

Évaluation de modèles décisionnels légers pour la prise de décision en temps réel : application au jeu vidéo Doom

Localisation : Site du Futuroscope, France

Date de début : mai 2027 (flexible)

Durée : 2 mois maximum

Encadrants : Mickaël BARON baron@ensma.fr)

Mots clés : modèles décisionnels, intelligence artificielle, inférence à faible latence, open source

Contexte du stage

L'essor récent des grands modèles de langage (LLM) a poussé leur utilisation au-delà de la simple génération de texte, notamment pour l'orchestration d'agents autonomes et la prise de décision. Cependant, l'approche générative classique montre de fortes limites pour les environnements interactifs ou temps réel :

- **La latence** : La génération auto-régressive mot à mot (token par token) prend plusieurs centaines de millisecondes à plusieurs secondes.
- **Le surcoût computationnel et l'instabilité** : Demander du JSON à un modèle de 8B à 70B paramètres pour simplement trancher une question binaire ou choisir une action est inefficace et sujet aux hallucinations de formatage.

Récemment (septembre 2026), le concept de « modèle décisionnel » (Decision Model, illustré par des initiatives propriétaires comme Jev de TypeSafe AI) a émergé : des modèles qui abandonnent la génération de texte libre pour se concentrer sur l'évaluation immédiate d'un état et la production de décisions structurées (choix, scores, probabilités calibrées).

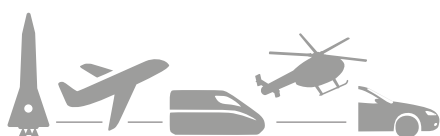
L'enjeu de ce stage est de dépasser l'écosystème propriétaire de Jev en explorant, concevant et évaluant des architectures décisionnelles 100 % Open Source, capables d'atteindre des temps d'inférence de l'ordre de quelques millisecondes.

Objectif du stage

Le stage ne s'appuiera pas sur Jev (qui est une API propriétaire fermée), mais aura pour but d'étudier les alternatives open source auto-hébergeables comme Laya, OpenJev, indecis, etc. Ainsi le candidat devra

Comparer les outils open source existants pour trouver ceux qui répondent le mieux au besoin :

- Produire des réponses simples et précises, comme choisir une action, donner une note



ou estimer une probabilité.

- Répondre rapidement en local (en moins de 15 à 20 ms) tout en utilisant peu de mémoire.
- Respecter le format demandé sans générer de réponses incorrectes ou inattendues.

Tester les modèles dans le jeu DOOM avec ViZDoom :

- Créer un bot qui joue tout seul afin de tester les modèles dans des situations rapides et changeantes.
- Le jeu fonctionne à 35 images par seconde, soit environ 28 ms par image. Cela permet de vérifier si le modèle peut choisir rapidement la bonne action : attaquer, esquiver, récupérer des soins ou changer d'arme.

Missions principales

Les missions principales sont décrites ci-dessous.

- Étudier et comparer les solutions décisionnelles open source (par exemple Laya, Open-Jev et indecis) afin d'identifier celles qui répondent le mieux aux besoins du projet.
- Évaluer leurs performances selon plusieurs critères : pertinence des décisions, latence d'inférence (objectif : moins de 15 à 20 ms), consommation mémoire et respect fiable des formats de sortie.
- Concevoir et mettre en place un bot autonome pour ViZDoom, capable de prendre des décisions dans un environnement de jeu rapide et changeant.
- Tester les modèles en situation réelle de jeu et évaluer leur capacité à choisir rapidement des actions adaptées : attaquer, esquiver, récupérer des soins ou changer d'arme.

Pour les points 2 et 3, de nombreux dépôts GitHub sont disponibles. Le candidat pourra, dans un premier temps, reproduire les expérimentations existantes, puis proposer des améliorations, voire une refonte complète.

Profil du candidat

Le candidat doit être étudiant en licence 3 dans le domaine de l'informatique.

La curiosité, l'autonomie, la rigueur et l'envie d'utiliser de nouvelles technologies.

Documents à fournir

Toute candidature incomplète ne pourra pas être étudiée.

- Curriculum vitæ ;
- les relevés de notes permettant d'apprécier le parcours et le potentiel du candidat ;
- tout autre document utile pour appuyer la candidature (par exemple : présentation de projets, lien vers un dépôt GitHub ou portfolio).

