Promotion 2022

3ème année, Majeur DIA

Atos - Avantix

LBDS FR MCS ADE EW Signal Proc AI Telco

655 avenue Galilée - 13799 AIX EN PROVENCE

Stage - Ingénieur Machine Learning - Active Learning

MAZOUZ REIHAN

Tuteur école : FAUBERTEAU Frederic

Tuteur entreprise : BELAFDIL Chakib

Remerciements

Je tiens à remercier toutes les personnes qui ont contribué au succès de mon stage.

Tout d'abord, mon tuteur d'entreprise, Chakib Beladil, pour son soutien, sa compétence et sa pertinence apportés tout au long de mon stage.

Ensuite, mes collègues stagiaires et employés du pôle BDS pour leur bonne humeur quotidienne qui a permis de rendre l'ambiance plus agréable.

Aussi, je tiens à remercier Busi Jean Daniel, directeur de projet, pour le suivi hebdomadaire de mon avancée et ses remarques régulières afin d'améliorer mes travaux

Je voudrai également remercier Charton Philippe, mon manager, pour son attention particulière sur mon intégration dans l'entreprise et dans le projet de stage.

Enfin, une reconnaissance particulière à l'ensemble du groupe Atos et plus particulièrement la société Avantix de m'avoir fait confiance et permis de réaliser mon stage dans ses locaux à Aix-en-Provence.

Table des matières

Liste des tableaux

Liste des figures

Glossaire et abréviations

Terme	*Définition*
Impulsion	Une impulsion est une onde électromagnétique émise par un radar impulsionnel
Emission	Une émission est une suite d'impulsions ayant la même fréquence au sein d'une direction d'arrivée (DA)
Sous mode	Chaque émission est associée à un sous mode imposant les caractéristiques de chaque impulsion. Les caractéristiques des sous modes sont listées dans le « tableau des sous modes »
DI	Abréviation de « Durée d'Impulsion ». On retrouvera aussi le terme en anglais : PW (« Pulsation Width »)
PRI	Le PRI (Pulsation Rate Impulsion) est le motif des durées séparant 2 impulsions successives. Elle caractérise le RADAR.
Toa	Sigle de "Time Of Arrival"
dToa	La dToa est la différentielle entre les temps d'arrivées de chaque impulsion au sein d'une même émission. Elle est calculée à partir de la Toa et permet de reliée l'impulsion à la PRI de son sous mode.
DA	Sigle de « Direction of Arrival ».
Deep Learning	Les techniques d'apprentissage profond constituent une classe d'algorithmes d'apprentissage automatique.
Outliers	Une donnée aberrante est une valeur ou une observation qui est « distante » des autres observations effectuées sur le même phénomène.
Feature	Propriété mesurable ou caractéristiques d'un phénomène.
Labéliser	Etiqueter ou labéliser une donnée signifie lui associer un label afin de pouvoir la classifier.
Couche Dense	Couche de base d'un réseau de neurone. Ile prend simplement une entrée, et applique une transformation basique avec sa fonction d'activation.
Facteur de dropout	Le dropout est une méthode de régularisation qui permet contrer l'overfitting. Le terme « Dropout » fait référence à la suppression de neurones dans les couches d'un modèle de Deep Learning. Le facteur de dropout est donc le facteur de désactivation des neurones par couche lors de l'apprentissage.
Batch	Hyperparamètre d'un réseau de neurone définissant le nombre de donnée à considérer avant la mise à jour des paramètres du modèle.
Epoch	Le nombre d'époch est un hyperparamètre définissant le nombre de fois que le modèle va rencontrer l'ensemble des données d'apprentissage.
Transfer learning	Transfert de connaissance d'un modèle travaillant sur une tâche à une autre.
Data augmentation	Technique utilisée pour augmenter la quantité de donnée en ajoutant des copies modifiées de données existantes ou des nouvelles données crées synthétiquement

Terme	*Définition*
Imbalanced Learning	Apprentissage automatique considérant le déséquilibre des classes dans la distribution d'un ensemble de données.
BNN	Bayesian Neural Network
Budget	Nombre maximum de requêtes autorisées
CNN	Convolutional Neural Network
Decision Tree	Decision Tree (arbre de décision) est un algorithme permettant de classer en construisant des seuils optimaux sur les caractéristiques d'entrées. Algorithme de base utilisé par le Random Forest.
DeepAL	Deep Active Learning est une méthode d'active learning utilisant des modèles de Deep Learning.
IA	Intelligence Artificielle
k-NN	K Nearest Neighbors (k plus proches voisins) est un algorithme de type « instance based » ; lors de l'étape d'inférence l'entrée est étiquetée selon les étiquettes de ses k plus proches voisins (habituellement au sens de la distance euclidienne) par vote majoritaire.
LSTM	Long Short-Term Memory est un réseau de neurones de type récurrent (voir RNN) qui a deux sorties, l'une qui transmet l'information court-terme et l'autre long-terme. Réseau qui ne corrige qu'en partie l'absence de mémoire à long-terme.
MLP	Multi-Layer Perceptron est un type de réseau composé uniquement de perceptrons agencés en couches. Il n'y a pas de définition universelle mais la communauté s'accorde à dire qu'un réseau est profond (deep) quand il possède au moins deux couches cachées.
Oracle	Humain ou machine labelisant la donnée
Outlier	Un outlier pourrait se traduire par : donnée aberrante. C'est une valeur ou une observation qui est « distante » des autres observations effectuées sur le même phénomène.
Overfitting	Le terme d'overfitting est employé pour décrire le surapprentissage d'un modèle statistique lors d'un apprentissage automatique.
Perceptron	La plus simple des architectures d'un réseau de neurones artificiel, celui-ci ne possède qu'une seule couche et qu'une seule sortie. Il connecte la sortie aux entrées en appliquant une combinaison linéaire des entrées (à partir des poids w_i), l'ajout d'un biais (scalaire noté b) et une fonction d'activation qui doit être non-linéaire, différentiable et monotone (notée σ). La formule suivante résume le calcul réalisé par un perceptron : $o = \sigma(\sum_{i=1}^{N} x_i . w_i + b)$ où o est la sortie et N le nombre d'entrées.
Random Forest	Random Forest (Forêt aléatoire) est un algorithme permettant de créer plusieurs arbres choisissant aléatoirement les caractéristiques à utiliser pour chacun d'eux. Voir Decision Tree.
Requête	Demande d'étiquetage d'une donnée envoyée par l'Oracle
RNN	Recurrent Neural Network est un type de réseau capable de traiter des données d'entrées séquentielles. La sortie du réseau est concaténée avec la séquence d'entrée suivante. RNN fait référence au réseau éponyme le plus triviale mais également à tous types de réseaux récurrents comme le LSTM et le GRU.
Stopping Criteria	Condition stoppant le processus d'Active Learning

Terme	*Définition*
SVM	Support Vector Machine. Algorithme de classification très utilisé pour ses bonnes performances et sa rapidité d'exécution. Son principe est de rechercher des hyperplans séparateurs entre les classes qui maximisent la marge. Le SVM a pour principal défaut de ne pas converger lorsque le nombre d'entrée est grand. Le SVM a été pensé pour résoudre des problèmes linéaires mais grâce au « kernel trick », l'algorithme peut résoudre les problèmes non linéaires.
Méthodes ensemblistes	Méthode consistant à combiner plusieurs modèles de Machine Learning afin de diminuer la variance lors de la décision. Les méthodes de Bagging et de Bootstrap sont des méthodes ensemblistes
Score de rappel	Proportion des items pertinents proposés parmi l'ensemble des items pertinents. La formule est donnée par : $\frac{True_{Positive}}{True_{Positive}+False_{negative}}$
Score de précision	Proportion des items pertinents proposés parmi l'ensemble des items proposés. La formule est donnée par : $\frac{True_{Positive}}{True_{Positive}+False_{positive}}$
F1 score	Moyenne harmonique du score de rappel et du score de précision

Résumé

Dans le cadre d'un projet de classification d'émissions de signaux radars à impulsions, des opérateurs humains labélisent des données de façon manuelle. De ce fait, l'intégration d'un processus d'Active Learning permettrait de limiter le temps humain alloué sans rogner les performances du modèle entrainé.

L'Active Learning est un processus semi supervisé mettant en jeux des oracles humain ou machine labélisant la donnée. Le principe est de minimiser le nombre de données labélisées passées en entraînement d'un algorithme de Machine/Deep Learning tout en conservant au maximum ses performances. Mon stage doit ainsi permettre de répondre aux interrogations liées à la faisabilité de l'intégration de ce processus au sein du projet.

Après avoir étudié les différentes stratégies existantes dans l'état de l'art, les plus pertinentes dans le cas du système ont été retenues. Elles ont été implémentées puis testées sur les différentes bases de données. Les résultats obtenus sont conformes aux attentes et la stratégie de sélection par incertitude, ayant les meilleurs résultats, a été celle retenue. En effet, on estime un gain d'au moins 50% de données labélisées et allant jusqu'à 90% si la base de données est suffisamment complète.

De manière générale, il a été constaté que l'Active Learning est un processus très efficace lorsque l'on s'approche du plateau d'apprentissage des algorithmes de Machine Learning. L'écriture d'un code générique permet de rapidement mettre en place le processus d'Active Learning quel que soit le projet de classification par algorithmes de Machine Learning et d'obtenir des résultats. Néanmoins, le code doit être testé sur des algorithmes de Deep Learning avant de conclure quant à son efficacité dans ces conditions.

Enfin, mon stage s'est conclu sur un travail portant sur l'étude des réseaux siamois pour le regroupement d'émissions radars. L'objectif est de comparer les performances avec celles d'un modèle de Machine Learning qui est censé être livré en production dans le cadre du même projet que celui de classification ci-dessus.

Après avoir implémenté et optimisé ce type d'architecture, celle-ci a effectivement su réaliser la tâche de regroupement avec des performances comparables voire meilleures que celles du modèle existant.

1 Introduction

La mission s'intéresse au regroupement et à la classification de signaux radars à impulsion. Dans le cadre de ce projet, les impulsions radars sont reçues successivement par plusieurs récepteurs. Ces derniers peuvent recevoir dans une période donnée des impulsions de plusieurs radars provenant de différentes directions d'arrivée. Il est donc important de savoir distinguer les impulsions afin de regrouper celles qui appartiennent au même émetteur et de reconnaitre la classe du radar.

Dans le problème de classification, des opérateurs humains labélisent manuellement les données une à une. Dans ce contexte, réduire la quantité de données labélisées est d'une importance cruciale afin de gagner en temps de travail humain. L'Active Learning est la solution envisagée pour répondre à cette problématique. L'intégration d'un tel processus représente la première partie du stage.

Dans un second temps, les réseaux siamois sont testés pour répondre au problème de regroupement des émissions radars. Les résultats seront comparés avec les algorithmes de Machine Learning classiques qui ont été retenus et qui ont des bonnes performances.

Dans ce rapport seront présentées les différentes étapes, notamment un état de l'art sur l'Active Learning et une analyse exploratoire des données, permettant d'atteindre les objectifs cités ci-dessus.

2 Description de la mission

2.1 Présentation de l'entreprise

2.1.1 Atos

La société Avantix est une filiale du groupe Atos. Atos est une entreprise de services du numérique (ESN) française, né en 1997 de la fusion deux entreprises de service informatiques, Axime et Sligos. Elle fait partie des 10 plus grandes ESN au niveau mondial, avec un chiffre d'affaires annuel de près de 11 milliards d'euros en 2019 et environ 110 000 employés répartis dans 73 pays.

Devenue Atos Origin en 2000, à la suite de la fusion entre Atos et Origin, elle reprend le nom d'Atos en 2011 après l'acquisition de Siemens IT Solutions and Services. Depuis son rachat de Bull, Atos est le n°1 européen du Big Data, de la Cybersécurité et des supercalculateurs. Elle est notamment partenaire officielle des jeux olympiques de Paris 2024.

Ci-dessous un historique des différentes acquisitions du groupe Atos-Origin.

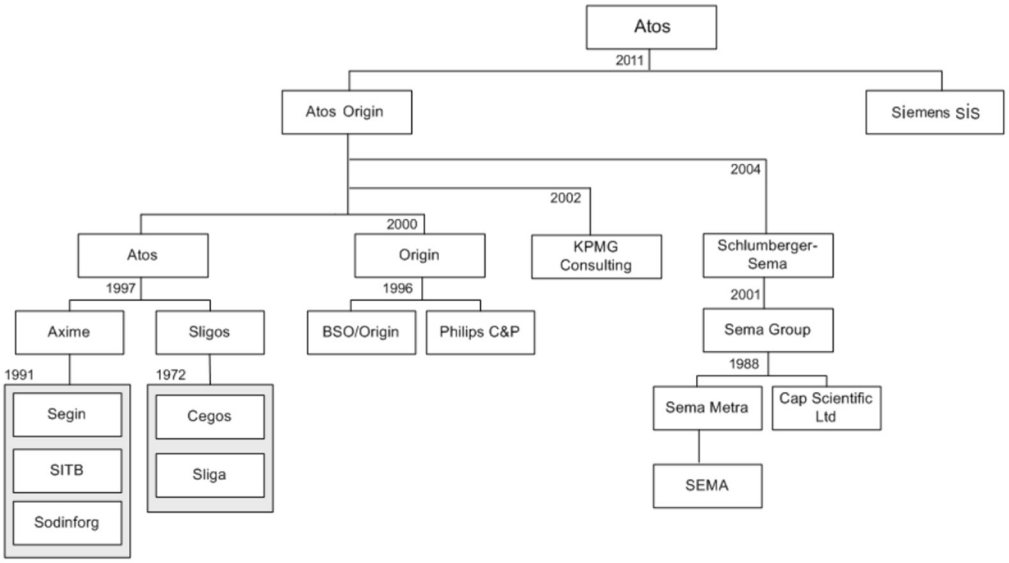


Figure 1 : Historique des acquisitions du groupe Atos-Origin

Les métiers d'Atos sont organisés en trois divisions :

- *Infrastructure & Data Management* : la gestion des centres de données (datacenters), du service desk et des communications unifiées ;

- *Business Applications & Platform Solutions* : le conseil et l'intégration de systèmes informatiques ;
- *Big Data & Cybersecurity* : la fabrication et gestion de serveurs et supercalculateurs et la cybersécurité ;

Les services d'Atos sont regroupés au sein d'une *Digital Transformation Factory*, s'appuyant sur quatre piliers :

- *Cloud* : la mise en place et gestion de clouds privés et hybrides ;
- Espace numérique de travail (*Digital workplace* en anglais) : la gestion du lieu de travail numérisé et les communications unifiées ;
- *SAP HANA* **(en)** : l'implémentation et gestion du progiciel de gestion intégré SAP ;
- *Atos Codex* : l'analyse et gestion des données de bout en bout (business analytics, analyse prédictive).

Une spécificité d'Atos est sa spécialisation dans les supercalculateurs et le calcul quantique. Atos est le seul fabricant européen de supercalculateurs, face aux États-Unis, à la Chine et au Japon.

Le secteur BDS d'Atos représente environ la moitié de son chiffre d'affaires global et est évalué entre 2 et 3 milliards d'euros. Tout comme les secteurs des supercalculateurs et du calcul quantique, Atos est soutenu par l'état pour ses activités lié au BDS.

2.1.2 Avantix

Avantix est une entreprise française qui conçoit des systèmes de haute technologie pour des secteurs industriels stratégiques dans le monde entier.

Avantix est située à Aix-en-Provence et fondée en 1979 sous le nom d'I2E puis d'Amesys. Étant initialement un groupe

• D'étude et développement dans les domaines des Telecom et de la Défense

• Une société de services en ingénierie informatique,

Elle fut affiliée au groupe Crescendo Industries en 2007. Le 1er janvier 2010, la société Crescendo Industries entre dans le capital du groupe Bull, société française spécialisée dans l'informatique professionnelle. Cette intégration a entrainé automatiquement la présence d'Avantix au sein du groupe. En 2014, Bull est racheté par Atos, Avantix anciennement rattaché au groupe Bull est alors affectée au pôle MCS, pour Mission Critical Systems(voir organigramme en haut à droite), lui-même intégrée au pôle BDS, pour Big Data & Security. Développant des applications dans le domaine de la technologie de guerre électronique, elle est en concurrence directe avec des entreprises telles que Thalès, Airbus Défense, Naval Group ou Inéo Défense.

Ses solutions combinent l'électronique, la RF (radio fréquence), le traitement du signal et les technologies de l'information, offrant une innovation rapide à ses clients. Les équipes d'Avantix permettent un déploiement rapide et une maintenance à long terme de ses solutions.

Avec plus de 25 ans d'expérience en EW (Electronic Warfare), Avantix est un pionnier depuis la création du premier système d'interception de communication tactique il y a 15 ans et n'a cessé depuis de créer de nouveaux systèmes.

Les technologies de guerre électronique développées par Avantix se concentrent sur la protection des forces armées. Avantix fournit des composants modulaires ainsi que des systèmes de bout en bout, garantissant un flux de données fluide des capteurs au traitement et à la décision. Ses solutions sont conçues pour être robustes, étendant la portée opérationnelle des missions.

Avantix est divisée en deux grands pôles d'activités. Le premier se nomme ALSE (Air Land Sea Electronics) et le second dans lequel j'ai été accueilli se nomme Electronic Warfare.

En interne, les collaborateurs sont répartis dans plusieurs pôles d'expertise métier (Logiciel, FPGA, RF et Cellulaire : voir organigramme). Cette organisation permet aux différents chefs de projets de pouvoir répartir convenablement les tâches contribuant par conséquent au bon déroulement et à l'avancement rapide des projets de l'entreprise.

L'ensemble des technologies développées par Avantix reposent sur divers domaines. Le pôle dans lequel j'ai travaillé est le pôle BDS (pour Big Data Security) et plus spécifiquement dans celui portant sur « Signal Processing, AI and Telecommunication ».

2.1.3 Organigrammes

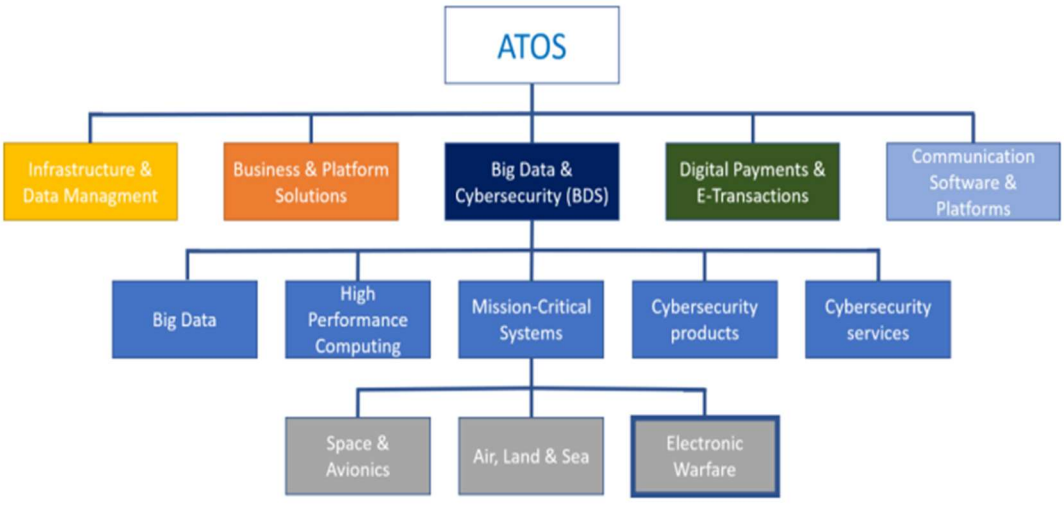


Figure 2 : Organisation structurelle d'Atos

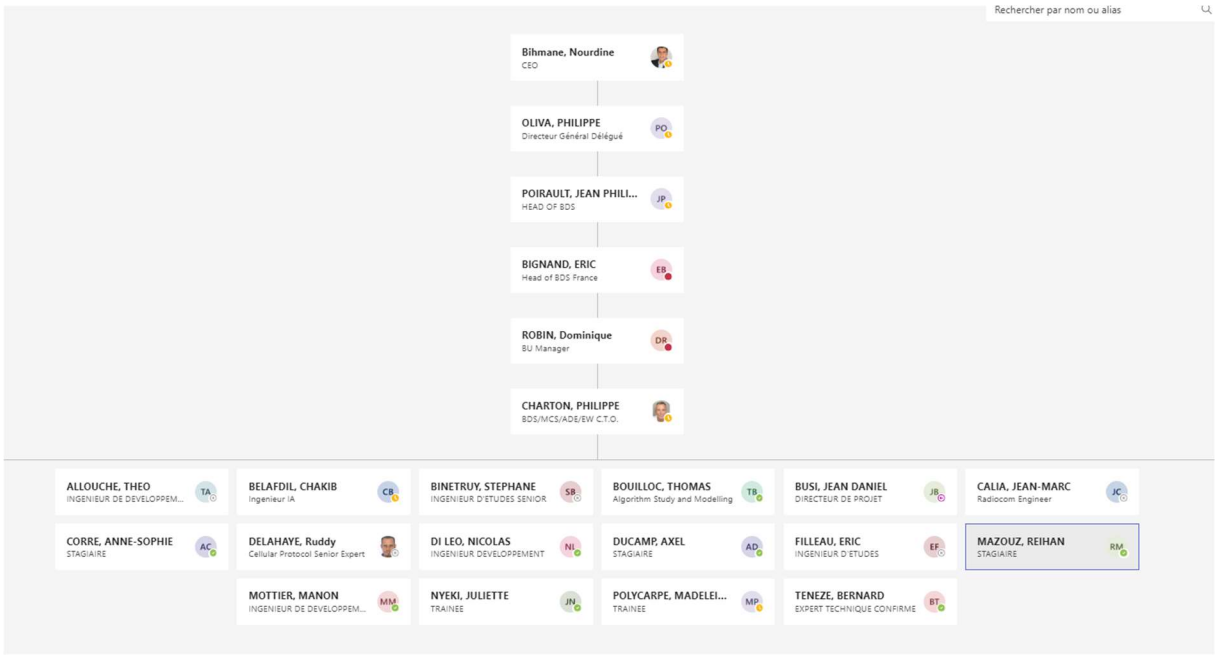


Figure 3: Organisation branche BDS

2.2 La mission

2.2.1 Cadre

Dans le cadre de ce stage, je dispose d'un bureau au sein dans l'open space RF (radio-fréquence) à Avantix Aix-en-Provence.

Je dispose aussi d'un ordinateur Lenovo fonctionnant sous Windows fournit par Atos ainsi que d'un serveur de 376 Go en RAM et composé de deux GPU Tesla v100 de 32 Go. J'ai un compte office 365 et accès à diverses ressources internes à Atos tel que myAtos ou le Atos Learning Center.

Je travaille en autonomie sur le sujet d'Active Learning mais une équipe constituée d'un peu moins d'une dizaine de personne travaillent sur le projet dans sa globalité. Un autre stagiaire travaille également sur la tâche de classification et a dû réaliser lui aussi une analyse exploratoire des données. Dans ce contexte, nous avons partiellement travaillé ensemble.

Une réunion hebdomadaire d'une heure avec mon maître de stage a lieu le lundi matin de 9h30 à 10h30 sur Teams. Il reste néanmoins disponible le restant de la semaine si nécessaire.

2.2.2 Le projet

Le projet dans lequel j'ai pu travailler est un système de regroupement et de classification de signaux radars à impulsion. On le dénommera système X dans la suite de ce rapport pour des raisons de confidentialité. Pour ces mêmes raisons, le fonctionnement du système dans sa globalité ne sera pas explicité en détail et certains noms et variables seront intentionnellement falsifiés.

Pour aborder le sujet de ce projet, il faut tout d'abord étudier le fonctionnement d'un radar à impulsion.

Chaque radar est constitué d'un émetteur produisant un signal radiofréquence (RF) et d'un récepteur. Le signal se réfléchissant sur la cible, il est réceptionné par le radar qui l'émet et la différence entre le temps du début de l'émission et le temps du début de la réception permet de déterminer la distance radar /cible.

Un radar à impulsion émet une série d'impulsions pour constituer une émission. Chaque impulsion n'est pas réceptionnée et le taux d'impulsions manquées par le récepteur s'appelle « taux de mitage ».

Le signal RF émis a des caractéristiques spécifiques qui le différencient de tous les autres signaux et définissent ses capacités et ses limites. La durée d'impulsion (DI), la fréquence de répétition d'impulsion (PRI) et la fréquence moyenne de l'impulsion (simplement fréquence) sont, entre autres, toutes des caractéristiques du signal radar déterminées par l'émetteur radar. La définition des termes utilisés pour décrire ces caractéristiques est essentielle pour comprendre le fonctionnement du radar.

La DI est la durée durant laquelle l'émetteur envoie de l'énergie RF sous la forme d'une impulsion.

La PRI (période de répétition des impulsions) est l'intervalle de répétition des impulsions et est définie les durées entre le début de deux impulsions successives.

Aussi chaque impulsion est caractérisée par une fréquence moyenne. Il convient de préciser que les radars à impulsion étudiés ne peuvent pas avoir un delta en fréquence d'émission supérieur à 1.2GHz.

La classe d'un émetteur radar s'appelle un sous-mode. Chaque sous mode est déterminé, entre autres, par une liste de valeurs de fréquences moyennes, de PRI et de DI. L'objectif étant de classifier des émissions radars, celles-ci sont constituées de séries d'impulsions toutes appartenant au même sous mode, déterminer ce dernier permet de déterminer la classe de l'émission.

On considère que deux émetteurs localisés dans une même direction d'arrivée ne peuvent fonctionner avec le même sous-mode afin d'éviter les problématiques d'interférence.

Ainsi le travail de regroupement d'émission consiste à regrouper les émissions qui appartiennent au même émetteur. Celui de classification consiste à reconnaître l'émetteur d'où sont issues les émissions.

Ci-dessous une illustration du processus d'émission et de réception des émissions.

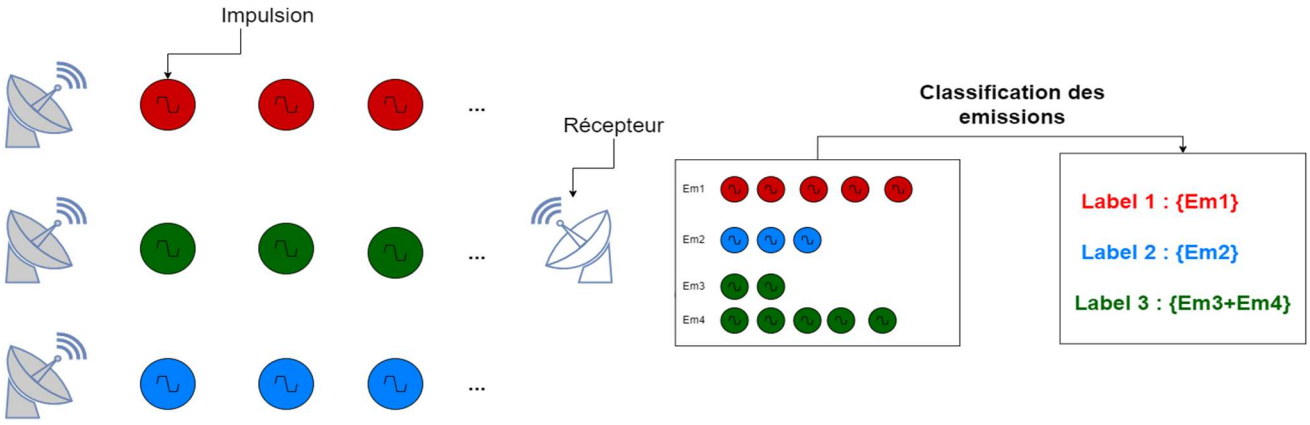


Figure 4: Réception et classification des émissions

2.2.3 Objectif de stage

L'objectif principale du stage est d'étudier l'intégration d'un processus d'Active Learning dans le système X, en effet, les données radars recueillis doivent être labélisées par des opérateurs humains. Ces labélisations peuvent être très couteuse en temps et en coût pour le client, notamment si la quantité de donnée à labéliser est importante. Pour ces raisons, l'Active Learning serait un moyen efficace pour drastiquement diminuer la quantité de donnée à labéliser par les opérateurs.

En plus d'étudier les processus d'Active Learning, je me suis intéressé en fin de stage à la tâche dite de « regroupement des émissions » par des réseaux siamois.

Ci-dessous un tableau récapitulatif des différentes tâches exécutées durant mon stage (certaines d'entre elles ont été réalisées en parallèles) :

Tableau 1 : Résumé des objectifs de stage

Dénomination de la tâche	Durée de la tâche	Parmi les objectifs de stage ?
Etat de l'art sur l'Active Learning	1 mois	Oui
Analyse exploratoire de la base de donnée 2.1	1 mois et 2 semaines	Oui
Application de l'Active Learning sur la base 2.1	1 mois	Oui
Analyse exploratoire des classes 2.1	2 semaines	Non
Application de l'Active Learning sur la base 3.0	1 mois	Oui
Confirmation des résultats	3 semaines	Oui
Regroupement par réseaux Siamois	1,5 mois	Non

2.2.4 Contenu du stage

2.2.4.1 Etat de l'art sur l'Active Learning

L'Active Learning est un procédé semi-supervisé où un oracle est sollicité pour labéliser des données. L'objectif est de maximiser les performances d'un modèle de Machine Learning en ayant minimisé la quantité de données labélisées fournies lors du processus d'apprentissage.

Les principes fondamentaux de l'Active Learning sont :

- Certaines données sont plus importantes que d'autres lors du processus d'apprentissage
- Labéliser une donnée n'est pas nul en coût

Afin d'appréhender au mieux l'intégration d'un processus d'Active Learning au sein du système X, il a fallu étudier dans sa globalité l'état de l'art de l'Active Learning et du Deep Active Learning. Ci-dessous nous présenterons un résumé de cet état de l'art.

Tout d'abord, il existe différents processus d'Active Learning :

- *Membership Query Strategy :* L'apprenant désigne la zone confuse pour lui, créer une donnée dans cette zone et demande à l'oracle de la labéliser. C'est un processus qui peut amener à des situations ambiguës où l'oracle ne sait pas labéliser la donnée générée. Au contraire c'est un processus prometteur dans le cas où l'apprenant est dans un contexte exploratoire.
- *Stream Based Selective Sampling :* Les données sont passées successivement à l'apprenant et c'est à lui de désigner si la labélisation de la donnée est pertinente ou non. C'est un processus peu couteux en puissance de calcul est donc nécessairement adapté aux applications mobiles ou de type systèmes embarqués.
- *Pool Based Selective Sampling :* L'ensemble de données non labélisées est accessible par l'apprenant. C'est le processus le plus couramment utilisé. L'algorithme d'Active Learning, ayant accès à l'ensemble des données et non à un flux de donnée, aura théoriquement plus de facilité à effectuer la meilleure requête. Ainsi, le nombre de requête à réaliser peut diminuer drastiquement.

Le *Pool Based Selective Sampling* est le processus d'Active Learning le plus adapté au projet, la suite de l'étude se concentre donc sur ce processus.

On a mis en avant sept stratégies d'Active Learning qui sont résumés dans le tableau ci-dessous :

Tableau 2: Résumé des stratégies d'Active Learning

Query Strategy	Description
Uncertainty Sampling	Sélectionner les données dont le modèle a le plus de mal à labeliser. On distinguera plusieurs stratégies permettant de définir la confusion du modèle pour une donnée x.
Query by committee	Entraînement d'un comité de modèles, sélectionner les données confuses pour la majorité des modèles
Expected Model Change	Sélectionner la donnée qui implique un maximum de changement au modèle (maximiser la norme du gradient)
Expected Error Reduction	Entraîner le modèle sur l'ensemble de données labelisées plus tout couple (x,y) non labelisé, calculer l'erreur induite, sélectionner celui dont l'erreur est la plus faible
Variance Maximisation	Minimiser la variance induite par la labélisation
Density Weighted Methods	Se baser sur la répartition des données pour sélectionner celles qui sont dans des zones d'information
Batch Uncertainty Sampling	Sélectionner un ensemble de donnée incertain pour le model tout en minimisant leurs corrélations

Dans le cadre du projet, les stratégies retenues pour l'implémentation sont Uncertainty Sampling, Density Sampling, Batch Uncertainty Sampling et Query by committee car elles sont moins couteuses en termes de complexité en temps et, selon l'état de l'art, sont souvent moins performantes.

On distingue aussi l'Active Learning du Deep Active Learning qui est son équivalent appliqué au Deep Learning. Certaines questions sont légitimes :

- Comment calculer l'incertitude ?
- Comment réduire la quantité de données labélisées alors que le *Deep Learning* nécessite, par sa nature, une grande quantité de données en entraînement
- Comment rendre certaines méthodes d'Active Learning compatibles aux réseaux profonds (pas toujours les mêmes entrées et sorties)

Les réponses viennent de la considération de plusieurs techniques.

- Privilégier les méthodes de sélection par *batch* permet d'avoir une réelle cohérence dans les phases d'entrainement, en effet l'apprentissage dit « un à un » n'a que très peu d'impact sur un réseaux profond.
- Considérer les *epochs* lors des phases d'apprentissage et la corrélation entre les données sélectionnées permet un apprentissage plus effectif.

- Le *transfert learning* est primordial pour le gain de temps lorsque le nombre de requête est important.
- Une *data augmentation* sur les données labélisées permet d'augmenter la quantité de donnée à passer en entraînement du modèle tout maintenant le nombre de données labélisées.
-

Toujours dans le cadre du projet X et bien que détaillé dans le rapport sur l'état de l'art élaboré, le Deep Active Learning ne sera pas considéré.

2.2.4.2 Analyse Exploratoire

Un travail de nettoyage et d'analyse de la base de données a été réalisé afin de mieux appréhender les problématiques du projet.

Il existe deux bases de données d'émissions simulées permettant d'expérimenter nos algorithmes de classification et de regroupement. La base de données sur laquelle l'analyse exploratoire a été réalisée est la base 2.1. Celle-ci est constituée de 144 326 701 d'impulsions, chacune catégorisée par 12 paramètres. **Elle est entièrement générée artificiellement et nous ne disposons pas des données réelles.**

Les 12 caractéristiques sont détaillées dans le tableau ci-dessous.

Tableau 3 : Description des caractéristiques de la base d'impulsion

Caractéristique	*Définition*
isValid	Si l'émission réceptionnée est considéré comme valide
Toa	Mesure de l'instant de réception de l'impulsion
Freq	Fréquence moyenne
DI	Durée d'impulsion
Niv	Niveau
RSB	Rapport signal sur bruit
DOA	Référence de la direction d'arrivée de l'impulsion
Id_sm	Identifiant du sous mode associé à l'impulsion
files	Référence du fichier de stockage de la donnée

Un tableau des sous modes nous permet d'associer une émission à son sous-mode. Dans la base 2.1, le tableau des sous modes est constitué de 60 sous modes.

Les caractéristiques de l'impulsion permettant la classification sont la fréquence moyenne, la durée d'impulsion (DI) et la période de répétition des impulsions (PRI). La PRI est le temps écoulé entre le début de deux impulsions successives.

Le tableau des sous modes contient les valeurs pouvant être prises par ces caractéristiques pour chaque sous mode. Ces valeurs sont prises en suivant une loi qui peut être linéaire, sinusoïdale, aléatoire etc…

L'analyse exploratoire s'est réalisée tout au long du stage en fonction du besoin. Les différentes étapes ont été :

1. Découverte de la base de données
2. Nettoyage de la base de données
3. Etude de la dToa
4. Identification des sous modes similaires

2.2.4.2.1 Découverte de la base de données

La base de données des impulsions est constituée de 50% d'impulsions validées et 50% d'impulsions non validées. Après nettoyage, nous sommes à 44% des impulsions validées pour 55% non validées. Le terme validé signifie que l'impulsion est reçue par les trois récepteurs.

La distribution des impulsions par sous mode n'est pas équilibrée avec près de 70 millions d'impulsions pour le sous mode 61 contre 19 000 pour le sous mode 42 soit 49% de base de données contre 0.01%. Le nettoyage de la base de données ne permet pas de diminuer ce déséquilibre.

En termes d'émissions, la distribution est un peu plus équilibrée avec 6% de l'ensemble des émissions appartenant au sous mode 65 contre 0.2% pour le sous mode 7.

Pour les impulsions, on obtient 95% d'entre elles réparties sur 6 sous modes contre 54 quand il s'agit des émissions.

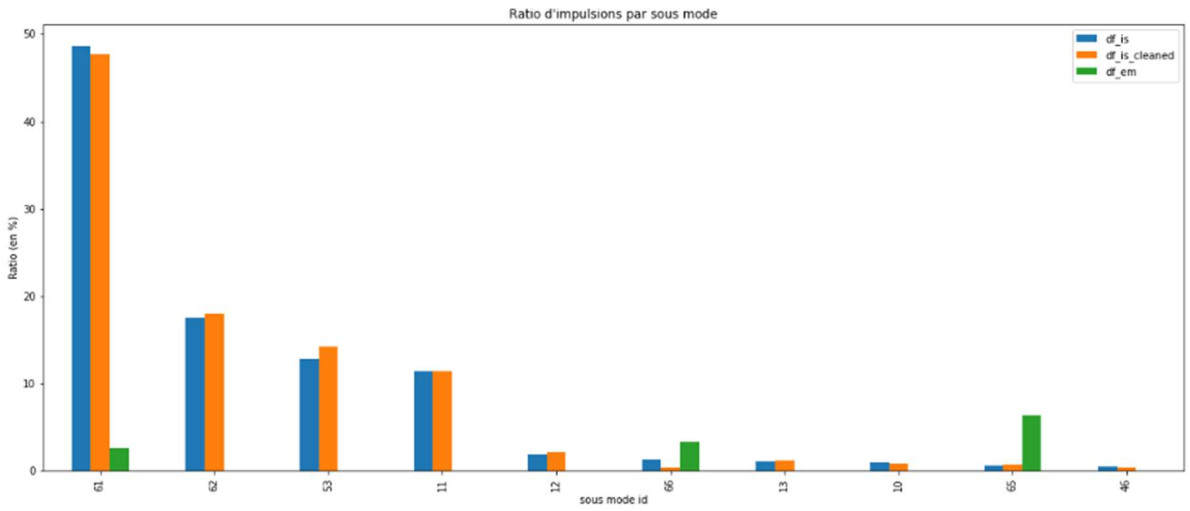


Figure 5 : Analyse Exploratoire / Ratio d'impulsions et d'émissions par sous mode pour les 10 sous modes avec le plus d'impulsions

2.2.4.2.2 Nettoyage de la base de données

Le nettoyage de la base de données consiste en trois parties :

- **Suppression des données inférieures à la DI minimale du tableau des sous modes**
  En tout, environ 15% des données sont inférieures à la DI minimale. En effet, cela est dû au phénomène des impulsions découpées. Lors de la réception d'une impulsion, si le niveau de l'intensité du signal est à la limite du seuil de détection alors on rencontre ce phénomène. Cela a pour conséquence de découpes une impulsion en artificiellement plusieurs impulsions ayant des DI inférieures à celles attendues. On conserve ainsi 85% des impulsions.

- **S'assurer que le delta en fréquence d'un sous mode ne soit pas supérieur à 1.2 GHz**
  Une contrainte qui semble se vérifier chez tous les émetteurs radar existants est qu'ils ne peuvent fonctionner sur des plages de fréquences allant au-delà de 1.2 GHz. A partir de cette contrainte, il est possible de nettoyer les points aberrants présents dans la base de données. En tout, c'est 252 impulsions supprimées via cette contrainte.

- **Vérifier le nombre d'impulsions par sous mode**
  Dans la base 2.1, bien que la répartition des impulsions par sous mode ne soit pas équilibrée avec presque la moitié des impulsions qui appartiennent à un seul sous mode et environ 95% de la totalité des impulsions qui appartiennent à seulement 5 sous modes. Néanmoins le sous mode ayant le moins d'impulsions en contient tout de même 435 qui est équivalents à 65 émissions, ce qui est jugé comme suffisant.

2.2.4.2.3 Etude de la dToa

La dToa (différentiel entre les temps d'arrivées) représente le délai entre la détection de deux signaux successifs. On le calcul en soustrayant les temps entre la réception d'impulsions successives au sein d'une émission. Elle permet de relier l'impulsion à la PRI de son sous mode dans le tableau des sous modes. La dToa serait égale à la PRI dans le cas où aucune impulsion ne serait manquée par le récepteur. En pratique ce n'est pas le cas et plusieurs impulsions successives peuvent être non réceptionnées ce qui explique des dToa avec des valeurs disproportionnées comparées à la PRI attendue.

Le but de cette étude est de comprendre pourquoi la dToa est mal exploitée par la classification des algorithmes de Machine Learning sur la base de données 2.1.

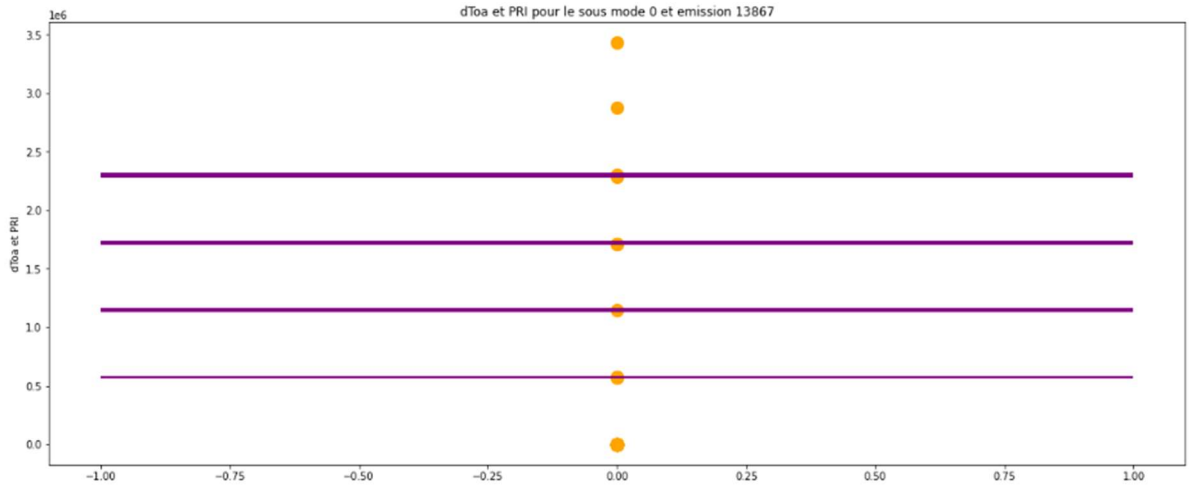
Dans les faits, il est difficile de superposer les dToas obtenues sur les PRI de chaque sous-mode avec celles attendues car il existe des dToas avec des dimensions inappropriées (trop élevées ou trop basses). Il faut donc préalablement supprimer les "outliers".

Une première approche pour pouvoir comparer les dToas de deux sous modes différents consiste en un affichage de densité de dToas. On peut ainsi observer qu'une divergence plus ou moins grande est

obtenue entre des sous modes distincts aux abscisses correspondantes aux multiples des PRI attendues par sous mode. Une étude statistique tels qu'un test de Kolmogorov Smirnov permet de confirmer que la dToa des sous modes suivent bien une loi Multi Gaussiennes de moyennes égales aux multiples des PRI attendues et de variances variables.

Cette technique est intéressante pour mesurer visuellement l'ampleur du bruit sur les dToas mais n'est pas efficace quand il s'agit d'associer une émission à un sous mode car le nombre d'impulsions contenues dans une émission est trop faible pour raisonner en terme de densité.

Une seconde approche, se concentrant sur une émission choisit aléatoirement parmi celles d'un sous mode sélectionné, permet d'obtenir une superposition visuelle des multiples des PRI attendues du sous mode et des dToa mesurées de l'émissions.



Bien qu'on arrive toujours à superposer les dToa de l'émission avec les multiples des PRI de son sous mode, on ne peut que très rarement classifier une émission grâce aux seules données de ses dToa. En effet, les PRI des différents sous modes sont proches et parfois se confondent. De plus, il faut ajouté à cela le bruit de la dToa lié à la réception de l'émission.

2.2.4.2.4 Identification des sous modes similaires

Afin de comprendre pourquoi certains sous mode sont confondus par les méthodes de classification, il faut étudier la similarité entre ces derniers.

On distinguera trois techniques de mesure de similarité différentes :

- Par hashage des features
- Par un calcul de distance par feature
- Par DTW

2.2.4.2.5 Identification des sous modes similaires grâce à une méthode de hachage

Pour déterminer les similarités entre les sous modes, nous utilisons une technique de hachage. Un hachage sha246 de chaque features par sous mode nous donne un identifiant unique du sous mode et de ses caractéristiques. On peut ainsi déterminer les sous modes ayant des caractéristiques identiques.

Une illustration en graphique permet de mettre en évidence ces similarités. Chaque sous mode est représenté par un nœud et deux nœuds sont liées si leurs identifiants uniques sont identiques. Travailler en termes de séquences de caractéristiques permet d'aller plus en profondeur dans la recherche de similarités entre sous mode.

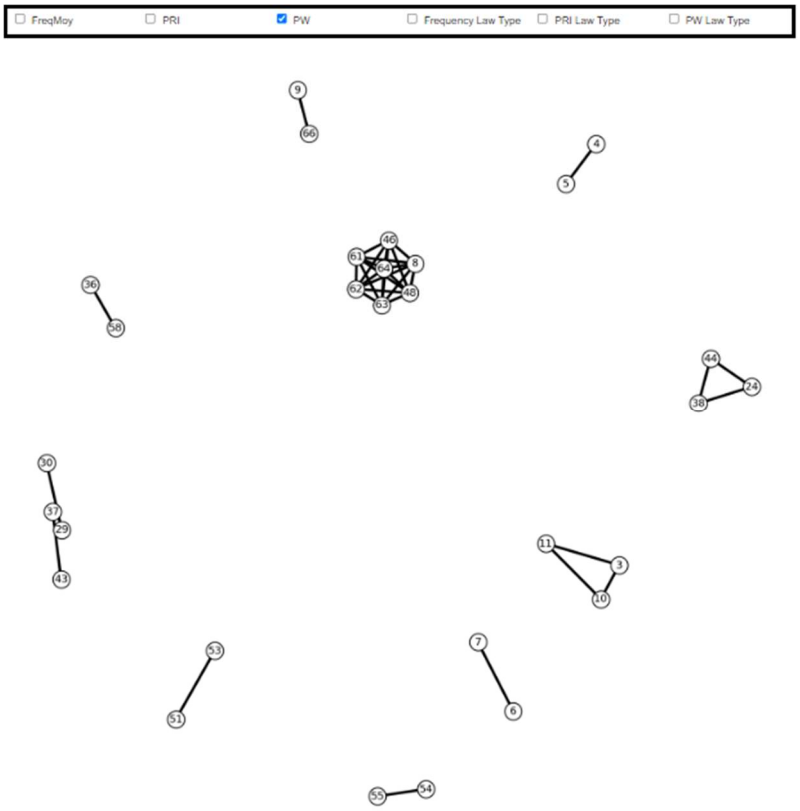


Figure 6 : Représentation avec NetworkX des sous modes similaires grâce à la méthode d'identification par hashage SHA246

Une limitation de cette technique est qu'elle ne prend pas en compte les petites divergences sur une caractéristique de deux sous modes différents : si deux sous modes ne sont pas strictement identiques pour une caractéristique préalablement sélectionnée, ils n'auront aucuns liens de similitudes sur cette caractéristique.

2.2.4.2.6 Calcul de distance entre sous mode pour mesurer la similarité

Afin d'obtenir une mesure de la similarité plus précise qu'avec la méthode de hachage précédente, un calcul de distance euclidienne entre les différentes features a été réalisé. La mesure de la distance entre les features de deux sous modes distincts se fait suivant la formule ci-dessous :

$$d(s1,s2) = \frac{\sum_i \sum_j \min_j \left(euclidian_{distance}(x_i, y_j)\right) + \sum_j \sum_i \min_i \left(euclidian_{distance}(x_i, y_j)\right)}{2}$$

Ici s1 et s2 sont respectivement la feature que l'on veut comparer pour le sous mode 1 et celle pour le sous mode 2. Les $\{x_i\}$ *sont les points de la feature du sous mode 1 et de même pour les* $\{y_j\}$ *et le sous mode 2.*

Les résultats obtenus permettent effectivement de déterminer les sous modes et leur degré de similarité par feature.

Cependant, cette méthode est imprécise dans la détermination du degré de similarité entre un sous mode et un autre car elle ne prend en compte qu'une seule feature à la fois. Concaténer ou fusionner les distances ne permet pas forcément d'obtenir des résultats cohérents car le poids à attribuer par feature n'est pas évident. Aussi cette méthode ne prend pas en compte les ordres de réception des impulsions au sein de l'émission.

Une *heatmap* permet de visualiser la distance normalisée entre chaque sous mode pour une (ou plusieurs) feature(s) sélectionnée(s).

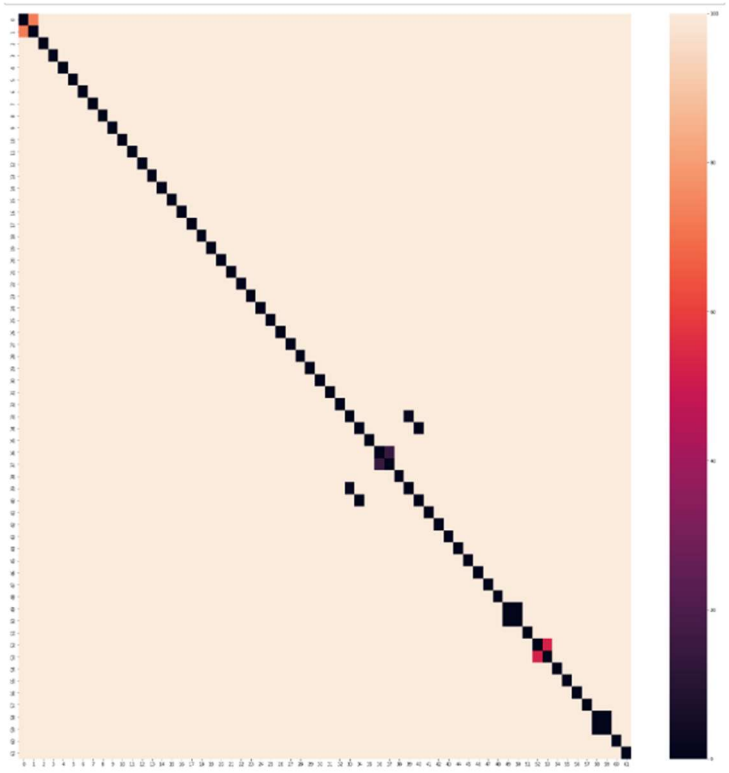


Figure 7 : Heatmap représentant le degré de similarité des sous modes

2.2.4.2.7 Calcul de distance par « Dynamic Time Warping » entre sous mode pour mesurer la similarité

Contrairement à la mesure de distance précédente, celle-ci se base sur une méthode de « Dynamic Time Warping ». La particularité de cette stratégie est qu'elle permet de mesurer la distance sur des séries temporelles de longueurs différentes et multivariées.

Afin de trouver le motif des séries temporelles, on considère le Plus Petit Commun Multiple (PPCM) entre la taille du motif de la fréquence moyenne et celle du motif de la PRI pour un sous mode donné. A partir de ce PPCM, on peut déterminer la séquence la plus longue considérant tous les couples fréquence moyenne et PRI possible en évitent les redondances. La DI étant constante pour l'ensemble des sous modes, on l'ajoute à nos couples à l'issue du processus.

2.2.4.2.8 Nombre d'impulsions minimales pour caractériser un sous mode

Un sous-mode est caractérisé par un ensemble de combinaisons entre fréquence moyenne, PRI et DI. Pour déterminer le nombre d'impulsion minimum pour qu'une émission est toutes les combinaisons possibles de son sous mode, il faut considérer la probabilité de « louper » certaines impulsions lors de la réception.

Soit A l'évènement « l'ensemble des impulsions possibles est obtenu ».

A_i L'évènement « obtenir l'impulsion i ». Alors on a $P(A \text{ à } l'étape\ q) = P(\bigcap_i A_i \text{ à } l'étape\ q)$

Où $q = kN + r$ avec N le nombre d'impulsions possibles qui vaut le PPCM entre le nombre de fréquence moyenne et de PRI du sous mode donné, k et r dans $\mathbb{N}$ et $0 \leq r < k$.

En supposant l'indépendance entre les A_i, on a : $P(\bigcap_i A_i \text{ à } l'étape\ q) = \prod_0^N P(A_i \text{ à } l'étape\ q)$

Or obtenir l'impulsion i à l'étape kN suit une loi géométrique, soit :

$$P(A_i \text{ à } l'étape\ q) = \begin{cases} \sum_{j=0}^{k} p(1-p)^j \ + (1-p)^k p, & si\ j \geq r \\ \sum_{j=0}^{k} p(1-p)^j \ , & si\ j < r \end{cases}$$

Ainsi, on peut numériquement obtenir le nombre minimum d'impulsions suffisant à la caractérisation d'une émission avec une probabilité supérieur à un seuil définit (par exemple 95%).

Il faut par exemple 91 impulsions dans une émission du sous mode 0 pour obtenir toutes les combinaisons possibles (PPCM de 10 pour le sous mode 0) avec une probabilité supérieure à 95%. Ici, la probabilité de réception d'une impulsion est fixée à 50%.

Déterminer le nombre d'impulsions minimales afin d'obtenir toutes les caractéristiques possibles d'un sous mode permet de mesurer la capacité de classification d'une émission en fonction de son nombre d'impulsion. C'est une information qui peut être intéressante dans le cas de certains sous mode qui sont très similaires.

2.2.4.3 Active Learning

L'analyse exploratoire de base de données 2.1 a permis de mieux appréhender l'intégration de l'Active Learning appliqué à la classification des émissions radars.

1. La méthodologie d'évaluation

Appliquer le processus d'Active Learning nécessite un ensemble de données non labélisées. Ainsi, pour tester nos algorithmes, on sélectionne une base de données de test contenant 10 exemples par sous mode avec leur label associé. Le reste sera considéré comme l'ensemble de données non labélisées. On définit un « stopping criteria » comme étant la condition nécessitant l'interruption du processus d'Active Learning. Ce critère est fixé à la limite du score de rappel obtenu lorsqu'on atteint le « plateau » de performance lors du processus d'apprentissage.

Pour évaluer nos algorithmes, on vérifie le nombre de données requêtées avant d'atteindre le stopping criteria, ainsi que la performance obtenue en labélisant le reste des données de l'ensemble non labélisé. Pour des raisons de gain de temps d'entraînement, nous utiliserons un « ExtraTrees » avec 100 arbres pour la totalité des stratégies testées.

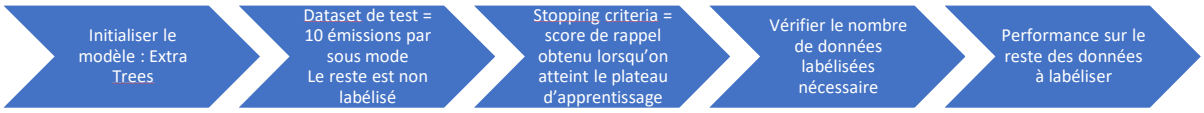


Figure 8: Processus d'évaluation de l'Active Learning

2.2.4.3.1 Les différentes stratégies

Dans le rapport sur l'état de l'art de l'Active Learning, plusieurs stratégies ont été étudiées afin de conclure quant à leur possible application au problème de classification des émissions radars à impulsions.

Ci-dessous la liste des stratégies d'Active Learning retenues et qui ont par la suite été évaluées sur les bases de données simulées :

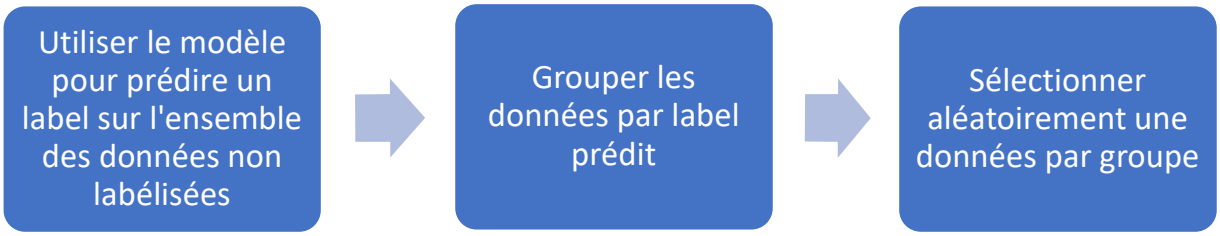
- Random sampling : Sélection aléatoire de la donnée. C'est une stratégie naïve qui nous sert de référence.
- Uncertainty Sampling : Sélection de la donnée dont le modèle est le moins certain
- Entropy Sampling : Sélection de la donnée qui maximise l'entropie
- Density Sampling : Sélection de la donnée qui maximise l'entropie et qui est dans une zone dense afin d'éviter les outliers et de ne labéliser que des données représentatives
- Batch Uncertainty Sampling : Sélection par incertitude par batch en sélectionnant des données non corrélée
- Classes Prediction Sampling : Sélection des données de sorte à homogénéiser la distribution au sein des différentes classes.
- Classes Prediction Sampling with uncertainty: Ajout d'une couche de sélection par incertitude à la stratégie précédente
- Confusion Uncertanty Sampling : Sélectionne les données maximisant l'incertitude parmi les classes les plus confuses pour le modèle selon sa matrice de confusion

Les stratégies de « Uncertainty Sampling », « Entropy Sampling », « Density Sampling » et « Batch Uncertainty Sampling » sont détaillés dans l'état de l'art, à retrouver en annexe.

Les stratégies de « Classe Prediction Sampling », « Classe Prediction Sampling with Uncertainty » et « Confusion Uncertainty Sampling » ont été élaboré spécifiquement pour le projet et ne sont donc pas détaillés dans l'état de l'art.

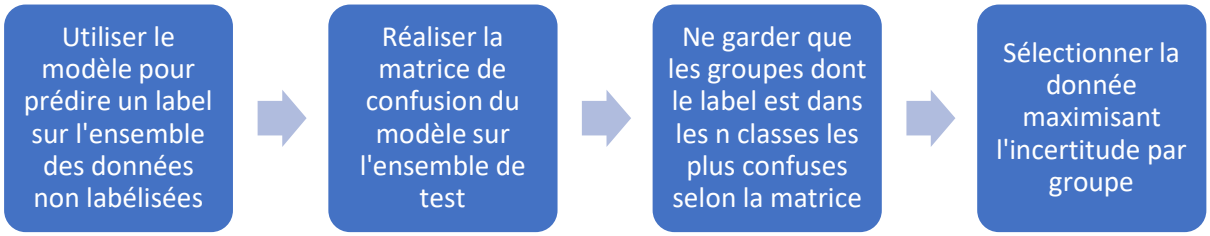
Voici les processus de ces trois types de sélection :

2.2.4.3.1.1 Classes Prediction Sampling



2.2.4.3.1.2 Classes Prediction Sampling with uncertainty

2.2.4.3.1.3 Classes Prediction Sampling with uncertainty



2.2.4.3.2 Choix de la stratégie

En appliquant une méthode de sélection aléatoire de la donnée à labéliser, on atteint 96% de rappel au bout de 20 000 labélisations pour un modèle de ExtraTrees. On va donc choisir 96% en rappel comme étant le « stopping criteria ».

Tableau 4 : Résultats des stratégies d'AL sur la base 2.1

Stratégie	Nombre de données labélisée pour atteindre le stopping criteria	Moyenne du temps d'une requête en seconde
Random sampling	9934	NA
Uncertainty Sampling	1050	0.52
Entropy Sampling	1491	0.56
Density Sampling	1615	14.37
Batch uncertainty sampling	4120	1
Classes Prediction Sampling	5049	3.31
Classes Prediction Sampling with uncertainty	1754	3.49
Confusion uncertanty sampling	3756	0.63

En observant le nombre de donnée labélisées avant d'obtenir les 96% de rappel, on constate que la stratégie la plus efficace est celle de sélection par incertitude divisant par 9 le nombre de donnée nécessaire comparé à la technique de sélection aléatoire.

Le temps moyen de requête peut aussi être un facteur impactant le choix de la stratégie. Les requêtes de la stratégie de sélection par incertitude sont les plus rapides parmi les stratégies implémentées autres que la sélection aléatoire. Celles de sélection par densité sont les plus lentes. Cependant, on va considérer que le temps de requête n'est pas un facteur déterminant au vu des temps relativement bas enregistrés. De plus il faut considérer que les temps de requêtes diminuent au fur et à mesure que le nombre de requête augmente et que la taille de l'ensemble de données non labélisées diminue.

Enfin, la notion de batch est aussi un paramètre à considérer. En sélectionnant la stratégie de sélection par incertitude et en sélectionnant uniquement des données non corrélées par batch, une étude sur le rappel obtenu permet d'affirmer que les performances diminuent lorsque la taille du batch augmente.

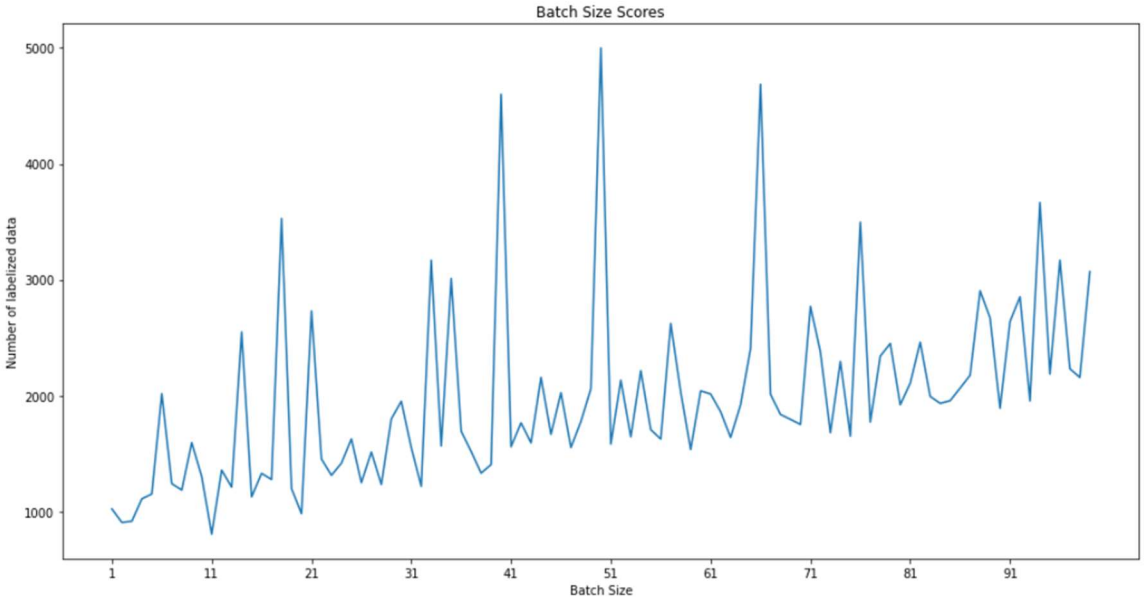


Figure 9 : Evolution des performances en fonction de la taille du batch

Une recommandation pour notre cas d'utilisation serait de laisser libre aux équipes d'annotation de sélectionner la quantité de donnée à requêter par itération.

Aussi, les stratégies personnalisées de « Classes Prediction Sampling with Uncertainty » et « Confusion Uncertanty Sampling » ont des résultats intéressants lorsqu'il s'agit de comparer avec les sélections par batch. En effet, ces stratégies sélectionnent autant de données qu'il n'y a de classes respectivement de classes confuses. Les algorithmes peuvent être réadapté pour que le batch soit spécifié.

2.2.4.4 Regroupement des émissions radars par réseaux siamois

La tâche de regroupement se positionne en amont de la tâche de classification. Elle permet en effet de constituer la base d'émission pour la classification. Il s'agit de regrouper les émissions associées aux mêmes émetteurs au sein d'une même direction d'arrivée. On suppose que chaque émission est issue d'un unique émetteur.

Le contexte dans lequel se déroule cette partie du stage est un projet en livraison pour la production où un algorithme de Machine Learning réalise la tâche de regroupement avec des performances respectant le cahier des charges. L'objectif est tout de même de tester les réseaux siamois afin de vérifier si les performances sont supérieures ou non.

Enfin, il est nécessaire de préciser que cette tâche ne fait pas partie des objectifs initiaux du stage qui ont été atteint mais permet de renforcer son contenu et fait suite à mon intérêt pour l'exercice.

1. Le système de regroupement des émissions

Le système de regroupement conservé est une classification binaire couplé à une algorithme de clustering.

La classification binaire consiste à déterminer si deux émissions proviennent du même émetteur ou non. Les émissions sélectionnées pour le regroupement sont des émissions qui appartiennent à la même direction d'arrivée. On réalise la combinaisons deux à deux de toutes les émissions respectant ces conditions, celles-ci seront passées en entrée d'un algorithme de Machine Learning réalisant la classification binaire.

Pour le clustering, un algorithme de regroupement hiérarchique ascendant a été retenu. Il permet de déterminer, à partir de la matrice de distance, les émetteurs d'origines. La matrice de distance est générée à partir à l'ensemble des données de test et des prédictions de l'algorithme de classifications binaires qui sont des probabilités entre 0 pour « ne pas regrouper » et 1 « regrouper ».

e regroupement par classification hiérarchique ascendante (HAC), aussi nommé « bottom-up approach », considère initialement que chaque donnée est un cluster puis il regroupe les données entre elles itérativement pour réaliser des clusters de plus en plus étendus jusqu'à obtenir qu'un seul cluster contenant toutes les données. On se doit donc de définir un seuil à partir duquel il est nécessaire de stopper le processus de regroupement.

Aussi, certaines émissions peuvent provenir de sources inconnues. Il sera donc requis dans un second temps de tester les algorithmes avec des émissions de sous modes inconnus. Pour cela, 40% des sous modes seront éliminés des données d'entraînement, 20% des données de validation et les données de test reste inchangées.

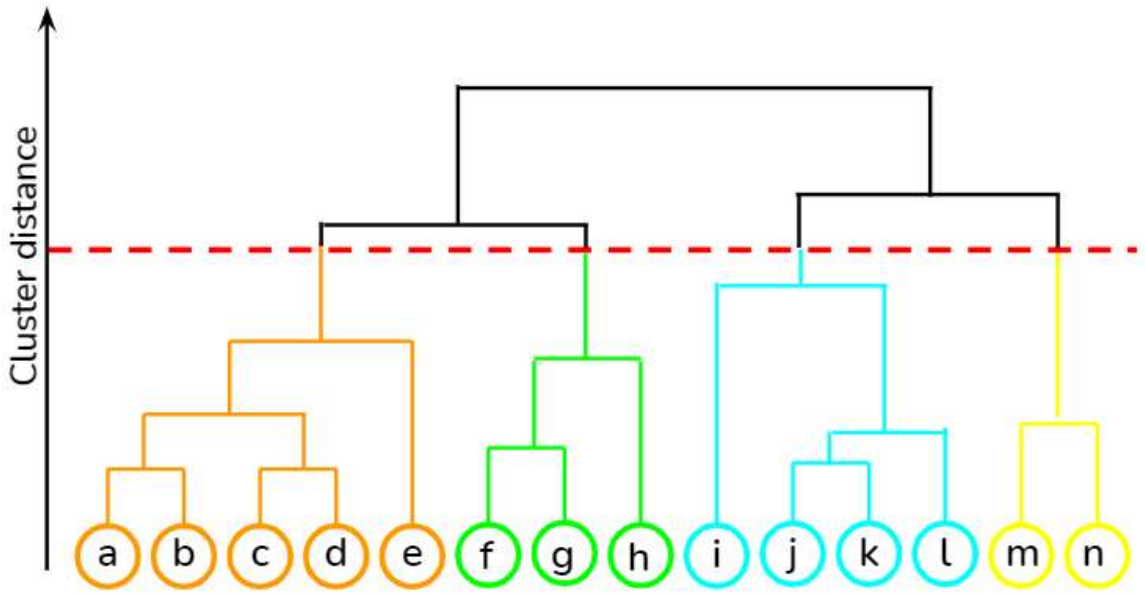


Figure 10: Représentation d'un algorithme de regroupement par hiérarchie ascendant

2.2.4.4.1 Objectifs et métriques

Dans le cadre de ce projet, on évaluera au préalable la tâche de classification binaire à partir du « F1 score » afin de d'obtenir un premier avis sur l'efficacité de nos algorithmes de classifications. Cependant seuls les scores de complétude seront pris en compte afin de sélectionner le meilleur modèle. On imposera une homogénéité supérieure à 99%.

2.2.4.4.2 Les réseaux siamois

Les réseaux siamois sont un type d'architecture de réseaux de neurones qui contient au moins deux sous-réseaux avec la même configuration. C'est-à-dire même paramètres, mêmes poids et suivant les mêmes mises à jour en miroir. Ils sont utilisés pour reconnaitre les vecteurs similaires d'où l'évidence de son utilisation pour la tâche de classification binaire.

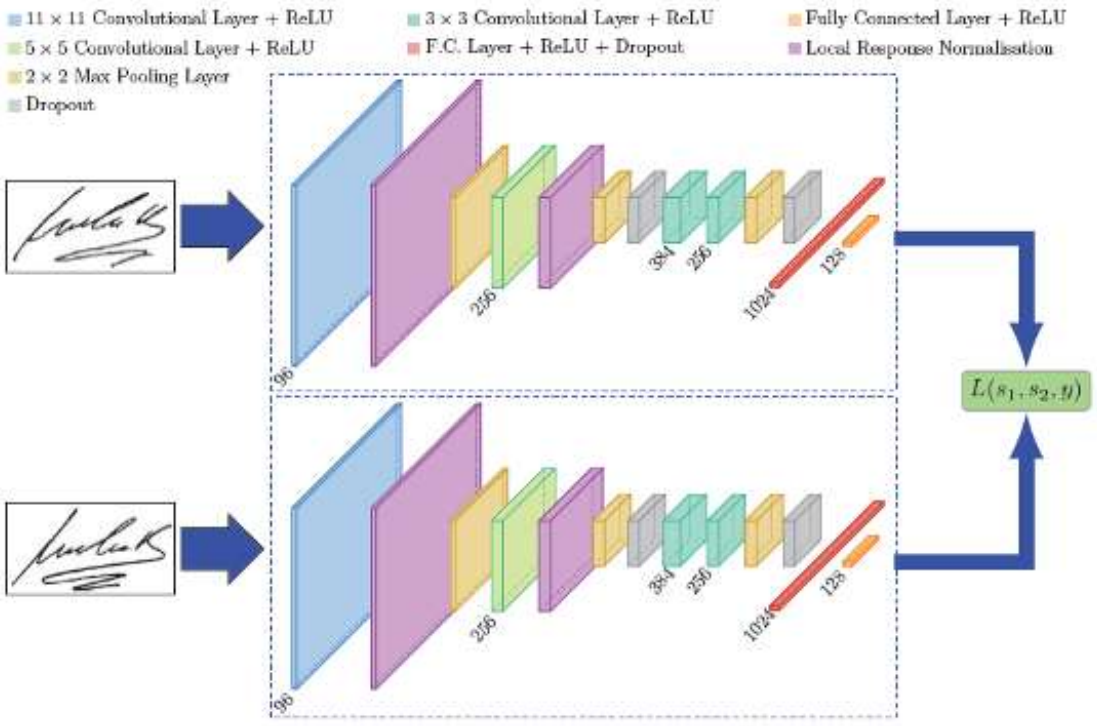


Figure 11: Illustration d'un réseau siamois appliqué à la reconnaissance de signatures

$$(Y_1, Y_2)$$

L'avantage de ce type d'architecture est divers :

- Plus de robustesse aux poids des classes
- S'assimile bien avec les méthodes ensemblistes
- Fournit une distance de similarité entre les données placées en entrées

La fonction de coût doit être adaptée aux réseaux siamois. Les fonctions de coût les plus utilisées sont :

- « Triplet Loss » qui tente de maximiser la distance avec l'entrée négative tout en minimisant celle de l'entrée positive. Pour cela, le réseau doit donc prendre 3 instances en entrée.

$$L(A,P,N) = \max(\| f(A) - f(P) \|^2 - \| f(A) - f(N) \|^2 + \alpha, 0)$$

  A est l'entrée du modèle, P et N sont des instances positives et négatives et α est une marge arbitraire pour étendre la distance.

- « Constractive Loss » qui est une généralisation de la distance euclidienne.

$$G(X_1, X_2, Y) = \frac{1-Y}{2} * D(Y_1, Y_2)^2 + \frac{Y}{2} \max(0, m - D(Y_1, Y_2)$$

  Où $D(Y_1, Y_2)$ est la distance euclidienne entre l'entrée Y1 et Y2 qui sont les sorties du réseau pour X1 et X2.

Dans notre cas nous calculerons la distance euclidienne entre les sorties des deux parties siamoises, celles-ci seront passés en paramètre d'une seconde couche constituée de couches « Dense » et qui fournira en sortie la probabilité de regroupement. Cette seconde partie du réseau aura une *loss* de « binary crossentropy ».

Afin de déterminer les paramètres des réseaux siamois, un « grid search » a été réalisé sur les deux couches siamoises en termes de nombre de couche, nombre de neurones par couche et facteur de dropout.

Ci-dessous une illustration de l'architecture optimal déterminée à l'issue de l'optimisation

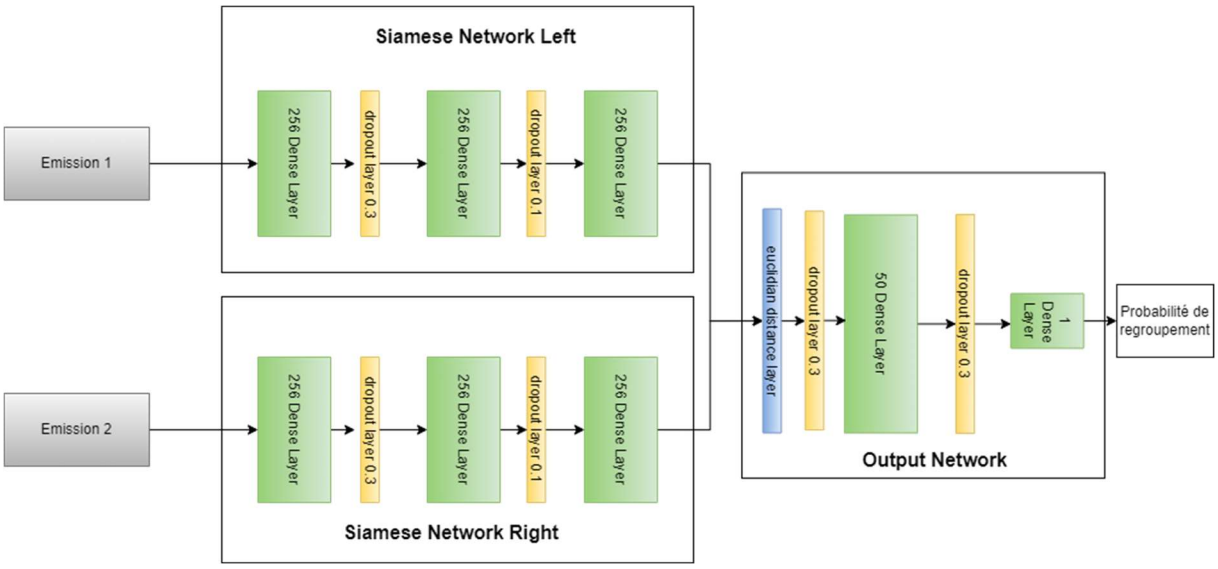


Figure 12: Architecture du réseau de neurone siamois optimisé

2.2.4.4.3 Exploration des features importantes au regroupement

Afin de déterminer l'importance des features passées en entrée du réseau siamois, il a fallu le tester sur différentes configurations. Pour cela, on extrait les statistiques issues des features sélectionner et on compare la moyenne sur 20 entraînements des F1 scores obtenus.

Ci-dessous, un tableau récapitulant les différents résultats.

Tableau 5 : Résultats réseaux siamois par features

Type de donnée en entrée	F1 score valid	F1 score test
[Niv,Di,dToa,FreqMoy]	89.97	89.42
[Niv,dToa,FreqMoy]	90.74	90.63
[Di,dToa,FreqMoy]	94.59	94.92
[Niv,dToa]	93.01	92.34
[Niv]	8.16	5.57
[Di]	87.04	87.09
[FreqMoy]	88.56	89.36
[dToa]	83.	83

On remarque que l'ensemble [Di,dToa,FreqMoy] obtient les meilleurs résultats mais que de manière générale, le regroupement donne d'assez bons scores sauf lorsqu'on passe uniquement des statistiques sur le Niveau.

On continuera cette étude avec les paramètres issues des features Di ; dToa et Frequence Moyenne.

2.2.4.4.4 Ajout de l'algorithme d'agglomérative clustering pour les scores d'homogénéité et de complétude et ajout des sous modes inconnus

L'ajout de sous modes inconnu fait chuter le F1 score de quelques pourcents pour atteindre 87,85%.

La couche agglomérative clustering permet de mesurer le score d'homogénéité de complétude puis de les comparer avec ceux de l'algorithme de Machine Learning obtenant les meilleurs scores précédemment.

Le meilleur modèle de Machine Learning a obtenu un score d'homogénéité à 99.43% et un score de complétude de 98.79% quant au modèle de réseaux siamois, il y a un gain en homogénéité qui passe à 99.93% et en complétude qui atteint 99.61%. Cette amélioration dans le score de complétude confirme l'intérêt de ce type d'architecture pour la tâche de regroupement avec un réduction de l'erreur de l'ordre de 310%.

2.2.4.4.5 Apprentissage déséquilibré

Afin d'améliorer une nouvelle fois les scores, une approche d'*Imbalanced Learning* est envisagée. En effet, le nombre de donnée regroupées passé en entrée des réseaux siamois sont plus nombreux que les données non regroupées avec environ 74% des données qui nécessitent d'être regroupées dans l'ensemble d'entraînement.

Certaines techniques d'Imbalanced Learning tel que SMOTE ne semblent pas appropriés pour les réseaux siamois et nécessites d'être adaptées. Plutôt que des méthodes d'*oversampling*, le paramètre « class_weight » de *keras* permet d'imposer au réseau de neurone un poids plus important pour la classe la moins peuplée.

Le F1 score augmente considérablement passant de 94,92% à 99.77% sans inconnues alors que la complétude passe de 99.61% à 99.73%.

…

2.2.5 Difficultés rencontrées

Dans l'ensemble, le stage s'est bien déroulé avec des objectifs atteints dans les temps imposés.

Les difficultés que j'ai pu rencontrer portent notamment sur la complexité calculatoire de nombreux algorithmes d'Active Learning. Bien que le serveur mis à disposition possède plusieurs cartes graphiques et une importante mémoire RAM, la plupart des algorithmes d'Active Learning mettaient entre une demi-journée et la journée entière avant de fournir des résultats convenables. Sur la base 3.0 contenant d'avantage de données que la 2.1, les algorithmes pouvaient mettre plusieurs semaines. Dans ces conditions, il était parfois difficile de fournir des résultats lors de chaque réunion hebdomadaire et avoir une vision complète sur les tâches à réaliser était primordiale pour avancer en parallèle sur plusieurs objectifs.

Aussi, ne pas disposer d'une base de données réelles et le contexte confidentiel du projet impose le doute sur nos résultats et nos interprétations. C'est une difficulté rencontrée par l'ensemble de l'équipe et qui n'est pas propre à mes travaux de stage.

Enfin, pour la tâche de regroupement, les performances atteintes par les algorithmes testés étaient déjà très bonnes et tenter de les améliorer pouvait sembler décourageant. En effet les algorithmes

atteignant des scores proches de 100%, chaque petite décimale de gagnée était important. C'était cependant intéressant d'avoir des algorithmes permettant de comparer les résultats.

3 Conclusion

3.1 Synthèse de la mission

La mission principale de mon stage était « exploration des processus d'Active Learning et application au problème de classification des émissions ». Les objectifs fixés proposaient en grande partie de découvrir l'Active Learning ce qui a été réalisé notamment grâce à un rapport sur l'état de l'art approuvé et archivé. De plus, l'application de certaines méthodes d'Active Learning ont été introduites sur le projet de classification d'émissions et ont donné des résultats très prometteurs pour une possible intégration avec un gain de 90% de données labélisées sur la base 2.1. Toujours dans le cadre de l'Active Learning, un code concis permet de réutiliser que d'autres projets les procédés qui ont été testés.

Enfin, l'intégration d'un réseau siamois dans le cadre de la tâche de regroupement des émissions a permis d'atteindre de meilleurs scores que les précédents algorithmes avec un gain d'environ 300% sur l'erreur en complétude. Des progrès peuvent encore être réalisés avec l'adaptation de la fonction de coût ou l'optimisation de l' *« output network ».*

3.2 Synthèse globale

Je ne regrette pas d'avoir réalisé ce stage à Atos Avantix, recherchant avant tout un stage orienté technique dans un grand groupe. Je suis satisfait sur plusieurs points :

- Les objectifs initiaux du stage ont été atteints dans le délai imparti et d'autres objectifs « bonus » ont pu être fixés afin de compléter le contenu de ce stage. J'estime que ces travaux pourront aider les équipes d'Atos à peaufiner un projet déjà bien ficelé mais surtout que ces travaux, qui ont à vocation d'être plutôt généralistes, pourront être réutilisé dans de nouveaux projets d'intelligence artificielle.
- J'ai trouvé les personnes avec qui j'ai pu travailler très compétentes et où la discussion s'est faite de façon naturelle. Je suis satisfait de mon intégration et suis persuadé de garder le contact avec nombreux de mes collègues.
- J'ai effectivement trouvé la technicité que je recherchais dans ce stage et je considère que je me suis réellement amélioré dans les domaines sur lesquels j'ai pu travailler.
- Ayant toujours réalisé mes stages en start-up, ce stage m'a permis de découvrir le travail d'un ingénieur dans un grand groupe comme Atos. Aussi, le fait que le contenu était légèrement orienté recherche était intéressant pour mon cursus, hésitant à continuer en doctorat.
- Enfin, j'ai apprécié l'outil mis à disposition par Atos nommé « Atos Learning Center » qui permet de se former à tout type de domaine avec du contenu publié de façon régulière. J'ai ainsi pu suivre des formations en intelligence artificielle et en blockchain.