janvier 2026

S6.C.01 Machine learning- deep learning

Analyse intelligente des avis Yelp avec ML, Deep Learning et IA agentique

1- Contexte

Les plateformes en ligne telles que Yelp**,** TripAdvisor **ou** Booking reposent largement sur les avis des utilisateurs pour orienter les choix des consommateurs. Être capable de comprendre le contenu de ces avis, d'en estimer la polarité et d'en extraire les informations clés constitue aujourd'hui un enjeu majeur pour les entreprises.

Ce projet de SAE s'inscrit dans ce contexte et s'appuie sur le **Yelp Open Dataset**, issu de la plateforme Yelp. Yelp (https://www.yelp.com) est un service de recommandation en ligne qui permet aux utilisateurs de :

- rechercher des commerces locaux (restaurants, hôtels, bars, coiffeurs, garages, etc.) ;
- consulter des avis rédigés par d'autres clients ;
- noter ces établissements avec un score de 1 à 5 étoiles ;
- publier leurs propres revues, parfois accompagnées de photos.

Yelp Open Dataset regroupe des millions de données réelles : revues textuelles, notes, informations sur les commerces et profils d'utilisateurs. Il est composé de plusieurs fichiers, dont notamment :

- review.json : avis textuels associés à une note (1 à 5 étoiles) ;
- business.json : informations sur les commerces (catégories, localisation, note moyenne, etc.) ;
- user.json : informations sur les utilisateurs (nombre d'avis, activité, réputation, etc.) ;
- photo.json : métadonnées sur les photos associées aux établissements.

Ce jeu de données constitue un support réaliste pour explorer les problématiques des questions relatives à l'analyse de données et d'intelligence artificielle.

L'objectif de la SAE est de concevoir des systèmes capables, à partir du texte d'une revue, de:

- prédire la **note attribuée** (rating),
- déterminer si l'avis est **positif ou négatif**,
- et analyser plus finement le contenu en identifiant les **aspects positifs et négatifs** exprimés par l'utilisateur.

2- Travail demandé :

Le travail qui vous est demandé comporte trois phases

janvier 2026

A- Analyse de données

Avant de se lancer dans la phase de prédiction, il est demandé, d'effectuer quelques analyses afin de mieux comprendre la répartition des données de ce Dataset, la répartion des avis en fonction de différents facteurs (catégorie du business, la popularité du reviewern …). Voici une liste non exhaustive d'analyses potentielles (vous pourrez en rajouter) :

- Répartition des avis par catégorie (Restaurants, Bars, Hotels, etc.)
- Lien entre le nombre total d'avis d'un business et la note moyenne du business (Les business très populaires sont-ils plus sévèrement jugés ?
- Les "gros reviewers" sont-ils plus sévères que les autres ?
- Est-ce que les utilisateurs expérimentés, ont tendance à faire des reviews plus détaillées ?
- Les avis négatifs sont-ils plus longs que les avis positifs ? (→ longueur moyenne des reviews par classe de note (1 → 5)
- Comparer les vocabulaires dans les avis négatifs et les avis postifis (sélectionner les 10 tops mots en utilisant par exemple tf.idf)
- Les établissements avec beaucoup de photos ont-ils de meilleurs avis ? lien entre le nombre de photos d'un business et le la note moyenne)

Ces analyses devront s'appuyer sur plusieurs fichiers du dataset (reviews, business, users, photos).

B- Modèles de prédiction demandés :

Deux tâches de prédiction sont attendues :

1. **Prédiction de la polarité des commentaires**
   Il s'agit de mettre en place un modèle prédictif capable classifier une revue en *positive*, *négative* ou *neutre*.
   Une règle simple pourra être utilisée pour créer les labels :
   - score > 3 → positif
   - score < 3 → négatif
   - score = 3 → neutre
2. **Prédiction du score (rating)**
   Construire un modèle capable de prédire la note (1 à 5 étoiles) attribuée par l'utilisateur à partir du texte de son commentaire.

La mise en place di modèle de prédictions doit respecter les conditions suivantes :
Il est demandé d'exploiter et de comparer :

1- <u>Plusieurs représentations du texte :</u>
- mots simples (sac de mots / bag-of-words),
- TF-IDF,
- Embeddings issus de modèles pré-entraînés de type **BERT** ou un **LLM de type GPT**.

Ces représentations devront être évaluées et comparées dans le cadre des tâches de classification.

2- Différentes méthodes d'apprentissage
   - des algorithmes « classiques » de Machine Learning (régression logistique, SVM, etc.) ;
   - SURTOUT des modèles de Deep Learning (MLP ou CNN) ;
   - Au moins un modèle basé sur l'architecture Transformer (prendre un modèle déjà finetuné ou à fine tuner vous-même, ce point sera traité dans un des cours) (*cette contrainte peut être liée à la contrainte ci-dessous, relative à l'IA générative*)

3- Utilisation d'une IA générative

Vous intégrerez une approche fondée sur un modèle de langage :

- **Classification en zero-shot et few-shot**
  À partir du texte d'une revue, demander à un LLM de prédire directement si l'avis est positif ou négatif, sans (ou avec très peu de) données d'entraînement.
- **Extraction d'aspects (Aspect-Based Sentiment Analysis)**
  Une même revue peut être positive sur certains aspects et négative sur d'autres (par exemple pour un hôtel : lit, bruit, propreté ; pour un restaurant : nourriture, service, prix).

  L'objectif est de fournir une revue à une IA afin qu'elle produise une sortie structurée identifiant :
  - les aspects mentionnés,
  - le sentiment associé à chacun d'eux (positif / négatif).

Pour cette partie, vous pourrez vous appuyer sur :
- LangChain, LlamaIndex ou un outil équivalent,
- des prompts structurés,
- un enchaînement de tâches (raisonnement, appel d'outils, production de sortie structurée).

Rendus :

- Code Python
- Dépôt Git

Démonstration / Evaluation du travail