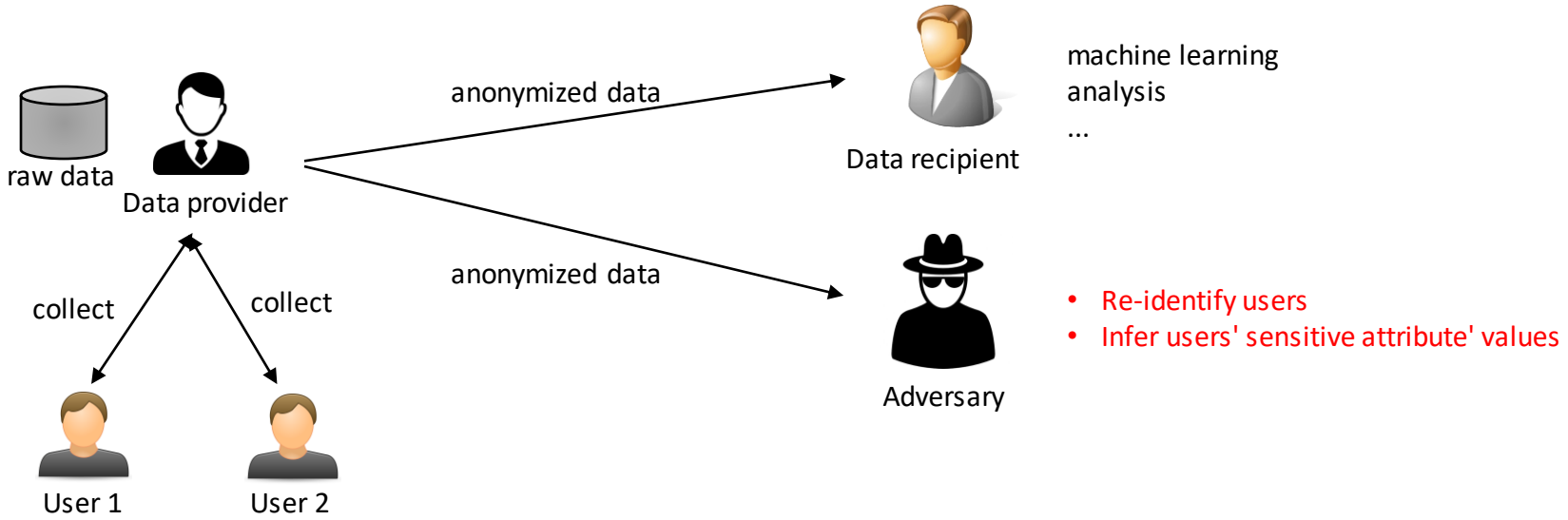
Time-Aware Anonymization of Knowledge Graphs

Anh-Tu Hoang, Barbara Carminati, Elena Ferrari

{anhtu.hoang, barbara.carminati, elena.ferrari}@uninsubria.it

DiSTA, University of Insubria, Italy

Data Publishing Scenarios



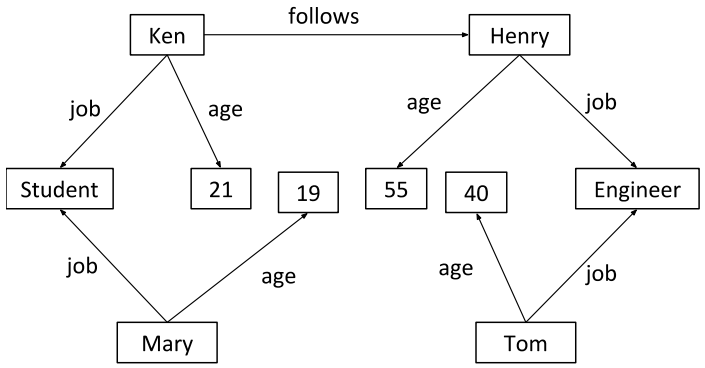
Anonymization: k-anonymity
Supported data types: relational data, graphs

Disadvantages:
- Lack of the associations between users' attributes and relationships.
- Limited supported data types.

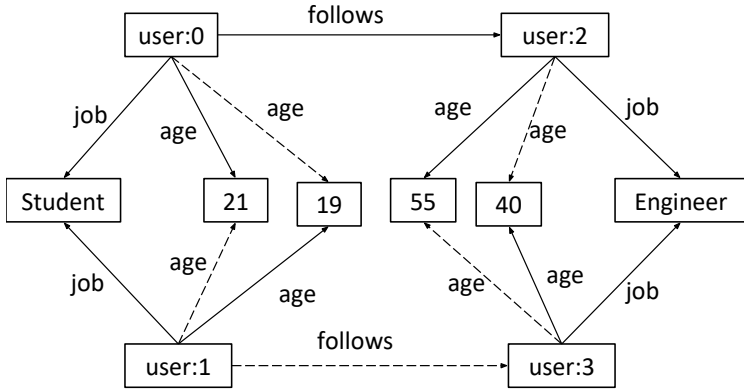
Anonymization of Knowledge Graphs (KGs)

Our previous principles:
- k-Attribute Degree[1]
- k^w-Time Varying Attribute Degree[2]

Advantage: can replace state-of-the-art anonymization principles for graphs (relationships) and relational data (attributes).



raw KG



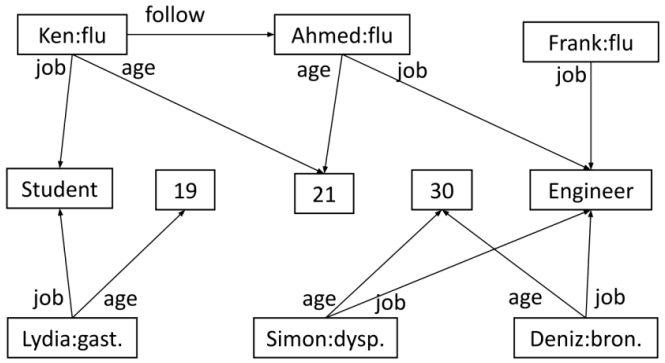
2-anonymous KG

For every user, there are at least k-1 other users having the same attribute values and relationship out-/in-degrees to those of his/hers.

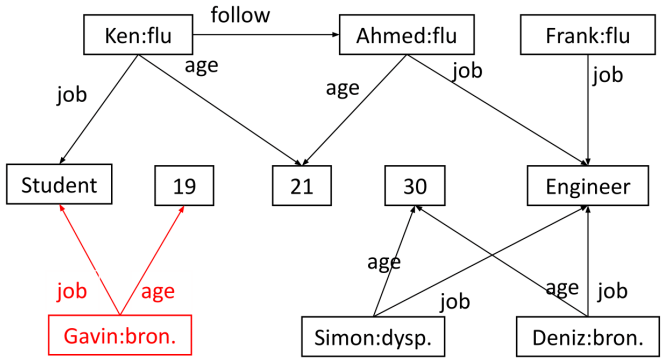
[1] Anh-Tu Hoang, Barbara Carminati, Elena Ferrari: Cluster-Based Anonymization of Knowledge Graphs. ACNS (2) 2020: 104-123
[2] Anh-Tu Hoang, Barbara Carminati, Elena Ferrari: Privacy-Preserving Sequential Publishing of Knowledge Graphs. ICDE 2021: 2021-2026

Sequential Publishing of Knowledge Graphs

G_1

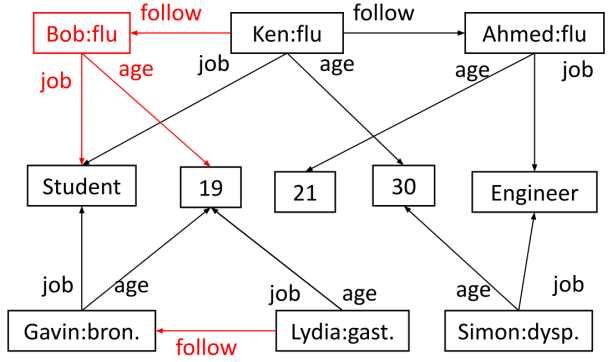


G_2



new users: Gavin
deleted users: Lydia

G_3

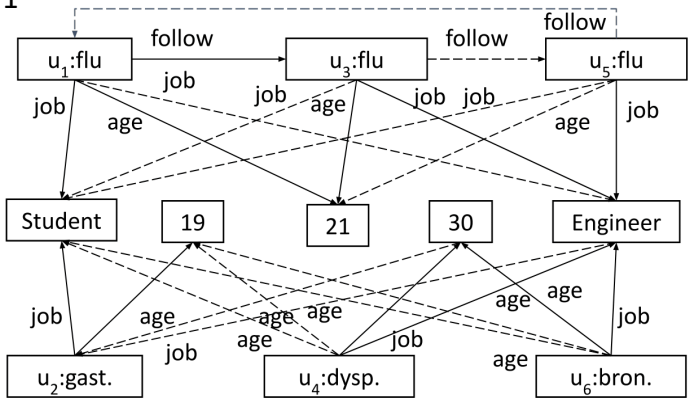


new users: Bob
deleted users: Deniz
re-inserted users: Lydia
updated users: Ken

Threat Model

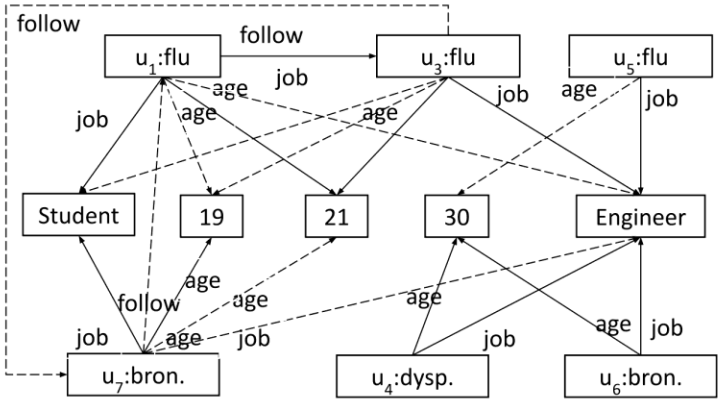
Anonymized versions satisfying 3-Attribute Degree

G'_1



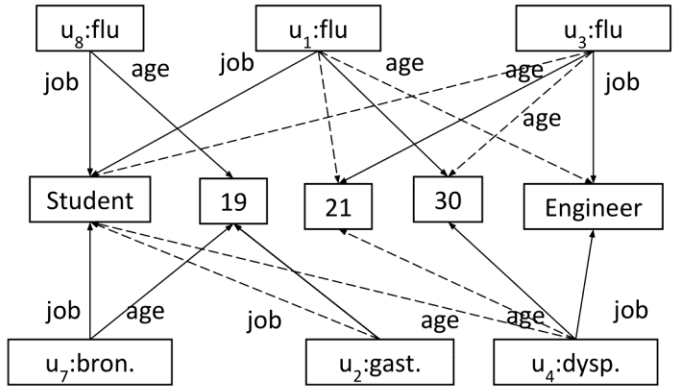
$l_1(u_1)=l_1(u_3)=l_1(u_5)$
$l_1(u_2)=l_1(u_4)=l_1(u_6)$

G'_2



$l_2(u_1)=l_2(u_3)=l_2(u_7)$
$l_2(u_4)=l_2(u_5)=l_2(u_6)$

G'_3



$l_3(u_1)=l_3(u_3)=l_3(u_4)$
$l_3(u_2)=l_3(u_7)=l_3(u_8)$

Adversary Knowledge Extracted from the versions:
- the sequence of attribute values and relationship degrees
- the sequence of signatures

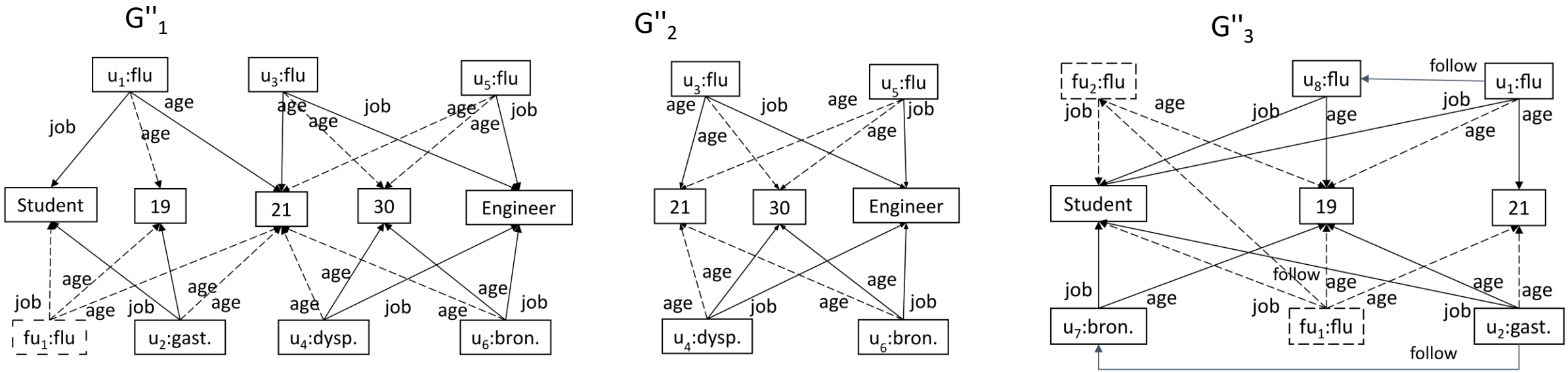
Target user u_1:
- u_1 is re-identified in G_2 since his/her sequence of attribute values and relationship degrees is unique.
- u_1's sensitive value in G_1 is inferred as flu.

$I(u_1)=<I1(u_1), I2(u_1), I3(u_1)>$
- $I_1(u_1)$={(job,Student),(job,Engineer),(age,21), (follow-out-degree, 1), (follow-in-degree, 1)}
- $I_2(u_1)$={(job,Student),(job,Engineer),(age,21), (age, 19), (follow-out-degree, 1)}
- $I_3(u_1)$={(job,Student),(job,Engineer),(age,21),(age(30)}

Sig1(u1)={flu}, Sig2(u1)={flu, bron.}, Sig3(u1)={flu,dysp}

(k,l)-Sequence Attribute Degree ((k,l)-sad)

- Let g_t be a sequence of published anonymized KGs at time t. g_t satisfies (k,l)-sad if and only if for every user u in g_t:
  - There exists a set of users C(u) whose the sequence of attribute values and degrees in g_t are identical to those of u.
  - Signatures of u are identical in all KGs in g_t.



The Sequence of Anonymized KGs satisfying (k,l)-sad.

Anonymization Algorithm

A cluster c is valid if:
1/ |c| >= k
2/ all users in c have the same sequence of attribute values and degrees in previous anonymized KGs
3/ signatures of c's users share at least l distinct values
4/ the signatures are identical to those of them in previous anonymized KGs

parameters: k, l, max_dist

clustering algorithm
(k-medoids, HDBSCAN)

raw KG at
time t

Clusters Generation

- generate clusters
- modify to make clusters valid

clusters

Knowledge Graph Generalization

anonymized KG
at time t

ADS-Table
(storing the sequence of attribute values and relationship degrees in published KGs of all users)

Flexibility:
- Data providers can use any clustering algorithms
- Protect only identities or both identities and sensitive attribute values

High-quality anonymized KGs

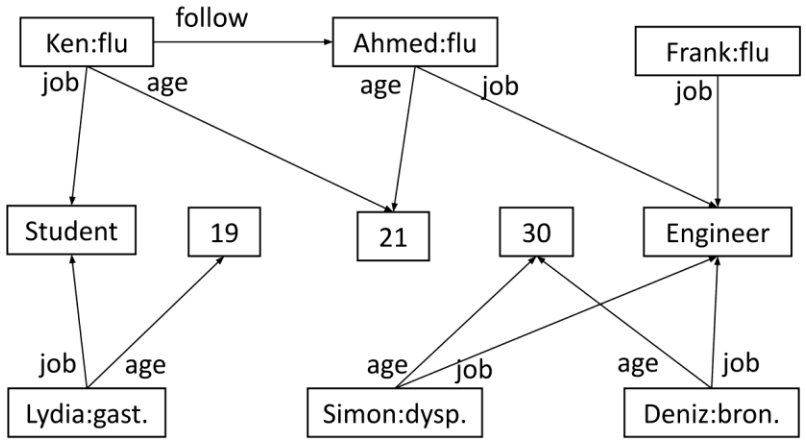
Clusters Generation

- Gather users having the same sequence of anonymized data in previous anonymized KGs.
  - A cluster of new users
  - Clusters of existed users whose anonymized are published in previous anonymized KGs.
- For each cluster, use one of three modification strategy: New Users Handling, Deleted Users Handling, Updated/Re-Inserted Users Handling.
- New Users Handling: modifies the cluster of new users.
- Deleted Users Handling: removes users from clusters of existed users.
- Updated/Re-Inserted Users Handling: splits clusters to improve quality.

Clusters Generation (1)

G_o

k=3,l=2



Info	Users
(I_0^0)	{u1, u2, u3, u4, u5, u6}

ADS-Table H_0

- user indexes: u1(Ken), u2(Lydia), u3(Ahmed), u4(Simon), u5(Frank), u6(Deniz)
- cluster of new users: {u1, u2, u3, u4, u5, u6}

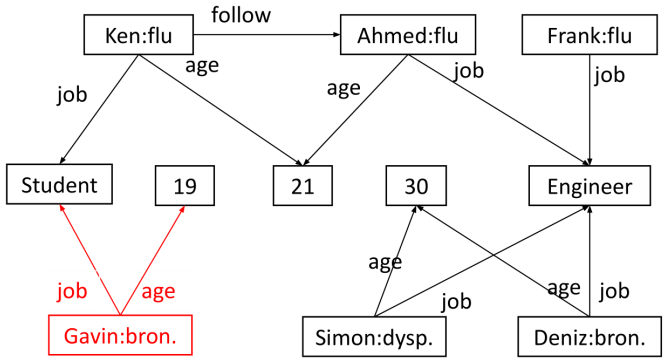
New users handling:
- split the cluster with provided clustering algorithm: {u1, u2}, {u5}, {u3,u4,u6}
- detect invalid clusters: {u1,u2}, {u5}
- modify invalid clusters:
  + add a fake user fu1: {u1,u2,fu1}
  + add u5 to valid cluster: {u3,u4,u6,u5}

- return clusters: {u1, u2,fu1}, {u3,u4,u6,u5}

Clusters Generation (2)

G_1

k=3,l=2



Info	Users
(I1)	{u1, u2, fu1}
(I2)	{u3, u4, u5, u6}
()	{u7}

ADS-Table H_1

- user indexes: u1(Ken), u2(Lydia), u3(Ahmed), u4(Simon), u5(Frank), u6(Deniz), u7(Gavin)
- deleted user: u2
- cluster of new users: {u7}
- clusters of existed users: {u1, u2, fu1}, {u3,u4,u6,u5}

New Users Handling:
- do not add fake users to {u7} since adding fake users (at least 2 users) generates lower quality data than removing the cluster (removing 1 user)

Deleted Users Handling:
- delete users removed by data providers: {u1, fu1}, {u3,u4,u6,u5}
- remove invalid cluster: {u1, fu1}

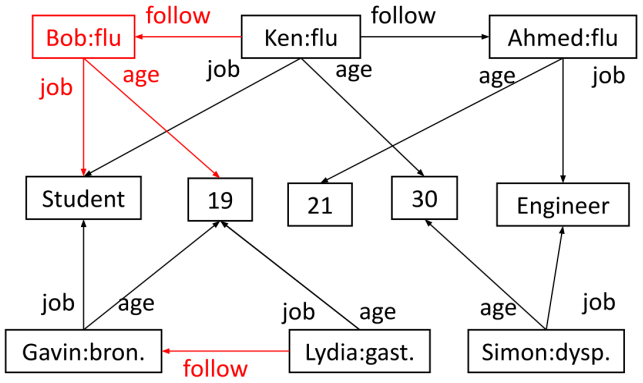
Updated/Re-Inserted Users Handling:
- split big clusters such that their signatures are unchanged.

Return Clusters: {u3,u4,u6,u5}

Clusters Generation (3)

G_2

k=3,l=2



Info	Users
(I1, ∅)	{u1, u2, fu1}
(I2, I2')	{u3, u4, u5, u6}
()	{u7, u8}

ADS-Table H_2

- user indexes: u1(Ken), u2(Lydia), u3(Ahmed), u4(Simon), u5(Frank), u6(Deniz), u7(Gavin), u8(Bob)
- deleted user: u4,u5
- cluster of new users: {u7, u8}
- re-inserted user: u2
- clusters of existed users: {u1, u2, fu1}, {u3,u4,u6,u5}

New Users Handling:
- add a fake user fu2: {u7,u8,fu2}

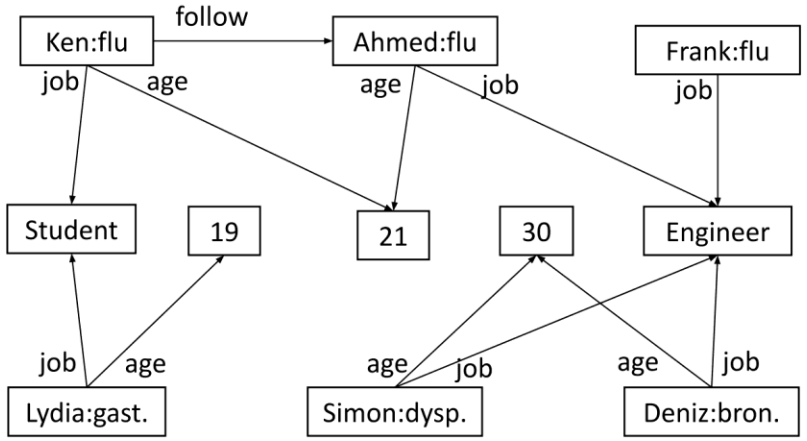
Deleted Users Handling:
- remove users removed by data providers: {u3,u6}
- remove invalid cluster: {u3,u6}

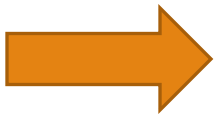
Updated/Re-Inserted Users Handling:
- split big clusters such that their signatures are unchanged.

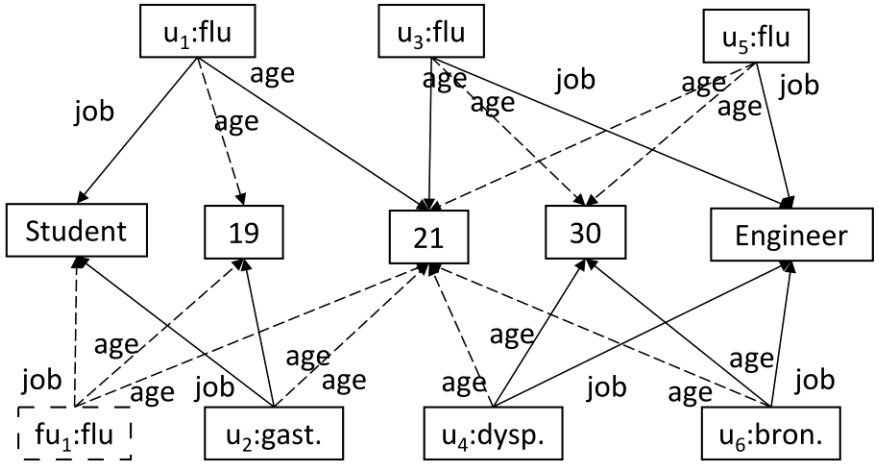
Return Clusters: {u7,u8,fu2}, {u3,u4,u6,u5}

Knowledge Graph Generalization

- Add and remove fake edges to ensure that the attribute values and relationship degrees of users in the same clusters are identical.







{u1, u2,fu1}, {u3,u4,u6,u5}

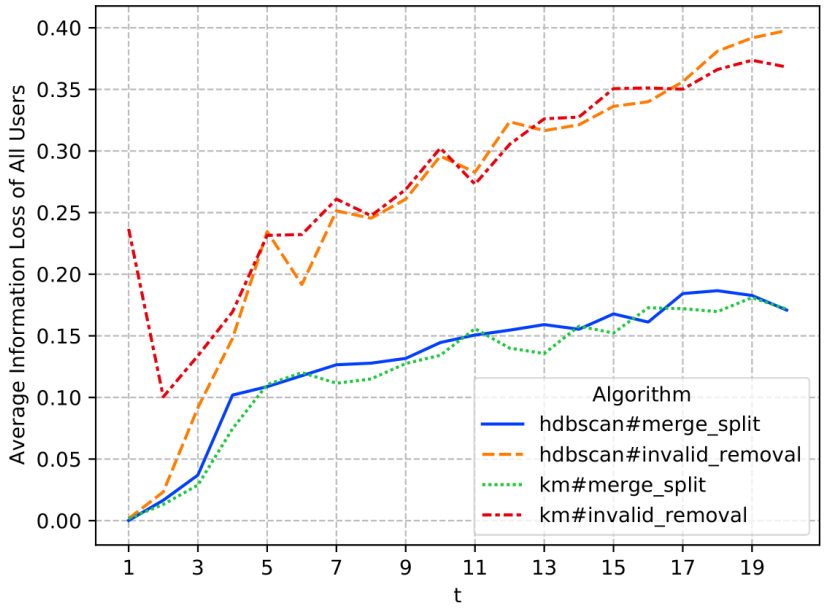
Evaluation

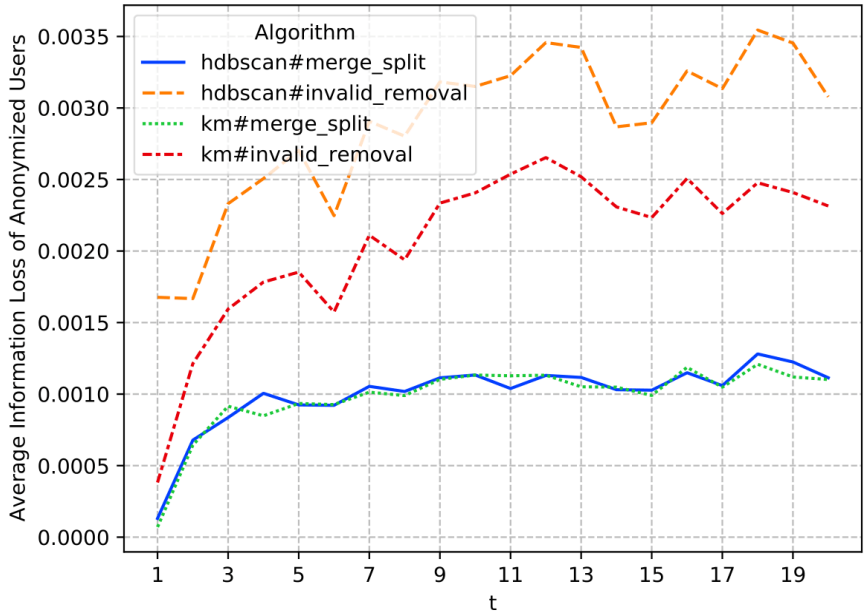
- Four real-life datasets: Email-Eu-core, Email-temp, Yago, Freebase
- Metrics:
  - Attribute Degree Information Loss of Remaining Users (RADM): calculates the information loss of users in anonymized KGs.
  - Overral Attribute Degree Information Loss Metric (ADM): calculates RADM and considers the information loss of removed users as 1.
- Experiments:
  - Evaluate the impact of parameters
  - Evaluate the impact of monitor published KGs
  - Comparative Evaluation

Impact of clustering algorithms

Spliting clusters improve the quality of anonymized KGs.
k-Medoids are better than HDBSCAN in anonymizing KGs.

- Merge_split: uses three strategies to split clusters
- Invalid_removal: removes invalid clusters

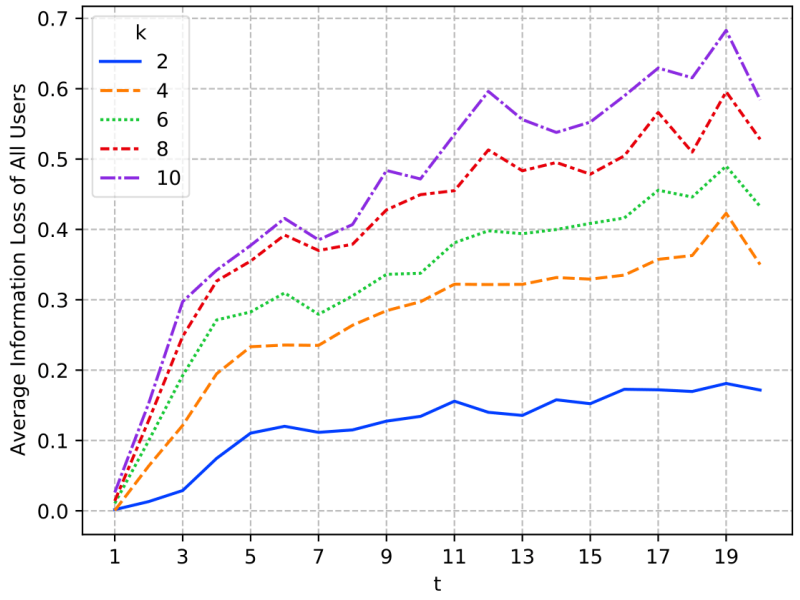


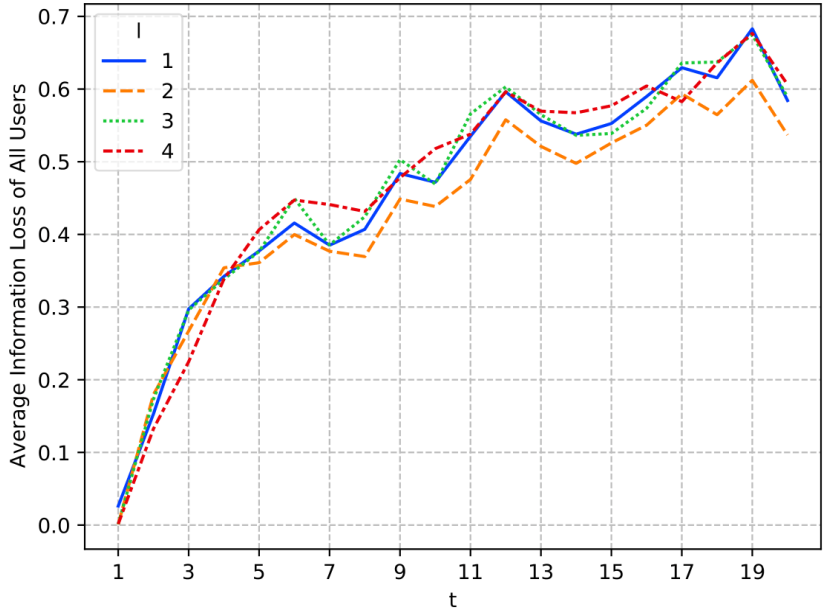


Email-temp

Impact of k, l

K has higher impact than l on the information loss of anonymized KGs
- Increasing k increases the information loss
- Increasing l does not always increase the information loss

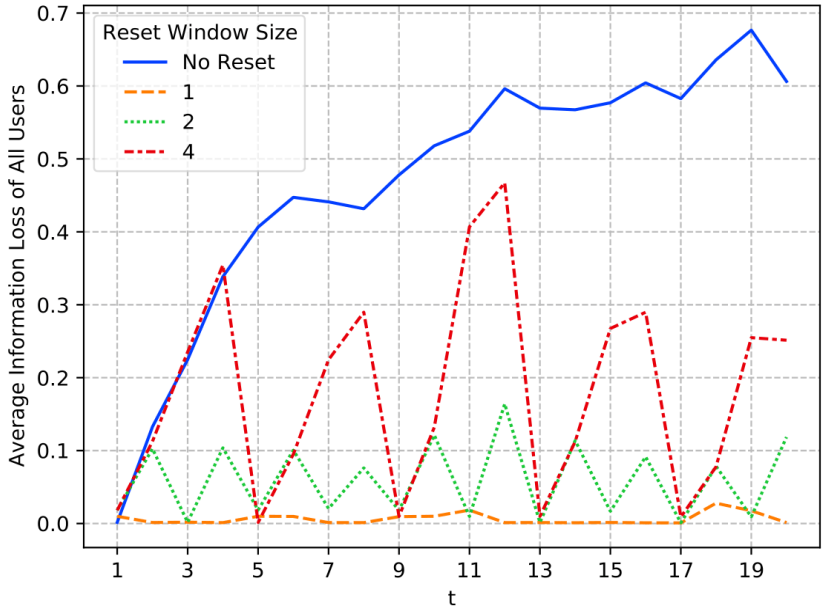


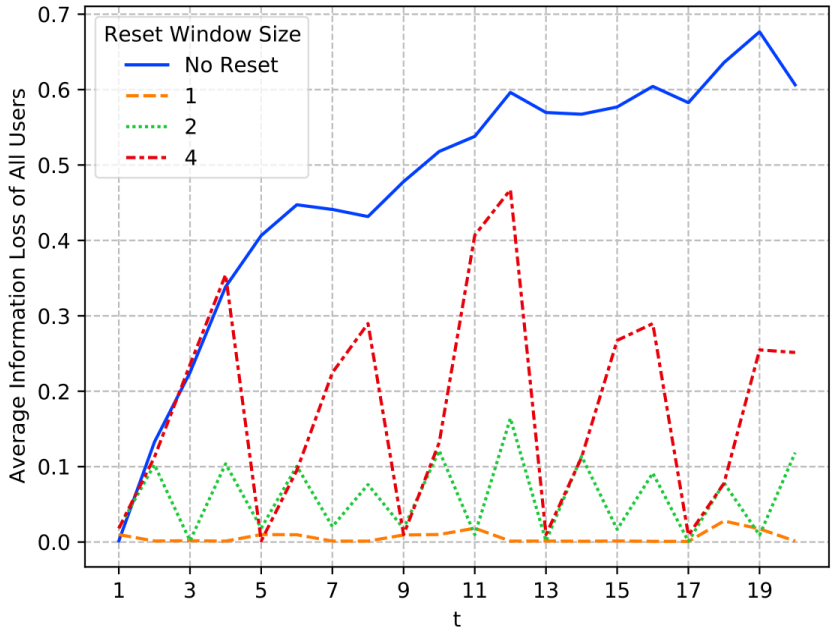


Email-Temp

Impact of monitor published KGs

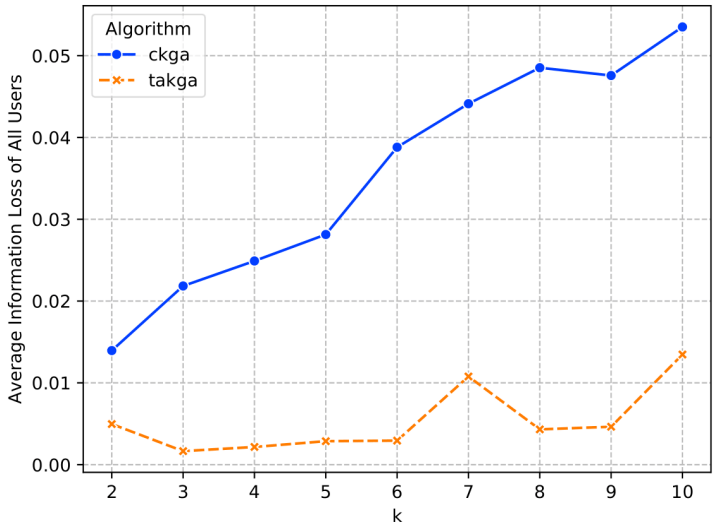
The more monitored KGs, the higher the information loss is.
However, data providers can decide to decrease the information loss by resetting the monitored KGs.



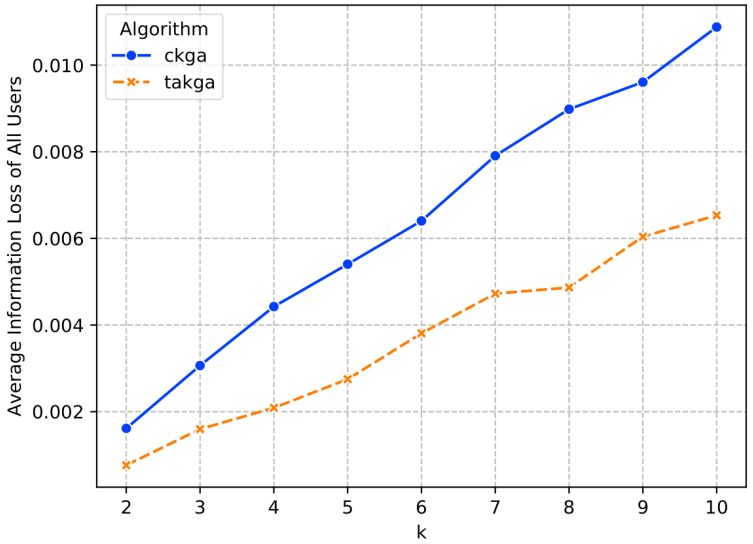


Comparative Evaluation

Generate higher quality anonymized KGs than previous work on anonymizing a single snapshop.



Email-Temp



Freebase

Conclusion

- (k,l)-Sequence Attribute Degree: a new principle for protecting users' privacy in KGs
  - Can use for not only KGs but also relational data, graphs.
  - Flexible on deciding the privacy protection: linking attack, attribute attack.
- Anonymization Algorithm:
  - Minimize the information loss.
  - Handle most popular data updates: insert, delete, update, re-insert users.
- Future work
  - Decentralized models for anonymizing KGs

References

[1] Anh-Tu Hoang, Barbara Carminati, Elena Ferrari: Cluster-Based Anonymization of Knowledge Graphs. ACNS (2) 2020: 104-123

[2] Anh-Tu Hoang, Barbara Carminati, Elena Ferrari: Privacy-Preserving Sequential Publishing of Knowledge Graphs. ICDE 2021: 2021-2026

Thank you for your attention