Scientific Paper Clustering

Darren Liu

Introduction

Motivation: There are lots of scientific papers online with different topics. Clustering papers that shares similar topics can provide researchers paper recommendations to help recommend them papers that they would be interested in.

Objectives and goals: Cluster scientific papers and provide a recommender that recommends five related papers for each paper.

Methodology

The data I used:

The data I used comes from arXiv dataset.

It contains data for 2,142,715 scientific papers including authors, title, doi, categories, abstract, etc.

Methodology

Tools I used:

Pandas for exploratory data analysis

Matplotlib and Seaborn for plotting

Scikit Learn and spaCy for modeling and natural language processing

Methodology

How I used the data:

1. Convert the original JSON file into a dataframe.
2. Only select papers which the latest versions are released after 2020.
3. Drop duplicate rows.
4. Randomly ample 50,000 rows from the filtered dataframe.
5. Get rid of all the '\n' characters in their abstracts.
6. Get stop words from package spacy.lang.en.stop_words and add customized stop words.
7. Get punctuations from package string.punctuations.
8. Use package en_core_sci_lg from spacy for lemmatization and filter all stop words and punctuations.

Methodology

Topic modeling:

Use LSA to turn the words into 20 topics.

Methodology

Word to vector:

Use TfidfVectorizer to turn abstracts into vectors.

- min_df = 0.016
- max_df = 0.05

Methodology

Topic modeling:

Use LSA to turn the words into 20 topics.

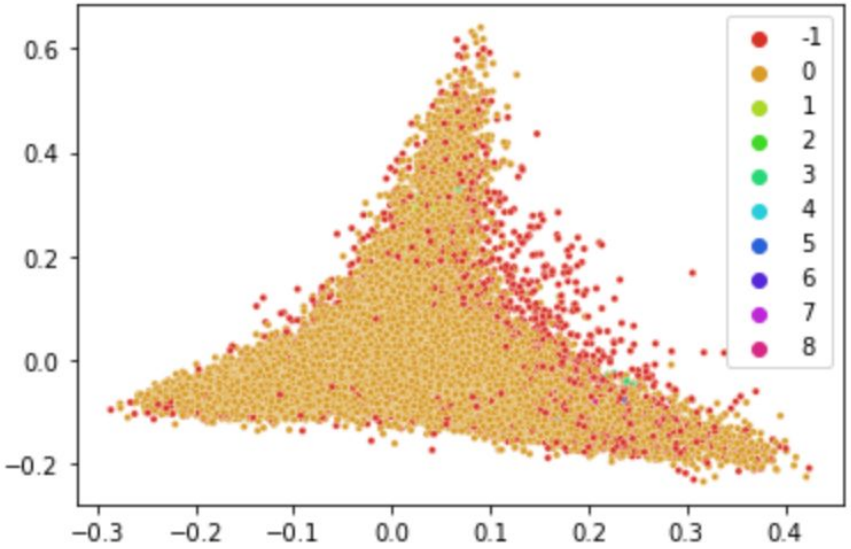
- 2d, background, detector, imaging, light, reconstruction, resolution, sensitivity, sensor, ...
- answer, gravitational, language, natural, production, question, resource, scalar, search, social, ...
- channel, link, metric, platform, software, technology, user, …

Results

DBSCAN clustering:

Hyperparameters:

- epsilon = 0.15
- min_samples = 5

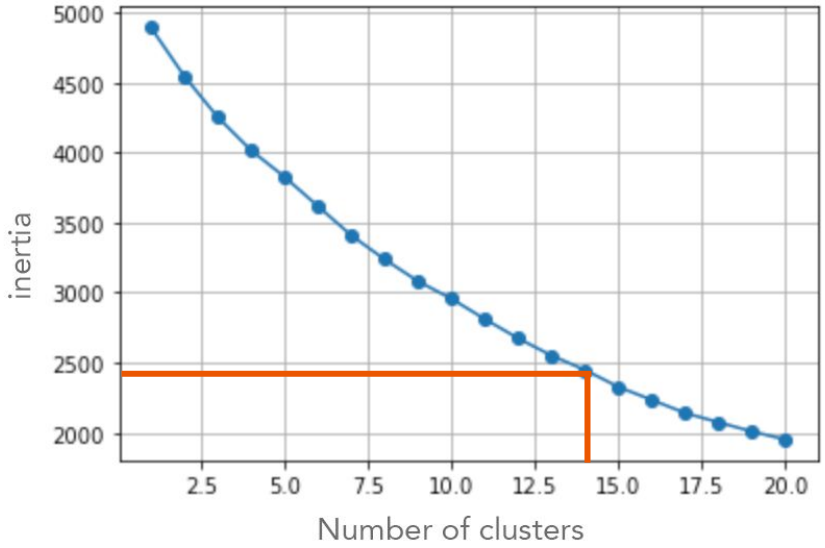


Results

K-means clustering:

Select number of clusters:

- K = 14



Results

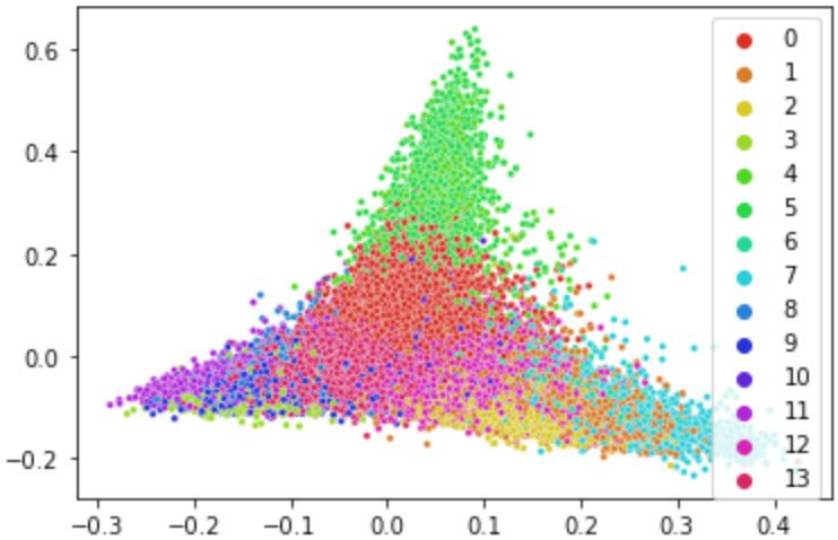
K-means clustering:

Select number of clusters:

- K = 14

Other hyperparameters:

- random_state = 42
- n_init = 20

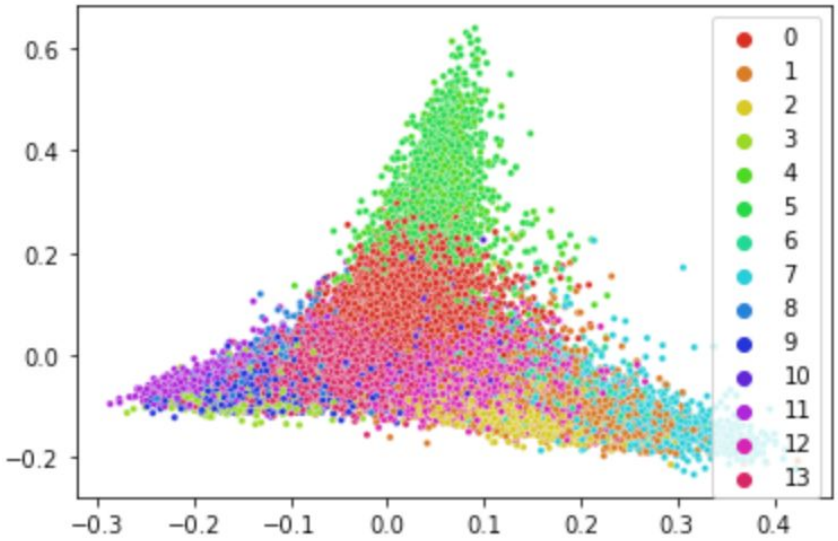


Results

K-means clustering:

Some cluster examples:

- On Whitehead's cut vertex lemma
- Combinatorics on bounded free Motzkin paths and its applications
- Persistent sheaf cohomology
- …



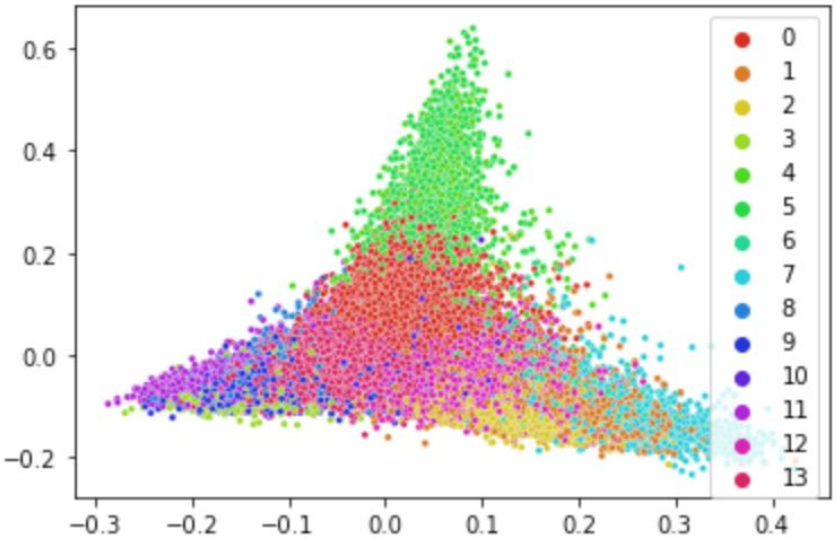
Results

K-means clustering:

Some cluster examples:

- On Whitehead's cut vertex lemma
- Combinatorics on bounded free Motzkin paths and its applications
- Persistent sheaf cohomology
- …

Math

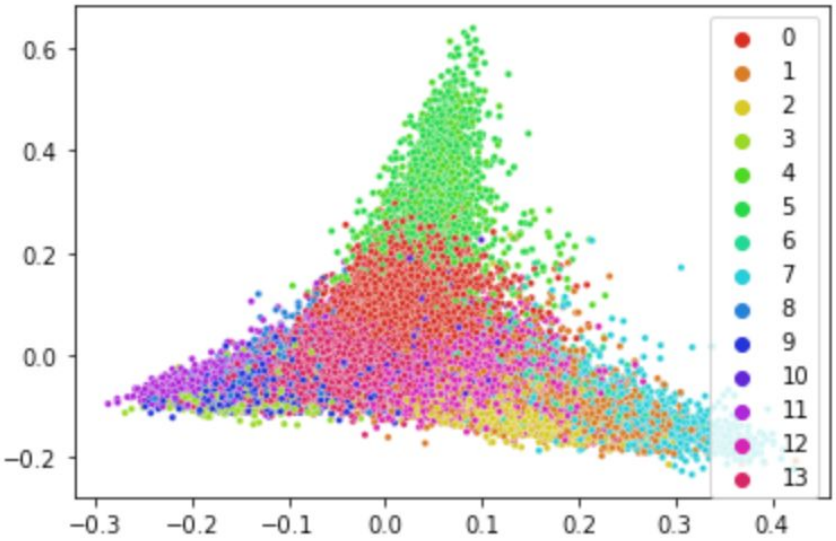


Results

K-means clustering:

Some cluster examples:

- Analyzing time series activity of Twitter political spambots
- DeF-DReL: Systematic Deployment of Serverless Functions in Fog and Cloud environments using Deep Reinforcement Learning
- Measuring what Really Matters: Optimizing Neural Networks for TinyML
- …

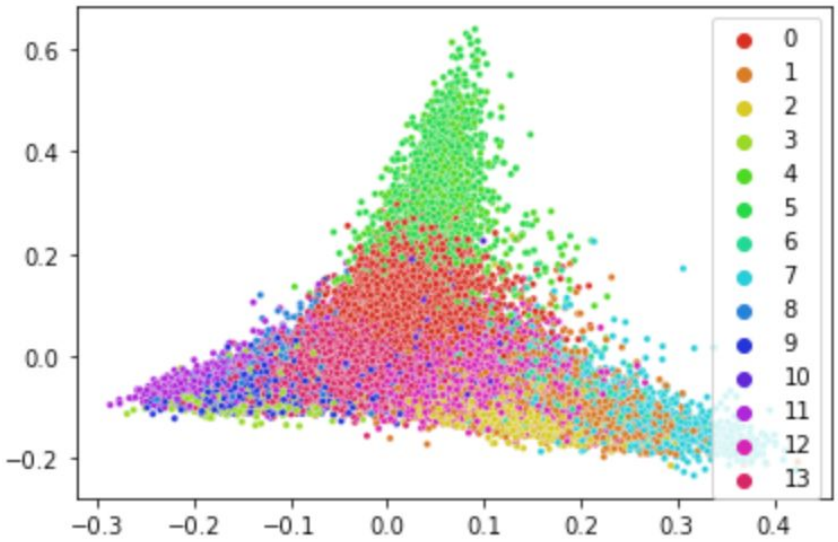


Results

K-means clustering:

Some cluster examples:

- Analyzing time series activity of Twitter political spambots
- DeF-DReL: Systematic Deployment of Serverless Functions in Fog and Cloud environments using Deep Reinforcement Learning
- Measuring what Really Matters: Optimizing Neural Networks for TinyML
- …



Conclusions

K-means clustering is obviously a better method for this dataset.

Some recommendations the recommender provides:

For paper “On Whitehead's cut vertex lemma”:

Recommended papers in the same cluster:

- Local rainbow colorings for various graphs
- Even vertex ζ-graceful labeling on Rough Graph
- Revisiting and improving upper bounds for identifying codes

Conclusions

K-means clustering is obviously a better method for this dataset.

Some recommendations the recommender provides:

For paper “On Whitehead's cut vertex lemma”:

Recommended papers in different clusters:

- Numerical Solution of the 3-D Travel Time Tomography Problem
- On irreducible representations of a class of quantum spheres

Future Work

1. To expand the dataset and keep adding latest papers.
2. Get individual researcher’s information to provide better recommendations

Thanks for your attention!