base" pour l'évaluation des modèles d'apprentissage.

Ce protocole utilise une validation croisée en k étapes, utilisant une proportion de $\frac{1}{k}$ des cycles de vies pour l'ensemble d'entraînement. Le découpage en fenêtres glissantes devient alors une étape de prétraitement de la méthode d'apprentissage, afin d'éviter les fuites de données entre l'ensemble d'entrainement et de test. En effet, si on effectuait une validation croisée sur les fenêtres glissantes sans aucune contrainte, une fenêtre $W_i = \{x_i, \ldots, x_{i+(l-1)}\}$ pourrait être dans l'ensemble d'entraînement, et la fenêtre W_{i+1} dans l'ensemble de test. Cela créerait une fuite de données, due aux points partagés par W_i et W_{i+1}. Pour évaluer la performance du modèle, les métriques sont calculées entre la classe prédite $\hat{z}_i$ de chaque fenêtre W_i et sa vraie classe z_i.

Notons que lors de l'utilisation réelle du modèle appris, on traitera un flux de données, par lots ou en temps réel, qui représentera les données du cycle de vie actuel de la machine : si au temps t le modèle donne $\hat{z}_{t-(l-1)} = 1$ pour la fenêtre $W_{t-(l-1)}$, contenant les dernières données émises par la machine, un processus de maintenance sera déclenché.

3.1 Problèmes liés au protocole de base

Considérons un cycle de vie $X_i = \{x_1, \ldots, x_m\}$ et les fenêtres glissantes de taille l préalablement extraites de X_i. À partir d'un modèle prédictif, on obtient un vecteur $\hat{Z}_i = \{\hat{z}_1, \ldots, \hat{z}_{m-(l-1)}\}$.

Si le protocole de base défini dans la section précédente est suffisant pour évaluer la performance du modèle sur la tâche d'apprentissage, il n'est en aucun cas adapté pour estimer la performance du système de maintenance prédictive qui utilisera ce modèle. On doit en fait distinguer deux contextes d'évaluation : l'évaluation du modèle sur la tâche d'apprentissage, et l'évaluation du modèle sur l'application.

Dans le cadre des métriques sur la tâche d'apprentissage, les coûts liés au processus de maintenance ne sont pas pris en compte. De plus, il est important de définir les hypothèses suivantes :

— Un diagnostic est-il réalisé avant chaque opération de maintenance déclenchée suite à une alerte du modèle ? Ce diagnostic peut-il annuler une intervention déclenchée suite à un faux positif ?

— Quel est l'impact d'une maintenance sur les données générées par la machine ? Comment le comportement de la machine est-il affecté par ces opérations ?

En effet, les métriques ne doivent pas être les mêmes si un diagnostic peut détecter les faux positifs sans influer sur le comportement de la machine ou si une maintenance qui modifie le comportement (p. ex. via le changement d'une pièce) est obligatoirement effectué à chaque fois qu'on a $\hat{z}_t = 1$. Notons qu'il est possible d'affecter des poids aux différents types d'erreurs, mais l'emplacement de l'erreur est rarement pris en compte.

Si une maintenance effectuée au temps t suite à une alerte modifie le comportement de la machine, il n'y a aucune garantie que, dans le cas réel, le modèle voit les données situées

après t. Supposons qu'il existe un biais qui pousse le modèle à prédire $\hat{z}_1 = 1$ pour tous les cycles de vie, mais que toutes les autres prédictions sont correctes. Dans ce cas, des métriques telles que l'exactitude entre $\hat{Z}_i$ et Z_i seraient très bonnes, alors que le modèle déclencherait systématiquement des maintenances précoces. La Figure 3 illustre cet exemple. Dans le cas où un diagnostic est effectué, si les faux positifs peuvent être détectés, il faut alors considérer le coût du diagnostic.

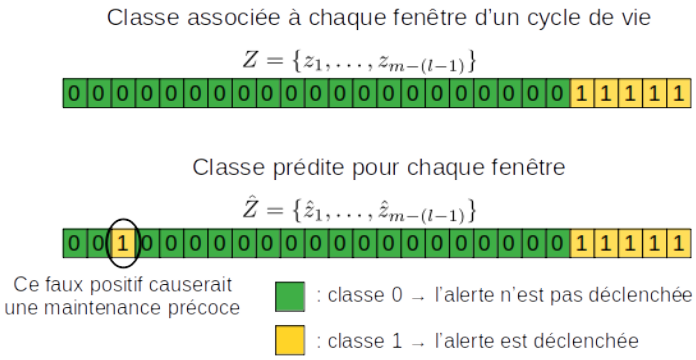


FIG. 3 – *Malgré une bonne précision, si aucun diagnostic n'est effectué avant les maintenances, un faux positif précoce conduira à une maintenance inutile.*

Les métriques d'évaluation de la tâche d'apprentissage (à moins d'un score parfait), en plus d'être spécifique à la formulation du problème, ne sont pas fiables pour estimer la performance du modèle sur l'application. Elles ne prennent pas en compte les particularités du processus de maintenance, ni les coûts qui y sont associés. Dans la prochaine section, nous proposons un protocole adapté aux problématiques identifiés dans cette section, ainsi qu'une métrique permettant d'estimer le coût d'un système de maintenance prédictive.

4 Protocole expérimental pour la maintenance prédictive

Dans la suite, pour définir nos solutions aux problématiques introduites dans la section précédente, nous nous plaçons dans le cadre suivant :

— On souhaite prédire l'occurrence d'une panne, sans chercher à déterminer la partie la provoquant
— Une procédure de diagnostic permet de détecter les faux positifs. On supposera que les signes de dégradation peuvent être identifiés par l'équipe technique si on a $\hat{z}_j = z_j = 1$, et qu'une maintenance sera alors effectuée.
— Le diagnostic provoque un arrêt de la machine.
— Il n'y a pas de dérive conceptuelle pour la prédiction des pannes.

Nous pensons que notre protocole reste facilement adaptable à des hypothèses différentes. Au fur et à mesure de nos définitions, nous fournirons des exemples de modifications possibles pour s'adapter à d'autres hypothèses.

Dans le cadre où on souhaite prédire les pannes indépendamment des parties les provoquant, on a en entrée du protocole un ensemble de cycles de vie $\mathcal{X} = \{X_1, \ldots, X_n\}$ avec $X_i = \{x_1, \ldots, x_m\}$ un cycle de vie de taille m. Si les parties fautives devaient être identifiées, on aurait également un vecteur de classe $Y = \{y_1, \ldots, y_n\}$, contenant la partie défaillante à la fin de chaque cycle. Pour un modèle de classification, les fenêtres glissantes extraites de X_i qui avait, dans le cas binaire, la classe 1, prendrait alors la classe y_i.

Puisqu'on considère qu'il n'y a pas de dérive conceptuelle, on peut conserver l'utilisation d'une validation croisée, où $\mathcal{X}$ est divisé en deux ensembles disjoints, $\mathcal{X}_{Train}$ et $\mathcal{X}_{Test}$ à chaque étape de validation. Si une dérive conceptuelle était présente, il faudrait utiliser une validation croisée temporelle, définie par Bergmeir et Benítez (2012), pour tenir compte de l'adaptation du modèle au changement de concepts.

Enfin, pour définir des métriques propres à l'évaluation de l'application, puisqu'on suppose qu'un diagnostic est effectué avant chaque maintenance, on peut prendre en compte tous les faux positifs émis par le modèle lors du calcul, jusqu'à la première occurrence de $\hat{z}_j = z_j = 1$. On rappellera z_j correspond à la vraie classe, et $\hat{z}_j$ à la classe prédite par le modèle. Si une maintenance était effectuée sans vérification préalable, seulement la première alerte ($\hat{z}_j = 1$) de chaque cycle devrait être considéré. En effet, si la maintenance modifie le comportement de la machine, les données situées après j n'ont aucune garantie d'être vues par le modèle lors de l'application sur le cas réel.

4.1 Estimer le coût d'un système de maintenance prédictive

Sachant les contraintes imposées par nos hypothèses sur le processus de maintenance, on souhaite définir une métrique estimant la performance d'un modèle sur l'application. Pour cela, nous proposons une métrique prenant en compte les différents coûts liés au processus de maintenance. Elle permet d'estimer le coût associé aux opérations de maintenance d'un cycle de vie, et, par extension, d'estimer la rentabilité du système de maintenance. Dans de nombreux ouvrages traitant de l'ingénierie de fiabilité, on trouve des listes exhaustives des coûts liés au processus de maintenance (voir Arquès, 2009, Chapitre 1 Section 4) :

— Le coût lié au non-fonctionnement de la machine.
— Le coût de formation du personnel de maintenance
— Le coût du diagnostic, incluant les frais de déplacement de l'équipe technique.
— Le coût des pièces à remplacer lorsqu'elles sont disponibles.
— Si une pièce n'est pas disponible, il faut majorer son coût par celui de l'immobilisation de la machine, car elle n'est plus productive.
— Le coût de la gestion des stocks et de leur immobilisation.
— Les coûts énergétiques liés à la maintenance et au fonctionnement de la machine en fonction de son état (p. ex. un problème d'injection ou d'allumage sur un moteur entraine une surconsommation).

Nous pouvons nous aider de cette liste pour définir notre estimation du coût d'un cycle de vie, en modélisant ces coûts par les variables suivantes :

— t_m : le moment où le cycle de vie se termine.
— t_f : le moment où une alerte est levée dans le cycle de vie.
— $C_{fail}(t)$: le coût associé au non-fonctionnement de la machine et à son immobilisation. Ce coût est fonction du temps pour tenir compte des périodes où la machine devrait être

productive, et des périodes où elle n'est pas utilisée (p. ex. avec un planning hebdomadaire).
— C_{tech} : le coût de déplacement de l'équipe technique et du diagnostic de la machine. On suppose ce coût constant.
— $C_{replace}(t_m - t_f)$: le coût lié à la pièce à remplacer, ainsi qu'à son replacement précoce. Nous supposons que le coût des pièces est fixe, et que le manque à gagner pour un remplacement précoce peut être calculé en multipliant la différence $t_m - t_f$ par une constante.
— Δ_{alert} : le temps écoulé entre une panne, son observation et le départ de l'équipe technique.
— Δ_{tech} : le temps écoulé entre le départ de l'équipe technique, l'arrivée sur site et la réalisation du diagnostic.
— Δ_{repair} : le temps nécessaire à la réalisation de la maintenance.

Par simplicité, nous considérons que l'équipe de maintenance est toujours disponible et que les pièces nécessaires à une maintenance sont toujours en stock. Ainsi, le coût d'une panne non prédite, noté C_{neg}, est égale à :

$$C_{neg} = C_{tech} + C_{replace}(0) + \int_{t_m}^{t_m+\Delta_{alert}+\Delta_{tech}+\Delta_{repair}} C_{fail}(dt) \tag{2}$$

On peut ensuite distinguer deux coûts pour les opérations de maintenance prédictive. Le premier coût est celui d'une alerte justifiée, qui sera confirmée par le diagnostic de l'équipe technique et donnera lieu à une maintenance, noté C_{pos}. Le second coût, est celui d'une alerte non justifiée, qui ne sera pas confirmée par le diagnostic, noté C_{fpos}. Ces deux coûts sont définis tels que :

$$C_{pos} = C_{tech} + C_{replace}(t_m - t_f) + \int_{t_f}^{t_f+\Delta_{tech}+\Delta_{repair}} C_{fail}(dt) \tag{3}$$

$$C_{fpos} = C_{tech} + \int_{t_f}^{t_f+\Delta_{tech}} C_{fail}(dt) \tag{4}$$

L'avantage du système de maintenance prédictive est de pouvoir planifier des interventions pendant des périodes où $C_{fail}(t)$ est minimum, afin de minimiser les pertes de production. Considérons un cycle de vie $X_i = \{x_1, \ldots, x_m\}$ et les fenêtres glissantes de taille l préalablement extraites de X_i. À partir d'un modèle prédictif, on obtient un vecteur $\hat{Z}_i = \{\hat{z}_1, \ldots, \hat{z}_{m-(l-1)}\}$, contenant les prédictions pour chaque fenêtre. Le coût C_{X_i} de ce cycle de vie est alors défini tel que :

$$C_{X_i} = \sum_{j=1}^{m-(l-1)} \begin{cases} C_{pos} & \text{si } \hat{z}_j = z_j = 1\text{, ensuite on arrete la somme} \\ C_{fpos} & \text{si } \hat{z}_j = 1 \text{ et } z_j = 0 \end{cases} \tag{5}$$

Si aucune maintenance n'a été effectué pour le cycle X_i (c.-à-d. aucune occurrence de $\hat{z}_j = z_j = 1$), on affectera alors $C_{X_i} = C_{X_i} + C_{neg}$. Le coût global du système de maintenance

prédictive peut alors être comparé à système de maintenance réactive, qui intervient quand les pannes sont constatées :

$$C_{global} = \sum_{i=1}^{n} (C_{neg_i} - C_{X_i}) \tag{6}$$

Une valeur positive de C_{global} indiquera que le système se basant sur le modèle prédictif en entrée de notre protocole est rentable par rapport au système contre lequel il est comparé. À l'inverse, une valeur négative indiquera que le modèle ne permet pas de réaliser des économies. On peut aussi remplacer C_{neg_i} par le résultat obtenu avec un autre modèle, pour comparer différents modèles prédictifs.

Dans le cadre où ces coûts ne sont ni connus ni estimables, on peut remplacer chaque partie des équations de $C_{neg}, C_{pos}, C_{fpos}$ par des constantes. Une stratégie possible pour fixer les valeurs de ces constantes est de considérer des coûts relatifs. Par exemple, considérons λ_{neg}, la constante remplaçant l'intégrale de C_{neg}. Une valeur $\lambda_{neg} = 1$ donne alors le coût lié au non-fonctionnement de la machine pour une panne non prévue. Si on suppose que le coût de non-fonctionnement peut être réduit de moitié pour une maintenance prévue, on fixera alors $\lambda_{pos} = 0.5$ pour remplacer l'intégrale de C_{pos}. En fonction de la complexité du processus de diagnostic, on pourra par exemple fixer $\lambda_{fpos} = 0.1$ pour remplacer l'intégrale de C_{fpos}. On pourra fixer de la même manière les valeurs de C_{tech} et de la constante à multiplier par $t_m - t_f$ dans $C_{replace}$.

Si les coûts ne sont pas connus, il faudra veiller à présenter les résultats avec plusieurs configurations pour les valeurs des constantes de remplacement. L'objectif n'étant évidemment pas de trouver la configuration qui maximise C_{global}, mais de comprendre dans quelles configurations le modèle pourrait être bénéfique.

5 Conclusion et perspectives

Dans cet article, nous avons introduit le domaine de la maintenance prédictive et ainsi qu'une approche pour construire un jeu de donnée de maintenance prédictive à partir des données émises par une machine. Nous avons ensuite présenté une formulation de la tâche d'apprentissage pour les problèmes de régression et de classification. Après un rapide constat des protocoles expérimentaux utilisés dans la littérature et des problèmes qu'ils peuvent créer, nous avons défini un protocole expérimental adaptable aux contraintes imposées par le processus de maintenance. Enfin, nous avons introduit une métrique, permettant d'estimer le coût d'un système de maintenance prédictive.

Pour définir notre protocole, nous avons assumé un bon nombre d'hypothèses sur la formulation du problème, le traitement des données brutes et le type de modèles utilisé. Notre premier objectif sera d'étudier la capacité de notre protocole à s'adapter aux autres formulations du problème, et aux autres méthodes de prétraitements des données. L'idée étant que, indépendamment de l'approche, on peut calculer la métrique de coût en fonction des alertes émises par un modèle. Il faudra ensuite vérifier empiriquement l'intérêt de ce protocole sur des cas concrets.

À plus long terme, notre objectif est de pouvoir comparer équitablement une grande variété d'approches. Nous souhaitons mettre à disposition de la communauté un outil permettant

de comparer équitablement un modèle aux autres approches existantes, notamment en proposant une interface pour évaluer un modèle sur des jeux de données publics de maintenance prédictive.

Références

Arquès, P. (2009). *Diagnostic prédictif et défaillances des machines : théorie, traitement, analyse, reconnaissance, prédiction*. Editions Technip.

Bampoula, X., G. Siaterlis, N. Nikolakis, et K. Alexopoulos (2021). A deep learning model for predictive maintenance in cyber-physical production systems using lstm autoencoders. *Sensors 21*(3).

Bergmeir, C. et J. M. Benítez (2012). On the use of cross-validation for time series predictor evaluation. *Information Sciences 191*, 192–213.

Esteban, A., A. Zafra, et S. Ventura (2022). Data mining in predictive maintenance systems : A taxonomy and systematic review. *WIREs Data Mining and Knowledge Discovery 12*(5), e1471.

Gutschi, C., N. Furian, J. Suschnigg, D. Neubacher, et S. Voessner (2019). Log-based predictive maintenance in discrete parts manufacturing. *Procedia CIRP 79*, 528–533. 12th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 18-20 July 2018, Gulf of Naples, Italy.

O'Connor, P. et A. Kleyner (2012). *Practical Reliability Engineering, Fifth Edition*.

Orhan, S., N. Aktürk, et V. Çelik (2006). Vibration monitoring for defect diagnosis of rolling element bearings as a predictive maintenance tool : Comprehensive case studies. *NDT & E International 39*(4), 293 – 298.

Saxena, A., K. Goebel, D. Simon, et N. Eklund (2008). Damage propagation modeling for aircraft engine run-to-failure simulation. *International Conference on Prognostics and Health Management*.

Schlagenhauf, T. et N. Burghardt (2021). Intelligent vision based wear forecasting on surfaces of machine tool elements. *SN Applied Sciences 3*(12), 1–13.

Sipos, R., D. Fradkin, F. Moerchen, et Z. Wang (2014). Log-based predictive maintenance. pp. 1867–1876. Association for Computing Machinery.

Vallim Filho, A. R. d. A., D. Farina Moraes, M. V. Bhering de Aguiar Vallim, L. Santos da Silva, et L. A. da Silva (2022). A machine learning modeling framework for predictive maintenance based on equipment load cycle : An application in a real world case. *Energies 15*(10).

Wang, J., C. Li, S. Han, S. Sarkar, et X. Zhou (2017). Predictive maintenance based on event-log analysis : A case study. *IBM J. Res. Dev. 61*.

Zschech, P., J. Bernien, et K. Heinrich (2019). Towards a taxonomic benchmarking framework for predictive maintenance : The case of nasa's turbofan degradation.

Summary

Industry 4.0 and predictive maintenance are hot topics, industrial applications are numerous, as well as articles presenting predictive models to solve specific use-cases. However, there is a lack of work dealing with issues related to the evaluation and comparison of these models. This can lead to the use of experimental protocols and metrics, which are not suitable for predictive maintenance. Also, the comparison between different types of predictive models (e.g. regression and classification) is not systematic, as the metrics used are different. In this position paper, we wish to draw the attention of the community to the problematics behind the evaluation of predictive maintenance models. First, we introduce the field of predictive maintenance as well as a commonly used formulation for the learning task. Then, we show the flaws of classical experimental protocols, and propose an alternative for evaluating and comparing models according to the maintenance process associated with the application.

Modeling and Management of Spatio-temporal Uncertainty of Flood Events

Manel Chehibi*, Ahlem Ferchichi*,**
Imed Riadh Farah*

*RIADI Laboratory, National School of Computer Science, University of Manouba,
Manouba University campus, 2010 Manouba, Tunisia
webmaster@ensi.rnu.tn,
https://ensi.rnu.tn/
**University of Ha'il, Hail, Saudi Arabia
info@uoh.edu.sa
https://www.uoh.edu.sa/en

Abstract. Today, one of the most common natural hazards in the world is flooding, and over the years flooding has caused significant loss of life and property damage. Remote sensing technology and data derived from satellite imagery are useful for knowing the extent of flood, which is useful for flood risk management. An important prerequisite for flood risk management is the existence of spatial and temporal information on the extent of the flood. In general, this spatio-temporal information from remote sensing data is uncertain. The objective of our work is to model the spatial and temporal uncertainties relating to the date and extent of floods, in order to provide information and advice to the right measurements to adapt to flood problems. The estimate of the date and extent of the flood is based on the analysis of the extents of other floods that have occurred in the same area. There is always a level of spatial-temporal uncertainty inherent in such estimates.

1 Introduction

Today, one of the most common natural hazards in the world is flooding (Ha et al., 2021), and over the years flooding has caused significant loss of life and property damage (Cai et al., 2021). In 2019, 361 events occurred worldwide, of which flooding was the largest event with a total of 170 incidents, representing 47% of the total (Miau and Hung, 2020). These events affected around 3 billion people and caused 5100 deaths (Miau and Hung, 2020). These floods are caused by many factors such as climate change (Zhang et al., 2021), the socio-economic factor, geology, topography (Das, 2020), land use change (Hu et al., 2020) and spatial and temporal variabilities (Merwade et al., 2008) which become major factors of uncertainty in the management of flood risks.

Remote sensing technology and data derived from satellite imagery are useful for mapping flooded areas (Shen et al., 2019), which is useful for flood risk management (Moreira et al., 2021). Due to their wide spatial coverage and their great temporal availability, they can

facilitate this type of spatio-temporal analysis. An important prerequisite for flood risk management is the existence of spatial and temporal information on the actual extent of the flood (Kurte et al., 2019). In general, this spatio-temporal information from remote sensing data is uncertain.

The main sources of uncertainty in spatio-temporal information are: cloud cover during periods of heavy flooding, mixed pixel and image quality, sub-optimal solar lighting, spatial and temporal resolution. Many of the probabilistic and non-probabilistic methods have been used for the modeling and management of uncertainties such as interval theory, fuzzy set theory (Goguen, 1973), probability theory, possibility theory (Zadeh, 1978; Dubois, 1988) and belief function theory (Dempster, 1967; Shafer, 1976).

The objective of our work is to model the spatial and temporal uncertainties using the theory of belief functions to detect the extent of flooding, in order to provide information and advice to the right measurements for adapt to flood problems.

This paper is structured as follow: Section 2 recalls the main concepts of belief functions theory, Allen's relations and the Region Connection Calculus 8 (RCC-8), Section 3 details our proposed approach for managing uncertain and imprecise spatio-temporal information, Section 4 presents an experimental study of our approach before concluding in section 5.

2 Background

In this section, we give a brief recall on the theory of belief functions (This section is mainly taken from the article (Chehibi et al., 2018)), and on the qualitative relations.

2.1 Theory of belief functions

The theory of belief functions, also called Dempster-Shafer theory, was first introduced by Dempster (Dempster, 1967) and mathematically formalized by Shafer (Shafer, 1976). This theory models imprecise, uncertain and missing data.

In the theory of belief functions, a *frame of discernment*, noted $\Theta = \{H_1, ..., H_N\}$, is a set of N exhaustive and mutually exclusive hypotheses $H_i, 1 \leq i \leq N$. only one of them is likely to be true.

The *power set*, $2^\Theta = \{A/A \subseteq \Theta\} = \{\emptyset, H_1, ..., H_N, H_1 \cup H_2, ..., \Theta\}$, enumerates 2^N sub-assemblies of Θ. It includes not only hypotheses of Θ, but also, disjunctions of these hypotheses.

The true hypothesis in Θ is unknown; thus, a degree of belief is assessed to subsets of 2^Θ reflecting our degree of faith on the truth of each subset of 2^Θ.

A *basic belief assignment (bba)*, also called *mass function*, is noted m^Θ and defined such that:

$$\begin{aligned} & m^\Theta : 2^\Theta \rightarrow [0,1] \\ & m^\Theta(\emptyset) = 0 \\ & \sum_{A \subseteq \Theta} m(A) = 1 \end{aligned} \tag{1}$$

The mass $m^{\Theta}(A)$ represents the degree of belief on the truth of $A \in 2^{\Theta}$. When $m^{\Theta}(A) > 0$, A is called *focal element*.

2.2 Qualitative relations

2.2.1 Allen's interval algebra

Allen's interval algebra (Allen, 1983) is one of the most known and used formalisms in temporal reasoning. A significant part of the work on temporal representation and reasoning is concerned with time intervals. It is an algebra based on 13 primitive and mutually exclusive relations that can be applied between two time intervals $A = [a, a^{'}]$ and $B = [b, b^{'}]$. These relationships are: before (b), after (bi), meets (m), met by (mi), overlaps (o), overlapped by (oi), starts (s), started by (si), during (d), contains (di), finishes (f), finished by (fi) and equals (e). Each of these relations corresponds to a particular order of the four bounds of the two intervals. For example, the statement A overlaps B (A o B) corresponds to $(a < b) \wedge (b < a^{'}) \wedge (a^{'} < b^{'})$.

2.2.2 The Region Connection Calculus 8 (RCC-8)

Region connection calculus 8 (RCC-8) (Randell et al., 1992) is one of the most known and used formalisms in spatial representation and reasoning developed by Randell, Cui, and Cohn. A significant part of the work on qualitative spatial reasoning (QSR) is concerned with the RCC-8 model. This model describes the possible spatial relations between two spatial regions in the form of eight basic topological relations. These relationships are: DC(a,b) (a is disconnected from b), EC(a,b) (a is externally connected with b), PO(a,b) (a partially overlaps b), TPP(a,b) (a is a tangential proper part of b), NTPP(a,b) (a is a nontangential proper part of b), TPPi(a,b) (a has a tangential proper part b), NTPPi(a,b) (a has nontangential proper part b), EQ (a,b) (a is equal to b).

3 Proposed approach

Our proposed approach includes four phases: the uncertainty representation phase, the modeling of uncertain relationships between events, the measurement of similarity and the aggregation of events phase.

3.1 Representing of Uncertainty

In this section, the uncertainty of spatial, temporal and spatiotemporal events is represented and modeled using intervals. The quantification of this uncertainty is based on the theory of belief functions.

3.1.1 Spatial uncertainty

Spatial uncertainty S_U refers to the uncertainty of the extent of the flood. It is due to several possible flood extents. This uncertainty is represented using an interval-based method. This representation (fig 1) consists of a less uncertain inner interval S_{LU} of the extent of the flood and a more uncertain outer interval S_{MU}. The degrees of uncertainty of the two intervals are

modeled by a mass function m where:
$S_U = \{S_{MU} = [S_{MU1}, S_{MU2}], m(S_{MU}); S_{LU} = [S_{LU1}, S_{LU2}], m(S_{LU})\}$
with $S_{LU} \subseteq S_{MU}$ and $m(S_{MU}) = \bar{m}(S_{LU}) = 1 - m(S_{LU})$

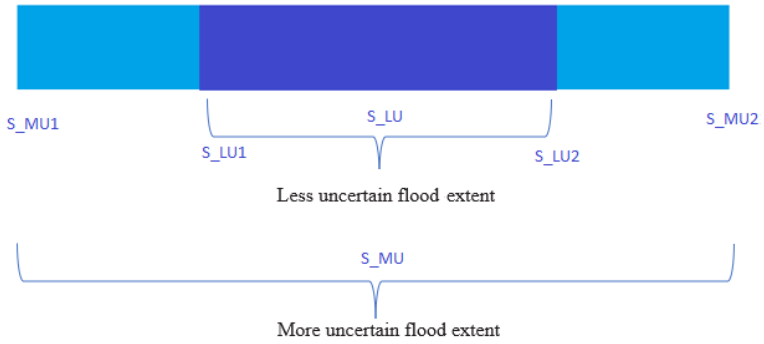


FIG. 1 – *Spatial uncertainty*

3.1.2 Temporal uncertainty

Temporal uncertainty T_U refers to the uncertainty of the date of the flood. This is due to several possible dates of the flood. This uncertainty is represented using an interval-based method. This representation (fig 2) consists of a less uncertain inner interval T_{LU} of the date of the flood and a more uncertain outer interval T_{MU}. The degrees of uncertainty of the two intervals are modeled by a mass function m where:
$T_U = \{T_{MU} = [T_{MU1}, T_{MU2}], m(T_{MU}); T_{LU} = [T_{LU1}, T_{LU2}], m(T_{LU})\}$
with $T_{LU} \subseteq T_{MU}$ and $m(T_{MU}) = \bar{m}(T_{LU}) = 1 - m(T_{LU})$

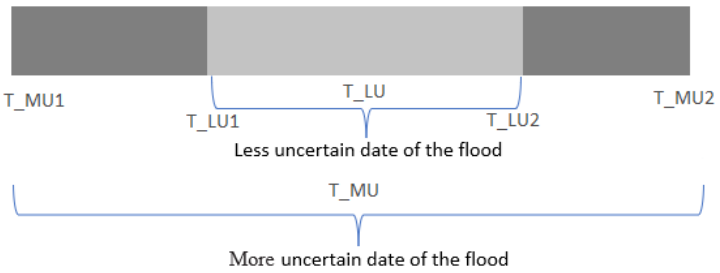


FIG. 2 – *Temporal uncertainty*

3.1.3 Spatio-temporal uncertainty

Spatio-temporal uncertainty ST_U refers to the uncertainty of the date and extent of the flood. This is due to several possible dates and extents of the flood. This uncertainty is represented using an interval-based method. This representation (fig 3) consists of a less uncertain inner interval ST_{LU} of the date and extent of the flood and a more uncertain outer interval ST_{MU}. The degrees of uncertainty of intervals are modeled by a mass function m where:
$ST_U = S_U \otimes T_U = \{ST_{LU}; ST_{MU}; ST_{LMU}; ST_{MLU}\}$

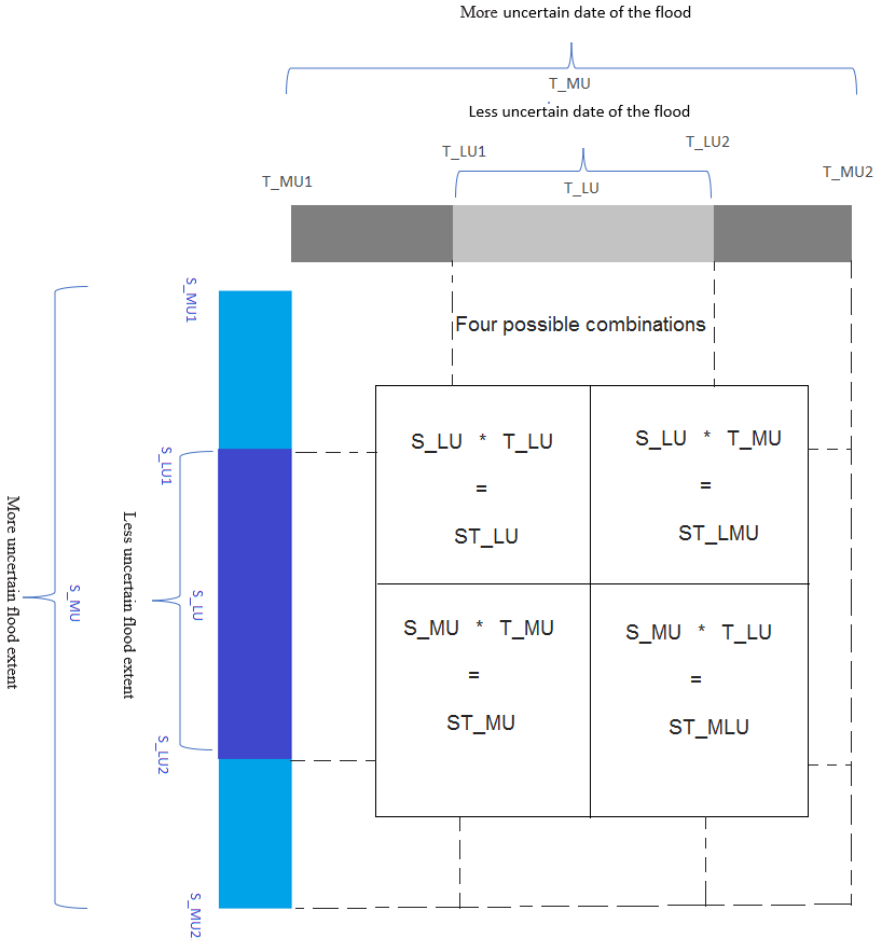


FIG. 3 – *Spatio-temporal uncertainty*

with $ST_{LU} = \{(S_{LU} \otimes T_{LU}); m(ST_{LU}) = m(S_{LU}) * m(T_{LU})\}$
$ST_{MU} = \{S_{MU} \otimes T_{MU};\ m(ST_{MU}) = m(S_{MU}) * m(T_{MU})\}$

TAB. 1 – *Spatio-temporal relations*

Temporal relation	**Spatial relation**	**Spatio-temporal relation**
before (b), after (bi)	Disconnected (DC)	Disconnected (DC)
meets (m), met by (mi)	Externally connected (EC)	Externally connected (EC)
overlaps (o), overlapped by (oi)	Partially overlaps (PO)	Partially overlaps (PO)
starts (s), finishes (f)	is a tangential proper part (TPP)	is a tangential proper part (TPP)
started by (si), finished by (fi)	has a tangential proper part (TPPi)	has a tangential proper part (TPPi)
during (d)	is a nontangential proper part (NTPP)	is a nontangential proper part (NTPP)
contains (di)	has a nontangential proper part (NTPPi)	has a nontangential proper part (NTPPi)
equal(e)	equal (EQ)	equal (EQ)

$ST_{LMU} = \{S_{LU} \otimes T_{MU},\ m(ST_{LMU}) = m(S_{LU}) * m(T_{MU})\}$
$ST_{MLU} = \{S_{MU} \otimes T_{LU},\ m(ST_{MLU}) = m(S_{MU}) * m(T_{LU})\}$

3.2 Modeling the uncertain relationships of flood events

The nature of the relationship between two flood events depends on the nature of their spatial and temporal relationships. Uncertain relationships between spatial intervals representing flood extent are modeled using the RCC-8 model and uncertain relationships between temporal intervals representing date/time of flooding are modeled using Allen's interval algebra. The possible relations between intervals are deduced from their structure and the corresponding mass values. Table 1 presents some possible spatio-temporal relationships that may exist between two flood events and which may be useful for the analysis and evaluation of flood events. These spatio-temporal relationships are represented using RCC-8 model.

For simplicity, we assume as example that $ST1_U$ is a fully uncertain flood event (uncertain spatio-temporal event) and $ST2$ is a definite flood event.
$ST1_U = \{S1_U \otimes T1_U\}$
where $S1_U = \{S1_{MU} = [E3, E6],\ m(S1_{MU}),\ S1_{LU} = [E4, E5];\ m(S1_{LU})\}$;
and $T1_U = \{T1_{MU} = [D3, D6];\ m(T1_{MU});\ T1_{LU} = [D4, D5];\ m(T1_{LU})\}$.

$ST2 = \{S2 \otimes T2\}$
where $S2 = [E1, E2];\ m(S2) = 1$;
and $T2 = [D1, D2];\ m(T2) = 1$.

With: $D2 > D3$ / $D2 < D4$ and $E2 > E3$ / $E2 < E4$

Thus, the uncertain spatio-temporal relationship between the two flood events is:

$PO(ST1_U, ST2),\ m(ST1_{MU})$
$DC(ST1_U, ST2),\ m(ST1_{LU})\}$

3.3 Similarity of flood events

In many applications especially for flood risk management, comparing the similarity of spatio-temporal events can help in making a judgment or decision. Also, if two events are similar, it will be very useful to merge them and then have more reliable information. We propose here our method of measuring the similarity of flood events. This method is based on their spatio-temporal relationships and the masses of beliefs of the intervals corresponding to these events. Let $I1$ and $I2$ be two intervals, and any intersection between them is the interval INT. We base the similarity measure on the relationship between the length of INT and the length of $I1$ and $I2$. We therefore have for $Sim(I1, I2)$
$Sim(I1, I2) = (|INT|/|I1| + |INT|/|I2|)/2$
Here, we are interested in the evaluation of the similarity between two uncertain Flood events, $ST1_U$ and $ST2_U$. Since the inner intervals of events are more certain, we can rely on their intersection to determine the degree of similarity between events using the following rule:
$\{Sim(ST1_{LU}, ST2_{LU}) = (Sim(S1_{LU}, S2_{LU}) + Sim(T1_{LU}, T2_{LU}))/2\}$
The outer intervals can also be taken into account to determine the similarity of events, but as a secondary factor since they are less certain. It should be noted that if we have a strong belief in inner intervals, we can overlook the external intervals' similarity and say that E1 and E2 are thought to be quite comparable events.
For some types of relationships between flood events, the similarity value can be calculated without measuring the length of the intersection between the intervals, for example:
1) $TPP(S1_{LU}, S2_{LU})$ or $NTPP(S1_{LU}, S2_{LU})$:
$|INT| = |S1_{LU}|\ and\ Sim(S1_{LU}, S2_{LU}) = (|S1_{LU}|/|S1_{LU}| + |S1_{LU}|/|S2_{LU}|)/2 = (1 + |S1_{LU}|/|S2_{LU}|)/2$
2) $TPPi(S1_{LU}, S2_{LU})$ or $NTPPi(S1_{LU}, S2_{LU})$:
$|INT| = |S2_{LU}|\ and\ Sim(S1_{LU}, S2_{LU}) = (|S2_{LU}|/|S1_{LU}| + |S2_{LU}|/|S2_{LU}|)/2 = (1 + |S2_{LU}|/|S1_{LU}|)/2$
Also, depending on the nature of the relationship between two events, the similarity value can be inferred directly:
1)$DC(S1_{LU}, S2_{LU})$ So $Sim(S1_{LU}, S2_{LU}) = 0$
2)$EQ(S1_{LU}, S2_{LU})$ So $Sim(S1_{LU}, S2_{LU}) = 1$

3.4 Aggregation of events

If the events' similarity is assessed and they appear to be considerably similar, then a merger or combination of events could be considered. As a result, we obtain the aggregated event $ST12_U$. The merge operation is performed by applying the operator max on the upper limits of the inner and outer intervals of the two events and min on the lower limits. The mass of a combined interval is equal to the mass of the first interval multiplied by the mass of the second interval. To satisfy the condition of sum of belief masses, it is necessary to normalize them to be equal to 1.
For example, let:
$ST1_U = \{S1_{MU} = [170, 300],\ m(S1_{MU});\ S1_{LU} = [200, 280],\ m(S1_{LU});$

$T1_{MU} = [10, 20],\ m(T1_{MU});\ T1_{LU} = [13, 19],\ m(T1_{LU})\}$ and
$ST2_U = \{S2_{MU} = [190, 350],\ m(S2_{MU});\ S2_{LU} = [220, 330],\ m(S2_{LU});$
$T2_{MU} = [14, 25],\ m(T2_{MU});\ T2_{LU} = [17, 22],\ m(T2_{LU})\}$
therefore:
$S12_{MU} = [min[170, 190],\ max[300, 350]];\ m(S12_{MU}) = m(S1_{MU}) * m(S2_{MU})$
$S12_{LU} = [min[200, 220],\ max[280, 300]];\ m(S12_{LU}) = m(S1_{LU}) * m(S2_{MU})$
$T12_{MU} = [min[10, 24],\ max[20, 25]];\ m(T12_{MU}) = m(T1_{MU}) * m(T2_{MU})$
$T12_{LU} = [min[13, 17],\ max[19, 22]];\ m(T12_{LU}) = m(T1_{LU}) * m(T2_{MU})$
Then:
$S12_{MU} = [170, 350];\ m(S12_{MU})$
$S12_{LU} = [200, 330];\ m(S12_{LU})$
$T12_{MU} = [10, 25];\ m(T12_{MU})$
$T12_{LU} = [13, 22];\ m(T12_{LU})$
Then:
$ST1_U = S12_{MU} = [170, 350];\ m_{norm}(S12_{MU}), S12_{LU} = [200, 330];\ m_{norm}(S12_{LU});$
$T12_{MU} = [10, 25];\ m_{norm}(T12_{MU}), T12_{LU} = [13, 22];\ m_{norm}(T12_{LU})$

4 Experiments

Our area of interest is Chad. In fact, this country suffers from frequent floods. These floods cause displacement of residents and loss of life and material damage in several areas, especially those located on the shores of Lake Chad Or those crossed by the Logone or Chari rivers.

Floods are uncertain spatio-temporal events. Estimating when the flood occurred and the area affected by the flood (the extent of the flood) is not obvious enough. This is why we use Remote sensing technology and data derived from satellite imagery.

The extent of the flood is created by detecting changes in Sentinel-1 (SAR) data. To do this, we use different satellite images. These images are obtained by determining time intervals and not time points before and after the flood. In fact, this allows selecting a sufficient number of tiles to cover the area of interest. These intervals are of the form:
$< Before\_start,\ Before\_end;\ After\_start,\ After\_end >$.

In this work we are only interested in the area of the flood extent. At this point, our goal is to estimate the area of flood extent in 2015 based on the information extracted from these images (those from 2016 to 2021). This means that these images will be our source of information.

These information indicated that:
1) The most flood events occurred during the month of August with a mass equal to $0, 57$, and the remaining events occurred during the months of June, July and September with a mass equal to $0, 43$. Thus, August is the most certain time interval for a flood event to occured in 2015, with $m([01/08/2015 - 31/08/2015]) = 0, 57$ and $m([01/06/2015 - 31/09/2015]) = 0, 43$.

2) Most of the floods events affected between 150000 hectares and 450000 hectares with a mass equal to $0,71$, and the remaining events affected either less than 150000 hectares or more than 450000 hectares with a mass equal to $0,29$. Thus, We can estimate the extent of the flood that occurred in 2015 with $m([150000-450000]) = 0.71$ and $m([100000-700000]) = 0,29$.

In this work and in order to have more relevant information on the extent and date of the flooding in 2015, a second source of information will be used. This second source is a database of floods that occurred in Chad between 2012 and 2015.

The information in this database indicated that:
1) The most flood events occurred during the month of August with a mass equal to $0,68$, and the remaining events occurred during the months of July, September and October with a mass equal to $0,32$. Thus, August is the most certain time interval for a flood event to occur, with $m([01/08/2015 - 31/08/2015]) = 0.68$ and $m([01/07/2015 - 31/10/2015]) = 0.32$.
2) Most of the floods affected between 100000 and 500000 hectares, and the remaining events affected either less than 100000 hectares or more than 500000 hectares. Thus, We can estimate the extent of the flood that occurred in 2015 with $m([100000 - 500000]) = 0.53$ and $m([5000 - 800000]) = 0.47$.

So, we now have two spatio-temporal information about the uncertain flood event, provided by two different sources of information. According to the first source of information:
$ST1_U = \{S1_{MU} = [100000 - 700000],\ m(S1_{MU}) = 0,29;$
$S1_{LU} = [150000 - 450000],\ m(S1_{LU}) = 0.71;$
$T1_{MU} = [01/06/2015 - 31/09/2015],\ m(T1_{MU}) = 0,43;$
$T1_{LU} = [01/08/2015 - 31/08/2015],\ m(T1_{LU}) = 0,57\}$
According to the second source of information:
$ST2_U = \{S2_{MU} = [5000 - 800000],\ m(S2_{MU}) = 0.47;$
$S2_{LU} = [100000 - 500000],\ m(S2_{LU}) = 0.53;$
$T2_{MU} = [01/07/2015 - 31/10/2015],\ m(T2_{MU}) = 0.32;$
$T2_{LU} = [01/08/2015 - 31/08/2015],\ m(T2_{LU}) = 0.68\}$

The relationship between the two outer spatial uncertain information is: $NTTP(S1_{MU}, S2_{MU})$
The relationship between the two inner spatial information is: $NTTP(S1_{LU}, S2_{LU})$
Then the relationship between the two spatial information is: $NTTP(S1_U, S2_U)$.
The relationship between the two outer temporal uncertain information is: $T1_{MU}\ O\ T2_{MU}$
The relationship between the two inner temporal uncertain information is: $T1_{LU}\ E\ T2_{LU}$
Then The relationship between the two temporal uncertain information is: $T1_{MU}\ O\ T2_{MU}$ with $m(O(T1_{MU}, T2_{MU})) = 0,43$ or $T1_{LU}\ E\ T2_{LU}$ with $m(O(T1_{MU}, T2_{MU})) = 0,57$

Since, the relationship between the two inner spatial information is: $NTTP(S1_{LU}, S2_{LU})$, then the similarity value can be calculated without measuring the length of the intersection between the intervals:
$Sim(S1_{LU}, S2_{LU}) = (1 + |S1_{LU}|/|S2_{LU}|)/2 = (1 + (450000 - 150000)/(500000 - 100000))/2 = 0.875$
Since, the relationship between the two inner temporal information is: $T1_{LU}\ E\ T2_{LU}$, then the similarity value can be inferred directly:

$Sim(S1_{LU}, S2_{LU}) = 1$
We note that we have a strong belief in inner intervals, so, we can overlook the external intervals' similarity and say that $ST1_U$ and $ST2_U$ are thought to be quite comparable events.

Since, the events' similarity is assessed and they appear to be considerably similar, then a merger or combination of events could be considered.
$ST1_U = \{S1_{MU} = [100000{-}700000],\ m(S1_{MU}) = 0,29;\ S1_{LU} = [150000{-}450000],\ m(S1_{LU}) = 0.71;$
$T1_{MU} = [01/06/2015 - 31/09/2015],\ m(T1_{MU}) = 0,43;\ T1_{LU} = [01/08/2015 - 31/08/2015],\ m(T1_{LU}) = 0,57\}$
and
$ST2_U = \{S2_{MU} = [5000{-}800000],\ m(S2_{MU}) = 0.47;\ S2_{LU} = [100000{-}500000],\ m(S2_{LU}) = 0.53;$
$T2_{MU} = [01/07/2015 - 31/10/2015],\ m(T2_{MU}) = 0.32;\ T2_{LU} = [01/08/2015 - 31/08/2015],\ m(T2_{LU}) = 0.68)\}$
therefore:
$S12_{MU} = [min[100000, 5000],\ max[700000, 800000]];\ m(S12_{MU}) = m(S1_{MU}) * m(S2_{MU}) = 0.29 * 0.47$
$S12_{LU} = [min[150000, 100000],\ max[450000, 500000]];\ m(S12_{LU}) = m(S1_{LU}) * m(S2_{MU}) = 0.71 * 0.53$
$T12_{MU} = [min[01/06/2015, 01/07/2015],\ max[31/09/2015, 31/10/2015]];\ m(T12_{MU}) = m(T1_{MU}) * m(T2_{MU}) = 0.43 * 0.32$
$T12_{LU} = [min[01/08/2015, 01/08/2015],\ max[31/08/2015, 31/08/2015]];\ m(T12_{LU}) = m(T1_{LU}) * m(T2_{MU}) = 0.57 * 0.68$
Then:
$S12_{MU} = [5000, 800000];\ m(S12_{MU}) = 0,1363$
$S12_{LU} = [100000, 500000];\ m(S12_{LU}) = 0,3763$
$T12_{MU} = [01/06/2015, 31/10/2015];\ m(T12_{MU}) = 0,1376$
$T12_{LU} = [01/08/2015, 31/08/2015];\ m(T12_{LU}) = 0,3876$
Then:
$ST1_U = \{S12_{MU} = [5000, 800000];\ m_{norm}(S12_{MU}) = 0.38,$
$S12_{LU} = [100000, 500000];\ m_{norm}(S12_{LU}) = 0.62;$
$T12_{MU} = [01/06/2015, 31/10/2015];\ m_{norm}(T12_{MU}) = 0.375,$
$T12_{LU} = [01/08/2015, 31/08/2015];\ m_{norm}(T12_{LU}) = 0,625\}$

The combined spatio-temporal information estimates that the date of the uncertain flood event is during the month of August 2015 with a degree of certainty equal to 0.62 and that the extent of the flood is between 100000 and 500000 hectares with a certainty equal to 0.625. Which is really true, in fact, according to the database, a flood event happened on 30/08/2015 and the extent of this flood is evaluated at 118000 hectares.

5 Conclusions

Since the flood is an event with spatial and temporal uncertainties, we propose in this paper a novel approach based on belief function theory to represent and manage the combined

spatio-temporal uncertainty of a flood. Belief function theory is chosen because it is an ideal solution for modeling and quantifying non-specific uncertainty and because of the combining rules that allow merging information from multiple sources.
In our approach, spatial, temporal and spatio-temporal information are represented with intervals. Each interval consists of a more certain inner part and a more uncertain outer part. In fact, this interval structure provides considerable flexibility for the representation of subjective uncertainty. The degree of the belief on each part is expressed by means of a mass function.
In this work, the relationships between uncertain flood events are deduced based the relationships between spatial and temporal information and their corresponding mass. These relationships are useful for the analysis and evaluation of flood events.
Our proposed approach also provides some similarity measurement rules that allow to compare the similarity of flood events and then help to make a judgment or decision.
If the events' similarity is assessed and information from multiple sources appear to be considerably similar, then a combination of flood events could be considered. The purpose of the combination operation is to obtain more reliable information about the uncertain flood event.
The proposed approach for modeling and managing uncertain floods were conducted at the Chad site. Chad was chosen as our area of interest because of the frequent floods that sweep the country and cause displacement of residents and loss of life and material damage. Sentinel-1 images and information from a database are used in our experiments. The experimental results prove the effectiveness of the proposed approach. It shows also how coupling Dempster-Shafer approach with qualitative relationships offers a useful solution for modeling and managing spatio-temporal uncertainty of flood events.
As future work, first, we plan to more rigorously address the similarity of flood events part. For example, according to our approach, if the relation between two flooding events is disconnected or meets, then their degree of similarity is 0 although the meets relation seems to reflect a small similarity between the events because their distance contrary to the disconnected relationship is zero. This notion of distance is therefore to be taken into consideration for our future work. Then, we plan to extend our proposed approach for the 2-dimensional (2D) problem.

References

Allen, J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM 26*(11), 832–843.

Cai, S., J. Fan, and W. Yang (2021). Flooding risk assessment and analysis based on gis and the tfn-ahp method: a case study of chongqing, china. *Atmosphere 12*(5), 623.

Chehibi, M., M. Chebbah, and A. Martin (2018). Independence of sources in social networks. In *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pp. 418–428. Springer.

Das, S. (2020). Flood susceptibility mapping of the western ghat coastal belt using multi-source geospatial data and analytical hierarchy process (ahp). *Remote Sensing Applications: Society and Environment 20*, 100379.

Dempster, A. P. (1967). The annals of mathematical statistics. *Upper and Lower Probabilities Induced by a Multivalued Mapping 38*, 325–339.

Dubois, D. (1988). Théorie des possibilités; applications a la représentation des connaissances en informatique. Technical report.

Goguen, J. (1973). La zadeh. fuzzy sets. information and control, vol. 8 (1965), pp. 338–353.-la zadeh. similarity relations and fuzzy orderings. information sciences, vol. 3 (1971), pp. 177–200. *The Journal of Symbolic Logic 38*(4), 656–657.

Ha, H., C. Luu, Q. D. Bui, D.-H. Pham, T. Hoang, V.-P. Nguyen, M. T. Vu, and B. T. Pham (2021). Flash flood susceptibility prediction mapping for a road network using hybrid machine learning models. *Natural hazards 109*(1), 1247–1270.

Hu, S., Y. Fan, and T. Zhang (2020). Assessing the effect of land use change on surface runoff in a rapidly urbanized city: A case study of the central area of beijing. *Land 9*(1), 17.

Kurte, K., A. Potnis, and S. Durbha (2019). Semantics-enabled spatio-temporal modeling of earth observation data: An application to flood monitoring. In *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Advances on Resilient and Intelligent Cities*, pp. 41–50.

Merwade, V., F. Olivera, M. Arabi, and S. Edleman (2008). Uncertainty in flood inundation mapping: current issues and future directions. *Journal of Hydrologic Engineering 13*(7), 608–620.

Miau, S. and W.-H. Hung (2020). River flooding forecasting and anomaly detection based on deep learning. *IEEE Access 8*, 198384–198402.

Moreira, L. L., M. M. de Brito, and M. Kobiyama (2021). A systematic review and future prospects of flood vulnerability indices. *Natural Hazards and Earth System Sciences 21*(5), 1513–1530.

Randell, D. A., Z. Cui, and A. G. Cohn (1992). A spatial logic based on regions and connection. *KR 92*, 165–176.

Shafer, G. (1976). *A mathematical theory of evidence*, Volume 42. Princeton university press.

Shen, X., D. Wang, K. Mao, E. Anagnostou, and Y. Hong (2019). Inundation extent mapping by synthetic aperture radar: A review. *Remote Sensing 11*(7), 879.

Zadeh, L. A. (1978). Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems 1*(1), 3–28.

Zhang, Y., Y. Wang, Y. Chen, Y. Xu, G. Zhang, Q. Lin, and R. Luo (2021). Projection of changes in flash flood occurrence under climate change at tourist attractions. *Journal of Hydrology 595*, 126039.

Prise en compte de données séquentielles hétérogènes dans l'apprentissage profond : application aux données de soins intensifs

Mamadou Ben Hamidou CISSOKO*, Nicolas LACHICHE*,Vincent CASTELAIN**

* ICube, Université de Strasbourg, ** Hôpitaux Universitaires de Strasbourg

Résumé. Les données massives des dossiers médicaux électroniques (DME) constituent une opportunité pour l'analyse de la santé à grande échelle. La prédiction des résultats cliniques tels que la durée de séjour ou la mortalité dans les unités de soins intensifs (USI) joue un rôle important dans l'amélioration des performances des systèmes de santé. Une question difficile concerne les séries chronologiques d'événements multivariés contenues dans les DME. Cette question est de savoir comment modéliser correctement les dépendances temporelles entre de nombreux événements cliniques différents. Nous proposons une architecture d'apprentissage profond appelée Time-distributed LSTM (Long Short Term Memory) pour des données USI (TD-LSTM-USI), qui apprend les dépendances temporelles entre les différents événements cliniques afin d'en améliorer le pouvoir prédictif. Nous montrons l'efficacité de l'approche proposée sur deux tâches médicales de référence (mortalité et durée de séjour).

1 Introduction

L'analyse des séries chronologiques cherche à utiliser les données recueillies lors d'observations antérieures pour élaborer un modèle capable de représenter la structure de la série afin de prédire et de classer les événements futurs. En raison de la pertinence de cette compétence prédictive, l'analyse des séries chronologiques est devenue un aspect crucial de la modélisation des données dans divers secteurs, notamment les prévisions financières, les diagnostics médicaux et la surveillance de l'environnement (Mudelsee, 2019).

Ces dernières années, la disponibilité des données médicales a augmenté grâce à l'adoption généralisée des dossiers médicaux électroniques (DME) dans les systèmes d'information des hôpitaux. Dès lors, l'intérêt et la recherche se sont accrus pour l'application d'algorithmes d'apprentissage machine (ML), allant des modèles linéaires simples aux réseaux d'apprentissage profond aux données cliniques, notamment pour l'analyse des données des dossiers médicaux électroniques (DME) des unités de soins intensifs. Compte tenu de la nature des données physiologiques du DME, il existe souvent un grand nombre de valeurs manquantes par manque de collecte. Cet inconvénient limite l'utilisation des modèles ML traditionnels, qui nécessitent principalement des données de séries chronologiques périodiques. Un DME stocke les données relatives aux visites des patients à l'hôpital avec des caractéristiques hétérogènes, notamment les diagnostics, les résultats de laboratoire, les médicaments prescrits, les

procédures, les données d'image ainsi que certaines informations démographiques. Il contient également des notes cliniques rédigées par les praticiens médicaux pour rendre compte de l'état du patient et des événements médicaux survenus pendant son séjour ou sa visite. Les données résultantes comprennent des trajectoires de santé différentes avec des longueurs variables et des périodes différentes entre les observations consécutives pour chaque variable clinique, car toutes les variables cliniques ne sont pas mesurées à intervalles de temps réguliers. Ainsi, les mesures peuvent se produire sporadiquement dans le temps en fonction de l'état sous-jacent du patient. De ce fait les intervalles de temps entre les observations consécutives varient d'un patient à l'autre et même au sein du parcours de santé d'un même patient.

Afin de résoudre ce problème, la plupart des travaux existants ont directement modélisé les observations contenant des valeurs manquantes, par exemple en les transformant en une série temporelle de distributions sur les valeurs possibles. Zheng et al. (2017) ont proposé une méthode générale pour représenter ces séries temporelles d'événements multivariés en séries temporelles régulières non biaisées. Cette méthode est basée sur l'agrégation des mesures en intervalles de temps discrets à chaque épisode de soins pour créer des séries temporelles multivariées avec un intervalle de temps régulier pour des données DME échantillonnées irrégulièrement. A partir de là, un modèle d'inférence est utilisé pour apprendre la caractéristique pour chaque variable médicale à chaque point de temps.

Au fil des ans de nombreuses stratégies basées sur les DME ont été développées allant des procédures statistiques de base pour l'imputation, telles que le zéro, la moyenne, la moyenne mobile, la dernière observation (Kantardzic, 2011; Lipton et al., 2015) à des approches basées sur les réseaux de neurones récurrents ont été utilisées sur les données DME en raison de leurs capacités de traitement des informations séquentielles (Lipton et al., 2015; Purushotham et al., 2018). Cependant, les approches existantes basées sur les réseaux de neurones ne modélisent pas les données par heure mais donnent toutes les données du patient au LSTM en une seule fois. Cela peut facilement conduire à un ajustement excessif des données cliniques multivariées contenant des dépendances temporelles cruciales entre de nombreux événements cliniques. Car la nature séquentielle et la dépendance temporelle ne sont pas prises en compte. Dans les séries chronologiques d'événements multivariés produits à partir des DME, la question relative sur comment modéliser correctement les dépendances temporelles entre de nombreux événements cliniques différents reste d'actualité. Généralement, dans les réponses à cette question, les données sont fournies en une seule fois mais pas par heure. Par exemple, l'administration de norépinéphrine dépend de la présence d'une hypotension (pression artérielle basse) avec un niveau croissant de lactate qui peut être soumis à une intubation récente d'un patient présentant un choc septique. Lee et Hauskrecht (2019) montrent comment la modélisation de la dépendance à court terme peut améliorer la prévisibilité d'événements futurs multivariés mais il s'est avéré que les réseaux neuronaux récurrents peuvent facilement omettre ou atténuer ces informations (dépendances temporelles) lors de leurs calculs dans l'état caché (Pascanu et al., 2013). Par conséquent, dans ce travail, nous abordons le problème mentionné ci-dessus de la modélisation des dépendances à long terme dans les séries chronologiques d'événements cliniques multivariés en proposant une architecture appelée Time-distributed LSTM aux données USI (TD-LSTM-USI). Cette architecture est capable de traiter des séries chronologiques cliniques multivariées complexes avec une plus grande expressivité tout en apprenant des dépendances temporelles entre des nombreux évènements cliniques. Pour évaluer notre modèle, nous utilisons les données cliniques réelles dérivées des DME des patients en soins intensifs

dans la base de données MIMIC-III (Johnson et al., 2016). Nous étudions également l'effet des différentes stratégies d'imputation sur la performance du modèle. Ainsi, les principales contributions de ce travail sont les suivantes :

— Nous proposons un réseau neuronal multimodal capable d'apprendre les dépendances temporelles sur les données cliniques pour améliorer la fiabilité de la prédiction de la mortalité et de la durée de séjour ;
— La structure **Time-distributed** est proposée pour apprendre les dépendances temporelles à court et à long terme entre différentes caractéristiques cliniques. Elle permet d'exploiter pleinement les informations à différentes échelles de temps ;
— Nous avons évalué notre modèle sur des ensembles de données de soins de santé du monde réel et obtenu de meilleurs résultats que l'état de l'art pour les différentes tâches de prédiction en utilisant moins de caractéristiques, validant ainsi l'efficacité de l'approche proposée dans un contexte clinique ;

Cet article est structuré comme suit : la section 2 est relative aux travaux connexes à notre étude. La section 3 décrit les jeux de données, les sujets d'études, le développement du modèle et les méthodes d'évaluation utilisés. Dans la section 4, nous décrivons la base de données MIMIC-III Johnson et al. (2016) et les étapes de prétraitement utilisées pour obtenir les differents ensembles de données de référence. Les différents résultats sont discutés dans la section 5. Enfin, la section 6 conclut cette étude.

2 Travaux connexes

Plusieurs méthodes de prévision et d'imputation ont été proposées au cours de la dernière décennie pour traiter les valeurs manquantes dans des séries chronologiques d'événements multivariés produites à partir de DME échantillonnés de manière irrégulière et/ou éparse. Ainsi de même pour les dépendances temporelles entre différentes caractéristiques cliniques. Ces méthodes peuvent être classées en deux types : (i) représentation temporelle et granularité temporelle, (ii) méthodes d'imputation conventionnelles.

2.1 Représentation temporelle et granularité temporelle

Les séries chronologiques initiales d'événements multivariés basées sur les DME sont constituées d'événements irréguliers enregistrés progressivement. Pour modéliser ces séries temporelles d'événements cliniques multivariés, on peut les transformer en une représentation en temps discret en utilisant une segmentation basée sur une fenêtre (Rajkomar et al., 2018; Lee et Hauskrecht, 2019) mettant en correspondance plusieurs événements qui se produisent dans une fenêtre temporelle donnée dans un vecteur binaire de taille fixe. Lee et Hauskrecht (2021) ont également proposé un cadre dans lequel ils incorporent trois modules de mécanismes différents : abstraction, contexte récent et mémoire de périodicité pour traiter les caractéristiques temporelles des séries chronologiques d'événements cliniques multivariés. Dans leur approche, la séquence d'entrée des événements cliniques est représentée par un vecteur binaire. Ces approches aboutissent à une représentation plus détaillée des états du patient, mais peuvent également conduire à des séquences plus longues et plus éparses, ce qui rend la modélisation de la dépendance des événements beaucoup plus difficile et coûteuse en termes de calcul.

2.2 Méthodes d'imputation conventionnelles

Ces méthodes d'imputation conventionnelles peuvent être classées en deux catégories. La première catégorie comprend les méthodes d'imputation statistique telles que les méthodes d'imputation par la moyenne simple, par la médiane (Acuna et Rodriguez, 2004) et par le ratio. D'autres méthodes statistiques classiques de séries chronologiques tels que la moyenne mobile intégrée autorégressive (ARIMA), qui élimine les parties non stationnaires d'une séquence et l'ajuste à un modèle stationnaire paramétré ont été utilisées. L'une des limites de leur approche de l'utilisation des données chronologiques du DME comme données non chronologiques est la sous-utilisation des données chronologiques du DME.

La deuxième catégorie comprend une série des modèles neuronaux, notamment les réseaux neuronaux récurrents (RNN). Baytas et al. (2017) ont proposé un LSTM sensible au temps (T-LSTM), en modifiant les cellules du LSTM qui ajuste l'état caché en fonction des intervalles de temps irréguliers. Chung et al. (2014) ont utilisé une approche simple pour traiter les valeurs manquantes et les intervalles de temps irréguliers dans les données de séries temporelles régulières en agrégeant les séries temporelles, et en proposant des méthodes d'imputation simples sur ces séries comprenant l'imputation moyenne et le remplissage avant avec les valeurs passées avant de les fournir à un classeur. Lipton et al. (2016) montrent également que la performance de la prévision de la mortalité peut être améliorée en passant des vecteurs de masque (un vecteur binaire pour indiquer si la variable est renseignée ou pas à un instant t) comme caractéristiques supplémentaires à un classeur RNN. Mais ils perdent toujours des informations à grain fin en agrégeant chaque série temporelle en intervalles horaires. Harutyunyan et al. (2019) ont proposé un LSTM qui gère chaque variable séparément avec son vecteur de masque dans une couche LSTM bidirectionnelle indépendante, puis une concaténation de toutes les sorties est effectuée. Ainsi, les informations relatives à une variable unique peuvent être apprises. Cette approche est coûteuse car chaque variable doit être apprise dans un réseau indépendant, ce qui entraîne des calculs supplémentaires.

Notre méthode est conçue pour traiter des séries temporelles multivariées avec des valeurs manquantes massives. Elle peut traiter directement les séries temporelles et les indicateurs binaires pour les valeurs manquantes sans considérer une procédure en deux étapes. Les caractéristiques calculées manuellement des différentes variables ne sont pas nécessaires car elle peut explorer les caractéristiques des différentes variables en même temps automatiquement sans coût de calcul supplémentaire.

3 Matériels et méthodes

Dans cette section, nous présentons le cadre que nous proposons pour la prédiction des données hétérogènes. Nous commençons par présenter la notation et une description des séries temporelles physiologiques, puis nous présentons les modèles utilisés pour les tâches mentionnées ci-dessus.

3.1 Représentation des données & Architecture utilisée

Étant donné une série temporelle multivariée comportant **D** variables physiologiques, y compris les analyses de laboratoire et les signes vitaux, sur **T** points temporels pour chaque sé-

jour (icustay) **n** dans notre cohorte de **N** séjours, nous la désignons comme $\mathbf{X} = \{\mathbf{x}_1,...,\mathbf{x}_t,...,\mathbf{x}_T\}^T \in \mathbb{R}^{T\times D}$, où $x_t \in \mathbb{R}^D$ représente la t-ième observation de toutes les variables observées aux instants $_t$, et x_t^d est le d-ième élément ou variable dans x_t . Dans ce contexte, étant donné que la série temporelle X comprend des valeurs manquantes, nous introduisons le vecteur de masque dans la série temporelle, $\mathbf{M} = \{\mathbf{m}_1,...,\mathbf{m}_t,...,\mathbf{m}_T\}^T \in \mathbb{R}^{T\times D}$,de même taille que X, pour indiquer quelles variables sont observées ou manquantes à l'instant t. En particulier,

$$m_t^d = \begin{cases} 1, & \text{si } x_t^d \text{ , est renseignée} \\ 0, & \text{sinon} \end{cases} \tag{1}$$

Et un ensemble d'observations statiques noté $\mathbf{S} = \{\mathbf{s}_i,..,\mathbf{s}_{i+1}\}$ où $s_i \in \mathbf{N}^S$ est extrait pour chaque séjour. Étant donné un ensemble de données de séries chronologiques cliniques pour **N** séjours, nous définissons notre cohorte de N exemples de sujets comme $\{(X^{(n)}; M^{(n)}; S^{(n)})\}_{n=1}^N$.

La prédiction de la mortalité est un problème de classification binaire. La durée de séjour (LOS) est un problème de régression.

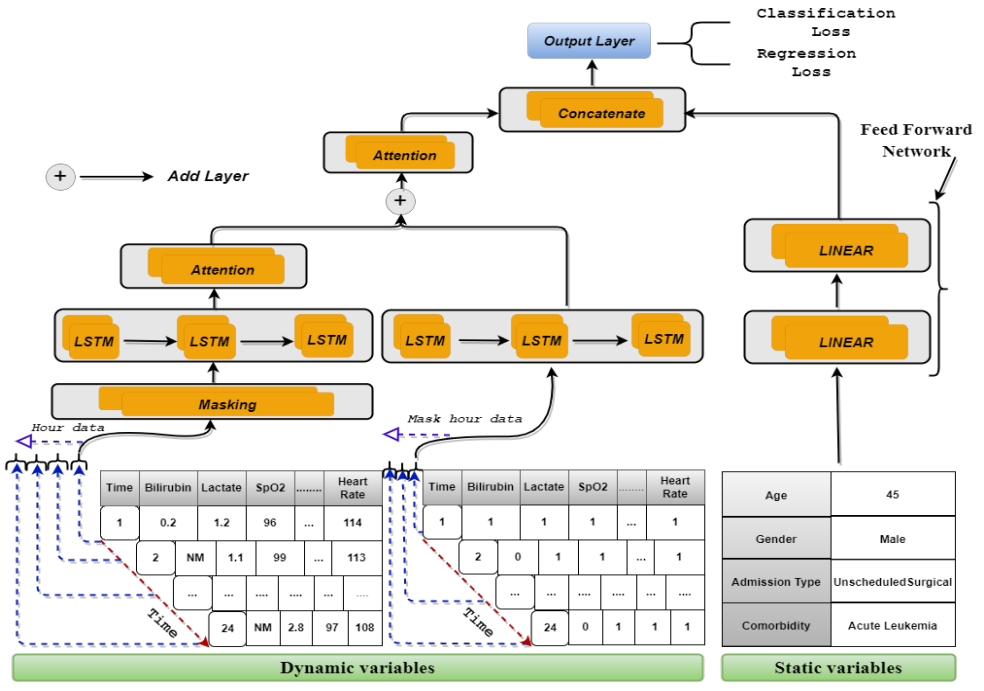


FIG. 1 – *L'architecture globale de notre modèle. La séquence originale d'événements irréguliers est envoyée sous forme des données horaires dans une couche LSTM. Notre modèle apprend les dépendances temporelles entre les différentes caractéristiques cliniques afin d'améliorer la prédiction.*

3.2 Développement du modèle & Mesures d'évaluation

Pour toutes les tâches de prédiction, nous évaluons notre architecture par une stratégie de validation croisée double en 5 plis. Le seuil de classification est réglé sur les données d'entraînement.

Toutes les mesures de performance sont évaluées sur les données de test. Pour les tâches de régression, nous mesurons l'erreur absolue moyenne (MAE) et la racine de l'erreur quadratique moyenne(RMSE), tandis que pour les tâches de classification, nous indiquons la moyenne et l'écart type 5 plis utilisant le F1-Score. Le F1-score est recommandé en présence de classes déséquilibrées. C'est notre cas car le nombre de décès est près de 10% voir tableau 1.

Pour toutes les tâches, nous utilisons la même cohorte et extrayons les données des 24 et 48 premières heures de chaque séjour en soins intensifs . Dans toutes les architectures, nous utilisons la fonction d'activation ReLU pour ajouter de la non-linéarité aux modèles, sauf dans la couche finale. Des lots d'exemples de (n=128) pour les tâches de classification et (n=64) pour les tâches de régression ont été fixés l'entraînement. Pour éviter le surapprentissage, une régularisation par décroissance de poids L_2 est appliquée à toutes les unités de récurrence avec un facteur d'échelle de 0,001. Adam est utilisé comme optimiseur avec un taux d'apprentissage de 10^{-3} et un taux de décroissance de 10^{-4}. Le nombre d'époque est fixée à 100 pour tous les modèles et l'arrêt précoce est utilisé pour éviter le surapprentissage. Nous utilisons également un taux d'abandon de 0,3 tout en éliminant de manière aléatoire les neurones d'entrée du réseau pendant la phase d'apprentissage afin d'éviter un surapprentissage.

4 EXPÉRIMENTATIONS

4.1 La base des données

MIMIC-III (Medical Information Mart for Intensive Care III) (Johnson et al., 2016), est une base de données publique sur les soins intensifs gérée par le Laboratory for Computational Physiology du Massachusetts Institute of Technology (MIT). MIMIC-III comprend des données de santé anonymes associées à 46 520 patients distincts qui ont séjourné dans les différentes unités de soins intensifs du Beth Israel Deaconess Medical Center (BIDMC) entre 2001 et 2012. Les données couvrent 58 976 admissions à l'hôpital, dont 61 532 admissions distinctes dans les différentes unités de soins intensifs. Cette base de données comprend des informations sur la santé des patients telles que les données démographiques, les signes vitaux, les résultats des tests de laboratoire, les médicaments, les codes de diagnostic, ainsi que les notes cliniques.

4.2 Sélection de la cohorte

Pour créer la cohorte correspondante, nous avons d'abord exclu tous les patients âgés de moins de 15 ans et ceux qui ne disposaient d'aucun enregistrement des signaux temporels requis. L'admission d'un patient à l'hôpital peut correspondre à zéro ou plusieurs épisodes de soins intensifs c'est-à-dire transféré d'une unité de soins intensifs à une autre plusieurs fois au cours de son séjour. Dans cet article, nous considérons tous les séjours associés à un patient pendant son admission à l'hôpital. Après avoir appliqué ces critères d'exclusion, nous avons obtenu un total de 53 304 séjours distincts éligibles pour nos différentes tâches de prédiction

avec un âge médian de 65,8 ans (Q1-Q3 : 52,8-77,8). Le taux de mortalité en milieu hospitalier et en unités des soins intensifs étaient respectivement de 12,4% et 8,8%. La durée moyenne de séjour en unités de soins intensifs était de 4,15 jours (plus de détails sont fournis dans le tableau 1).

TABLE 1 Caractéristiques et mesures des résultats en matière de mortalité. Durée du séjour (LOS). Les variables continues sont présentées sous forme de médiane [écart interquartile Q1-Q3] ; les variables binaires ou catégorielles sous forme de nombre (%).

		Overall	Alive at Hospital	Dead at Hospital
ICU ADMISSIONS		53304	46718	6586
AGE, median [Q1,Q3]		65.8 [52.8,77.8]	64.7 [52.0,76.9]	73.4 [60.1,82.8]
GENDER, n (%)	F	23276 (43.7)	20276 (43.4)	3000 (45.6)
	M	30028 (56.3)	26442 (56.6)	3586 (54.4)
ADMISSION TYPE , n (%)	Medical	32578 (61.1)	27729 (59.4)	4849 (73.6)
	ScheduledSurgical	6296 (11.8)	6104 (13.1)	192 (2.9)
	UnscheduledSurgical	14430 (27.1)	12885 (27.6)	1545 (23.5)
Outcomes				
HOSPITAL DEATH, n (%)	Alive	46718 (87.6)	-	-
	Dead	6586 (12.4)	-	-
ICU DEATH, n (%)	Alive	48591 (91.2)	-	-
	Dead	4713 (8.8)	-	-
ICU LOS* (days), median [Q1,Q3]		2.1 [1.2,4.2]	2.1 [1.2,4.0]	3.1 [1.3,7.2]

4.3 Sélection et Extraction des caractéristiques

Les ensembles de données de référence extraits sont traités pour obtenir les caractéristiques qui seront utilisées pour les tâches de prédiction. Nous collectons 4 caractéristiques démographiques statiques (âge, sexe, type d'admission, comorbidité) et 60 mesures de laboratoire et mesures vitales qui varient dans le temps pour chaque séjour du patient.

Nous avons choisi deux ensembles de caractéristiques, comme décrit ci-dessous, pour permettre une étude comparative exhaustive.

— Ensemble de caractéristiques A : Cet ensemble de caractéristiques comprend les 15 caractéristiques utilisées dans le calcul du score SAPS-II (Le Gall et al., 1993). Nous fusionnons les données en fonction des connaissances médicales. Ainsi, pour calculer le score de l'échelle de coma de Glasgow, nous additionnons les valeurs GCSVerbal, GCSMotor et GCSEyes. Pour calculer le débit urinaire, nous additionnons les caractéristiques représentant le débit urinaire. Pour la température corporelle, nous convertissons l'échelle Fahrenheit en Celsius et calculons également le rapport PaO2/FiO2 au lieu de les considérer comme des caractéristiques individuelles.

— Ensemble de caractéristiques B : Cet ensemble de caractéristiques se compose de 66 caractéristiques et comprend les 15 caractéristiques de l'ensemble de caractéristiques A.

Transformation en données de séries chronologiques horaires :
Nous avons transformé chaque caractéristique temporelle en données pour les premières 24 heures et les premières 48 heures de chaque admission en USI. Chaque caractéristique temporelle d'un séjour aux soins intensifs est agrégée en tranches horaires régulièrement espacées (0-1 heure, 1-2 heures, etc.). Certaines caractéristiques peuvent avoir plusieurs enregistrements au cours du processus d'échantillonnage. Comme nous avons besoin d'une valeur représentative pour chaque caractéristique à un pas de temps particulier, nous avons agrégé les enregistre-

ments multiples en fonction de la caractéristique en prenant soit le maximum, soit le minimum ou la somme à un pas de temps particulier. Par exemple, nous additionnons la caractéristique de sortie d'urine et prenons la valeur minimale la pression artérielle moyenne (PAM), GCSVerbal, GCSMotor, et GCSEyes et le maximum pour les autres caractéristiques.

Outre en échantillonnant les données toutes les heures, nous avons constaté que l'absence des valeurs pour les 60 variables de laboratoire et des signes vitaux est supérieur à plus de 93%, car chaque patient peut ne se voir prescrire que quelques tests en fonction de ses besoins médicaux, et ces tests sont peu fréquents dans le temps.

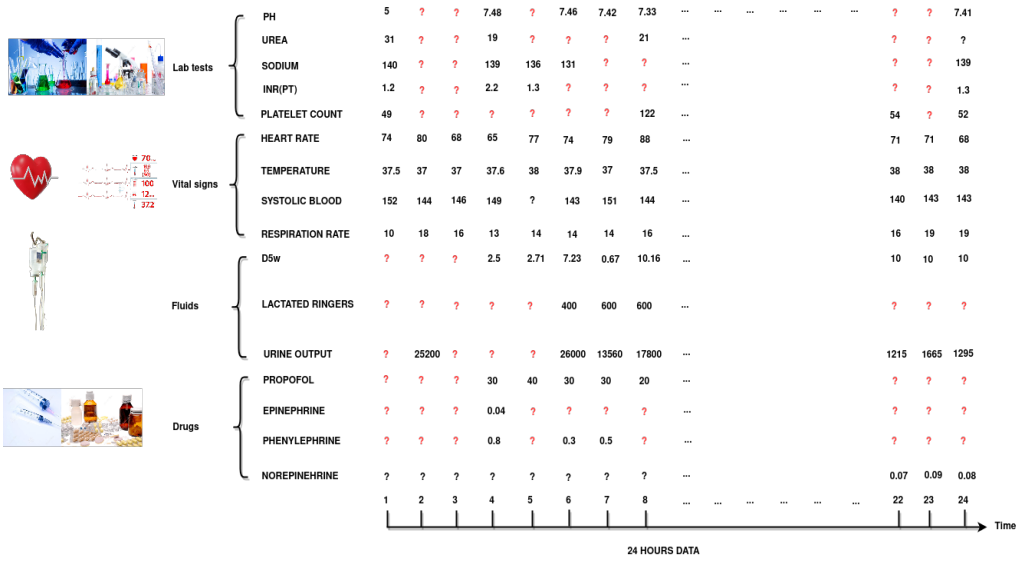


FIG. 2 – *Exemple d'événements temporels hétérogènes*

4.4 Tâches de prédiction

Dans cette partie, nous considérons deux tâches de prédiction de référence en soins intensifs à savoir la prédiction de la mortalité et de la durée du séjour. Plusieurs travaux s'intéressant à ces tâches ont été effectués allant de la prediction de la mortalité des patients atteints de la septicémie ou des patients sous la ventilation mécanique (Purushotham et al., 2018; George et al., 2021; Kong et al., 2020; Rajkomar et al., 2018) pour évaluer les algorithmes d'apprentissage automatique sur les données des DME. Dans tous les travaux listés ci-dessus, les auteurs se sont intéressés à une tâche de prédiction spécifique. Cependant, dans cette étude, nous nous intéressons au tâche de prédiction de la mortalité de façon générale.

1. **Prédiction de la mortalité** : La prédiction de la mortalité est reconnue comme l'un des principaux résultats d'intérêt. L'objectif principal de cette tâche est de prédire si un patient décède pendant son séjour à l'hôpital (milieu hospitalier ou USI).
2. **Prédiction de la durée de séjour** : La durée de séjour des patients est utilisée dans les hôpitaux pour mesurer la gravité de l'état d'un patient ainsi que pour la planifi-

cation et la gestion des ressources. Les patients dont la durée de séjour est prolongée consomment davantage de ressources hospitalières et présentent souvent des pathologies chroniques complexes qui ne mettent pas immédiatement leur vie en danger mais qui sont difficiles à traiter. Pour réduire les dépenses de santé, ces patients doivent être identifiés et traités le plus rapidement possible. La prédiction de la durée de séjour des patients pourrait donc jouer un rôle important dans la gestion des soins de santé.

5 Résultats

Dans cette section, nous évaluons l'architecture TD-LSTM-USI que nous proposons sur des tâches décrites précédemment 4.4 et le comparons au travail de (Song et al., 2018). Nous montrons que notre architecture surpasse le réseau d'interpolation-prédiction proposé par (Song et al., 2018) sur les différentes tâches. Nous nous posons également la question de l'impact de l'utilisation de différents ensembles de caractéristiques et de la durée d'observation sur la performance de prédiction. Nous discutons des différentes mesures d'évaluation utilisées en section 3.2.

5.1 Évaluation de la tâche de prédiction de la mortalité

La table 2 montre les différentes valeurs du F1-score de notre modèle, en utilisant chacune des méthodes d'imputations, sur la tâche de prédiction de la mortalité en milieu hospitalier. La table 3 évalue les mêmes méthodes d'imputation sur la tâche de prédiction de la mortalité en USI. De par ces résultats, plusieurs constats émergent. Le premier constat est que quelle que soit la méthode d'imputation utilisée notre architecture affiche un F1-score supérieur à 50% avec les données de 24 premières heures de séjour. Ce score reste le même avec l'ensemble des caractéristiques A ou B utilisées comme variables d'entrées. Le second constat est que lorsque plus de données sont utilisées par une durée d'observation de 48 heures, la performance du modèle augmente considérablement sur les différentes tâches de prédiction avec un F1-score moyen de plus de 60%. Par ailleurs, nous avons utilisé la méthode du Critical Difference Diagram (Demšar, 2006) afin de comparer par paires les six techniques d'imputations que nous avons utilisées. Les résultats obtenus à travers l'application de cette méthode nous permet d'affirmer qu'il y a une différence significative entre les techniques d'imputations sur les deux tâches de prédiction de la mortalité (figure 3) et que c'est la méthode Forward-Zero-Imputation qui est la meilleure parmi les six utilisées. En conclusion des différentes méthodes d'imputations des valeurs manquantes utilisées, nous pouvons affirmer que le choix de la méthode a un impact significatif sur la performance de notre modèle. Nous avons choisi d'utiliser la méthode Forward-Zero-Imputation par la suite.

Nous avons encore utilisé la méthode du CD pour mieux comparer la performance de notre modèle sur l'utilisation des différentes caractéristiques en l'occurrence l'ensemble A et l'ensemble B. Cette comparaison consiste à utiliser la seule méthode de Forward-Zero-Imputation sur les deux durées d'observation (24h & 48h) et qui sera ensuite utilisé afin de comparer notre modèle par rapport au modèle défini dans (Song et al., 2018). Les résultats (figure 4) montrent que quand plus de caractéristiques et plus de données sont collectées, la performance du modèle s'améliore. Cependant, d'après le diagramme du CD, les quatre types de données sont

TABLE 2 *Mortalité en milieu hospitalier*

IMPUTATION METHODS	24 HOURS DATA		48 HOURS DATA	
	SAPS II FEATURES	ALL FEATURES	SAPS II FEATURES	ALL FEATURES
ZERO-IMPUTATION -WO-M	0.53	0.53	0.58	0.59
MEDIAN-IMPUTATION -WO-M	0.52	0.56	0.56	0.62
ZERO-IMPUTATION + MASKING VECTOR	0.53	0.58	0.58	0.62
ZERO-IMPUTATION	0.54	0.57	0.58	0.62
FORWARD-ZERO-IMPUTATION	0.55	0.58	0.61	0.63
FORWARD-MEDIAN-IMPUTATION	0.53	0.60	0.60	0.62

TABLE 3 Mortalité en USI

IMPUTATION METHODS	24 HOURS DATA		48 HOURS DATA	
	SAPS II FEATURES	ALL FEATURES	SAPS II FEATURES	ALL FEATURES
ZERO-IMPUTATION -WO-M	0.53	0.55	0.58	0.60
MEDIAN-IMPUTATION -WO-M	0.52	0.56	0.58	0.62
ZERO-IMPUTATION + MASKING VECTOR	0.53	0.58	0.61	0.65
ZERO-IMPUTATION	0.52	0.58	0.60	0.65
FORWARD-ZERO-IMPUTATION	0.55	0.59	0.62	0.66
FORWARD-MEDIAN-IMPUTATION	0.51	0.59	0.61	0.66

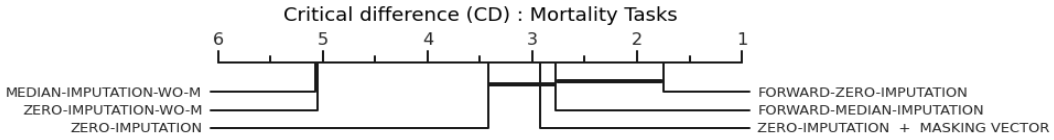


FIG. 3 – *Diagramme de différence critique montrant la comparaison des différentes méthodes d'imputation par paire sur la prédiction de la mortalité sur les données de (24 hrs & 48 hrs) en utilisant les différents ensembles de caractéristiques*

statistiquement différents les uns des autres. Nous avons retenu la représentation qui apporte le plus d'information : l'ensemble de caractéristiques B observées sur une durée de 48 heures.

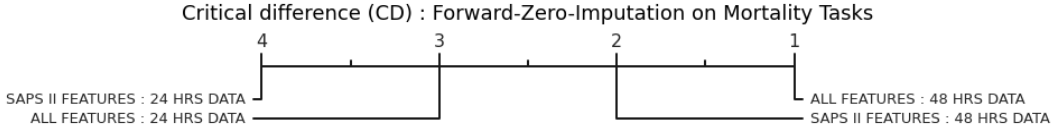


FIG. 4 – *Diagramme de différence critique de la prédiction de la mortalité en utilisant les différents ensembles de caractéristiques (A & B) sur les données de (24 hrs & 48 hrs)*

En dépit de l'évaluation de la capacité de notre modèle en utilisant la méthode du F1-score, nous l'avons aussi comparé avec l'approche proposée dans (Song et al., 2018). Cette

TABLE 4 TD-LSTM-ICU VS SAnD on MORTALITY TASKS : 48 HOURS DATA

TASKS	TD-LSTM-ICU		SAnD	
	SAPS II FEATURES	ALL FEATURES	SAPS II FEATURES	ALL FEATURES
HOSPITAL-MORTALITY	**0.61**	**0.63**	0.59	0.61
ICU-MORTALITY	**0.62**	**0.66**	0.60	0.64

comparaison est faite sur la base des tâches de prédiction de la mortalité en milieu hospitalier et dans les USI. La table 4 affiche les résultats obtenus. Ainsi, nous constatons que notre modèle dépasse celui proposé dans (Song et al., 2018) sur les deux tâches de prédictions de plus de 2%. La supériorité de la performance de notre approche à celle proposée par (Song et al., 2018) peut s'expliquer par le fait que le modèle issu de notre approche parvient à apprendre la dépendance temporelle entre les différents évènements puisque les données sont modélisées par heure, contrairement à beaucoup d'autres architectures (Purushotham et al., 2018) où les données du patient sont fournies en une seule fois.

5.2 Évaluation de la durée de séjour

La prédiction de la durée de séjour nécessite la prise en compte des dépendances temporelles entre les différents évènements cliniques. Notre architecture est capable de capturer cette dépendance de manière efficace et surpasse l'état de l'art en obtenant de meilleures performances en termes d'erreur quadratique moyenne (en jours). La table 5 compare l'erreur absolue moyenne et la racine de l'erreur quadratique des prédictions des durées de séjour prédites par notre approche et par celle de l'état de l'art (Song et al., 2018). Avec les données

TABLE 5 TD-LSTM-ICU VS SAnD on LOS TASK : 48 HRS DATA

TASK : LOS	TD-LSTM-ICU		SAnD	
	SAPS II FEATURES	ALL FEATURES	SAPS II FEATURES	ALL FEATURES
MAE	**1.951 ± 0.057**	**1.925 ± 0.076**	3.177 ± 0.155	3.172 ± 0.15
RMSE	**5.374 ± 0.06**	**5.323 ± 0.078**	6.947 ± 0.088	6.942 ± 0.086

des 48 premières heures et quelque soit les caractéristiques (ensemble A ou B) utilisées notre modèle affiche une performance meilleure que celui de l'état de l'art. Les prédictions de notre modèle se situent dans une fourchette de 5 à 5.5 jours par rapport aux données réelles.

À partir de tous les résultats des tâches de prédiction, nous faisons les observations suivantes : notre architecture est beaucoup plus performante lorsque davantage de caractéristiques sont utilisées pour la prédiction. Cela implique qu'elle peut apprendre de meilleures représentations des caractéristiques à partir de plusieurs modalités de données.

6 Conclusion

Dans cet article, nous avons identifié la nature des irrégularités présentes dans les données DME, tant au niveau temporel qu'au niveau des caractéristiques mesurées. Nous montrons que la capture des dépendances temporelles au niveau des caractéristiques cliniques peut améliorer

les performances de l'analyse des données DME. Pour répondre à cette question, nous avons proposé un modèle basée sur LSTM capable de gérer la dépendance temporelle au niveau de l'irrégularité des caractéristiques cliniques. Il apprend l'impact des informations précédentes sur l'état du patient pour améliorer les informations extraites à différents niveaux de fonctionnalités.

Nous démontrons également que l'approche proposée atteint une meilleure performance dans toutes les tâches que les modèles de l'état de l'art, avec un grand ou un petit nombre de caractéristiques cliniques utilisées en entrée pour les modèles de prédiction.

Nous notons que, comme les données de MIMIC-III sont générées dans un seul système de DME, elles peuvent contenir des biais systématiques. Une étude future intéressante consistera à explorer comment notre modèle formé sur cet ensemble de données se généralise à d'autres ensembles de données cliniques.

Références

Acuna, E. et C. Rodriguez (2004). The treatment of missing values and its effect on classifier accuracy. In *Classification, clustering, and data mining applications*, pp. 639–647. Springer.

Baytas, I. M., C. Xiao, X. Zhang, F. Wang, A. K. Jain, et J. Zhou (2017). Patient subtyping via time-aware lstm networks. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 65–74.

Chung, J., C. Gulcehre, K. Cho, et Y. Bengio (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv :1412.3555*.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine learning research 7*, 1–30.

George, N., E. Moseley, R. Eber, J. Siu, M. Samuel, J. Yam, K. Huang, L. A. Celi, et C. Lindvall (2021). Deep learning to predict long-term mortality in patients requiring 7 days of mechanical ventilation. *PloS one 16*(6), e0253443.

Harutyunyan, H., H. Khachatrian, D. C. Kale, G. Ver Steeg, et A. Galstyan (2019). Multitask learning and benchmarking with clinical time series data. *Scientific data 6*(1), 1–18.

Johnson, A. E., T. J. Pollard, L. Shen, L.-w. H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, et R. G. Mark (2016). Mimic-iii, a freely accessible critical care database. *Scientific data 3*(1), 1–9.

Kantardzic, M. (2011). *Data mining : concepts, models, methods, and algorithms*. John Wiley & Sons.

Kong, G., K. Lin, et Y. Hu (2020). Using machine learning methods to predict in-hospital mortality of sepsis patients in the icu. *BMC medical informatics and decision making 20*(1), 1–10.

Le Gall, J.-R., S. Lemeshow, et F. Saulnier (1993). A new simplified acute physiology score (saps ii) based on a european/north american multicenter study. *Jama 270*(24), 2957–2963.

Lee, J. M. et M. Hauskrecht (2019). Recent context-aware lstm for clinical event time-series prediction. In *Conference on Artificial Intelligence in Medicine in Europe*, pp. 13–23. Springer.

Lee, J. M. et M. Hauskrecht (2021). Modeling multivariate clinical event time-series with recurrent temporal mechanisms. *Artificial intelligence in medicine 112*, 102021.

Lipton, Z. C., D. Kale, et R. Wetzel (2016). Directly modeling missing data in sequences with rnns : Improved classification of clinical time series. In *Machine learning for healthcare conference*, pp. 253–270. PMLR.

Lipton, Z. C., D. C. Kale, C. Elkan, et R. Wetzel (2015). Learning to diagnose with lstm recurrent neural networks. *arXiv preprint arXiv :1511.03677*.

Mudelsee, M. (2019). Trend analysis of climate time series : A review of methods. *Earth-science reviews 190*, 310–322.

Pascanu, R., T. Mikolov, et Y. Bengio (2013). On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pp. 1310–1318. PMLR.

Purushotham, S., C. Meng, Z. Che, et Y. Liu (2018). Benchmarking deep learning models on large healthcare datasets. *Journal of biomedical informatics 83*, 112–134.

Rajkomar, A., E. Oren, K. Chen, A. M. Dai, N. Hajaj, M. Hardt, P. J. Liu, X. Liu, J. Marcus, M. Sun, et al. (2018). Scalable and accurate deep learning with electronic health records. *NPJ digital medicine 1*(1), 1–10.

Song, H., D. Rajan, J. Thiagarajan, et A. Spanias (2018). Attend and diagnose : Clinical time series analysis using attention models. In *Proceedings of the AAAI conference on artificial intelligence*, Volume 32.

Zheng, K., J. Gao, K. Y. Ngiam, B. C. Ooi, et W. L. J. Yip (2017). Resolving the bias in electronic medical records. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 2171–2180.

Summary

The ever-growing mass of electronic health record (EHR) data provides an opportunity for large-scale data-driven health analysis and intelligent medical care. Predicting clinical outcomes such as length of stay or mortality in intensive care units (ICUs) plays an important role in improving the performance of healthcare systems. One difficult issue with predictive multivariate event time-series produced from EHRs that has not been adequately addressed is how to properly model the temporal dependencies between many different clinical events. We propose a deep learning model called Time-distributed LSTM to ICU data (TD-LSTM-ICU), which learns the temporal dependencies among different clinical events to improve the predictive power. To demonstrate the effectiveness of the proposed approach, we applied it to two benchmark medical tasks (mortality and length of stay).

Epi_DCA : Adaptation et mise en œuvre de la théorie du danger pour la veille épidémiologique

Bahdja Boudoua[1,3], Mathieu Roche[1,4],
Maguelonne Teisseire [1,3], Annelise Tran[1,2,4]

[1] UMR TETIS, Univ. Montpellier, AgroParisTech, CIRAD, CNRS, INRAE, Montpellier, France.
[2]UMR ASTRE, Univ. Montpellier, CIRAD, INRAE, Montpellier, France.
[3]INRAE, UMR TETIS, Montpellier, France.
[4] CIRAD, UMR TETIS, F-34398 Montpellier, France.

Résumé. Le rôle des systèmes de surveillance basés sur les évènements (SBE) est de détecter les nouvelles épidémies en explorant les informations sanitaires publiées en ligne dans un large éventail de sources formelles et informelles. Les facteurs de risque (environnementaux, climatiques, liés aux pratiques d'élevage etc.) ne sont en général pas pris en compte par ces systèmes. Dans cet article, nous souhaitons poser les premières bases d'une démarche générique (indépendante d'une maladie ou d'un hôte spécifique) qui permet de renforcer ou non un événement détecté par les systèmes de veille en y intégrant les facteurs de risque disponibles. Epi_DCA est une adaptation de l'algorithme des cellules dendritiques (DCA), inspiré de la théorie du danger. Il permet de combiner les différents facteurs de risque aux données épidémiologiques issues des systèmes de veille. Un premier test est effectué sur le cas d'étude Influenza aviaire (IA). Par la suite, l'approche proposée sera évaluée sur différents cas d'étude (fièvre du Nil occidental et peste porcine africaine) afin de tester sa robustesse et sa généricité.

1 Introduction

Pour faire face aux maladies émergentes qui représentent un risque croissant pour la santé publique, de nombreux pays adoptent une stratégie de veille sanitaire. Celle-ci repose sur deux composantes : la surveillance basée sur les indicateurs (SBI) (issus de sources officielles telles que l'OIE (Office International des Epizooties : Organisation mondiale de la santé animale), l'OMS (Organisation Mondiale de la Santé) ou la FAO (Food and Agriculture Organization)) et la surveillance basée sur les évènements (SBE) issus de sources non-officielles (médias en ligne, réseaux sociaux, etc.).
Les systèmes de SBE tels que ProMed (Carrion and Madoff, 2017), HealthMap (Freifeld et al., 2008), et PADI-Web (Valentin et al., 2020) sont utilisés quotidiennement afin de détecter les évènements de santé inhabituels. Ils collectent et analysent un flux quotidien de données textuelles non structurées (articles) à partir d'internet, en utilisant des mots-clés ou des combinaisons de mots-clés (Barboza et al., 2014). Par la suite, ces articles sont triés selon leur pertinence

et classés par date, localisation géographique, source, et maladie. La surveillance basée sur les évènements permet ainsi l'obtention de nombreuses informations mais présente certaines limites. En particulier, les facteurs de risque liés à l'apparition des maladies ne sont pas toujours retrouvés dans les données textuelles et ne sont pas pris en compte par les SBE. Parallèlement, la cartographie du risque en épidémiologie permet de mettre en évidence les zones favorables à l'apparition d'une maladie en s'appuyant sur la répartition spatiale des facteurs de risque associés (Hess et al., 2018). La connaissance de ces facteurs de risque est essentielle pour mieux cibler les zones de surveillance et adapter les mesures de lutte et de prévention (Bergmann et al., 2021). Dans cet article, les premières bases d'une approche inspirée de la théorie du danger sont établies : Epi_DCA est l'adaptation de l'algorithme des cellules dendritiques (DCA) à la problématique de veille sanitaire et afin de classer les articles analysés comme pertinents (l'article traite d'une émergence ou ré-émergence d'influenza aviaire) vs non-pertinents. Cette méthode permet de combiner les facteurs de risque aux données épidémiologiques issues des systèmes de veille tout en prenant en compte la dimension spatio-temporelle des évènements épidémiologiques.

La suite de l'article est structurée de la façon suivante. L'état de l'art sur la théorie du danger et les travaux relatifs au DCA sont présentés en Section 2. Notre méthode Epi_DCA est décrite en Section 3 puis un cas d'étude et les résultats préliminaires sont présentés en Section 4. Un bilan et les perspectives sont abordés en Section 5.

2 État de l'art

2.1 Théorie du danger

La théorie du danger (Matzinger, 2002) est basée sur le fonctionnement des cellules immunitaires dendritiques (DCs). Les DCs jouent un rôle essentiel dans le déclenchement des réponses immunitaires. Elles sont parmi les premières cellules exposées à l'environnement extérieur et ont la capacité de détecter et d'interpréter une multitude d'informations moléculaires potentiellement contradictoires. L'interprétation de ces informations (signaux) au système immunitaire adaptatif conduit au déclenchement ou non d'une réponse contre les menaces perçues.

La théorie du danger stipule que la reconnaissance d'un antigène par une DC ne réside pas dans la distinction entre le soi et le non-soi mais dépend plutôt du contexte environnemental (signaux) dans lequel l'antigène est identifié. Les DCs existent dans l'un des trois états suivants : "immature", "semi-mature" et "mature". A leur état initial, les DCs sont "immatures". Ensuite, en fonction de la concentration des signaux auxquels elles sont exposées, elles se différencient soit en cellules "semi-matures" pour inhiber la réponse immunitaire, soit en cellules "matures" pour l'activer (Gu et al., 2013).

Cette théorie et le comportement des DCs a servi d'inspiration pour le développement d'un algorithme de classification, l'algorithme des cellules dendritiques (DCA) (Greensmith, 2007).

Le DCA a été appliqué avec succès à un large éventail d'applications, par exemple dans le domaine de la sécurité informatique (Jim et al., 2022; Sharaff et al., 2021), ou encore en sismologie (Zhou et al., 2020). Il présente les avantages suivants lorsqu'il est appliqué à des problèmes en temps réel :

— Il ne nécessite pas de longues périodes d'apprentissage ;

— Il permet d'intégrer des données hétérogènes par le biais de deux types de signaux ;
— Il a montré des résultats prometteurs quant à la réduction du nombre de faux positifs (Mohsin et al., 2014).

2.2 Algorithme des cellules dendritiques (DCA) et travaux associés

Le DCA a été initialement conçu pour être utilisé comme un algorithme de détection d'anomalies.
Son processus comprend 4 phases : pré-traitement et catégorisation des données, détection des antigènes (AGs) par les cellules, évaluation du contexte cellulaire, et classification des antigènes.
La première phase comprend deux étapes principales : la réduction des attributs et la catégorisation du signal. Pour réduire le nombre d'attributs, certains travaux ont recours à l'avis d'experts lors de la phase de pré-traitement des données. D'autres utilisent des méthodes statistiques telles que l'analyse en composantes principales (ACP) (Chelly and Elouedi, 2016). Les caractéristiques les plus intéressantes de l'ensemble de données sont ainsi sélectionnées puis classées dans l'une des catégories de signaux définies du DCA, à savoir :

— Les signaux de danger (*danger signals*) qui augmentent proportionnellement à la présence de données représentant une situation "anormale" ;
— Les signaux sécuritaires (*safe signals*) qui augmentent proportionnellement à la présence de données représentant une situation "normale".

Après cette première phase, chaque antigène est caractérisé par son signal de danger et son signal sécuritaire.
Lors de la phase de détection, chaque cellule dendritique (DC) est exposée aléatoirement à des antigènes (AGs). Les signaux de sortie cumulés (CSM) sont calculés selon l'équation suivante :

$$Eq.1 \qquad CSM = W_{SS} \times S_S + W_{DS} \times S_D$$

où S_S et S_D représentent les valeurs et W_{SS} et W_{DS} les poids des signaux sécuritaires (S) et des signaux de danger (D), respectivement.
Les pondérations utilisées dans le DCA peuvent être dérivées empiriquement des données ou de valeurs définies par l'utilisateur. Les signaux de danger ont toujours un poids positif et les signaux sécuritaires un poids négatif.
Les CSM des DCs ont deux rôles : d'abord de définir un contexte cellulaire (mature ou semi-mature), et ensuite de stopper l'exposition des cellules aux antigènes (Farzadnia et al., 2021). En effet, pour limiter le temps d'exposition des DCs, chaque DC se voit attribuer une valeur de seuil de migration lors de sa création. Suite à la mise à jour des CSM, si la valeur de CSM dépasse la valeur du seuil de migration, alors l'exposition est stoppée (Chelly Dagdia and Elouedi, 2020).
Lors de la troisième phase, le contexte cellulaire est utilisé pour étiqueter les antigènes collectés par les DCs, et cette information est finalement utilisée dans la génération d'un coefficient d'anomalie qui sera traité dans la phase finale (classification) (Chelly and Elouedi, 2016). Le coefficient d'anomalie noté MCAV (pour Moleculare Antigen Value) reflète le degré d'anomalie d'un antigène donné (plus le MCAV est proche de 1, plus la probabilité qu'un antigène soit anormal est grande), et est calculé en divisant le nombre de DCs matures par le nombre total

de DCs exposées à un antigène, comme indiqué dans l'Eq.2

$$Eq.2 \qquad \text{MCAV} = \frac{\alpha}{(\alpha+\beta)}$$

où α et β sont respectivement le nombre de DC matures et immatures qui ont été exposées à l'antigène.
Une fois le MCAV calculé pour chaque antigène, dans sa dernière phase l'algorithme DCA peut effectuer sa tâche de classification, en comparant le MCAV de chaque antigène à un seuil d'anomalie (paramètre défini par l'utilisateur, ou estimé par apprentissage).
Les premières versions du DCA présentaient un nombre important de paramètres et d'éléments stochastiques (Greensmith, 2007), tels que l'exposition aléatoire et les seuils de migration variables des cellules (Greensmith and Aickelin, 2008). L'estimation de ces paramètres était souvent faite de façon arbitraire et cette limite de l'algorithme a été soulignée par plusieurs études critiques (Chelly and Elouedi, 2016). Différentes versions ont déjà été présentées afin de réviser et d'améliorer le DCA en fonction du problème étudié (Elisa et al., 2018; Zhou and Liang, 2021).
Cependant, il n'y a pas suffisamment d'études qui abordent la problématique de l'exposition aléatoire et de la définition du seuil de migration des DCs. D'autre part, à notre connaissance le DCA n'a pas été appliqué à la thématique de la veille sanitaire de manière à intégrer données épidémiologiques et leurs facteurs de risque en prenant en compte leurs dimensions spatiale et temporelle.
Les contributions principales du travail présenté ici sont les suivantes :

1. Nous adaptons le DCA à la problématique de la veille sanitaire afin de combiner les différents facteurs de risques aux données épidémiologiques issues des SBE ;
2. Nous intégrons l'information spatiale dans la phase de détection. L'exposition des cellules n'est pas aléatoire mais dépend de la distance entre les DCs et les AGs, ainsi que recommandé par (Farzadnia et al., 2021) ;
3. Enfin, nous intégrons l'information temporelle afin d'obtenir un seuil de migration indépendant de la valeur des signaux cumulés CSM.

3 Transposition à la veille sanitaire

3.1 Pré-traitement et catégorisation des données

Dans le contexte de nos travaux, les évènements extraits d'articles détectés par les systèmes SBE représentent nos antigènes (ce que l'on veut classer), associés à des données environnementales (Figure 1 étape de collecte des données) par une correspondance spatiale. Dans une phase de pré-traitement et catégorisation (Figure 1 Phase 1), ces données d'entrées sont converties en deux catégories de signaux : signaux de danger (*danger signals*) et signaux sécuritaires (*safe signals*). Les données épidémiologiques issues de ces articles (source de l'information, hôte, maladie) constituent les signaux de danger.

Nous nous référons à la connaissance d'experts afin d'établir un barème et donner une note à chaque donnée observée, par exemple : +20 si la source de l'article est officielle, +15 si l'hôte est un animal domestique, etc. Si aucune de ces données n'est présente, le signal de danger est

nul et l'article n'est pas pris en compte.
Les données environnementales représentent les signaux sécuritaires. Un signal sécuritaire maximal indique que l'environnement est défavorable à l'apparition de la maladie, un signal sécuritaire égal à 0 indique au contraire un environnement favorable dans lequel tous les facteurs de risque identifiés sont présents.

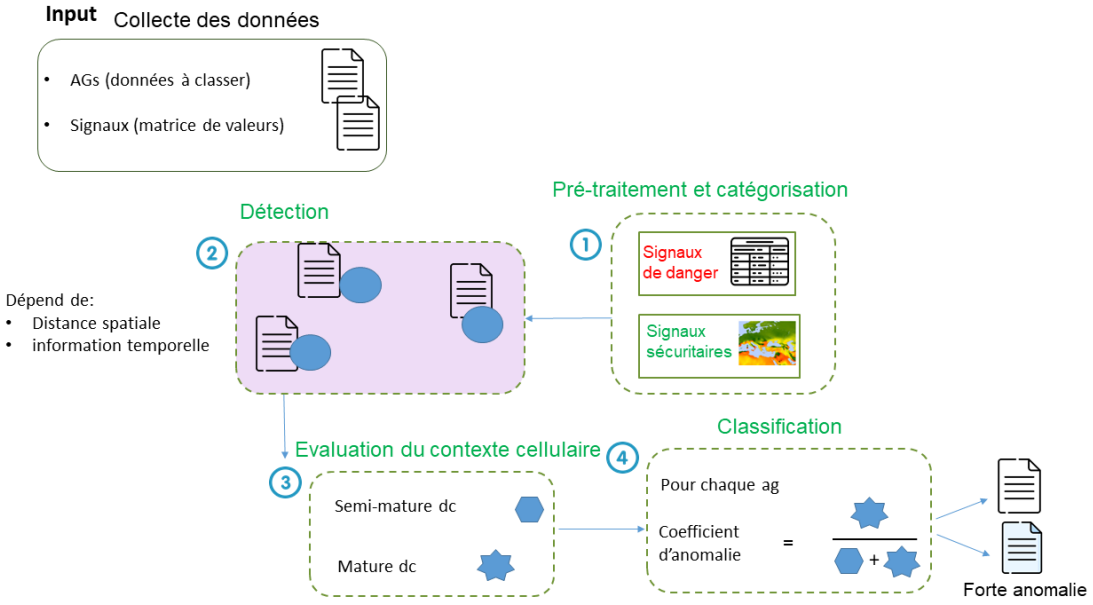


FIG. 1 – *Epi_DCA illustré en 4 phases*

À la suite du pré-traitement et catégorisation des données, Epi_DCA s'articule en 3 phases.

3.2 Phase de détection

A la phase de détection, les DCs sont exposées aux AGs (Figure 1 Phase 2). Le CSM est ensuite calculé en combinant les signaux pondérés de danger et sécuritaires. La pondération est adaptée à la maladie étudiée : pour une maladie fortement influencée par les facteurs environnementaux (par exemple une maladie à transmission vectorielle comme la fièvre du Nil Occidental, ou impliquant un réservoir sauvage, comme l'influenza aviaire) un poids plus important sera attribué aux signaux sécuritaires qui caractérisent le contexte environnemental dans notre approche.
Dans Epi_DCA, l'exposition des DCs dépend de : (1) la distance spatiale entre les DCs et les AGs et (2) le rayon de couverture R des DCs (Figure 2). À chaque pas de temps, les signaux

de sorties cumulés (CSM) des DCs sont mis à jour selon :

$$\begin{cases} CSM_{t+1} = CSM_t + (\Delta_{dist} \times CSM_{entrant}) \\ CSM_0 = 0 \end{cases}$$

avec Δ_{dist} un coefficient de distance inversement proportionnel à la distance spatiale. En d'autres termes, plus la distance est grande, plus la contribution du $CSM\_entrant$ est faible. Cela traduit le fait que la propagation de certaines maladies est liée à la distance entre les évènements observés (Salje et al., 2016).

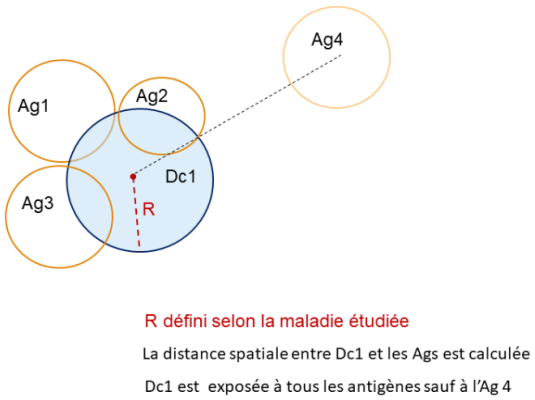


FIG. 2 – *Exemple d'exposition d'une cellule immunitaire dendritique (DC) aux antigènes (AGs)*

Selon la littérature, le seuil de migration (mt) des DCs est déterminé par l'utilisateur, et l'exposition d'une DC aux AGs ne s'arrête que si la valeur cumulée du signal de sortie dépasse le mt. Dans Epi_DCA la migration des DCs dépend aussi de l'information temporelle des antigènes entrants (autrement dit la date de publication des articles) : l'exposition des DCs est interrompue et les signaux cumulés sont remis à 0 lorsqu'aucun antigène n'est détecté pendant une période de temps définie (en fonction de la maladie considérée), ce qui permet de ne pas biaiser la surveillance par des évènements antérieurs (lointains) au même emplacement qui n'ont plus d'impact sur un évènement épidémiologique en cours.
Le pseudo-code complet pour effectuer la phase de détection est présenté dans **Phase 2 - Procedure Detection phase**.

3.3 Phase d'évaluation du contexte cellulaire

La phase d'évaluation du contexte est effectuée une fois que les DCs ont migré, l'évaluation du contexte cellulaire prend en compte les signaux de sortie cumulés et le nombre d'exposition de chaque cellule. Chaque cellule est étiquetée comme "DC mature" ou "DC semi-mature". Le label "mature" signifie que la cellule concernée a été grandement exposée aux signaux de

danger durant la période définie contrairement aux cellules "semi-matures". Cette information sera utilisée lors de la phase de classification.
Le pseudo-code complet permettant de définir le contexte cellulaire est présenté dans **Phase 3 - Procedure Context Assessment phase**.

Phase 2 – Procedure Detection phase

procedure DETECTION PHASE
    Input $ag : antigen\ (Date,\ Coord,\ Host,\ Source,\ Subtype, \emptyset)$
$DCs :\ Cell\ (Date,\ Coord,\ NbExp,\ CSM,\ Context)$
    Output updated ag : antigen (Date, Coord, Host, Source, Subtype, ListCell), updated DCs : Cell (Date, Coord, NbExp ,CSM, Context)

▷ In the detection phase, all the DCs are exposed to the incoming antigen ag.
▷ Compute the exposure for each cell to the antigen (if exists)

    for each cell dc of DCs **do**
        $DistCellAnt \leftarrow calculation\ distance\ between\ ag\ and\ dc$
        $Diffdays CellAnt \leftarrow calculation\ diffdays\ between\ ag\ and\ dc$
        if $DistCellAnt < Disease.space\_limit$ **and**
          $Diffdays CellAnt < Disease.time\_limit$ **then**
            $\Delta_{dist} \leftarrow \frac{(Disease.space\_limit - DistCellAnt)}{Disease.space\_limit}$
            $dc.CSM \leftarrow dc.CSM + (\Delta\_dist \times dc.CSM)$
            $dc.NbExp \leftarrow dc.NbExp + 1$
            $ag.ListCell \leftarrow ag.ListCell + dc$
        end if
    end for
    $DCs \leftarrow DCs + NewCell(ag, Disease)$
end procedure

Phase 3 – Procedure Context Assessment Phase

Input DCs (Date, Coord, NbExp, CSM, Context)
Output updated : DCs (Date, Coord, NbExp, CSM, Context)

$Ratio\_Exp = \frac{Mean(DCs.CSM)}{Mean(DCs.NbExp)}$

▷ $Ratio\_Exp$ depends on the disease, it is used as a threshold to assign the dc.context

for each cell dc of DCs **do**
  if $\frac{dc.CSM}{dc.NbExp} > Ratio\_Exp$ **then**
    $dc.context \leftarrow mature$
  else
    $dc.context \leftarrow semi - mature$
  end if
end for

▷ the dc.context will be used in the last phase (classification) to generate an anomaly coefficient for each ag

3.4 Phase de classification

Enfin, dans la phase de classification des évènements détectés par les SBE (Figure 1 phase 4), les signaux sortants sont utilisés pour générer un coefficient d'anomalie propre à chaque antigène et qui prend ainsi en compte à la fois les informations sanitaires issues des articles (danger signals), le contexte environnemental (safe signals), et l'information spatio-temporelle des évènements épidémiologiques.
Ce coefficient d'anomalie est compris entre 0 et 1, plus sa valeur tend vers 1 plus la probabilité que l'antigène soit anormal est grande. Le seuil d'anomalie (AnomalyThreshold) est fixé à 0.5 comme cela est proposé dans la littérature (Chelly and Elouedi, 2016). Le pseudo-code complet pour la phase de classification est présenté dans **Phase 4 - Procedure Classification phase**.

Phase 4 – Procedure Classification phase

Input AGs set of antigens
Output updated : CoeffAnomaly of each ag of AGs

for each antigen ag of AGs **do**
Compute anomaly coefficient
  $Coeff \leftarrow \frac{ag.ListCell.mature()}{ag.ListCell.length()}$
  ▷ sum of matures cells exposed to ag divided by sum of total exposed cells to ag
  if $Coeff > disease.AnomalyThreshold$ **then**
    $ag.AnomalyCoef \leftarrow anomalous$
  else
    $ag.AnomalyCoef \leftarrow normal$
  end if
end for

4 Cas d'étude IA et résultats préliminaires

Le premier cas d'étude auquel nous nous intéressons est la grippe aviaire (IA - influenza aviaire), d'une part parce qu'il s'agit d'une maladie médiatisée et de ce fait, un nombre conséquent d'articles est détecté par les SBE et d'autre part parce que l'émergence et la diffusion de cette maladie dépendent de différents facteurs de risque (proximité des zones humides, population d'oiseaux sauvages, population d'oiseaux domestiques, etc.). Pour des raisons de disponibilité des données, nous nous sommes intéressés à la région d'Asie du Sud-Est et à la base de données HealthMap (Freifeld et al., 2008).

4.1 Collecte des données et paramètres

Nous avons constitué un jeu de données de 174 articles publiés entre août 2018 et juillet 2019, issus du système de veille HealthMap. Le jeu de données, qui sera rendu public, est constitué actuellement de 87 articles pertinents et 87 articles non pertinents. Dans ce contexte, un article pertinent décrit au minimum un évènement (foyer) d'IA avec sa localisation spatiale (figure 3). Un article non pertinent décrit des mesures sanitaires/économiques, ou traite d'une maladie différente de l'influenza aviaire. Ces articles ont été classés selon leur pertinence manuellement après lecture des textes par un épidémiologiste.

BEIJING (Reuters) - China has confirmed two cases of H5N6 avian bird flu on poultry farms in southwestern province of Yunnan, the Agriculture Ministry said on Wednesday.

Local authorities have culled 10,280 birds following the outbreaks, the Ministry of Agriculture and Rural Affairs said in a statement on its website.

Outbreaks infected a total of 11,340 birds in two farms in Tengchong city and Luquan county in Yunnan, and killed 9,820 of them, the statement said.

FIG. 3 – *Extrait d'un article pertinent détecté par Healthmap*

4.2 Signaux de danger

Les données épidémiologiques issues des articles détectés (source d'information, hôte, maladie) sont utilisées pour générer les signaux de danger. Nous nous référons aux connaissances d'experts afin d'établir un score pour chaque donnée observée **(Tableaux 1 et 2)**. Par exemple, les sources officielles telles que l'OIE, la FAO ont un score plus élevé que les sources non officielles (médias en ligne, réseaux sociaux) et l'influenza aviaire hautement pathogène (IAHP) a un score plus élevé que l'influenza aviaire faiblement pathogène (LPAI). Les signaux de danger ont par la suite été affinés de façon empirique.

Articles ID	**Epidemiological data**		
	Source	**Subtype**	**Host**
ID_1	FAO	HPAI	Wild birds
ID_2	Twitter	Unspecified	Unspecified
ID_3	OIE	LPAI	Domestic birds

TAB. 1 – *Aperçu de la base de données Healthmap*

Articles	**Danger signals**			
	Source	**Subtype**	**Host**	**Total**
Ag_1	30	40	30	**100**
Ag_2	20	0	10	**30**
Ag_2	30	30	20	**80**

TAB. 2 – *Données épidémiologiques (Tableau 1) converties en signaux de danger*

4.3 Signaux sécuritaires

Pour générer les signaux sécuritaires, nous avons créé une carte de risque d'occurrence d'IA selon la méthode décrite par (Stevens et al., 2013) en utilisant des données récentes sur les populations humaines, les oiseaux domestiques et sauvages (figure 4). Ensuite, les événements ont été associés aux données environnementales par correspondance spatiale à l'aide d'un Système d'Information Géographique (SIG). Le logiciel QGIS [1] a été utilisé.

La valeur des signaux sécuritaires est comprise entre 0 et 100 et diminue proportionnellement à la valeur de l'indice de risque d'occurrence d'IA.

4.4 Rayon de couverture et seuil de migration des DCs

Le virus de l'IA est susceptible de se propager par différentes voies (transport de volailles, migration des oiseaux...etc) (Yousefinaghani et al., 2020). Cela rend sa dynamique de diffusion complexe et difficile à déterminer. Ici nous fixons le rayon de couverture des DCs à 20 km, qui correspond à la distance pour laquelle les restrictions et mesures de contrôle sont mises en place (zone de surveillance) autour des foyers d'IA (Pittman and Laddomada, 2008). Quant au seuil de migration des DCs, il a été fixé à 21 jours, car au-delà de cette période si aucun nouvel évènement d'IA n'est détecté, le foyer concerné est considéré comme assaini.

4.5 Résultats

Les résultats de la classification des articles appartiennent à l'une des classes suivantes : True Positive (TP) : Article pertinent correctement classé, True Negative (TN) : Article non-pertinent correctement classé, False Positive (FP) : Article non-pertinent classé comme pertinent et False Negative (FN) : Article pertinent classé comme non-pertinent. Les métriques

1. www.qgis.org

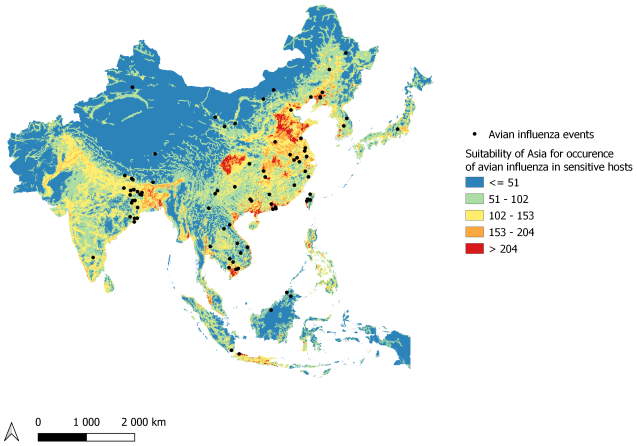


FIG. 4 – *Carte de risque d'occurrence d'influenza aviaire en Asie du Sud-Est obtenue par évaluation multicritère spatialisée (Stevens et al., 2013). L'indice de risque est compris entre 0 (risque faible) et 255 (risque fort)*

utilisées pour évaluer les performances de l'algorithme sont Précision, Rappel, et F-score. Ces métriques sont calculées selon les équations suivantes :

$$Précision = \frac{TP}{(TP + FP)} \tag{1}$$

$$Rappel = \frac{TP}{(TP + FN)} \tag{2}$$

$$F - Score = 2 \text{ X } \frac{Précision \text{ X } Rappel}{Précision \ + \ Rappel} \tag{3}$$

Métrique	Epi_DCA	EPI_DCA sans safe signals
Précision	0.850	0.827
Rappel	0.870	0.782
F-score	0.860	0.80

TAB. 3 – *Résultats de classification - Epi_DCA - 1er cas d'étude avec et sans intégration des signaux sécuritaires*

Dans un premier temps, notre méthode a été testée avec et sans intégration des signaux sécuritaires. Les résultats indiqués dans le Tableau 3, sont encourageants et suggèrent que la prise

en compte du contexte environnemental (dans ses dimensions spatio-temporelles) dans l'analyse des données épidémiologiques issues des SBE permet de renforcer les articles détectés par les SBE.

Par la suite, le logiciel Weka [2] a été utilisé pour tester quatre méthodes d'apprentissage supervisé (SVM, Naive Bayes, Knn et Random Forest) sur notre jeu de données en effectuant une validation croisée en 5 plis. Nous avons obtenu une F-mesure entre 0.861 (Naive Bayes) et 0.913 (SVM) ce qui montre que notre approche Epi_DCA, qui a la caractéristique d'être non supervisée, reste tout à fait compétitive.

Métriques	**Résultats**			
	SVM	Naive Bayes	K-nn	Random Forest
Précision	0.923	0.869	0.882	0.918
Rappel	0.914	0.862	0.879	0.902
F-Score	0.913	0.861	0.879	0.901

TAB. 4 – *Résultats de classification obtenus avec les méthodes d'apprentissage supervisé*

5 Conclusion et perspectives

Dans cette étude nous avons posé les premières bases d'Epi_DCA qui est l'adaptation du DCA à la problématique de la veille sanitaire. Les principales contributions de ce travail sont la prise en compte des facteurs de risque associés à la maladie ciblée ainsi que l'intégration de la dimension spatio-temporelle des évènements épidémiologiques dans la méthode. A partir du pseudo code présenté ici, nous avons implanté Epi_DCA en utilisant le langage R qui donnera lieu à une librairie que nous rendrons disponible pour la communauté. Les paramètres des deux types de signaux permettent d'ajuster la sensibilité de l'algorithme et de l'adapter à la maladie ciblée. Ainsi, en changeant le jeu de données et/ou la maladie cible, il nous est possible de se calibrer simplement et efficacement et d'augmenter grandement la réutilisabilité de notre code. Epi_DCA a été testé et évalué dans un premier temps sur le cas d'étude influenza aviaire et a montré des résultats prometteurs. Cette méthode ne nécessite pas d'apprentissage, elle permet de combiner des données hétérogènes par le biais de deux types de signaux et a montré de bons résultats quant à la réduction du nombre de faux positifs. Ces avantages ont conduit à l'explorer dans le contexte de veille sanitaire. La méthode proposée sera testée sur d'autres cas d'étude pour tester sa généricité : pour une maladie à transmission vectorielle comme la fièvre du Nil Occidental et pour une maladie transfrontalière comme la peste porcine africaine, et ce dans différents contextes géographiques.

Remerciements

Ce travail est financé par le projet « Monitoring outbreak events for disease surveillance in a data science context » (MOOD) du programme de recherche et d'innovation Horizon 2020

2. https ://www.cs.waikato.ac.nz/ml/weka/index.html

de l'Union européenne dans le cadre de la convention de subvention n° 874850 (https ://mood-h2020.eu /). Nous remercions également le projet HealthMap (https ://healthmap.org/), qui nous a fourni les données.

Références

P. Barboza, L. Vaillant, Y. Le Strat, D. M. Hartley, N. P. Nelson, A. Mawudeku, L. C. Madoff, J. P. Linge, N. Collier, J. S. Brownstein, et al. Factors influencing performance of internet-based biosurveillance systems used in epidemic intelligence for early detection of infectious diseases outbreaks. *PloS one*, 9(3) :e90536, 2014.

H. Bergmann, K. Schulz, F. J. Conraths, and C. Sauter-Louis. A review of environmental risk factors for african swine fever in european wild boar. *Animals*, 11(9) :2692, 2021.

M. Carrion and L. C. Madoff. Promed-mail : 22 years of digital surveillance of emerging infectious diseases. *International health*, 9(3) :177–183, 2017.

Z. Chelly and Z. Elouedi. A survey of the dendritic cell algorithm. *Knowledge and Information Systems*, 48(3) :505–535, 2016.

Z. Chelly Dagdia and Z. Elouedi. A hybrid fuzzy maintained classification method based on dendritic cells. *Journal of Classification*, 37(1) :18–41, 2020.

N. Elisa, L. Yang, and N. Naik. Dendritic cell algorithm with optimised parameters using genetic algorithm. In *2018 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8. IEEE, 2018.

E. Farzadnia, H. Shirazi, and A. Nowroozi. A new intrusion detection system using the improved dendritic cell algorithm. *The Computer Journal*, 64(8) :1193–1214, 2021.

C. C. Freifeld, K. D. Mandl, B. Y. Reis, and J. S. Brownstein. Healthmap : global infectious disease monitoring through automated classification and visualization of internet media reports. *Journal of the American Medical Informatics Association*, 15(2) :150–157, 2008.

J. Greensmith. *The dendritic cell algorithm*. PhD thesis, Citeseer, 2007.

J. Greensmith and U. Aickelin. The deterministic dendritic cell algorithm. In *International conference on artificial immune systems*, pages 291–302. Springer, 2008.

F. Gu, J. Greensmith, and U. Aickelin. Theoretical formulation and analysis of the deterministic dendritic cell algorithm. *Biosystems*, 111(2) :127–135, 2013.

A. Hess, J. Davis, and M. Wimberly. Identifying environmental risk factors and mapping the distribution of west nile virus in an endemic region of north america. *GeoHealth*, 2(12) : 395–409, 2018.

L. E. Jim, N. Islam, and M. A. Gregory. Enhanced manet security using artificial immune system based danger theory to detect selfish nodes. *Computers & Security*, 113 :102538, 2022.

P. Matzinger. The danger model : a renewed sense of self. *science*, 296(5566) :301–305, 2002.

M. F. M. Mohsin, A. A. Bakar, and A. R. Hamdan. Outbreak detection model based on danger theory. *Applied soft computing*, 24 :612–622, 2014.

M. Pittman and A. Laddomada. Legislation for the control of avian influenza in the european union. *Zoonoses and public health*, 55(1) :29–36, 2008.

H. Salje, D. A. Cummings, and J. Lessler. Estimating infectious disease transmission distances using the overall distribution of cases. *Epidemics*, 17 :10–18, 2016.

A. Sharaff, C. Kamal, S. Porwal, S. Bhatia, K. Kaur, and M. M. Hassan. Spam message detection using danger theory and krill herd optimization. *Computer Networks*, 199 :108453, 2021.

K. B. Stevens, M. Gilbert, and D. U. Pfeiffer. Modeling habitat suitability for occurrence of highly pathogenic avian influenza virus h5n1 in domestic poultry in asia : a spatial multicriteria decision analysis approach. *Spatial and spatio-temporal epidemiology*, 4 :1–14, 2013.

S. Valentin, E. Arsevska, S. Falala, J. De Goër, R. Lancelot, A. Mercier, J. Rabatel, and M. Roche. Padi-web : A multilingual event-based surveillance system for monitoring animal infectious diseases. *Computers and Electronics in Agriculture*, 169 :105163, 2020.

S. Yousefinaghani, R. A. Dara, Z. Poljak, and S. Sharif. A decision support framework for prediction of avian influenza. *Scientific Reports*, 10(1) :1–14, 2020.

W. Zhou and Y. Liang. A new version of the deterministic dendritic cell algorithm based on numerical differential and immune response. *Applied Soft Computing*, 102 :107055, 2021.

W. Zhou, Y. Liang, Z. Ming, and H. Dong. Earthquake prediction model based on danger theory in artificial immunity. *Neural Network World*, 30(4) :231, 2020.

Réduction du risque du coût d'un modèle dans la détection de fraude financière.

Hamza Chergui*,**, Lylia Abrouk*
Nadine Cullot*, Nicolas Cabioch**

*Université de Bourgogne
hamza.chergui@etu.u-bourgogne.fr,
lylia.abrouk,nadine.cullot@u-bourgogne.fr
**SKAIZen Group
hchergui,ncabioch@skaizengroup.fr
https://skaizengroup.eu/

Résumé. La lutte contre la fraude financière est un enjeu majeur pour les institutions financières. Ces dernières années, plusieurs approches basées sur l'analyse des transactions bancaires ont été proposées pour la détéction de fraude. Dans ce travail, nous proposons une approche basée sur les techniques d'apprentissage automatique. Le but est la détection de fraudes financières sur des transactions internationales et interbancaires du réseau SWIFT. Nous entraînons un modèle avec des caractéristiques calculées à partir des spécificités des transactions SWIFT. Nous définissons une mesure de risque du coût sur les prédictions d'un modèle que nous souhaitons réduire avec notre méthodologie. Les expérimentations ont été menées sur un jeu de données réel et validées par les experts du domaine.

1 Introduction

La lutte contre la fraude financière est une tâche complexe pour les institutions financières. Selon Knobel (2019) 98,9% des activités liées aux fraudes financières passent à travers les mailles du filet. Les institutions financières se doivent d'améliorer leurs systèmes sous peine de sanctions financières conséquentes des régulateurs du monde financier.
Les flux financiers sont composés de transactions et doivent être analysés par les institutions financières en étudiant les comportements des acteurs impliqués. Une transaction analysée frauduleuse est soit bloquée par le système, soit laissée dans le flux avec une alerte. Ces deux situations nécessitent une analyse manuelle d'un expert pour la gestion de ces transactions. Les systèmes implémentés dans les institutions financières sont basés sur des règles pré-définies, leurs mécanismes présentent une faiblesse exploitée par les fraudeurs : ils identifient ces règles et adaptent leur manière de frauder pour les contourner. Notre travail s'inscrit dans les travaux de recherche en collaboration avec l'entreprise SKAIZen Group qui vise à améliorer la détection de fraude avec des données provenant d'une société, appelé SWIFT [1]. Il s'agit d'une

1. https ://www.swift.com/

société mettant à disposition un réseau interbancaire proposant différents services comme le transfert d'argent entre différents comptes bancaires. Ce réseau permet de réaliser des transactions financières entre plus de 11000 organismes bancaires à travers près de 200 pays. Ces dernières années, des travaux utilisant des algorithmes d'apprentissage automatique ont été utilisés pour la détection transactions frauduleuses. Elles permettent de pallier les limites des systèmes de détection de fraudes basés sur des règles pré-définies, notamment avec des tâches de classification réalisées avec des modèles prédictifs. À travers notre travail, nous souhaitons présenter les différentes techniques d'apprentissage automatique et observer leurs utilités dans le domaine de la détection de fraude financière (DFF). Nous proposons ensuite une méthodologie pour entraîner un modèle qui sera évalué avec une mesure de risque du coût. Cette dernière est associée aux coûts financiers des prédictions du modèle et nous permettra de choisir un seuil de probabilité à partir duquel nous considérons une transaction comme frauduleuse. La suite de l'article est organisée de la manière suivante : dans la section 2, nous dressons un état de l'art des techniques d'apprentissage automatique dans le domaine des fraudes financières. Dans la section 3, nous présentons notre approche que nous validons avec des expérimentations sur un jeu de données réel dans la section 4. Enfin, nous concluons et abordons nos perspectives de recherche dans la section 5.

2 Travaux liés

Les méthodes basées sur les techniques d'apprentissage automatique pour la détection de fraudes peuvent être un réel atout pour les institutions financières grâce à leurs prédictions rapides et intelligentes. De nombreux travaux existent dans le domaine de la finance (Al-Hashedi et Magalingam, 2021) et plus particulièrement dans la détection de fraude par carte de crédit (Adewumi et Akinyelu, 2017). Nous proposons de présenter les techniques d'apprentissage automatique en plusieurs étapes (Chergui et al., 2022) :

L'obtention des données dans le milieu financier est difficile en raison de la politique de confidentialité des institutions financières. Il existe une réelle disparité des jeux de données utilisés dans la littérature. Nous trouvons des jeux de données publics [2], synthétiques (Lopez-Rojas et al., 2016) et privés. Une telle disparité rend difficile la comparaison des approches qui utilisent des schémas de données ainsi que des types de fraude différents. De plus, il est souvent difficile de reproduire les approches, car les algorithmes ne sont pas partagés.

L'extraction de caractéristiques permet d'enrichir le jeu de données afin de distinguer les transactions frauduleuses des transactions légitimes. Dans la littérature de la DFF, les travaux de Whitrow et al. (2009) et Bhattacharyya et al. (2011) conventionnent les caractéristiques à calculer, notamment en s'intéressant aux volumes et aux moyennes des montants des transactions réalisés par les acteurs sur différentes temporalités. En outre, les jeux de données comportant des fraudes sont déséquilibrés, il existe un nombre plus élevé de transactions légitimes que de transactions frauduleuses. Le pourcentage de transactions frauduleuses représente moins d'1% sur ces jeux de données. Ce déséquilibre peut impacter négativement l'apprentis-

2. https ://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

sage. Ainsi, des techniques *d'over/under sampling* ont été développées pour soit augmenter le pourcentage de la classe minoritaire, soit de diminuer le pourcentage de la classe majoritaire.

L'entraînement d'un modèle prédictif est basé soit sur un apprentissage supervisé, non supervisé ou semi-supervisé. Dans la DFF, l'apprentissage supervisé a pour but de classifier les transactions dans les classes *légitimes* ou *frauduleuses*. Des travaux existent sur la comparaison de différents algorithmes (ex : Random Forest, SVM, Naive Bayes, ...) comme ceux de Varmedja et al. (2019) et Khatri et al. (2020). Les algorithmes d'apprentissage non supervisé regroupent les données dans des clusters, la détection de fraude peut s'opérer de deux manières : soit avec des clusters jugés frauduleux, où toutes les transactions appartenant à ces clusters seront considérées comme frauduleuses (Le Khac et Kechadi, 2010) ; soit avec des algorithmes tels que *Isolation Forest* (Liu et al., 2008) ou *Local Outlier Factor* (Breunig et al., 2000) permettant d'identifier des anomalies au sein des clusters. Ces algorithmes ont montré leur efficacité auprès des travaux de John et Naaz (2019) ou Mishra et Chawla (2019). En outre, les techniques basées sur les réseaux de neurones ne sont pas performantes sur des données tabulaires (Borisov et al., 2021), ces techniques ne sont pas très présentes dans la littérature.

L'évaluation du modèle s'opère avec des mesures classiques de *précision*, *rappel* et *f1-score* (F1). Il y a également l'*AUC-PR* (aire sous la courbe de précision et de rappel), une mesure utilisée avec des jeux de données déséquilibrés qu'on retrouve dans des travaux de la DFF (Bahnsen et al., 2013). Des mesures de risque du coût ont vu le jour dans le domaine de la DFF pour calculer le coût des prédictions des modèles lors de l'évaluation. Ce risque du coût représente l'argent que l'institution financière risque du perdre avec le modèle.

Ces différentes étapes nous permettent d'avoir une vue sur les techniques d'apprentissage automatique utilisées au sein de la DFF. Nous proposons, dans la suite, une méthodologie pour entraîner un modèle sur des transactions SWIFT en plusieurs étapes : (1) la définition de nouvelles caractéristiques, (2) une méthode d'apprentissage pour entraîner plus rapidement un modèle et (3) une évaluation basée sur une mesure de risque du coût.

3 Méthodologie

Notre méthodologie présentée sur la figure 1 est composée de trois parties : dans la première, nous enrichissons notre jeu de données avec deux types de caractéristiques. Dans la deuxième partie, nous divisons notre jeu de données et entraînons des modèles prédictifs sur chaque partie du jeu de données. Enfin, dans la troisième partie, nous introduisons les mesures utilisées pour l'évaluation de ces modèles avec une mesure de risque du coût utilisée pour choisir le seuil de probabilité à partir duquel une transaction est frauduleuse.

3.1 Données et caractéristiques

Les attributs d'une transaction SWIFT et un label indiquant si la transaction est frauduleuse (Vrai) ou légitime (Faux) sont présentés dans le tableau 1. Il y a 3 acteurs : l'émetteur, l'intermédiaire et le bénéficiaire. La transaction est un transfert d'argent entre l'émetteur et le bénéficiaire, un intermédiaire intervient dans la transaction si l'émetteur et le bénéficiaire ne possèdent pas de relation d'échange établie. Les acteurs sont identifiés à travers un code appelé 'BIC' dont nous pouvons extraire le code du pays de l'acteur avec le quatrième et cinquième

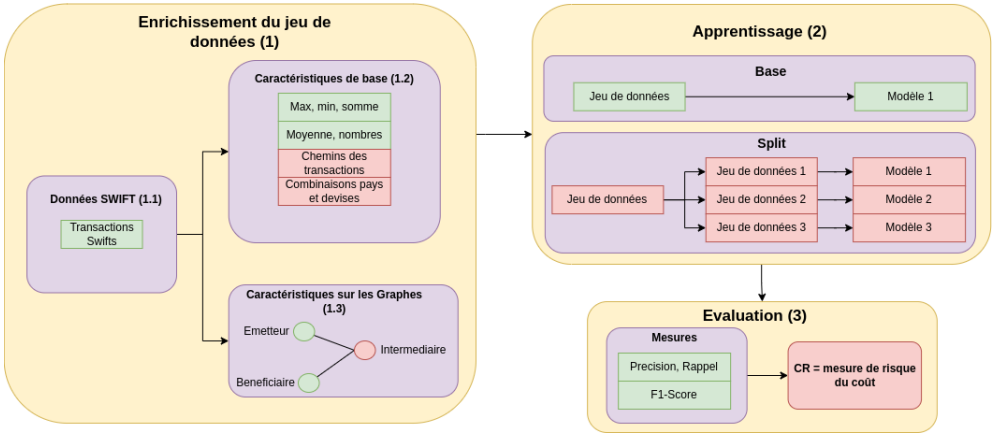


FIG. 1 – Schéma de la méthodologie.

caractère (ex : FR). Le montant représente la valeur de la transaction. D'une manière plus formelle, nous pouvons représenter un jeu de données J avec un ensemble de transactions t^i tel que $i \in [1, n]$ et n son nombre de transactions.

TAB. 1 – Exemples de transactions SWIFT avec un label

Émetteur	Intermédiaire	Bénéficiaire	Date	Devise	Montant	Label
BIC0FR01	BIC0IT01	BIC0FR02	210625	EUR	15006	Faux
BIC0US03	-	BIC0GB01	210625	GBP	33065	Faux
BIC0FR04	BIC0FR06	BIC0FR05	210626	EUR	100325	Vrai

Calcul des caractéristiques de base : à partir d'un jeu de données J, pour chaque acteur et pays, nous calculons des caractéristiques de base présentées dans le tableau 2, que nous ajoutons aux transactions. Ces caractéristiques nous permettent de modéliser leurs comportements (Bhattacharyya et al., 2011). Ainsi, pour chaque transaction t^i, nous calculons 5 caractéristiques sur l'agrégation des historiques de transaction des acteurs ou pays. Dans la suite de cet article, les caractéristiques des acteurs auront pour code $base\_acteurs$ et celles des pays $base\_pays$, ces codes seront utilisés lors de la partie expérimentations.

TAB. 2 – Caractéristiques de base

Caractéristiques	**Formalisations**
Montant maximum	$e_{max} = \underset{1<=i<=n}{max}\ t^i_{montant}$
Montant minimum	$e_{min} = \underset{1<=i<=n}{min}\ t^i_{montant}$
Somme des montants	$e_{sum} = \sum_{i=1}^{n} t^i_{montant}$
Nombre de transactions	$e_{count} = n$
Moyenne des montants	$e_{avg} = e_{sum}/e_{count}$

Calcul des caractéristiques basées sur les graphes : à partir des transactions d'un jeu de données, nous construisons deux graphes orientés et pondérés : le graphe ($G_{acteurs}$) possède comme nœuds les acteurs (émetteurs et bénéficiaires), comme arcs les transactions des émetteurs vers les bénéficiaires et comme poids des arcs le nombre de transactions réalisées entre eux. Le second graphe (G_{pays}) possède comme nœuds les pays (pays émetteurs et pays bénéficiaires), comme arcs les transactions entre les pays émetteurs et pays bénéficiaires et comme poids des arcs le nombre de transactions réalisées entre eux.

À partir de ces deux graphes, nous obtenons deux matrices d'adjacence : $A_{acteurs}$ de dimension $N \times N$ avec N le nombre d'acteurs du jeu de données J ; et A_{pays} de dimension $M \times M$ avec M le nombre de pays de J. Nous construisons également 6 vecteurs, 3 vecteurs de dimensions N avec $E_{acteurs}$ la moyenne des montants des transactions des acteurs, $E^{+}_{acteurs}$ étant la moyenne des montants des transactions des clients en tant qu'émetteur, et $E^{-}_{acteurs}$ étant la moyenne des montants des clients en tant que bénéficiaire. Nous construisons de la même façon les vecteurs E_{pays}, E^{+}_{pays} et E^{-}_{pays} de dimensions M.

À partir des matrices d'adjacences A et des vecteurs E, E^{+} et $E-$, nous calculons les 3 caractéristiques présentées dans le tableau 3, que nous ajoutons aux transactions. Ces caractéristiques permettent de modéliser le comportement des acteurs et pays sur : leur voisinage entier (e_g), leur voisinage en tant qu'émetteur ou pays émetteur (e_{g+}), et de leur voisinage en tant que bénéficiaire ou pays bénéficiaire (e_{g+}) (Huang et al., 2018). Dans la suite de cet article, les caractéristiques des acteurs auront pour code $graphe\_acteurs$ et celles des pays $graphe\_pays$. Ces codes seront utilisés lors de la partie expérimentations.

TAB. 3 – Caractéristiques de graphe.

Caractéristiques	**Formalisations**	
	Acteurs	Pays
Voisinage entier	$e_g = A_{acteurs}.E_{acteurs}$	$e_g = A_{pays}.E_{pays}$
Voisinage émetteur	$e_{g+} = A_{acteurs}.E^{+}_{acteurs}$	$e_{g+} = A_{pays}.E^{+}_{pays}$
Voisinage bénéficiaire	$e_{g-} = A_{acteurs}.E^{-}_{acteurs}$	$e_{g-} = A_{pays}.E^{-}_{pays}$

3.2 Apprentissage des modèles de classification

Les modèles de classification sont entraînés avec des algorithmes d'apprentissage supervisé par le biais d'un jeu de données labellisé. Nous effectuons une tâche de classification avec deux classes *fraude* et *légitime*. L'entraînement d'un modèle peut être long en fonction du nombre de données et de leur dimension. À travers notre méthodologie, nous souhaitons diminuer ce temps tout en conservant la qualité de prédiction de notre modèle. Pour réaliser cela, nous entrainons un modèle prédictif ($modele\_split$) en divisant notre jeu de données afin de réduire les temps d'entraînement et de calcul. Nous évaluons l'efficacité de ce modèle en le comparant à un modèle de base entraîné sur notre jeu de données non divisé ($modele\_base$).

Division du jeu de données : En nous basant sur l'hypothèse de nos experts nous disant que les transactions qui partagent les mêmes devises et pays ont des comportements de fraudes similaires. Nous avons extrait, du jeu de données J, les tuples T^j (devise, pays émetteur et

pays bénéficiaire) avec $j \in [1, N]$, N étant le nombre de combinaisons de tuple du jeu de données. Pour chaque tuple, nous calculons le nombre de transactions T_{count} et le montant moyen des transactions T_{avg}. En utilisant les techniques d'apprentissages non supervisés (k-means), et ces deux caractéristiques, nous avons regroupé les tuples dans k clusters avec des valeurs de T_{count} et T_{avg} proches. Le nombre k est choisi lors des expérimentations. Les transactions avec des tuples T dans les mêmes clusters ont été placés dans les mêmes jeux de données. Ainsi, nous avons divisé notre jeu de données J en k jeux de données J^p avec $p \in [1F, k]$.

Entraînement des modèles : Le $modele\_split$ est un modèle composé de k modèles entraînés sur chaque division du jeu de données. Le $modele\_base$ entraîné sur le jeu de données complet aura plus de données pour son apprentissage. Les modèles de $modele\_split$ possèdent moins de données pour leur apprentissage, cependant leurs données sont plus homogènes, car elles partagent des combinaisons de devises et pays avec des comportements similaires au niveau du nombre (T_{count}) et moyenne des montants des transactions (T_{avg}).

3.3 Évaluation des modèles

TAB. 4 – Matrice de confusion.

		Réalité	
		Fraude	Légitime
Prédiction	Fraude	VP	FP
	Légitime	FN	VN

Pour évaluer les modèles, nous utilisons la matrice de confusion présentée dans le tableau 4. Cette matrice nous donne le nombre de transactions frauduleuses correctement prédites VP, le nombre de transactions frauduleuses incorrectement prédites FP, le nombre de transactions légitimes correctement prédites VN et le nombre de transactions légitimes incorrectement prédites FN. À partir de cette matrice, nous pouvons calculer les mesures suivantes :

$$Precision = \frac{VP}{VP + FP}; \quad Rappel = \frac{VP}{VP + FN}; \quad F1 = \frac{2 * VP}{2 * VP + FP + FN} \quad (1)$$

Lorsqu'un modèle est entraîné, sa prédiction pour une transaction est une probabilité d'appartenance à une classe (frauduleuse, légitime). Par défaut, on attribue à la transaction la classe avec la plus grande probabilité. Dans le cas d'une classification à deux classes, nous pouvons faire varier le seuil de probabilité à partir duquel une transaction est frauduleuse. Ainsi pour chaque seuil, nous pouvons calculer la précision et le rappel du modèle pour tracer une courbe.

L'aire sous courbe de précision et de rappel (AUC-PR) est une mesure utilisée sur les modèles entraînés avec des jeux de données déséquilibrés. Dans le domaine financier, un modèle ayant une précision élevée et un rappel faible est un modèle dont les transactions prédites frauduleuses sont réellement frauduleuses avec peu de fausses alertes, allégeant ainsi le coût du travail des experts. Cependant, un rappel faible indique un faible nombre de transactions frauduleuses détectées, pouvant ainsi exposer les institutions financières à des sanctions par les

régulateurs.

Le coût du risque des prédictions : Bahnsen et al. (2013) ont proposé une matrice de coût du risque des prédictions d'un modèle pour le domaine de la détection de fraude par carte de crédit qui est représenté dans le tableau 5.

TAB. 5 – Matrice de risque du coût de Bahnsen et al. (2013).

		Réalité (t_i)	
		Fraude	Légitime
Prédiction (t_i)	Fraude	C_a	C_a
	Légitime	Amt_i	0

Cette matrice représente le coût d'une prédiction d'une transaction t_i pour une institution financière. D'une part, si la transaction est prédite comme frauduleuse, elle a un coût d'administration C_a représentant l'estimation du coût d'un expert pour l'analyse d'une transaction. D'autre part, si la transaction est prédite légitime alors qu'elle est frauduleuse, le coût est égal au montant (Amt_i) de la transaction.
Cette matrice de coût peut être critiquée sur deux aspects : (i) le coût d'une transaction légitime prédite frauduleuse peut avoir un coût supérieur, si un client voit sa transaction bloquée injustement, cela peut impacter la qualité du service du client et résulter en un coût supérieur ; (ii) l'institution financière ne perd pas le montant d'une transaction frauduleuse prédite légitime. En réalité, elle a un risque plus élevé d'être soumise à une sanction financière par les régulateurs du monde financier. Pour ces deux raisons et pour répondre aux besoins de nos experts, nous proposons une matrice de coût présentée dans le tableau 6.

TAB. 6 – Matrice de risque du coût proposée.

		Réalité (t_i)	
		Fraude	Légitime
Prédiction (t_i)	Fraude	C_a	$C_a + C_c$
	Légitime	C_s	0

C_a représente toujours le coût administratif de l'analyse d'une transaction. Nous introduisons C_c qui est ajouté au coût administratif dans le cas d'une transaction légitime classée frauduleuse, ce coût est lié à l'insatisfaction du client. C_s, dans le cas d'une transaction frauduleuse classée légitime, représente l'évaluation du risque d'être soumis à une sanction. À partir de la matrice proposée, nous définissons la formule de calcul de risque du coût suivante :

$$RC = VP * C_a + FP * (C_a + C_c) + FN * C_s \tag{2}$$

Les coûts C_a, C_c et C_s sont à définir avec des experts en fonction de l'évaluation des risques financiers des coûts.
Dans la suite, nous présentons nos expérimentations, dans lesquelles nous allons : (1) diviser notre jeu de données et calculer les caractéristiques de bases et graphes, (2) entraîner des modèles pour chaque jeu de données, en prenant en compte l'impact des caractéristiques, (3)

TAB. 7 – Jeux de données.

J^p	**Nombre**	**Moyenne des montants**
J	1321125	284624
J^0	331845	278739
J^1	689392	284120
J^2	94377	282366

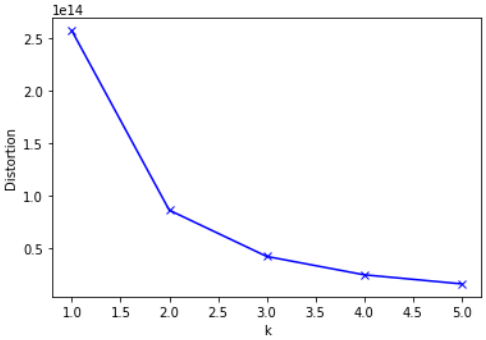


FIG. 2 – Méthode du coude.

choisir le seuil de réduction du risque du coût de notre modèle avec la formule proposée et (4) discuter de nos résultats.

4 Expérimentation

Nous avons réalisé nos expérimentations avec un jeu de données labelisé obtenu grâce à une collaboration avec l'entreprise SKAIZen Group. Ce jeu de données contient 1321125 transactions, 10198 acteurs, 151 devises et 248 pays. 7942 transactions sont labelisées frauduleuses. Les transactions sont réparties sur l'année 2019, nous avons utilisé les transactions de janvier à octobre pour l'entraînement de notre modèle et les transactions de novembre à décembre pour l'évaluation. Nous avons réalisé nos expérimentations en Python avec une machine possédant 16Go de RAM sur un CPU 7-10750H. Nous avons utilisé la librairie Scikit-Learn pour l'apprentissage des modèles.

4.1 Division du jeu de données et calcul des caractéristiques

En analysant le jeu de données, nous avons obtenu 3318 tuples uniques dispersées dans les 1321125 transactions. Pour chaque tuple T nous avons calculé T_{count} et T_{avg} sur lesquels nous avons réalisé un *clustering* avec l'algorithme *k-means*. Pour choisir le nombre de clusters, nous avons utilisé la méthode du coude (*elbow*) permettant d'identifier le nombre de clusters optimal pour notre jeu de données. La figure 2 illustre cette méthode qui calcule pour chaque k la distorsion. Cette dernière correspond à la somme des distances au carré de chaque point avec son centroïde. On définit la valeur de k où la courbe s'infléchit, dans notre cas le k optimal est égal à 3.

Ainsi, nous séparons notre jeu de données J en 3 jeux de données J^0, J^1 et J^2 dont nous présentons le nombre et la moyenne des montants de leurs transactions dans le tableau 7. Pour chaque jeu de données, nous calculons les caractéristiques présentées dans la section 3 : $base\_acteurs$ et $graphe\_acteurs$ pour les émetteurs et bénéficiaires et $base\_pays$ et $graphe\_pays$ pour les pays émetteurs et bénéficiaires. Nous avons ainsi ajouté 32 caractéristiques à chaque transaction en fonction de leurs acteurs et pays.

4.2 Entraînement des modèles

Pour choisir le modèle le plus adapté à notre structure de données et à nos caractéristiques, nous comparons différents algorithmes d'apprentissage supervisé. Les algorithmes les plus performants dans le domaine de la détection de fraude financière sont Random Forest (Xuan et al., 2018), XGBoost (Meng et al., 2020) et CatBoost (Alfaiz et Fati, 2022). Pour chaque algorithme, nous entraînerons deux modèles ($model\_base$ et $model\_split$) que nous évaluons avec les mesures présentées dans la section 3. Pour la mesure du risque du coût RC, nous avons défini, le coût d'administration $C_a = 5$, le coût d'une fausse alerte $C_c = 10$ et le coût d'une transaction frauduleuse prédite légitime $C_p = 20$. Les résultats et le temps d'entraînement (TE) des modèles sont présentés dans le tableau 8. Le choix des hyper-paramètres des algorithmes s'est réalisé avec une recherche exhaustive des paramètres indiqués dans le tableau 9. Pour identifier l'impact des caractéristiques calculées, nous avons conservé le meilleur modèle *XGBoost* qui a obtenu le meilleur F1 (0.78) et AUC-PR(0.66). Nous l'avons entraîné avec les 4 types de caractéristiques pour étudier leur influence sur l'apprentissage. Les résultats sont présentés dans le tableau 10, où nous remarquons une influence importante des caractéristiques liées aux acteurs.

TAB. 8 – Tableau récapitulatif des résultats des algorithmes.

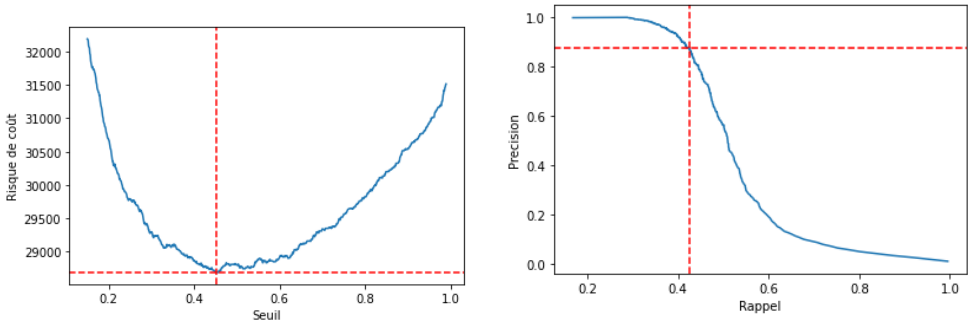
fieur		Précision	Rappel	F1	AUC-PR	RC	TE
RF	BASE	0.98	0.67	0.75	0.65	30170	2mn 57sec
	SPLIT	0.97	0.68	0.76	0.65	30050	2mn
XGBoost	BASE	0.95	0.71	0.78	0.66	28780	1mn 54sec
	SPLIT	0.93	0.70	0.78	0.64	29285	1min 35sec
CatBoost	BASE	0.99	0.66	0.74	0.66	30645	25sec
	SPLIT	0.99	0.66	0.74	0.65	30630	28sec

TAB. 9 – Recherches aléatoires des hyper-paramètres.

Random Forest		CatBoost		XGBoost	
arbres	5,10,100	*learning rate*	[0.01,0.05,0.1]	*learning rate*	[0.01,0.05,0.1]
profondeur	5,10,None	profondeur	5,10,None	profondeur	5,10,None
critere	gini, entropie	iterations	10,50,100	arbres	5,10,100

TAB. 10 – Tableau comparatif des caractéristiques.

Caractéristiques	Précision	Rappel	F1	AUR PR	RC	TE
$base\_acteurs$	0.94	0.69	0.77	0.64	29645	58sec
$base\_pays$	0.78	0.53	0.55	0.31	39490	1mn 1sec
$graphe\_acteurs$	0.95	0.69	0.76	0.64	29850	55sec
$graphe\_pays$	0.80	0.52	0.54	0.32	39585	1mn 1sec



(a) Risque du coût. (b) Courbe entre la précision et le rappel.

FIG. 3 – Définition du seuil de probabilité d'une transaction frauduleuse.

4.3 Risque du coût de prédiction

Après avoir observé l'impact des caractéristiques, le risque du coût de notre $model\_split$ est minimal avec l'algorithme *XGBoost* de 29280, pour rappel, le seuil probabilité à partir duquel une transaction est frauduleuse est fixé à 0.5 par défaut. À partir des prédictions du modèle et des probabilités d'appartenance à la classe frauduleuse, nous calculons le risque du coût pour chaque seuil de probabilité entre 0 et 1 avec un intervalle de 0.001. Les résultats sont présentés sur la figure 3a. Le coût minimal est obtenu lorsque la probabilité est de 0.45, et le CR associé est égal à 28690. La figure 3b nous indique que le rappel et la précision du modèle, associé à une probabilité de 0.45 pour considérer une transaction frauduleuse, sont respectivement de 0.93 et 0.71 qui nous donne un F1 de 0.78. Nous avons donc réduit le seuil de probabilité à 0.45 pour réduire le risque du coût de notre modèle, tout en gardant la même qualité de prédiction.

4.4 Résultats et discussions

Les résultats de nos expérimentations nous montrent que l'algorithme *XGBoost* est le plus adapté à notre structure de données, car il obtient le meilleur F1 (0.78) et AUC-PR(0.66). Les $model\_split$ et $model\_base$ possèdent sensiblement les mêmes résultats comme le montre le tableau 8 avec une différence de temps de 29 secondes pour l'apprentissage en faveur du $model\_split$, grâce à la division du jeu de données. Le tableau 10, indique le fort impact des caractéristiques : $base\_acteurs$ et $graph\_acteurs$ sur l'apprentissage. Le comportement des acteurs et leurs interactions sont importants pour la détection de fraude. Les caractéristiques sur les pays sont moins impactantes, mais les meilleurs résultats sont obtenus quand elles sont combinées avec celles des acteurs. Enfin, nous avons minimisé le risque du coût (RC) de notre $model\_split$, à 28690 qui était de base à 29285, grâce au seuil de probabilité fixé à 0.45 pour considérer une transaction frauduleuse, et cela, en conservant le même F1 (0.78).

5 Conclusion

Les techniques d'apprentissage automatique peuvent être de réels atouts pour les institutions financières afin de lutter contre les fraudes. Un apprentissage adapté, avec le calcul de caractéristiques, amène des résultats satisfaisants. À travers notre méthode, nous avons détaillé les caractéristiques calculées sur des transactions SWIFT. Ensuite, nous avons divisé notre jeu de données selon les devises et pays des transactions et entraîné un modèle pour chaque division. Ce qui nous a permis d'avoir des résultats similaires avec un temps d'entraînement plus court. Nous avons également étudié l'impact des caractéristiques calculées qui nous informe que celles basées sur les acteurs sont importantes pour l'apprentissage. Nous avons proposé une formule de risque du coût d'un modèle et nous avons réduit ce risque en choisissant un nouveau seuil de probabilité à partir duquel une transaction est considérée frauduleuse. Pour nos travaux futurs, nous souhaitons étudier l'impact des intermédiaires sur les transactions frauduleuses. Nous voulons aussi inclure l'aspect temporel des transactions dans le calcul des caractéristiques.

Références

Adewumi, A. O. et A. A. Akinyelu (2017). A survey of machine-learning and nature-inspired based credit card fraud detection techniques. *International Journal of System Assurance Engineering and Management 8*(2), 937–953.

Al-Hashedi, K. G. et P. Magalingam (2021). Financial fraud detection applying data mining techniques : A comprehensive review from 2009 to 2019. *Computer Science Review 40*, 100402.

Alfaiz, N. S. et S. M. Fati (2022). Enhanced credit card fraud detection model using machine learning. *Electronics 11*(4), 662.

Bahnsen, A. C., A. Stojanovic, D. Aouada, et B. Ottersten (2013). Cost sensitive credit card fraud detection using bayes minimum risk. In *2013 12th international conference on machine learning and applications*, Volume 1, pp. 333–338. IEEE.

Bhattacharyya, S., S. Jha, K. Tharakunnel, et J. C. Westland (2011). Data mining for credit card fraud : A comparative study. *Decision support systems 50*(3), 602–613.

Borisov, V., T. Leemann, K. Seßler, J. Haug, M. Pawelczyk, et G. Kasneci (2021). Deep neural networks and tabular data : A survey. *arXiv preprint arXiv :2110.01889*.

Breunig, M. M., H.-P. Kriegel, R. T. Ng, et J. Sander (2000). Lof : identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 93–104.

Chergui, H., L. Abrouk, N. Cullot, et N. Cabioch (2022). Détection de fraude financière dans un système de transactions interbancaires. *INFORSID'22*, 141–156.

Huang, D., D. Mu, L. Yang, et X. Cai (2018). Codetect : Financial fraud detection with anomaly feature detection. *IEEE Access 6*, 19161–19174.

John, H. et S. Naaz (2019). Credit card fraud detection using local outlier factor and isolation forest. *Int. J. Comput. Sci. Eng 7*(4), 1060–1064.

Khatri, S., A. Arora, et A. P. Agrawal (2020). Supervised machine learning algorithms for credit card fraud detection : a comparison. In *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pp. 680–683. IEEE.

Knobel, A. (2019). Swift data can be a global vantage point for tackling global money laundering.

Le Khac, N. A. et M.-T. Kechadi (2010). Application of data mining for anti-money laundering detection : A case study. In *2010 IEEE international conference on data mining workshops*, pp. 577–584. IEEE.

Liu, F. T., K. M. Ting, et Z.-H. Zhou (2008). Isolation forest. In *2008 eighth ieee international conference on data mining*, pp. 413–422. IEEE.

Lopez-Rojas, E., A. Elmir, et S. Axelsson (2016). Paysim : A financial mobile money simulator for fraud detection. In *28th European Modeling and Simulation Symposium, EMSS, Larnaca*, pp. 249–255. Dime University of Genoa.

Meng, C., L. Zhou, et B. Liu (2020). A case study in credit fraud detection with smote and xgboost. In *Journal of Physics : Conference Series*, Volume 1601, pp. 052016. IOP Publishing.

Mishra, S. et M. Chawla (2019). A comparative study of local outlier factor algorithms for outliers detection in data streams. In *Emerging Technologies in Data Mining and Information Security*, pp. 347–356. Springer.

Varmedja, D., M. Karanovic, S. Sladojevic, M. Arsenovic, et A. Anderla (2019). Credit card fraud detection-machine learning methods. In *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)*, pp. 1–5. IEEE.

Whitrow, C., D. J. Hand, P. Juszczak, D. Weston, et N. M. Adams (2009). Transaction aggregation as a strategy for credit card fraud detection. *Data mining and knowledge discovery 18*(1), 30–55.

Xuan, S., G. Liu, Z. Li, L. Zheng, S. Wang, et C. Jiang (2018). Random forest for credit card fraud detection. In *2018 IEEE 15th international conference on networking, sensing and control (ICNSC)*, pp. 1–6. IEEE.

Summary

The fight against financial fraud is a major challenge for financial institutions. In recent years, several approaches based on the analysis of banking transactions have been proposed for fraud detection. In this work, we propose an approach based on machine learning techniques for detecting financial fraud in international and interbank transactions of the SWIFT network. The learning of the model is carried out with new characteristics calculated from the specificities of SWIFT transactions. We define a risk measure of the cost on the predictions of a model we wish to reduce with our methodology. The experiments were conducted on a real dataset and validated by experts in the field.

Index