Université Paris Cité

Multidisciplinary project

Large Language Models with Graph Augmentation for Recommendation

PRESENTED BY

KPATOUKPA Kpodjro

NDIAYE Serigne Bassirou serigne

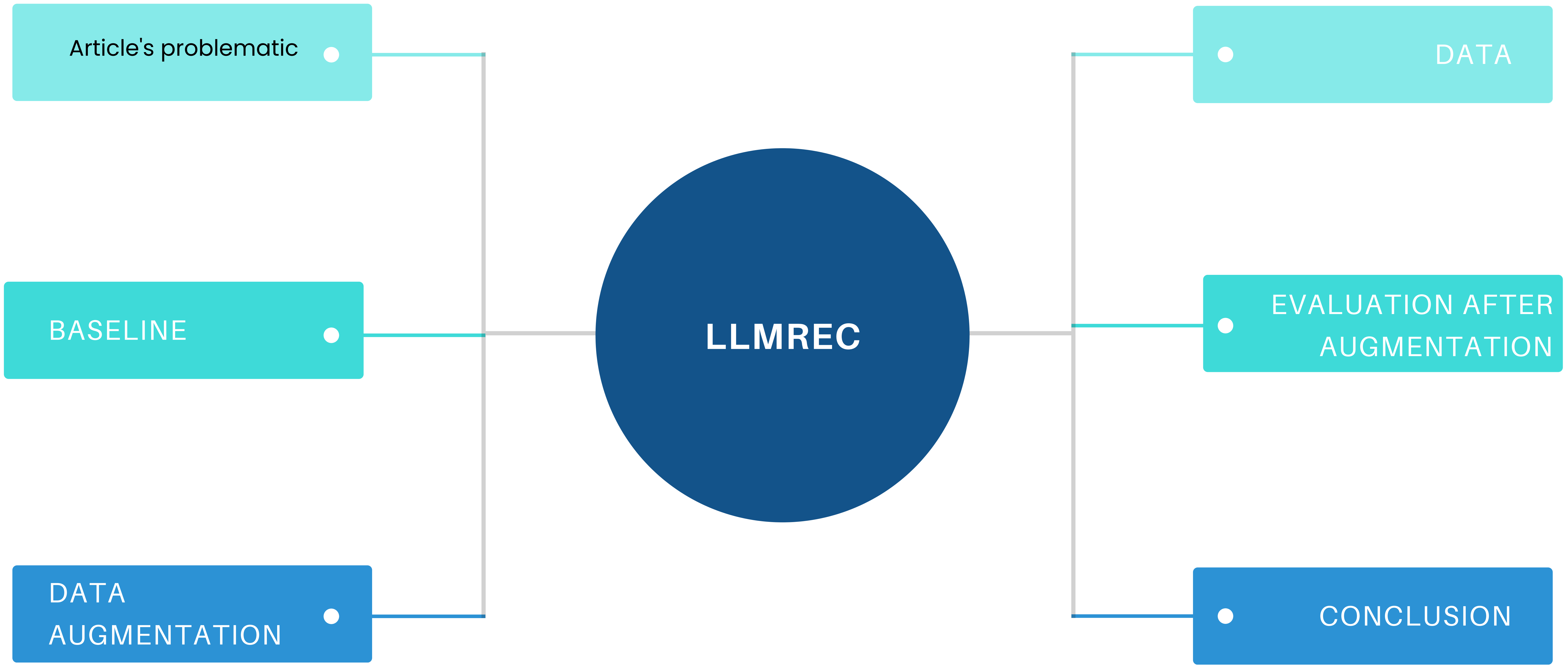
NGASSEU NDIFO Priscille Lyse

WELEHELA TAWEUTEU Orchelle Patricia

UNDER THE SUPERVISION OF

Pr. Lazhar Labiod

OVERVIEW


Article's problematic
BASELINE
DATA AUGMENTATION
LLMREC
DATA
EVALUATION AFTER AUGMENTATION
CONCLUSION

Article's problematic

The primary problem addressed in the article is the challenge of data sparsity and low-quality side information in recommendation systems. Specifically

1

ISSUES

- Data sparsity
- Low-Quality Side Information
- Noise and Bias in Data

2

INNOVATIVE APPROACH

Use LLM to augment user-item interactions, item attributes, and user profiles, with mechanisms to filter out noise, improving recommendation accuracy.

Used Data

The Netflix Prize was an open competition for the best collaborative filtering algorithm to predict user ratings for films, based on previous ratings

1

ORIGINAL DATA

Provided by Netflix, for the Netflix competition launched in october 2006

2

DESCRIPTION

Dataset of 100,480,507 ratings that 480,189 users gave to 17,770 movies.

3

SAMPLING

We selected our dataset using stratified sampling. We selected 150 movies for each user for training and 40 for testing.

Evaluation of Baseline Models for LLMRec

Objective

- Compare several classical recommendation models.
- Provide a benchmark to evaluate the contribution of the LLM model

Studied Models

- Matrix Factorization (MF)
- MLP (Multi-Layer Perceptron)
- LightGCN

Matrix Factorization

- Based on decomposing the user-item rating matrix.
- Each user and item is represented by a latent vector.
- Prediction = dot product of user and item vectors.

LightGCN

- Message-passing model on the user-item graph.
- No transformation or activation: only aggregation.
- Final representation = average of embedding layers.

MLP – Neural Collaborative Filtering

- Uses user and item embeddings.
- Feeds the embeddings into a neural network to learn a non-linear scoring function.

Confusion Matrix Analysis of the Models

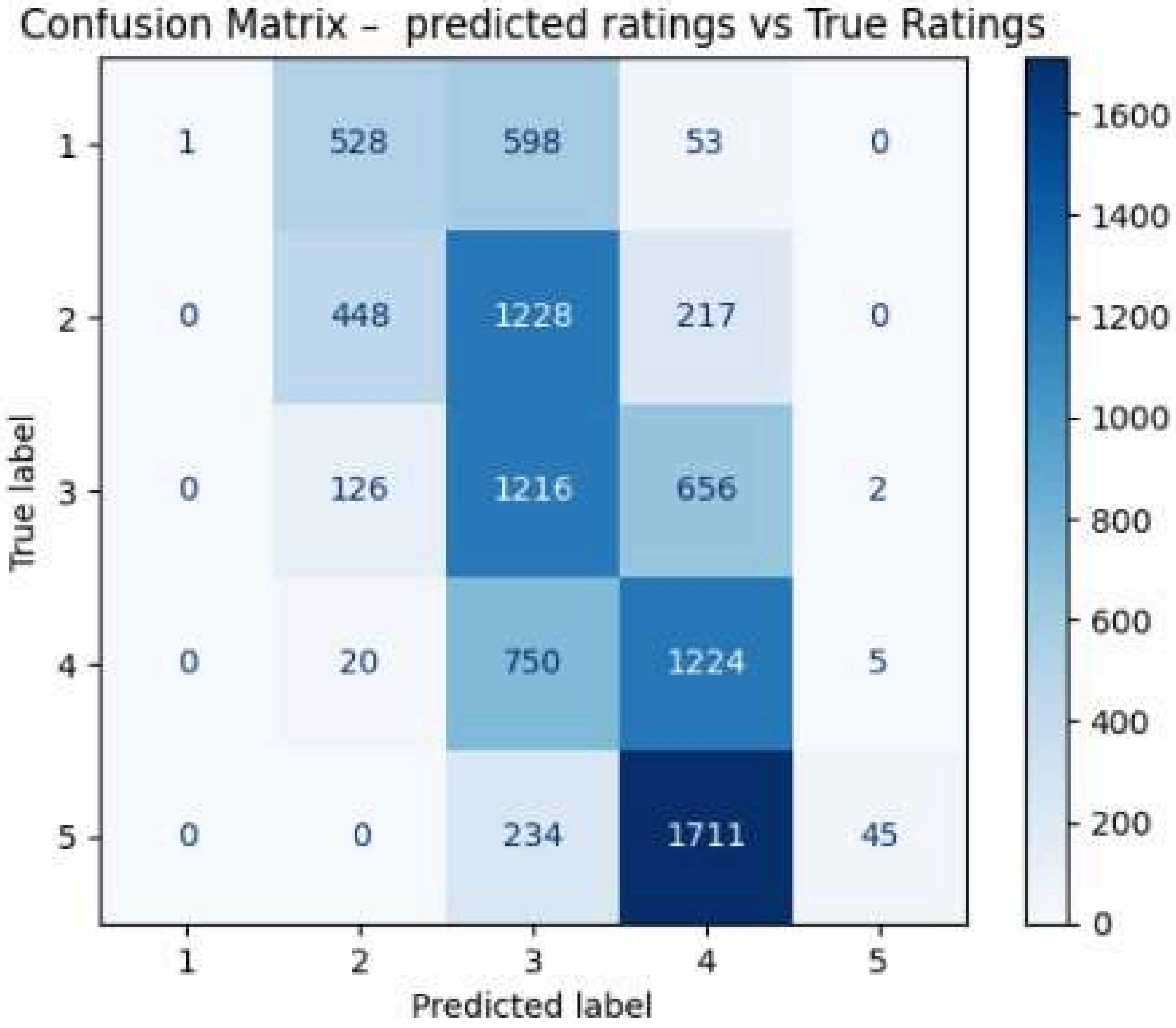


Fig1 : Matrice de confusion – LightGCN

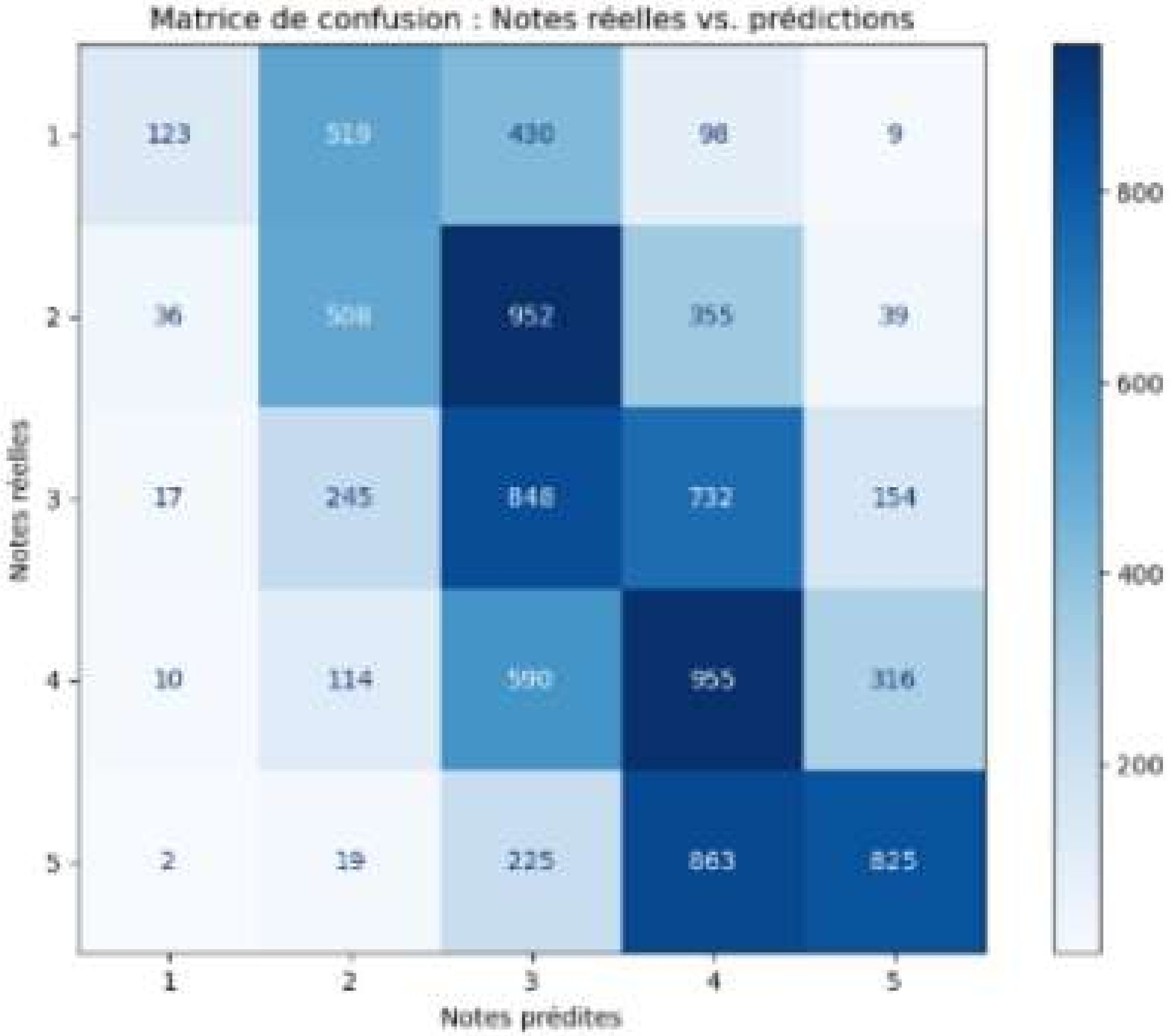


Fig 3 : Matrice de confusion – MLP

Performance Comparison – Metrics Table (on raw data without enrichment)

Modèle	Précision	Recall	F1-Score
Facto Matricielle	0.399	0.322	0.286
MLP	0.280	0.249	0.240
LightGCN	0.346	0.326	0.328

Summary & Transition

- LightGCN: most effective baseline overall.
- Supports the decision to use a more complex architecture such as LLM.

Dataset Enrichment : Movies & Users

Movie Attribute Augmentation

- Gemini & Mistral
- Input : title + release year
- Generated : director, language, country, genre …

Key Challenges

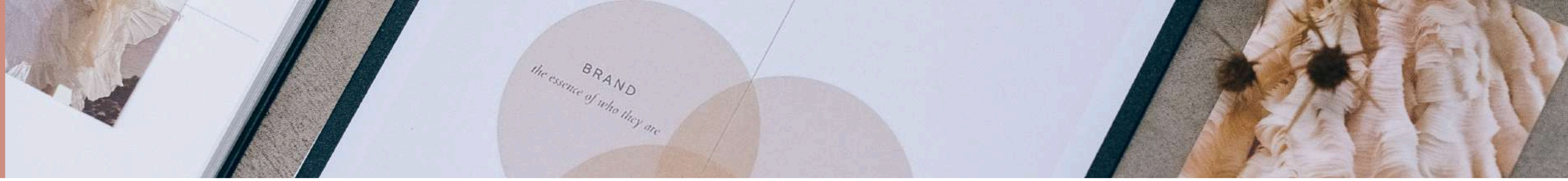
- Prompt format affected response time
- Rate/token limits
- Incomplete responses
- Inconsistent outputs

User Profiling

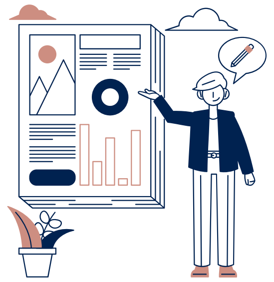
- Based on rated/enriched movies
- Inferred: gender, liked & disliked genres

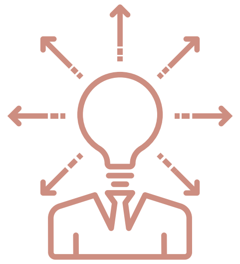
Solutions

- Used time.sleep()
- Used batch strategy
- Regenerate missing values selectively



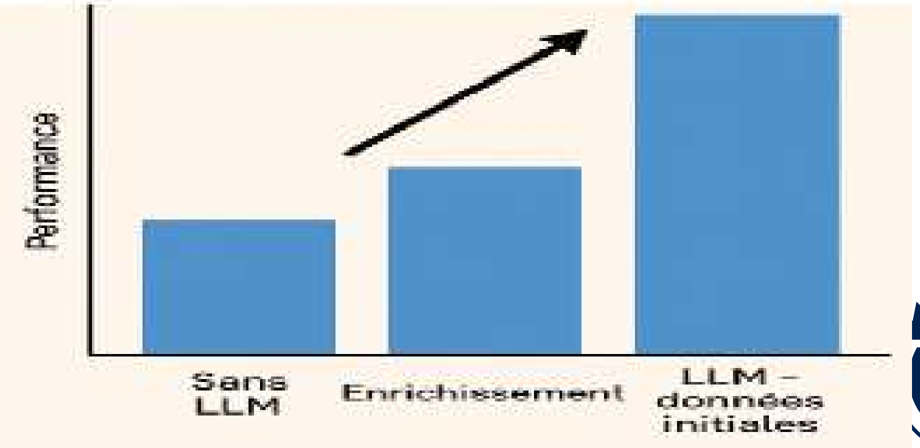
EVALUATION





Modèle (Avant Enrich.)	Accuracy	Précision (Macro)	Rappel (Macro)	F1-Score (Macro)
Factorisation Mat.	0.695	0.399	0.322	0.286
Gemini	0.686	0.688	0.686	0.686
Mistral	0.663	0.660	0.663	0.661
LightGCN	0.461	0.346	0.326	0.328
LightGCN Enrichi	0.280	0.161	0.163	0.155
MLP	0.330	0.280	0.249	0.240
MLP Enrichi	0.176	0.164	0.173	0.168

Global performance



LLM performance

The LLMs's performances do not change : we assume the use these generated attributes implicitly

Classical models

LightGCN is oriented graph : the most important and influence is the edge, not the content of the node.
MLP might suffer from the addition of columns which doesn't really explain

Deep learning models

An architecture of 4 hidden layers already has a performance of 65%.

Synthesis and recommandations

The primary problem addressed in the article is the challenge of data sparsity and low-quality side information in recommendation systems. Specifically

1

SYNTHESIS

- Intrinsec superiority of LLMs
- Make claasical Machien learning usable;
- Handle Noise and Bias in Data.

2

RECOMMANDATIONS

- Use LLM with mechanisms to filter out noise.
- Add more attributes (reviews, trailer success, user occupation, ...)

Conclusion

In summary, while the capabilities of LLMs open up new perspectives for the personalization of recommendations, their large-scale deployment must be guided by rigorous methodologies that optimize their complementarity with traditional methods while overcoming their inherent limitations. This work represents a promising step in the evolution of recommendation systems, highlighting the importance of synergy between advanced artificial intelligence and user-centered design.