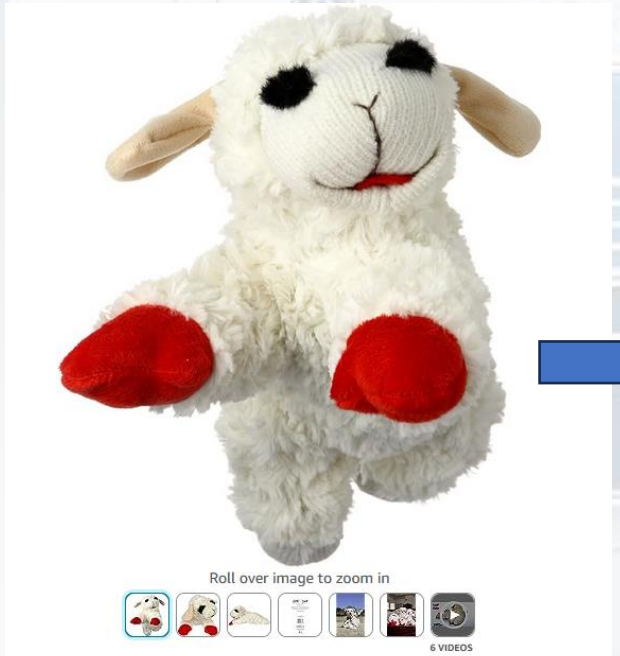
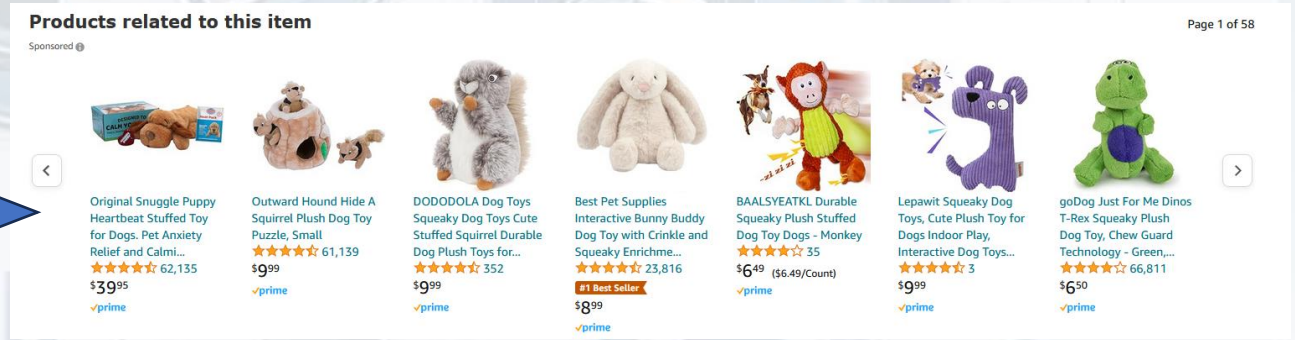
Recommending products for e-commerce using a laptop

Johan Jansen

Unsupervised learning – Final assignment
https://github.com/jjzd83/UML-wk5

Challenge: Getting from A to B





Results

- After testing three algorithms we were able to score 20,7% correctly predicted items using Alternative least squares and a laptop for 11 million rows.

Dataset

- Large e-commerce store, 2 csv for 2 consecutive months
- 14,68 gb or 11,1 million entries
- Source: ref #1, #2

- Converted to parquet and a sparse matrix
  - Each row a user session
  - Each column a # visits per product

Scoring: Precision @ 10 and Mean absolute precision

- Precision at k (where k=10): Out of 10 suggestions, how many are actually visited?
- Mean absolute precision: measures errors in confidence

Models

- Alternating Least Squares (ALS): This model decomposes the interaction matrix (Ref #3). Has linear assumptions, scales well.
- Bayesian Personalized Ranking (BPR): BPR uses a pairwise ranking approach. It optimizes a pairwise loss function. (Ref #4). Compute intensive, but ranks well.
- Logistic Matrix Factorization: A probabilistic approach that incorporates logistic regression to predict the probability of interaction. Less scalable. (Ref #5)

Results different models (default settings)

- Small set:

Model	Precision@10	MAP@10
ALS	0.082051	0.050657
BPR	0.000000	0.000000
LMF	0.066667	0.045632

- Large set:

Model	Precision@10	MAP@10
ALS	0.120611	0.059842
BPR	0.087190	0.037055
LMF	0.019175	0.006370

Results after optimization with ALS

- Using scikit optimize to find optimal settings for:
  - Factors: How many columns for the intermediate matrix?
  - Regularisation: How much punishment for a mistake
  - Alpha: How to weight a positive example against a non-existing?
- Ran the final test set and found the following scores of:

Dataset	Precision @ 10
Validation-set	20.8%
Test-set	20.7%

Conclusion

- Laptop setup:
- Alternating least squares:
  - Real-life score (Precision @ 10): 20,7%

Discussion (1/2)

Lessons learned:

- Improvement in running the experiments
- Using the counts data improved scores
- Stay critical of parameters when training with a small dataset.

Discussion (2/2)

Future work:

- Translate views into purchases
- Improve Logistic matrix factorisation by incorporating category data

References

1. https://www.kaggle.com/datasets/mkechinov/ecommerce-behavior-data-from-multi-category-store
2. https://rees46.com/en/open-cdp
3. Fast Matrix Factorization for Online Recommendation with Implicit Feedback, He et al, 2017
4. BPR: Bayesian Personalized Ranking from Implicit Feedback, Rendle et al, 2009
5. Logistic Matrix Factorization for Implicit Feedback Data, Johnson, 2014