09/01/2026

Système de recommandation basé sur les embeddings BERT

M2 MLSD

Hoang duy DAO
Fleur KASSI
Aida NEISANI
Imane TAGHI

Ce projet vise à développer un système de recommandation basé sur des représentations sémantiques issues de modèles de langage, et à le comparer à une méthode classique de Factorisation de Matrices.

1. Jeu de données

Nous utilisons le dataset Amazon Clothing fourni par la bibliothèque Cornac. Ce jeu de données contient des interactions utilisateur–produit sous forme de notes, ainsi que des descriptions textuelles des articles. Ces descriptions sont essentielles pour construire des représentations sémantiques des produits

2. Prétraitement

Les données ont été nettoyées et organisées sous la forme (utilisateur, item, texte). Un découpage train/test de type leave-one-out a été utilisé, où la dernière interaction de chaque utilisateur est conservée pour le test.

3. Embeddings des items

Chaque produit est représenté par un vecteur dense obtenu via un modèle BERT (sentence-transformers/all-MiniLM-L6-v2). Les descriptions textuelles des items sont concaténées puis encodées pour produire un embedding de dimension 384.

4. Embeddings des utilisateurs

Le profil utilisateur est construit en concaténant les descriptions des X=2 derniers articles consommés. Le texte résultant est ensuite encodé par BERT afin de générer un vecteur utilisateur dans le même espace que les items.

5. Modèle de recommandation

Les recommandations sont obtenues par similarité cosinus entre les embeddings utilisateur et item. Les produits déjà consommés sont exclus et les 10 items les plus similaires sont proposés.

6. Modèle de référence

Nous utilisons la Factorisation de Matrices (MF) de Cornac comme baseline. Ce modèle repose uniquement sur la matrice utilisateur–item sans information textuelle.

7. Résultats

Les performances sont évaluées avec Recall@10 et NDCG@10.

Baseline MF : Recall@10 = 0.0092, NDCG@10 = 0.0046
Modèle BERT : Recall@10 = 0.138, NDCG@10 = 0.083

Le modèle basé sur les embeddings surpasse largement la baseline, montrant l'importance de l'information textuelle pour des jeux de données sparsely observed.

8. Conclusion

L'utilisation de modèles de langage pour représenter les items et les utilisateurs permet de capturer des similarités sémantiques riches et améliore fortement la qualité des recommandations. Ce projet démontre l'intérêt des embeddings BERT pour les systèmes de recommandation modernes.

9. Architecture du dépôt Github

Le projet est organisé selon une architecture claire et modulaire afin de garantir la reproductibilité des expériences et la lisibilité du code. Le dépôt GitHub est structuré comme suit :

- data/ : contient les artefacts générés par le modèle, notamment les embeddings des items et des utilisateurs sous format NumPy (.npy). Ces fichiers permettent d'éviter de recalculer les représentations lors de nouvelles évaluations.

- results/ : regroupe les résultats expérimentaux finaux sous forme de fichiers JSON, incluant les performances du modèle basé sur les embeddings ainsi que celles de la baseline de factorisation de matrices.

- src/ : contient l'ensemble du code source du projet. On y trouve les scripts de chargement et préparation des données (dataset_textual.py), de génération des embeddings items et utilisateurs (embed_items.py, embed_users_concat.py), le moteur de recommandation (recommender.py), le modèle de référence MF (baseline_mf.py) ainsi que les scripts d'évaluation.

- README.md : décrit les étapes nécessaires pour installer les dépendances, exécuter l'entraînement des modèles et reproduire les résultats présentés dans le rapport.

Cette organisation permet de séparer clairement les données, le code et les résultats, facilitant ainsi l'extension du projet et sa compréhension par un tiers.