Co-CoLM: Context-aware Collaborative Language Model

-協調情報とコンテキスト情報の統合による推薦精度の向上-

上智大学大学院

応用データサイエンス学位プログラム

B2479972

小島 健志

2026年1月31日

概要

本論文では、新たな情報推薦のフレームワークとして、ユーザーとアイテムの関係性を捉えた協調情報と、ユーザーの置かれた状況の文脈（コンテキスト）情報を統合する **Co-CoLM: Context-aware Collaborative Language Model** を提案する。近年の大規模言語モデル（LLM）を用いた手法では、協調情報を十分に活用できておらず、その計算コストの高さも課題となっている。一方、ユーザーの文脈を利用するコンテキストアウェア型推薦においては、数値的な処理を行うモデルの設計上、文脈の意味を本質的に理解できない。そこで本論文では、軽量な言語モデルとその自然言語理解力を活かして、協調情報とコンテキスト情報を推薦システムに統合する手法を提案する。実験の結果、推薦の難しいスパース性の高いデータセットに対して、先端手法（CoLLM）と比較して 24.6% の精度向上を実現した。本手法は、運用効率だけでなく、説明可能性の向上が見込めることから、実務への応用が期待できる。

目 次

第1章　はじめに

情報推薦は、ユーザーの嗜好を理解し、膨大な情報から適切なアイテムを提示する技術として発展してきた。特に、Matrix Factorization（MF）[1] を中心とした協調フィルタリング（Collaborative Filtering）は、明示的な特徴量を使わずに、購買・視聴ログに基づいてその背後にある共起パターンを学習することで高い推薦性能を実現した。さらに、Neural Collaborative Filtering（NCF）[2] や LightGCN [3] といった深層学習ベースの手法が登場し、ユーザーとアイテムの関係性をより豊かに捉えられるようになった。そのため視聴・購買履歴が豊富なユーザー（Warm ユーザー）において顕著な性能向上を示してきた。

一方、現実世界の推薦タスクでは、従来の推薦システムが採用する「誰が（User）」「何を（Item）選ぶか」という二次元関係だけではユーザーの意思決定を説明できない。ユーザーの嗜好は、アイテムそのものだけでなく、アイテムが検討されている状況に依存するため、「どのような状況で（Context）」選んだのかという文脈（コンテキスト）情報が重視されている [4]。このことから、ユーザー行動を多次元テンソル（User–Item–Context）として扱う研究が進展した。

代表的な手法である Tensor Factorization(TF) [5] は、コンテキストを数値的な特徴として捉え、代数的構造に基づくモデル化によって、一定の成果を上げた。さらに深層学習の登場によって、ユーザーの行動を「どんな順番で行ったか」という時系列で捉える行動系列推薦（Sequential Recommendation）が発展した。GRU4Rec [6]、SASRec [7] や BERT4Rec [8] がその代表であり、ユーザーの行動の順番を学習することで、特に「次にどのアイテムを選ぶか」という予測において、従来の MF より高い精度を実現している。

しかし、これらの手法も限界を迎えている。まず、コンテキスト軸が増えるほどデータスパース性が指数関数的に増加する「次元の呪い（Curse of Dimensionality）」の問題に直面した。従来のテンソル分解モデルは、コンテキスト情報の追加で次元が増えるため、Warm ユーザーであって

も、ある特定のコンテキストの組み合わせで観測データが不足する。当然、未観測のコンテキストの組み合わせに対しては、十分な汎化性能を発揮できない [9]。

次に、従来のテンソル分解モデルは、コンテキスト情報を単なる数値的表現として扱うため、その意味を理解していない。例えば「寒い雨の日は室内で明るいコメディが好まれる」や「家族と視聴する際は過激な表現を含む作品が好まれない」といった内容がわからない。すなわち、従来のコンテキストアウェア型の推薦はあくまで代数的相互作用（Algebraic Interaction）に基づいたモデルであり、未知の状況や背景を理解して推論する能力を持たないのである。

この状況を大きく変えたのが、大規模言語モデル（LLM）の登場である。LLM の持つ自然言語の理解力と推論能力が注目され、情報推薦への応用（LLMRec）が急速に広がっている。はじめに、ChatGPT 等の汎用モデルを用いた Zero-shot/Few-shot アプローチ [10, 11] が登場し、続いて TALLRec [12] や OpenP5 [13] に代表されるタスク特化型の指示チューニング（Instruction Tuning）が提案された。さらに、CODER [14] や LLM2BERT4Rec [15] のように LLM の表現能力を既存の推薦モデルへ統合する試みや、そして UniMP [16] のようなマルチモーダル情報を統一的に扱うフレームワークも注目を集めている。

しかしながら、これらの純粋な LLM アプローチは、情報推薦の研究で長年蓄積してきた「ユーザー・アイテム間の共起パターン」を活かしきれていない。その結果、視聴・購買履歴の少ない新規ユーザー・アイテムに対する Cold シナリオでは改善が見られるものの、履歴データが豊富な Warm シナリオに対しては、協調フィルタリング手法に精度で劣るという課題を抱えていた。

こうした課題に対し、Zhang らは CoLLM（Collaborative Large Language Model）を提案し、そうした協調情報を LLM のトークン空間へマッピングすることで、協調情報と LLM の推薦システムを統合するフレームワークを示した [17]。CoLLM は、MF や LightGCN などで事前に学習された埋め込みベクトルを LLM の入力として射影することで、LLM がプロンプトの内容と協調情報を同時に活用できるように設計されている。これにより、LLMRec の弱点であったアイテム・ユーザー間の関係性の学習が反映され、Warm/Cold 双方のシナリオで精度の向上を実現した。

しかし、CoLLM にも課題が残されている。CoLLM が扱うのはあくまでユーザー・アイテムの二次元の関係に基づくものであり、その意思決定

に寄与するコンテキスト情報をモデル内部で扱う仕組みは存在しない。確かに、Amazon-Book のようなスパースな（99.9%）データセットにおいても、その協調情報を補完することで高い性能を示した [17]。しかし、これはあくまで「誰が何を買ったのか」という前提情報に留まる。ユーザーの気分や誰が一緒かといった視聴・購買時の情報が含まれてはいない。

以上を踏まえ、本論文では CoLLM の概念を拡張し、コンテキスト情報を統合する新たなフレームワーク **Co-CoLM (Context-aware Collaborative Language Model)** を提案する。

実験には、実務の世界により近い、98%以上のスパース性を持つデータセット LDOS-CoMoDa を用いた。さらに、実社会での応用を見据え、多くの企業にとって導入の障壁となる 7B クラス以上の LLM（Vicuna-7B 等）ではなく、軽量かつ高性能な Qwen2-1.5B [18] を採用した。実験の結果、コンテキスト情報のない MF や CoLLM 等のベンチマーク手法に対して、コンテキスト情報ありの Co-CoLM が最も高いモデル性能を示した。

本論文は、LLM 時代の情報推薦において高精度かつ説明可能で、実用的な新たなコンテキストアウェア型推薦の方向性を示すものでもある。

第2章　提案手法

本章では、Co-CoLM の具体像を示す。本フレームワークは、Zhang らによって提案された CoLLM [17] を拡張し、MF に基づく協調情報と自然言語によるコンテキスト情報を、LLM の入力空間上で統合するものである。

2.1　Co-CoLM フレームワークの概要

CoLLM は、事前学習済みの協調フィルタリングモデルから得られる潜在ベクトルを、マッピング層を介して LLM のトークン埋め込み空間へ直接投影することで、LLM の持つ言語知識とアイテム・ユーザー間の協調情報の双方を活用するモデルである。しかし、その礎となる MF はユーザーとアイテムの二者関係（User-Item Interactions）のみをモデル化しており、購買・視聴時におけるユーザーの「気分（Mood）」や「同行者（Social）」といったコンテキスト情報が与える影響を捉えていない。

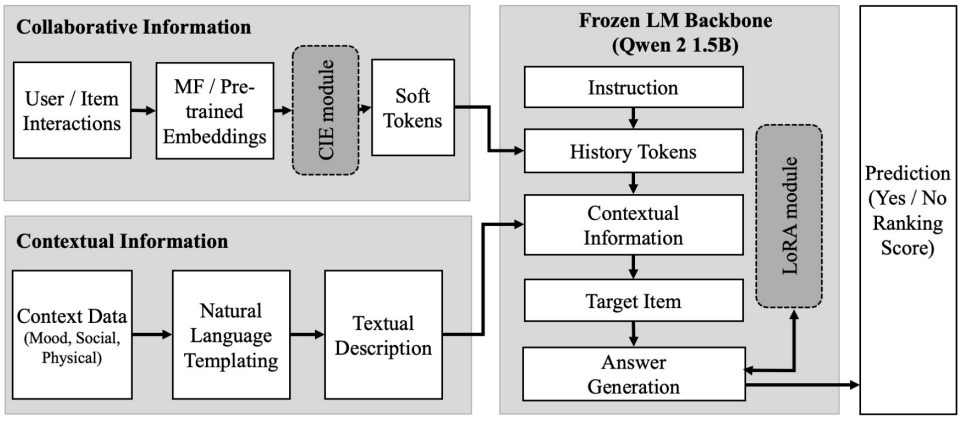

図 2.1: Co-CoLM の全体アーキテクチャ

そこで、本手法では、コンテキスト情報を自然言語としてプロンプトに

組み込むことで、CoLLM のアーキテクチャへ統合する。なお、LLM のバックボーンには、実用性と計算コストの観点から軽量な言語モデルである Qwen2-1.5B [18] を採用し、学習には LoRA (Low-Rank Adaptation) [19] を用いて効率的なファインチューニングを行う。全体像は図 2.1 に記載の通りである。

2.2 コンテキスト情報の統合

本節では、協調情報とコンテキスト情報を LLM が解釈可能な形式へ変換する具体的なエンコーディング手法について詳述する。

自然言語による挿入

本手法のアプローチは、コンテキスト情報を自然言語としてプロンプトに直接含める方法である。従来の TALLRec や CoLLM では、ユーザーの視聴履歴と推薦したいターゲットアイテムのタイトルのみが記述されていたが、本手法ではこれらに加え、自然言語で記述されたコンテキスト変数を挿入する。具体的には、テンプレートを用いて文章の中段に配置する手法をとる。

例えば、`<SocialContext>`、`<MoodContext>`というプレイスホルダーは、“Watching with family, feeling positive...” というテキスト情報に変換される。これにより、LLM が事前学習で獲得した意味的推論（Semantic Reasoning）能力を活用し、コンテキストとアイテム内容の適合性を判断させることを狙う。

協調情報の利用

一方で、ユーザーとアイテム間の情報については、CoLLM の手法を踏襲する。事前に学習した MF ベースの協調情報を用意し、そのユーザー・アイテムの潜在ベクトル（embeddings = 256）を、CIE モジュール（マッピング層）を介して LLM のトークン埋め込み空間へ射影する。

これにより、LLM はテキスト情報からコンテキストの意味を理解しつつ、同時にプロンプト内のソフトトークンからユーザー固有の嗜好パターンを直接的に参照できるようになる。

2.3 プロンプト構築

プロンプトは、以下のテンプレートに基づいて構成される。CoLLM の形式を踏襲しつつ、コンテキスト情報を追加している。

本論文で使用するプロンプトのテンプレートを Prompt 2.3.1 に示す。これは後述する Stage 2 で用いるものである。

Prompt 2.3.1: Co-CoLM のプロンプトテンプレート

```
#Question: A user has given high ratings to the following
    movies: <ItemTitleList>. Additionally, we have information
    about the user's preferences encoded in the feature <UserID
    >. Watching with <SocialContext>, feeling <MoodContext>,
    physically <PhysicalContext>. Using all available
    information, make a prediction about whether the user would
    enjoy the movie titled <TargetItemTitle> with the feature <
    TargetItemID>? Answer with "Yes" or "No". \n#Answer:
```

ここで、埋め込み表現を利用する <UserID> および <TargetItemID> の箇所は、実際の入力テキスト上では特殊トークン <unk> に置換される。モデルの内部処理において、この <unk> トークンに対応する埋め込み表現が、MF によって生成・マッピングされたソフトトークンへと差し替えられる。3 種類の <Context> については、テンプレートに従った自然言語テキスト（例: "family"）がそのまま挿入される。

2.4 学習プロセス

学習プロセスは、2 段階のチューニングアプローチを採用する。

Stage 1: テキスト情報による推薦能力の獲得

まず、LLM を基本的な映画推薦タスクへ適応させるため、テキスト形式の視聴履歴とアイテム情報、コンテキスト情報を用いて指示チューニング（Instruction Tuning）を行い、LoRA モジュールを学習する。これにより LLM は協調情報を入力する前に、テキストベースでの推論能力を獲得する。これは TALLRec と同等のアプローチである。

Stage 2: 協調情報の統合

Stage 1 でチューニングされた LLM に対し、協調情報を統合する。ここでは計算効率を重視し、CoLLM の基本戦略（Default Strategy）に準拠し、MF のパラメータおよび Stage 1 で学習した LoRA モジュールを凍結する。その上で、協調情報を LLM の空間へ射影する CIE モジュール（マッピング層）を学習する。

目的関数には、次トークン予測（Next Token Prediction）に基づく負の対数尤度を用いる。

$$\mathcal{L} = -\sum_{(x,y)\in\mathcal{D}} \log P(y|x, \mathcal{E}_{collab}, \mathcal{E}_{context}; \Theta) \tag{2.1}$$

ここで、x は入力プロンプト、y は正解ラベル（"Yes" または "No"）、Θ は学習可能パラメータ（CIE モジュール）を表す。

ここで CIE モジュールについては、単純な線形変換では異なるベクトル空間の整合が困難であるため、GELU（Gaussian Error Linear Unit）活性化関数を含む 2 層の多層パーセプトロン（MLP）として実装している。この非線形変換により、低次元のコンテキストベクトルを、Qwen2-1.5B の隠れ層次元（$d_{model} = 1,536$）を持つソフトトークンへと変換し、プロンプトに組み込んでいる。

学習設定

学習の安定化と過学習の抑制のため、学習率のスケジュールにはコサインアニーリング（Cosine Annealing）を採用した。また、検証データの AUC を用いて、20 エポック連続で改善が見られない場合に学習を早期終了する設定を導入している。

第3章　実験

本章では、提案手法である Co-CoLM の有効性を検証するために実施した実験の詳細と結果について述べる。

3.1　実験設定

3.1.1　データセット

コンテキスト情報を活用した推薦の有効性を評価するため、本論文では LDOS-CoMoDa データセット [20] を使用した。このデータセットは、映画の評価データに加え、視聴時の季節や気分などの豊富なコンテキスト変数があるのが特徴である（データは継続収集されており、本データセットは 2025 年 5 月に入手したものである）。

表 3.1 にデータセットの概要を示す。このデータの総インタラクション数は 2084 件と小規模であり、一人あたりの評価履歴は 18.6 件と少なく、スパース性（Sparsity）が約 98.4%と高い。これは、一般的なベンチマークである MovieLens-1M [21] の 100 万件、1 人平均 165 件の評価、スパース性 95.5%と比較してもデータがスパース（疎）であるため、従来の協調フィルタリングにとって推薦難度の高い環境ということを意味している。なお、前処理としてコンテキスト情報の欠損値と重複のあった 91 行を除外している。

表 3.1: LDOS-CoMoDa データセットの基本統計

Dataset	Users	Items	Interactions	Sparsity
LDOS-CoMoDa	112	1,192	2,084	98.44%

3.1.2　コンテキスト情報と特徴量選択

LDOS-CoMoDa データセットには、時間帯、曜日、季節、場所、天気、同行者、気分、体調など多様な特徴量が含まれている。しかし、これら全ての変数をモデルに入力することは、必ずしも精度の向上にはつながらない。LLM の入力トークン長（Context Window) には限りがあり、また推薦判断に直接寄与しない変数はノイズとして推論を妨げることがあるためである。

実際に、主要 8 変数を用いた場合（Full Context）と、特に重要と仮定した 3 変数（Selected Context）を用いた場合の比較実験を行った。すると、表 3.2 に示す通り、全変数を用いたモデルでは大幅な精度の低下が確認された。例えば、Co-CoLM では全変数を用いた場合（Full Context）は AUC が 0.636 に留まったが、変数を 3 つに絞ることで 0.749 まで向上した。

表 3.2: コンテキスト変数の数による精度の比較 (Full vs. Selected)

Model	Context Type	Variables	AUC
Co-CoLM	Full Context	8	0.636
Co-CoLM	**Selected**	**3**	**0.749**

したがって本論文では、ユーザーの意思決定プロセスに最も直接的な影響を与え、かつ精度向上に寄与する **Mood（気分）**、**Social（同行者）**、**Physical（体調）** の 3 変数を採用することとした。表 3.3 にその詳細を記す。この選択は、天気や季節といった外部環境要因と比較し、ユーザーの意思決定により直接的かつ重要な影響を与えるという仮説に基づいている。実際、この 3 変数は、LDOS-CoMoDa データセットを使用した推薦システムにおける SHAP 値を用いた因果分析（説明可能性に関する研究） [22] において、最も影響力の大きい要素として特定されており、その妥当性が裏付けられている。

3.1.3　データの分割と前処理

モデルの学習および評価にあたり、データセットを訓練（Training）、検証（Validation）、テスト（Test）の 3 つに分割した。分割数は表 3.4 に示す通りである。評価の信頼性を担保するためにテストデータを十分に

表 3.3: 実験に使用したコンテキスト変数（詳細定義）

変数名	値の定義 (Values)	説明
Mood	positive, neutral, negative	視聴時のユーザーの感情状態で、気晴らし等のジャンル選好に影響を与える
Social	alone, partner, friends, colleagues, parents, public, family	誰と一緒に視聴するかを示し、TPO やコンテンツの社会的適合性を左右する
Physical	healthy, ill	ユーザーの健康状態を示し、休息目的とするものか、または認知的負荷の高い作品に耐えられるか等に影響する

確保（30%）する一方で、検証データはハイパーパラメータ調整に必要な最小限の数（10%）に留めるように配分した。

3.1.4 推薦問題の定式化と目的変数の設定

本論文では、LLM を用いた推薦タスクを、与えられたコンテキスト情報に基づきユーザーがアイテムを好むか否かを判定する二値分類問題として定式化する。

正例の定義

本実験における予測対象の設定において難易度の調整を行っている。TALLRec や CoLLM といった先行研究では、5 段階評価のうち「4」以上を正例（Positive）として扱うことが一般的である。だが、閾値を 4 とした予備実験では正例率が約 65.8%に達し、推薦タスクが容易化して性能差が出ないことが確認された。そのため本論文では最高評価「5」のみを正例とした。つまり、本当に好みの映画かどうかを判定するものである。この場合、正例率は約 30.6%となり、より高度なモデルが求められる。

履歴情報の構築 (User History Construction)

一方で、プロンプト内の入力情報として提示するユーザーの視聴履歴については、ユーザーの広範な嗜好パターンをモデルに理解させる必要がある。そのため、予測対象のラベル設定とは異なり、評価「4」以上のアイテムを「ユーザーが好んだ映画」とみなし、履歴系列を構築した。また、LLM の入力トークン長および計算コストの制約を考慮し、先行研究 [12] の設定に準拠して、参照する履歴数は各ユーザーにつき最大 10 件に制限している。ただし、先行研究のように視聴時のタイムスタンプはないので、インデックス順に並べている。

3.1.5 評価シナリオ：Warm vs. Cold

本論文の主眼は、履歴情報が豊富なユーザーだけでなく、情報が乏しい状況下でもコンテキストを活用して高精度な推薦が可能か検証することにある。そこで、テストデータをユーザーおよびアイテムの出現頻度に基づいて以下の 2 つのシナリオに分類した。

- **Warm シナリオ:** 訓練データにおいて、ユーザーとアイテムの双方が 3 回以上出現しているインタラクション。協調フィルタリングが機能しやすい状態である。
- **Cold シナリオ:** 訓練データにおいて、ユーザーまたはアイテムの出現回数が 0 回または 1 回のみのインタラクション。完全に観測データがない状態だけでなく、履歴が少ない状態（Soft Cold）を含む。

表 3.4 に示すように、テストデータの半数以上が Cold シナリオに該当している。なお、これら 2 つの定義に含まれない中間層（頻度 2 回など）は Not Cold として全体の評価に含まれるため、シナリオ別分析からは除外している。

3.1.6 比較手法

本実験では、提案手法の有効性を多角的に検証するため、以下の 3 つのカテゴリから比較手法を選定した。

表 3.4: データの分割数と Warm/Cold の分布

Split	Total Samples	Warm Samples	Cold (Soft) Samples
Training	1,250	321	636
Validation	208	33	153
Test	626	91	466

*Note: Warm と Cold の合計は、Not Cold を含んでいないため、合計値と一致しない。

- **Traditional Baselines**: 従来の協調フィルタリングおよび深層学習ベースの手法。
  - Matrix Factorization (MF) [1]
  - LightGCN [3]
  - SASRec [7]
  - Tensor Factorization (TF) [5]: コンテキスト情報をテンソルとして扱う従来手法。
- **LLM Baselines**: 大規模言語モデルを用いた既存の推薦手法。
  - TALLRec [12]: コンテキスト情報を用いず、テキスト履歴のみで指示チューニングを行う手法。
  - CoLLM (MF) [17]: 協調情報（MF 埋め込み）を LLM に統合するが、コンテキスト情報は扱わないベースライン。
- **CoLLM (Tensor)**: CoLLM の構造を拡張し、コンテキスト情報を CP 分解によってベクトル化して統合したモデル。提案手法（自然言語型）との比較のために、本研究で独自に構築した拡張モデルである。詳細は後述する。

CoLLM (Tensor) の詳細設定

CoLLM (Tensor) は、コンテキスト情報を「自然言語」ではなく「数値的な潜在ベクトル」として扱い、LLM に入力するアプローチである。本モデルでは、多次元テンソル分解（CP 分解: CANDECOMP/PARAFAC）を用いることで、コンテキストベクトルに協調情報の役割も担わせている。

具体的には、ユーザー u、アイテム i、およびコンテキストを構成する3つの変数（Mood m, Social s, Physical p）の相互作用 $\hat{T}$ を、以下のCP分解モデルによって表現した。

$$c_r = m_r \odot s_r \odot p_r \tag{3.1}$$

$$\hat{T}(u, i, m, s, p) = \sum_{r=1}^{R} u_r \cdot i_r \cdot c_r \tag{3.2}$$

ここで R はテンソルのランク（潜在因子の次元数）を表す。式 (3.1) に示すように、コンテキスト因子 c_r は、各コンテキスト変数に対応する潜在ベクトル m_r, s_r, p_r の要素ごとの積（Hadamard積 $\odot$）として合成される。

この合成されたコンテキストベクトル $\mathbf{c} = [c_1, \ldots, c_R]$ を、マッピング層（CIEモジュール）$g_\phi(\cdot)$ を介してLLMの入力空間へ投影する設計とした。

ハイパーパラメータ設定: テンソルランク R の決定にあたり、$R \in \{16, 32, 64, 128\}$ の範囲でグリッドサーチを行った。表3.5に示す通り、検証スコア（Valid AUC）は $R = 128$ で最大となったが、$R = 16$ との差はわずか0.002であった。一方で、テストスコア（Test AUC）は $R = 16$ が最大値（0.6818）を記録し、高ランクにおける過学習の傾向が確認された。そのため、計算効率と汎化性能のバランスを考慮し、本比較手法では最Rank=16を採用した。

表 3.5: CoLLM (Tensor) におけるランク数 R による性能比較

Rank (R)	**Valid AUC**	**Test AUC**
16	0.6973	**0.6818**
32	0.6923	0.6740
64	0.6954	0.6656
128	**0.6994**	0.6672

3.2 実験結果と考察

3.2.1 コンテキスト情報の表現形式による推薦精度の比較

本節では、コンテキスト情報の有無、およびその表現形式（自然言語型 vs テンソル分解型）が推薦精度に与える影響を検証する。

表 3.6 に、各手法の性能を示す。評価指標として、全体の予測精度を示す AUC と、理想のランキング性能を示す nDCG [23] に加え、ユーザーごとの精度の公平性を測る uAUC [24]（Overall のみ）を掲載した。

nDCG の算出においては、テストデータ内の最高評価（Rating 5）を正例、それ以外（Rating 1-4）を負例としてランク付けを行った。本モデルは「Yes/No」の二値分類形式でありながら、出力層における「Yes」トークンのロジット（Logit）値を連続的な予測スコアとして抽出・利用している。これにより、単なるクラス分類にとどまらず、連続値に基づいた詳細なランキング評価が可能となり、スパースなデータ環境下でも安定した性能評価を実現している。

表 3.6: コンテキスト情報の有無による性能比較

	Overall			Warm Scenario		Cold Scenario	
Model	**AUC**	**uAUC**	**nDCG**	**AUC**	**nDCG**	**AUC**	**nDCG**
No Context							
Matrix Factorization	0.472	0.491	0.658	0.466	0.708	0.487	0.556
LightGCN	0.493	0.523	0.669	0.491	0.727	0.478	0.643
SASRec	0.537	0.507	0.682	0.486	0.790	0.550	0.697
TALLRec	0.724	0.626	0.715	0.768	0.802	0.699	0.737
CoLLM (MF)	0.601	0.537	0.684	0.611	0.788	0.571	0.711
Context							
Tensor Factorization	0.667	0.610	**0.745**	0.543	0.758	0.690	**0.776**
CoLLM (Tensor)	0.731	**0.658**	0.727	0.738	0.837	0.721	0.745
Co-CoLM	**0.749**	0.654	0.740	**0.769**	**0.837***	**0.745**	0.748

*Co-CoLM は 0.8372 で、CoLLM (Tensor) の 0.8368 を上回った。

コンテキスト情報の導入効果

表 3.6 の結果から、LightGCN や SASRec といった深層学習ベースの手法であっても、本実験のようなスパースな環境下では、十分に学習でき

ずランダム予測に近い性能（AUC 0.49〜0.54）に留まっている。これは、LLM ベースの手法の優位性を裏付けている。

さらに、コンテキスト情報を持たないベースラインの CoLLM (MF)（AUC 0.601, uAUC 0.537）と比較して、コンテキストを統合した提案手法 Co-CoLM は、AUC 0.749 となり、既存手法よりも 24.6%の精度向上となった。特に Warm シナリオで AUC 0.769、nDCG 0.837 と比較手法を上回った。

一方、Cold シナリオでも AUC は 0.745 と最高位であり、もともとの LLM 推薦の良さを引き継いでいることがわかる。また、全体の uAUC においては 0.654 と、ベースラインの 0.537 から 0.1 以上改善した。コンテキスト情報によって、ユーザーごとの予測精度のばらつきを抑え、安定した推薦につながったことが確認できる。

テンソル分解型モデルとの比較

全体の AUC において、Co-CoLM（0.749）が CoLLM (Tensor) の 0.731 および TF の 0.667 を上回った。テンソル分解型モデルとの比較から、LLM の持つコンテキストを意味として捉える力の優位性が示された。

一方、ランキング性能である nDCG では、TF が全体（0.745）および Cold シナリオ（0.776）で最高値を示した。これは、従来の代数モデルがランキング推薦において依然として強いことを示している。しかし、TF は AUC が比較的低く、全体的な精度に課題が残っており、前述したように、Warm シナリオでは Co-CoLM が優勢である。

3.3 議論と展望

本節では、実験結果から得られた知見を総括し、本論文の限界と将来の展望について論じる。

3.3.1 自然言語表現の優位性と意味的推論の役割

実験結果は、スパース性の高い環境において、コンテキスト情報を自然言語として LLM に与える提案手法が、従来の協調フィルタリングやテンソル分解等よりも優れた推薦精度（AUC）を実現することを示した。これは、映画推薦において協調情報によるユーザー嗜好の把握と、LLM に

よるコンテキストの意味理解が効果的に補完し合っていることを示唆している。

本実験では、自然言語統合の有効性を厳密に検証するため、比較手法としてコンテキスト情報を埋め込みベクトルにして統合する CoLLM (Tensor) を構築し、評価を行った。このアプローチは、uAUC（公平性）などで一定の安定性を示した。しかし、推薦精度の要となる AUC においては、提案手法の Co-CoLM が CoLLM (Tensor) を上回る結果となった。これは、コンテキストを単なる数値ベクトルとして入力するよりも、自然言語のまま記述するほうが、LLM がそれを正確に捉えられることを示している。

すなわち、LLM 本来の強みである「言語による推論能力」を最大限に引き出すことこそが、高精度なコンテキストアウェア推薦の実現に必要だと言える。したがって、今後の実用展開においては、協調情報とコンテキスト情報に加え、LLM の推論能力を活かしたアプローチが鍵となるだろう。

また、1.5B パラメータという比較的小規模な言語モデルを採用して実用的な精度が得られた点は、計算リソースが限られる実務環境への展開において重要な示唆を与えるものである。

3.3.2 限界

本論文にはいくつかの限界が存在する。

- **データセットの規模と多様性:** 本実験では、豊富なコンテキスト情報を持つ LDOS-CoMoDa データセットを採用したが、その総インタラクション数は約 2,000 件と小規模である。そのため、データ量が豊富な大規模環境におけるスケーラビリティは十分に検証できていない。今後は、大規模データセットに対して擬似的なコンテキストを付与するなどの方法で、検証範囲を拡大する必要がある。

- **LLM バックボーンの依存性:** 本実験では Qwen2-1.5B のみの評価を行った。LLM の推論能力はモデルサイズに強く依存するため、7B や 70B クラスの大規模モデル、あるいは異なるモデル群を用いた場合の挙動は未検証である。計算コストと精度の関係を明らかにするためにも、異なるバックボーンモデルを用いた感度分析が今後の課題である。

3.3.3　今後の展望：反実仮想による説明可能性の向上

本論文は「精度」の側面で成果を上げたが、今後はLLMの言語生成能力を活かし、「説明可能性（Explainability）」を高めていくことが求められるだろう。

近年、AIアルゴリズムに意思決定が影響を受ける中、推薦システムの信頼を高めることが重要だと指摘されている [25]。説明可能性の向上は、単なる付加機能ではなく、システムの透明性を高め、ユーザーの信頼と満足度を向上させるために欠かせない [26] 。Co-CoLMを使えば、従来の手法を上回る性能を達成するだけでなく、「なぜそのアイテムが推薦されたのか」という推薦根拠を提示できる。つまり説明可能性も同時に獲得することが可能になる。

特に有望なのが、反実仮想（Counterfactual）的なアプローチである。例えば、「もし同行者が『家族』でなければ、この映画は推薦されなかったか？」といった問いに対して、「同行者が『友達』であれば、スコアが低下してこの映画は推薦されなかった」などと回答を得るものだ。コンテキスト変数を操作しながら推薦リストを分析することで、説得力のある説明が可能となる。本論文で提案したCo-CoLMフレームワークはコンテキスト変数を明示的に扱えるため、こうした分析アプローチ [22] との親和性が高い。精度と説明可能性の両立は、次のステップである。

第4章　結論

本論文では、LLM を用いた推薦システムの新たな形として、ユーザーとアイテムの静的な協調関係だけでなく、状況に応じた動的なコンテキスト情報を効果的に統合するフレームワーク Co-CoLM を提案した。

98%以上の高いスパース性を持つ LDOS-CoMoDa データセットを用いた実験の結果、提案手法はベースラインの CoLLM(MF) と比較して約 24.6% の精度向上を達成した。特に、軽量な言語モデルを基盤としながらも、コンテキスト情報を「意味的」に処理することで、データがスパースな環境でも高精度な推薦が可能であることを実証した。

これまで AI の「精度」と「説明可能性」はトレードオフの関係にあるとされてきた [27]。しかし、経営学において「両利きの経営（Organizational Ambidexterity）」[28] が、相反する「知の深化」と「知の探索」の両立を説いたように、近年の AI 研究においてもこれら二つの価値の統合が求められている。

Co-CoLM によるコンテキスト認識型アプローチは、まさにこの「両利きの AI（Ambidextrous AI）」の実現に向けた実用的な一歩であり、本論文が今後の実務応用と研究発展の一助となることを期待する。

謝辞

本論文の執筆にあたり、入学当初より情報推薦の領域と LLM の可能性についてご指導いただき、研究の方向性を示してくださった深澤佑介准教授に深く感謝いたします。先生の熱心なご指導と温かい励ましがなければ、本稿を完成させることはできませんでした。

また、倉田正充准教授からは、アカデミアの門を叩くことの真の意味を学びました。ゼミや授業を通じて、その内容のみならず、研究者としての心構えや姿勢、研究アプローチ等を学べたことが財産です。

さらに、大原佳子教授には、ゼミの垣根を越えて相談に乗っていただきました。先生自身が上智大学のアットホームさを体現されており、その温かさに触れ、安心して学生生活を送ることができました。

加えて、問い合わせに即応し、学生の学習環境を高めてくださった北村好一氏をはじめとする職員の皆様、共に切磋琢磨したゼミメンバーの皆様、学業を応援してくださった職場の皆様に感謝申し上げます。

最後に、どんな時も励ましてくれ、課題や研究の時間をつくることに快く応じかつ尽力してくれた妻、家族に、心からの感謝を捧げます。ありがとうございました。

参考文献

[1] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[2] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web*, pages 173–182, 2017.

[3] Xiangnan He, Kuan Deng, Xiang Wang, Yanxin Li, and Yizhou Zhang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 639–648, 2020.

[4] Gediminas Adomavicius and Alexander Tuzhilin. Context-aware recommender systems. In *Recommender systems handbook*, pages 217–253. Springer, 2010.

[5] Alexandros Karatzoglou, Xavier Amatriain, Linas Baltrunas, and Nuria Oliver. Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In *Proceedings of the 4th ACM Conference on Recommender Systems*, pages 79–86, 2010.

[6] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*, 2015.

[7] Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, pages 197–206. IEEE, 2018.

[8] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 1441–1450, 2019.

[9] Steffen Rendle. Factorization machines. In *2010 IEEE International conference on data mining*, pages 995–1000. IEEE, 2010.

[10] Junling Liu, Chao Liu, Peilin Zhou, Renjie Lv, Kang Zhou, and Yan Zhang. Is chatgpt a good recommender? a preliminary study. *arXiv preprint arXiv:2304.10149*, 2023.

[11] Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. Large language models are zero-shot rankers for recommender systems. In *European Conference on Information Retrieval*, pages 364–381. Springer, 2024.

[12] Keqin Bao, Jizhi Zhang, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. Tallrec: An effective and efficient tuning framework to align large language model with recommendation. In *Proceedings of the 17th ACM conference on recommender systems*, pages 1007–1014, 2023.

[13] Shuyuan Xu, Wenyue Hua, and Yongfeng Zhang. Openp5: Benchmarking foundation models for recommendation. *arXiv preprint arXiv:2306.11134*, 2023.

[14] Yiqiao Jin, Yunsheng Bai, Yanqiao Zhu, Yizhou Sun, and Wei Wang. Code recommendation for open source software developers. In *Proceedings of the ACM Web Conference 2023*, pages 1324–1333, 2023.

[15] Jesse Harte, Wouter Zorgdrager, Panos Louridas, Asterios Katsifodimos, Dietmar Jannach, and Marios Fragkoulis. Leveraging large language models for sequential recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 1096–1102, 2023.

[16] Tianxin Wei, Bowen Jiang, Ruirui Li, Peiyan Zhang, Cheng Yang, Chang Zhou, Jingren Zhou, and Xu Chen. Towards unified multimodal personalization: Large vision-language models for generative recommendation and beyond. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

[17] Yang Zhang, Fuli Feng, Jizhi Zhang, Keqin Bao, Qifan Wang, and Xiangnan He. Collm: Integrating collaborative embeddings into large language models for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 2025.

[18] Qwen Team et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2(3), 2024.

[19] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

[20] Andrej Košir, Ante Odi'c, Matevž Kunaver, Marko Tkalčič, and Jurij F Tasič. Database for contextual personalization. *Elektrotehniški vestnik*, 78(5):270–274, 2011.

[21] F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19, 2015.

[22] Jinfeng Zhong and Elsa Negre. Shap-enhanced counterfactual explanations for recommendations. In *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, pages 1365–1372, 2022.

[23] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, 2002.

[24] Yiyu Liu, Qian Liu, Yu Tian, Changping Wang, Yanan Niu, Yang Song, and Chenliang Li. Concept-aware denoising graph neural network for micro-video recommendation. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pages 1099–1108, 2021.

[25] Meera Narvekar, Krish Bharucha, Varun Vishwanath, Neel Gabani, and Shaun Fernandes. Enhancing interpretability in diverse recommendation systems through explainable ai techniques. *Journal of Computational Analysis and Applications (JoCAAA)*, 32(1):447–456, Feb. 2024.

[26] Yongfeng Zhang, Xu Chen, et al. Explainable recommendation: A survey and new perspectives. *Foundations and Trends® in Information Retrieval*, 14(1):1–101, 2020.

[27] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.

[28] Charles A O Reilly and Michael L Tushman. The ambidextrous organization. *Harvard business review*, 82(4):74–83, 2004.