Amazon Electronic Products Network Analysis

EE-558 A Network Tour of Data Science

Süha Kagan Köse - Tao Sun - Xiangzhe Meng - Xingce Bao

Introduction

- Amazon is reshaping and recording our life
- Invisible social network
- Products in Amazon are connected together
- Analyse electronic product network




Data acquisition & exploration

- The electronic subset of **Amazon Product Dataset**, provided by Julian McAuley, UCSD

- Product **metadata** (descriptions, category information, price, brand image features and product **links**: also viewed/ also bought)

- Millions of **reviews** (**reviewer ID**, ratings, text) spanning from May 1996 to July 2014

- **Electronic product metadata** and **5-core review datasets**

Before preprocessing

Electronic product metadata

	asin	imUrl	description	categories	title	price	salesRank	related	brand
0	0132793040	http://ecx.images-amazon.com/images/I/31JIPhp%...	The Kelby Training DVD Mastering Blend Modes i...	[[Electronics, Computers & Accessories, Cables...	Kelby Training DVD: Mastering Blend Modes in A...	NaN	NaN	NaN	NaN
1	0321732944	http://ecx.images-amazon.com/images/I/31uogm6Y...	NaN	[[Electronics, Computers & Accessories, Cables...	Kelby Training DVD: Adobe Photoshop CS5 Crash ...	NaN	NaN	NaN	NaN
2	0439886341	http://ecx.images-amazon.com/images/I/51k0qa8f...	Digital Organizer and Messenger	[[Electronics, Computers & Accessories, PDAs, ...	Digital Organizer and Messenger	8.15	{'Electronics': 144944}	{'also_viewed': ['0545016266', 'B009ECM8QY', '...	NaN

Electronic product reviews

	reviewerID	asin	reviewerName	helpful	reviewText	overall	summary	unixReviewTime	reviewTime
0	AO94DHGC771SJ	0528881469	amazdnu	[0, 0]	We got this GPS for my husband who is an (OTR)...	5.0	Gotta have GPS!	1370131200	06 2, 2013
1	AMO214LNFCEI4	0528881469	Amazon Customer	[12, 15]	I'm a professional OTR truck driver, and I bou...	1.0	Very Disappointed	1290643200	11 25, 2010
2	A3N7T0DY83Y4IG	0528881469	C. A. Freeman	[43, 45]	Well, what can I say. I've had this unit in m...	3.0	1st impression	1283990400	09 9, 2010

What we mainly did for data preprocessing …

Drop useless columns from two datasets.
Only keep columns asin (product ID) & related for product dataset and columns asin & reviewerID for review dataset.

For review dataset, set product asin as index.
For each product asin, we collect a list of reviewerID which stand for those who have commented on this product.

For product dataset, check the also_bought related list and drop those products which are not in the electronic product dataset.

For review dataset, only keep reviews concerning the products presented in the product dataset and drop the others.

After preprocessing

Electronic product metadata

asin	also_bought
B0083S3NC8	[B007W1KES8, B005KQ0S8S, B007W1KEFG, B005F15N2...
B0047FHOWG	[B0019SHZU0, B002ZIMEMW, B002HJ9PTO, B004AZ38Z...
B0067SJC80	[B000X1R5HM, B004MU8VCS, B002YIG9AQ, B005CTCD6...
B002WQP2IA	[B000U0HAR6, B000068O4E, B000068O4C, B00356J8K...
B006IC4YZQ	[B001G54ILA, B005SDWP3O, B0036QL1JY]

Electronic product reviews

asin	reviewerID
B0083S3NC8	[A18I2DO90GZCQY, A2C4BO8UURNWNN, A2RF9FHC4HC3J...
B0047FHOWG	[A3OTFTP2WVZVQY, A10KIQXOE926FN, A1Q165PZVZS34...
B0067SJC80	[A3QLALFN0WGF87, A395EVHF1TAQN0, AOEDXOKYPII2Z...
B002WQP2IA	[AN3ILH8NOGNH4, A3FGRP5N72WES1, ADN4437IJDIPP,...
B006IC4YZQ	[A3W3PWGZ36249Y, AZLDKR28KT3FB, A3M3DL4G9NS3Z2...

Study on MacBook and Surface

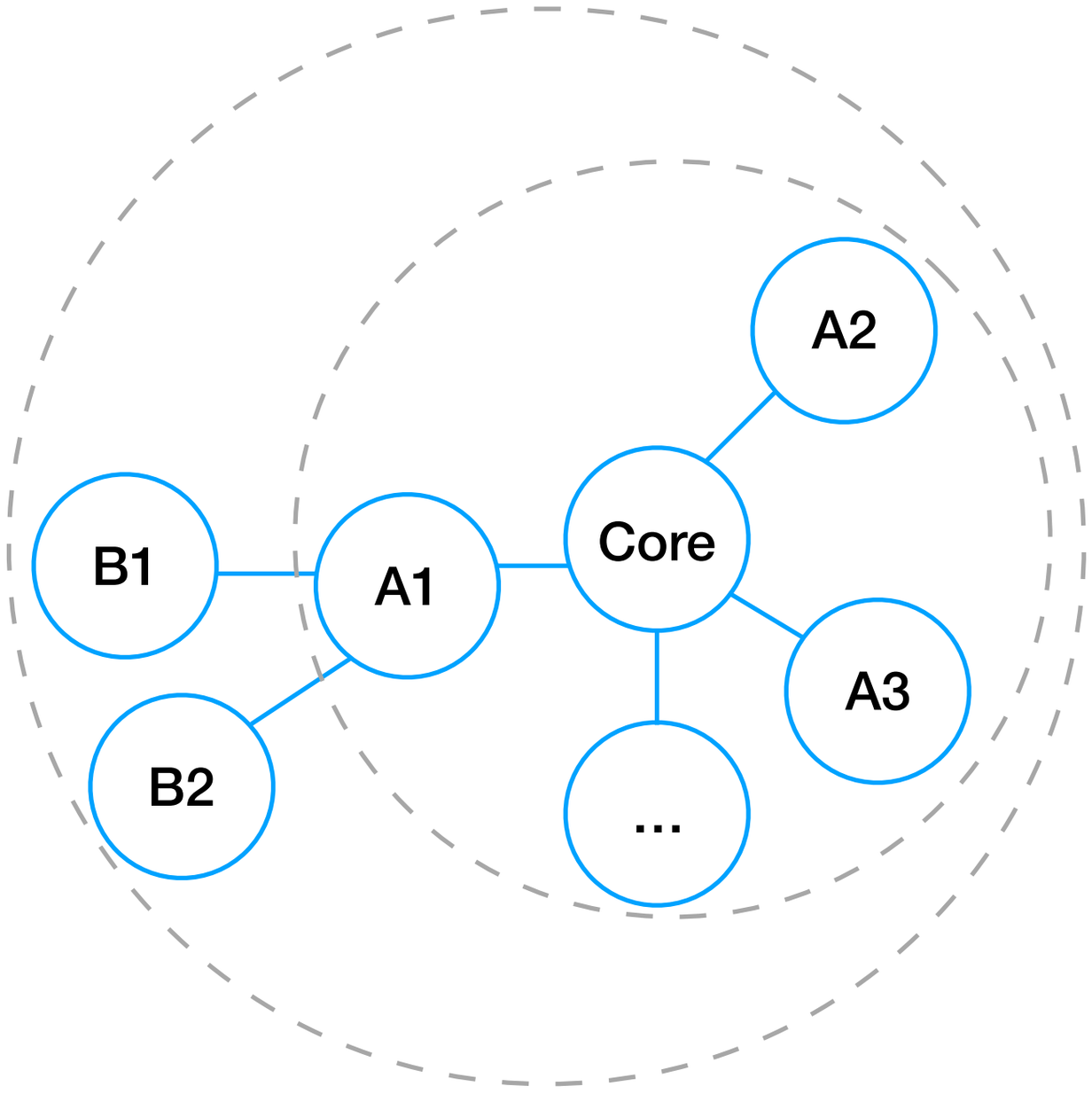
Surface

MacBook

Network Building

Selecting Nodes

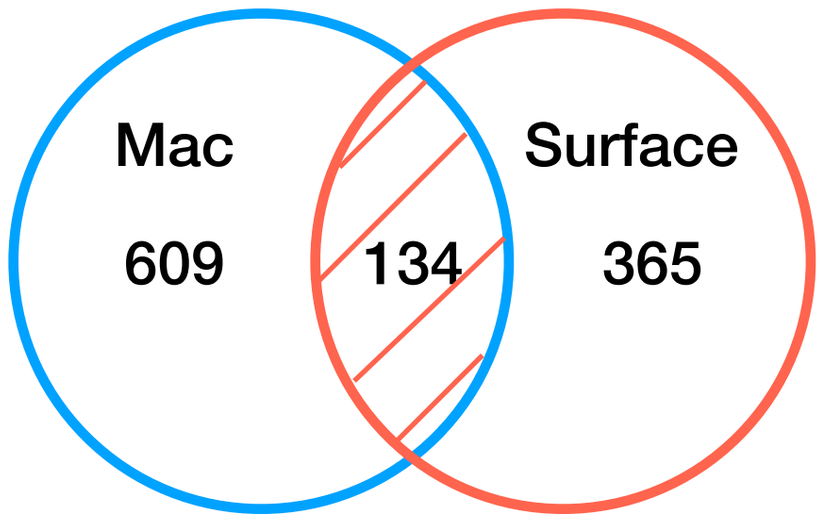
- “Core&Layer” Method
- Based on “also_bought”
- Two layers here
  - Size of dataset
  - Tractability



Network Building

Labeling Nodes

- Two cores: Mac & Surface
- Assign all duplicates to Surface
- Mac : Surface = 609 : 499
- Total: 1108



Network Building

Weighing Edges

- Two different approaches
- Weight = F(# Shared “also_bought”)
- Weight = F(# Shared “also_bought”, # Shared Reviewers)

$$Weight = 0.2 * \frac{\#Shared\_Neighbors}{Average\_Shared\_Neighbors} + 0.8 * \frac{\#Shared\_Reviewers}{Average\_Shared\_Reviewers}$$

- What will happen with different weight of the two?

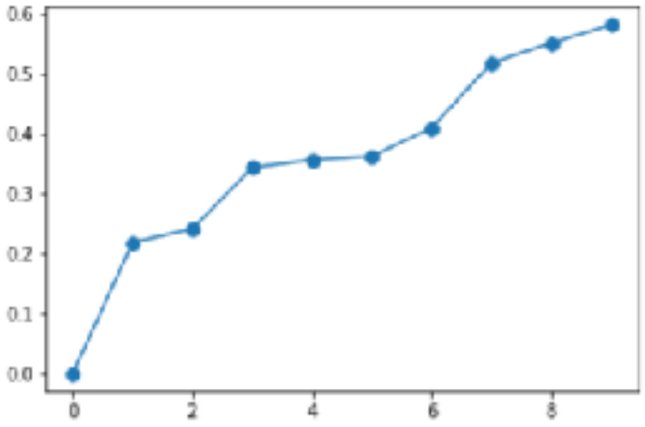
Spectral Graph Theory

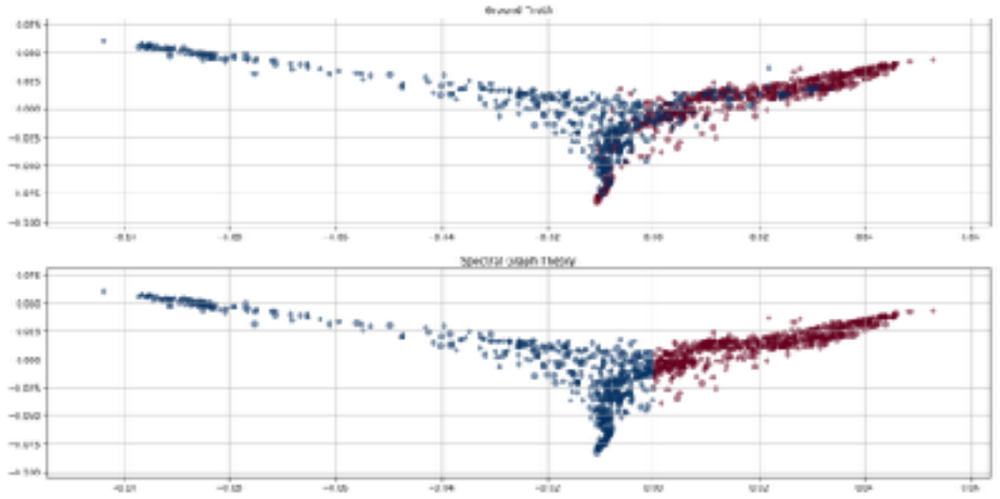
From Similarity Matrix to Laplacian Eigenmaps

- Similarity = Weight / Max_Weight
- “Distance”? No.
- Weight itself actually represents the similarity

Spectral Graph Theory

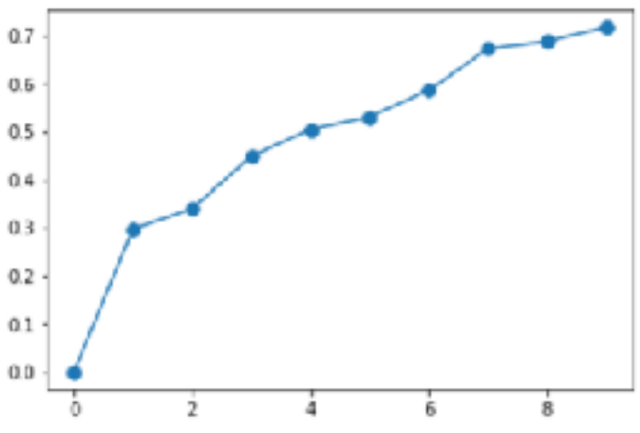
From Similarity Matrix to Laplacian Eigenmaps

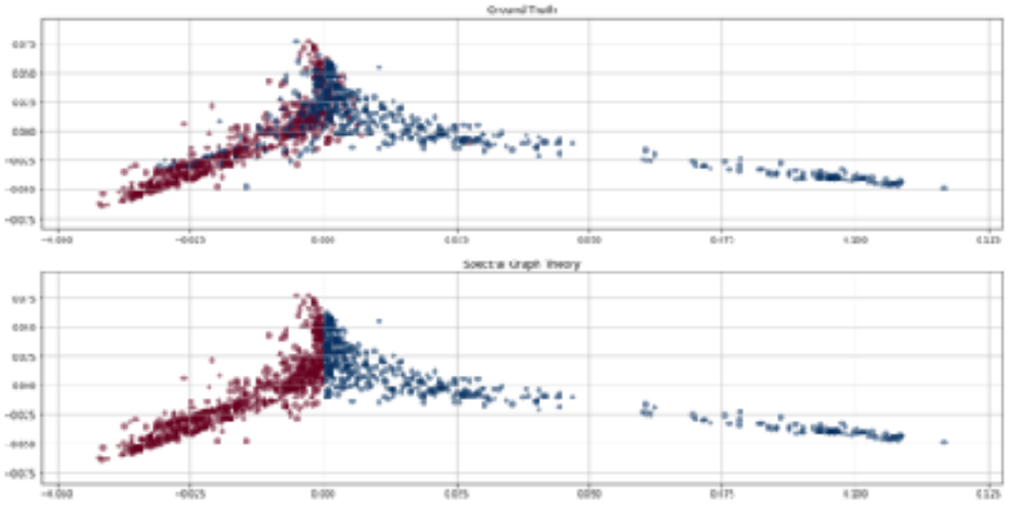




Error: 17.24 %

“also_bought”





Error: 15.61 %

With Reviewer Weight: 0.2/0.8

Spectral Graph Theory

From Similarity Matrix to Laplacian Eigenmaps

- Comparison between two weight definitions
- Information of reviewers is of importance
- also_bought/reviewer: .8 / .2 -> .5 / .5 -> .2 / .8
- Result gets better

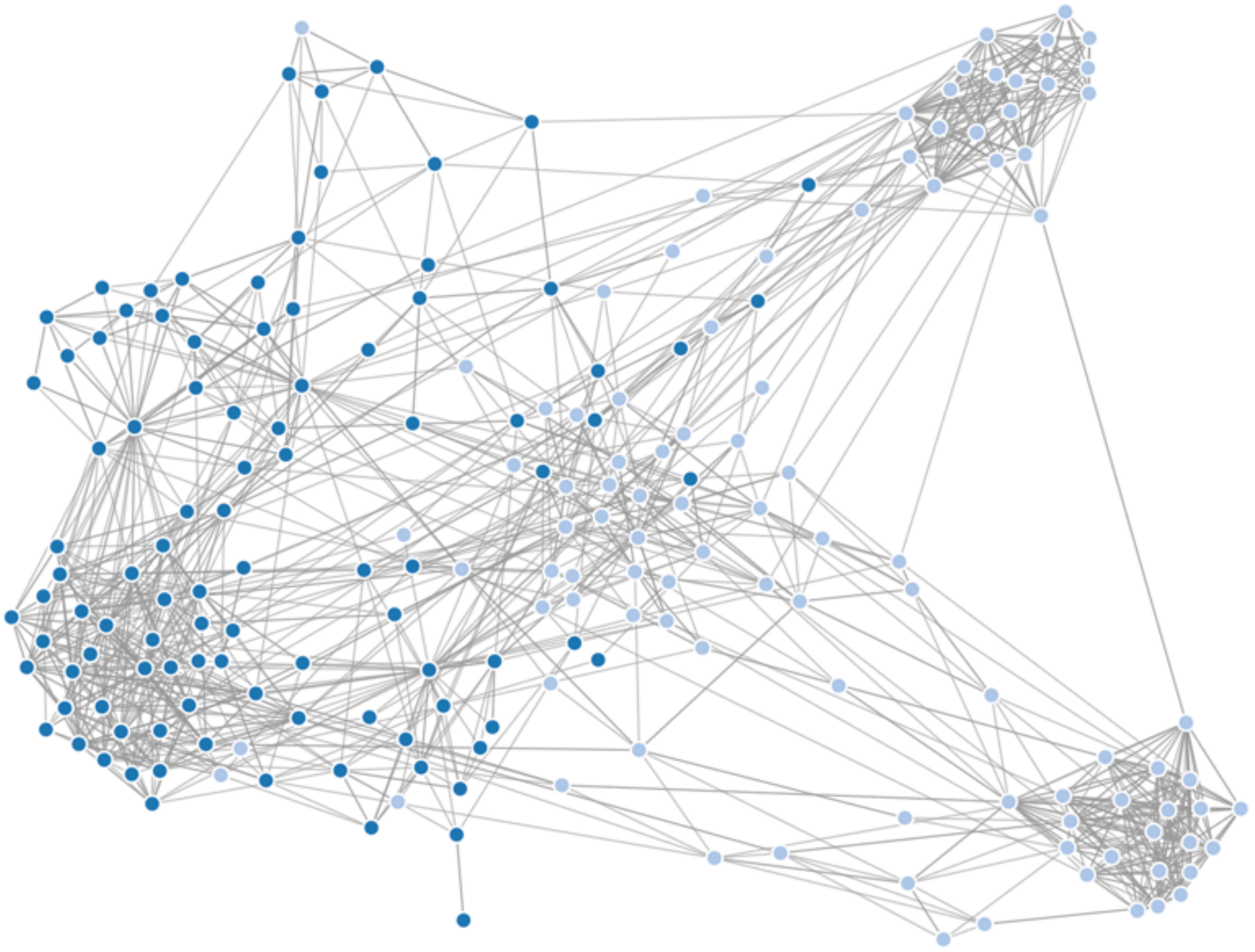
$$Weight = 0.2 * \frac{\#Shared\_Neighbors}{Average\_Shared\_Neighbors} + 0.8 * \frac{\#Shared\_Reviewers}{Average\_Shared\_Reviewers}$$

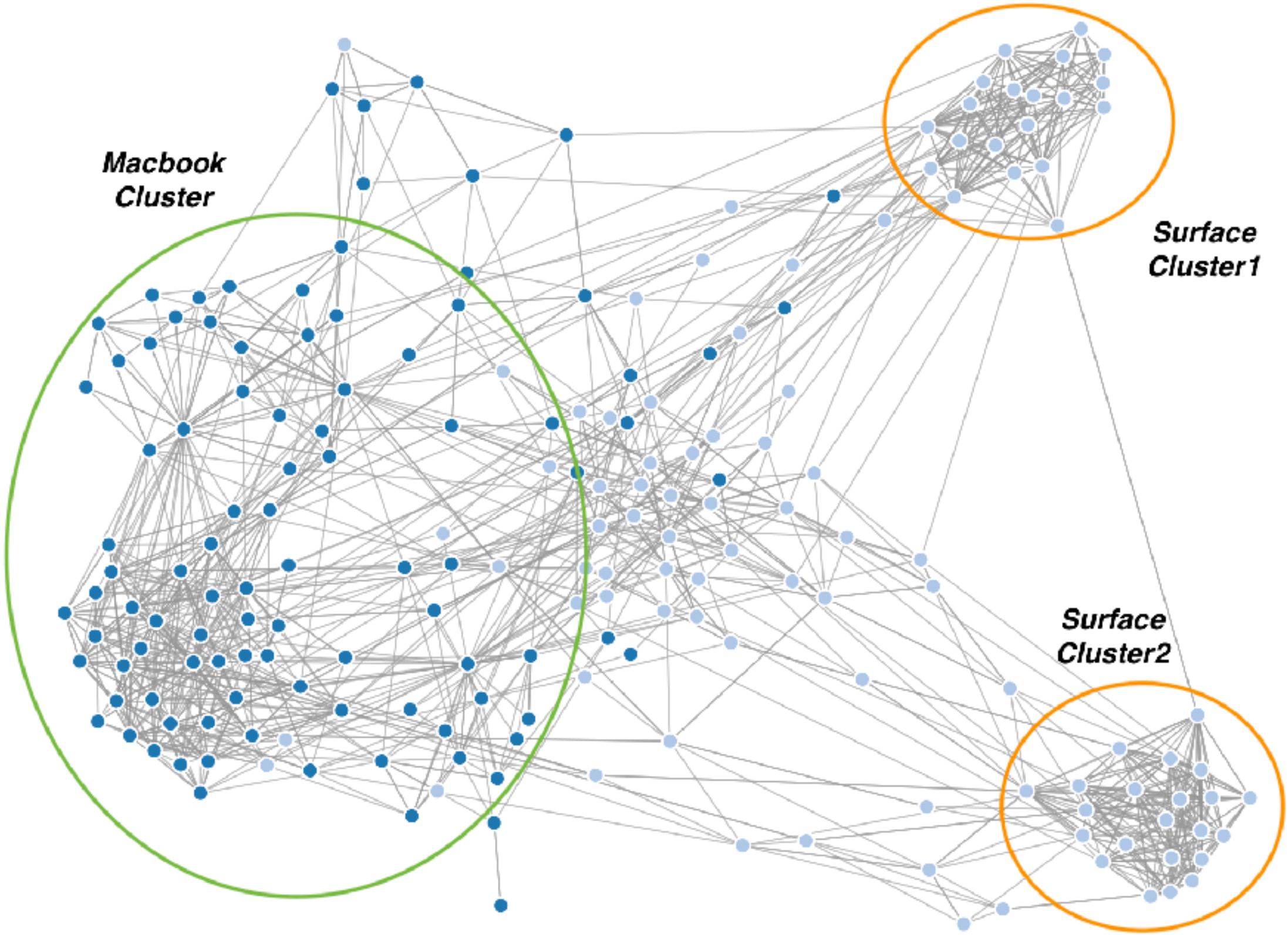
Network Graph

- Volume of the dataset is too large (# of products)
- NetworkX doesn’t work well anymore
- **D3.js**, a JavaScript library for producing dynamic, interactive data visualizations in web browsers
- **2 graphs** concerning the relationship among products
- **200 nodes** and **1830 links**

Network Visualisation

https://xiangzhemeng.github.io/ntds/index.html




Macbook
Cluster
Surface
Cluster1
Surface
Cluster2

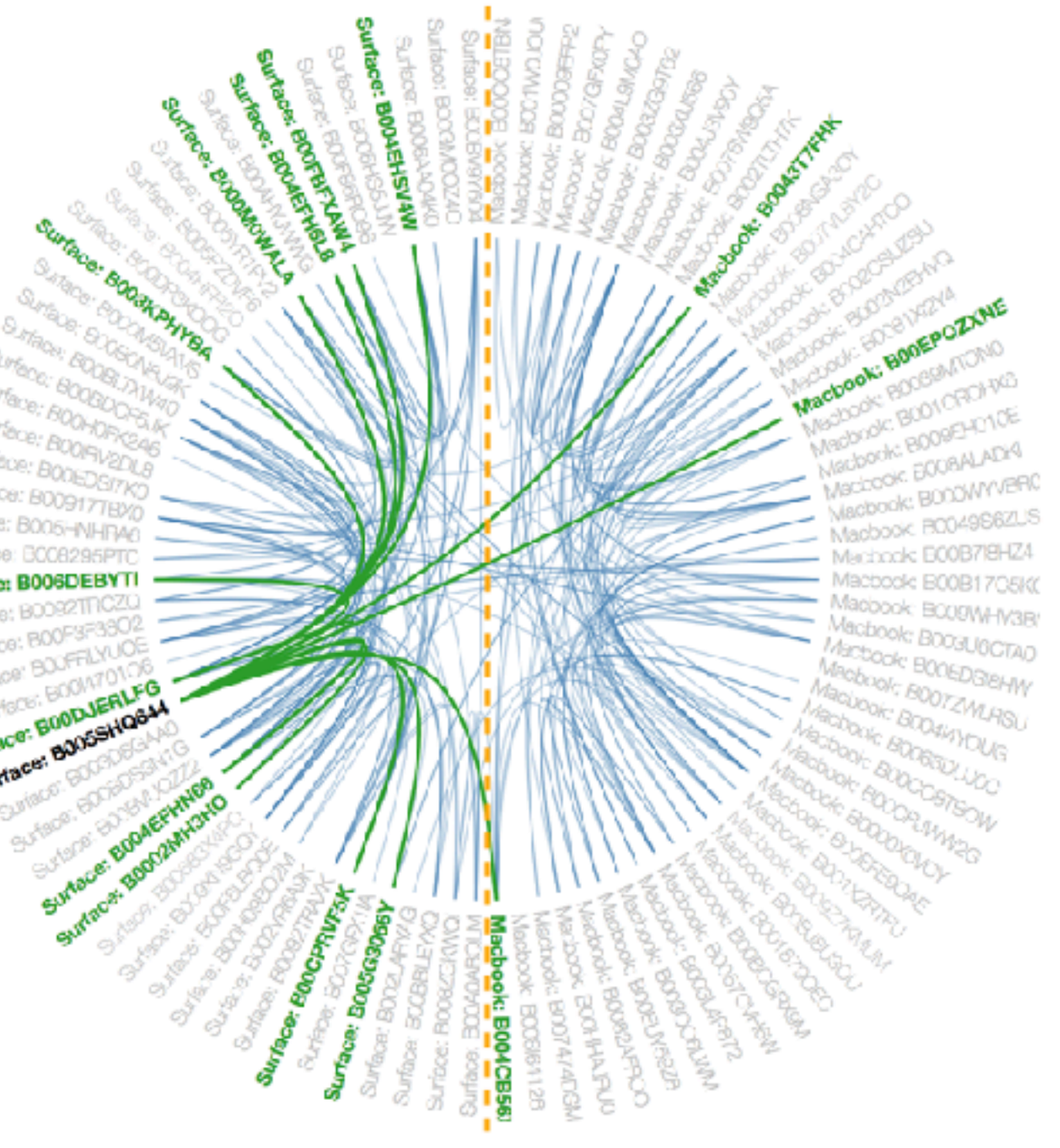
Circle-shaped Network

https://xiangzhemeng.github.io/ntds/index.html

MacBook

Surface

From MacBook to Surface



From Surface to MacBook

Animated version available online!

https://xiangzhemeng.github.io/ntds/index.html

Study on MacBook, Surface and ThinkPad

Surface

MacBook

ThinkPad

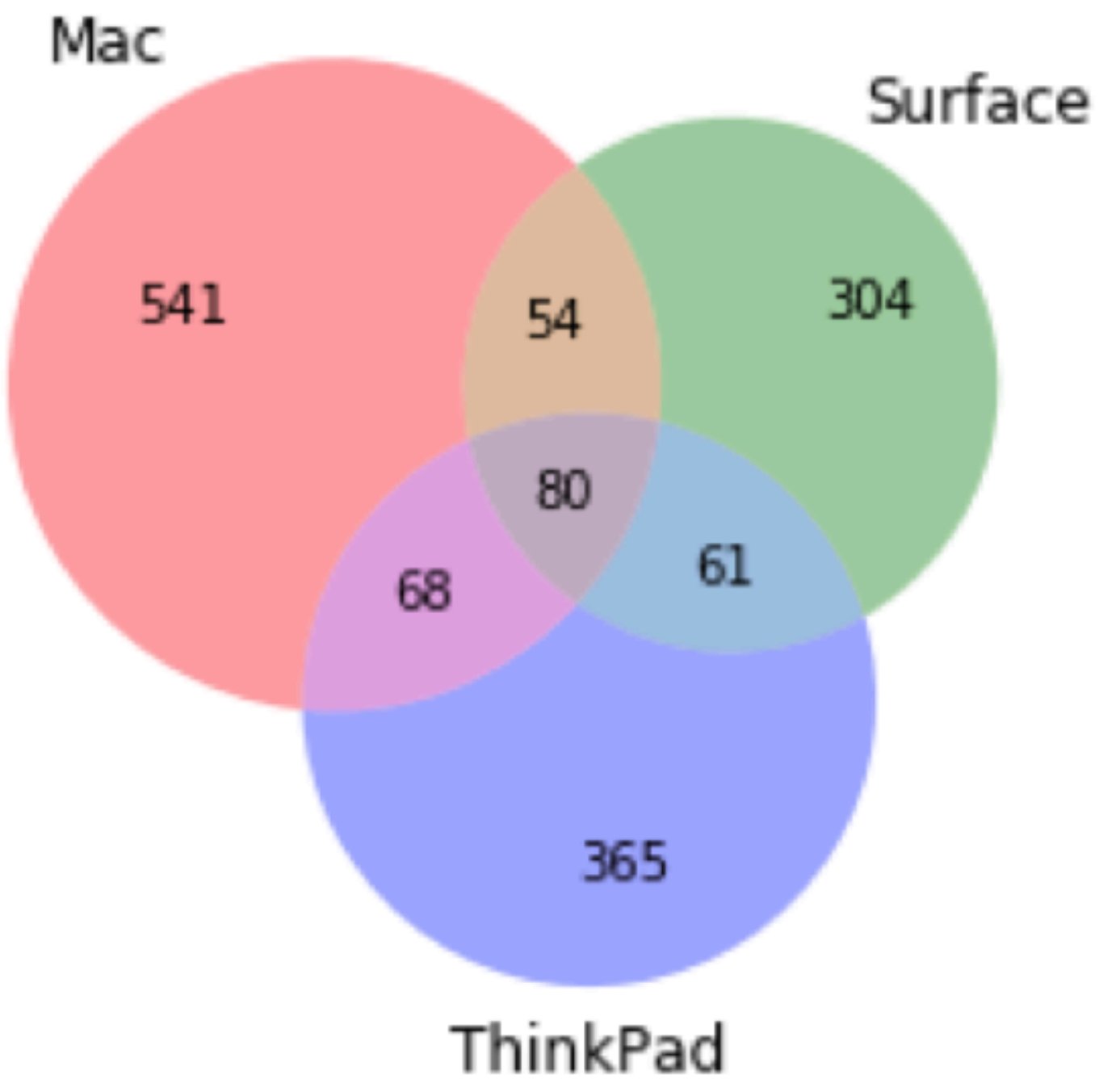
Collecting Products

Same method as before:

(I) 'also_bought' products

(II)'also_products' of those products

~700, **500**, **600** products in MacBook, Surface, ThinkPad group respectively.


Mac
Surface
541
54
304
80
68
61
365
ThinkPad

Create Network

Same method as before:

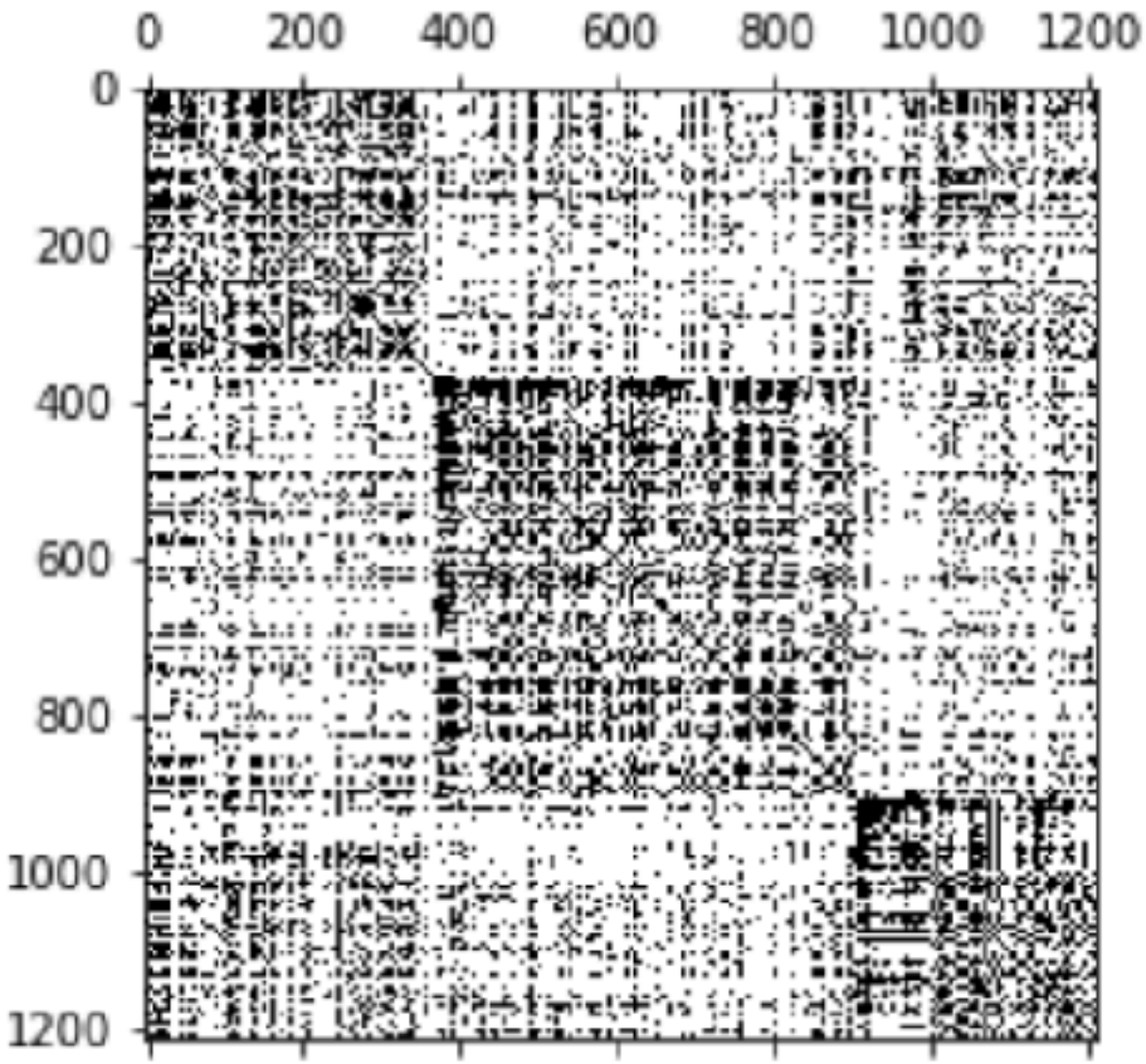
Node: Product

Edge: If 'also_bought' with another product

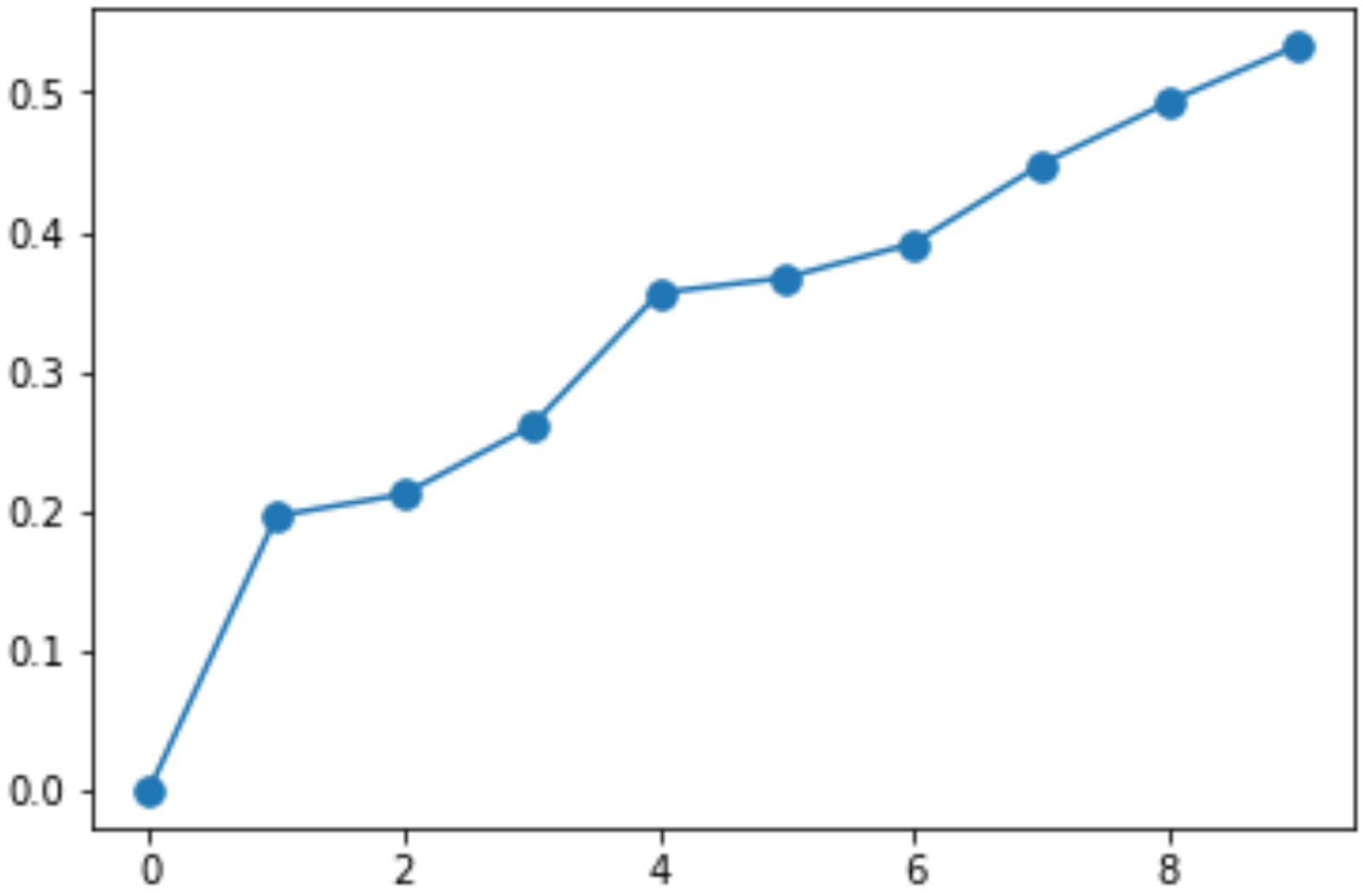
Weights: Two methods:

$$Weight = \#Shared\_Neighbors$$

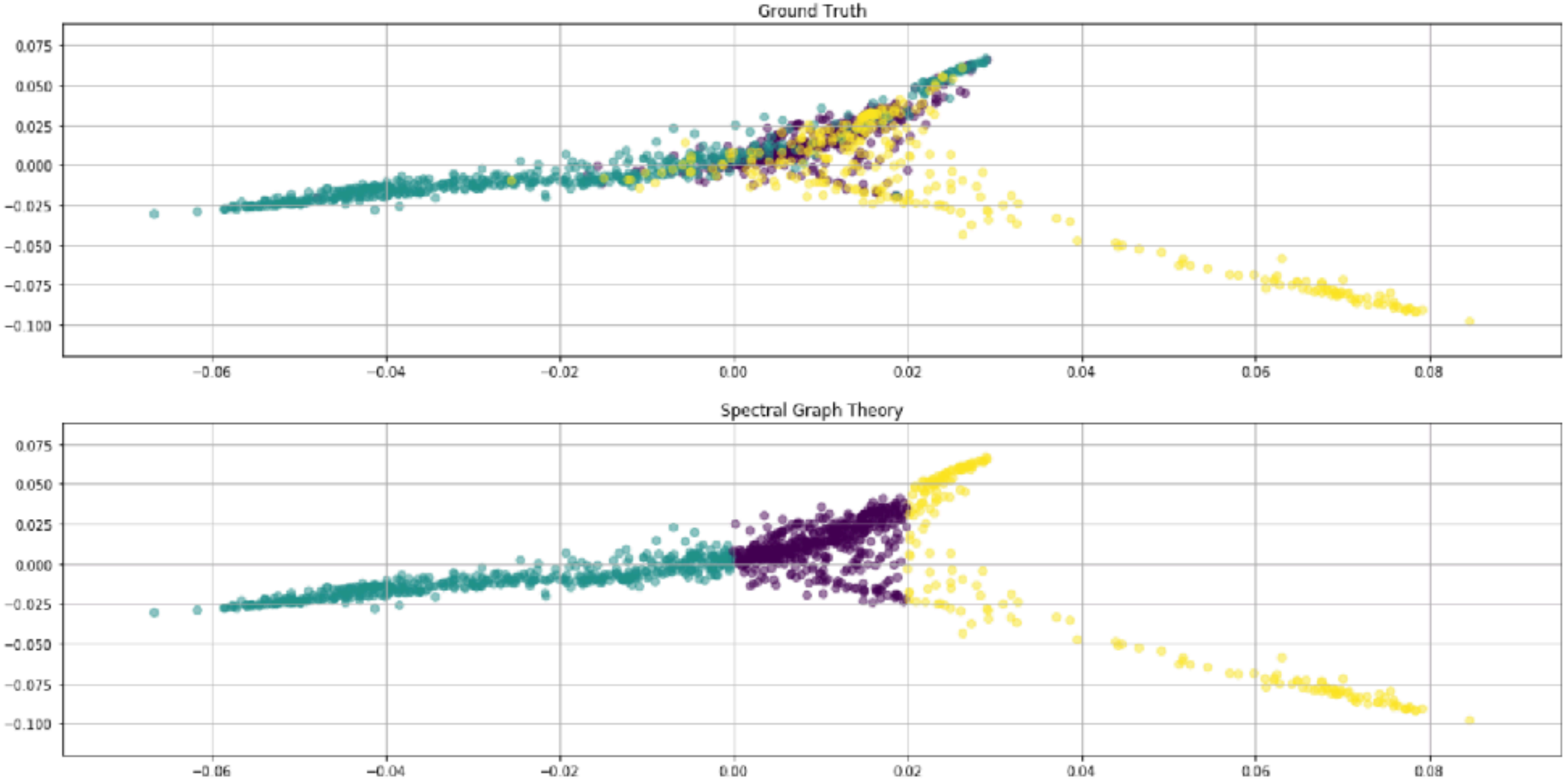
$$Weight = 0.2 * \frac{\#Shared\_Neighbors}{Average\_Shared\_Neighbors} + 0.8 * \frac{\#Shared\_Reviewers}{Average\_Shared\_Reviewers}$$



Laplacian (D-A)



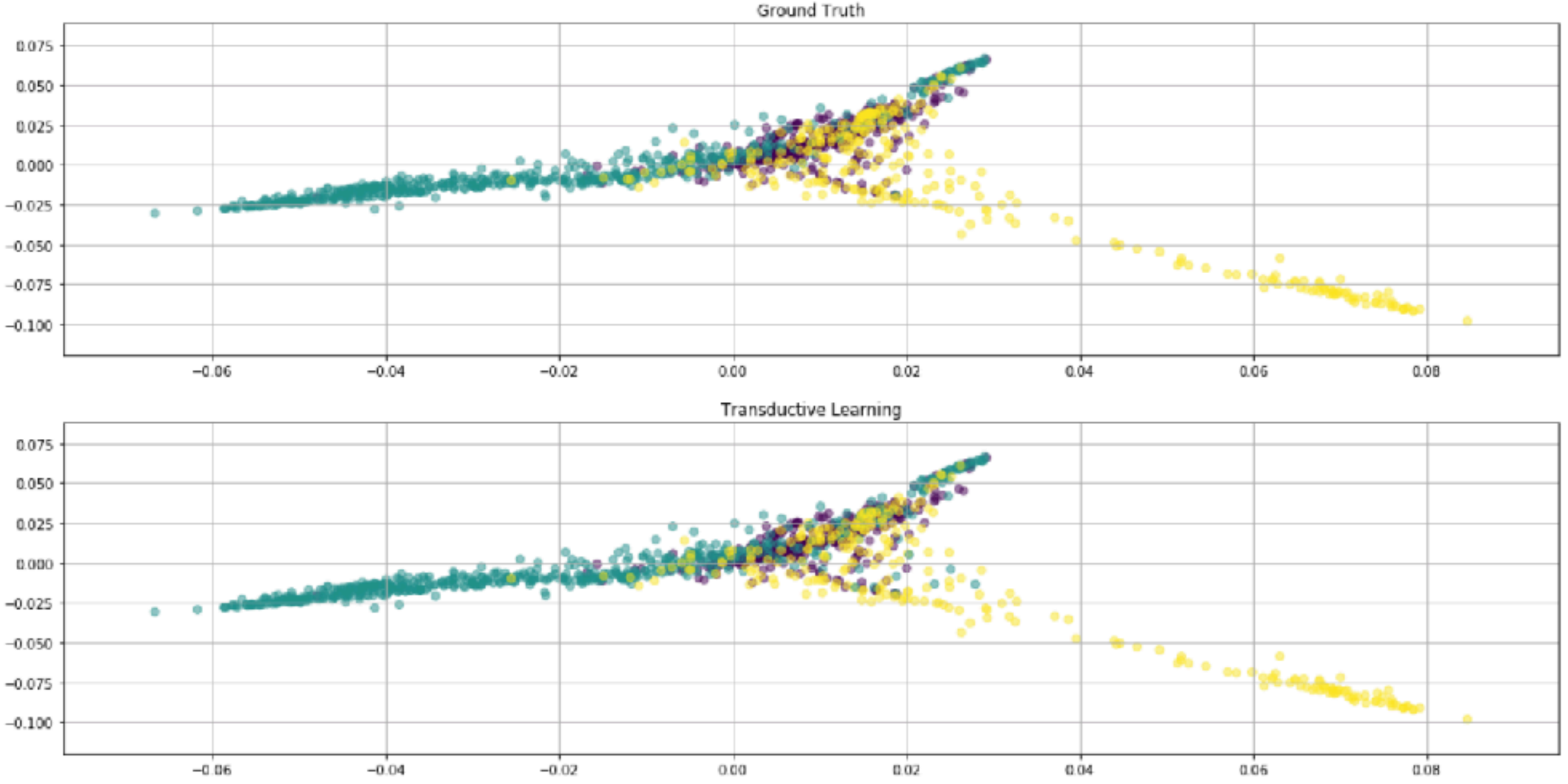
Eigenvalues



Graph with EigenValues (%29)

Transductive Learning

Alpha	Error (%)
0.1	49
0.3	33
0.5	24
0.7	13
0.9	4



Transductive Learning (%4)

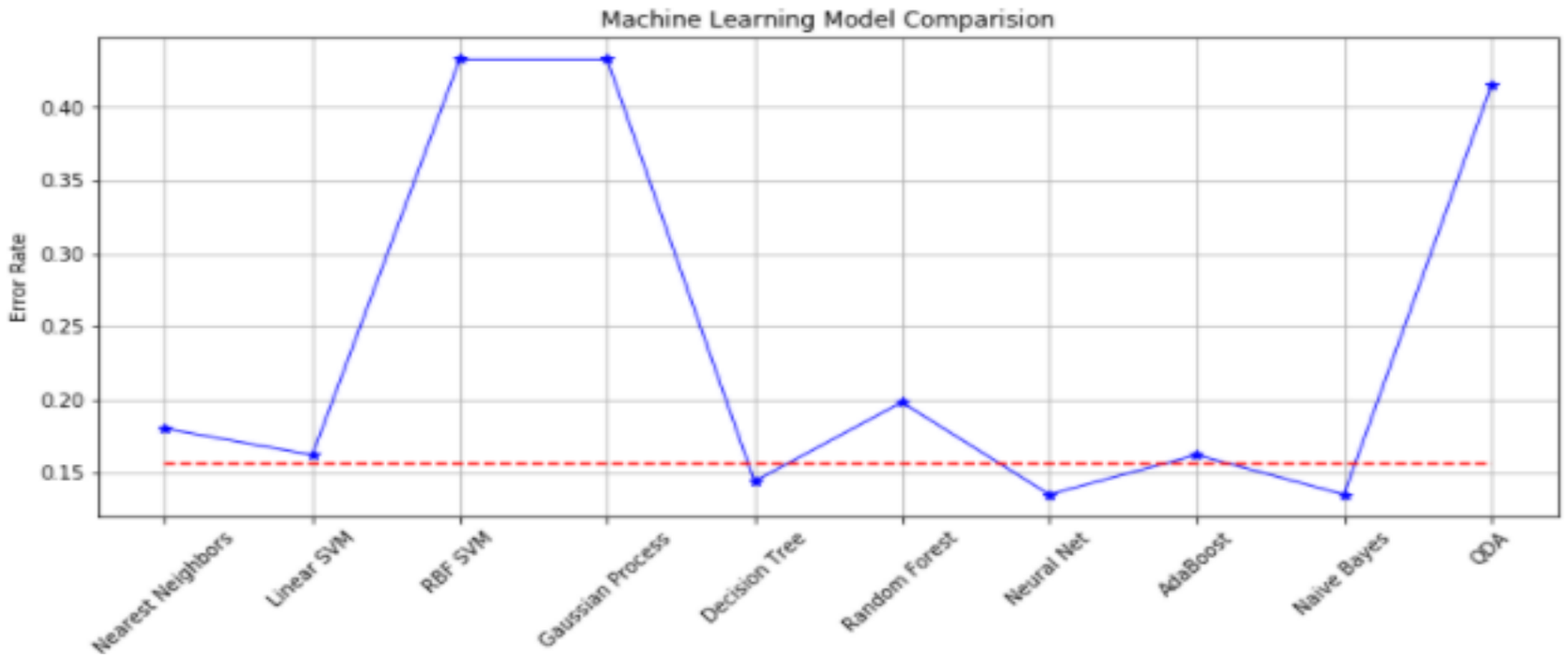
Beat Machine Learning?

- **Feature Definition:**
  - The similarity with each product
    - E.g. Three products A,B,C then three features: similarity with A, similarity with B and similarity with C
  - We can just use similarity matrix as feature matrix
- **Untuned Models:**
  - SVM, Decision Tree, Naive Bayes …

Beat Machine Learning?

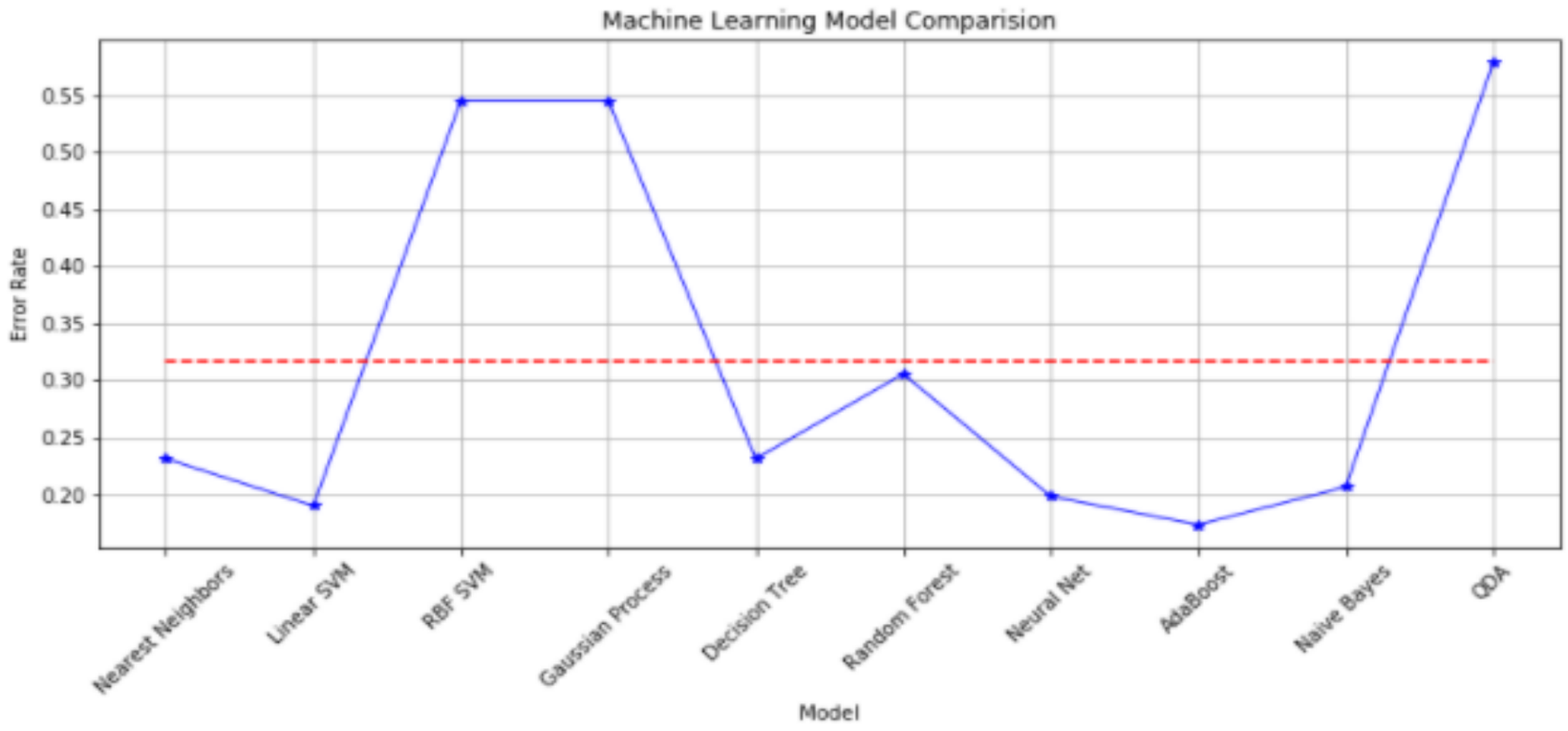
Error rate

2 groups



Error rate

3 groups



Why huge difference?

Linear SVM and SVM should be powerful but why?

Soft-margin classifier

Kernel method

Hinge loss function

Fit the maximum-margin hyperplane in a transformed feature space.

$$k(x, x') = \langle \Phi(x), \Phi(x') \rangle,$$

- **In fact, our data dimension is already very high and we do not need to map it in a higher dimensional space.**
- **When the data dimension and the sample number is similar, using the kernel method is very easy to cause overfit.**

Why huge difference?

- Gaussian process:
    - Not suitable for data with dozens of features
- QDA:
    - Overfit problem
- Random Forest & Decision Tree:
    - Not good result
- Boosting:
    - Unstable influence of decision tree

Recommendation:

- Neural Net:
    - Work pretty well even without changing a lot of parameters
- Naive Bayes:
    - Good at dealing with large number of features

Conclusion

- **Spectral Graph Theory** did a good job both in two-group and three-group classifications
- The **information of reviewers** directly reflects the connection of two products
- **Manually chosen cores** —> clusters may not be that obvious
- **How to label our datapoints** right and meaningful becomes our further research direction

✦ **Thanks for your listening! Any questions?**