

# Un Environnement Intelligent pour la Résolution des Erreurs liées aux Environnements de Programmation : une Approche Basée sur l'Apprentissage Automatique.

## Encadrants

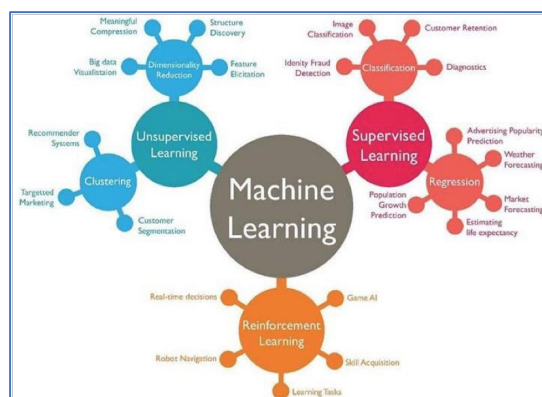
Abdelhak-Djamel Seriali ([seriali@lirmm.fr](mailto:seriali@lirmm.fr)), Bachar Rima ([bachar.rima@lirmm.fr](mailto:bachar.rima@lirmm.fr)) & Anas Shatnawi (Entreprise Berger Levrault)

Est-ce que ça vous est déjà arrivé de rester de longues minutes et mêmes des heures à ne pas trouver la solution à un problème lié à l'environnement de programmation que vous utilisez ? A titre d'exemple, pourquoi mon projet ne compile pas ? Que signifie cette erreur retournée par le compilateur en lien avec le gestionnaire de dépendances MAVEN ou GRADLE ?

Comment vous faites généralement pour résoudre ce type de problème ? Vous essayeriez d'analyser l'erreur pour trouver la solution. Malheureusement, certaines erreurs ne sont pas évidentes à résoudre à cause de toutes les librairies utilisées par les frameworks récents qui sont presque des BlackBox (Spring, Angular, Java/Android, NodeJs, Php/Symfony, etc.). Probablement, vous penseriez à des sites d'aide tels que StackOverflow ? Ou peut-être que vous demanderiez à un de votre collègue s'il aurait déjà rencontré cette erreur.

Dans le premier cas (Stackoverflow), vous essayeriez de trouver, dans la liste tous les problèmes exposés par StackOverFlow, la description la plus proche du problème que vous êtes en train d'essayer de résoudre. Ensuite, vous appliqueriez la solution en question et ensuite vous appliqueriez le principe de « Wait and See ! ». Vous répéteriez ce processus jusqu'à que l'erreur soit corrigée. Félicitations, vous avez réussi ! Mais après combien de temps et d'effort ?

Dans le deuxième cas, votre collègue (ou vos collègues) essaierait de comprendre votre problème pour voir s'il l'a déjà rencontré. Dans la meilleure des cas, il appliquerait sa solution sur votre problème. Ça marche ? Oui ! Dans ce cas, Bravo à votre collègue ! Non, ça ne marche pas encore ? Quoi faire dans ce cas ?



Notre objectif dans ce projet est de vous permettre d'arrêter de perdre du temps avec ce type d'erreur. Comment ? par « apprentissage automatique ». Nous proposons que les traces de manipulation de l'environnement par le développeur (tous les développeurs) en relation avec les erreurs, depuis le moment de leur signalement par l'environnement et jusqu'à leur résolution, soient sauvegardées/enregistrées. On aura une grande base de données de traces (BIG DATA). Ces traces seront sauvegardées selon un modèle bien étudié. Nous définirons ensuite une intelligence artificielle (modèle d'apprentissage automatique) qui permet de reconnaître la

similarité d'une nouvelle expérience (avec ses propres traces) avec certaines d'autres qui sont déjà existantes/enregistrées. Vous auriez dans ce cas, votre problème résolu. Bravo l'IA !

Nous expérimenterons la démarche décrite ci-dessus avec un modèle de réseau de neurones profond (Deep learning) appliqué sur des traces collectées dans l'environnement Android studio. La démarche sera codée en Java ou en python (choix laissé au groupe d'étudiants).

### **Pourquoi choisir ce projet ?**

- Utiliser l'intelligence artificielle/l'apprentissage automatique (réseaux de neurones) sur des problèmes de développement logiciel.
- Comprendre le fonctionnement des environnements de gestion de dépendances.
- Apprendre la génération et utilisation des traces (logs) logicielles.
- Une collaboration avec une entreprise (Grand groupe – Berger Levrault).
- Avoir une très bonne note de projet de TER si vous réalisez les trois premières tâches ci-dessus ! ☺
- Etc.