

Master Thesis
Recommender Systems Comparison

Vasileios Symeonidis

27-05-2017

Contents

I	Master Thesis	i
1	Intro	i
2	Collaborative filtering	i
2.1	Content based	i
2.2	Latent Factors	i
3	Our Experiment	i
3.1	Infrastructure	i
3.1.1	Apache Spark	i
3.2	Dataset	ii
3.3	Metrics	ii
3.3.1	Mean Absolute Error	ii
3.3.2	Execution Time	ii
4	Results	ii
5	Conclusion	ii
6	References	iii
II	Appendices	iv
A	Code used	iv
A.1	User Based Collaborative Filtering	iv
A.2	Product Based Collaborative Filtering	iv
A.3	Latent Factors	iv
A.4	infra code	iv
B	Metrics	iv
B.1	What is the mean absolute error	iv
B.2	Time	iv

Part I

Master Thesis

1 Intro

This is the introduction for this master thesis. Why we need recommendation systems? Retailers can propose the right product to the right target group. User get advertisements they may be interested in.[2]

History, what has been tried so far?

2 Collaborative filtering

What is collaborative filtering. [2]

2.1 Content based

$$w = R^{-1}M^T$$

Normalized

$$w = (\lambda I + R^T R)^{-1} R^T M$$

2.2 Latent Factors

test sadsad
sad
asd
ds

$$w = (\lambda I + X^T X)^{-1} X^T Y$$

Figure 1: **Equation**

3 Our Experiment

3.1 Infrastructure

3.1.1 Apache Spark

[1]

3.2 Dataset

[3]

3.3 Metrics

3.3.1 Mean Absolute Error

3.3.2 Execution Time

4 Results

Table 1: Content Based Algorithm Results

Training Dataset	Testing Dataset	Mean Absolute Error	Execution time (ms)
ml-100k/u1.base	ml-100k/u1.test	1.6467431428213226	30514
ml-100k/u2.base	ml-100k/u2.test	1.6055222166704628	27714
ml-100k/u3.base	ml-100k/u3.test	1.608925907479106	27164
ml-100k/u4.base	ml-100k/u4.test	1.6259192043203685	26687
ml-100k/u5.base	ml-100k/u5.test	1.6284658627202895	27124
ml-100k/ua.base	ml-100k/ua.test	1.6425364580036836	26640
ml-100k/ub.base	ml-100k/ub.test	1.6357196576385744	26861

Table 2: Latent Factors Algorithm Results

Training Dataset	Testing Dataset	Mean Absolute Error	Execution time (ms)
ml-100k/u1.base	ml-100k/u1.test	1.1818684937209607	10195
ml-100k/u2.base	ml-100k/u2.test	1.1800652808093945	6517
ml-100k/u3.base	ml-100k/u3.test	1.1783366748334452	5377
ml-100k/u4.base	ml-100k/u4.test	1.1730543877181654	5433
ml-100k/u5.base	ml-100k/u5.test	1.1686585291940668	5217
ml-100k/ua.base	ml-100k/ua.test	1.2008035300836668	5214
ml-100k/ub.base	ml-100k/ub.test	1.2134460078406009	5083

5 Conclusion

As a conclusion we can see that als is better on both metrics from the content based.

6 References

References

- [1] “Apache Spark lightning-fast cluster computing.” <https://spark.apache.org/>. Accessed: 2017-05-21.
- [2] P. Melville and V. Sindhwani, “Recommender systems,” *Encyclopedia of Machine Learning and Data Mining*, pp. 1056–1066, 2017.
- [3] “MovieLens grouplens.” <https://grouplens.org/datasets/movielens/>. Accessed: 2017-05-22.

Part II

Appendices

A Code used

A.1 User Based Collaborative Filtering

A.2 Product Based Collaborative Filtering

A.3 Latent Factors

A.4 infra code

B Metrics

B.1 What is the mean absolute error

B.2 Time

List of Tables

1	Content Based Algorithm Results	ii
2	Latent Factors Algorithm Results	ii

List of Figures

1	Equation	i
---	--------------------	---