CS224W Course Project Writeup

De-anonymizing social networks

Danqi Chen
Stanford University
danqi@stanford.edu

Botao Hu
Stanford University
botaohu@stanford.edu

Shuo Xie
Stanford University
shuoxie@stanford.edu

December 10, 2012

1 Introduction

The problem of de-anonymizing social networks is to identify the same users between two anonymized social networks [7] (Figure 1). Network de-anonymization task is of multifold significance, with user profile enrichment as one of its most promising applications. After the de-anonymization and alignment, we can aggregate and enrich user profile information from different online networking services and make the bundled profiles available for end-users as well as third-party applications.

In our project, we aim to develop effective algorithms for de-anonymizing real-world social networks. Specifically, we focus on two tasks: one is to align the networks of Flickr[1] and Instagram[2] and the other is to align Flickr and Twitter [3]. Our work is motivated by the two parts of information that network data is composed of: network structure and node attributes. Preliminary tests have shown that de-anonymizing algorithm based merely on node attributes, e.g. user names, is computationally efficient but not satisfactorily accurate. On the other hand, algorithms that rely on network structures, which bring in more relationship information, may contribute to the precision of de-anonymization. However, not only may the structure of the real-world social networks be quite different, but also the computation costs will be intractably high since the maximum common subgraph-isomorphism [9] is a NP-hard problem. Hence it is very difficult to align two networks merely based on their structures without any auxiliary information. In view of these facts, we decide to develop approaches that can combine network structure information and node attributes to do the alignment.

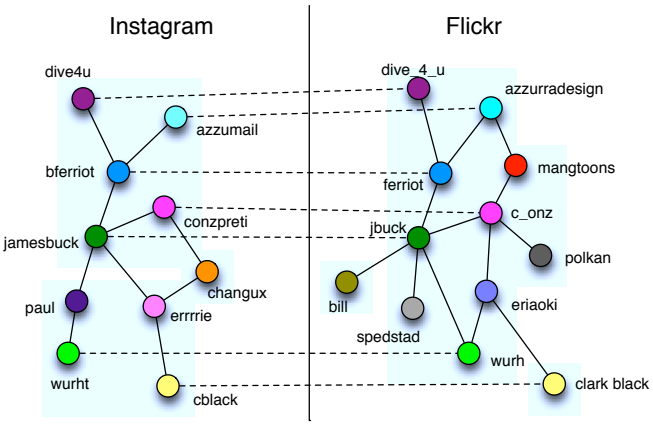


Figure 1: An example of matched user accounts in Instagram & Flickr.

In the end we developed various approaches including greedy-based approaches and network alignment methods. We also carried out a series of experiments to verify their performances on the real-world social network datasets. Our results show that we could identify nearly 70% of the users based on both the user names and the network structure.

This paper is organized as follows. We first give the problem formulation and discuss some related

[1] http://www.flickr.com/
[2] http://instagram.com/
[3] http://twitter.com/

work. Then we introduce our collected datasets and present our observations and analysis of the real-world social networks. Based on our findings, we later introduce our algorithms and demonstrate how they are applied in practice. Finally, we show our experimental results.

2 Related Work

Our network alignment task can be formulated as follows: Given two directed/undirected networks $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ and each node $v \in V_1$ or $v' \in V_2$ is associated with a set of node attributes. Assume that there exists a set $S = \{(v, v') : v \in V_1, v' \in V_2\}$ in which v and v' are actually the same user in G_1 and G_2 respectively, our goal is to identify as many as possible pairs (v, v') in the set S according to the network structure G_1 and G_2 and the node attributes. Concerning the limits on data collection, in our paper we only use user names as the node attribute. In fact user name is a good indication of possibly same users (we will show this later). However, we believe that some other node attributes such as user tags, profile information (location, education, interests, etc) can further improve the network de-anonymization results.

[7] is the first paper to demonstrate feasibility of large-scale, passive de-anonymization of real-world social networks. In this paper, a generic re-identification algorithm that uses only the network structure is developed. The algorithm is composed of two main stages: seed identification and propagation — they first identify a small number of seed nodes and map them to each other and later they use the seed nodes as "anchors" to propoate the de-anonymization to more and more nodes. They show that 30.8% of the verifiable members of Twitter & Flickr could be recognized with 12% error rate. This paper demonstrated feasibility of successful re-identification based solely on the network topology, however, the accuracy is still not satisfying.

Network alignment is another well-formed problem, which aims to perform matching or alignment between the vertices of two undirected graphs. The problem is formulated as a quadratic program(QP): given two undirected graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ and a feasible matching set L between V_1 and V_2, its goal is to find a matching M between V_1 and V_2 using only edges from $L (M \subseteq L)$ such that the number of "overlapped" edges is maximized. Here an edge $(i, j) \in E_1$ is *overlapped* iff. (i, i') and (j, j') belong to M and $(i', j') \in E_2$. However, solving the QP formulation is NP-hard, and different approaches have been adopted in existing network alignment works to relax the constraints or find proper heuristic functions. The approaches include:

- IsoRank[10] and its variants[4]: approximating the objective without concerning the matching constraints directly.
- Linear Programming(LP) approaches[3]: linear programming relaxations.
- Belief Propagation(BP) approaches[1]: transforming the QP formulation into a maximization of a factored probability distribution.

These approaches have been applied successfully in some applications such as finding common pathways in biological networks [10, 11] and ontology alignment between Citeseer papers and DBLP papers[2]. However, compared with our real-world social networks, those networks are of smaller scale and are more structured. To our knowledge, no network alignment algorithm has been applied to the task of de-anonymizing social networks.

In addition, in last year's course project [5], Krietmann proposes a simulated annealing algorithm to align the networks of two language versions (German and French) of Wikipedia. However, his approach is also dependent exclusively on network structure. The social networks we are dealing with are largely different from Wikipedia. Moreover, our task seems more challenging as the network structure of two social networks tend to be much more different from each other than the two networks of two language versions of Wikipedia.

3 Data Analysis

3.1 Datasets

In our paper, we use the data from three large online social networks, namely Twitter, Flickr and Instagram. All the three social networks are directed, i.e., each user has a list of following users and a list of followed-by users. We crawl the Flickr and Instagram data by ourselves, in which both following and followed-by lists of users can be accessible via the website APIs. And we download the Twitter dataset from [6], which is accessible online [4].

One concern we have with these datasets is that this Twitter dataset was crawled in 2009 and may be outdated, thus may not be matched well with the latest data from Flickr. To verify this, we did a test on the ground truth data(see below) and found that among 45,637 users who have both Flickr and Twitter accounts, 25,630(56.2%) are covered in this 2009 Twitter dataset, which is enough for our experiments.

Table 1 lists some important statistics of the social network data we've obtained. We find that both the scale of the networks and their average degree(in-degree and out-degree) are quite different. To make the edge information more effective, in our experiment we converted all the three directed networks to undirected networks (and also removed all isolated vertices) so that edge (v_1, v_2) exists if $(v_1 \rightarrow v_2)$ and $(v_2 \rightarrow v_1)$ in the original directed network. The reason for making this conversion is that we believe bilateral relationship is more steady than one-way relationship. For instance, if both v_1 follows v_2 and v_2 follows v_1 exist in one social network, v_1 and v_2 are likely to be "friends" and know each other, so the edge (v_1, v_2) is likely to appear in other social networks. However, a directed edge $(v_1 \rightarrow v_2)$ may be extremely "unreliable" especially when v_2 is a famous person(for ex. Lady Gaga is followed by millions of users in Twitter and the followers are changing everyday). The statistics of the converted undirected dataset are given in Table 2.

[4]http://an.kaist.ac.kr/traces/WWW2010.html

Dataset	$\|V\|$	$\|E\|$	Avg. Degree
Instagram	57,559,925	3,317,058,584	57.63
Flickr	15,332,780	251,197,130	16.38
Twitter	41,652,230	1,468,365,182	35.25

Table 1: The statistics of datasets(directed)

Dataset	$\|V\|$	$\|E\|$	Avg. Degree
Instagram	25,981,813	537,932,984	41.41
Flickr	9,685,876	112,953,841	23.32
Twitter	22,580,419	531,703,974	47.09

Table 2: The statistics of datasets(undirected)

To further analyze the overlap between real-world social networks and later evaluate our experimental results, we have to get some ground truth data first – the true mapping between the different online social network accounts of the same user. To achieve this, we use the user profile data from about.me [5] to be our ground truth – About.me is a personal web hosting service, which had at least 1 million users by October, 2011[6]. The site offers registered users a simple platform to link multiple online identities, relevant external sites, and popular social networking websites such as Google+, Twitter, Facebook, LinkedIn, Flickr, YouTube, Foursquare. These links on user profile is actually manually labelled mapping generated by users themselves, which can be seen as a source of non-error ground truth. Finally, we collected 494,490 users and the number of accounts from different social networking websites that appear in the user profiles dataset is shown in Figure 2.

As can be seen in the figure, 228,070(46.1%) users have Twitter accounts, 60,666(12.3%) users have Instagram accounts and the number for Flicker is 54,104(10.9%). The users which have both Twitter and Flickr accounts or Flickr and Instagram accounts in about.me and also can be identified from our social network data are our final collection of ground truth – Finally we get 18,941

[5]http://about.me

[6]http://techcrunch.com/2011/10/17/about-mes-ceo-on-how-to-hit-a-million-users-in-300-days-figure-out-who-your-entourage-is/

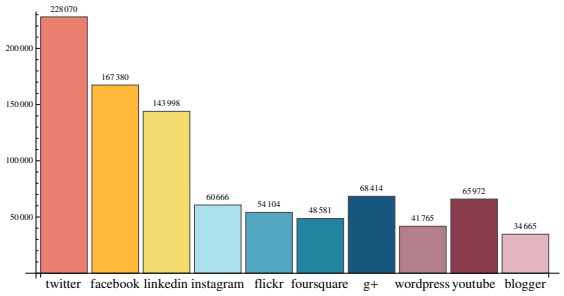


Figure 2: The number of accounts in the various social networks we obtained from about.me

Twitter & Flickr ground truth pairs and 4,691 Instagram & Flickr ground truth pairs.

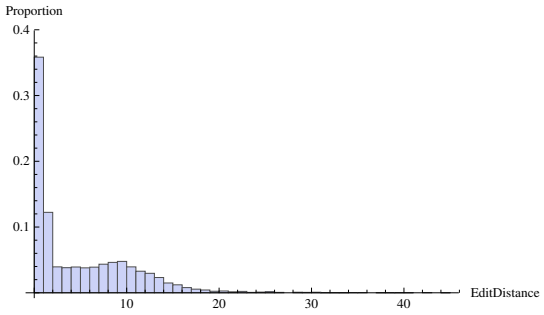
3.2 User Name Similarity

The most intuitive method to re-identify the same user in two networks is to compare the user names of the accounts in two networks. In this subsection, we aim to explore how the user names of the same user in different social networks are correlated. For the dataset of Twitter & Flickr and Instagram & Flickr, we investigate the name similiarity in their ground truth pairs.

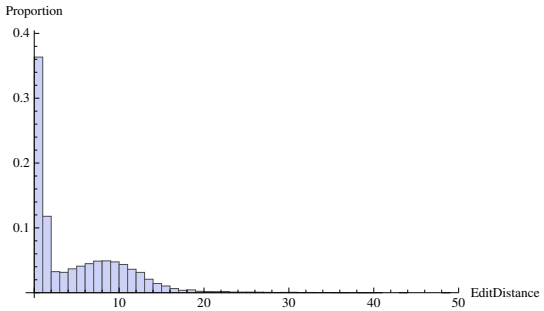
We could define the similarity of two strings a and b by $\sigma(a,b) = \mathrm{lev}_{a,b}(|a|,|b|)$ where lev is the *Levenshtein distance* a.k.a. *edit distance*. Informally, the Levenshtein distance between two words is equal to the number of single-character edits required to change one word into the other. Mathematically, it is defined by the recursion

$$\mathrm{lev}_{a,b}(i,j) = \begin{cases} \max(i,j) & \min(i,j)=0 \\ \min \begin{cases} \mathrm{lev}_{a,b}(i-1,j)+1 \\ \mathrm{lev}_{a,b}(i,j-1)+1 \\ \mathrm{lev}_{a,b}(i-1,j-1)+[a_i \neq b_j] \end{cases} & \text{else} \end{cases}$$

Figure 3 gives the histogram of the similarity of the two user names the same user have in Instagram & Flickr and Flickr & Twitter. **A strong long-tail effect can be observed**: only 36% of the Instagram & Flickr (and also 36% of the Twitter & Flickr) users have the exact same user name in both two websites; and in total, 47% of the Instagram & Flickr (and 48% of the Twitter & Flickr) users have the edit distance of the user name larger than 2. The table 3.2 shows some examples of the user names of ground truth users. We can see that a majority of users use different user names in different social networks, hence we cannot identify many users only based on the user names, which addresses the necessity of taking network structures into consideration.



(a) Instagram & Flickr



(b) Twitter & Flickr

Figure 3: Histogram of the similarity (Edit Distance) of the two screen names in Flickr and Twitter of the same user in ground truth.

In. user name	Fl. user name	Edit Distance
beckami	ami.becker	7
cblack	carl black	4
andycampbell	andersoncampbell	5
kkbyrns	kimbyrns	2
conzpreti	c_onz	6

Table 3: Some examples with user name similarity

	In. & Fl.	Tw. & Fl.
Ground truth size	4691	18941
Edge coverage 1st graph	25.3%	18%
Edge coverage 2nd graph	45.3%	39%

Table 4: Network structure similarity

3.3 Network Overlap

Since adopting only the name similarity to match users in different social networks can not cover users spanning at the long-tail, we try to utilize the network structure information to help us match users in different networks. The intuition behind is that **if two users are friends in one social network, there is a high probabilty that they are friends in another social network**. If there is a big overlap between two social networks, it is promising to use the network structure information to improve the results.

To verify this, we extract the subgraphs of Flickr and Twitter (Instagram and Flickr) induced by the nodes in ground truth. We count the number of edges in Flickr that have their corresponding edges in Twitter existing; and vice versa. The result is listed in 3.3. As we see, 25.3% edges in the Instagram subgraph have corresponding edges in Flickr network; and **45.3%** edges in the Flickr subgraph have corresponding edges in Instagram; 18% edges in the Twitter subgraph have corresponding edges in Flickr network; and **39%** edges in the Flickr subgraph have corresponding edges in Instagram. We can see that the overlap between Flickr and Instagram is higher than that between Flickr and Twitter, and this is very reasonable as Instagram and Flickr are both photo-sharing websites, and we expect that there is much more overlap between Flickr and Instagram.

These figures imply that the network structures of these two networks are, to some extent, similar. This similarity can bring us some information to align the network with higher accuracy. However, since a large part of of edges in one network are not matched in the other, it is still very difficult to align the two networks based on their pure structures without any auxiliary information.

Based on the above argument, we decide to develop approches that can combine both of them.

4 Methology

In this section, we introduce two main approaches – one is based on network alignment algorithms, and the other is based on greedy heuristics. Prior to that, we discuss the bottleneck of the algorithms that hinders their application on large-scale social networks and propose our solutions.

4.1 The Bottleneck

In fact, the problem of de-anonymizing two large-scale social networks is a very complex and difficult task – If we want to match two social networks with $|V_1|$ and $|V_2|$ nodes respectively, and if we simply enumerate every node in G_1 and then enumerate every node in G_2, the time complexity is at least $|V_1| \times |V_2|$ which is already immensely huge on the original dataset.

A possible solution for this is to constrain the candidate set of nodes for each node $v \in G_1$ – if we can constrain a set $candidate(v) \subseteq V_2$ for each $v \in V_1$, then we only need to enumerate $v' \in candidate(v)$ instead of the whole node set V_2. If the average of $|candidate(v) : v \in V_1|$ is much less than $|V_2|$, then the computation time for aligning two networks may be reduced significantly. Now the problem turns into that how could we decide $candidate(v)$ for each $v \in V_1$ as fast as possible? It seems quite difficult to choose the candidate set solely depending on the network structure, as the number of nodes and the average degree vary a lot between two social networks. In our project, we propose to use the similarity of user names as a constraint, and select $candidate(v)$ according to the similarity of user names of v and v'.

If we still apply the Edit Distance, we have to enumerate each pair of $(v, v') : v \in V_1, v' \in V_2$ and the computation cost of each pair is still as high as $|name_1(v)| \times |name_2(v')|$, which is not practical. Our alternative strategy is to use some fast approximate string matching method – the basic idea is to use the distribution of bi-grams of a string to con-

struct a vector. In this way, the following problem of finding nearest neighbours approximately among a set of vectors will be a classical problem, which can be solved much more efficiently.

In our following experiments, we will use the method introduced in [8] and constrain $candidate(v)$ such that

$$cosine(bigram(v), bigram(v')) \geq 0.5,$$

which is of much lower computation cost than using Edit Distance.

4.2 Network Alignment Approaches

One of the main approaches we have tried is to embrace the network alignment algorithms into our data settings – In original problem settings, we need to get L, which is a feasible matching set. Since the running time of the algorithms depends on the number of L, therefore to apply the algorithms on the large-scale social networks, we have to constrain L to be a very sparse matrix.

A very natural solution to this problem is to apply the candidate sets $\{candidate(v) : v \in V_1\}$ we generated before to get L:

$$L = \{(v, v') : v \in V_1, v' \in candidate(v)\}.$$

In this case, we not only combine the user name information and the network structure information, but also reduce the running time to a great extent.

As in shown in [1], the *Matching Relaxation* Algorithm(MR) [3] and *Message Passing* Algorithm(BP) are the only two algorithms that can scale up to large and sparse networks, therefore we will use the two algorithms in the following experiments.

4.3 Greedy-based Approaches

Besides network alignment approaches, we have also tried some greedy-based algorithms. The basic idea is to start with some seed nodes that have already been identified in both networks and map them to each other; later we use seed nodes to propagate the de-anonymization process to more and more nodes. Specifically, at each iteration we pick an unmapped node in one network and try to map it to the "most similar" node in the other network, where the similarity score $score(v, v')$ is computed based on some heuristics.

The idea is quite straightforward, however we need to consider the following problems:

- **The choice of seed nodes:** As we don't have any ground truth nodes in the networks we try to align, we make an assumption that all the accounts with exactly the same user names in two social networks belong to the same user. Empirically speaking, it is the real case for most of the time, with few exceptions. With this assumption, we can match v and v' such that $name_1(v) = name_2(v')$ first and use them as the seed nodes.

- **The choice of mapping order:** In our experimemt, priority is given to an unmapped node with highest number of edges, i.e.,

  $$degr(v) = |\{v_2 : v_2 \text{ is matched and } (v, v_2) \in E_1\}|$$

- **The choice of similarity:** How to define the similarity score matters a lot to the performance of the algorithm. We aim to combine the network structure information and the user names. Meanwhile, we cannot enumerate each $v' \in V_2$ for each v because of the computation cost. In responding to these considerations, we propose to use the candidate sets we generated using fast string matching method before, and constrain $v' \rightarrow beincandidate(v)$ only. In this case, we treat v and v' as completely dissimilar if $v' \notin candidate(v)$. After we made this constraint, we compute the "overlapped" edges that (v, v') generated if we map v to v', which is defined as:

  $$overlap(v, v') = |\{(v_2, v_2') : v_2 \text{ and } v_2' \text{ are already matched}, (v, v_2) \in E_1, (v', v_2') \in E_2\}|$$

  The more overlapped edges there are, the more v is likely to be mapped to v'. If there is a tie, we choose v' with the smallest edit distance between $name_1(v)$ and $name_2(v')$.

Therefore in our algorithm, the overall similarity score can be defined as:

$$s(v, v') = \begin{cases} overlap(v, v') \times 100 & v' \in candidate(v) \\ -\sigma(name_1(v), name_2(v')) & \\ 0 & \text{otherwise} \end{cases}$$

The higher the similarity score is, the more similar v is to v'. Therefore in each iteration we pick one unmapped node v' with the largest similarity score $s(v, v')$.

Time Complexity The running time of our algorithm still has a bottleneck – how could we pick an unmapped node v with largest $degr(v)$? In our implemenation, we use a priority queue (for ex. heap) to maintain the largest $degr(v)$ for all unmapped nodes in V_1.At each step, we get v from the priority queue with largest $degr(v)$, and then enumerate all $v' \in candidate(v), (v, v_2) \in E_1$ and check if v_2 is mapped to some v_2' and if $(v', v_2') \in E_2$ to compute $overlap(v, v')$; we also compute $\sigma(name_1(v), name_2(v'))$ to obtain the final similiarity score. Assume the average degree of networks G_1 is $\bar{d}_1$, the average of $|\{candidate(v)\}|$ is c and the average length of user names is l, we can implement the algorithm within time complexity of

$$O(|E_1| \log |V_1| + |V_1| c(\bar{d}_1 + l^2)).$$

Since the average degree of social networks is $20 \sim 50$ and the average length of user names is also quite small ≈ 10, if we can constrain the candidate sets to a small value of c, then the time complexity of this algorithm is acceptable and the algorithm can be applied to large-scale social networks.

5 Experimental Results

5.1 Experimental Settings

Unfortunately, we don't have enough resources and time to run on the full datasets – the largest network has more than 5 billion edges! However, we randomly sample three sub-networks, two for Instagram & Flickr and one for Twitter & Flickr. The size of the networks and the number of ground truth pairs are given in Table 5. We can see that our sampled sub-networks are large enough to prove the efficiency of our algorithms.

5.2 Results

We tried several approaches on all the three datasets and the results are given in Table 6. The approaches include:

- ExactMatch: Baseline – we match the users with exactly same user names.
- MP: Message Passing(MP) algorithm - Our experiments show that it cannot be applied to the two larger datasets.
- MR: Matching Relaxation(MR) algorithm - Our experiments show that it cannot be applied to the two larger datasets.
- Random: we match v and v' randomly sampled from $candiate(v)$.
- Greedy+Overlap: our greedy algorithm using $overlap(v, v')$ as similarity score.
- Greedy+Overlap+Editdistance: our greedy algorithm using $overlap(v, v') * 100 - \sigma(name_1(v), name_2(v'))$ as similarity score.

For approaches MP, MR, Random, Greedy+Overlap, Greedy+Overlap+EditDistance, we all apply the "fast approximate string match" method to find the candidate sets $candidate(v)$, and if there exists $name_1(v) = name_2(v')$, there is $candidte(v) = \{v'\}$.

Based on the results, we can see that our greedy algorithm using number of "overlapped" edges and edit distance can reach the highest accuracy — 68.64%, 66.07%, 64.74% in the three datasets respectively, which is very impressive. On the contrast, the two network alignment based algorithms (MP and MR) can only reach 48.61% and 43.95% of accuracy on the first dataset and cannot scale up to the two larger datasets. It is worth noting that

Dataset	$\|V_1\|$	$\|E_1\|$	$\|V_2\|$	$\|E_2\|$	GroundTruth
Instagram & Flickr - 1	67,370	1,459,828	43,940	956,488	1,183
Instagram & Flickr - 2	552,895	30,024,799	342,183	21,487,664	781
Twitter & Flickr	178,819	94,033,003	48,527	1,147,430	1,816

Table 5: Experimental Settings

	Instagram & Flickr - 1	Instagram & Flickr - 2	Twitter & Flickr
ExactMatch	39.14%	37.00%	39.76%
MP	48.61%	-	-
MR	43.95%	-	-
Random	47.68%	44.7%	45.04%
Greedy+Overlap	57.23%	47.6%	59.31%
Greedy+Overlap+EditDistance	68.64%	66.07%	64.76%

Table 6: Experimental Results

after we incorporate the EditDistance to the similarity function, the results are greatly improved.

6 Conclusion

In our project, we tried to de-anonymize social networks – to our knowledge, this is the first attempt to perform large-scale social network alignment by combining both user names and network structure. Our approaches can be succefully implemented on large-scale real-world networks in reasonable time, while other methods based on network alignment can not. Finally our approaches can identity near 70% of users with evaluation of ground truths, which is very impressive. However, it is a pity that we cannot run our algorithm on the full datasets we have gathered, and we believe that its performance can be further improved if we can incorporte more node attributes.

References

[1] M. Bayati, D. Gleich, A. Saberi, and Y. Wang. Message passing algorithms for sparse network alignment. *arXiv preprint arXiv:0907.3338*, 2009.

[2] W. Hu, Y. Qu, and G. Cheng. Matching large ontologies: A divide-and-conquer approach. *Data Knowl. Eng.*, 67(1):140–160, Oct. 2008.

[3] G. W. Klau. A new graph-based method for pairwise global network alignment. *BMC Bioinformatics*, 10(Suppl 1):S59, 2009.

[4] G. Kollias, S. Mohammadi, and A. Grama. Network Similarity Decomposition (NSD): A Fast and Scalable Approach to Network Alignment. *IEEE Transactions on Knowledge and Data Engineering*, PP(January):1, 2011.

[5] P. Kreitmann. CS224W: Project Writeup. pages 1–12, 2011.

[6] H. Kwak, C. Lee, H. Park, and S. Moon. What is Twitter, a social network or a news media? In *WWW '10: Proceedings of the 19th international conference on World wide web*, pages 591–600, New York, NY, USA, 2010. ACM.

[7] A. Narayanan and V. Shmatikov. De-anonymizing Social Networks. *2009 30th IEEE Symposium on Security and Privacy*, pages 173–187, May 2009.

[8] N. Okazaki and J. Tsujii. Simple and efficient algorithm for approximate dictionary

matching. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 851–859, Beijing, China, August 2010.

[9] J. Raymond and P. Willett. Maximum common subgraph isomorphism algorithms for the matching of chemical structures. *Journal of computer-aided molecular design*, 16(7):521–533, 2002.

[10] R. Singh, J. Xu, and B. Berger. Pairwise global alignment of protein interaction networks by matching neighborhood topology. In *Proceedings of the 11th annual international conference on Research in computational molecular biology*, RECOMB'07, 2007.

[11] R. Singh, J. Xu, and B. Berger. Global alignment of multiple protein interaction networks with application to functional orthology detection. *Proceedings of the National Academy of Sciences*, 2008.