The Truth Value Project

truthvalue.org

presented by

Massimo Di Pierro
@ DePaul University

The Team

Luca de Alfaro, UC Santa Cruz, USA

Massimo Di Pierro, DePaul University, Chicago, USA

Stefano Moret, EPFL, Switzerland

Eugenio Tacchini, Università Cattolica, Piacenza, Italy

Gabriele Ballarin, Independent Researcher, Italy

Marco L. Della Vedova, Università Cattolica, Brescia, Italy

- about me and this paper
- data collection
- unsupervised learning
- supervised learning (logistic / topic modeling)
- supervised learning (harmonic algorithm)
- results
- code examples
- conclusions

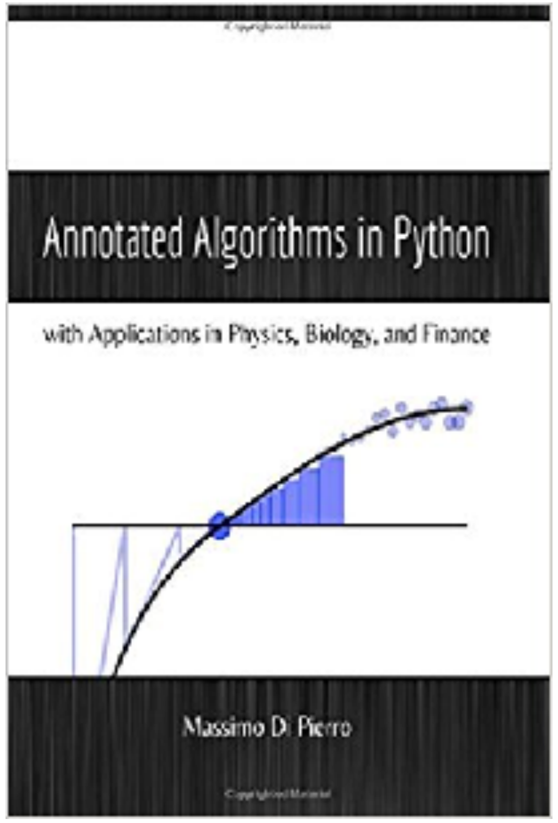
About me

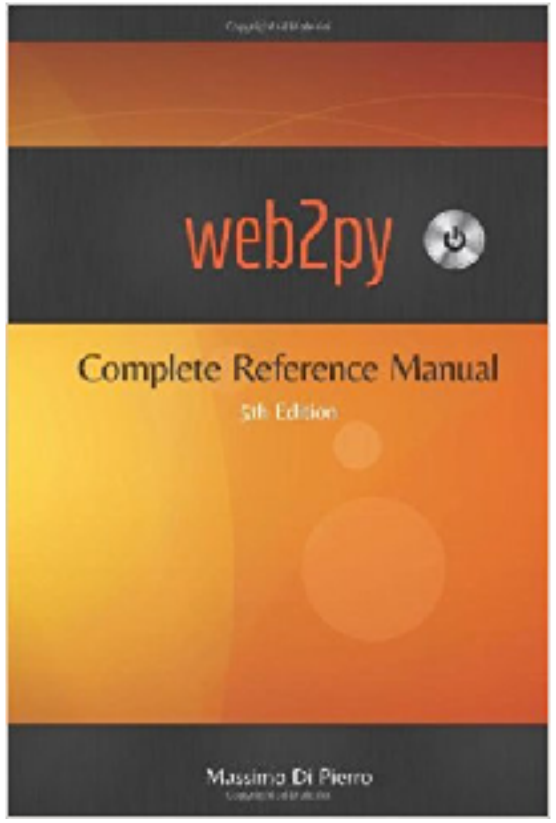
1999: PhD in Physics

1999 - 2002: postdoc at Fermilab

2002 - today: Prof in CS at DePaul (co-director MS-Computational Finance)

2007: created web2py (best framework for Python web app, still!)





2017

Some Like it Hoax:
Automated Fake News Detection in Social Networks

Eugenio Tacchini[1], Gabriele Ballarin[2], Marco L. Della Vedova[3], Stefano Moret[4], and Luca de Alfaro[5]

[1] *Università Cattolica, Piacenza, Italy.* eugenio.tacchini@unicatt.it
[2] *Independent researcher.* gabriele.ballarin@gmail.com
[3] *Università Cattolica, Brescia, Italy.* marco.dellavedova@unicatt.it
[4] *École Polytechnique Fédérale de Lausanne, Switzerland.* moret.stefano@gmail.com
[5] *Department of Computer Science, UC Santa Cruz, CA, USA.* luca@ucsc.edu

https://arxiv.org/pdf/1704.07506.pdf

2018 **Facebook** **Twitter**

Automatic online Fake News Detection combining Content and Social Signals

Reputation Systems for News on Twitter: A Large-Scale Study

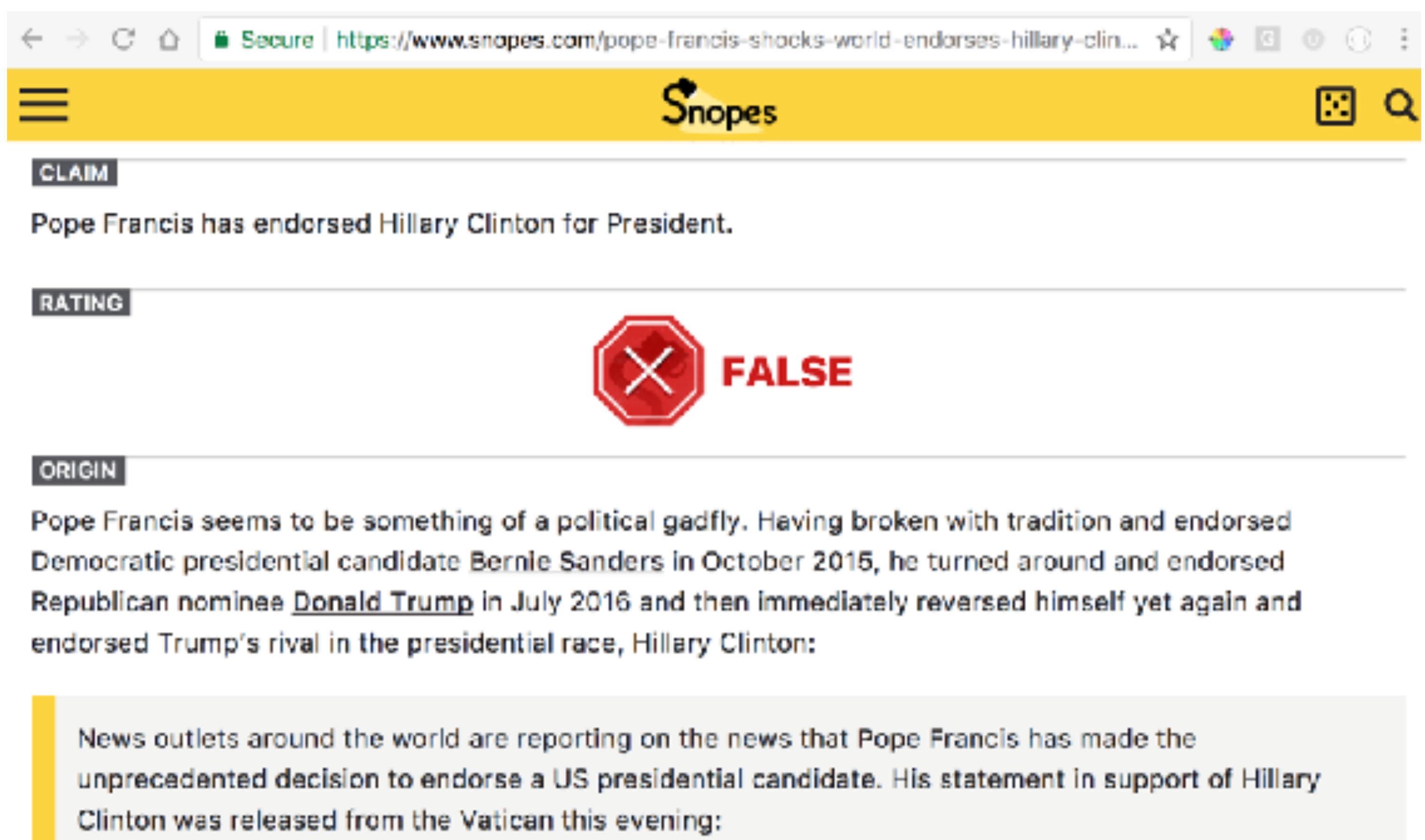
Goal

~~find fake news~~

help humans identify news items that
should be double checked

Fake News!


URL
Red Nation Rising
@RedNationRising
Follow
Pope Francis Shocks World, Endorses Donald Trump for President, Releases Statement #RedNationRising
endingthefed.com/pope-francis-s ...
10:36 PM - 13 Oct 2016
56 Retweets 46 Likes
user


Secure https://www.snopes.com/pope-francis-shocks-world-endorses-hillary-clin...
Snopes
CLAIM
Pope Francis has endorsed Hillary Clinton for President.
RATING
FALSE
ORIGIN
Pope Francis seems to be something of a political gadfly. Having broken with tradition and endorsed Democratic presidential candidate Bernie Sanders in October 2015, he turned around and endorsed Republican nominee Donald Trump in July 2016 and then immediately reversed himself yet again and endorsed Trump's rival in the presidential race, Hillary Clinton:
News outlets around the world are reporting on the news that Pope Francis has made the unprecedented decision to endorse a US presidential candidate. His statement in support of Hillary Clinton was released from the Vatican this evening:

~~Fake News~~

Fake or Misleading News

Fox News @FoxNews

Follow

.@POTUS: "We have ended the war on American Energy and we have ended the war on beautiful clean coal. We are now very proudly an exporter of energy to the world." #SOTU fxn.ws/2GvHyd5

DONALD TRUMP

"We are now, very proudly, an exporter of energy to the world."

— *PolitiFact National* on *Wednesday, January 31st, 2018*

Still a net energy importer

Fake or Misleading News

vote

who is tweeting?

I am biased

How does my bias play a role in this analysis?

Choice of news sources to follow

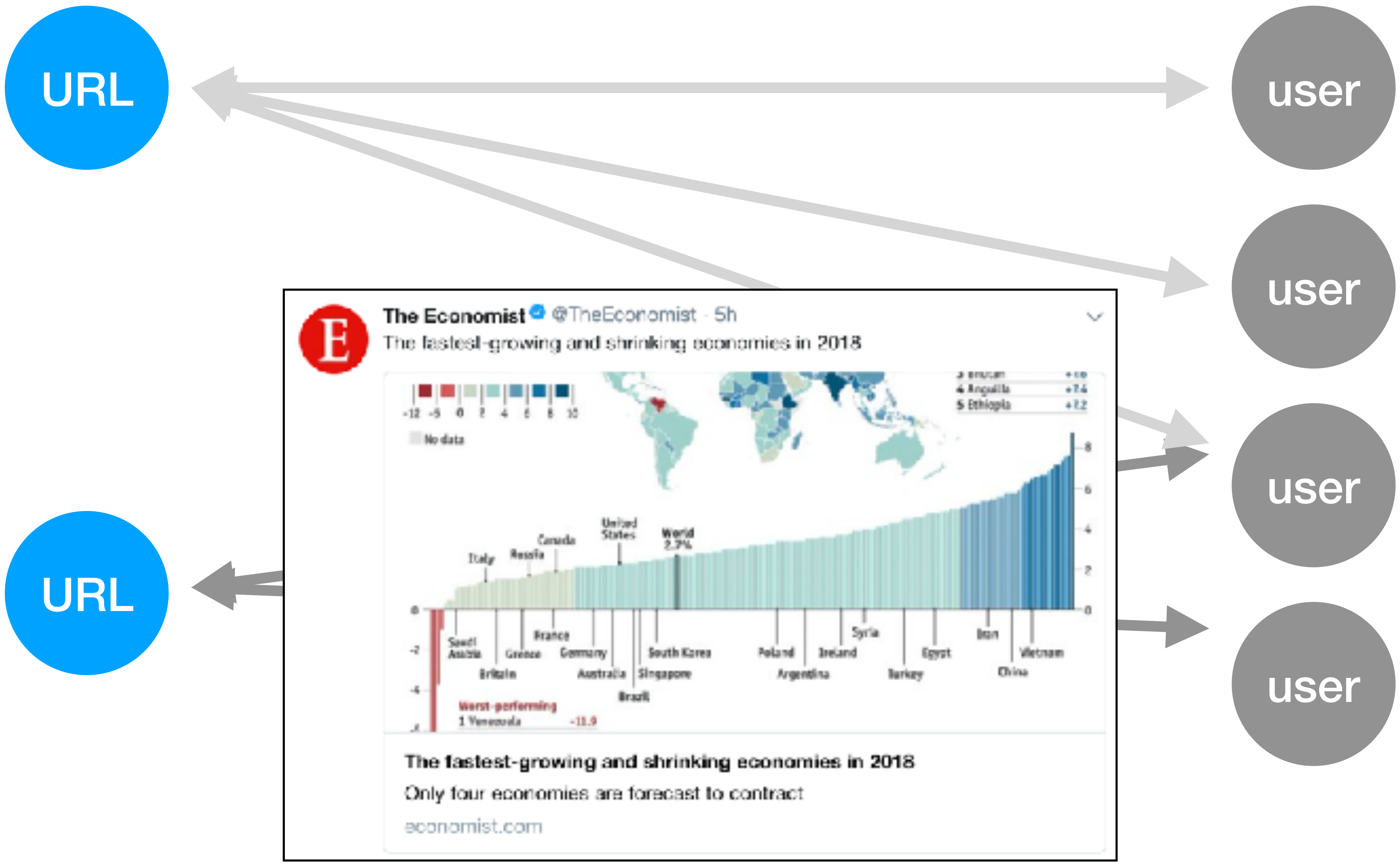
Choice of sources of ground truth

Collected Data

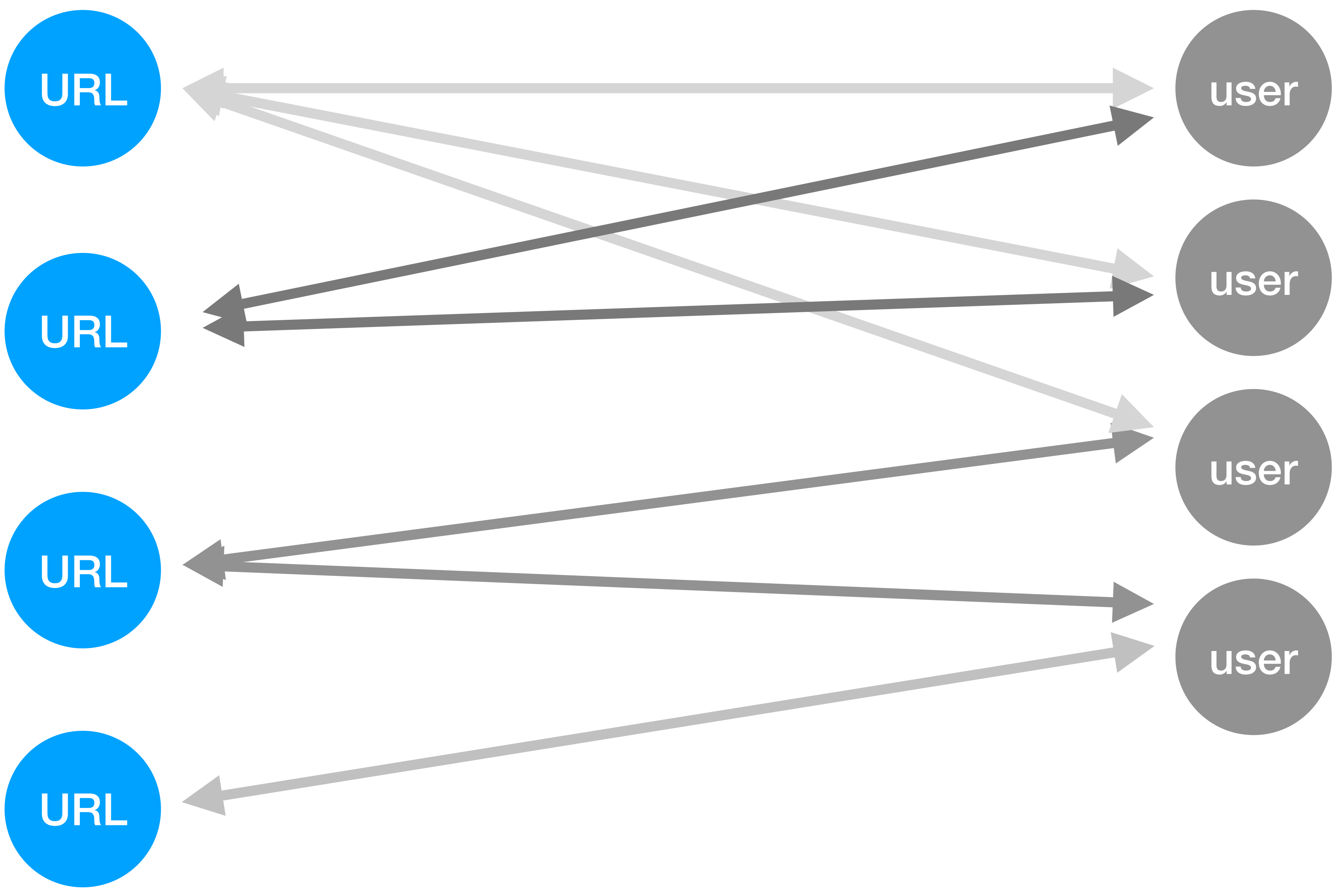
Master Graph

Master Graph

Master Graph



Master Graph



Data Structure

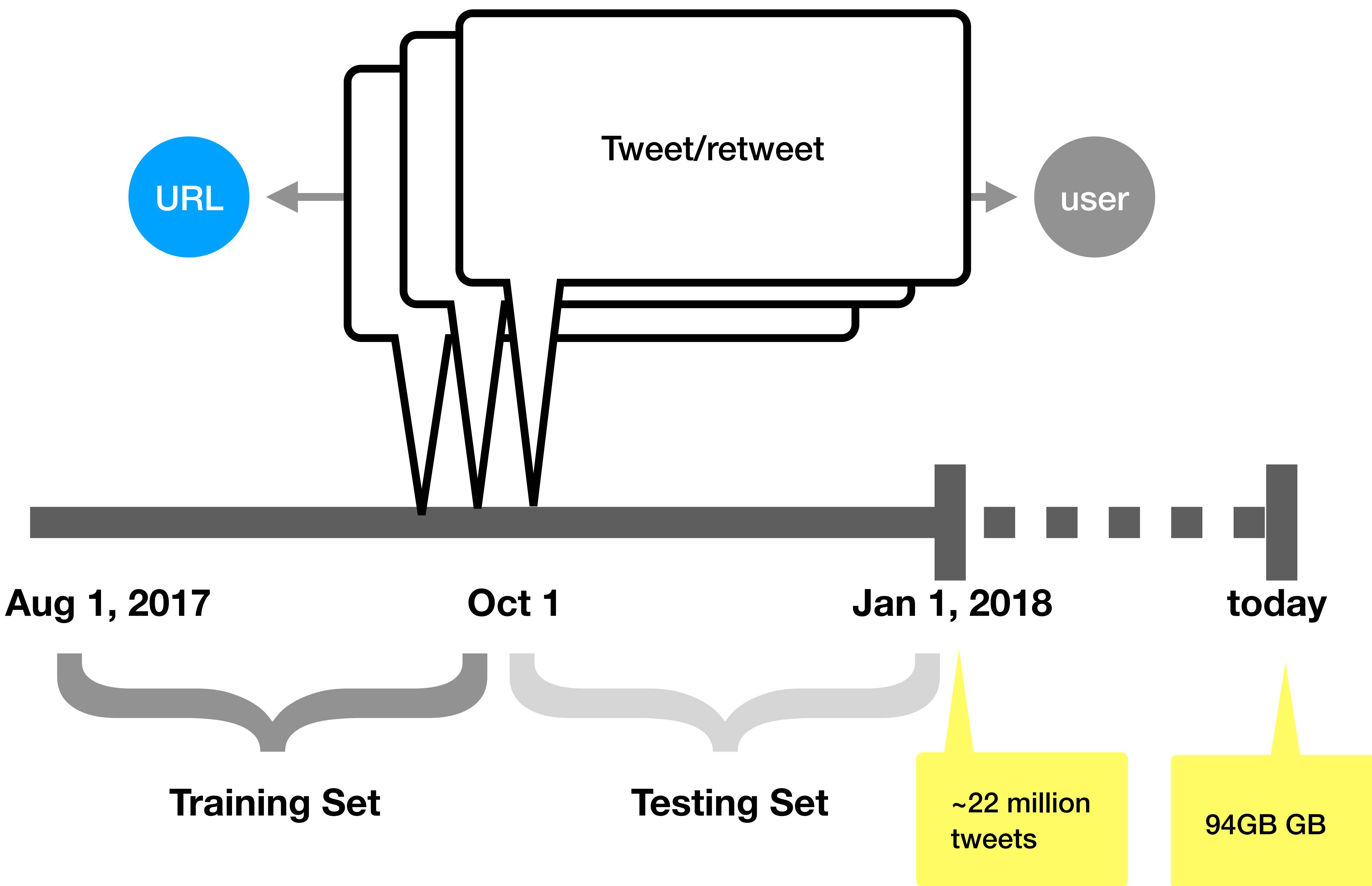
normalized URL
page title
description
semantic metadata
full page HTML

text of the tweet
timestamp
username
URL
normalized URL

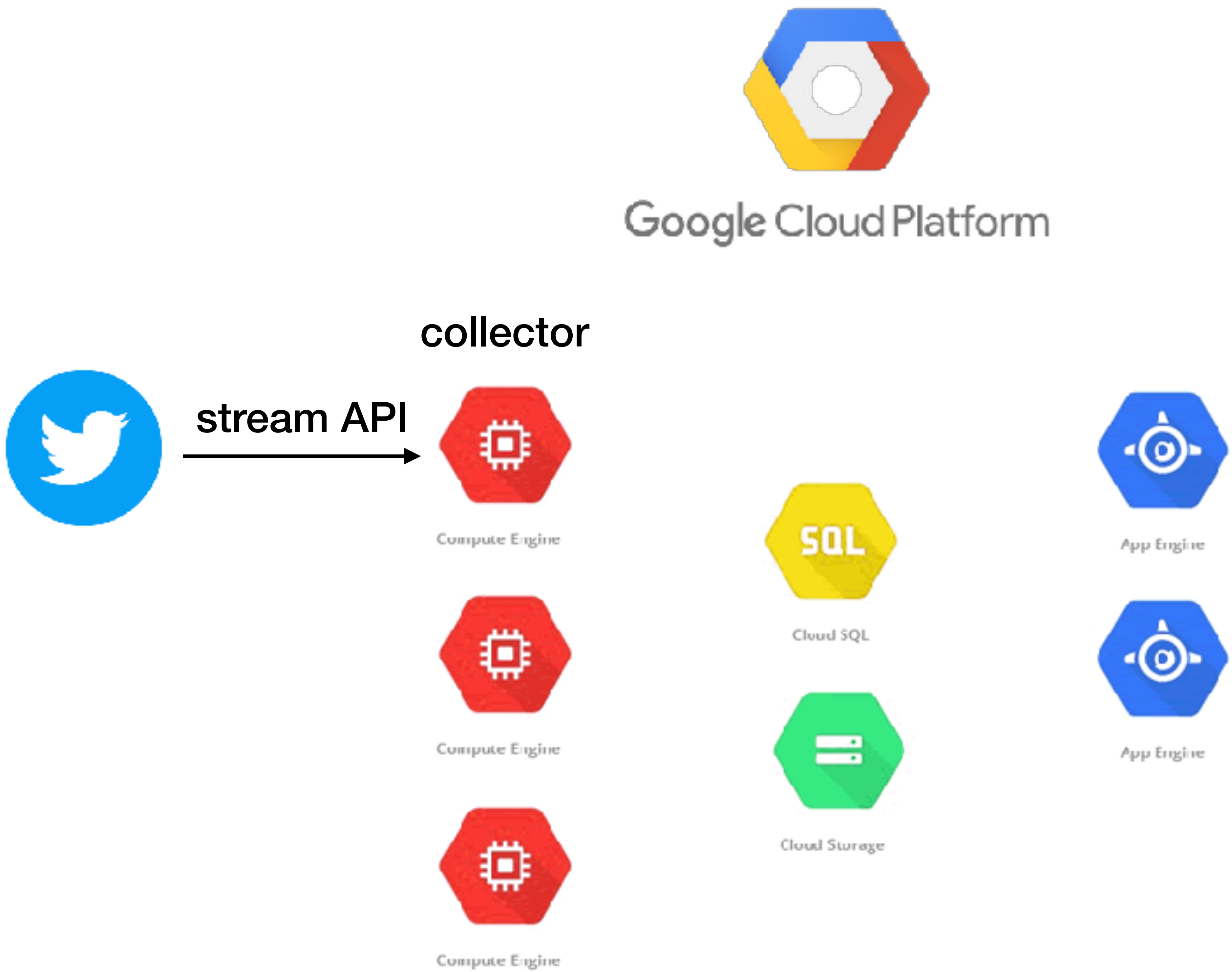
username
number of tweets
number of followers
...

(for every tweet/ retweet)

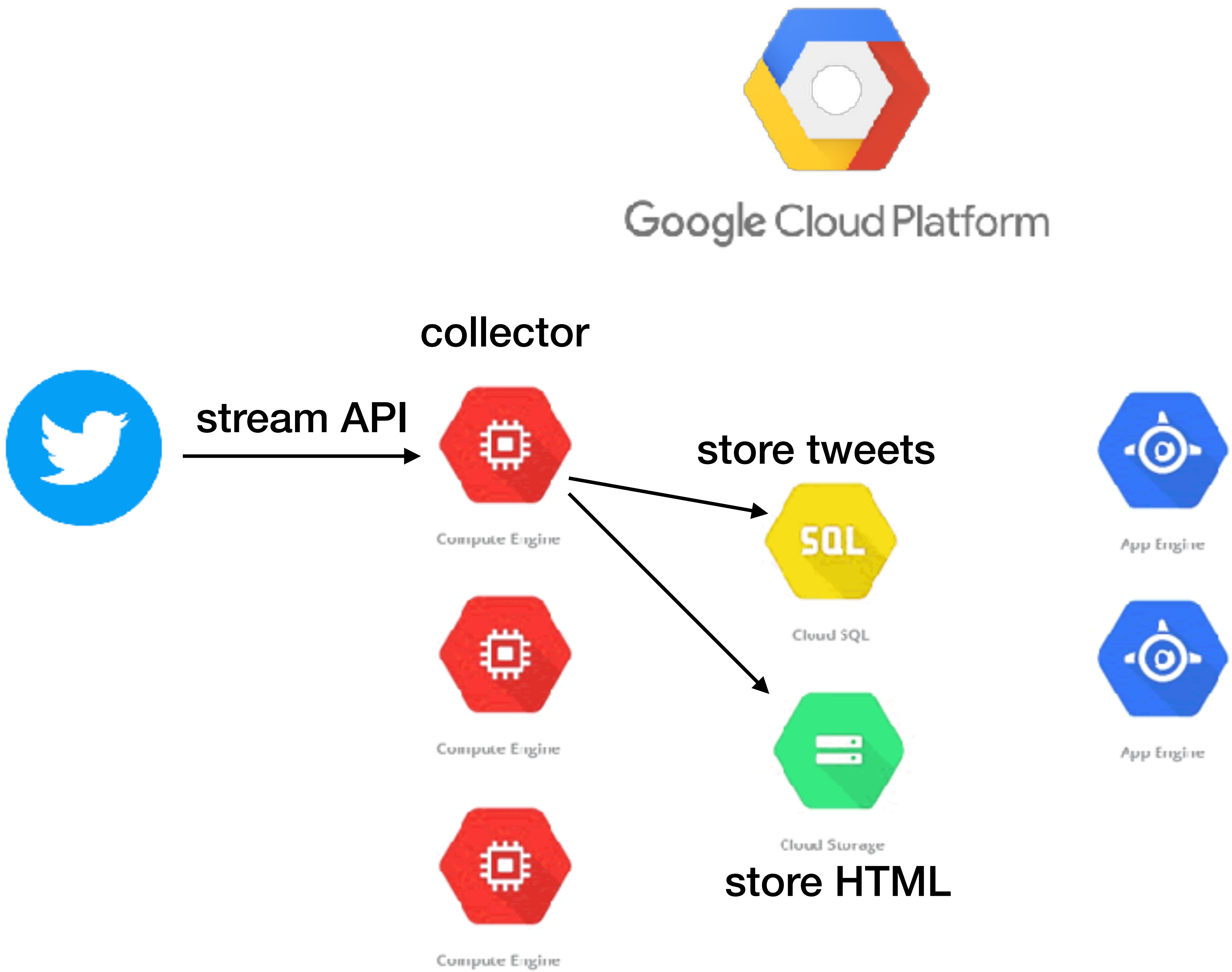
Data Collection - time



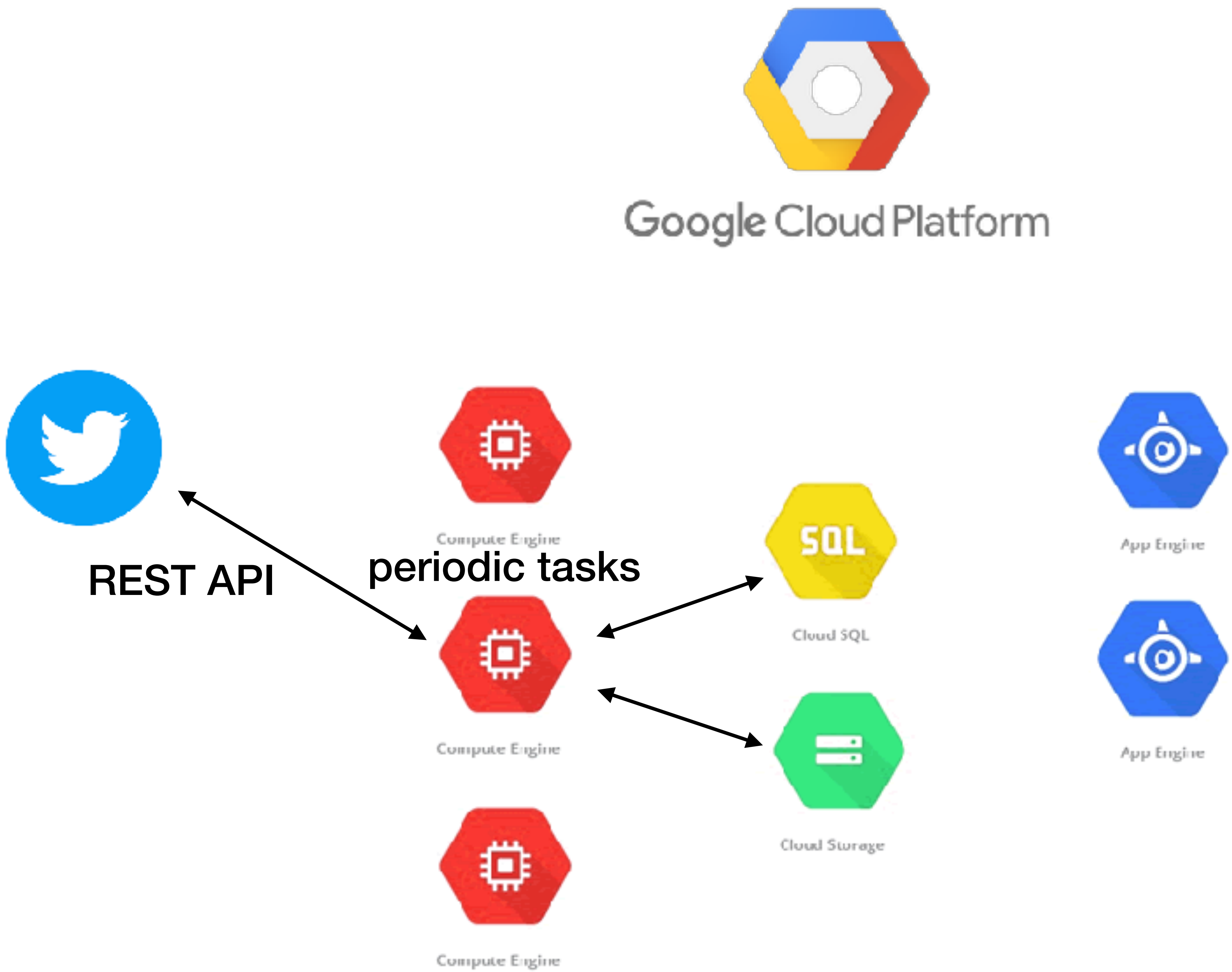
Data Collection - Architecture



Data Collection - Architecture

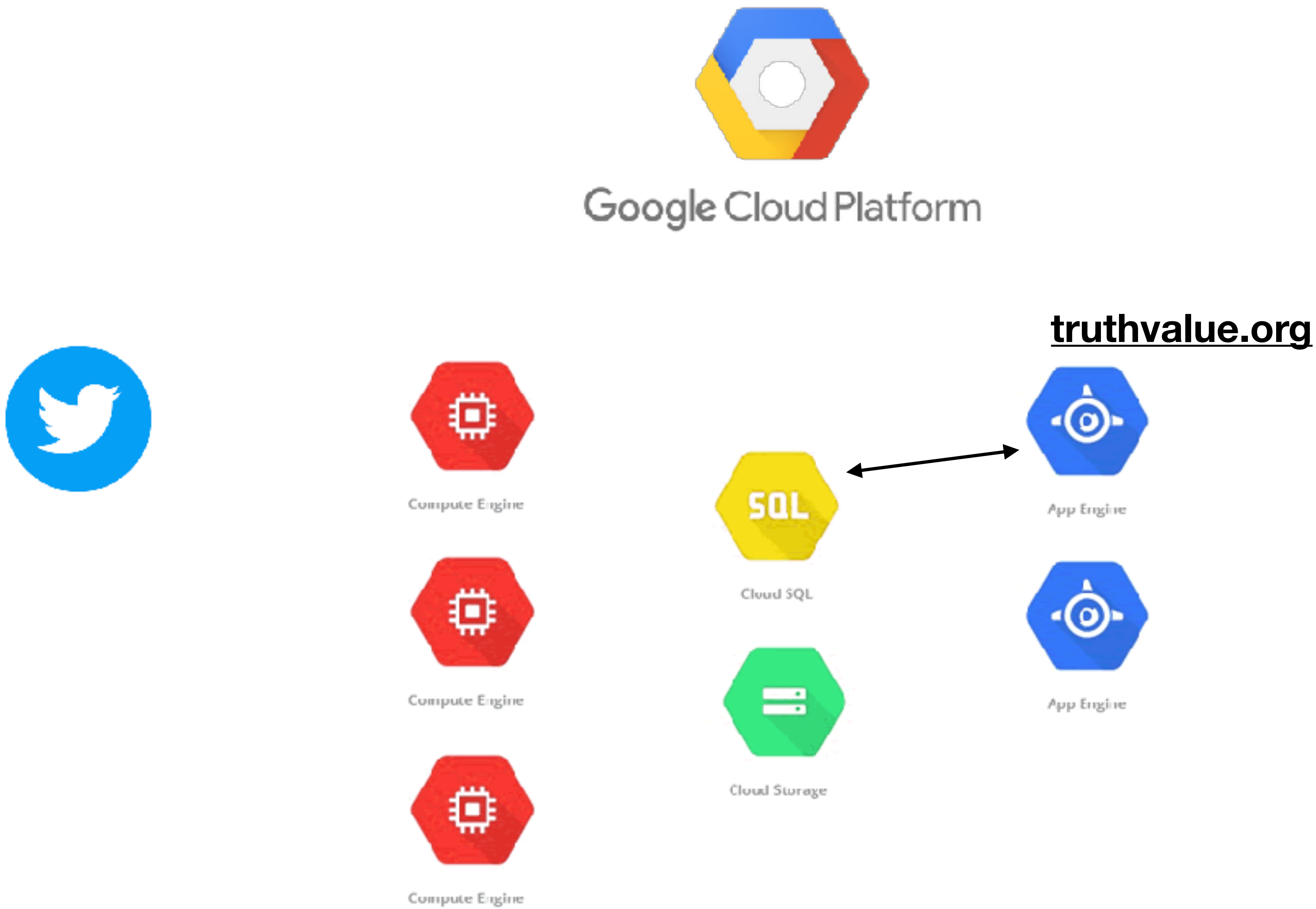


Data Collection - Architecture

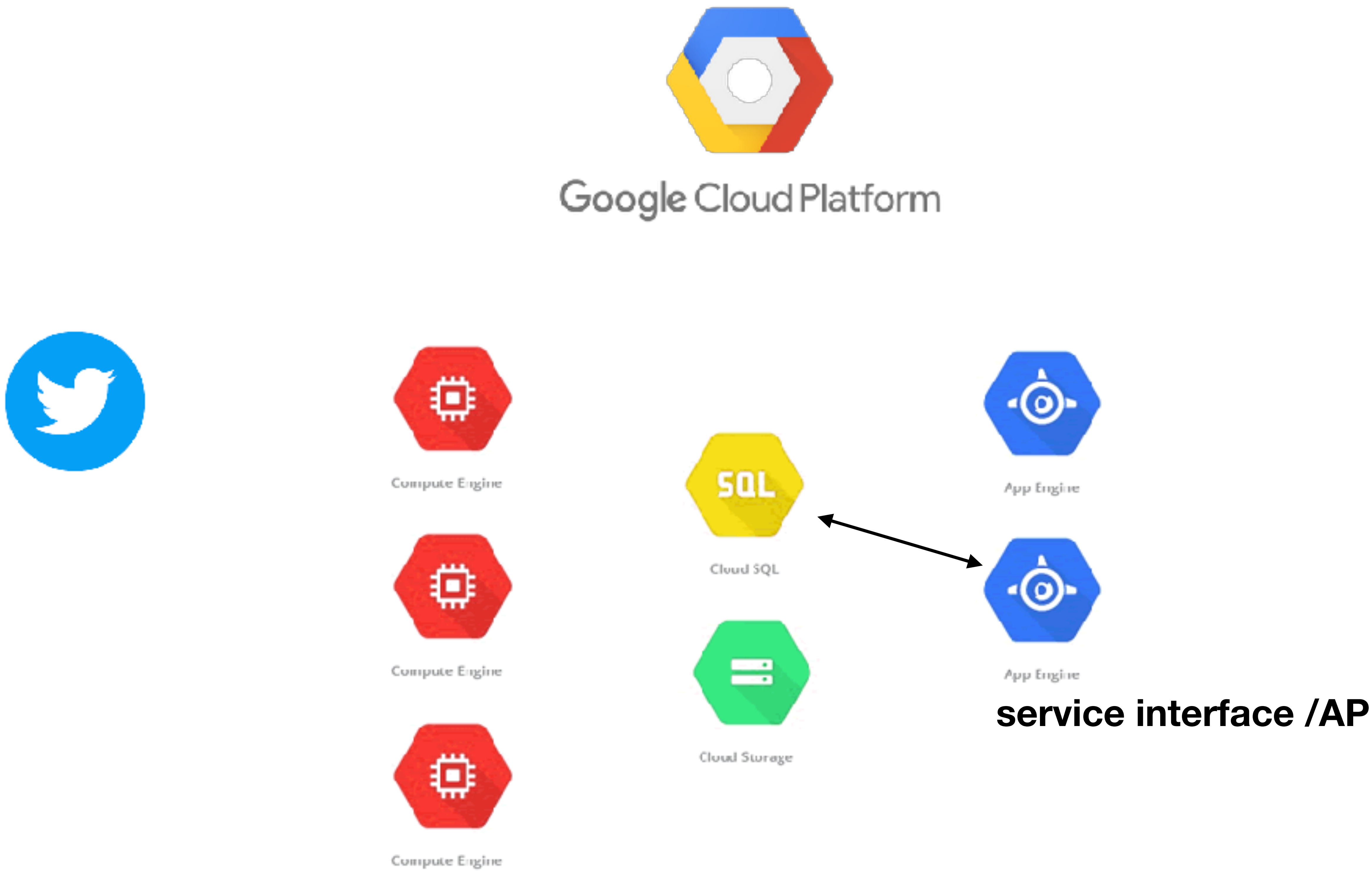


Data Collection - Architecture

Data Collection - Architecture



Data Collection - Architecture



What/Who do we follow?

Mainstream news: including ABC News, Breitbart, BuzzFeed, CBS News, Channel 7 News, CNN, Fox News, MSNBC, NBC News, The Huffington Post, The Economist, The Guardian, The Hill, The Onion, The New York Post, The New York Times, The Times, The US Herald, The Washing- ton Post, USA Today, US News

The Associated Press and Reuters

ArXiV, Nature, and Science Magazine

Politifact and Snopes

News from 160 selected users

Low-quality news from our Ground Truth (OpenSources & MetaCert)

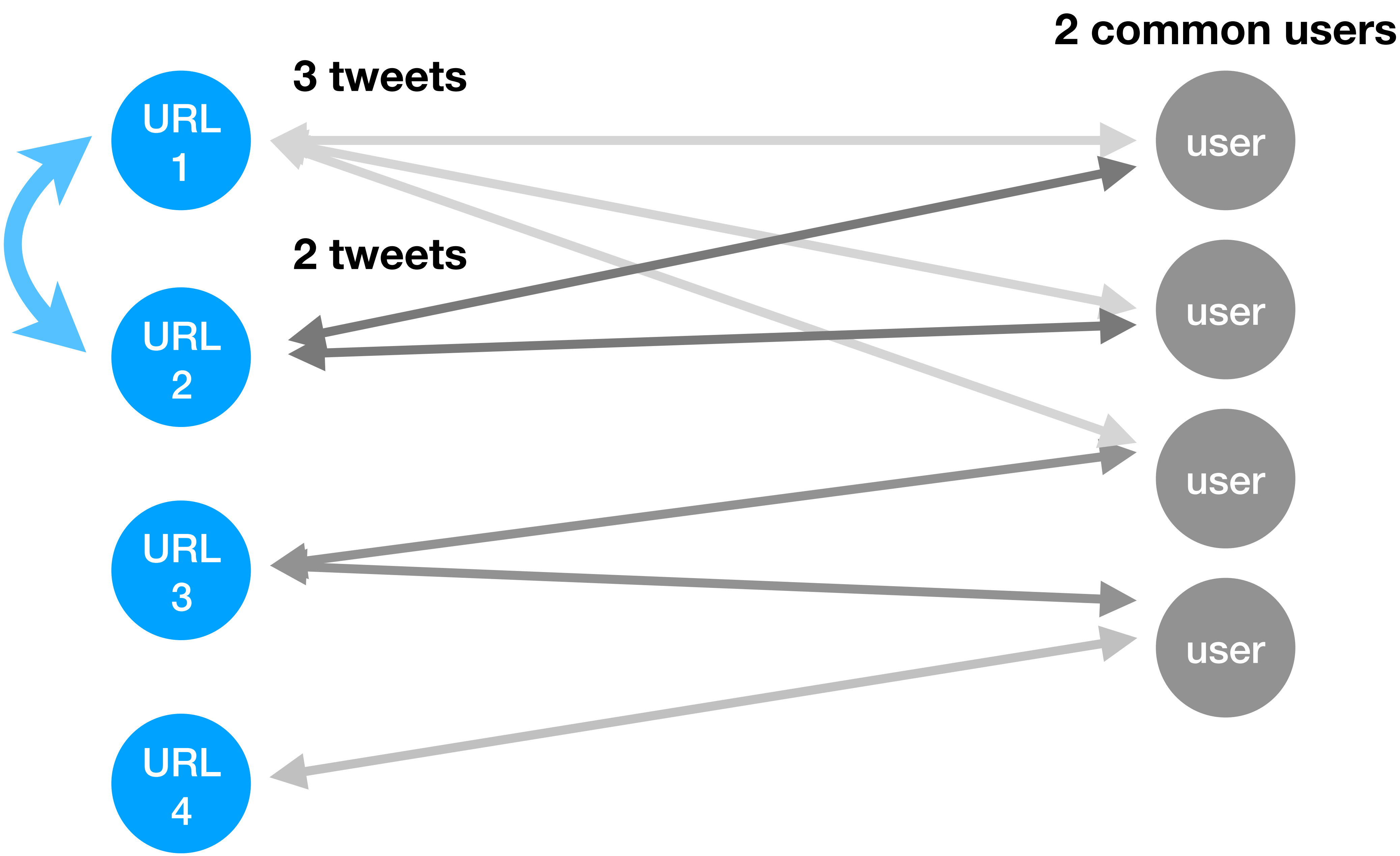
Data Collection - Results

Site	%	Site	%
youtube.com	4.319	wordpress.com	0.754
nytimes.com	3.145	nypost.com	0.625
theguardian.com	2.904	thehill.com	0.619
huffingtonpost.com	1.964	latimes.com	0.616
washingtonpost.com	1.944	breitbart.com	0.609
arxiv.org	1.585	cbsnews.com	0.563
usatoday.com	1.504	reuters.com	0.426
indiatimes.com	1.458	reddit.com	0.388
foxnews.com	1.262	dailycaller.com	0.367
blogspot.com	1.202	newsmax.com	0.336

Table 1: The 20 news sites with the most URLs in our dataset in the period from September 1 to November 30, 2017.

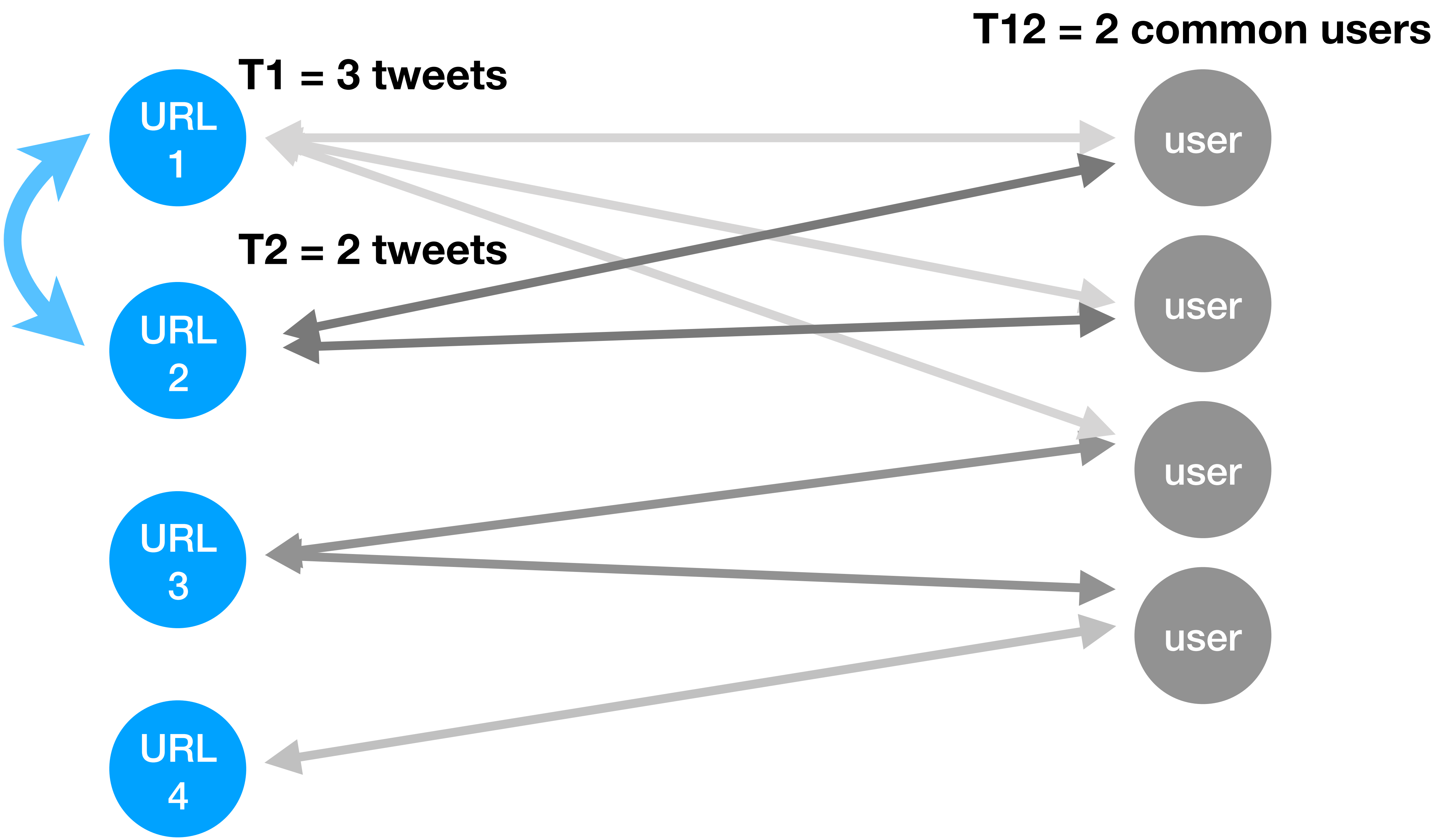
Unsupervised Learning

Master Graph



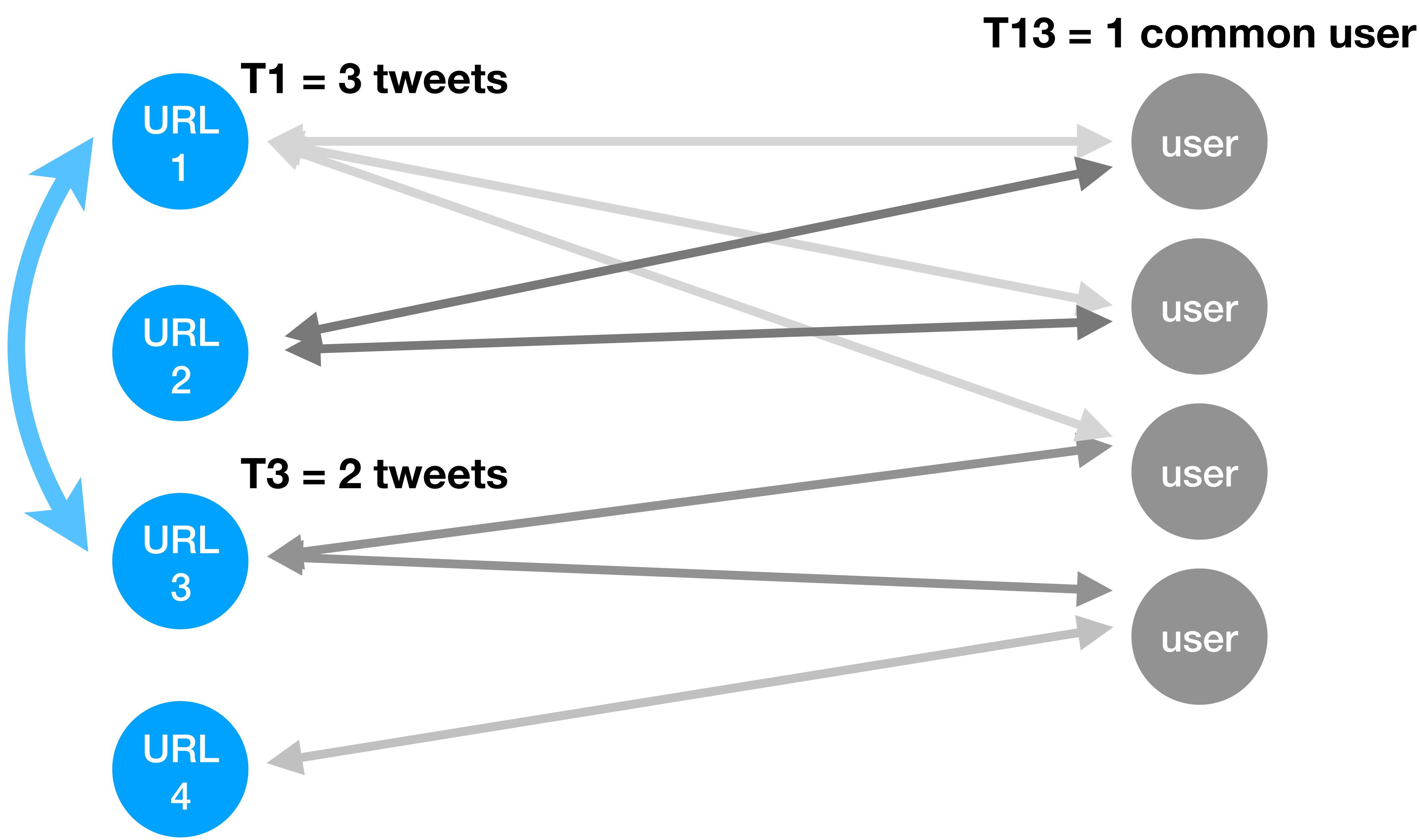
Master Graph

Similarity(1,2) = T12 / sqrt(T1 * T2) = 0.81% similarity



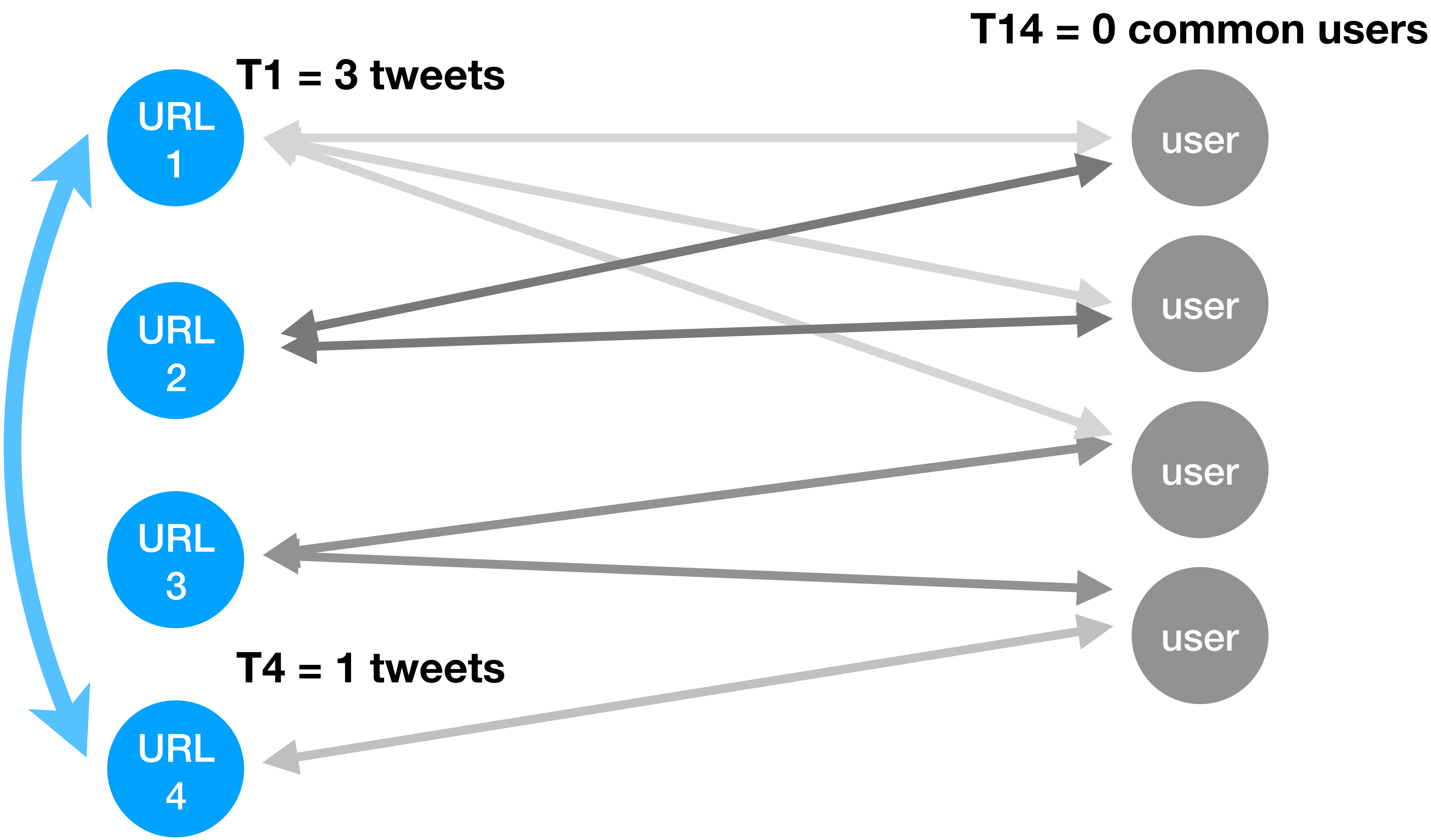
Master Graph

Similarity(1,3) = T13 / sqrt(T1 * T3) = 0.40% similarity



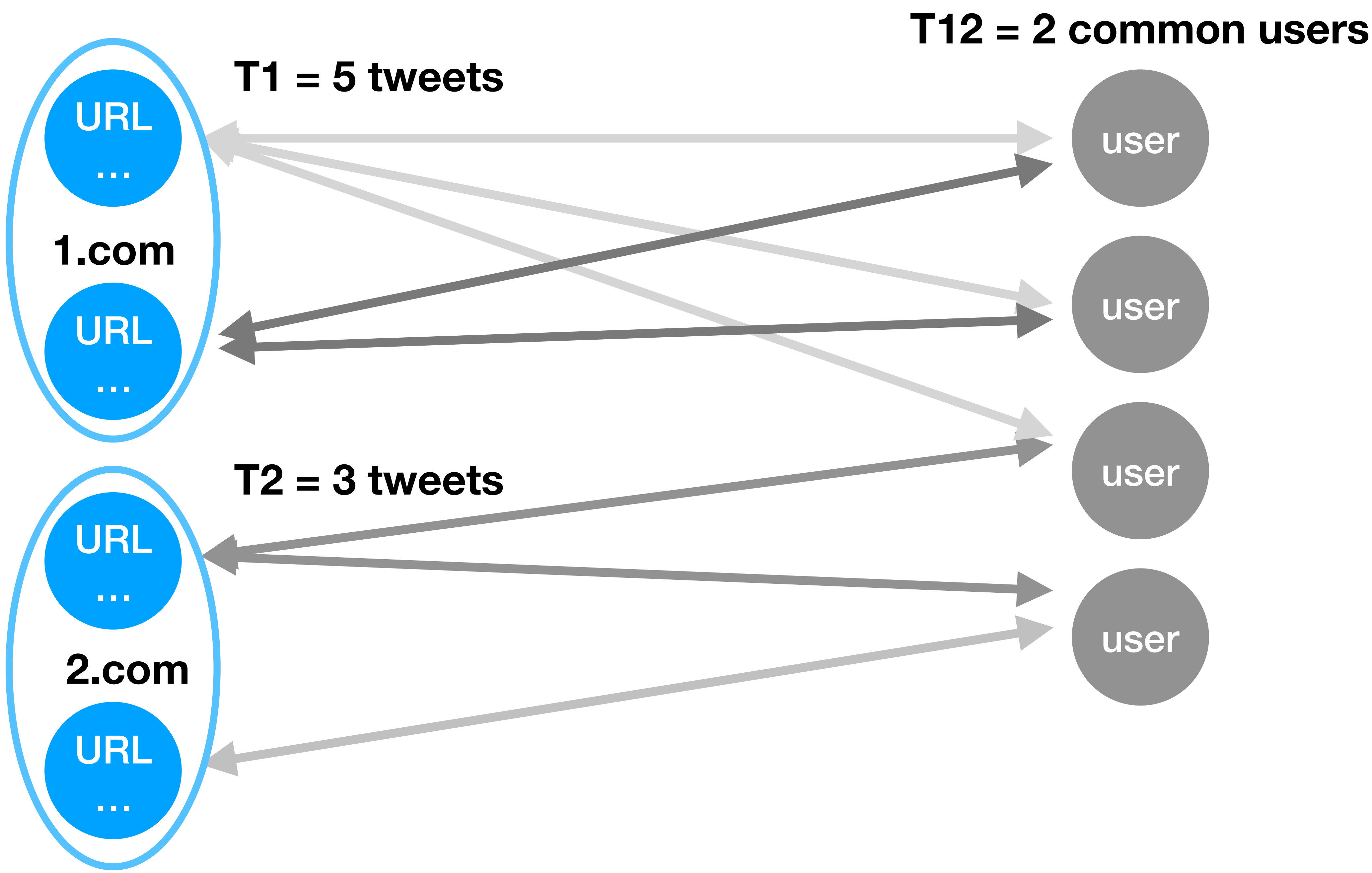
Master Graph

Similarity(1,4) = T14 / sqrt(T1 * T4) = 0% similarity

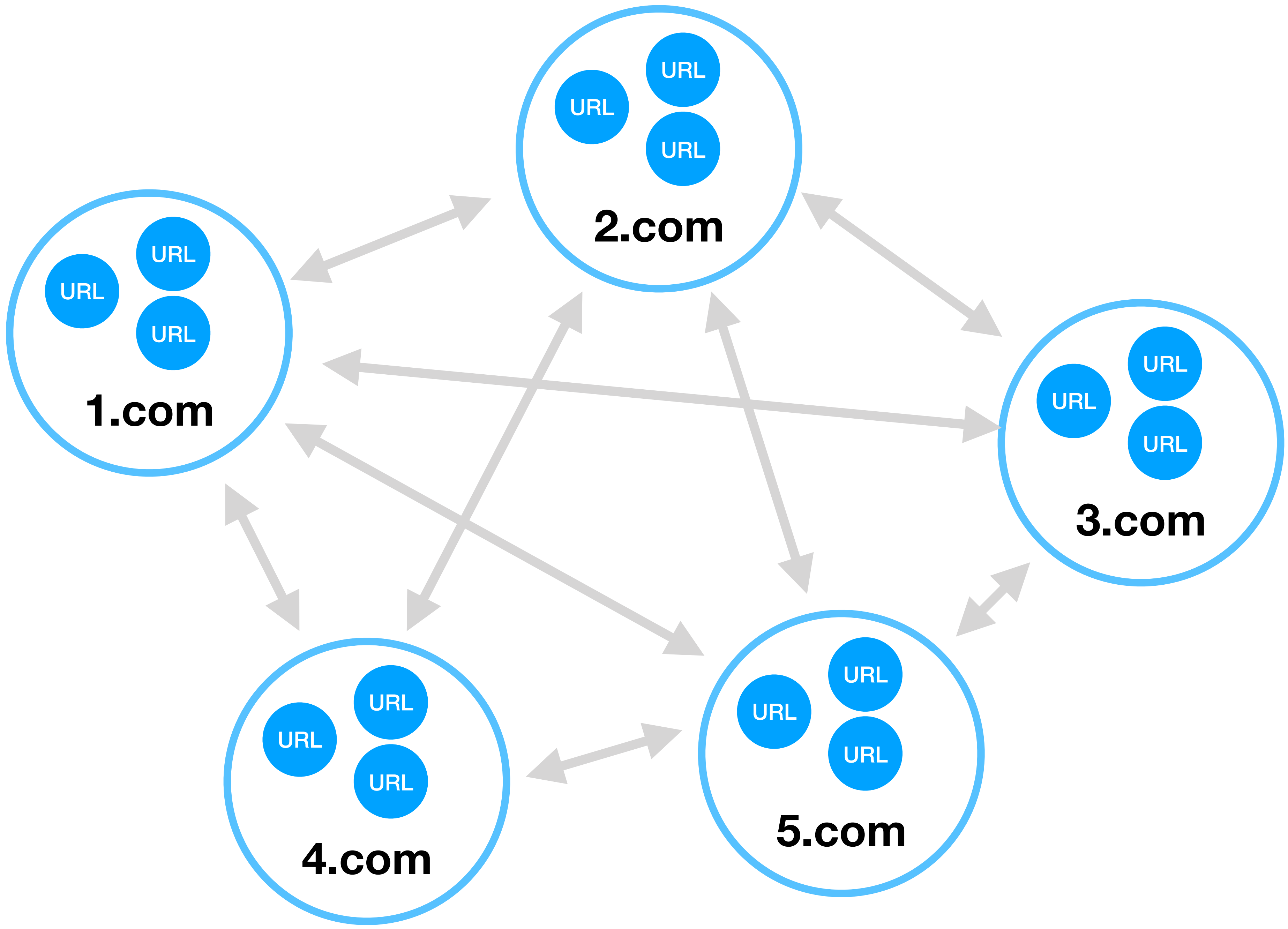


Domains Graph

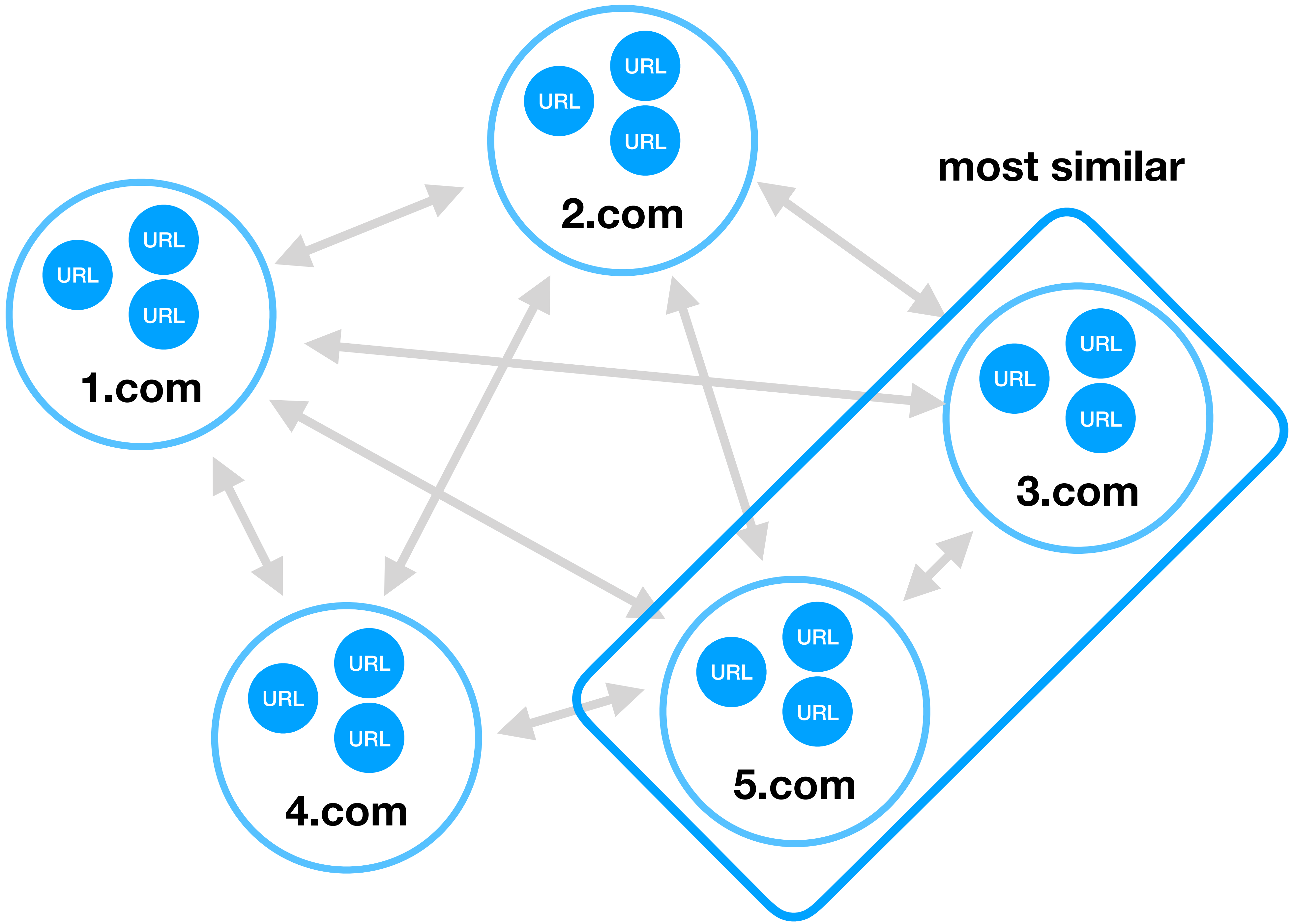
Similarity(1.com, 2.com) = T12 / sqrt(T1 * T2) = 51% similarity



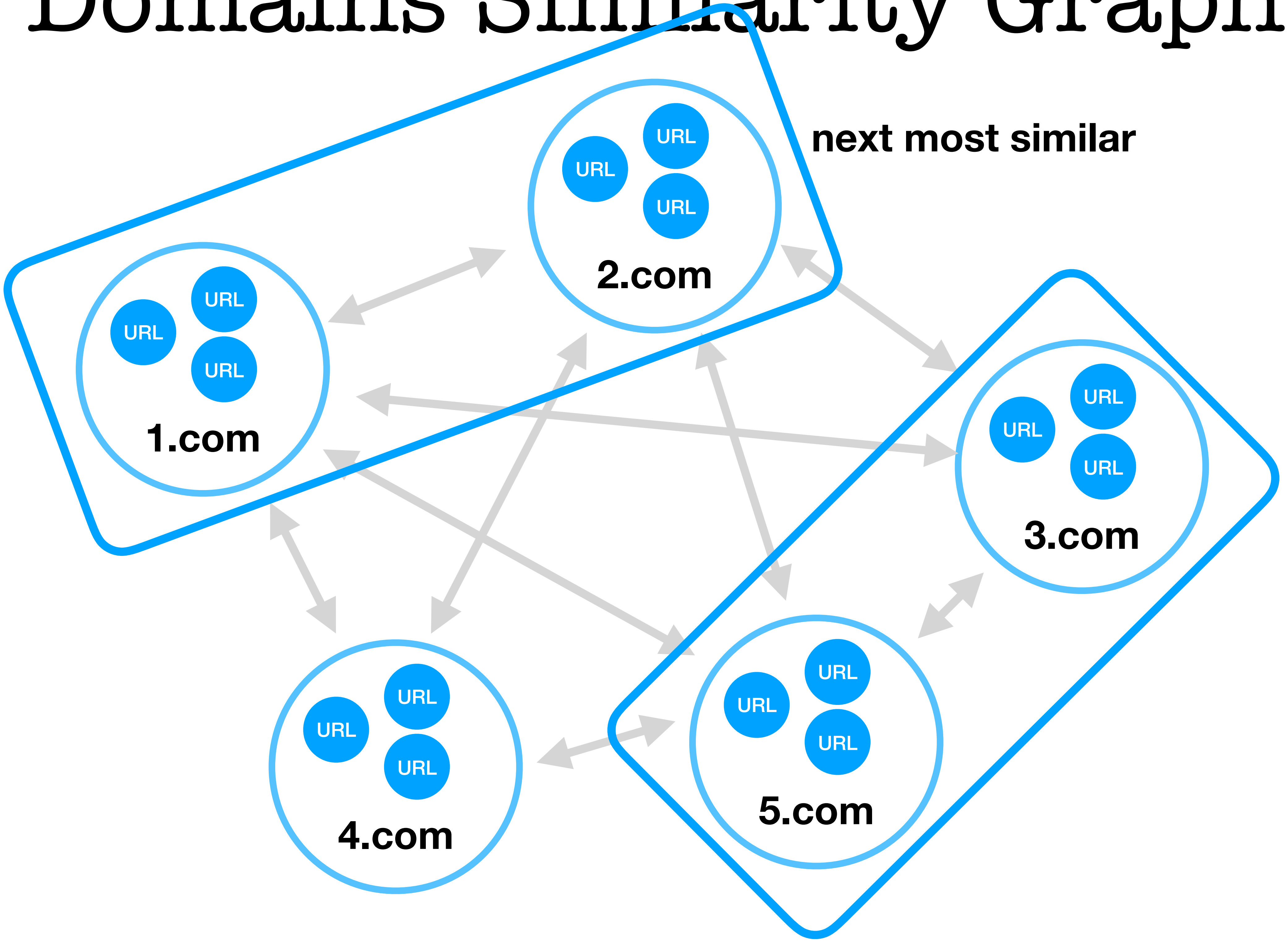
Domains Similarity Graph

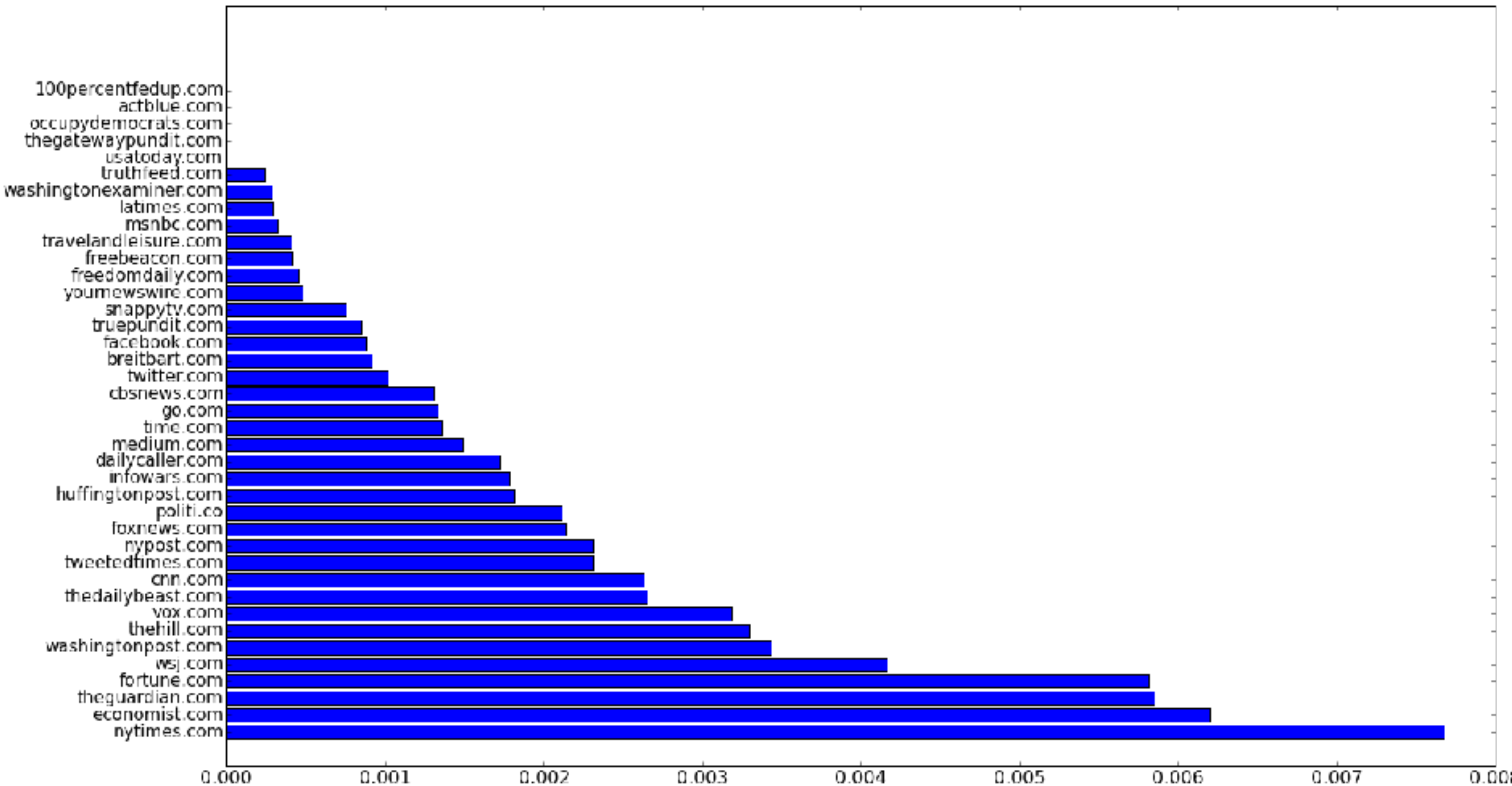


Domains Similarity Graph

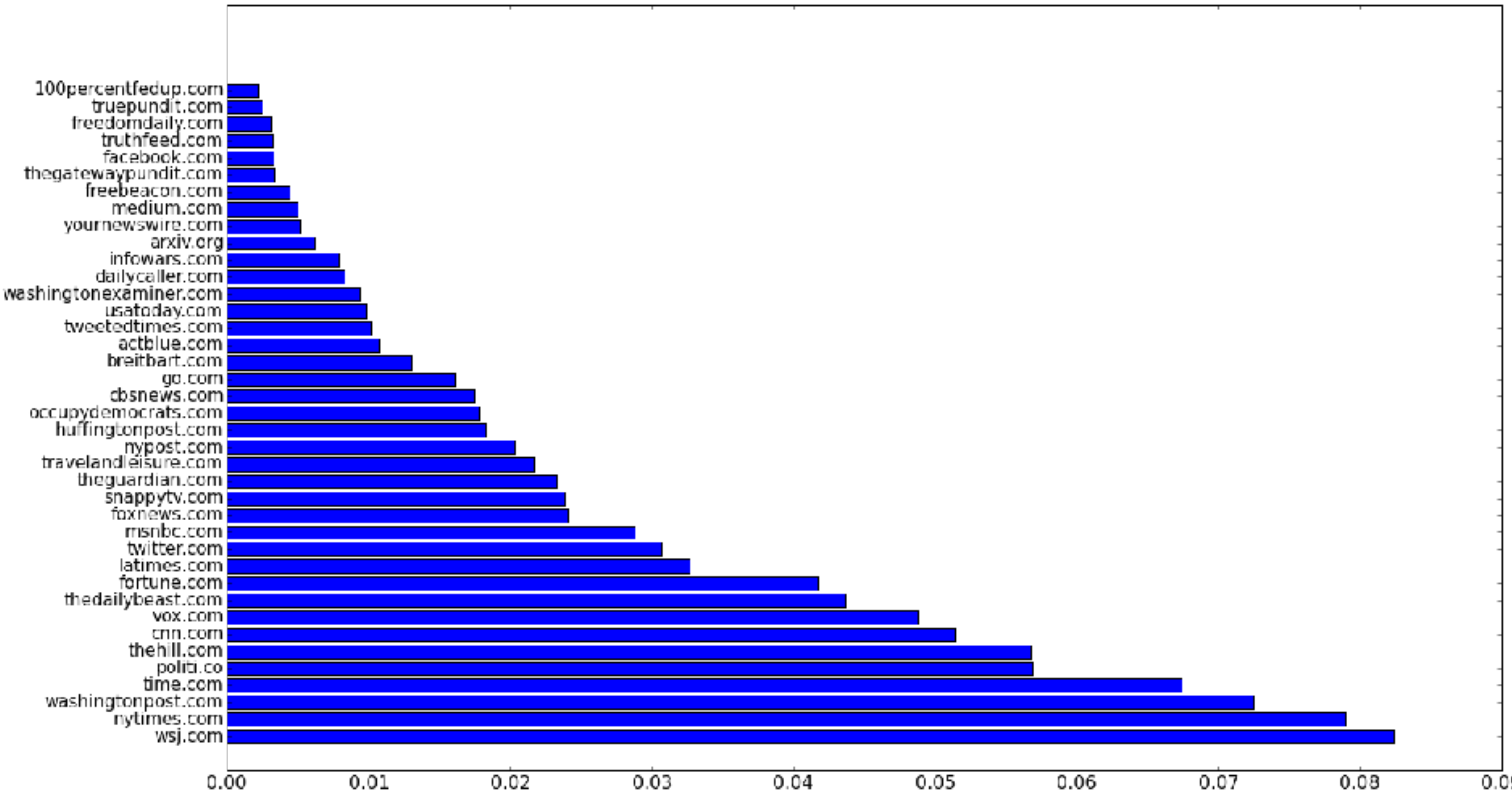


Domains Similarity Graph

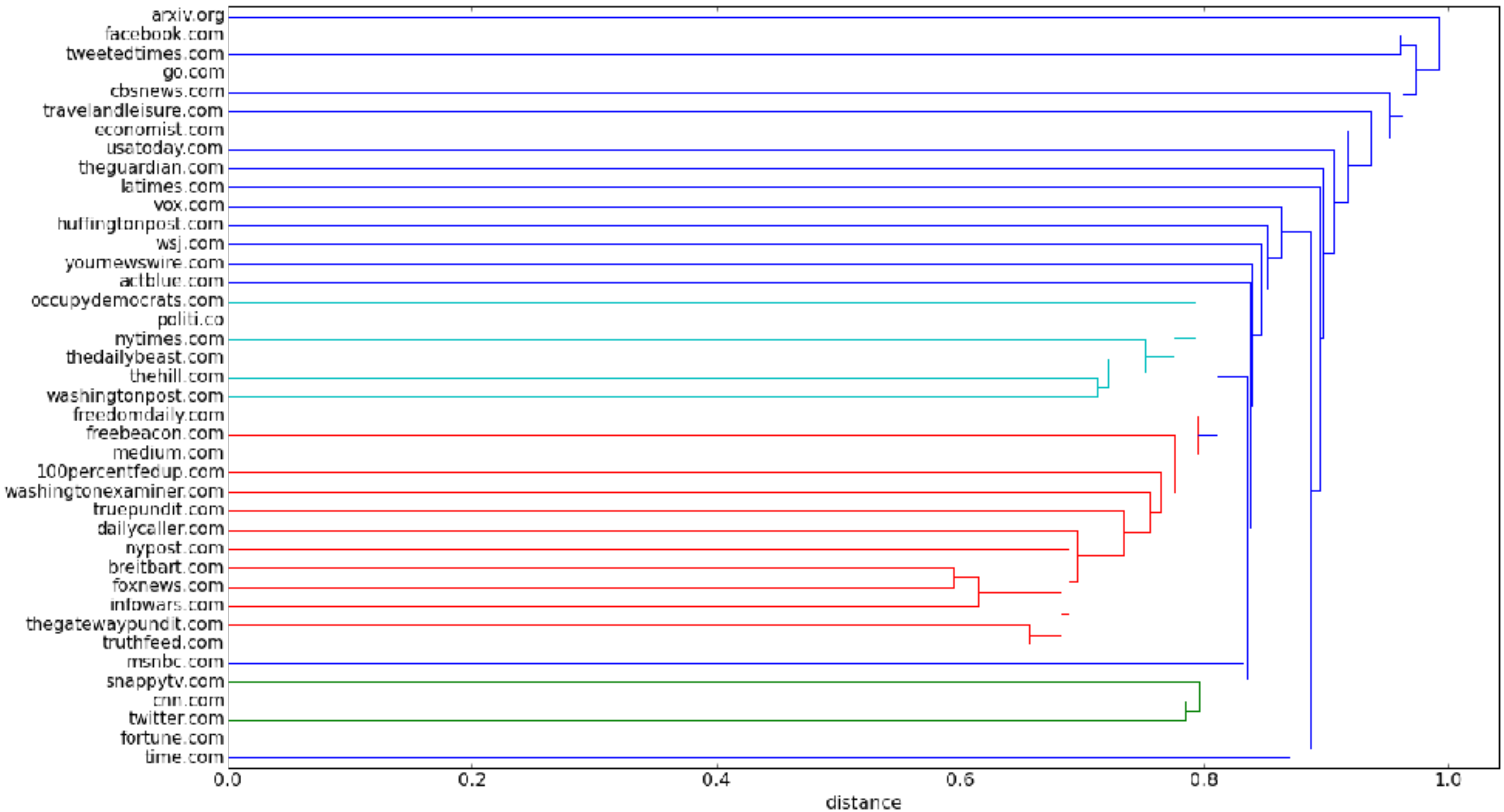




similarity with arXiv.org



similarity with economist.com



arxiv.org
facebook.com
tweetedtimes.com
go.com
cbsnews.com
travelandleisure.com
economist.com
usatoday.com
theguardian.com
latimes.com
vox.com
huffingtonpost.com
wsj.com
yournewswire.com
actblue.com
occupydemocrats.com
politi.co
nytimes.com
thedailybeast.com
thehill.com
washingtonpost.com
freedomdaily.com
freebeacon.com
medium.com
100percentfedup.com
washingtonexaminer.com
truepundit.com
dailycaller.com
nypost.com
breitbart.com
foxnews.com
infowars.com
thegatewaypundit.com
truthfeed.com
msnbc.com
snappytv.com
cnn.com
twitter.com
fortune.com
time.com
0.0
0.2
0.4
0.6
0.8
1.0
distance

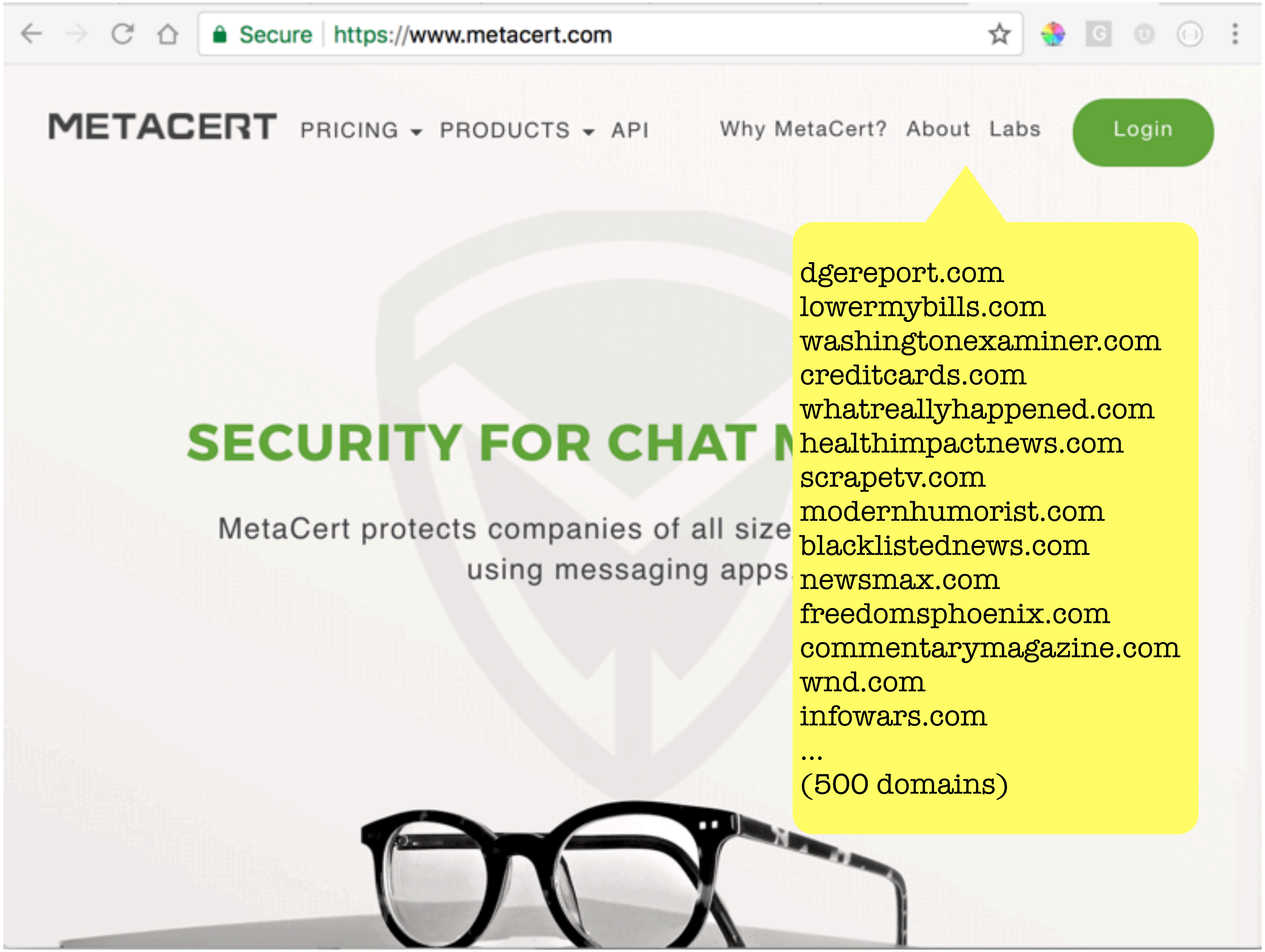
Supervised Learning

Ground Truth

what do we assume to be
certainly true or false?

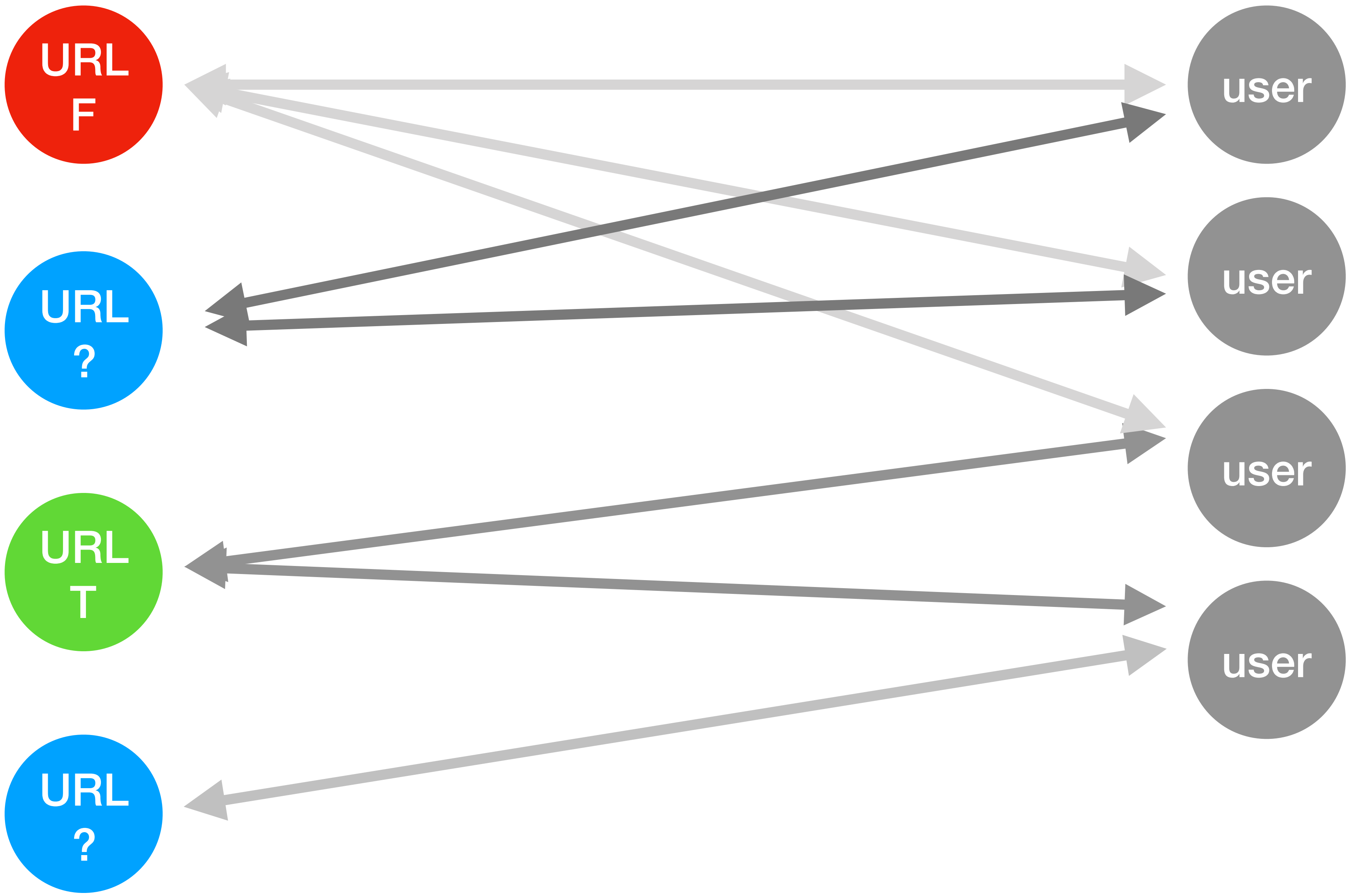
100percentfedup.com
lifenews.com
yournationnews.com
redflagnews.com
16wmpo.com
gaia.com
uschronicle.com
foodbabe.com
theusa-news.com
conspiracywire.com
thenewamerican.com
flashnewscorner.com
elitereaders.com
civictribune.com
...
(582 domains)

https://github.com/OpenSourcesGroup/opensources

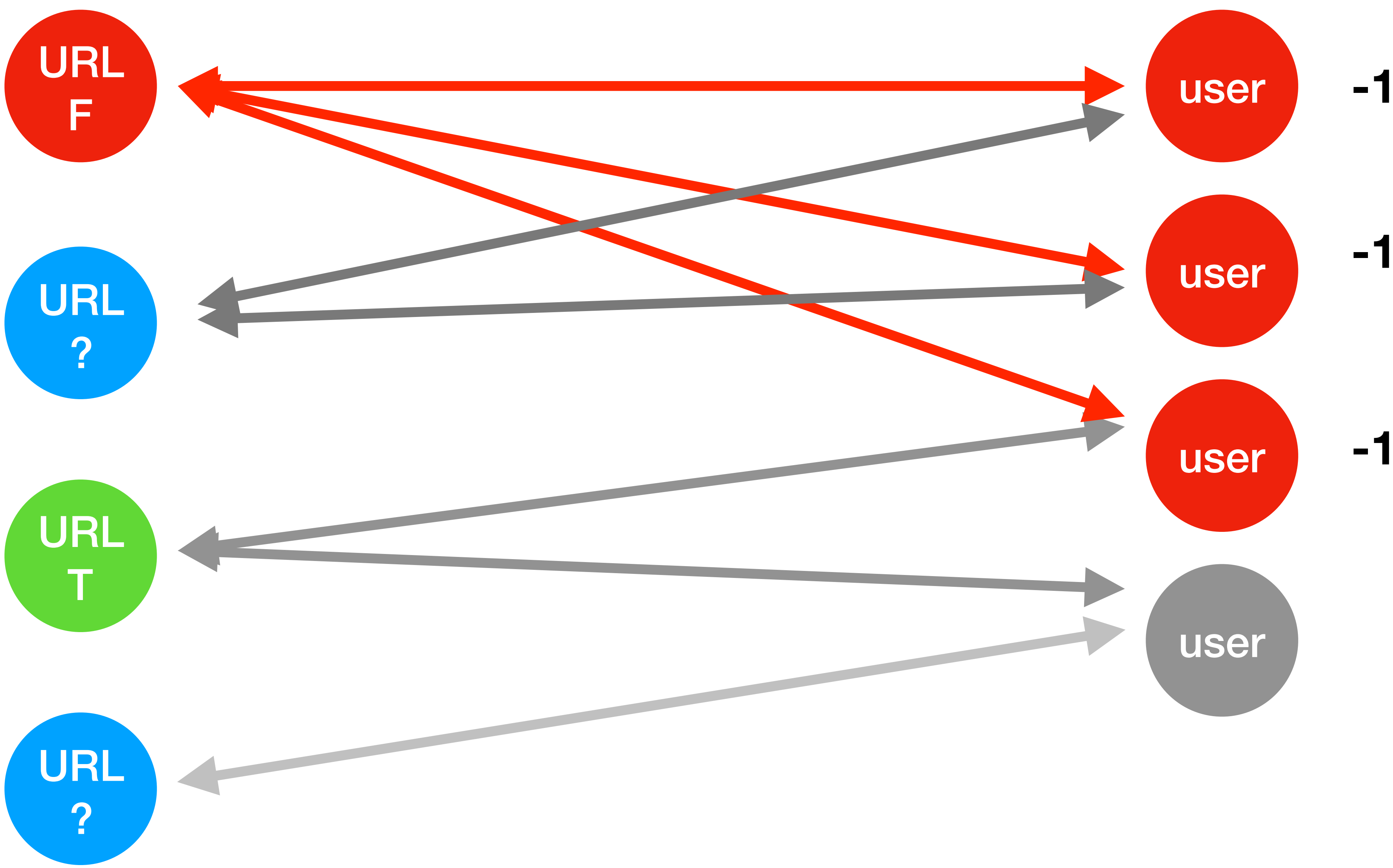

Secure
https://www.metacert.com
METACERT
PRICING
PRODUCTS
API
Why MetaCert?
About
Labs
Login
SECURITY FOR CHAT M
MetaCert protects companies of all size
using messaging apps.
dgereport.com
lowermybills.com
washingtonexaminer.com
creditcards.com
whatreallyhappened.com
healthimpactnews.com
scrapetv.com
modernhumorist.com
blacklistednews.com
newsmax.com
freedomsphoenix.com
commentarymagazine.com
wnd.com
infowars.com
...
(500 domains)

Supervised Learning
Harmonic

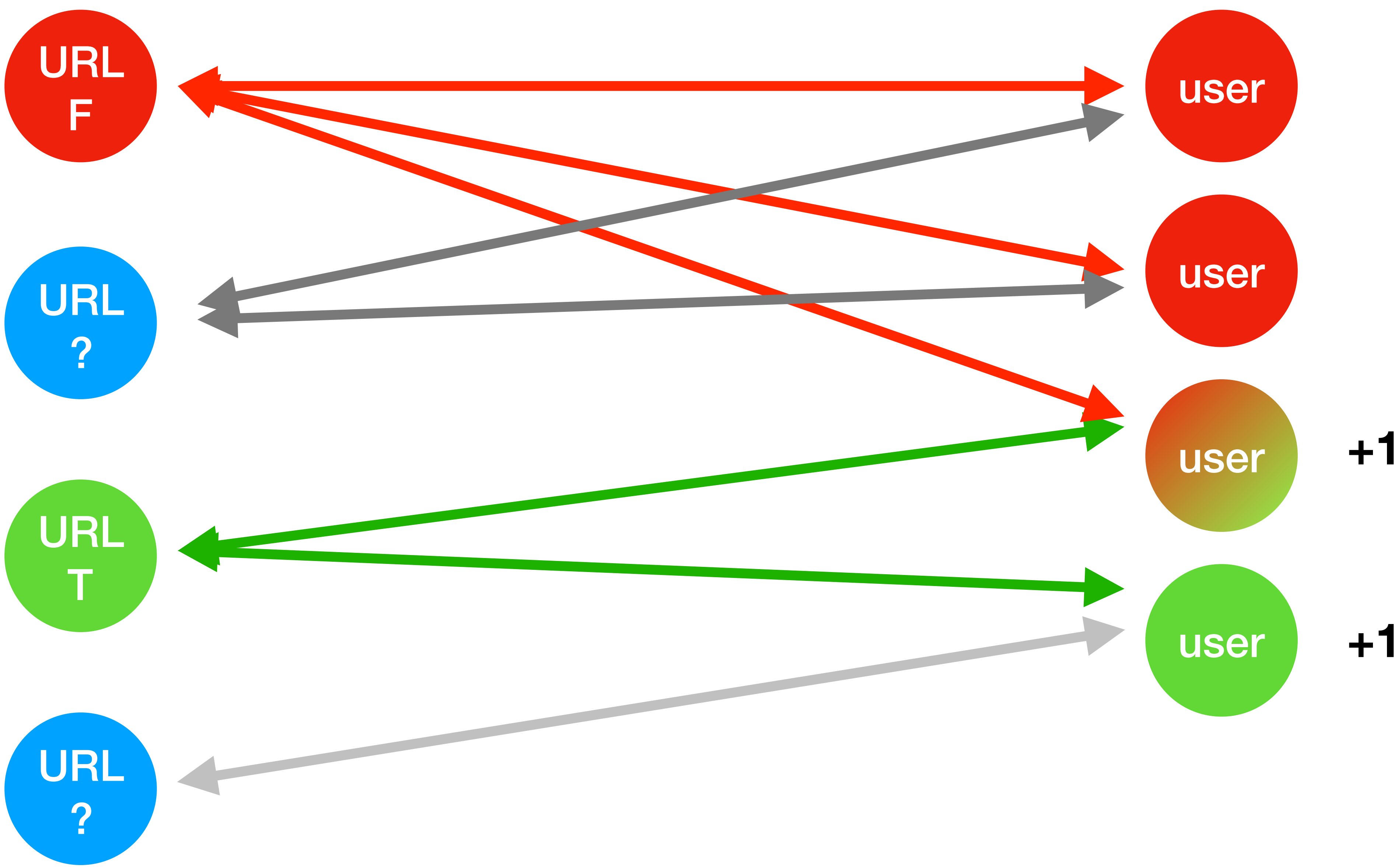
Master Graph



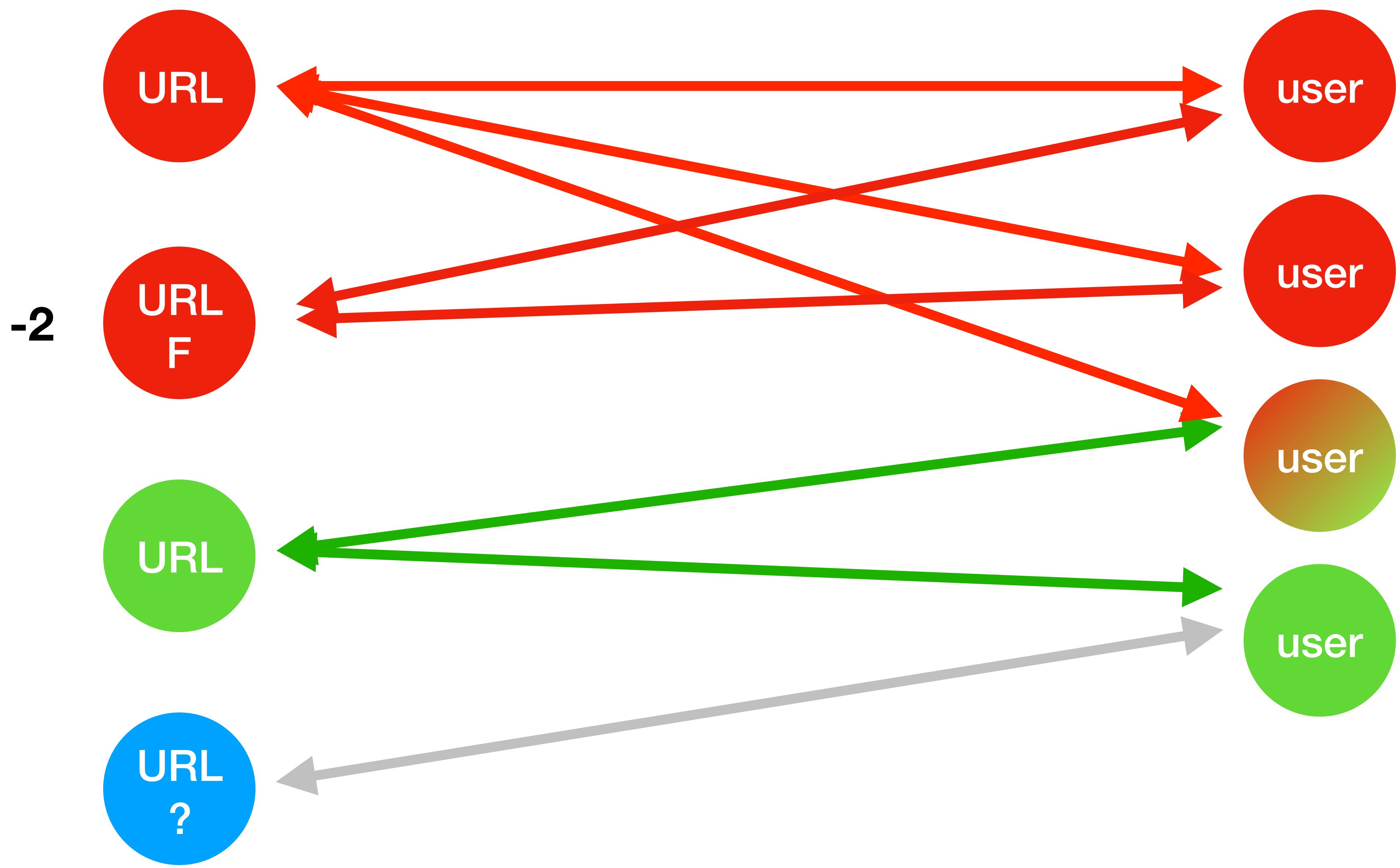
Information Propagation



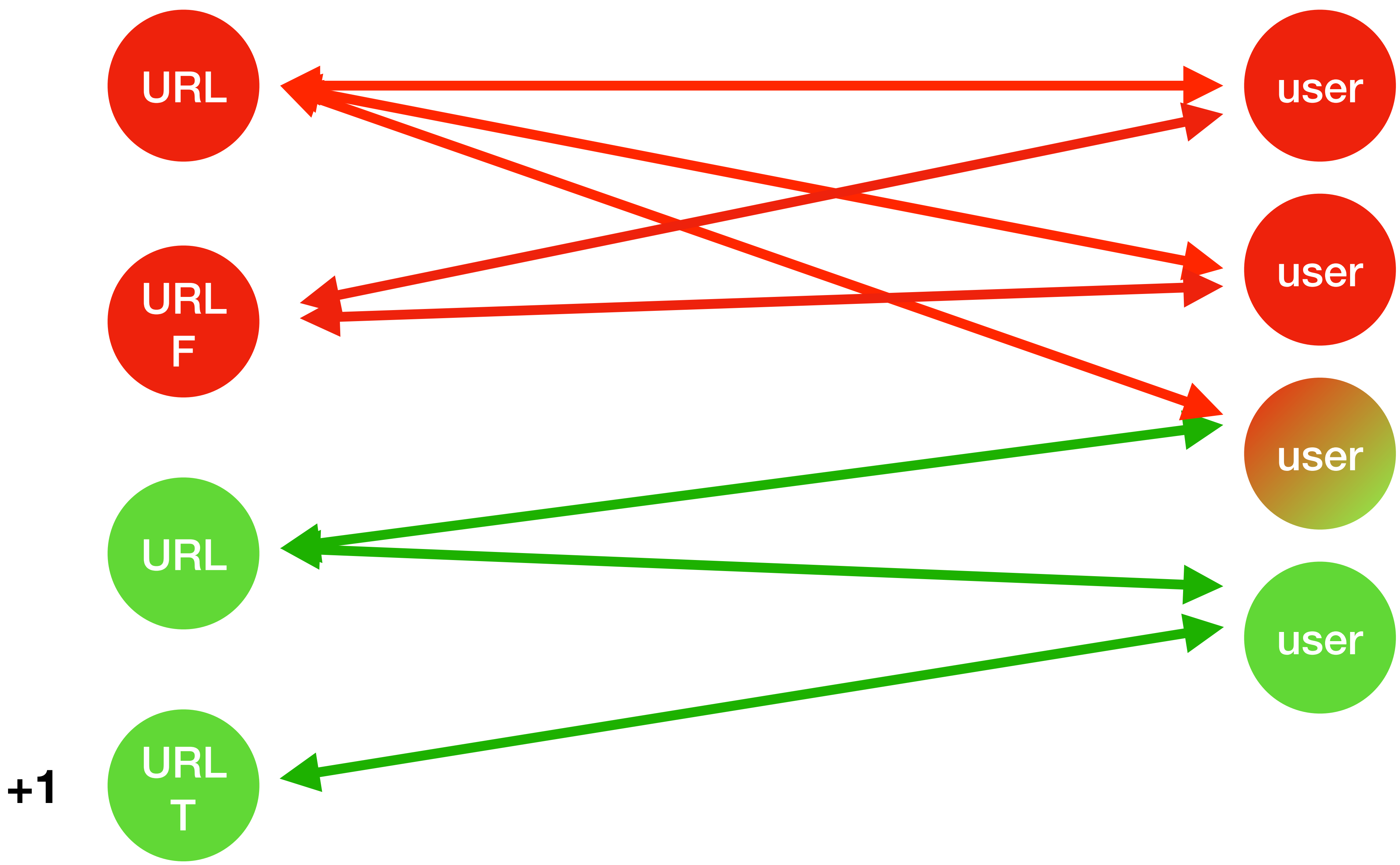
Information Propagation



Information Propagation



Information Propagation



Modes of Operation

Fixed-point

Loop and propagate reputation from URLs to users and back until convergence
Usually converges in few (4-8) iterations

Dynamic

Real time update when a new Tweet is discovered

Supervised Learning

Logistic Regression
Bag of Word Model
Topic Modeling

Bag or Words Model

Corpora

It was the best of times,
it was the worst of times,
it was the age of wisdom,
it was the age of foolishness,

Bag of Words
it, was, the, best, of, times

Vocabulary

- "it"
- "was"
- "the"
- "best"
- "of"
- "times"
- "worst"
- "age"
- "wisdom"
- "foolishness"

Features

```
1 "it was the worst of times" = [1, 1, 1, 0, 1, 1, 1, 0, 0, 0]
2 "it was the age of wisdom" = [1, 1, 1, 0, 1, 0, 0, 1, 1, 0]
3 "it was the age of foolishness" = [1, 1, 1, 0, 1, 0, 0, 1, 0, 1]
```

10 features

Bag or Words Model

Corpora

it was the best of times,
it was the worst of times,
it was the age of wisdom,
it was the age of foolishness,

Bag of Words
it, was, the, best, of, times

Vocabulary

- "it"
- "was"
- "the"
- "best"
- "of"
- "times"
- "worst"
- "age"
- "wisdom"
- "foolishness"

filter
TF-IDF

- ~~"it"~~
- ~~"was"~~
- ~~"the"~~
- ~~"best"~~
- ~~"of"~~
- "times"
- "worst"
- "age"
- "wisdom"
- "foolishness"

frequency = 1
weight = 1.0

frequency = 1/2
weight = 2.0

frequency = 1/4
weight = 4.0

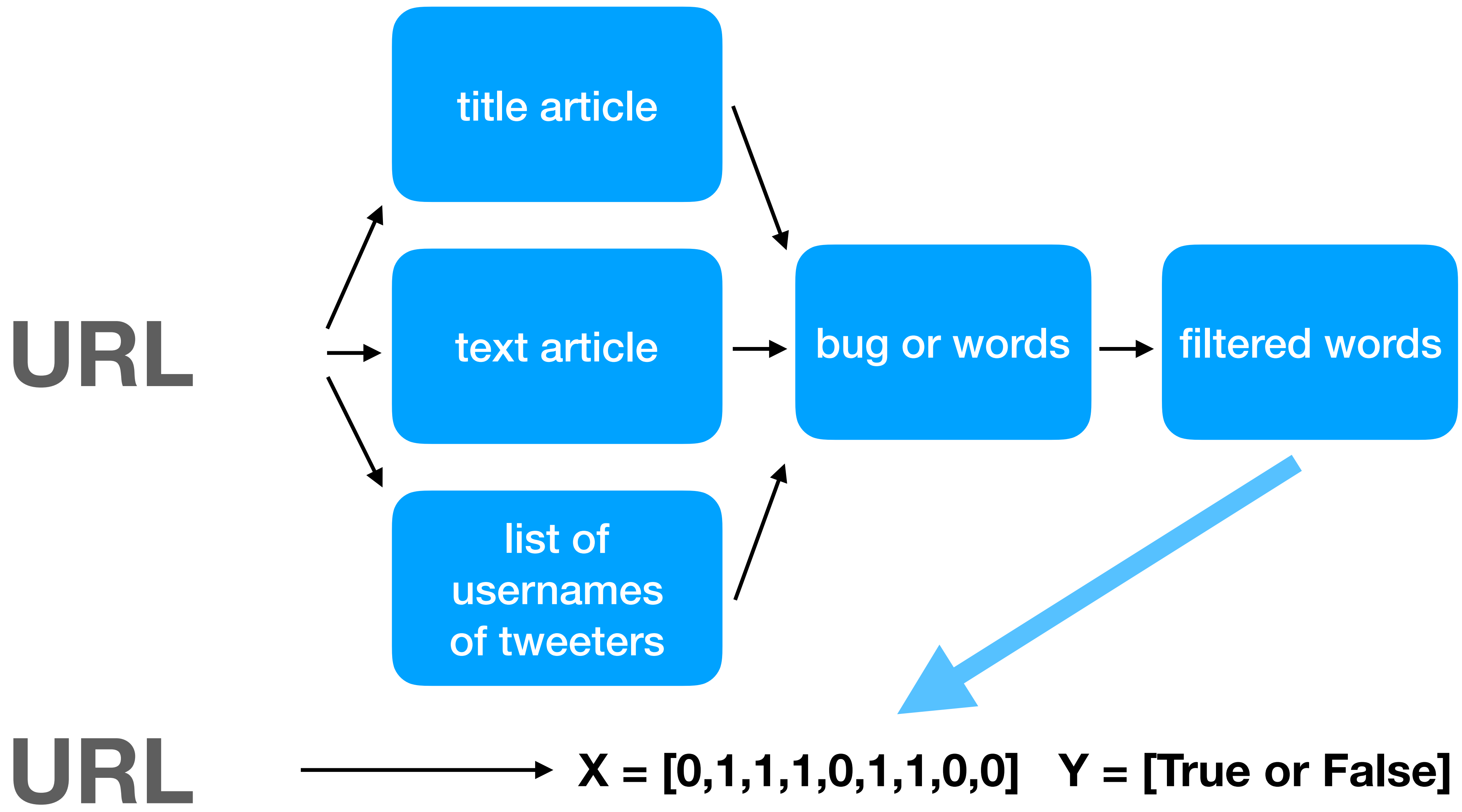
Features

"it was the worst of times" = [~~1, 1, 1, 0, 1~~, 1, 1, 0, 0, 0]
"it was the age of wisdom" = [~~1, 1, 1, 0, 1~~, 0, 0, 1, 1, 0]
"it was the age of foolishness" = [~~1, 1, 1, 0, 1~~, 0, 0, 1, 0, 1]

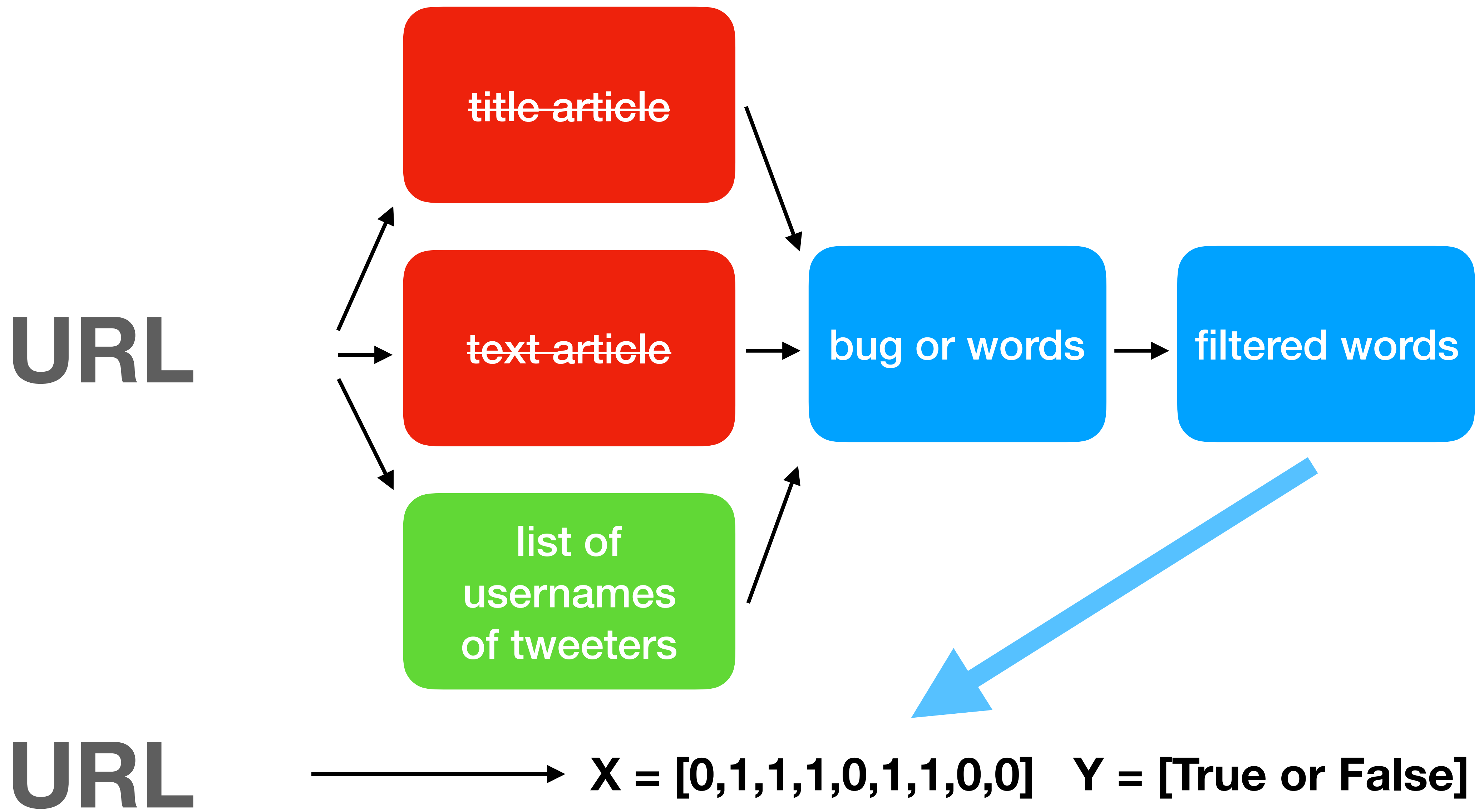
5 features

TF-IDF = **Term Frequency** Inverse Document Frequency

Bag or Words Model for News



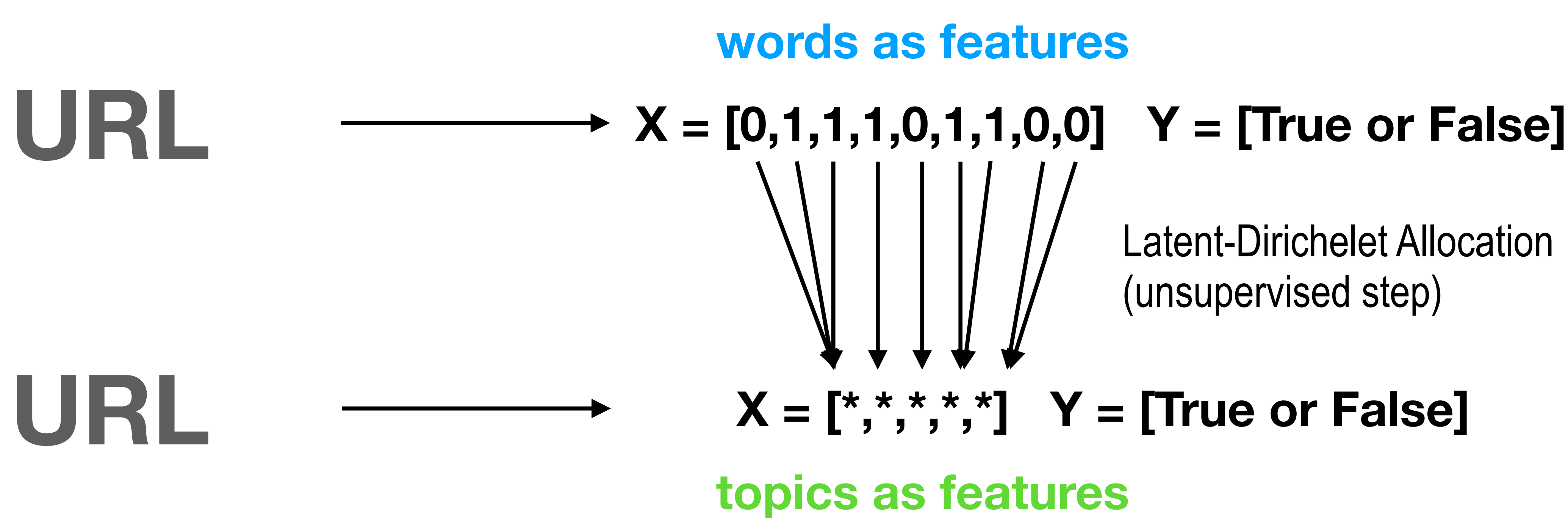
Bag or Words Model for News



Bag or Words Model Limitation

Problem: A limitation of the logistic-based classification model is that we can assign coefficients only to users and words that we have seen in our ground truth.

Solution: Topic-based models extract the common behavioral similarities among users, summarizing them in a number of topics that each user is likely to share. We then train using topics as features. (“topics” not in typical english meaning)



Training vs Testing

Dataset	URLs	Tweets
Training dataset	787,601	14,587,984
Min-2 training dataset	275,400	14,075,783
Testing dataset	607,299	7,967,170

Table 2: Sizes of training and testing datasets for logistic-regression based classifiers.

	Number of URLs			
	Opensources	Metacert	Common	Total
Train	7,069	7,032	4,876	144,137
Test	2,664	2,331	1,810	121,460

Table 4: Number of URLs in the Opensources and Metacert lists that appear in our training and testing sets for topic modeling. The different proportion of URLs that belong to the Opensources and Metacert sets, compared to the total URLs, depends on the fact that the training set cosists only of URLs that were shared at least twice.

Results

Confusion Matrix

Confusion Matrix

	Positive	negative
Predicted positive	*TP*	*FP*
Predicted negative	*FN*	*TN*

$$\text{Recall} = \frac{TP}{TP+FN}$$

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\text{True Positive Rate} = \frac{TP}{TP+FN}$$

$$\text{False Positive Rate} = \frac{FP}{FP+TN}$$

Recall / Precision for Ground Truth

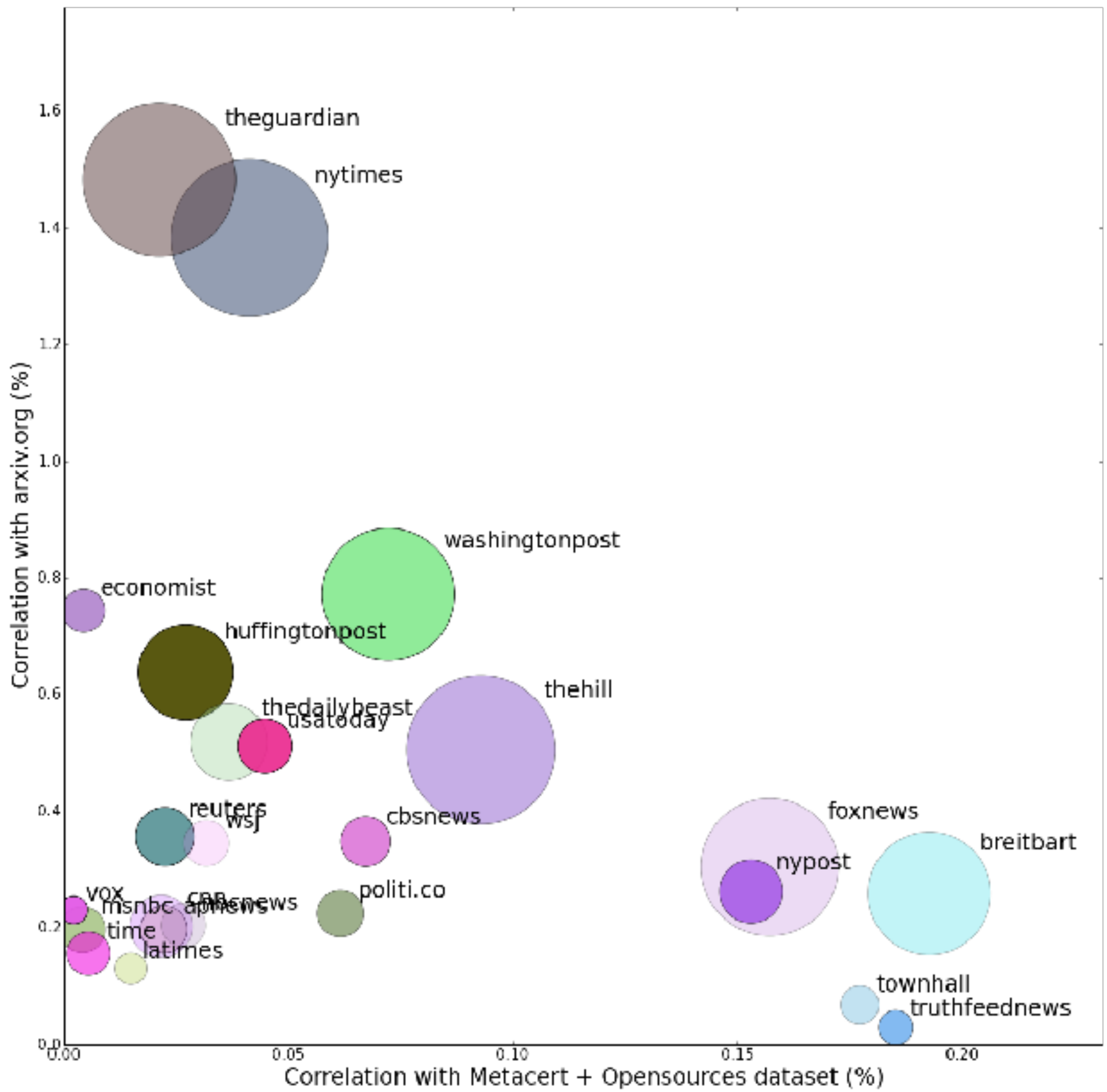
		Full Training			Min-2 Training				Harmonic	
		LR-U	LR-UT	LR-T	LR-U	LR-UT	LR-T	Topics	1x	4x
All URLs	Hoax recall:	57.64	61.14	57.25	46.84	53.25	53.21	54.80	91.20	74.34
	Nonhoax recall:	97.40	97.75	94.18	98.76	98.51	94.82	93.88	89.74	96.69
	Hoax precision:	33.30	38.15	18.15	46.14	44.61	18.81	16.50	16.64	32.87
URLs with ≥2 shares	Hoax recall:	72.12	71.69	62.69	63.94	66.76	59.19	66.56	91.20	74.34
	Nonhoax recall:	95.62	96.66	91.73	97.55	97.84	93.25	92.23	89.74	96.60
	Hoax precision:	41.44	49.19	24.54	46.14	57.07	27.34	26.60	16.64	32.87
URLs with ≥5 shares	Hoax recall:	82.57	81.27	67.38	79.03	78.94	64.93	76.32	91.83	69.79
	Nonhoax recall:	95.47	96.66	90.14	97.00	97.49	92.24	92.83	85.44	95.32
	Hoax precision:	50.24	57.43	27.46	52.83	63.56	31.66	36.12	24.79	43.75
URLs with ≥10 shares	Hoax recall:	87.14	85.43	69.90	85.43	84.80	68.06	83.13	93.52	70.83
	Nonhoax recall:	95.64	96.63	88.91	96.77	97.33	91.50	93.20	84.54	95.43
	Hoax precision:	54.28	60.16	27.24	61.10	65.37	32.23	40.70	25.44	46.62

Table 5: Hoax and non-hoax recall, and hoax precision, expressed as percentages, for the news reputation systems compared in this paper. The methods are LR-U: logistic regression based on users; LR-UT: logistic regression based on users and text; LR-T: logistic regression based on text; Topics: topic analysis, and Harmonic: harmonic crowdsourcing algorithm. For logistic regression, we report results for two training sets: the full one, and the one consisting of URLs that have been shared at least twice. For the other methods, we only report results on the full training set. For Harmonic, we report the results with both 1x and 4x sampling of good URLs. The results are based on the Opensources ground truth.

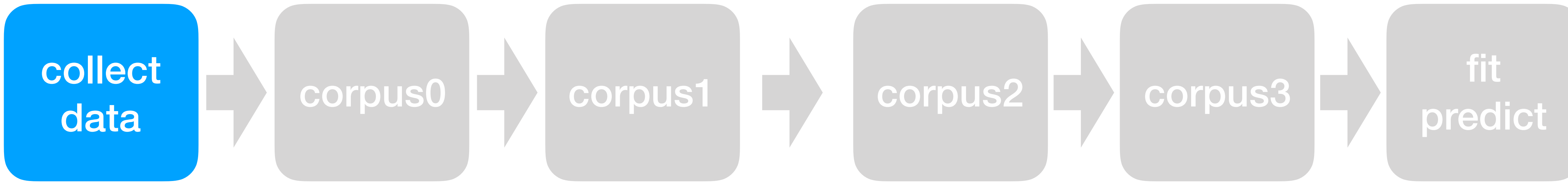
Probability Fake/Unreliable

	LR-UT				LR-U				Harmonic			
	Full Train		Min-2 Train		Full Train		Min-2 Train		Entire, 1x		Entire, 4x	
	OS	MC	OS	MC	OS	MC	OS	MC	OS	MC	OS	MC
nytimes.com	0.88	0.83	0.42	0.35	0.83	0.73	0.19	0.14	4.10	3.95	1.37	1.18
theguardian.com	1.16	1.02	0.48	0.40	0.97	0.93	0.29	0.16	5.13	4.81	1.75	1.60
huffingtonpost.com	2.61	1.96	1.12	0.65	1.43	1.40	0.53	0.46	7.91	7.24	2.72	2.36
washingtonpost.com	1.92	2.12	0.77	0.80	3.07	2.98	0.38	0.33	7.32	7.46	2.45	2.10
arxiv.org	0.08	0.08	0.00	0.00	0.11	0.11	0.05	0.06	0.59	0.45	0.19	0.22
usatoday.com	1.09	1.13	0.56	0.42	0.84	0.63	0.17	0.14	4.43	4.23	1.48	1.12
foxnews.com	1.91	2.01	1.08	0.91	3.07	2.77	1.42	0.78	40.43	37.34	8.05	5.67
nypost.com	3.42	3.20	1.99	1.71	3.70	3.36	1.12	1.28	17.94	17.01	4.90	3.92
thehill.com	3.39	3.43	1.79	1.60	4.64	3.55	1.91	1.09	14.71	14.09	3.96	3.31
latimes.com	0.21	0.17	0.13	0.13	0.38	0.42	0.17	0.17	2.71	2.75	0.83	0.56
breitbart.com	5.10	3.39	3.73	1.79	11.04	9.37	6.32	4.76	77.96	82.20	20.75	13.05
cbsnews.com	1.53	1.88	0.57	0.44	1.62	1.22	0.52	0.39	11.07	10.46	3.92	3.36
reuters.com	1.16	1.11	0.44	0.44	1.07	0.93	0.62	0.36	3.54	3.26	1.29	0.98
dailycaller.com	25.00	41.38	16.76	32.07	50.58	56.59	39.63	50.58	86.41	85.99	29.07	20.77
townhall.com	31.41	16.74	21.33	14.13	38.88	23.49	28.71	16.2	78.53	76.85	31.31	18.96
truthfeednews.com	21.31	2.95	16.8	2.33	18.82	1.87	15.71	2.64	98.28	97.58	81.92	35.75
hotair.com	9.40	2.03	2.99	0.96	15.60	10.15	10.58	7.48	87.38	86.22	11.40	7.01
freedomdaily.com	88.89	87.96	89.81	86.11	81.48	78.70	79.63	77.78	95.88	96.30	79.42	71.60
conservativedailypost.com	93.20	92.37	95.88	95.46	84.54	81.03	79.38	73.61	98.18	98.18	92.66	87.89
lucianne.com	15.65	96.56	2.67	96.56	89.31	96.56	77.48	96.56	99.14	98.45	26.42	34.37
redstate.com	39.23	14.67	38.92	9.57	39.23	18.18	26.48	11.96	78.81	71.72	23.60	10.77
theblaze.com	43.06	24.07	37.96	14.81	16.67	12.50	5.09	4.63	76.01	73.99	20.61	11.92
newsbusters.org	18.22	21.78	14.22	21.78	29.33	31.56	20.89	31.11	89.98	89.74	17.53	11.20
zerohedge.com	28.18	19.86	18.24	6.47	31.18	23.33	16.40	9.24	80.24	67.62	45.88	17.12

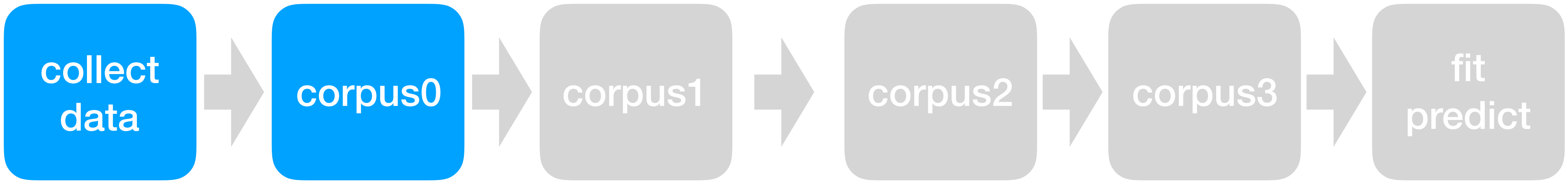
Table 6: Percentage of URLs that are classified as hoaxes for some news sites, including the top news websites of Table 1. Min-2 Train is the training set consisting of URLs that were shared at least twice; Full Train is the full training set. OS stands for Opensources ground truth; MC stands for Metacert ground truth.



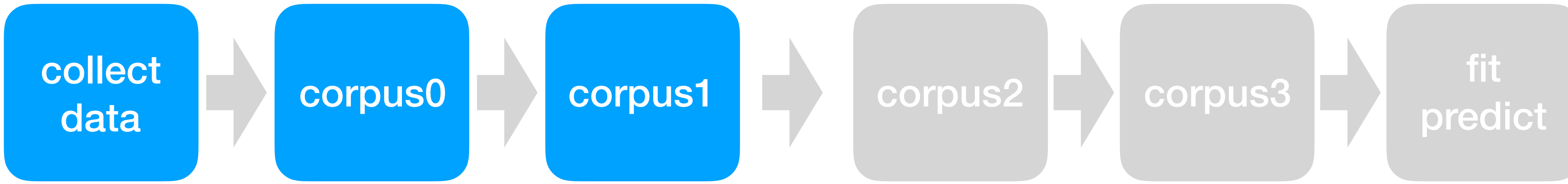
theguardian
nytimes
washingtonpost
economist
huffingtonpost
thehill
thedailybeast
usatoday
reuters
wsj
cbsnews
foxnews
breitbart
nypost
politi.co
vox
msnbc
time
latimes
townhall
truthfeednews
Correlation with arxiv.org (%)
Correlation with Metacert + Opensources dataset (%)
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
0.00
0.05
0.10
0.15
0.20




```python
import twitter
import tweepy

auth = twitter.OAuth(access_token, access_token_secret,
                     consumer_key, consumer_secret)


# STREAM API
class MyListener(tweepy.StreamListener):
    def on_status(self, tweet):
 # process tweet

listener = MyListener()
stream = tweepy.Stream(auth=auth, listener=listener)
stream.filter(follow=words_to_follow)


# REST API
twitter = twitter.Twitter(auth=auth)
result = twitter.search.tweets(q=query, count=100)
```

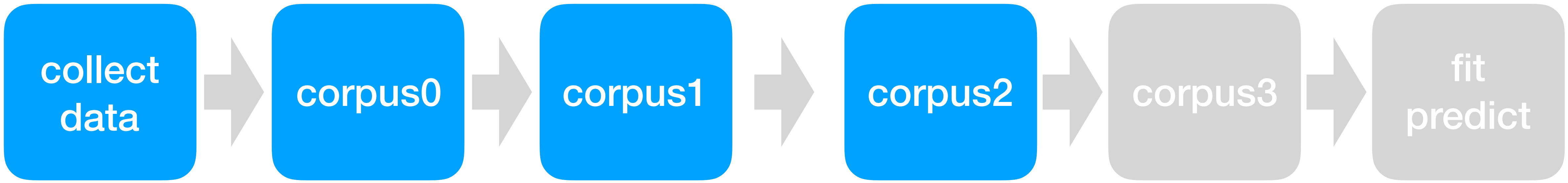



```python
import random

words = ['cat','dog','is','the','on','table','under']

corpus0 = [[random.choice(words)
            for i in range(random.randint(0,10))]
            for j in range(100)]

y = [random.choice((True, False)) for j in range(100)]
# [['dog','is','on','the','table'], ...]
```



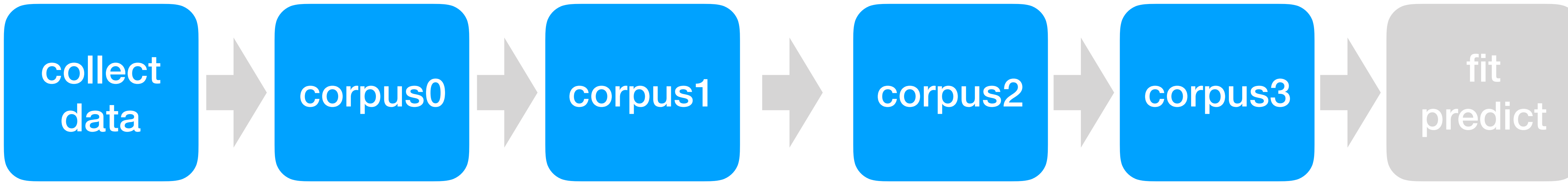

```python
from gensim import corpora, models, similarities

dictionary = corpora.Dictionary(corpus0)

dictionary.filter_extremes(no_below=10, keep_n=100)
# [u'on', u'is', u'dog', u'cat', u'under']
corpus1 = [dictionary.doc2bow(doc) for doc in corpus0] # transform
# [[(0, 1), (2, 1)], …]
```

word 0 “on” appears 1 in document 0



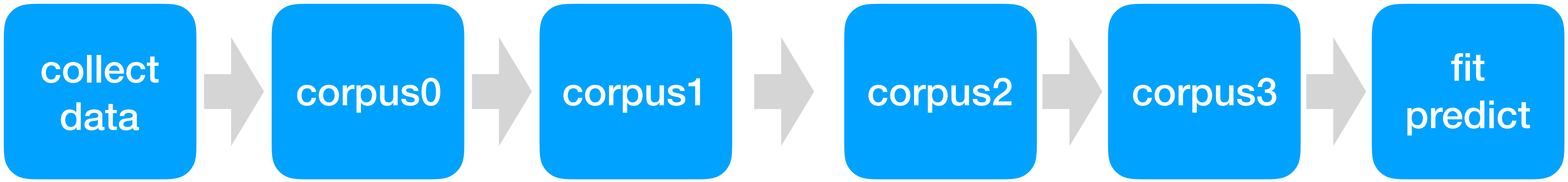

```
tfidf = models.TfidfModel(corpus1)
corpus2 = tfidf[corpus1] # transform
print corpus2[0]
# [[(0, 0.2884171765165675), (2, 0.25082562116562995)], ...]
```

word 0 “on” appears in document 0 with weight 0.288….




```
num_topics = 20

topics_model = models.LdaMulticore(corpus2, id2word=dictionary,
                                   num_topics=num_topics)
corpus3 = topics_model[corpus2] # transform
# [[[(2, 0.017760250785947698)], ...]
```

topic 2 appears in document 0 with weight 0.017…




```python
import numpy as np
from sklearn.linear_model import LogisticRegression

# corpus to features
X = np.ndarray((len(corpus3), num_topics))
for row, doc in enumerate(corpus3):
    for col, weight in doc:
        X[row,col] = weight

# split training vs testing
X_train, y_train = X[:50], y[:50]
X_test,  y_test  = X[50:], y[50:]

# fit/predict
model = LogisticRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
# [ True, True, False, False, False, ...]

# metrics
tn, fp, fn, tp = confusion_matrix(y_test, y_pred).ravel()
recall = float(tp) / (tp + fn)
precision = float(tp) / (tp + fp)
```

Conclusions

Conclusions

- This study was done in 4 months
- There is much more that can be done
- We continue acquire data
- We did not look at time patterns of sharing
- We did not attempt to classify users as bots