

Co-simulation of Continuous Systems: A Tutorial

<https://arxiv.org/abs/1809.08463>

0. Introduction

- 협업 필터링(Collaborative Filtering)은 추천 시스템의 핵심 기법
- 기존 방식: Matrix Factorization(MF) → 사용자와 아이템의 잠재 요인(latent factors)을 내적(dot product)으로 결합
- 내적 연산은 선형 결합만 표현 가능 → 복잡한 비선형 상호작용을 포착하기 어려움
- 신경망을 활용해 사용자-아이템 상호작용을 비선형 함수로 모델링
- 기여
 - Generalized Matrix Factorization(GMF), Multi-Layer Perceptron(MLP), Neural Matrix Factorization(NeuMF) 아키텍처 제안
 - MF와 신경망을 통합한 새로운 추천 모델
 - MovieLens, Pinterest 데이터셋으로 성능 검증

1. Overview

- 내적(dot product)을 일반화하여 사용자와 아이템의 임베딩을 비선형 신경망 구조로 결합
- 아키텍처 구성
 - GMF: 기존 MF 구조를 일반화 → 내적 대신 element-wise product 후 가중합
 - MLP: 사용자·아이템 임베딩을 concat하여 MLP에 입력 → 비선형 상호작용 학습
 - NeuMF: GMF + MLP 병렬 결합 → 선형+비선형 상호작용을 동시에 모델링
- 출력: 확률적 예측 (sigmoid activation)
- 학습: Negative Sampling + Binary Cross-Entropy Loss

- 구현: 미니배치, Adam 최적화, 임베딩 차원과 레이어 깊이에 따른 변화를 실험

2. Challenges

- 내적 한계: 기존 MF는 단순 선형 모델이라 복잡한 사용자-아이템 관계를 반영 못함
- 희소성 문제: 추천 데이터는 사용자-아이템 매트릭스가 극도로 sparse → 학습 어려움
- 모델 복잡성: 신경망 구조는 expressive하지만 학습 속도·자원 소모가 커짐
- 일반화 vs 과적합: 딥러닝 구조를 사용할 경우 오히려 과적합 위험 존재
- 학습 안정성: Negative sampling, 초기화 전략 등에 따라 성능 차이 큼

3. Method

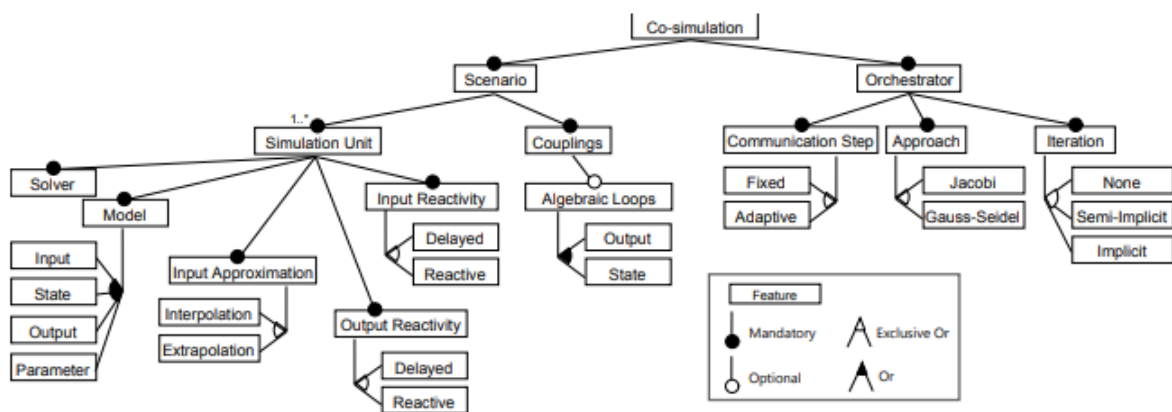


Figure 1: Co-simulation concept breakdown.

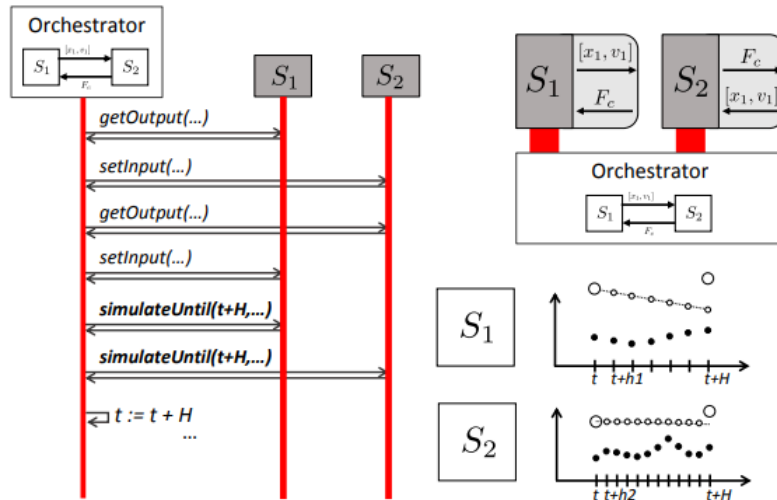


Figure 2: Example co-simulation coordination (left), co-simulation scenario (top right), and internal behavior of simulation units (bottom right).

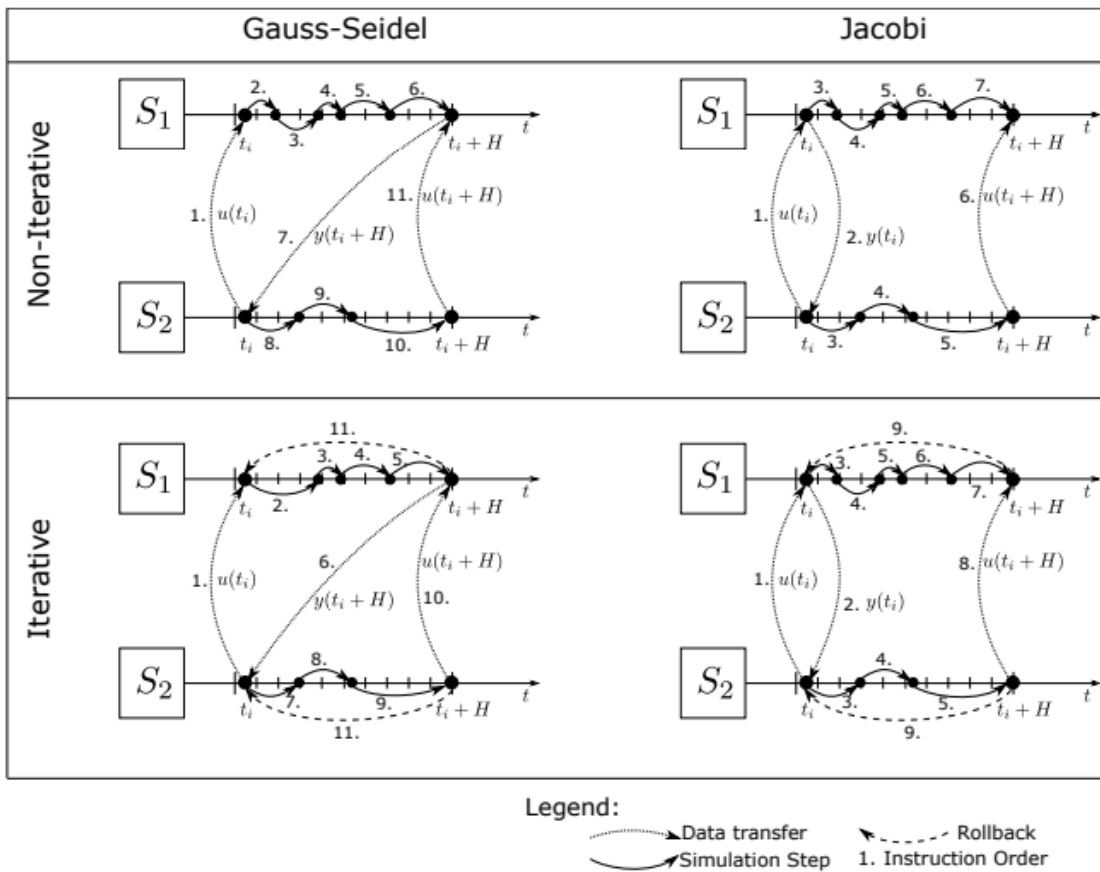
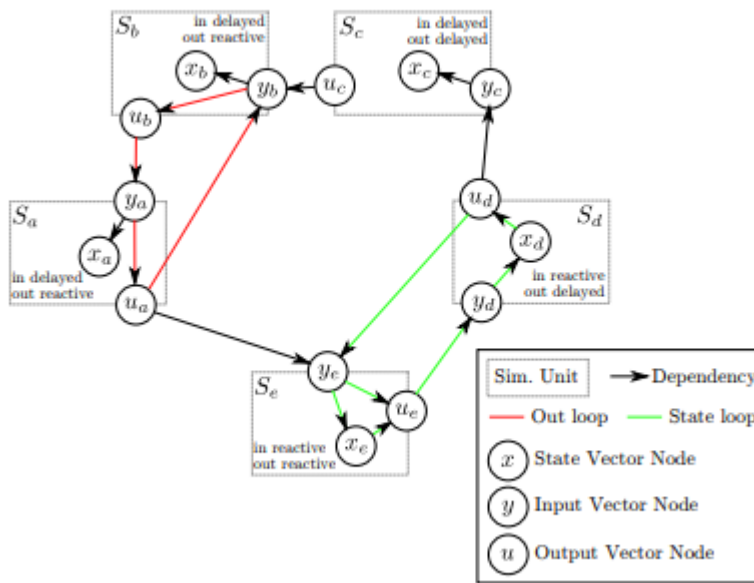


Figure 3: Overview of orchestration algorithms.



- 입력 표현
 - 사용자, 아이템 각각 embedding vector로 표현
 - dimension은 hyperparameter로 조정 가능
- 모델 구조
 - GMF:
 - element-wise product(u, v) \rightarrow weighted sum \rightarrow sigmoid
 - MF의 내적을 일반화한 구조
 - MLP:
 - concat(u, v) \rightarrow multi-layer perceptron \rightarrow sigmoid
 - 깊은 비선형 관계 학습
 - NeuMF:
 - GMF와 MLP 병렬 결합 \rightarrow 두 결과를 concat \rightarrow 최종 출력층 sigmoid
 - 선형/비선형 장점 결합
- 학습 방식
 - Loss: Binary Cross-Entropy (positive interaction=1, negative=0)
 - Negative Sampling: observed user-item pair 외에 negative sample 생성
 - Optimization: Adam 사용

- 추가 기법
 - Pretraining: GMF, MLP를 따로 학습 후 NeuMF 초기화에 활용 → 성능 개선
 - Dropout, regularization 적용 가능

4. Experiments

- 데이터셋
 - MovieLens 1M: 6천 명 사용자, 4천 아이템, 100만 rating
 - Pinterest: 1만8천 사용자, 9만 아이템, implicit feedback 데이터
- 실험 세팅
 - 평가 지표: Hit Ratio (HR@K), Normalized Discounted Cumulative Gain (NDCG@K)
 - 비교 대상: ItemPop, eALS, BPR, MF-BPR 등 전통적 방법
 - Negative sampling: 각 positive interaction당 4개 negative 생성
 - Top-K recommendation 실험
- 하이퍼파라미터
 - Embedding dimension $d \in \{8, 16, 32, 64\}$
 - MLP hidden layers: [64,32,16,8] 구조 등
 - Batch size 256, Adam optimizer 사용

5. Results

- 전반 성능
 - NeuMF가 MovieLens, Pinterest 모두에서 SOTA 모델(MF, BPR, eALS)보다 우수
 - HR@10, NDCG@10 지표에서 큰 향상
- 세부 결과
 - GMF와 MLP는 각각 장점 존재 → NeuMF가 가장 좋은 성능
 - Pretraining으로 GMF+MLP 학습 후 NeuMF 초기화 → 추가 성능 개선

- 임베딩 차원 키울수록 성능 상승, 하지만 과적합·비용 증가
- Ablation
 - Negative sampling 수, MLP 레이어 수, embedding dimension 변화 → 성능 곡선 분석
 - shallow model은 단순 관계에선 충분하지만 복잡 패턴 학습은 한계
 - deep MLP는 더 많은 사용자-아이템 상호작용 패턴 포착 가능

6. Insight

- 내적(dot product)을 신경망으로 일반화하여 추천 정확도를 높임
- 딥러닝 기반 추천 모델은 전통적 MF보다 expressive power가 큼
- Negative sampling, pretraining, architecture 설계가 성능에 크게 기여
- 다양한 상호작용 패턴 포착 가능
- 선형+비선형 결합 구조로 일반화 성능 우수
- 계산 비용 큼, 대규모 데이터에서 학습 비용 문제
- overfitting 위험 존재 → 정규화 필요
- temporal dynamics 반영 (시계열 정보 포함) 후속 연구 필요
- 더 효율적인 대규모 학습 기법 필요