Fusing collaborative and multimodal information for recommendation using LightGCN

Semantics in Intelligent Information Access

Francesco Musci
f.musci10@studenti.uniba.it

A.A. 2024/2025

Outline

Introduction

Information overload

- **Problem**: enormous amount of information on the Internet
- **Solution**: filter relevant content using recommender systems

Collaborative filtering

Users with similar past interactions will have similar preferences for future items

Issues:

- **Sparsity**: most user-item interactions are missing
- **Cold-start problem**: new users/items have no interaction history

Graph Convolutional Networks

Mechanism:

- Nodes update embeddings by aggregating neighbors
- Bipartite user-item graph structure
- Problem: over-smoothing

LightGCN

Core idea: some operations used in GCNs are detrimental to the recommendation task.

Removed components:

- Feature transformation layers
- Non-linear activation functions

Multimodal recommender systems

Motivation: Add rich item information beyond interactions

Modality examples:

- **Text**: descriptions, reviews
- **Images**: product photos, posters
- **Audio**: music tracks, soundtracks
- **Video**: trailers, clips

Benefits:

- Address cold-start and sparsity issues
- Capture item characteristics
- Improve recommendation quality
- Better generalization

Pipeline

Raw feature representation $\rightarrow$ **Feature interaction** $\rightarrow$ **Recommendation**

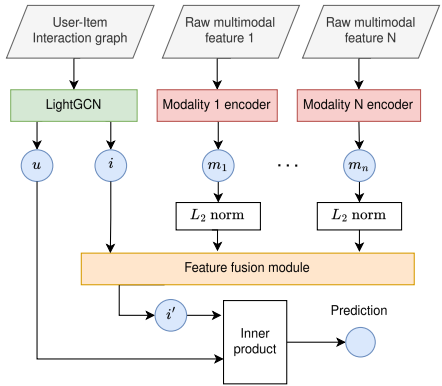
Methodology

Research Questions

- **RQ1**: does multimodal information improve the performance of LightGCN?
- **RQ2**: which architecture provides the best results?
- **RQ3**: how does our methodology perform w.r.t. the baselines?

Architecture overview

1. Extract the collaborative embeddings using LightGCN
2. Extract the multimodal embeddings with modality encoders
3. Normalize the multimodal embeddings and project them to same dimension as the collaborative embeddings
4. Fuse the embeddings using the average function
5. Generate the recommendations via inner product



Feature interaction

Preprocessing:

1. L2 normalization of multimodal embeddings
2. Linear projection to dimension d
3. Fusion with collaborative embeddings

Fusion formula

$$fuse(e_i, \|m_1^i\|_2 W_1, \ldots, \|m_n^i\|_2 W_n)$$

Late Fusion:

- Fuse during training
- Average function: equal weight to collaborative + each modality
- Embedding weight: $\frac{1}{n+1}$ where n = number of modalities

Experimentation

Raw feature representation

Text: MiniLM

- Distilled BERT model
- 99% performance, 2× faster
- Self-attention transfer

Image: Vision Transformer (ViT)

- 16×16 image patches
- Treats patches as tokens
- Transformer architecture

Video: R(2+1)D

- 3D convolutions factorized
- Separate spatial + temporal
- More non-linearity

Audio: VGGish

- CNN on spectrograms
- Audio → image representation
- Effective feature extraction

Datasets and metrics

	DbBook	MovieLens1M
Users	4528	4828
Items	3902	2464
Sparsity	99.92%	98.69%
Interactions	26533	346501
Modalities	Text, Image	Text, Image, Audio, Video

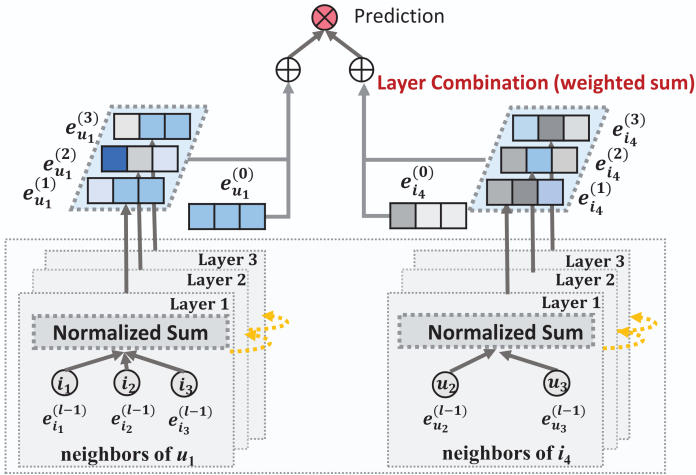
Evaluation Metrics:

- Precision@K
- Recall@K
- NDCG@K

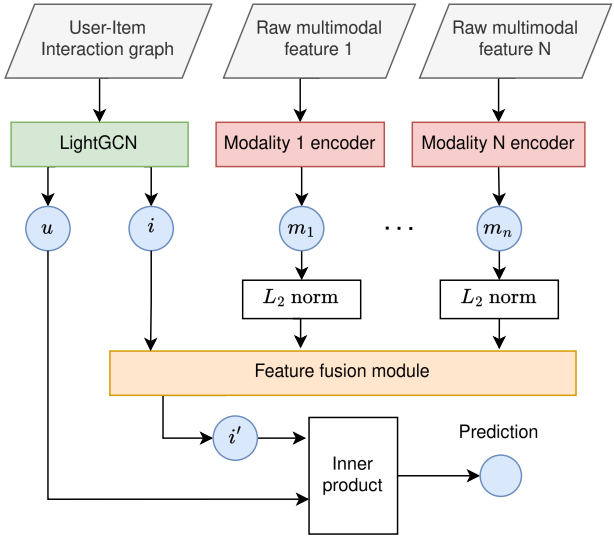
K values: 5, 10, 20, 50

- Batch size: 2048 (DbBook), 8192 (MovieLens1M)
- Learning rate: 0.001; Epochs: 500
- Hardware: NVIDIA A16 GPU

Experiments - LightGCN

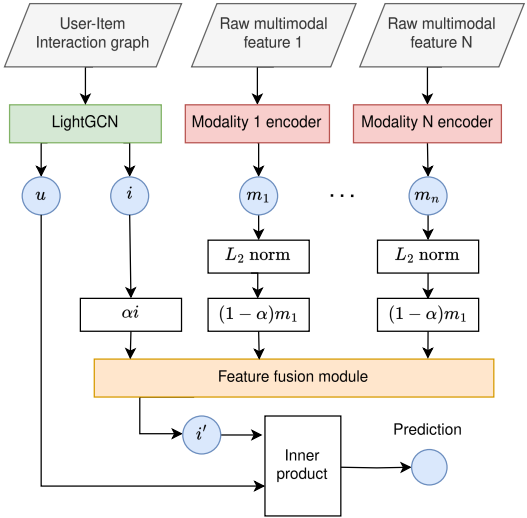


Experiments - Frozen/Unfrozen multimodal embeddings



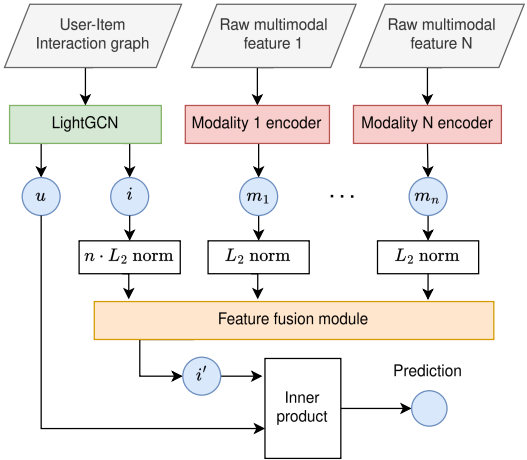
Experiments - Alpha weighting

$$fuse(\alpha e_i, (1-\alpha)m_1^i, (1-\alpha)m_2^i, \ldots)$$



Experiments - Item embedding normalization

$$fuse(n\|e_i\|_2, m_1^i, m_2^i, \ldots)$$



Results

LightGCN Baseline Results

	Top 10			Top 20		
#	Precision	Recall	NDCG	Precision	Recall	NDCG
0	0.0442	0.0699	0.0696	0.0327	0.1017	0.0820
1	0.0470	0.0740	0.0740	0.0350	0.1078	0.0873
2	0.0485	**0.0759**	0.0760	0.0359	0.1114	0.0898
3	**0.0488**	**0.0759**	**0.0763**	**0.0369**	**0.1139**	**0.0911**

Table: DbBook

	Top 10			Top 20		
#	Precision	Recall	NDCG	Precision	Recall	NDCG
0	0.1599	0.1288	0.1970	0.1337	0.2059	0.2085
1	0.1921	0.1555	0.2392	0.1569	0.2399	0.2488
2	0.2095	0.1689	0.2600	0.1704	0.2586	0.2696
3	**0.2210**	**0.1758**	**0.2728**	**0.1797**	**0.2700**	**0.2824**

Table: MovieLens1M

- Best performance at 3 layers
- MovieLens1M shows overfitting
- DbBook more stable training

RQ1: comparison with vanilla LightGCN (DbBook)

	Top 10			Top 20		
Model	Precision	Recall	NDCG	Precision	Recall	NDCG
LightGCN	0.0488	0.0759	0.0763	0.0369	0.1139	0.0911
LF-MMLGCN	**0.0618**	**0.0985**	**0.0961**	**0.0451**	**0.1412**	**0.1126**
LF-MMLGCN (α)	0.0605	0.0962	0.0931	0.0445	0.1384	0.1094
LF-MMLGCN (unfrozen)	0.0563	0.0901	0.0890	0.0419	0.1320	0.1053
LF-MMLGCN (normalized)	0.0577	0.0923	0.0902	0.0422	0.1313	0.1054

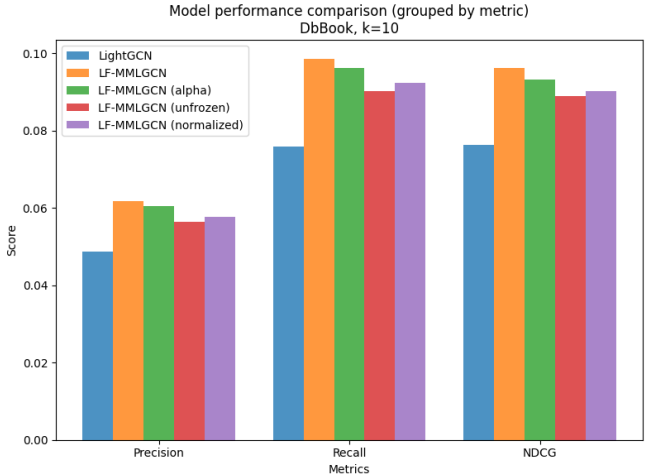
- 20-25% improvement over collaborative baseline
- Frozen embeddings work best
- Multimodal helps in sparse datasets

RQ1: comparison with vanilla LightGCN (MovieLens1M)

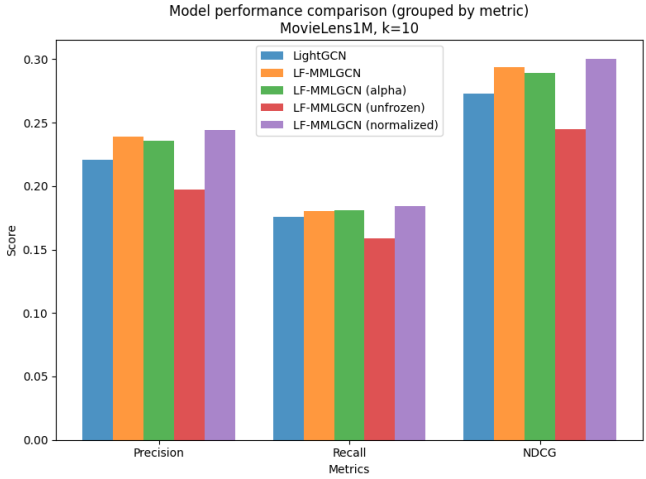
	Top 10			Top 20		
Model	Precision	Recall	NDCG	Precision	Recall	NDCG
LightGCN	0.2210	0.1758	0.2728	0.1797	0.2700	0.2824
LF-MMLGCN	0.2389	0.1802	0.2935	0.1915	0.2739	0.2976
LF-MMLGCN (alpha)	0.2359	0.1811	0.2892	0.1897	0.2750	0.2948
LF-MMLGCN (unfrozen)	0.1975	0.1585	0.2447	0.1615	0.2469	0.2548
LF-MMLGCN (normalized)	**0.2442**	**0.1843**	**0.3001**	**0.1943**	**0.2776**	**0.3027**

- 5-10% improvement - much smaller than DbBook
- Dense dataset has sufficient collaborative information

RQ2: architecture comparison (DbBook)



RQ2: architecture comparison (MovieLens1M)



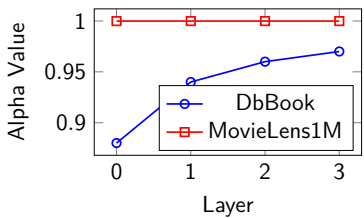
Alpha parameter analysis

DbBook:

- More weight to multimodal in early layers
- Compensates for lack of collaborative info

MovieLens1M:

- Minimal weight to multimodal
- Collaborative information sufficient



RQ3: baseline comparison (DbBook)

	Top 10			Top 20		
Model	Precision	Recall	NDCG	Precision	Recall	NDCG
LATTICE	0.0616	**0.0994**	**0.0971**	**0.0465**	**0.1450**	0.1154
MMGCN	0.0311	0.0456	0.0422	0.0273	0.0813	0.0569
VBPR	0.0159	0.0772	0.0553	0.0108	0.1059	0.0640
LF-MMLGCN	**0.0618**	0.0985	0.0961	0.0451	0.1412	**0.1162**

Table: Comparison between baseline models and LF-MMLGCN - DbBook

RQ3: baseline comparison (MovieLens1M)

	Top 10			Top 20		
Model	Precision	Recall	NDCG	Precision	Recall	NDCG
LATTICE	0.1947	0.1612	0.2395	0.1616	0.2508	0.2528
MMGCN	0.1360	0.1013	0.1615	0.1156	0.1650	0.1708
VBPR	0.1438	0.1050	0.1703	0.1220	0.1725	0.1799
LF-MMLGCN	**0.2389**	**0.1802**	**0.2935**	**0.1915**	**0.2739**	**0.2976**

Table: Comparison between baseline models and LF-MMLGCN - MovieLens1M

RQ3: baseline comparison

DbBook results:

- **Best overall**: LATTICE; LF-MMLGCN is better in a few instances
- **LF-MMLGCN**: competitive with LATTICE across the board

MovieLens1M results:

- **Best overall**: LF-MMLGCN in all instances
- **LATTICE**: worse than LF-MMLGCN by a considerable margin

In both cases, significant improvement over MMGCN and VBPR

Conclusions

Conclusions

1. **Importance of sparsity**: multimodal information most beneficial for sparse datasets (DbBook: 20-25% improvement; MovieLens1M: 5-10%)
2. **Number of layers**: multimodal helps more with a limited number of layers, as it compensates for lack of collaborative info
3. **Overfitting reduction**: multimodal features stabilize the training process
4. **Normalization**: normalizing the collaborative embeddings has an overall negative effect (reduced performance on DbBook and overfitting on MovieLens1M)
5. **Baseline**: the models are competitive against the baseline

Future work

Multimodal extensions:

- **Early vs. Late fusion**: experimentation and comparison
- **Advanced fusion mechanisms**: attention, gating
- **Embedding alignment**: collaborative-multimodal space alignment
- **Individual modality weighting**: individual learnable weighting parameter for each modality

Methodological improvements:

- **Magnitude analysis**: distinguish between information vs. magnitude effects
- **Architectural variants**: single-branch embedding, autoencoder

Thanks