Software mention clustering using machine learning

Daniel Garijo and Jinseok Kim

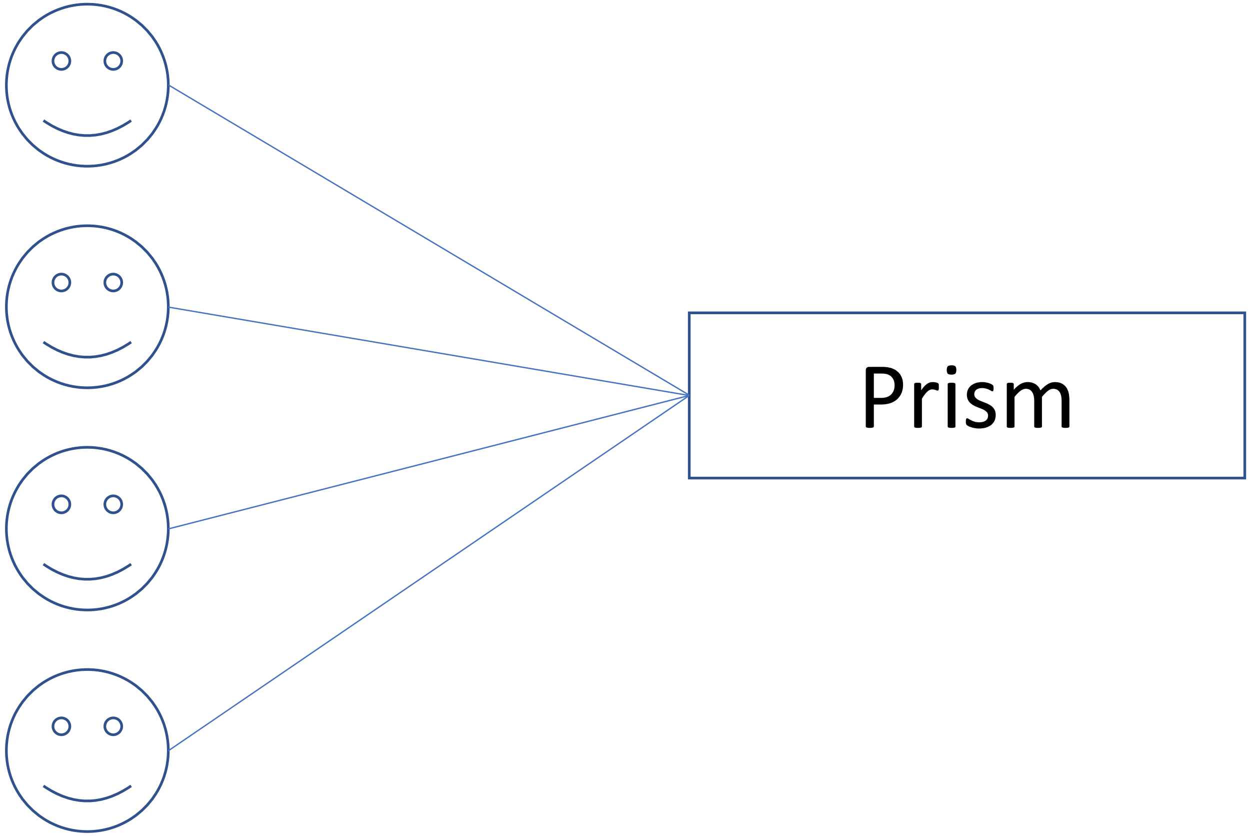
Mapping the Impact of Research Software in Science

Chan Zuckerberg Initiative, Redwood City, California

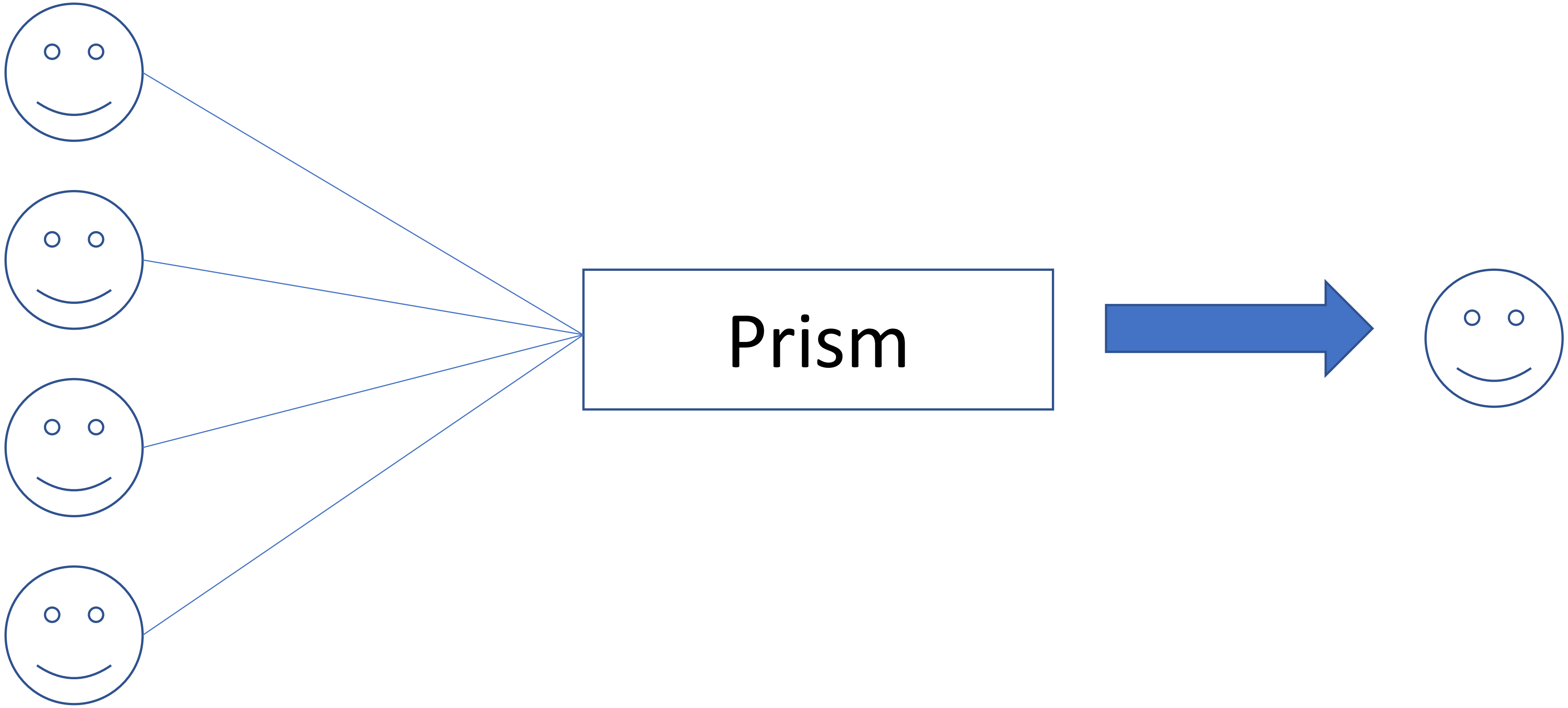
24-27 October 2023

- Figures and tables presented in these slides are sourced directly from a paper currently under review:
- Jinseok Kim and Jenna Kim (Under Review). 'ANDez: An Open-Source Python Tool for Author Name Disambiguation Using Machine Learning'.

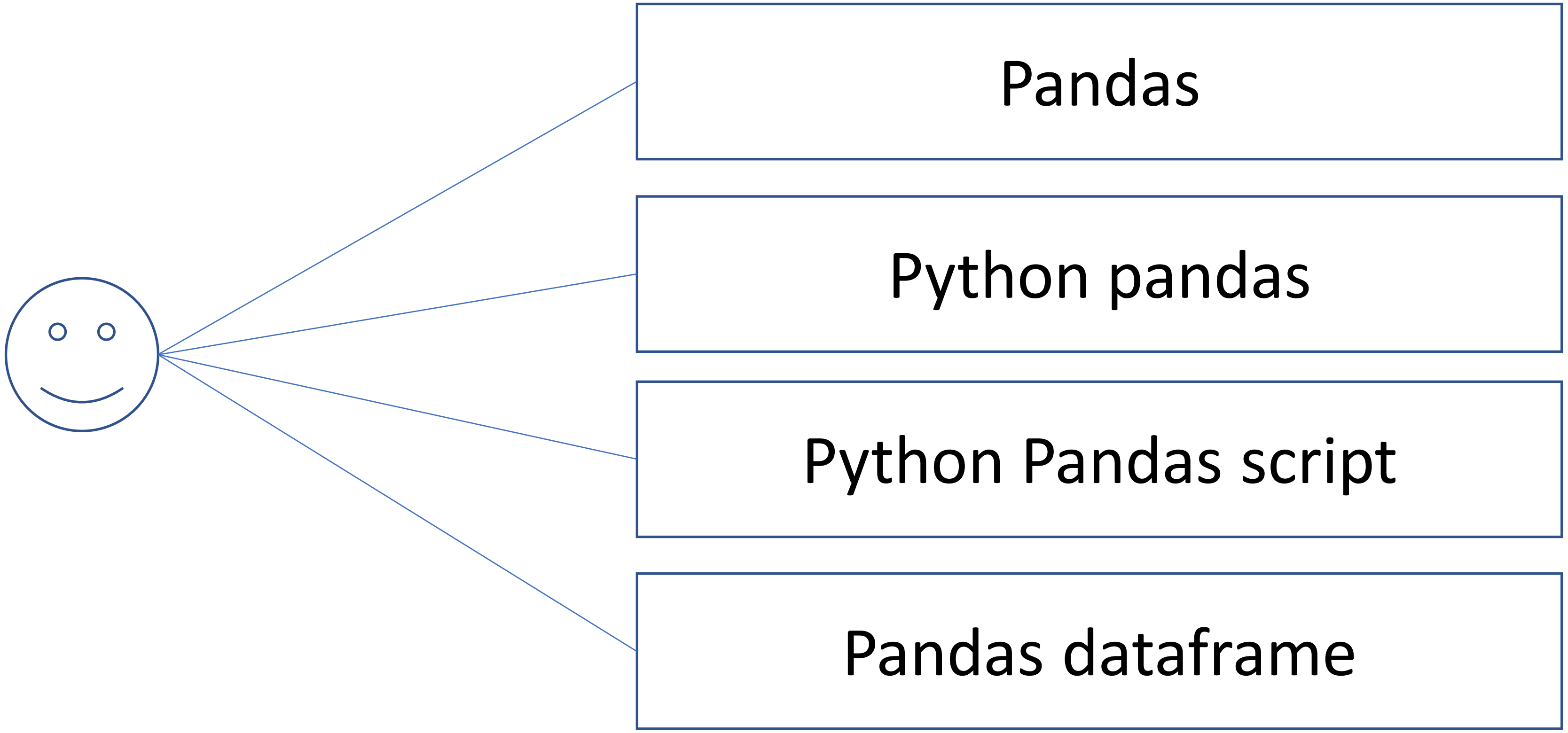
Software names are ambiguous: Homonyms



Why matters? Homonyms → **Merging of software entities**



Software names are ambiguous: Synonyms



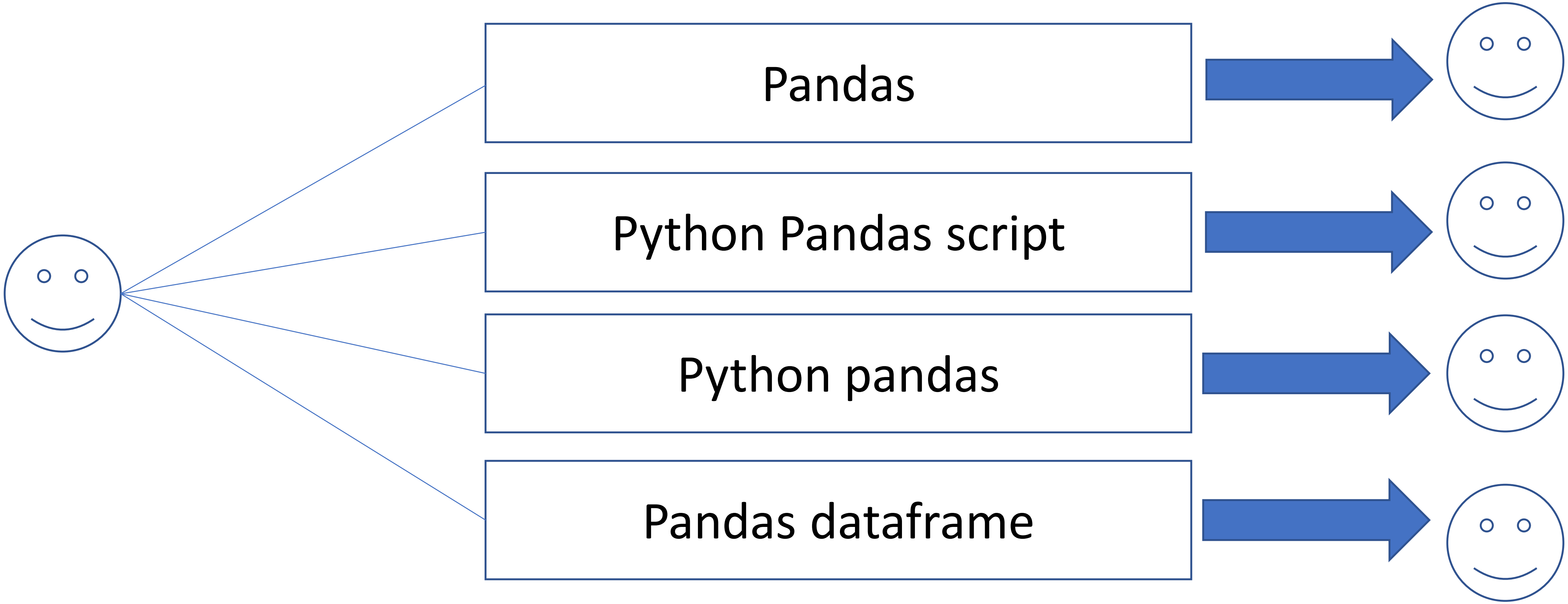
Pandas

Python pandas

Python Pandas script

Pandas dataframe

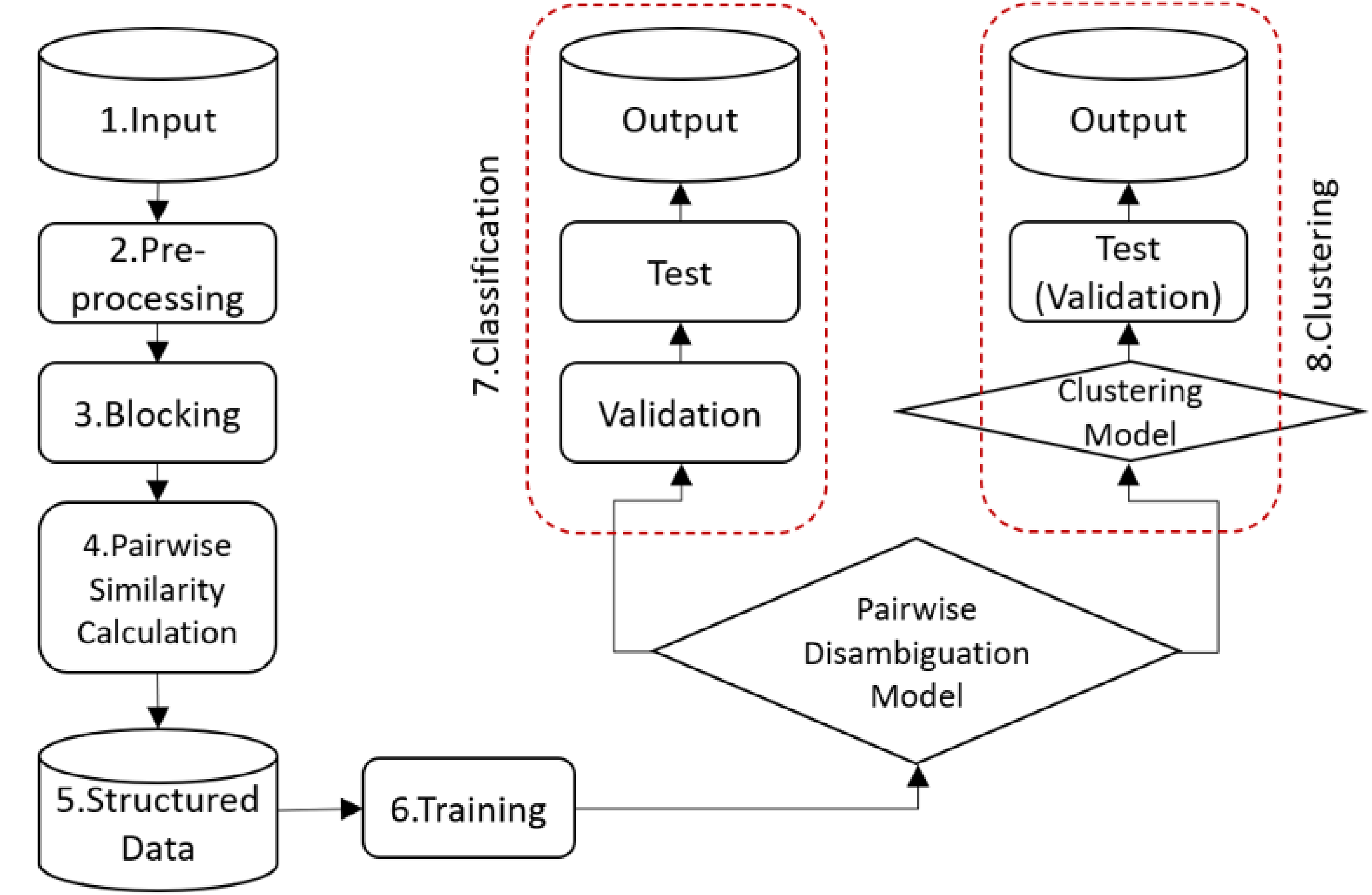
Why matters? Synonyms → **Splitting of software entities**



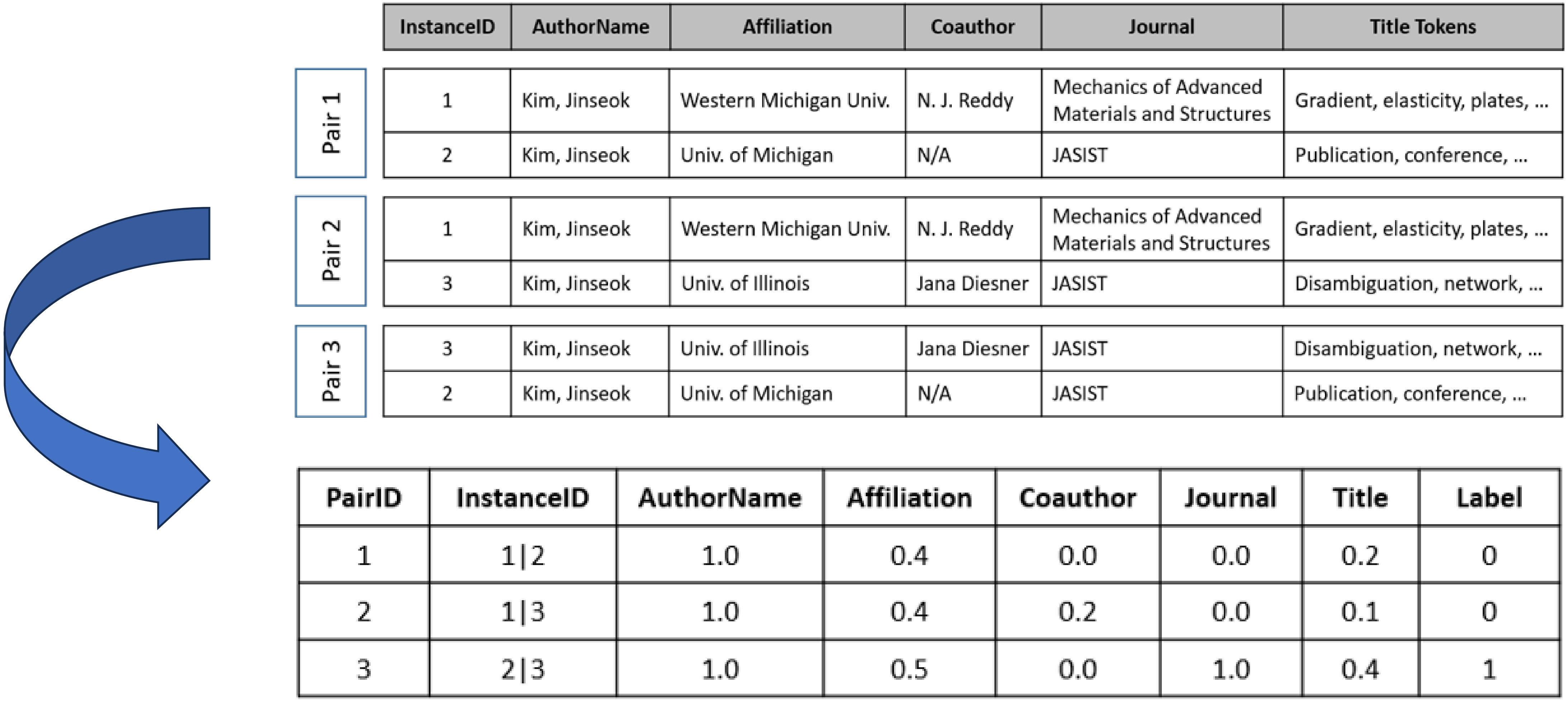
Using ANDez for software mention disambiguation

- An open-source platform that consolidates the processes of multiple high-performing ML techniques for **classification** and **clustering** in author name disambiguation (AND) → Modified for software name disambiguation (SND)
- Created using Python and widely used ML libraries (esp. scikit-learn)
- Offers a **transparent** and **uniform** framework for AND that brings together the execution of several ML methodologies
- GitHub repo: https://github.com/TEEDLab/ANDez
- Codeocean capsule: https://doi.org/10.24433/CO.0959364.v1

Workflow in ANDez



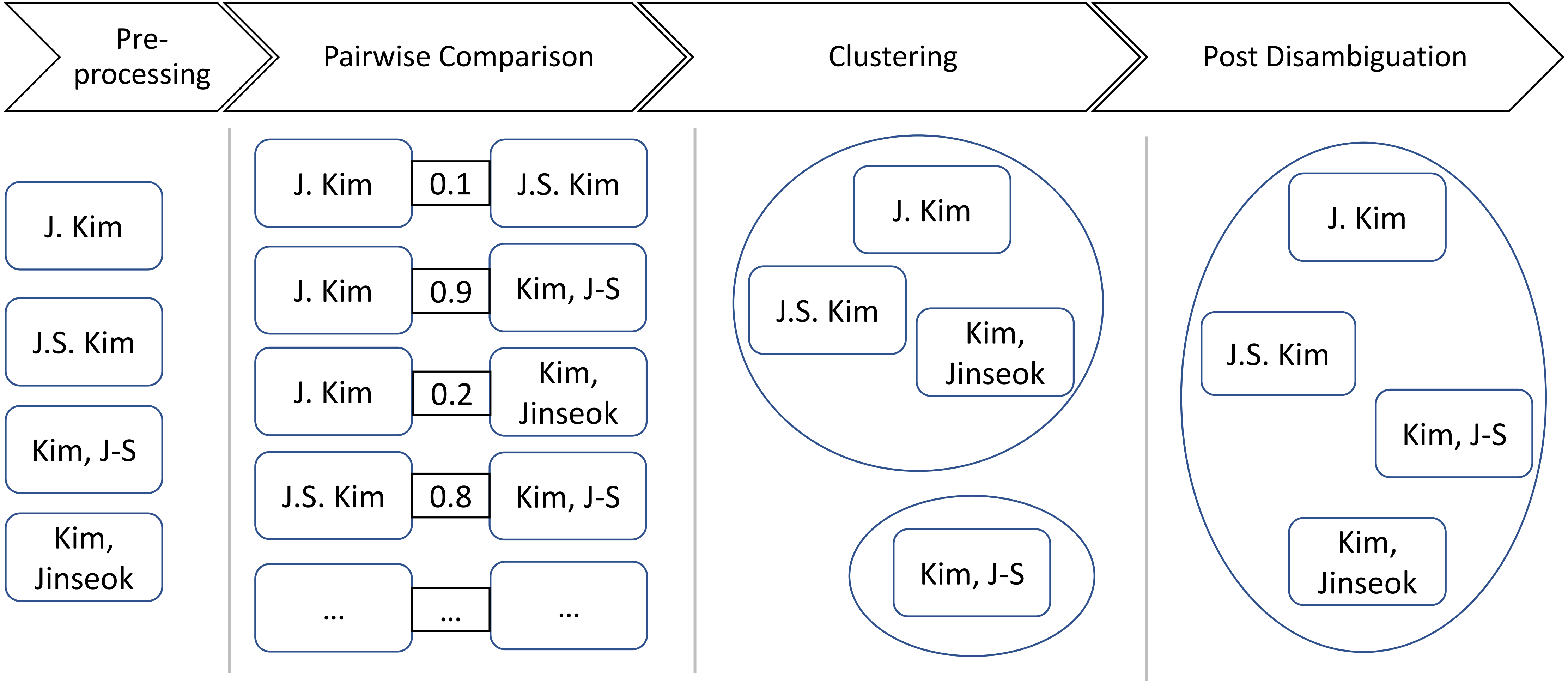
Key of name disambiguation in ANDez: Pairwise similarity comparison



	InstanceID	AuthorName	Affiliation	Coauthor	Journal	Title Tokens
Pair 1	1	Kim, Jinseok	Western Michigan Univ.	N. J. Reddy	Mechanics of Advanced Materials and Structures	Gradient, elasticity, plates, …
	2	Kim, Jinseok	Univ. of Michigan	N/A	JASIST	Publication, conference, …
Pair 2	1	Kim, Jinseok	Western Michigan Univ.	N. J. Reddy	Mechanics of Advanced Materials and Structures	Gradient, elasticity, plates, …
	3	Kim, Jinseok	Univ. of Illinois	Jana Diesner	JASIST	Disambiguation, network, …
Pair 3	3	Kim, Jinseok	Univ. of Illinois	Jana Diesner	JASIST	Disambiguation, network, …
	2	Kim, Jinseok	Univ. of Michigan	N/A	JASIST	Publication, conference, …

PairID	InstanceID	AuthorName	Affiliation	Coauthor	Journal	Title	Label
1	1|2	1.0	0.4	0.0	0.0	0.2	0
2	1|3	1.0	0.4	0.2	0.0	0.1	0
3	2|3	1.0	0.5	0.0	1.0	0.4	1

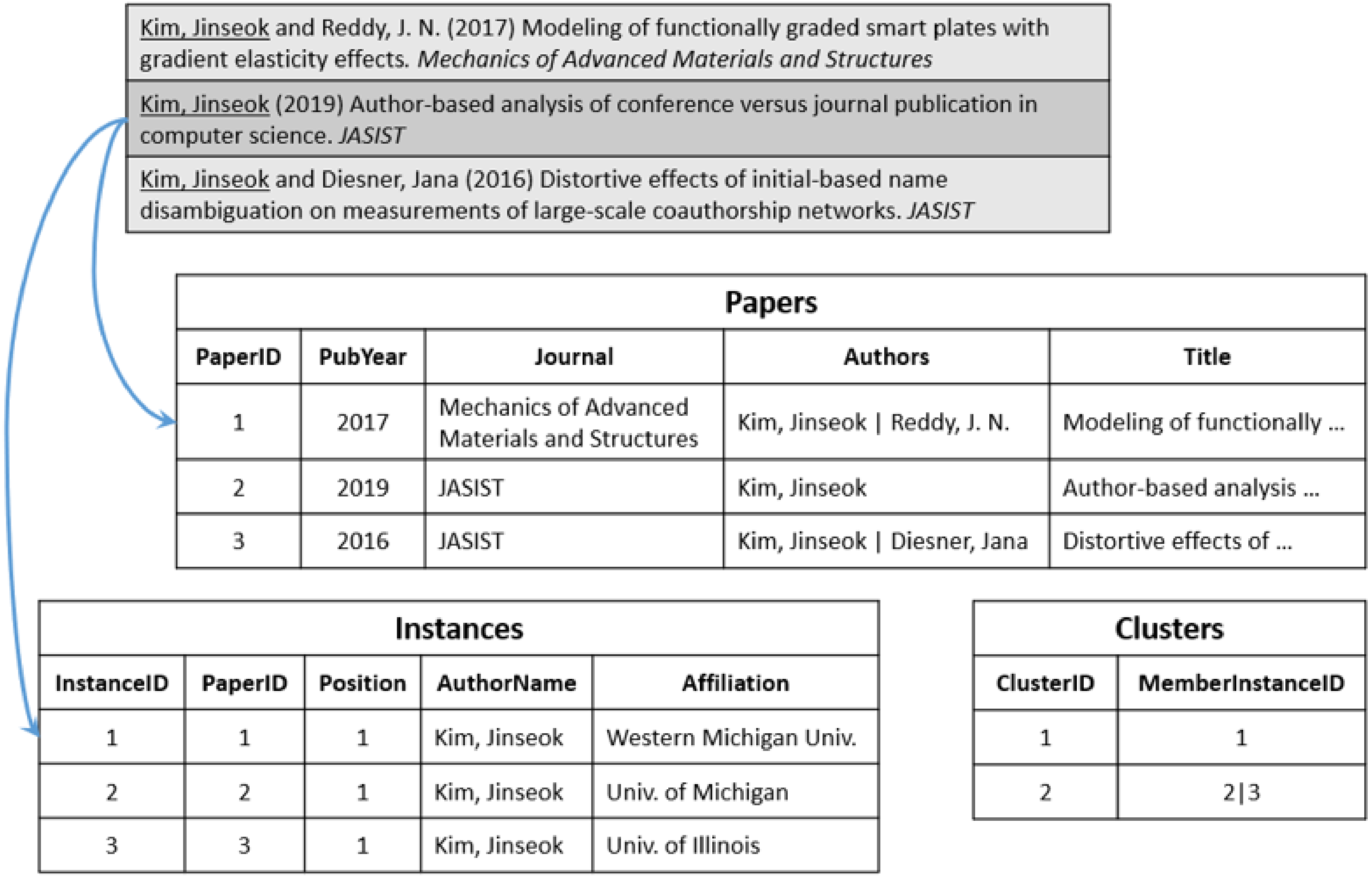
Workflow of name dismbiguation



Requirements for ANDez implementation for software names

- Raw data should be converted into specific formats
  - **Signature file**: contains information specific to an instance like instance ids, record ids, position of mentions in bylines (not required for SND), name string, auxiliary information such as existence of version mention, and class labels for each instance
  - **Record file**: contains information specific to a record like record ids, record years, record venues (e.g., journal names), list of authors, sentences or paragraphs with software mentions, record keywords, and any other information to be used
  - **Cluster file**: contains information of cluster membership of an instance to be used for training and evaluation of machine learning models

Examples of input files formatted for ANDez using author name disambiguation cases



Kim, Jinseok and Reddy, J. N. (2017) Modeling of functionally graded smart plates with gradient elasticity effects. *Mechanics of Advanced Materials and Structures*

Kim, Jinseok (2019) Author-based analysis of conference versus journal publication in computer science. *JASIST*

Kim, Jinseok and Diesner, Jana (2016) Distortive effects of initial-based name disambiguation on measurements of large-scale coauthorship networks. *JASIST*

Papers

PaperID	PubYear	Journal	Authors	Title
1	2017	Mechanics of Advanced Materials and Structures	Kim, Jinseok | Reddy, J. N.	Modeling of functionally ...
2	2019	JASIST	Kim, Jinseok	Author-based analysis ...
3	2016	JASIST	Kim, Jinseok | Diesner, Jana	Distortive effects of ...

Instances

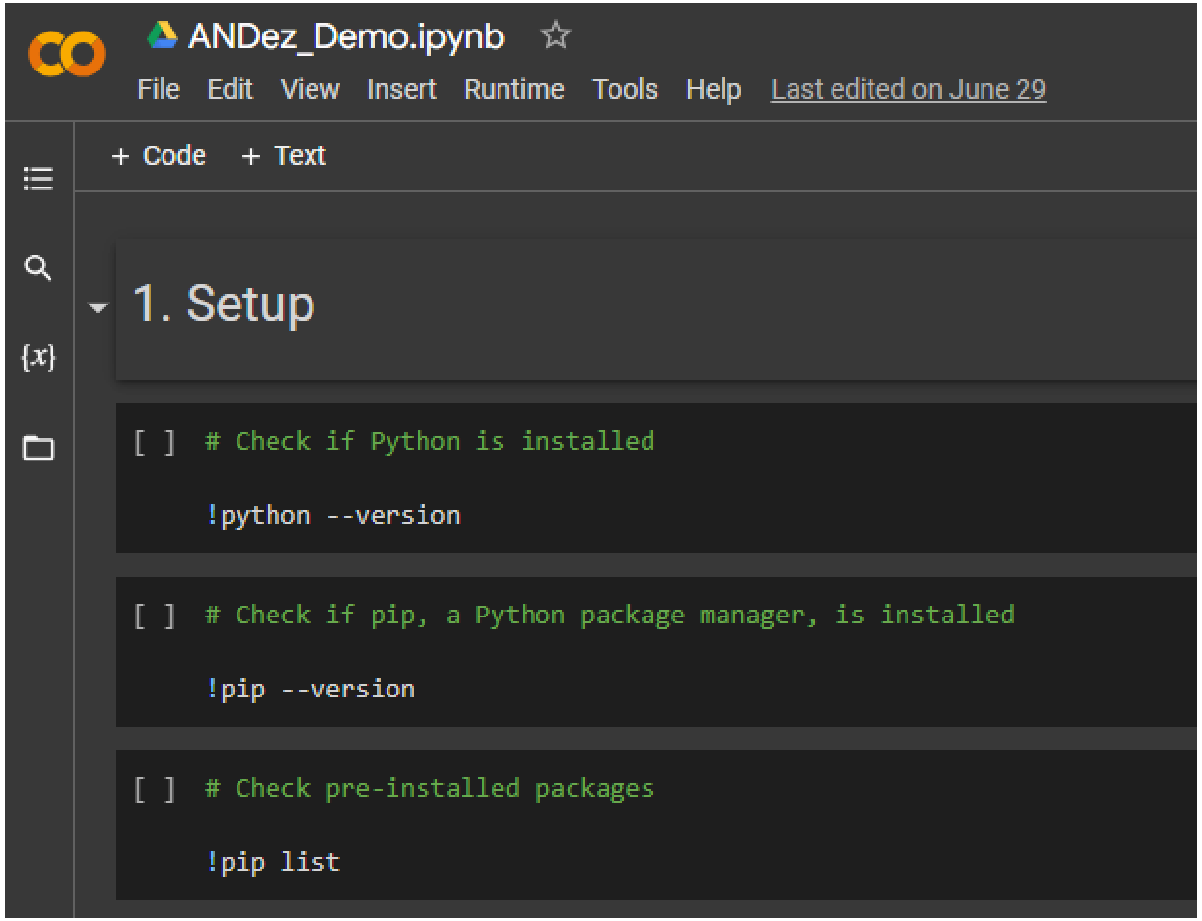
InstanceID	PaperID	Position	AuthorName	Affiliation
1	1	1	Kim, Jinseok	Western Michigan Univ.
2	2	1	Kim, Jinseok	Univ. of Michigan
3	3	1	Kim, Jinseok	Univ. of Illinois

Clusters

ClusterID	MemberInstanceID
1	1
2	2|3

Demo: Setup using Google Colab

NOTE: This demonstration utilizes author name disambiguation files as an example. Once the software name files are formatted appropriately, the same implementation process outlined here will be applied for AND.


ANDez_Demo.ipynb
File Edit View Insert Runtime Tools Help Last edited on June 29
+ Code + Text

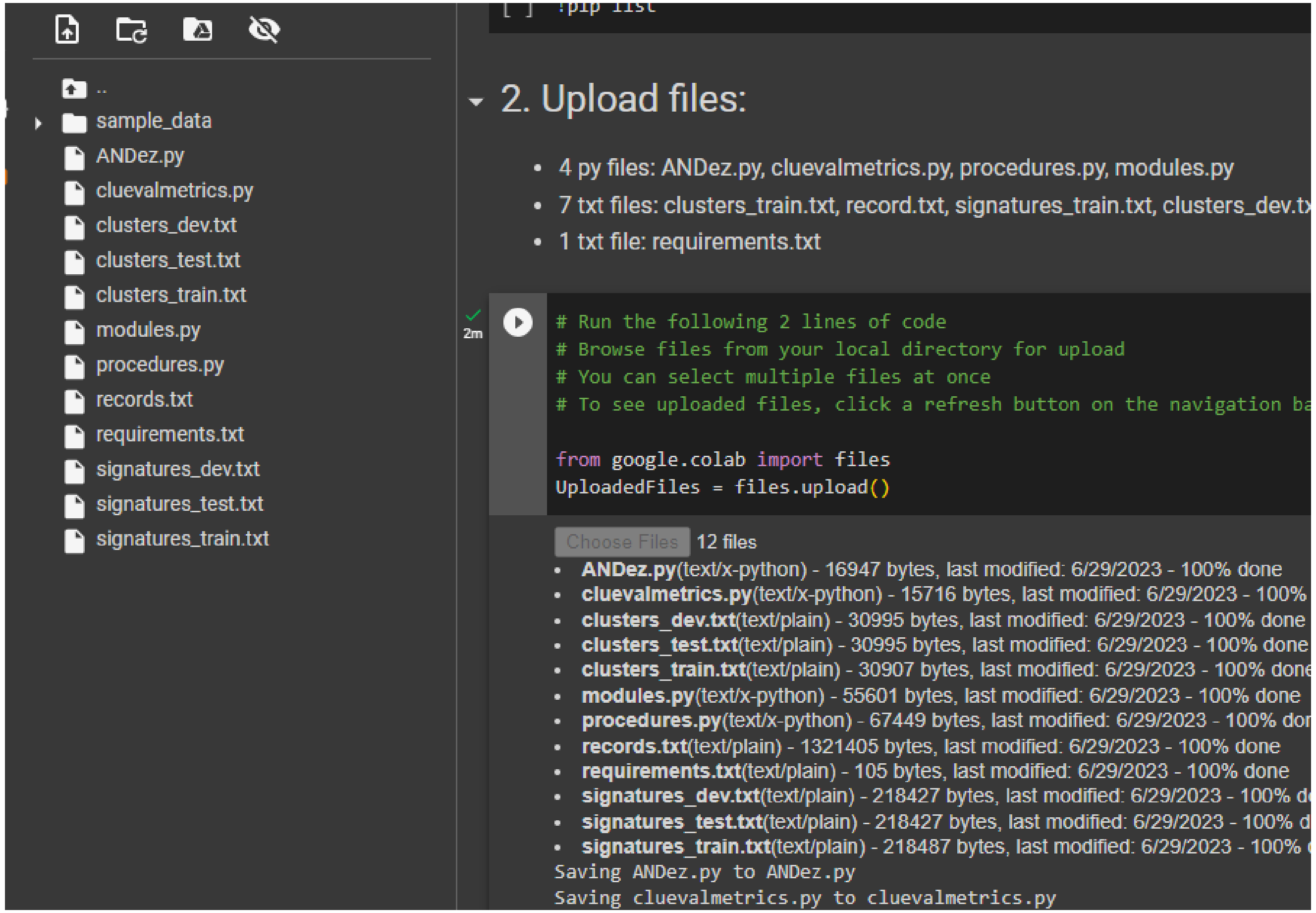
1. Setup

```
[ ]  # Check if Python is installed

     !python --version
```

```
[ ]  # Check if pip, a Python package manager, is installed

     !pip --version
```

```
[ ]  # Check pre-installed packages

     !pip list
```

Upload Files



```
[ ] !pip list
```

2. Upload files:

- 4 py files: ANDez.py, cluevalmetrics.py, procedures.py, modules.py
- 7 txt files: clusters_train.txt, record.txt, signatures_train.txt, clusters_dev.tx
- 1 txt file: requirements.txt

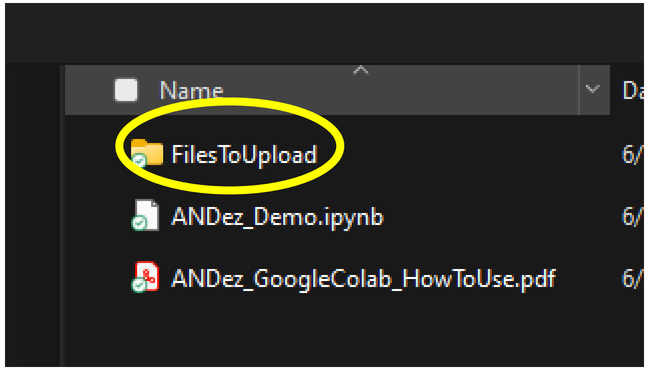
```
# Run the following 2 lines of code
# Browse files from your local directory for upload
# You can select multiple files at once
# To see uploaded files, click a refresh button on the navigation ba

from google.colab import files
UploadedFiles = files.upload()
```

Choose Files 12 files

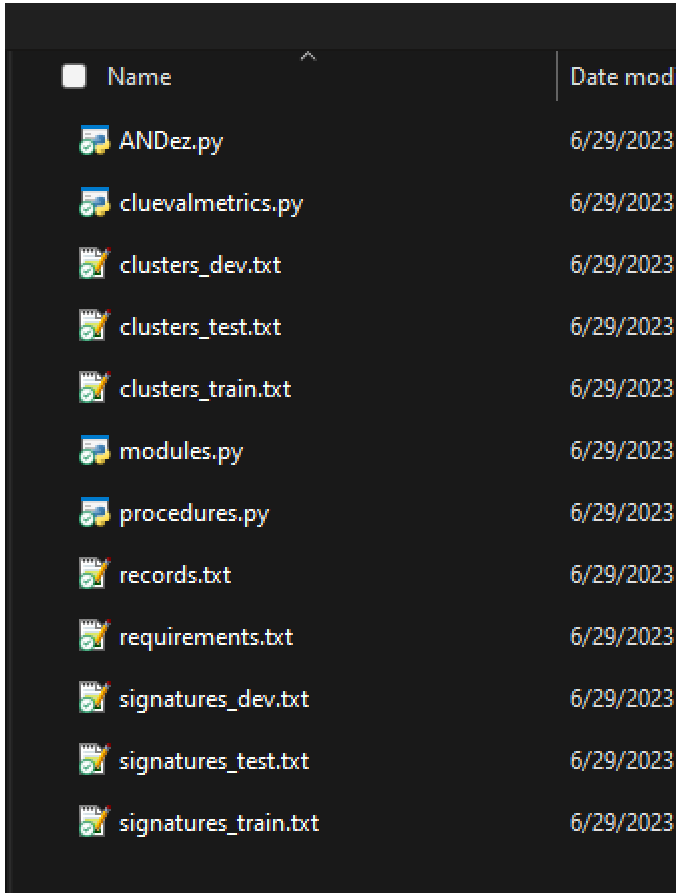
- **ANDez.py**(text/x-python) - 16947 bytes, last modified: 6/29/2023 - 100% done
- **cluevalmetrics.py**(text/x-python) - 15716 bytes, last modified: 6/29/2023 - 100%
- **clusters_dev.txt**(text/plain) - 30995 bytes, last modified: 6/29/2023 - 100% done
- **clusters_test.txt**(text/plain) - 30995 bytes, last modified: 6/29/2023 - 100% done
- **clusters_train.txt**(text/plain) - 30907 bytes, last modified: 6/29/2023 - 100% done
- **modules.py**(text/x-python) - 55601 bytes, last modified: 6/29/2023 - 100% done
- **procedures.py**(text/x-python) - 67449 bytes, last modified: 6/29/2023 - 100% dor
- **records.txt**(text/plain) - 1321405 bytes, last modified: 6/29/2023 - 100% done
- **requirements.txt**(text/plain) - 105 bytes, last modified: 6/29/2023 - 100% done
- **signatures_dev.txt**(text/plain) - 218427 bytes, last modified: 6/29/2023 - 100% d
- **signatures_test.txt**(text/plain) - 218427 bytes, last modified: 6/29/2023 - 100% d
- **signatures_train.txt**(text/plain) - 218487 bytes, last modified: 6/29/2023 - 100%

```
Saving ANDez.py to ANDez.py
Saving cluevalmetrics.py to cluevalmetrics.py
```

Upload all the files downloaded



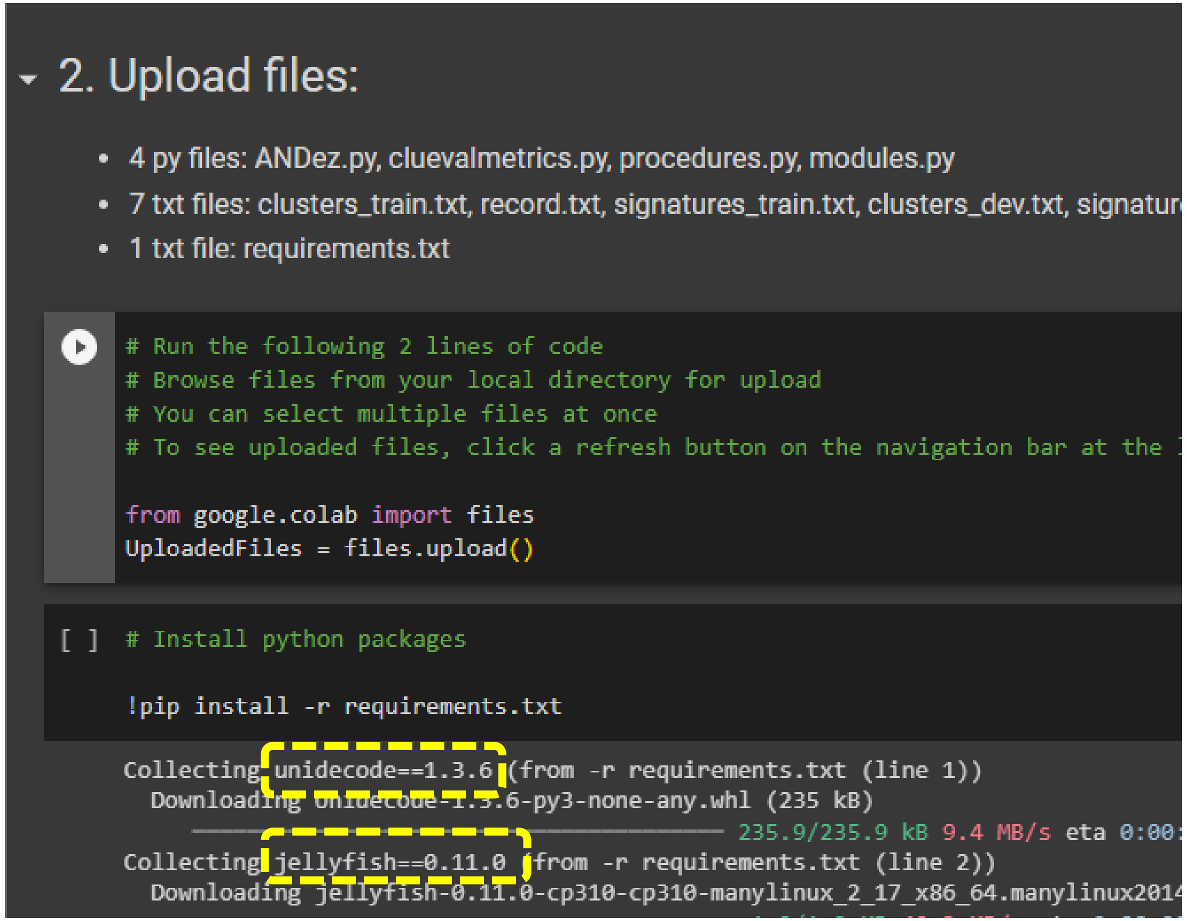
Name	Da
FilesToUpload	6/
ANDez_Demo.ipynb	6/
ANDez_GoogleColab_HowToUse.pdf	6/



Name	Date mod
ANDez.py	6/29/2023
cluevalmetrics.py	6/29/2023
clusters_dev.txt	6/29/2023
clusters_test.txt	6/29/2023
clusters_train.txt	6/29/2023
modules.py	6/29/2023
procedures.py	6/29/2023
records.txt	6/29/2023
requirements.txt	6/29/2023
signatures_dev.txt	6/29/2023
signatures_test.txt	6/29/2023
signatures_train.txt	6/29/2023

Install extra packages

on a local machine, more packages might need to be installed



2. Upload files:

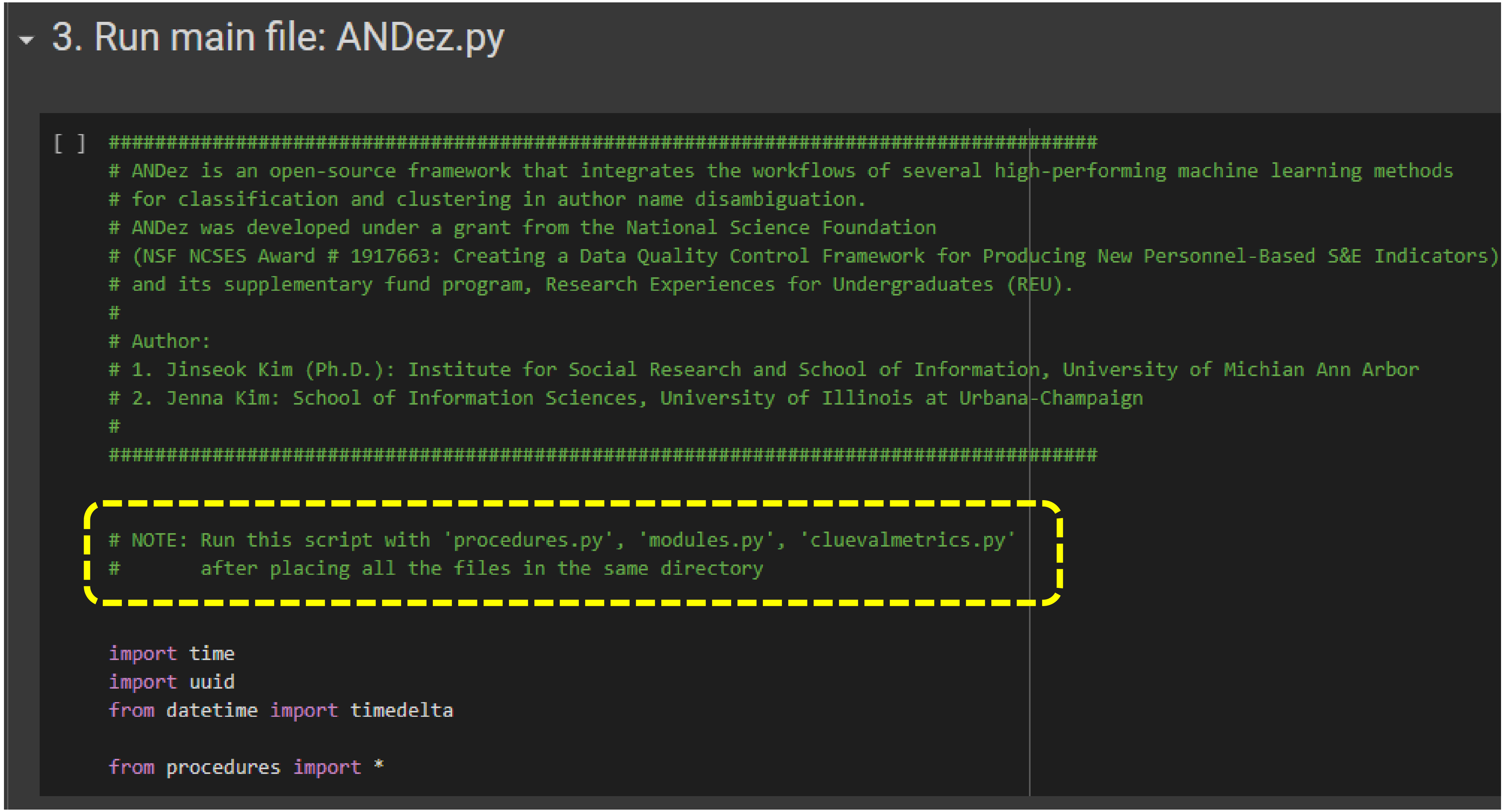
- 4 py files: ANDez.py, cluevalmetrics.py, procedures.py, modules.py
- 7 txt files: clusters_train.txt, record.txt, signatures_train.txt, clusters_dev.txt, signatur
- 1 txt file: requirements.txt

```
# Run the following 2 lines of code
# Browse files from your local directory for upload
# You can select multiple files at once
# To see uploaded files, click a refresh button on the navigation bar at the 

from google.colab import files
UploadedFiles = files.upload()
```

```
[ ]  # Install python packages

     !pip install -r requirements.txt
```

```
Collecting unidecode==1.3.6 (from -r requirements.txt (line 1))
  Downloading unidecode-1.3.6-py3-none-any.whl (235 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 235.9/235.9 kB 9.4 MB/s eta 0:00:
Collecting jellyfish==0.11.0 (from -r requirements.txt (line 2))
  Downloading jellyfish-0.11.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014
```

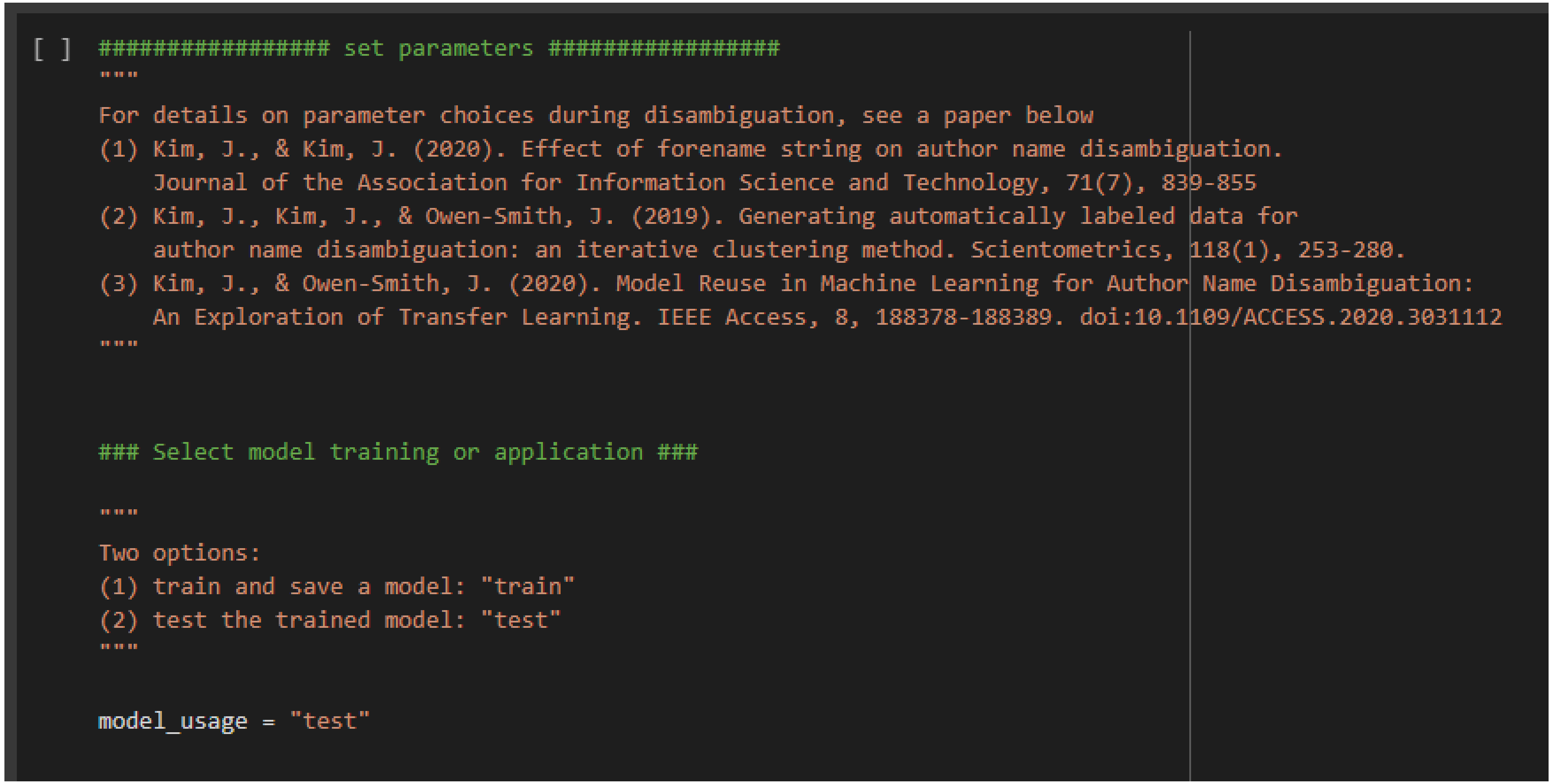
Before you run main file (ANDez.py), make sure that …



3. Run main file: ANDez.py

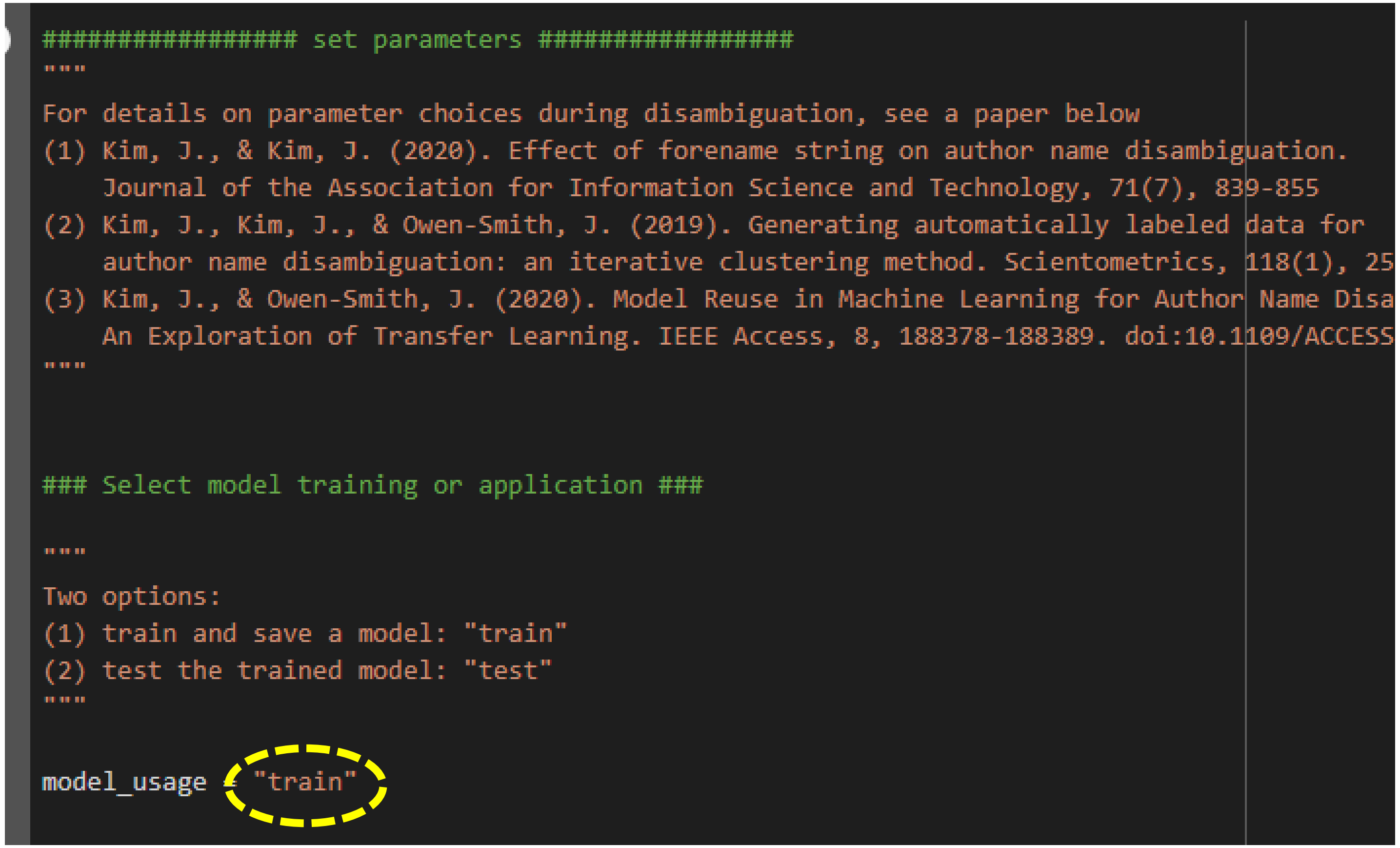
```
######################################################################################
# ANDez is an open-source framework that integrates the workflows of several high-performing machine learning methods
# for classification and clustering in author name disambiguation.
# ANDez was developed under a grant from the National Science Foundation
# (NSF NCSES Award # 1917663: Creating a Data Quality Control Framework for Producing New Personnel-Based S&E Indicators)
# and its supplementary fund program, Research Experiences for Undergraduates (REU).
#
# Author:
# 1. Jinseok Kim (Ph.D.): Institute for Social Research and School of Information, University of Michian Ann Arbor
# 2. Jenna Kim: School of Information Sciences, University of Illinois at Urbana-Champaign
#
######################################################################################

# NOTE: Run this script with 'procedures.py', 'modules.py', 'cluevalmetrics.py'
#       after placing all the files in the same directory

import time
import uuid
from datetime import timedelta

from procedures import *
```

Set parameters (= options): The first choice you need to make…



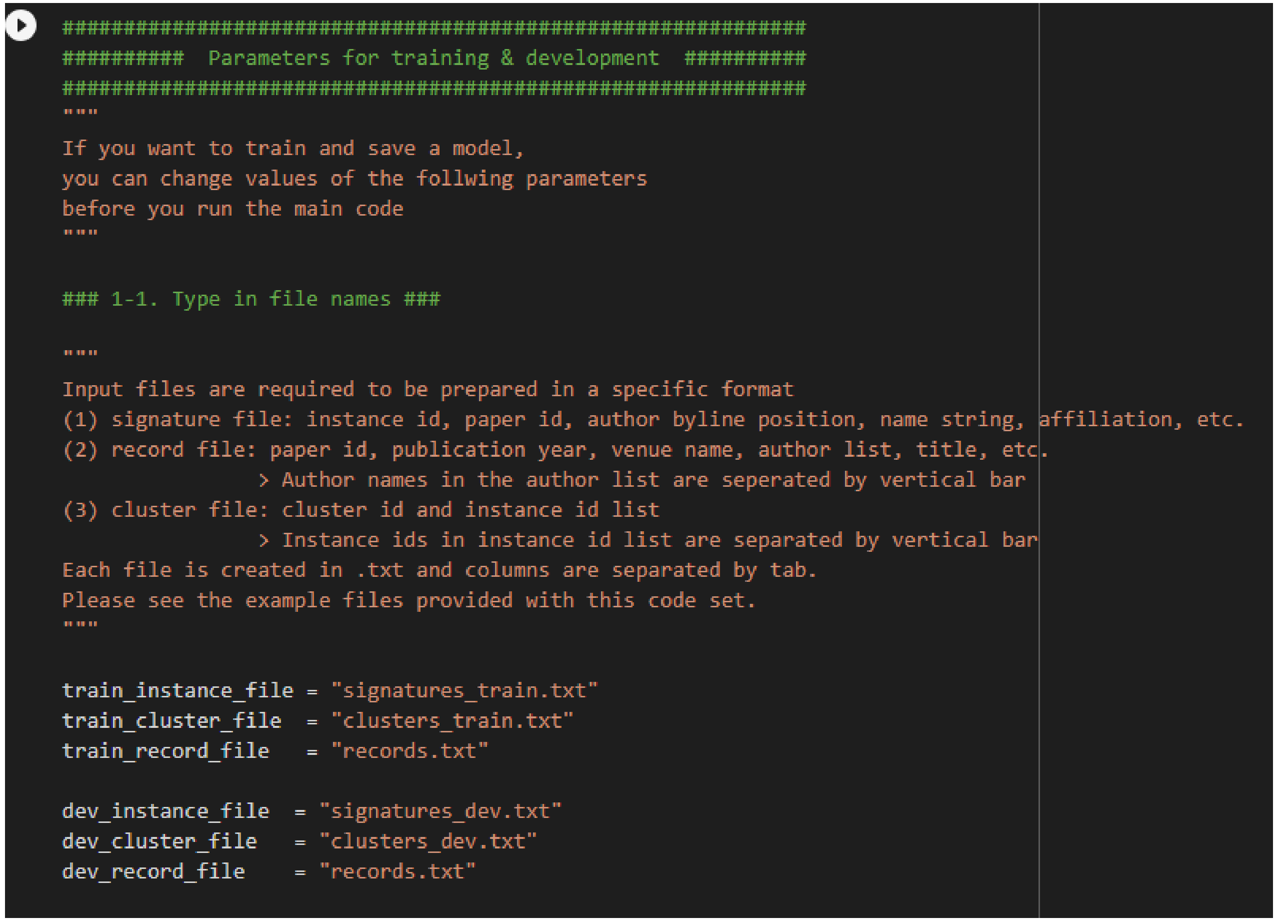
```
################# set parameters #################
"""
For details on parameter choices during disambiguation, see a paper below
(1) Kim, J., & Kim, J. (2020). Effect of forename string on author name disambiguation.
    Journal of the Association for Information Science and Technology, 71(7), 839-855
(2) Kim, J., Kim, J., & Owen-Smith, J. (2019). Generating automatically labeled data for
    author name disambiguation: an iterative clustering method. Scientometrics, 118(1), 253-280.
(3) Kim, J., & Owen-Smith, J. (2020). Model Reuse in Machine Learning for Author Name Disambiguation:
    An Exploration of Transfer Learning. IEEE Access, 8, 188378-188389. doi:10.1109/ACCESS.2020.3031112
"""



### Select model training or application ###

"""
Two options:
(1) train and save a model: "train"
(2) test the trained model: "test"
"""

model_usage = "test"
```

Set parameters (= options): The first choice you need to make…



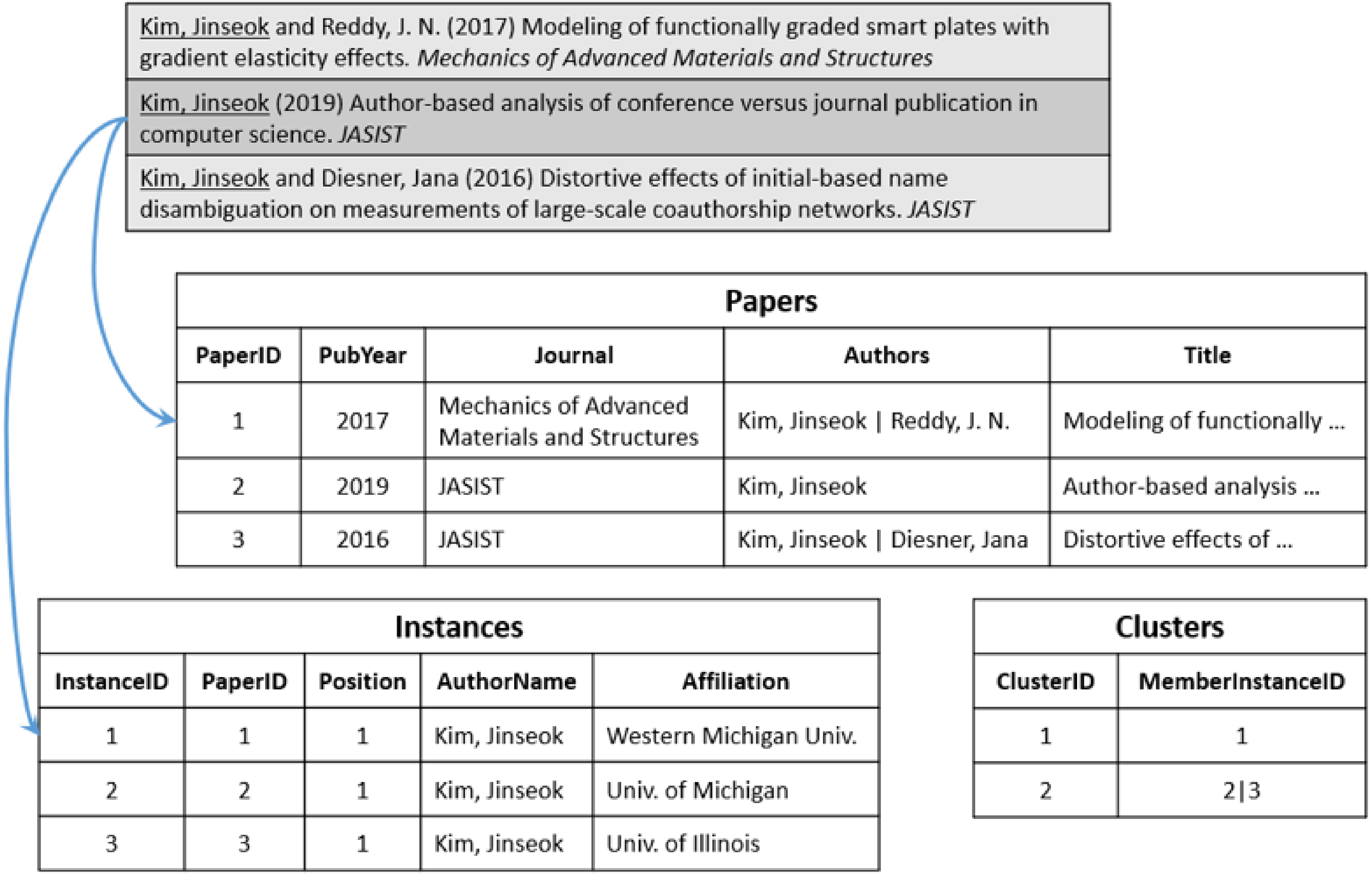
```
################# set parameters #################
"""
For details on parameter choices during disambiguation, see a paper below
(1) Kim, J., & Kim, J. (2020). Effect of forename string on author name disambiguation.
    Journal of the Association for Information Science and Technology, 71(7), 839-855
(2) Kim, J., Kim, J., & Owen-Smith, J. (2019). Generating automatically labeled data for
    author name disambiguation: an iterative clustering method. Scientometrics, 118(1), 25
(3) Kim, J., & Owen-Smith, J. (2020). Model Reuse in Machine Learning for Author Name Disa
    An Exploration of Transfer Learning. IEEE Access, 8, 188378-188389. doi:10.1109/ACCESS
"""



### Select model training or application ###

"""
Two options:
(1) train and save a model: "train"
(2) test the trained model: "test"
"""

model_usage = "train"
```

Type input file names



```
#############################################################
##########  Parameters for training & development  ##########
#############################################################
"""
If you want to train and save a model,
you can change values of the follwing parameters
before you run the main code
"""

### 1-1. Type in file names ###

"""
Input files are required to be prepared in a specific format
(1) signature file: instance id, paper id, author byline position, name string, affiliation, etc.
(2) record file: paper id, publication year, venue name, author list, title, etc.
                > Author names in the author list are seperated by vertical bar
(3) cluster file: cluster id and instance id list
                > Instance ids in instance id list are separated by vertical bar
Each file is created in .txt and columns are separated by tab.
Please see the example files provided with this code set.
"""

train_instance_file = "signatures_train.txt"
train_cluster_file  = "clusters_train.txt"
train_record_file   = "records.txt"

dev_instance_file  = "signatures_dev.txt"
dev_cluster_file   = "clusters_dev.txt"
dev_record_file    = "records.txt"
```

Input file format



Kim, Jinseok and Reddy, J. N. (2017) Modeling of functionally graded smart plates with gradient elasticity effects. *Mechanics of Advanced Materials and Structures*

Kim, Jinseok (2019) Author-based analysis of conference versus journal publication in computer science. *JASIST*

Kim, Jinseok and Diesner, Jana (2016) Distortive effects of initial-based name disambiguation on measurements of large-scale coauthorship networks. *JASIST*

Papers

PaperID	PubYear	Journal	Authors	Title
1	2017	Mechanics of Advanced Materials and Structures	Kim, Jinseok | Reddy, J. N.	Modeling of functionally …
2	2019	JASIST	Kim, Jinseok	Author-based analysis …
3	2016	JASIST	Kim, Jinseok | Diesner, Jana	Distortive effects of …

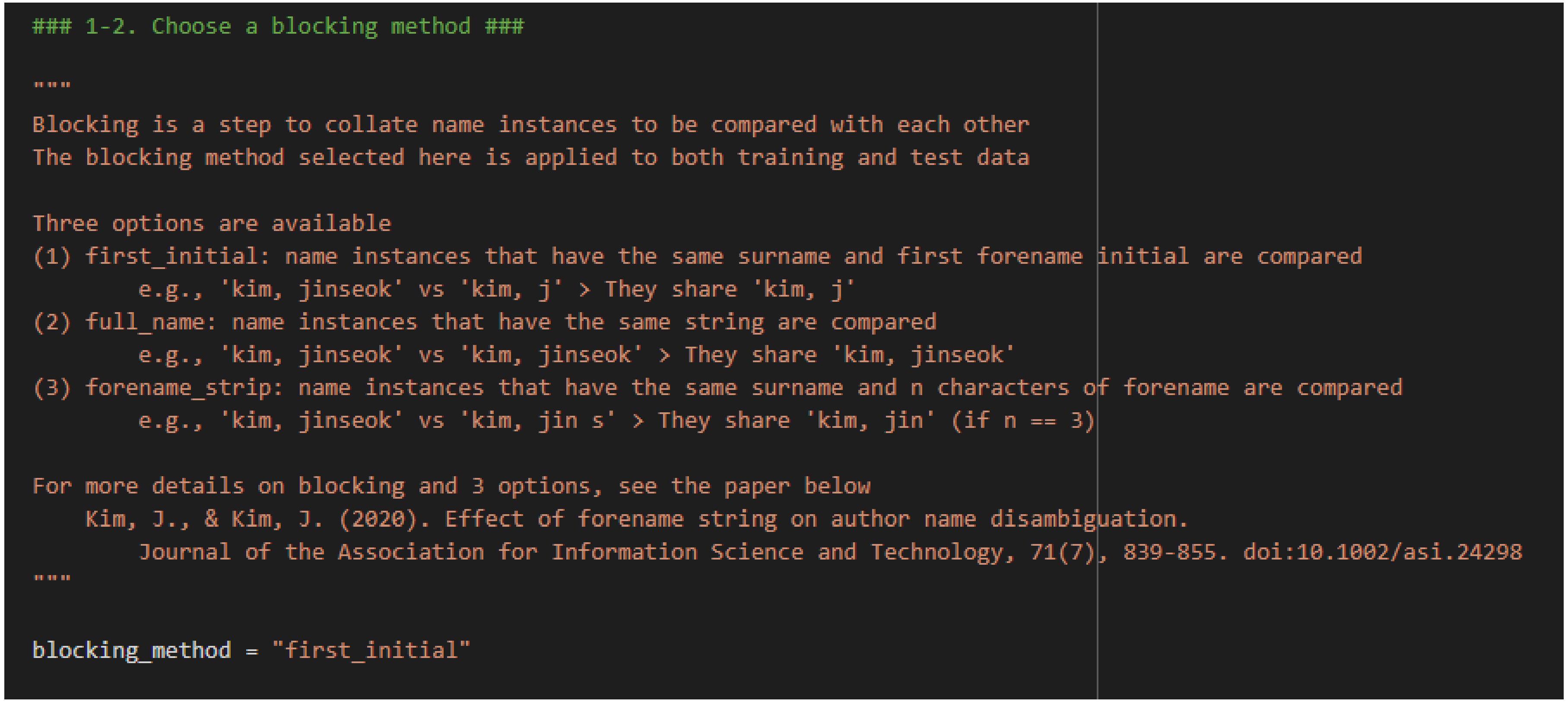
Instances

InstanceID	PaperID	Position	AuthorName	Affiliation
1	1	1	Kim, Jinseok	Western Michigan Univ.
2	2	1	Kim, Jinseok	Univ. of Michigan
3	3	1	Kim, Jinseok	Univ. of Illinois

Clusters

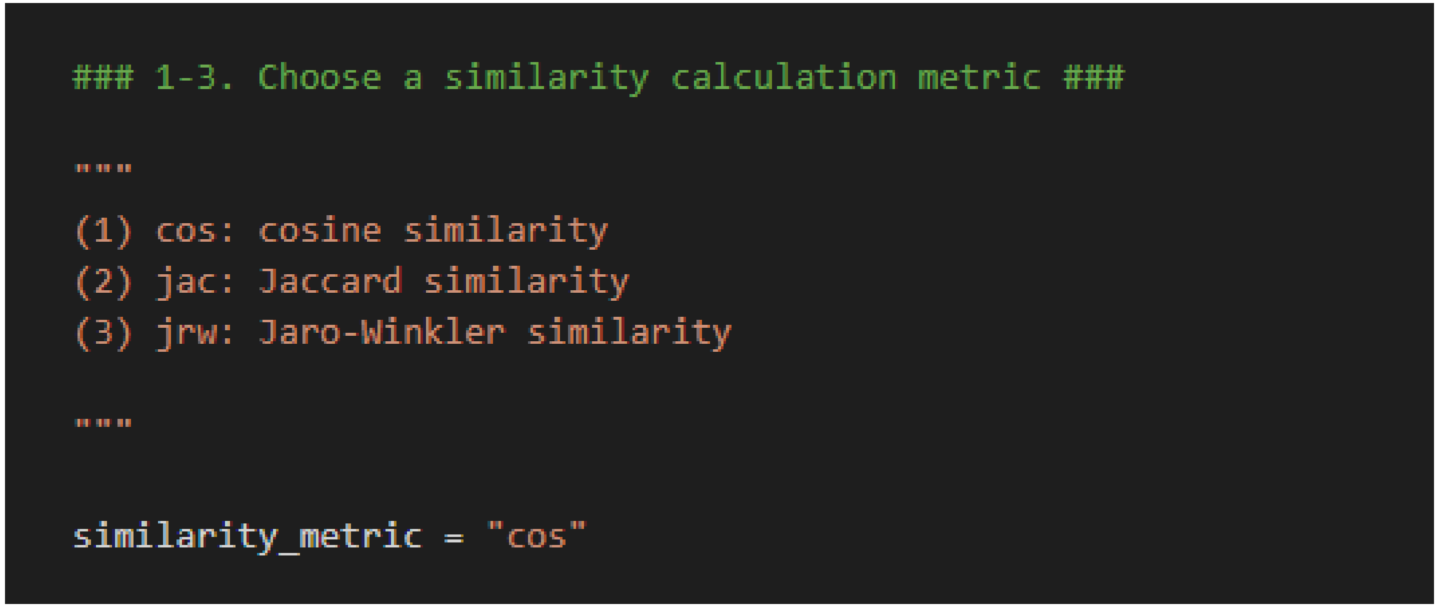
ClusterID	MemberInstanceID
1	1
2	2|3

Blocking



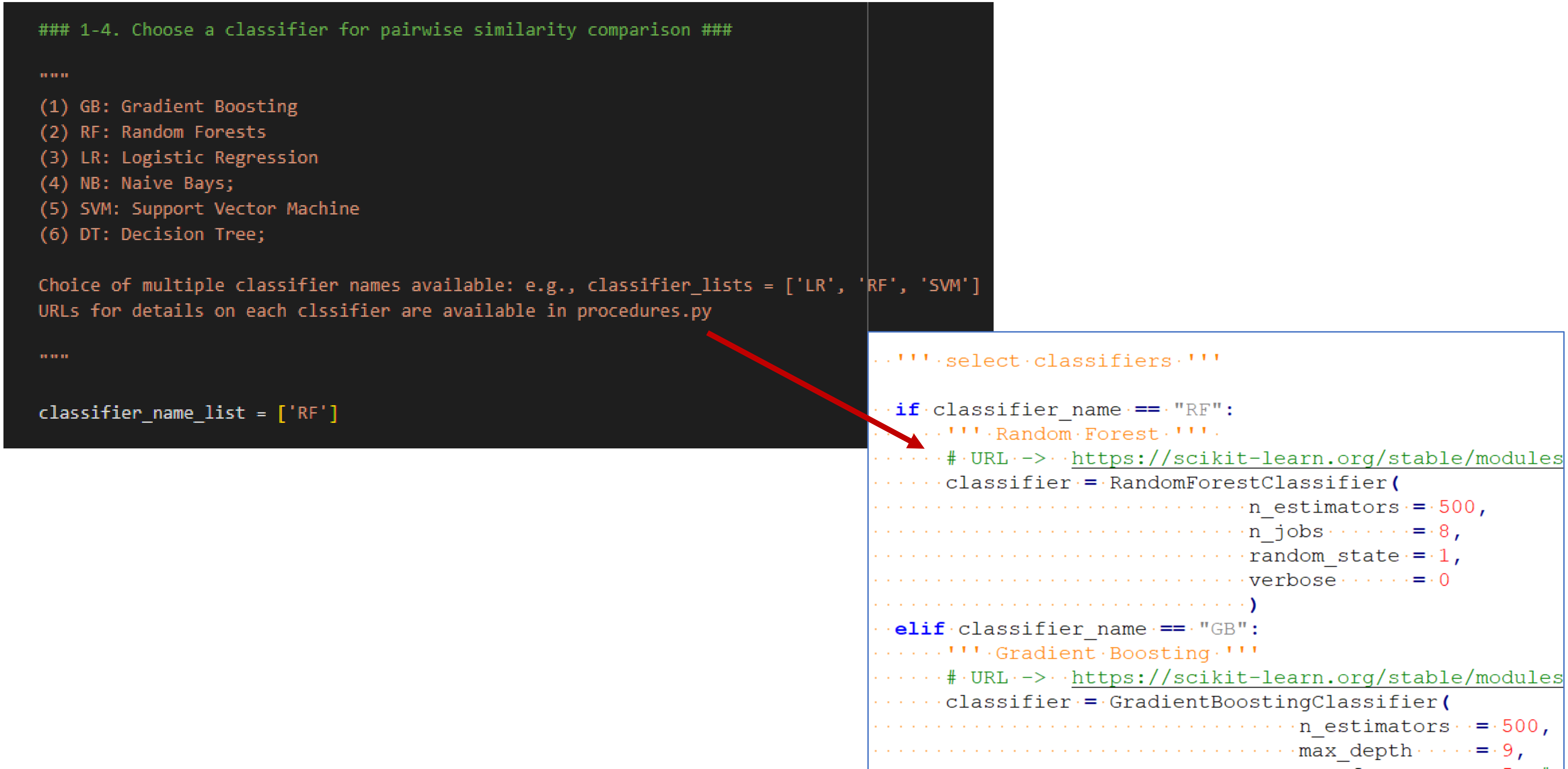
```
### 1-2. Choose a blocking method ###

"""
Blocking is a step to collate name instances to be compared with each other
The blocking method selected here is applied to both training and test data

Three options are available
(1) first_initial: name instances that have the same surname and first forename initial are compared
        e.g., 'kim, jinseok' vs 'kim, j' > They share 'kim, j'
(2) full_name: name instances that have the same string are compared
        e.g., 'kim, jinseok' vs 'kim, jinseok' > They share 'kim, jinseok'
(3) forename_strip: name instances that have the same surname and n characters of forename are compared
        e.g., 'kim, jinseok' vs 'kim, jin s' > They share 'kim, jin' (if n == 3)

For more details on blocking and 3 options, see the paper below
    Kim, J., & Kim, J. (2020). Effect of forename string on author name disambiguation.
        Journal of the Association for Information Science and Technology, 71(7), 839-855. doi:10.1002/asi.24298
"""

blocking_method = "first_initial"
```

Similarity calculation metric



```
### 1-3. Choose a similarity calculation metric ###

"""
(1) cos: cosine similarity
(2) jac: Jaccard similarity
(3) jrw: Jaro-Winkler similarity

"""


similarity_metric = "cos"
```

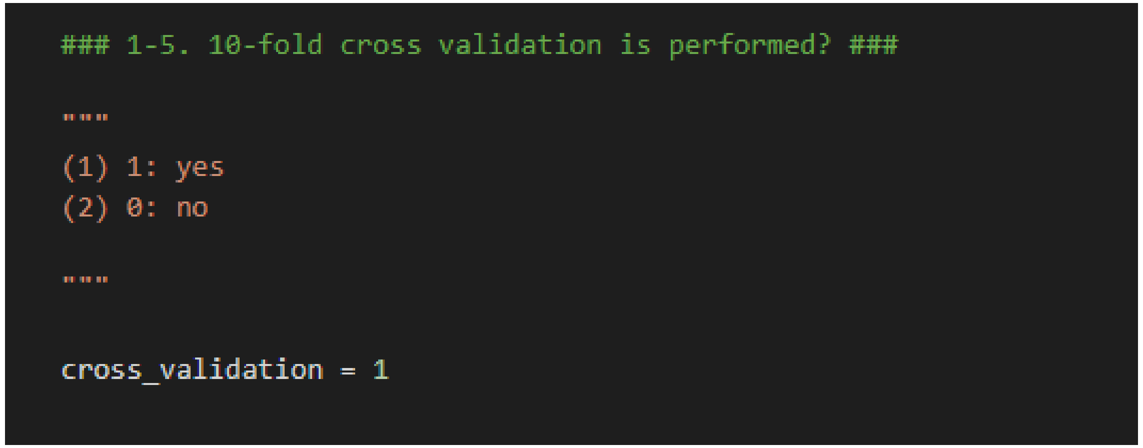
Machine learning algorithms



```
### 1-4. Choose a classifier for pairwise similarity comparison ###

"""
(1) GB: Gradient Boosting
(2) RF: Random Forests
(3) LR: Logistic Regression
(4) NB: Naive Bays;
(5) SVM: Support Vector Machine
(6) DT: Decision Tree;

Choice of multiple classifier names available: e.g., classifier_lists = ['LR', 'RF', 'SVM']
URLs for details on each clssifier are available in procedures.py

"""

classifier_name_list = ['RF']
```

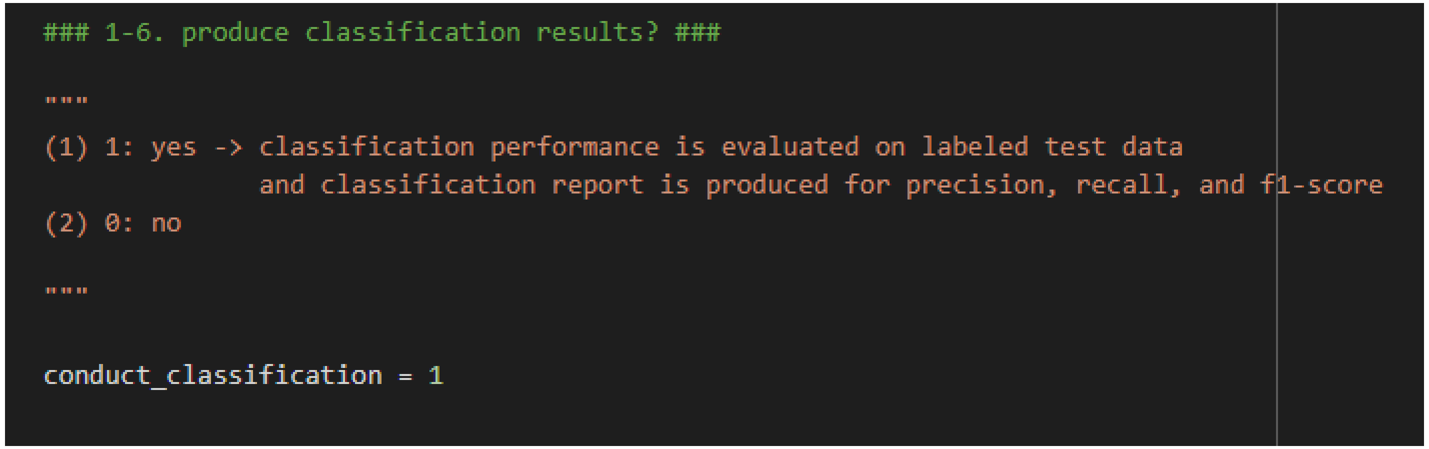
```
  ''' select classifiers '''

 if classifier_name == "RF":
     ''' Random Forest '''
     # URL ->  https://scikit-learn.org/stable/modules
     classifier = RandomForestClassifier(
                              n_estimators = 500,
                              n_jobs       = 8,
                              random_state = 1,
                              verbose      = 0
                              )
 elif classifier_name == "GB":
     ''' Gradient Boosting '''
     # URL ->  https://scikit-learn.org/stable/modules
     classifier = GradientBoostingClassifier(
                                  n_estimators  = 500,
                                  max_depth     = 9,
```

10-fold cross validation



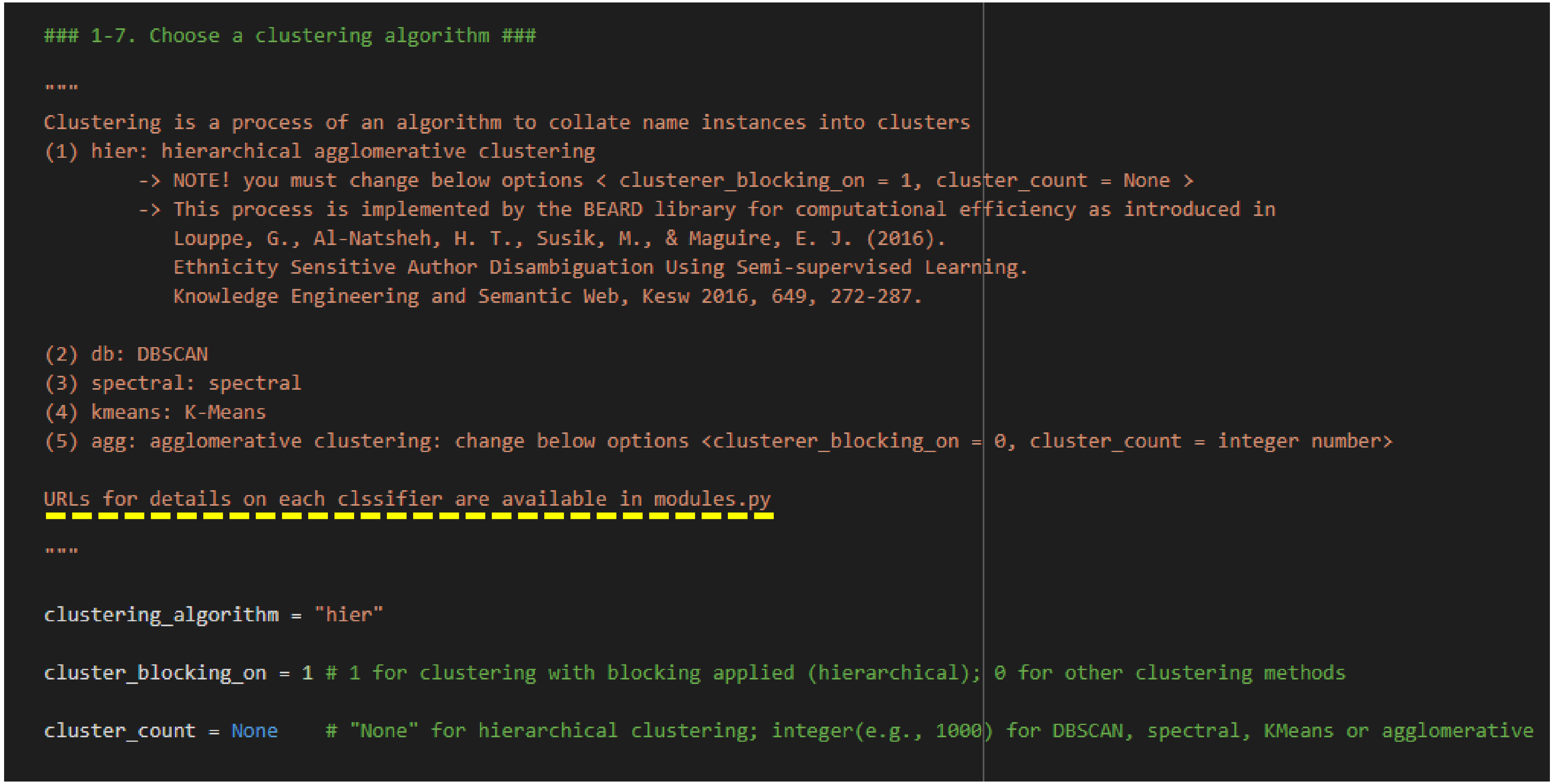
```
### 1-5. 10-fold cross validation is performed? ###

"""
(1) 1: yes
(2) 0: no
"""

cross_validation = 1
```

Producing classification results



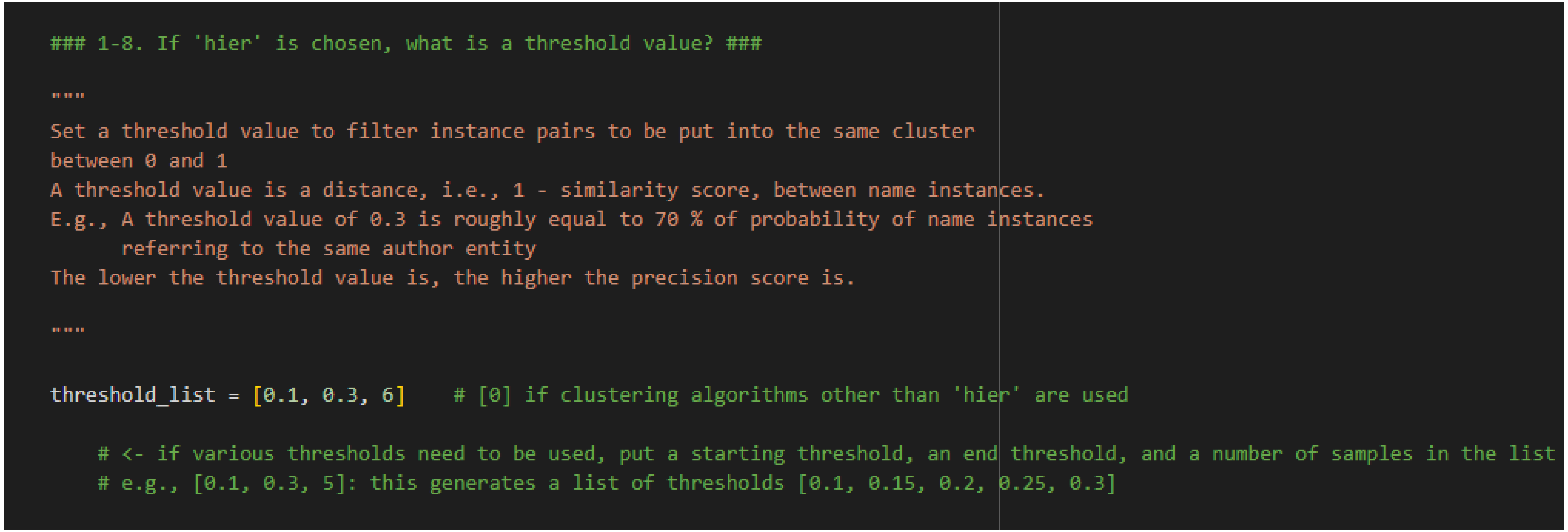
```
### 1-6. produce classification results? ###

"""
(1) 1: yes -> classification performance is evaluated on labeled test data
              and classification report is produced for precision, recall, and f1-score
(2) 0: no

"""

conduct_classification = 1
```

Choosing clustering algorithms



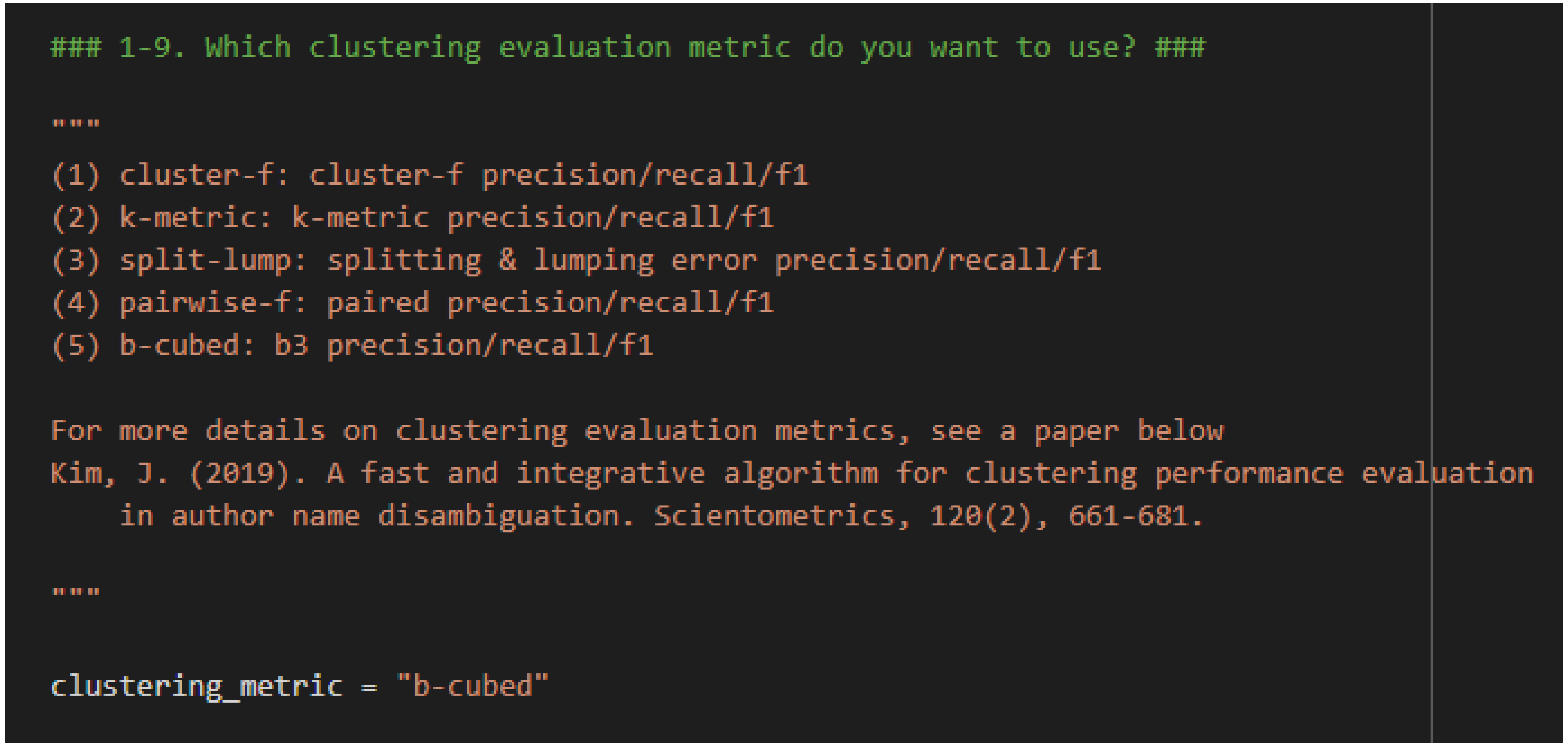
```
### 1-7. Choose a clustering algorithm ###

"""
Clustering is a process of an algorithm to collate name instances into clusters
(1) hier: hierarchical agglomerative clustering
        -> NOTE! you must change below options < clusterer_blocking_on = 1, cluster_count = None >
        -> This process is implemented by the BEARD library for computational efficiency as introduced in
           Louppe, G., Al-Natsheh, H. T., Susik, M., & Maguire, E. J. (2016).
           Ethnicity Sensitive Author Disambiguation Using Semi-supervised Learning.
           Knowledge Engineering and Semantic Web, Kesw 2016, 649, 272-287.

(2) db: DBSCAN
(3) spectral: spectral
(4) kmeans: K-Means
(5) agg: agglomerative clustering: change below options <clusterer_blocking_on = 0, cluster_count = integer number>

URLs for details on each clssifier are available in modules.py

"""

clustering_algorithm = "hier"

cluster_blocking_on = 1 # 1 for clustering with blocking applied (hierarchical); 0 for other clustering methods

cluster_count = None    # "None" for hierarchical clustering; integer(e.g., 1000) for DBSCAN, spectral, KMeans or agglomerative
```

A special parameter for ‘hier’
(Hierarchical Agglomerative Clustering – HAC- algorithm)



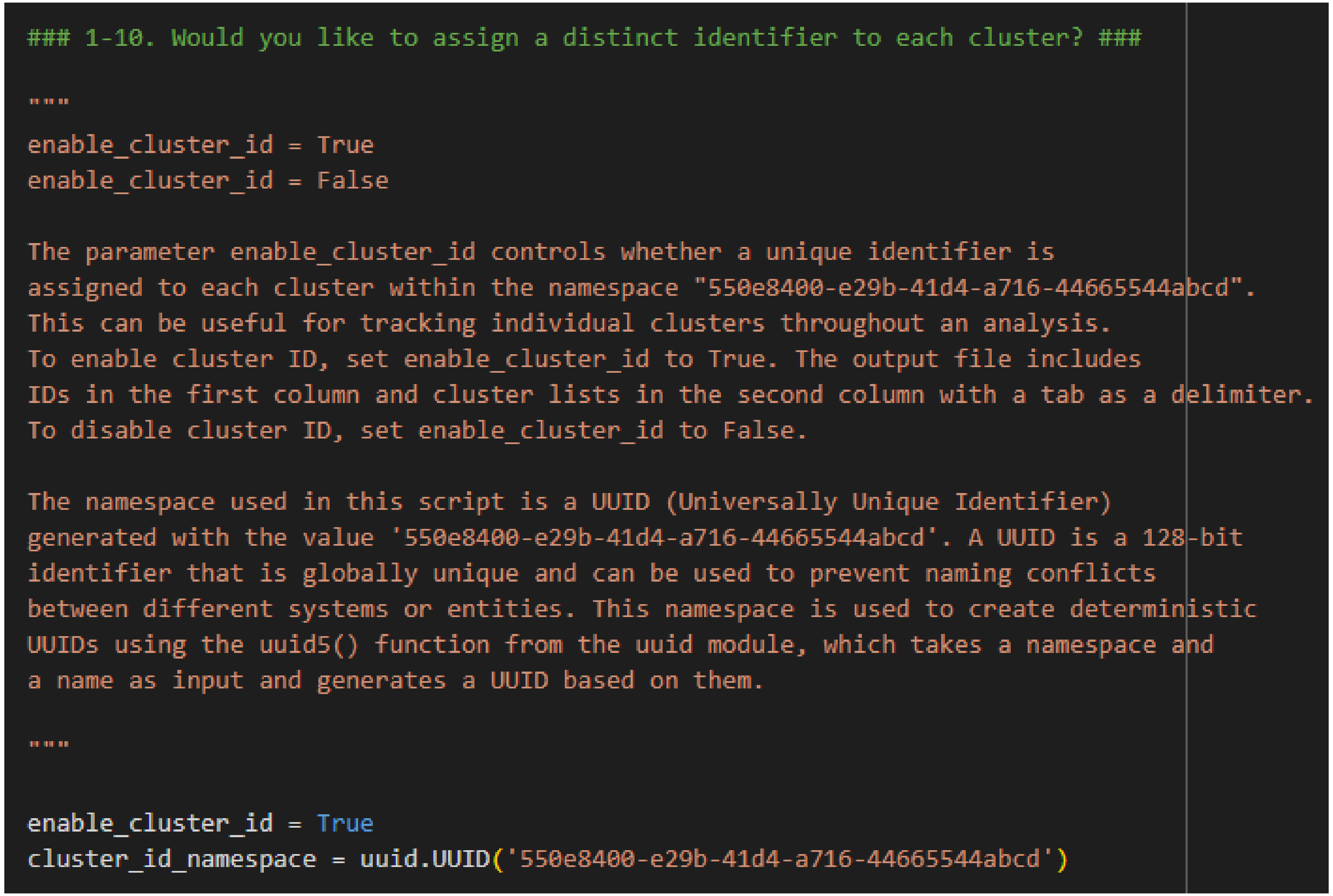
```
### 1-8. If 'hier' is chosen, what is a threshold value? ###

"""
Set a threshold value to filter instance pairs to be put into the same cluster
between 0 and 1
A threshold value is a distance, i.e., 1 - similarity score, between name instances.
E.g., A threshold value of 0.3 is roughly equal to 70 % of probability of name instances
      referring to the same author entity
The lower the threshold value is, the higher the precision score is.

"""

threshold_list = [0.1, 0.3, 6]    # [0] if clustering algorithms other than 'hier' are used

    # <- if various thresholds need to be used, put a starting threshold, an end threshold, and a number of samples in the list
    # e.g., [0.1, 0.3, 5]: this generates a list of thresholds [0.1, 0.15, 0.2, 0.25, 0.3]
```

Choosing evaluation metrics



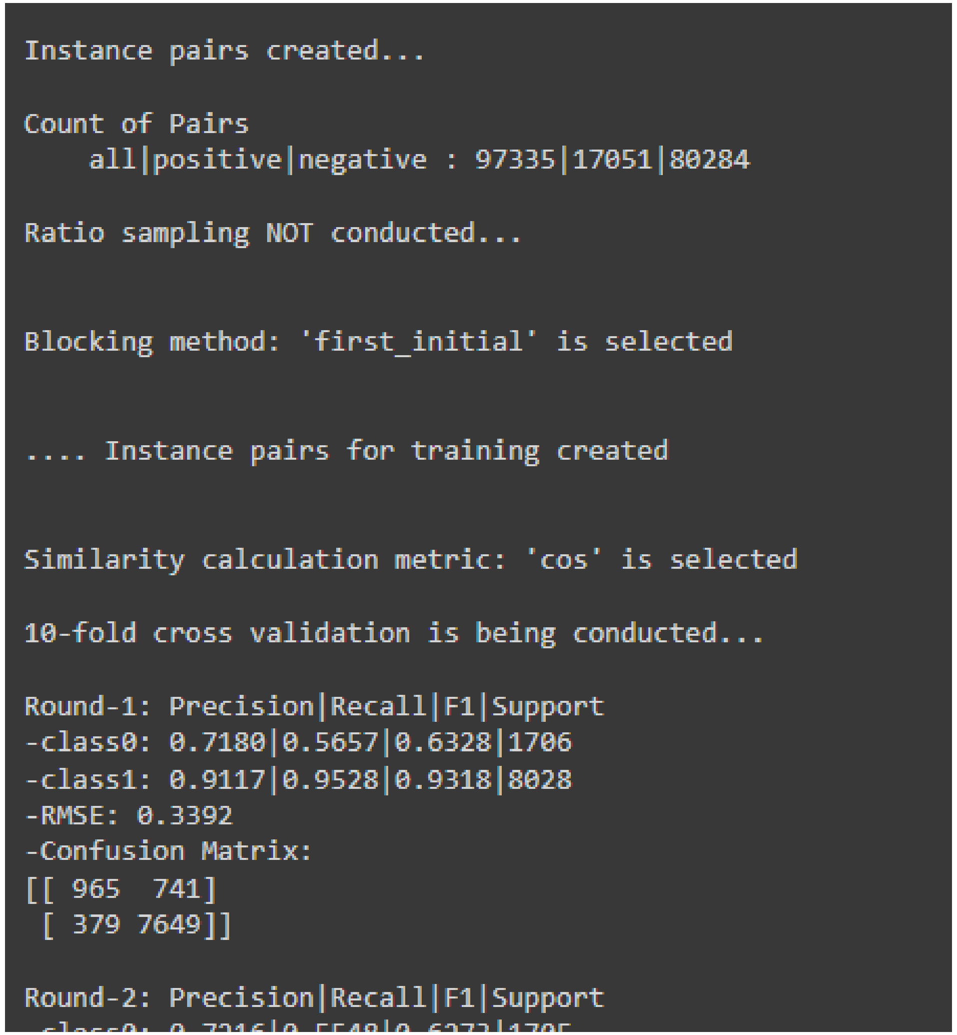
```
### 1-9. Which clustering evaluation metric do you want to use? ###

"""
(1) cluster-f: cluster-f precision/recall/f1
(2) k-metric: k-metric precision/recall/f1
(3) split-lump: splitting & lumping error precision/recall/f1
(4) pairwise-f: paired precision/recall/f1
(5) b-cubed: b3 precision/recall/f1

For more details on clustering evaluation metrics, see a paper below
Kim, J. (2019). A fast and integrative algorithm for clustering performance evaluation
    in author name disambiguation. Scientometrics, 120(2), 661-681.

"""

clustering_metric = "b-cubed"
```

Output with or without IDs

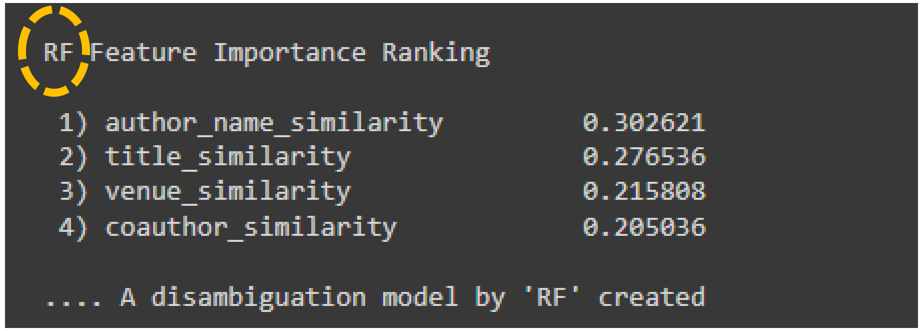


```
### 1-10. Would you like to assign a distinct identifier to each cluster? ###

"""
enable_cluster_id = True
enable_cluster_id = False

The parameter enable_cluster_id controls whether a unique identifier is
assigned to each cluster within the namespace "550e8400-e29b-41d4-a716-44665544abcd".
This can be useful for tracking individual clusters throughout an analysis.
To enable cluster ID, set enable_cluster_id to True. The output file includes
IDs in the first column and cluster lists in the second column with a tab as a delimiter.
To disable cluster ID, set enable_cluster_id to False.

The namespace used in this script is a UUID (Universally Unique Identifier)
generated with the value '550e8400-e29b-41d4-a716-44665544abcd'. A UUID is a 128-bit
identifier that is globally unique and can be used to prevent naming conflicts
between different systems or entities. This namespace is used to create deterministic
UUIDs using the uuid5() function from the uuid module, which takes a namespace and
a name as input and generates a UUID based on them.

"""


enable_cluster_id = True
cluster_id_namespace = uuid.UUID('550e8400-e29b-41d4-a716-44665544abcd')
```

Progress report

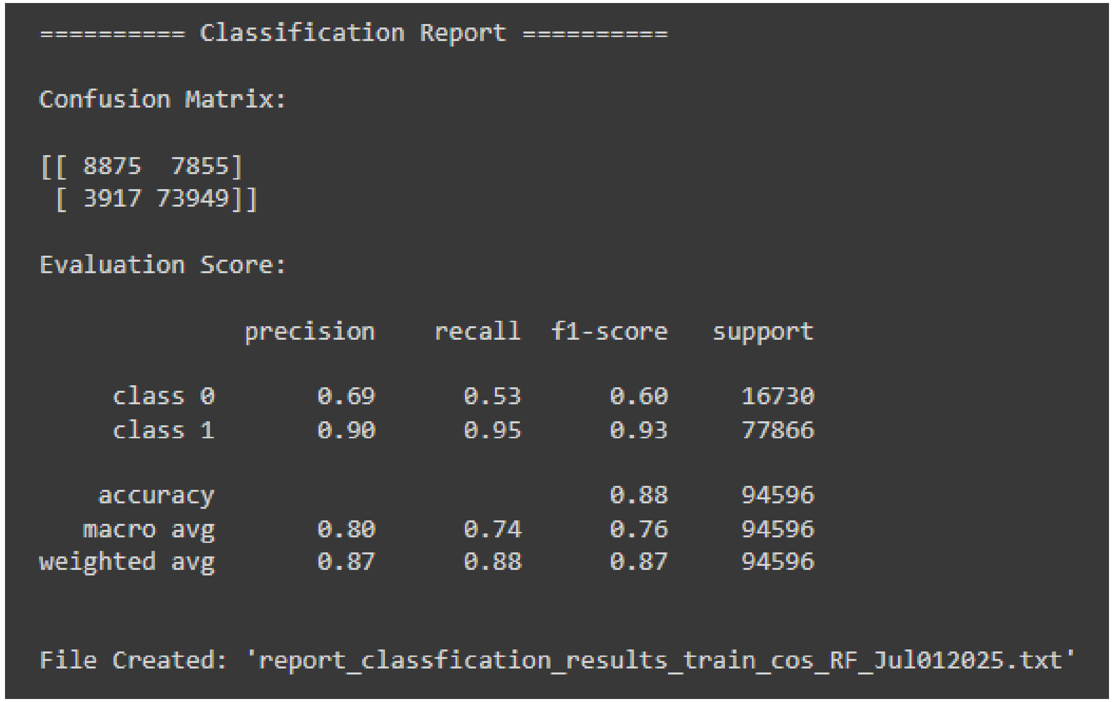


```
Instance pairs created...

Count of Pairs
    all|positive|negative : 97335|17051|80284

Ratio sampling NOT conducted...


Blocking method: 'first_initial' is selected


.... Instance pairs for training created


Similarity calculation metric: 'cos' is selected

10-fold cross validation is being conducted...

Round-1: Precision|Recall|F1|Support
-class0: 0.7180|0.5657|0.6328|1706
-class1: 0.9117|0.9528|0.9318|8028
-RMSE: 0.3392
-Confusion Matrix:
[[ 965  741]
 [ 379 7649]]

Round-2: Precision|Recall|F1|Support
-class0: 0.7216|0.5548|0.6273|1705
```

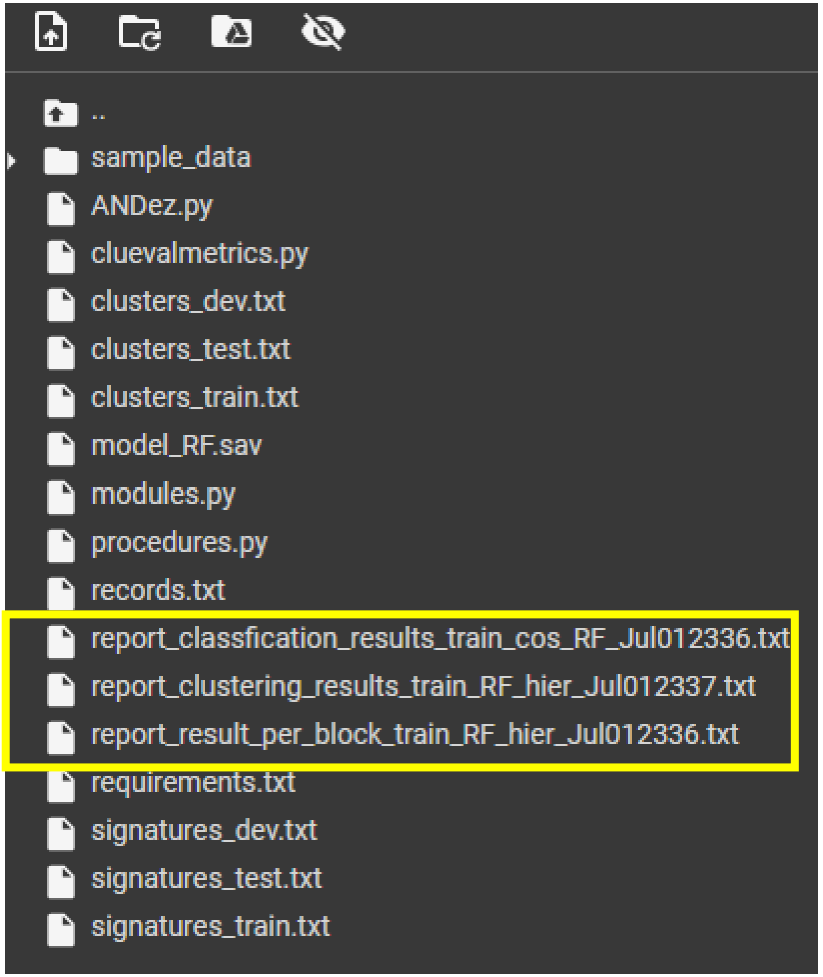


```
RF Feature Importance Ranking

 1) author_name_similarity        0.302621
 2) title_similarity              0.276536
 3) venue_similarity              0.215808
 4) coauthor_similarity           0.205036

.... A disambiguation model by 'RF' created
```

Evaluation report for classification and output files

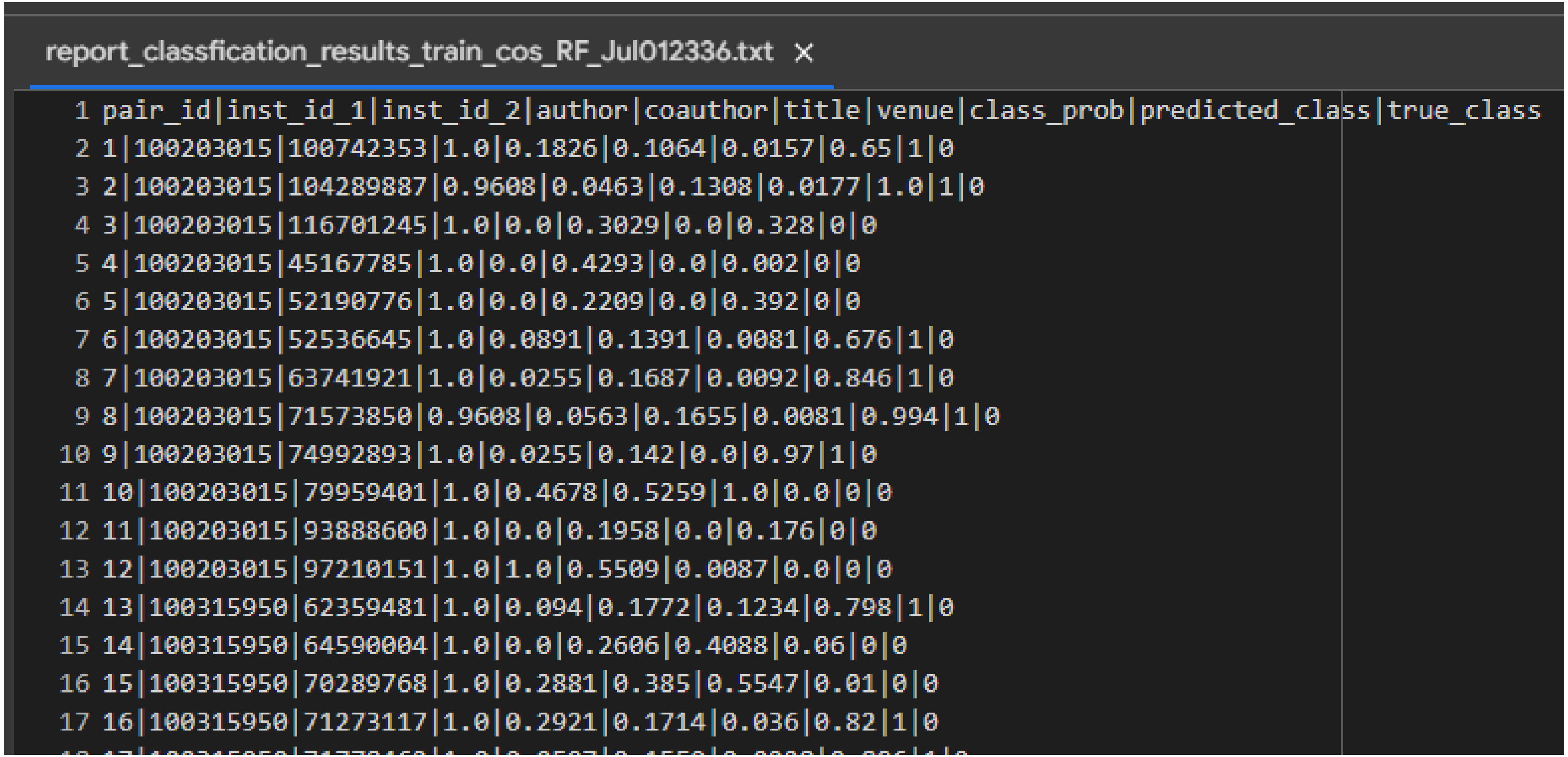


```
========== Classification Report ==========

Confusion Matrix:

[[ 8875  7855]
 [ 3917 73949]]

Evaluation Score:

              precision    recall  f1-score   support

     class 0       0.69      0.53      0.60     16730
     class 1       0.90      0.95      0.93     77866

    accuracy                           0.88     94596
   macro avg       0.80      0.74      0.76     94596
weighted avg       0.87      0.88      0.87     94596


File Created: 'report_classfication_results_train_cos_RF_Jul012025.txt'
```



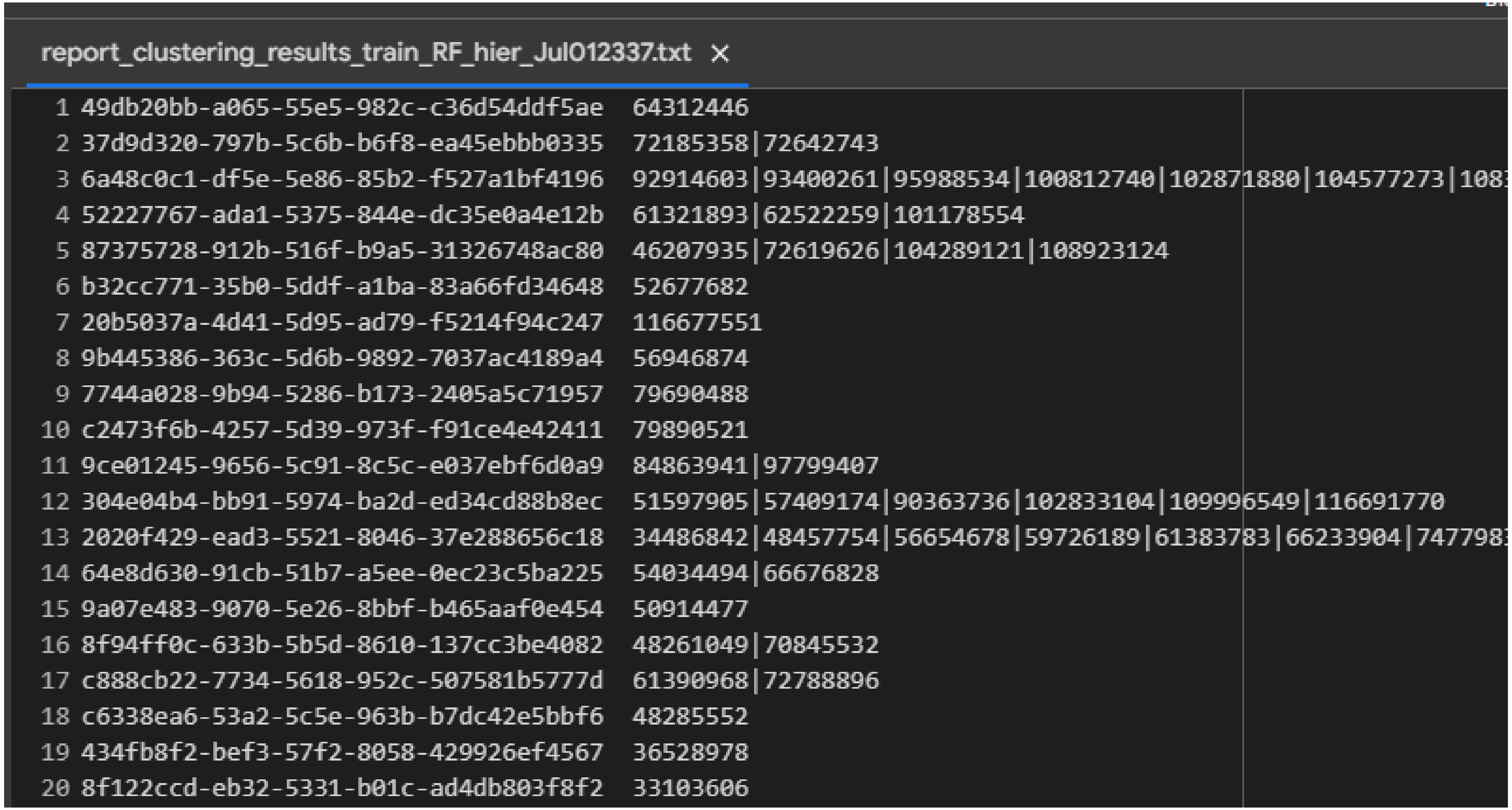
```
..
sample_data
ANDez.py
cluevalmetrics.py
clusters_dev.txt
clusters_test.txt
clusters_train.txt
model_RF.sav
modules.py
procedures.py
records.txt
report_classfication_results_train_cos_RF_Jul012336.txt
report_clustering_results_train_RF_hier_Jul012337.txt
report_result_per_block_train_RF_hier_Jul012336.txt
requirements.txt
signatures_dev.txt
signatures_test.txt
signatures_train.txt
```

Report for classification results



```
pair_id|inst_id_1|inst_id_2|author|coauthor|title|venue|class_prob|predicted_class|true_class
1|100203015|100742353|1.0|0.1826|0.1064|0.0157|0.65|1|0
2|100203015|104289887|0.9608|0.0463|0.1308|0.0177|1.0|1|0
3|100203015|116701245|1.0|0.0|0.3029|0.0|0.328|0|0
4|100203015|45167785|1.0|0.0|0.4293|0.0|0.002|0|0
5|100203015|52190776|1.0|0.0|0.2209|0.0|0.392|0|0
6|100203015|52536645|1.0|0.0891|0.1391|0.0081|0.676|1|0
7|100203015|63741921|1.0|0.0255|0.1687|0.0092|0.846|1|0
8|100203015|71573850|0.9608|0.0563|0.1655|0.0081|0.994|1|0
9|100203015|74992893|1.0|0.0255|0.142|0.0|0.97|1|0
10|100203015|79959401|1.0|0.4678|0.5259|1.0|0.0|0|0
11|100203015|93888600|1.0|0.0|0.1958|0.0|0.176|0|0
12|100203015|97210151|1.0|1.0|0.5509|0.0087|0.0|0|0
13|100315950|62359481|1.0|0.094|0.1772|0.1234|0.798|1|0
14|100315950|64590004|1.0|0.0|0.2606|0.4088|0.06|0|0
15|100315950|70289768|1.0|0.2881|0.385|0.5547|0.01|0|0
16|100315950|71273117|1.0|0.2921|0.1714|0.036|0.82|1|0
```

Report for clustering results



```
49db20bb-a065-55e5-982c-c36d54ddf5ae  64312446
37d9d320-797b-5c6b-b6f8-ea45ebbb0335  72185358|72642743
6a48c0c1-df5e-5e86-85b2-f527a1bf4196  92914603|93400261|95988534|100812740|102871880|104577273|108
52227767-ada1-5375-844e-dc35e0a4e12b  61321893|62522259|101178554
87375728-912b-516f-b9a5-31326748ac80  46207935|72619626|104289121|108923124
b32cc771-35b0-5ddf-a1ba-83a66fd34648  52677682
20b5037a-4d41-5d95-ad79-f5214f94c247  116677551
9b445386-363c-5d6b-9892-7037ac4189a4  56946874
7744a028-9b94-5286-b173-2405a5c71957  79690488
c2473f6b-4257-5d39-973f-f91ce4e42411  79890521
9ce01245-9656-5c91-8c5c-e037ebf6d0a9  84863941|97799407
304e04b4-bb91-5974-ba2d-ed34cd88b8ec  51597905|57409174|90363736|102833104|109996549|116691770
2020f429-ead3-5521-8046-37e288656c18  34486842|48457754|56654678|59726189|61383783|66233904|747798
64e8d630-91cb-51b7-a5ee-0ec23c5ba225  54034494|66676828
9a07e483-9070-5e26-8bbf-b465aaf0e454  50914477
8f94ff0c-633b-5b5d-8610-137cc3be4082  48261049|70845532
c888cb22-7734-5618-952c-507581b5777d  61390968|72788896
c6338ea6-53a2-5c5e-963b-b7dc42e5bbf6  48285552
434fb8f2-bef3-57f2-8058-429926ef4567  36528978
8f122ccd-eb32-5331-b01c-ad4db803f8f2  33103606
```

Questions & Feedbacks

- Jinseok Kim (jinseokk@umich.edu)