Data-efficient Fine-tuning for LLM-based Recommendation

Lin, X., Wang, W., Li, Y., Yang, S., Feng, F., Wei, Y., & Chua, T. S. (2024, July). Data-efficient Fine-tuning for LLM-based Recommendation. In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 365-374).

You Cheng Guo

National Cheng Kung University

國立成功大學
National Cheng Kung University

Outline

- Introduction
- Task Formulation
- DEALRec
- Experiment
- Related Work
- Conclusion

Outline

- Introduction
- Task Formulation
- DEALRec
- Experiment
- Related Work
- Conclusion

Introduction

- **LLMs in Recommendation Systems**
  - LLMs have shown promise in **CTR prediction**, **sequential recommendation**, and **explainable recommendation** tasks.
  - Fine-tuning is crucial because:
    - Existing LLM tasks differ significantly from recommendation tasks.
    - Recommendation data is constantly updated (e.g., TikTok: 160M videos/day, 942B interactions/day).
  - The Challenge
    - **High Costs**: Fine-tuning LLMs on large-scale recommendation data requires significant time and computational resources.
    - **Data Growth**: The continuous influx of new recommendation data necessitates frequent updates.

Introduction

- **Random Few-shot Fine-tuning**
  - Reduces costs compared to full fine-tuning.
  - Issue: Random samples may **miss crucial information** such as trending items or key user behaviors.
- **Core Problem**:
  - How to efficiently **identify representative samples** for LLM-based few-shot fine-tuning?

Introduction

- Existing Methods for **Coreset Selection**:
    - **Heuristic Methods**: Select hard or diverse samples (based on pre-defined metrics).
        - Limitation: May lead to **suboptimal selection** due to lack of empirical risk analysis.
    - **Optimization-based Methods**: Learn the optimal data subset to minimize empirical risk.
        - Limitation: **Computationally infeasible** for large-scale recommendation data.
- Existing methods rely on models trained on **full data**, which is impractical for LLMs.

Introduction

- To address these challenges, we define **two objectives**:
    a. **High Accuracy**
        - Select samples that minimize empirical risk.
        - Identify influential samples critical to model performance.
    b. **High Efficiency**
        - Reduce the cost of the data pruning process.
        - Break the dependency on full LLM training for sample selection.

Introduction

- **Objective**: Efficiently identify influential samples for LLM-based recommender fine-tuning.
- **Key Components**:
  a. **Influence Score**:
     - Estimates the impact of removing each sample on overall performance.

  b. **Effort Score**:
     - Prioritizes "hard" samples that LLMs struggle to learn.
- **Efficiency Boost**:
  - Use a **surrogate model** (smaller, traditional model) to approximate influence scores, reducing computation costs.

Introduction

- Contributions of This Work
  a. **New Task**: Data pruning for efficient LLM-based recommendation.
  b. **Proposed Method**: DEALRec to efficiently and accurately select influential samples.
  c. **Empirical Validation**: Extensive experiments on real-world datasets demonstrate:
     - DEALRec uses only **2% of data** to surpass full data fine-tuning.
     - **Time cost** reduced by up to **97%**.

Outline

- Introduction
- Task Formulation
- DEALRec
- Experiment
- Related Work
- Conclusion

Task Formulation

- **LLMs as Recommenders**:
  - Use **Large Language Models (LLMs)** directly as recommendation systems.
  - Fine-tuning is essential to adapt LLMs to recommendation tasks:
    1. Learn **item knowledge**.
    2. Understand **user behavior**.
- **Problem**: High cost of fine-tuning LLMs on large-scale, continuously updated recommendation data.

Task Formulation

- **High Resource Cost**:
  - Training LLMs on full data is computationally expensive.
- **Continuous Data Influx**:
  - Recommendation data grows rapidly (e.g., new users, new items, interactions).
  - Frequent updates are needed to maintain performance.
- **Random Few-Shot Sampling Issue**:
  - Reduces cost but may **miss critical samples** (e.g., trending items).

Task Formulation

- **Data puning**
  - **Goal**: Identify a small subset of **representative samples** for **few-shot fine-tuning**.
  - **Objective**: Ensure LLMs trained on the pruned data subset can achieve:
    - **High Accuracy**: Comparable to full-data fine-tuning.
    - **High Efficiency**: Significantly reduced computational costs.
  - **Formal Definition**:
    - Given training data $D = \{s_u \mid u \in U\}$s , select a subse $S \subset D$
      - U: User Set, D: Training Set, s_u : user u's training data.
    - Where $|S| = r|D|$ (selection ratio *r*) such that the LLM trained on *S* performs well on the test set.

Task Formulation

- $User\ set\ U = \{u_1, u_2, \ldots, u_{|U|}\}$
- $Item\ set\ I = \{i_1, i_2, \ldots, i_{|I|}\}$
- $Training\ Sample\ s = (x, y)$
- $x = [i_1, i_2, \ldots, i_{|x|}], x \subseteq I$
- $y \in I$
- $Training\ Set\ D = \{s_u = (x_u, y_u) \mid u \in U\}$
- $Objective : \min_\theta \sum_{s \in D} L(\theta, s)$

Task Formulation

1. **Heuristic Methods**:
   - Select samples based on pre-defined metrics (e.g., diversity or difficulty).
   - **Limitation**: Does not explicitly measure the influence of samples on model performance.
2. **Optimization-based Methods**:
   - Use bi-level optimization to find the optimal subset.
   - **Limitation**: Computationally infeasible for large datasets due to high costs.
3. **Why Existing Methods Fail**:
   - **High Training Cost**: Existing methods rely on models trained on **full data**, which is impractical for LLMs.

Task Formulation

1. **High Accuracy**:
   - Select samples that minimize empirical risk (good model performance).
2. **High Efficiency**:
   - Eliminate reliance on full-data training.
   - Use lightweight **surrogate models** to approximate sample importance.

Task Formulation

- **Input**: Full training data D.
- **Output**: Representative subset S⊂D for few-shot fine-tuning.
- **Constraints**:
    - Subset size ∣S∣=r∣D∣∣S∣=r∣D∣ (controlled by selection ratio rr).
    - Achieve **good performance** while reducing **training costs**.
- **Challenge**: How to efficiently and accurately identify influential samples?

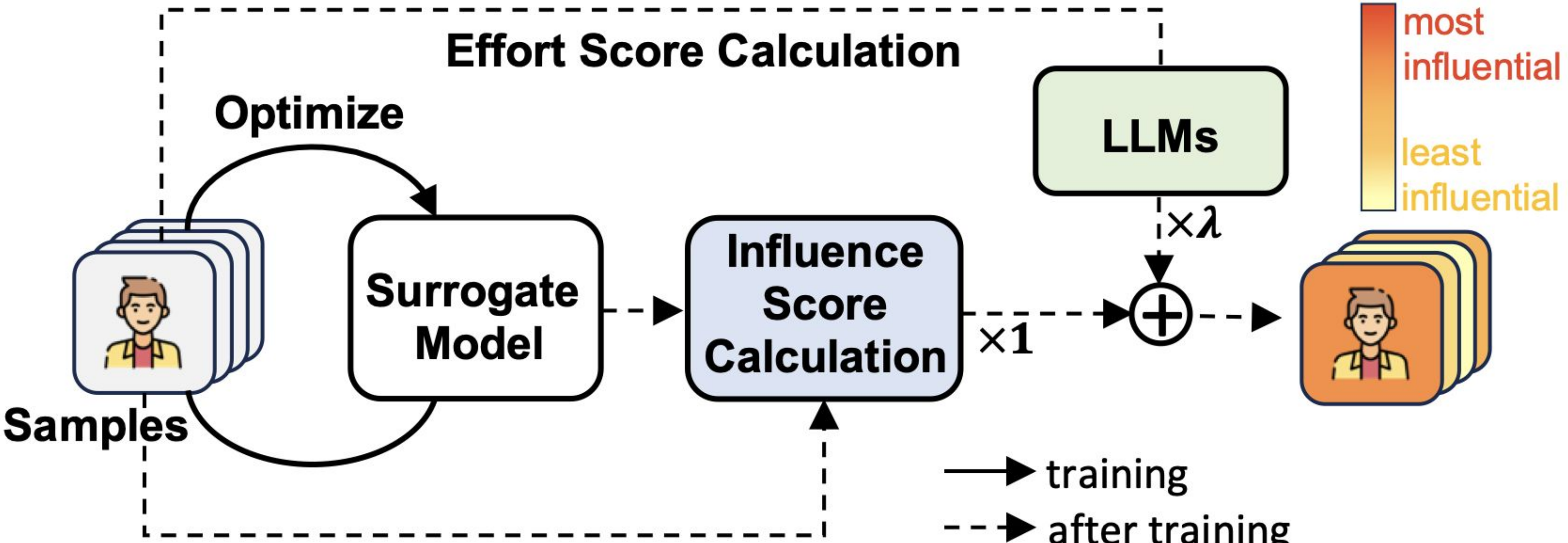
Task Formulation

- **Key Insight**:
  - Use **influence score** to assess a sample's importance.
  - Use **effort score** to prioritize hard-to-learn samples for LLMs.
- **Proposed Solution**:
  - A novel data pruning method: **DEALRec**
  - Efficiently identify influential samples for LLM-based recommendation.

Outline

- Introduction
- Task Formulation
- DEALRec
- Experiment
- Related Work
- Conclusion

DEALRec



Effort Score Calculation
Optimize
Surrogate Model
Influence Score Calculation
LLMs
×λ
×1
Samples
most influential
least influential
training
after training

DEALRec

- **Goal**: Identify **influential samples** for few-shot fine-tuning of LLMs.
- **Core Idea**: Combine two metrics to select data:
    a. **Influence Score**: Measures a sample's impact on model performance.
    b. **Effort Score**: Highlights hard-to-learn samples for LLMs.
- **Outcome**: Efficiently prune data while ensuring **high accuracy** and **low computational cost**.

DEALRec (Influence Score)

- **Purpose**: Estimate how removing a sample affects the overall model loss.
- **Final Result**:

$$I_{influence}(s) = \frac{1}{n^2}\nabla_\theta L(s,\hat{\theta})^T H_{\hat{\theta}}^{-1}[\sum_i \frac{1}{n}\nabla_\theta L(s_i,\hat{\theta})]$$

  - Efficiently approximated using **Hessian-vector products (HVP)**.
  - Symmetric property ensures computation is done **once for all samples**.

DEALRec (Effort Score)

- **Challenge**: Surrogate models are computationally efficient but have a **learning ability gap** compared to LLMs.
- **Solution**: Introduce the **Effort Score** to highlight "hard" samples for LLMs.
- **Definition**:

$$\delta s = \|\nabla \phi L^{LLM}(s)\|_2$$

  - $\nabla \phi L^{LLM}(s)$): Gradient norm of the LLM loss for sample s.
  - Larger δs: More effort required for LLMs to fit this sample → Indicates harder samples.

DEALRec (Overall Sample Score)

- Overall Sample Score combines **Influence Score** and **Effort Score**:

$$Is = I_{influence}(s) + \lambda\delta_s$$

  - λ: Regularization strength (tunable hyperparameter).
- **Intuition**:
  - Identify samples that are **both representative** of the full dataset and **challenging** for LLMs.

DEALRec (Coverage-Enhanced Sampling)

- **Problem**: Simply selecting top-ranked samples may cause redundancy and low data coverage.
- **Solution**:
  - Use **stratified sampling** to divide data into groups based on their scores.
  - Ensure balanced sampling across all groups to maximize **data diversity**.
- **Outcome**: Better **generalization** and **empirical risk reduction**.

Outline

Experiment

- RQ1: How does our proposed DEALRec **perform compared** to the coreset selection baselines for LLM-based recommendation and the models trained with full data?
- RQ2: How do the different **components of DEALRec** (i.e., influence score, gap regularization, and stratified sampling) **affect the performance**, and is DEALRec generalizable to different surrogate models?
- RQ3: How does DEALRec perform under **different selection ratios**?

Experiment

- **Datasets**:
  - **Games** (Amazon Reviews – Video Games)
  - **MicroLens-50K** (Micro-video recommendations)
  - **Book** (Amazon Reviews – Books)

Table 1: Statistics of the three datasets.

Datasets	# Users	# Items	# Interactions	Density
Games	49,156	17,332	342,329	0.04%
MicroLens-50K	49,887	19,217	359,048	0.04%
Book	88,263	86,272	5,303,707	0.07%

- **Metrics**:
  - Recall@K (K=10, 20, 50)
  - NDCG@K (Normalized Discounted Cumulative Gain)

Experiment

- **Baseline** Methods for Comparison
  1. **Random Sampling**: Select samples randomly.
  2. **GraNd**: Select samples with larger gradient norms.
  3. **EL2N**: Select samples with large prediction errors.
  4. **CCS**: Combines data coverage and sample importance.
  5. **TF-DCon**: Clusters user sequences based on representations.
  6. **RecRanker**: Selects users with more interactions to enhance diversity.
- **Backend Models**:
  - **BIGRec**: Uses LLaMA-7B with item titles.
  - **TIGER**: Learns item tokens with transformer architecture.

Experiment (RQ1)

	Methods	Games 1024-shot (r=2%) R@10	R@20	N@10	N@20	MicroLens-50K 1024-shot (r=2%) R@20	R@50	N@20	N@50	Book 1024-shot (r=1%) R@20	R@50	N@20	N@50
BIGRec	TF-DCon	0.0102	0.0157	0.0062	0.0078	0.0066	0.0099	0.0027	0.0034	0.0104	0.0144	0.0083	0.0092
	RecRanker	0.0112	0.0166	0.0074	0.0090	0.0024	0.0042	0.0011	0.0014	0.0108	0.0145	0.0090	0.0097
	CCS	0.0164	0.0246	0.0097	0.0122	0.0096	0.0131	0.0041	0.0049	0.0110	0.0145	0.0088	0.0096
	GraNd	0.0158	0.0250	0.0098	0.0125	0.0014	0.0032	0.0006	0.0010	0.0102	0.0136	0.0080	0.0087
	EL2N	0.0154	0.0256	0.0098	0.0128	0.0096	0.0045	0.0041	0.0016	0.0107	0.0149	0.0085	0.0094
	Random	0.0163	0.0241	0.0100	0.0122	0.0108	0.0151	0.0044	0.0054	0.0099	0.0134	0.0083	0.0090
	DEALRec	**0.0181***	**0.0276***	**0.0115***	**0.0142***	**0.0124***	**0.0160***	**0.0055***	**0.0064***	**0.0117***	**0.0155***	**0.0096***	**0.0104***
TIGER	TF-DCon	0.0051	0.0074	0.0033	0.0040	0.0006	0.0057	0.0002	0.0013	0.0028	0.0051	0.0020	0.0027
	RecRanker	0.0028	0.0045	0.0019	0.0024	0.0043	0.0064	0.0011	0.0014	0.0027	0.0052	0.0018	0.0025
	CCS	0.0050	0.0084	0.0031	0.0041	0.0026	0.0061	0.0010	0.0013	0.0026	0.0048	0.0018	0.0024
	GraNd	0.0042	0.0053	0.0027	0.0030	0.0006	0.0014	0.0003	0.0005	0.0008	0.0020	0.0006	0.0010
	EL2N	0.0034	0.0048	0.0024	0.0029	0.0011	0.0016	0.0004	0.0004	0.0005	0.0015	0.0004	0.0007
	Random	0.0062	0.0102	0.0039	0.0051	0.0037	0.0059	0.0011	0.0014	0.0033	0.0066	0.0022	0.0031
	DEALRec	**0.0074***	**0.0114***	**0.0062***	**0.0074***	**0.0058***	**0.0076***	**0.0020***	**0.0020***	**0.0039***	**0.0076***	**0.0026***	**0.0037***

Experiment (RQ1)

Table 3: Performance comparison between DEALRec under 1024-shot fine-tuning and the full fine-tuning of the BIGRec in terms of both accuracy and time costs. "%Improve." denotes the relative improvement achieved by DEALRec compared to the full fine-tuning. Models are trained for 50 epochs with the early stopping strategy.

	Games					MicroLens-50K					Book				
	R@10↑	R@20↑	N@10↑	N@20↑	Time↓	R@20↑	R@50↑	N@20↑	N@50↑	Time↓	R@20↑	R@50↑	N@20↑	N@50↑	Time↓
Full	0.0169	0.0233	0.0102	0.0120	36.87h	0.0081	0.0136	0.0038	0.0053	66.64h	0.0076	0.0108	0.0060	0.0068	84.77h
DEALRec	0.0181	0.0276	0.0115	0.0142	1.67h	0.0124	0.0160	0.0055	0.0064	1.23h	0.0117	0.0155	0.0096	0.0104	1.93h
% Improve.	**7.10%**	**18.45%**	**12.75%**	**18.33%**	**-95.47%**	**53.09%**	**17.65%**	**44.74%**	**20.75%**	**-98.15%**	**53.95%**	**43.52%**	**60.00%**	**52.94%**	**-97.72%**

Experiment (RQ2)

- **Tested Components**:
  a. **w/o Influence Score** → Lower performance.
  b. **w/o Effort Score** → LLM-specific samples missed.
  c. **Greedy Selection** → Redundancy reduces data diversity.

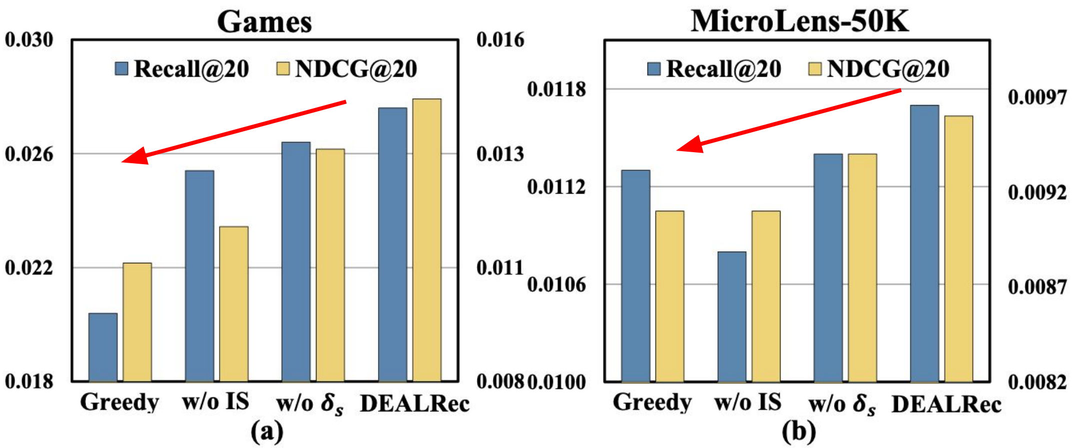


Figure 4: Ablation study of the influence score, effort score, and coverage-enhanced sample selection strategy.

Experiment (RQ2)

- **Observation**: DEALRec performs robustly across different surrogate models.

Table 4: Performance comparison between DEALRec with different surrogate models and the BIGRec under full training. "Time" presents the time costs for training the surrogate model on a single NVIDIA RTX A5000.

	R@10↑	R@20↑	N@10↑	N@20↑	Time↓
Full	0.0169	0.0233	0.0102	0.0120	/
BERT4Rec	0.0175	0.0258	0.0103	0.0128	0.76h
SASRec	0.0181	0.0276	0.0115	0.0142	0.45h
DCRec	0.0211	0.0283	0.0117	0.0137	0.61h

Experiment (RQ3)

- **Observation**: Effect of Selection Ratio r
  - Accuracy improves rapidly with rr increasing from 0.2% to 2%.
  - Beyond 2%, additional samples yield diminishing returns.

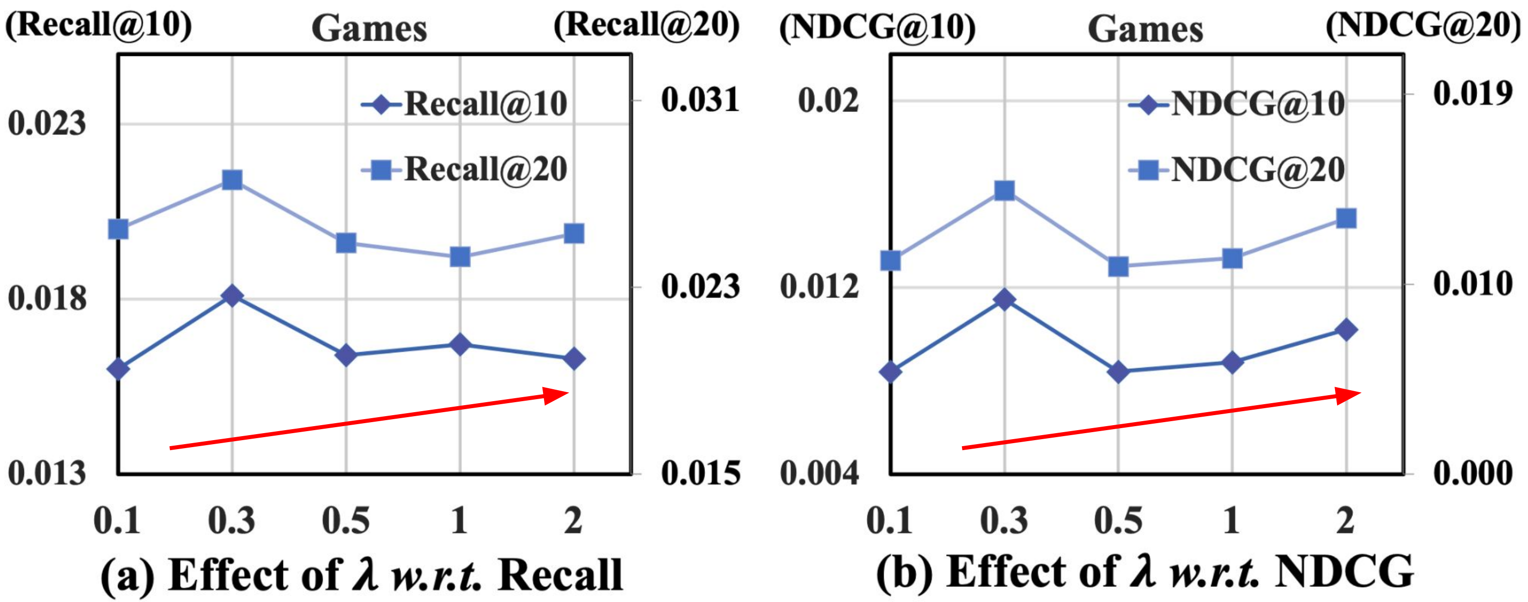


Figure 7: Performance of DEALRec with different λ.

Outline

- Introduction
- Task Formulation
- DEALRec
- Experiment
- Related Work
- Conclusion

Related Work(LLM-based Recommendation)

- **Growing Attention**:
  - Large Language Models (LLMs) have shown great potential in recommendation tasks [36, 52].
  - Successfully applied across various recommendation tasks, e.g., CTR prediction and explainable recommendations [4, 12, 27].
- **Early Approaches**:
  - Explored LLMs' **in-context learning** capabilities for recommendations [9, 42].
  - **Limitation**: Performance remains suboptimal **without fine-tuning** on domain-specific data [4].

Related Work(LLM-based Recommendation)

- **The LLM Fine-tuning Challenge**:
  - Fine-tuning LLMs on recommendation data is **computationally expensive** and time-consuming [12, 26, 31, 32, 53, 54].
  - This hinders their deployment in **real-world applications**.
- **Our Solution – Data Pruning**:
  - **Objective**: Identify **representative samples** to enable efficient **few-shot fine-tuning** of LLMs.
  - **Outcome**: Reduces resource costs while maintaining or improving recommendation performance.

Related Work(Coreset Selection)

- **Applications**: Data-efficient learning [44], Neural architecture search [40], Active learning [39].
- **Two Main Methods**:
  a. **Heuristic Methods** [7, 10, 44]:
     - Assume **difficult** or **diverse** samples are informative.
     - **Limitation**: May overlook the impact on **empirical risk**.
  b. **Optimization-based Methods** [21, 25, 50]:
     - Use bi-level or discrete optimization to minimize model error.
     - **Limitation**: Computationally infeasible for complex tasks like LLM-based recommendation.

Related Work(Coreset Selection)

- **Dependency on Full-Data Training**:
  - Existing methods often rely on training models on **full datasets** to select samples.
  - **Issue**: This approach is impractical for resource-heavy **LLM-based recommendation systems**.

Outline

- Introduction
- Task Formulation
- DEALRec
- Experiment
- Related Work
- Conclusion

Conclusion

- **Problem Addressed**:
  - Fine-tuning LLMs for recommendation is **costly** and inefficient on large-scale, continuously updated data.
- **Proposed Solution – DEALRec**:
  - Efficiently identifies **influential samples** using:
    - **Influence Score**: Measures a sample's impact on performance.
    - **Effort Score**: Highlights hard-to-learn samples for LLMs.
- **Results**:
  - Uses only **2% of data** to outperform full-data fine-tuning.
  - Reduces training time by **97%**.

Conclusion

- **Contributions**:
  - Introduced **data pruning** for LLM-based recommendation.
  - Developed DEALRec with validated **efficiency** and **accuracy** on real-world datasets.
- **Future Directions**:
  - Enhance surrogate models and extend DEALRec to other domains.

Conclusion

- 學習到的地方
  - 使用替代模型(**Surrogate Model**)快速估算數據樣本的影響分數，提升數據剪枝效率。
  - 為了解決大規模推薦數據導致的高成本問題，高效的核心數據選擇(**Coreset Selection**)方法提升 LLM 微調效率
  - 為了解決貪婪選取高分樣本(如基於影響分數或努力分數)可能導致數據樣本覆蓋範圍不足的問題，本文引入了 **Coverage-enhanced Sample Selection** 策略。