An Ensemble Approach to Link Prediction

Liang Duan, Shuai Ma*, Charu Aggarwal, Tiejun Ma, Jinpeng Huai

(Supplementary Material)

◆

1 APPENDIX A: THE CHOICE OF VALUES FOR K

As pointed out in [**?**], a small difference in k will lead to different conclusions on link prediction results and k should be the number of links in the ground truth data for a fair comparison. Here, we have further conducted three sets of experiments when k is varied from hundreds to the number of links in the ground truth data to evaluate the impacts of k on our ensemble-enabled approach compared with conventional methods AA, RA and BIGCLAM. These are useful for choosing a proper value for k.

1.1 Bagging+ vs. Bagging

In the first set of tests, we evaluated the impacts of k on the effectiveness and efficiency of our bagging+ methods compared with bagging methods.

Exp-4.1: Impacts of k. We varied k from hundreds to the number of links in the ground truth data [**?**]. We fixed $r = 30$ on Digg and YouTube, $r = 20$ on Wikipedia according to Exp-6 and fixed other parameters to their default values. The results of accuracy and efficiency are reported in Figures 1(a)–1(c) and Figures 1(d)–1(f), respectively.

The accuracy results tell us that (a) NMF(Biased+) is the best method on all datasets, (b) the accuracy of all methods decreases with the increase of k, and (c) the accuracy of the bagging+ methods is higher than that of their counterparts bagging methods when k is small and very close to that of their counterparts when k is large. This means that the bagging+ methods maintain the high accuracy while some of the edges have been removed compared with their counterparts bagging methods. This justifies the effectiveness of the bagging+ methods.

The efficiency results tell us that (a) the NMF(Biased+) is the fastest, (b) the three bagging+ methods are faster than their counterparts bagging methods, and (c) the running time of all methods increases slightly with the increase of k. For instance, NMF(Biased+) is $(1.2, 1.1, 1.2)$ times faster than NMF(Biased) on Digg, YouTube and Wikipedia, respectively. This justifies the efficiency of the bagging+ methods.

Exp-4.2: Impacts of network sizes. We evaluated the impacts of k with different network sizes when k is fixed to the number of links in the ground truth data. We used the same networks with different sizes as Exp-1.3, fixed k to the number of links in the ground truth data and used the same setting of other parameters as Exp-4.1. The results of accuracy and efficiency are reported in Figures 2(a)–2(c) and Figures 2(d)–2(f), respectively. Since we focus on the accuracy comparison in this section, we did not report the results on Twitter and Friendster as they do not have the ground truth data for comparison.

The accuracy results tell us that (a) the accuracy of bagging+ methods is very close to that of bagging methods, and (b) the accuracy of all methods decreases with the increase of network sizes because k is smaller on the small networks, which is consistent with the conclusion in Exp-4.1 that the accuracy is higher when k is smaller. It is hard for NMF(Biased) to provide a significant accuracy improvement when k is very large, but it still provides an efficiency advantage while keeping its accuracy close to NMF(Node) and NMF(Edge). Similar trends have been found for NMF(Biased+). This means that our bagging+ methods are effective on the networks with different sizes.

The efficiency results tell us that (a) NMF(Biased+) is the fastest, (b) bagging+ methods are faster than their counterparts bagging methods, and (c) the running time of all methods increases nearly linearly with the increase of network sizes. This justifies the efficiency of the bagging+ methods.

1.2 Comparison with AA, RA and BIGCLAM

In the second set of tests, we evaluated the impacts of k on the effectiveness and efficiency of our methods compared with AA, RA and BIGCLAM. We chose NMF(Biased) and NMF(Biased+) for the comparison since they are the best bagging methods according to Exp-4.1.

Exp-5.1: Impacts of k. Using the same setting as Exp-4.1, we evaluated the impacts of the number k of predicted links. The results of accuracy and efficiency are reported in Figures 3(a)–3(c) and Figures 3(d)–3(f), respectively.

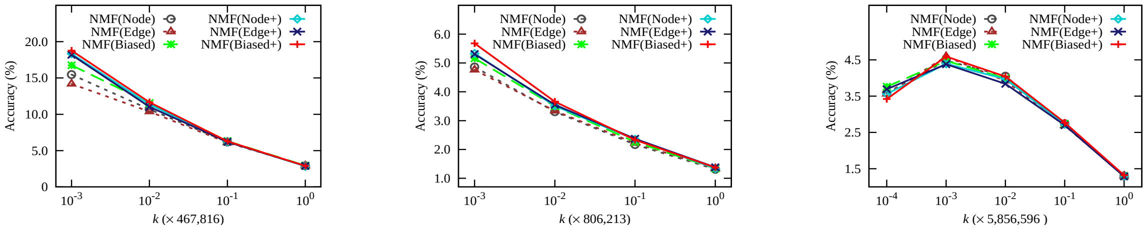
The accuracy results tell us that (a) the bagging+ and bagging methods outperform the other methods on most datasets, (b) both NMF(Biased) and NMF(Biased+) have a higher accuracy than NMF, AA, RA and BIGCLAM, except for Wikipedia, and (c) NMF is more accurate than AA, RA and BIGCLAM on most datasets. Moreover, NMF(Biased) and NMF(Biased+) perform consistently well on all networks (*i.e.,* more robust), unlike RA which works well on Wikipedia, but poorly on the other datasets. This means that our methods are more accurate and robust.

The efficiency results tell us that (a) NMF(Biased+) is the fastest compared with NMF(Biased), NMF and BIGCLAM, (b) the two bagging methods are much faster than NMF and BIGCLAM, and (c) the running time of AA and RA increases rapidly

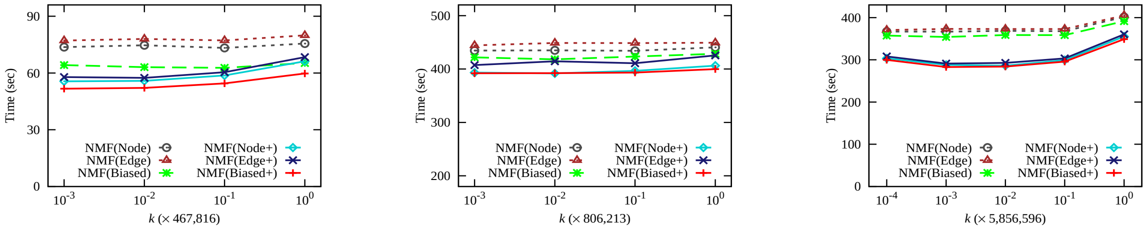
---

- *L. Duan, S. Ma (correspondence) and J. Huai are with SKLSDE lab, School of Computer Science and Engineering, Beihang University, China. E-mail: {duanliang, mashuai, huaijp}@buaa.edu.cn*
- *C. Aggarwal is with IBM Thomas J. Watson Research Center, USA. E-mail: charu@us.ibm.com*
- *T. Ma is with Centre for Risk Research, Department of Decision Analytics and Risk, University of Southampton, UK. E-mail: tiejun.ma@soton.ac.uk*

Manuscript received XXX, 2016; revised XXX, 2017.

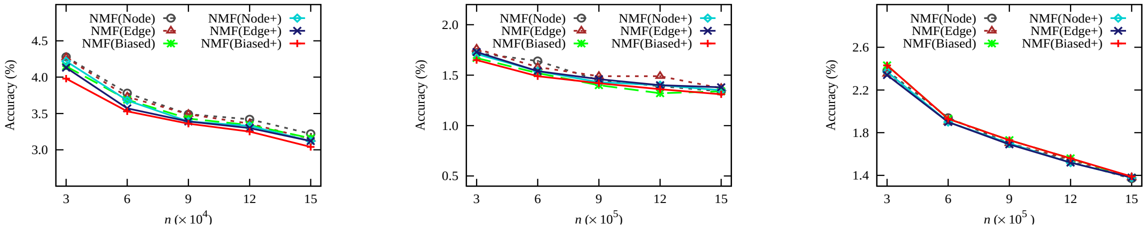


(a) Digg (b) YouTube (c) Wikipedia

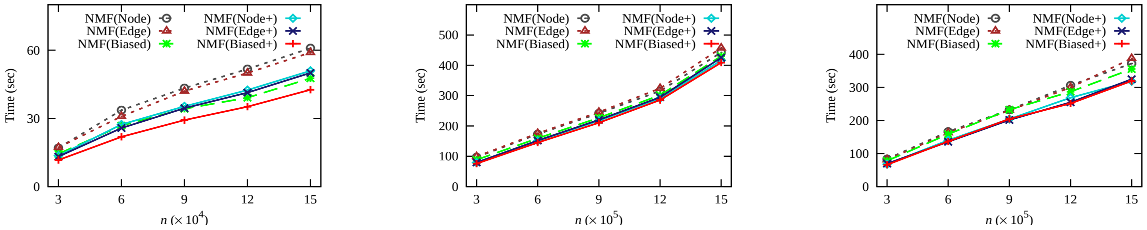


(d) Digg (e) YouTube (f) Wikipedia

Figure 1. Bagging+ vs. Bagging on accuracy and efficiency: with respect to the number k of predicted links.

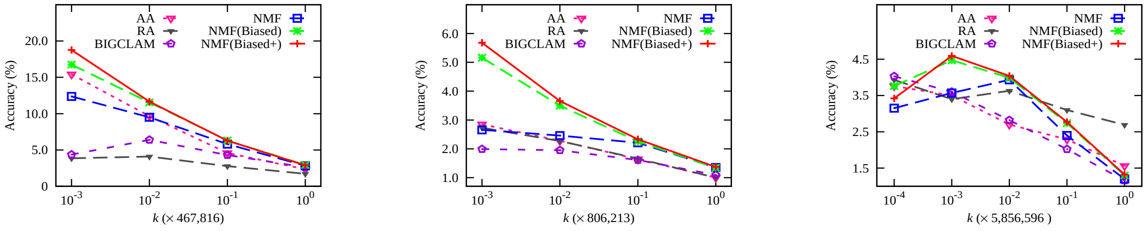


(a) Digg (b) YouTube (c) Wikipedia

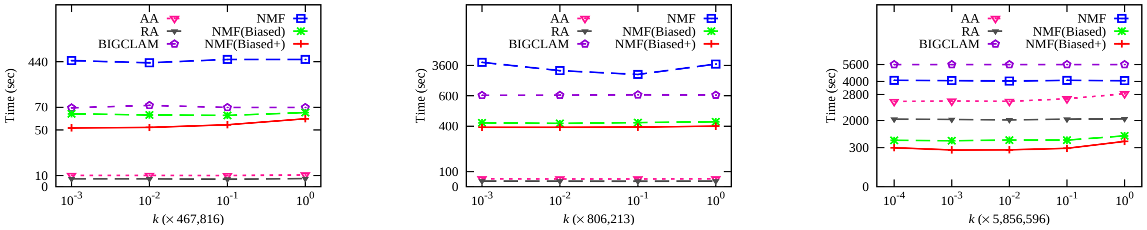


(d) Digg (e) YouTube (f) Wikipedia

Figure 2. Bagging+ vs. Bagging on accuracy and efficiency: with respect to the network sizes.

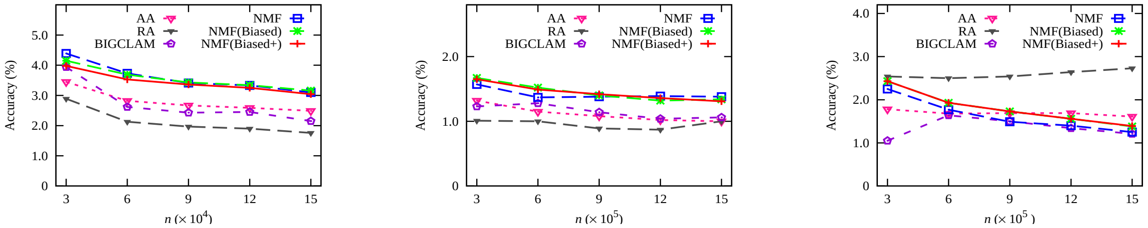


(a) Digg (b) YouTube (c) Wikipedia

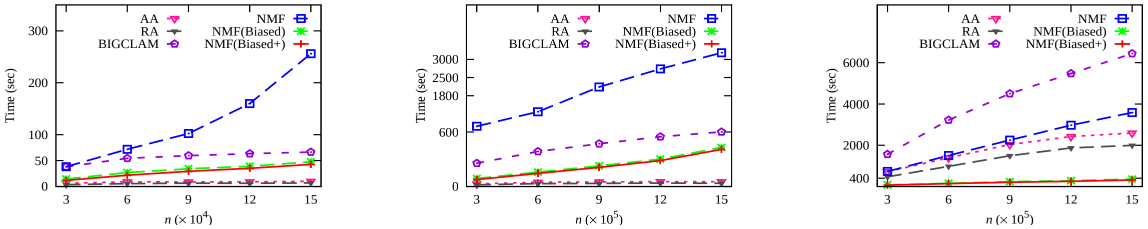


(d) Digg (e) YouTube (f) Wikipedia

Figure 3. Accuracy and efficiency comparison with AA, RA and BIGCLAM: with respect to the number k of predicted links.

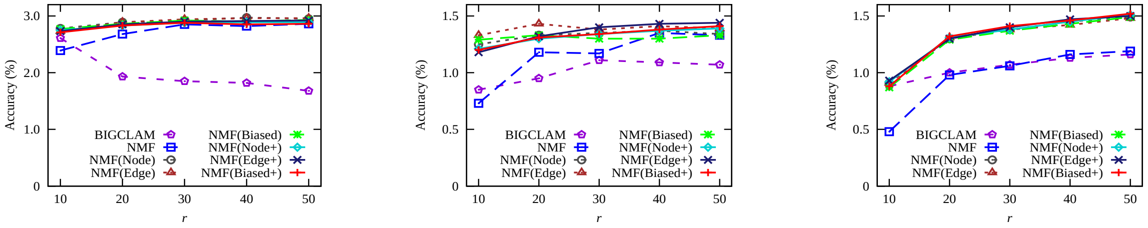


(a) Digg (b) YouTube (c) Wikipedia

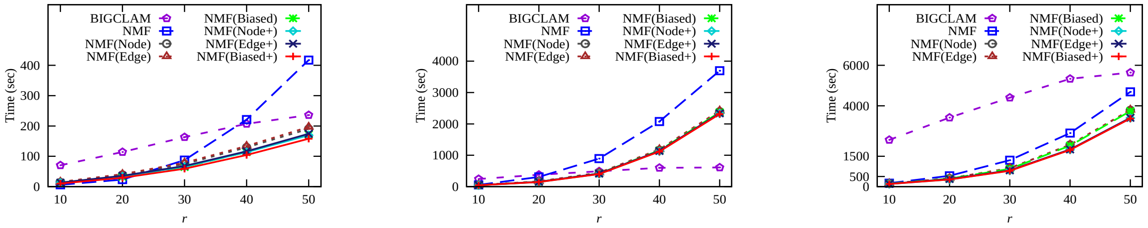


(d) Digg (e) YouTube (f) Wikipedia

Figure 4. Accuracy and efficiency comparison with AA, RA and BIGCLAM: with respect to the network sizes.



(a) Digg (b) YouTube (c) Wikipedia



(d) Digg (e) YouTube (f) Wikipedia

Figure 5. Accuracy and efficiency comparison: with respect to the number r of latent factors.

with the increase of the network degree since their complexities are $O(nd^2 \log(k))$. Indeed, the two bagging methods finished in 421 seconds on the three datasets.

Exp-5.2: Impacts of network sizes. Using the same setting as Exp-4.2, we evaluated the impacts of k with different network sizes. The results of accuracy and efficiency are reported in Figures 4(a) – 4(c) and Figures 4(d) – 4(f) , respectively.

The accuracy results tell us that (a) NMF(Biased) is more accurate than the other methods on most datasets, (b) NMF(Biased+) performs as well as it counterpart NMF(Biased), and (c) NMF has a higher accuracy then AA, RA and BIGCLAM on most datasets. This justifies the effectiveness of our methods.

The running time results tell us that (a) NMF(Biased+) is the fastest method compared with NMF(Biased), NMF and BIGCLAM, (b) AA and RA run fast on Digg and YouTube with small network degree but slow down on Wikipedia with larger network degree, and (c) the running time of all methods increase with the increases of network sizes. This is consistent with the conclusion in Exp-2.2 and also justifies the efficiency of our methods.

1.3 Impacts of Parameters

In the third set of tests, we evaluated the impacts of the parameter when k is the number of links in the ground truth data. Since the parameters μ, f, ρ and ϵ have similar impacts as shown in Section 4.2.3, we keep the default values for these parameters and report only the impacts of the parameter r on the accuracy and efficiency of bagging+, bagging, NMF and BIGCLAM.

Exp-6: Impacts of r. To evaluate the impacts of r, we varied r from 10 to 50, fixed k to the number of links in the ground truth data and fixed other parameters to their values. The accuracy and efficiency results are reported in Figures 5(a)–5(c) and Figures 5(d)–5(f), respectively.

The results tell us that (a) bagging+ and bagging methods are more accurate than the other methods, (b) the accuracy of bagging+ methods is very close to that of bagging methods, (c) the accuracy of the bagging+, bagging and NMF increases with the increase of r, and (d) the running time of all methods increase with the increases of r and NMF(Biased+) is still the fastest one. To obtain the highest accuracy, we keep the default value of r for NMF and BIGCLAM, and fixed $r = 30$ on Digg and YouTube

Table 1
Accuracy (%) comparison with 95% confidence intervals in Exp-1.2.

Dataset	k	NMF(Node)	NMF(Edge)	NMF(Biased)	NMF(Node+)	NMF(Edge+)	NMF(Biased+)
Digg	1×10^4	8.94 ± (0.147)	8.86 ± (0.186)	9.33 ± (0.343)	9.21 ± (0.153)	9.38 ± (0.120)	9.59 ± (0.197)
	2×10^4	7.54 ± (0.156)	7.62 ± (0.136)	7.88 ± (0.304)	7.74 ± (0.086)	7.82 ± (0.111)	8.00 ± (0.141)
	3×10^4	6.84 ± (0.118)	6.93 ± (0.111)	7.07 ± (0.190)	6.93 ± (0.114)	7.05 ± (0.157)	7.13 ± (0.108)
	4×10^4	6.39 ± (0.107)	6.43 ± (0.055)	6.52 ± (0.162)	6.46 ± (0.087)	6.57 ± (0.119)	6.59 ± (0.064)
	5×10^4	6.03 ± (0.089)	6.07 ± (0.077)	6.13 ± (0.083)	6.07 ± (0.059)	6.16 ± (0.055)	6.17 ± (0.075)
	6×10^4	5.72 ± (0.063)	5.78 ± (0.066)	5.79 ± (0.082)	5.75 ± (0.034)	5.81 ± (0.021)	5.81 ± (0.082)
	7×10^4	5.45 ± (0.070)	5.51 ± (0.093)	5.53 ± (0.101)	5.47 ± (0.034)	5.51 ± (0.041)	5.55 ± (0.108)
	8×10^4	5.23 ± (0.064)	5.30 ± (0.139)	5.32 ± (0.093)	5.26 ± (0.059)	5.28 ± (0.057)	5.31 ± (0.097)
	9×10^4	5.06 ± (0.065)	5.11 ± (0.113)	5.12 ± (0.062)	5.08 ± (0.067)	5.10 ± (0.036)	5.11 ± (0.090)
	10×10^4	4.91 ± (0.059)	4.94 ± (0.115)	4.95 ± (0.076)	4.92 ± (0.058)	4.94 ± (0.047)	4.95 ± (0.096)
YouTube	1×10^4	3.75 ± (0.162)	3.71 ± (0.191)	3.77 ± (0.373)	3.59 ± (0.312)	3.43 ± (0.169)	3.90 ± (0.171)
	2×10^4	3.16 ± (0.193)	3.17 ± (0.105)	3.24 ± (0.255)	3.06 ± (0.188)	2.94 ± (0.136)	3.32 ± (0.107)
	3×10^4	2.95 ± (0.204)	2.93 ± (0.089)	2.95 ± (0.226)	2.82 ± (0.181)	2.74 ± (0.102)	3.01 ± (0.137)
	4×10^4	2.78 ± (0.153)	2.74 ± (0.078)	2.77 ± (0.222)	2.66 ± (0.133)	2.62 ± (0.106)	2.82 ± (0.126)
	5×10^4	2.64 ± (0.154)	2.63 ± (0.059)	2.65 ± (0.220)	2.54 ± (0.123)	2.53 ± (0.112)	2.65 ± (0.100)
	6×10^4	2.55 ± (0.144)	2.54 ± (0.058)	2.57 ± (0.206)	2.43 ± (0.109)	2.44 ± (0.096)	2.55 ± (0.081)
	7×10^4	2.45 ± (0.125)	2.45 ± (0.062)	2.48 ± (0.181)	2.35 ± (0.091)	2.36 ± (0.095)	2.46 ± (0.090)
	8×10^4	2.38 ± (0.129)	2.38 ± (0.055)	2.40 ± (0.168)	2.29 ± (0.077)	2.29 ± (0.087)	2.38 ± (0.088)
	9×10^4	2.32 ± (0.126)	2.33 ± (0.034)	2.35 ± (0.159)	2.22 ± (0.081)	2.23 ± (0.083)	2.31 ± (0.098)
	10×10^4	2.26 ± (0.124)	2.27 ± (0.034)	2.28 ± (0.159)	2.15 ± (0.074)	2.17 ± (0.078)	2.25 ± (0.112)
Wikipedia	1×10^5	4.19 ± (0.174)	4.14 ± (0.201)	4.35 ± (0.150)	3.98 ± (0.190)	4.02 ± (0.134)	4.27 ± (0.149)
	2×10^5	3.61 ± (0.149)	3.56 ± (0.161)	3.72 ± (0.101)	3.51 ± (0.187)	3.48 ± (0.070)	3.73 ± (0.073)
	3×10^5	3.22 ± (0.131)	3.18 ± (0.155)	3.26 ± (0.092)	3.14 ± (0.095)	3.11 ± (0.087)	3.32 ± (0.053)
	4×10^5	2.93 ± (0.110)	2.92 ± (0.102)	2.97 ± (0.083)	2.89 ± (0.056)	2.87 ± (0.081)	3.04 ± (0.044)
	5×10^5	2.75 ± (0.082)	2.75 ± (0.070)	2.80 ± (0.064)	2.71 ± (0.049)	2.70 ± (0.056)	2.84 ± (0.033)
	6×10^5	2.63 ± (0.067)	2.62 ± (0.052)	2.66 ± (0.053)	2.59 ± (0.056)	2.58 ± (0.038)	2.71 ± (0.020)
	7×10^5	2.52 ± (0.060)	2.52 ± (0.046)	2.54 ± (0.044)	2.48 ± (0.060)	2.48 ± (0.029)	2.60 ± (0.027)
	8×10^5	2.43 ± (0.058)	2.43 ± (0.046)	2.45 ± (0.037)	2.40 ± (0.064)	2.40 ± (0.023)	2.50 ± (0.040)
	9×10^5	2.35 ± (0.057)	2.35 ± (0.034)	2.38 ± (0.037)	2.32 ± (0.062)	2.33 ± (0.023)	2.41 ± (0.042)
	10×10^5	2.28 ± (0.056)	2.28 ± (0.034)	2.31 ± (0.038)	2.25 ± (0.062)	2.26 ± (0.014)	2.33 ± (0.045)

Table 2
Accuracy (%) comparison with 95% confidence intervals in Exp-1.3.

Dataset	Network Size	NMF(Node)	NMF(Edge)	NMF(Biased)	NMF(Node+)	NMF(Edge+)	NMF(Biased+)
Digg	3×10^4	4.19 ± (0.110)	4.22 ± (0.138)	4.07 ± (0.095)	4.13 ± (0.069)	4.14 ± (0.095)	4.00 ± (0.069)
	6×10^4	4.85 ± (0.125)	4.86 ± (0.104)	4.81 ± (0.107)	4.77 ± (0.073)	4.78 ± (0.153)	4.71 ± (0.101)
	9×10^4	4.89 ± (0.076)	4.88 ± (0.081)	4.90 ± (0.087)	4.86 ± (0.071)	4.88 ± (0.124)	4.83 ± (0.126)
	12×10^4	4.89 ± (0.077)	4.91 ± (0.107)	4.87 ± (0.141)	4.90 ± (0.060)	4.93 ± (0.043)	4.90 ± (0.056)
	15×10^4	4.95 ± (0.063)	4.92 ± (0.077)	4.96 ± (0.128)	4.93 ± (0.063)	4.95 ± (0.072)	4.88 ± (0.076)
YouTube	3×10^5	2.01 ± (0.038)	2.07 ± (0.103)	2.01 ± (0.058)	2.04 ± (0.072)	2.03 ± (0.044)	1.99 ± (0.091)
	6×10^5	2.17 ± (0.016)	2.19 ± (0.084)	2.22 ± (0.046)	2.13 ± (0.104)	2.13 ± (0.070)	2.10 ± (0.069)
	9×10^5	2.21 ± (0.147)	2.25 ± (0.141)	2.23 ± (0.110)	2.13 ± (0.089)	2.16 ± (0.059)	2.17 ± (0.030)
	12×10^5	2.31 ± (0.098)	2.27 ± (0.087)	2.32 ± (0.155)	2.13 ± (0.066)	2.15 ± (0.078)	2.18 ± (0.059)
	15×10^5	2.28 ± (0.186)	2.29 ± (0.123)	2.33 ± (0.093)	2.17 ± (0.150)	2.18 ± (0.050)	2.22 ± (0.123)
Wikipedia	3×10^5	2.07 ± (0.030)	2.06 ± (0.037)	2.12 ± (0.024)	2.03 ± (0.037)	2.03 ± (0.046)	2.10 ± (0.014)
	6×10^5	2.19 ± (0.041)	2.21 ± (0.038)	2.26 ± (0.041)	2.17 ± (0.030)	2.14 ± (0.041)	2.18 ± (0.042)
	9×10^5	2.23 ± (0.029)	2.25 ± (0.027)	2.27 ± (0.029)	2.21 ± (0.042)	2.20 ± (0.023)	2.27 ± (0.020)
	12×10^5	2.24 ± (0.036)	2.26 ± (0.028)	2.25 ± (0.043)	2.17 ± (0.042)	2.21 ± (0.044)	2.28 ± (0.042)
	15×10^5	2.29 ± (0.037)	2.26 ± (0.057)	2.30 ± (0.020)	2.23 ± (0.030)	2.24 ± (0.061)	2.28 ± (0.024)

and $r = 20$ on Wikipedia for bagging+ and bagging methods.

Remarks. From these experimental results, we find that k has great impacts on predicting links:

(1) The bagging+ and bagging methods are more accurate than the other methods when k is varied from hundreds to the number of links in the ground truth data, except for RA on Wikipedia. Particularly, the bagging+ and bagging methods have significant improvements when k is small, and also more accurate than the others when k is even the number of links in the ground truth data.

(2) NMF(Biased) is more accurate than NMF(Node) and NMF(Edge) when k is small, and very close to the NMF(Node) and NMF(Edge) when k is large, *e.g.,* k is fixed to the number of links in the ground truth data. In this case, however, it also provides an efficiency advantage. Similar trends have been found for NMF(Biased+).

(3) The accuracy of all methods decreases with the increase of k, and becomes very low when k is very large. For instance, when k is the number of links in the ground truth data, the accuracy of AA and RA is only 1% on YouTube. Although our methods are more accurate than the other methods under this condition, we still keep the default value for k since a higher accuracy would be more useful in practice.

2 APPENDIX B: ACCURACY RESULTS WITH CONFIDENCE INTERVALS

We report the average of accuracy for comparison in Section 4, and here we further report the accuracy with 95% confidence intervals.

The confidence interval of accuracy is given by

$$(\overline{x} - t_{[n-1,\alpha/2]}\frac{s}{\sqrt{n}}, \overline{x} + t_{[n-1,\alpha/2]}\frac{s}{\sqrt{n}})$$

Table 3
Accuracy (%) comparison with 95% confidence intervals in Exp-2.1.

Dataset	k	AA	RA	BIGCLAM	NMF	NMF(Biased)	NMF(Biased+)
Digg	1×10^4	7.58	3.63	5.64 ± (0.836)	8.37 ± (0.361)	9.33 ± (0.343)	9.59 ± (0.197)
	2×10^4	6.19	3.18	5.08 ± (0.701)	7.31 ± (0.194)	7.88 ± (0.304)	8.00 ± (0.141)
	3×10^4	5.40	3.02	4.72 ± (0.507)	6.61 ± (0.163)	7.07 ± (0.190)	7.13 ± (0.108)
	4×10^4	4.87	2.86	4.46 ± (0.454)	6.17 ± (0.152)	6.52 ± (0.162)	6.59 ± (0.064)
	5×10^4	4.50	2.69	4.26 ± (0.396)	5.81 ± (0.104)	6.13 ± (0.083)	6.17 ± (0.075)
	6×10^4	4.17	2.34	4.09 ± (0.368)	5.53 ± (0.094)	5.79 ± (0.082)	5.81 ± (0.082)
	7×10^4	3.94	2.30	3.98 ± (0.341)	5.32 ± (0.079)	5.53 ± (0.101)	5.55 ± (0.108)
	8×10^4	3.77	2.33	3.86 ± (0.304)	5.13 ± (0.106)	5.32 ± (0.093)	5.31 ± (0.097)
	9×10^4	3.62	2.32	3.77 ± (0.272)	4.97 ± (0.079)	5.12 ± (0.062)	5.11 ± (0.090)
	10×10^4	3.50	2.32	3.70 ± (0.256)	4.83 ± (0.072)	4.95 ± (0.076)	4.95 ± (0.096)
YouTube	1×10^4	2.10	2.16	1.93 ± (0.203)	2.50 ± (0.233)	3.77 ± (0.373)	3.90 ± (0.171)
	2×10^4	2.00	1.92	1.80 ± (0.242)	2.51 ± (0.188)	3.24 ± (0.255)	3.32 ± (0.107)
	3×10^4	1.88	1.75	1.78 ± (0.254)	2.44 ± (0.097)	2.95 ± (0.226)	3.01 ± (0.137)
	4×10^4	1.81	1.71	1.73 ± (0.250)	2.41 ± (0.104)	2.77 ± (0.222)	2.82 ± (0.126)
	5×10^4	1.73	1.65	1.68 ± (0.215)	2.38 ± (0.103)	2.65 ± (0.220)	2.65 ± (0.100)
	6×10^4	1.73	1.68	1.64 ± (0.210)	2.32 ± (0.097)	2.57 ± (0.206)	2.55 ± (0.081)
	7×10^4	1.68	1.66	1.63 ± (0.213)	2.26 ± (0.072)	2.48 ± (0.181)	2.46 ± (0.090)
	8×10^4	1.65	1.65	1.60 ± (0.197)	2.22 ± (0.062)	2.40 ± (0.168)	2.38 ± (0.088)
	9×10^4	1.62	1.65	1.58 ± (0.193)	2.18 ± (0.089)	2.35 ± (0.159)	2.31 ± (0.098)
	10×10^4	1.57	1.65	1.56 ± (0.190)	2.12 ± (0.090)	2.28 ± (0.159)	2.25 ± (0.112)
Wikipedia	1×10^5	2.13	3.61	2.60 ± (0.146)	3.67 ± (0.136)	4.35 ± (0.150)	4.27 ± (0.149)
	2×10^5	2.24	3.40	2.42 ± (0.151)	2.80 ± (0.071)	3.72 ± (0.101)	3.73 ± (0.073)
	3×10^5	2.39	3.21	2.28 ± (0.190)	2.53 ± (0.062)	3.26 ± (0.092)	3.32 ± (0.053)
	4×10^5	2.38	3.10	2.17 ± (0.155)	2.43 ± (0.114)	2.97 ± (0.083)	3.04 ± (0.044)
	5×10^5	2.33	3.10	2.08 ± (0.127)	2.38 ± (0.152)	2.80 ± (0.064)	2.84 ± (0.033)
	6×10^5	2.29	3.10	2.01 ± (0.127)	2.30 ± (0.136)	2.66 ± (0.053)	2.71 ± (0.020)
	7×10^5	2.25	3.10	1.95 ± (0.114)	2.22 ± (0.125)	2.54 ± (0.044)	2.60 ± (0.027)
	8×10^5	2.22	3.10	1.90 ± (0.090)	2.16 ± (0.116)	2.45 ± (0.037)	2.50 ± (0.040)
	9×10^5	2.18	3.10	1.85 ± (0.114)	2.10 ± (0.115)	2.38 ± (0.037)	2.41 ± (0.042)
	10×10^5	2.14	3.09	1.82 ± (0.100)	2.05 ± (0.109)	2.31 ± (0.038)	2.33 ± (0.045)

Table 4
Accuracy (%) comparison with 95% confidence intervals in Exp-2.2.

Dataset	Network Size	AA	RA	BIGCLAM	NMF	NMF(Biased)	NMF(Biased+)
Digg	3×10^4	3.39	2.84	3.87 ± (0.240)	4.40 ± (0.095)	4.07 ± (0.095)	4.00 ± (0.069)
	6×10^4	3.50	2.54	3.08 ± (0.133)	4.89 ± (0.154)	4.81 ± (0.107)	4.71 ± (0.101)
	9×10^4	3.50	2.40	2.98 ± (0.219)	4.81 ± (0.083)	4.90 ± (0.087)	4.83 ± (0.126)
	12×10^4	3.50	2.37	3.08 ± (0.308)	4.85 ± (0.132)	4.87 ± (0.141)	4.90 ± (0.056)
	15×10^4	3.50	2.31	2.82 ± (0.161)	4.84 ± (0.278)	4.96 ± (0.128)	4.88 ± (0.076)
YouTube	3×10^5	1.53	1.22	1.43 ± (0.052)	1.87 ± (0.402)	2.01 ± (0.058)	1.99 ± (0.091)
	6×10^5	1.58	1.53	1.67 ± (0.103)	2.10 ± (0.725)	2.22 ± (0.046)	2.10 ± (0.069)
	9×10^5	1.58	1.53	1.56 ± (0.080)	1.91 ± (0.305)	2.23 ± (0.110)	2.17 ± (0.030)
	12×10^5	1.57	1.55	1.35 ± (0.186)	2.02 ± (0.390)	2.32 ± (0.155)	2.18 ± (0.059)
	15×10^5	1.57	1.65	1.56 ± (0.076)	2.18 ± (0.388)	2.33 ± (0.093)	2.22 ± (0.123)
Wikipedia	3×10^5	1.71	2.48	1.01 ± (0.029)	2.13 ± (0.065)	2.12 ± (0.024)	2.10 ± (0.014)
	6×10^5	1.87	2.67	1.94 ± (0.063)	2.05 ± (0.063)	2.26 ± (0.041)	2.18 ± (0.042)
	9×10^5	2.04	2.77	1.92 ± (0.063)	2.06 ± (0.059)	2.27 ± (0.029)	2.27 ± (0.020)
	12×10^5	2.12	2.93	1.86 ± (0.043)	2.15 ± (0.039)	2.25 ± (0.043)	2.28 ± (0.042)
	15×10^5	2.14	3.06	1.79 ± (0.123)	2.13 ± (0.122)	2.30 ± (0.020)	2.28 ± (0.024)

Table 5
Accuracy (%) comparison with 95% confidence intervals in Exp-4.1.

Dataset	k	NMF(Node)	NMF(Edge)	NMF(Biased)	NMF(Node+)	NMF(Edge+)	NMF(Biased+)
Digg	467	15.46 ± (1.503)	14.18 ± (1.003)	16.75 ± (0.927)	18.33 ± (0.933)	18.20 ± (1.176)	18.76 ± (1.040)
	4,678	10.69 ± (0.266)	10.38 ± (0.414)	11.55 ± (0.468)	11.36 ± (0.322)	11.03 ± (0.266)	11.66 ± (0.415)
	46,781	6.19 ± (0.100)	6.11 ± (0.133)	6.32 ± (0.185)	6.24 ± (0.111)	6.22 ± (0.096)	6.28 ± (0.112)
	467,816	2.91 ± (0.027)	2.92 ± (0.025)	2.94 ± (0.041)	2.86 ± (0.044)	2.89 ± (0.024)	2.89 ± (0.041)
YouTube	806	4.86 ± (0.295)	4.76 ± (1.025)	5.16 ± (0.929)	5.33 ± (1.074)	5.31 ± (0.482)	5.68 ± (0.300)
	8,062	3.31 ± (0.229)	3.34 ± (0.286)	3.50 ± (0.181)	3.51 ± (0.296)	3.55 ± (0.086)	3.66 ± (0.138)
	80,621	2.17 ± (0.121)	2.22 ± (0.108)	2.26 ± (0.122)	2.35 ± (0.099)	2.37 ± (0.111)	2.33 ± (0.072)
	806,213	1.31 ± (0.043)	1.34 ± (0.038)	1.34 ± (0.059)	1.34 ± (0.043)	1.38 ± (0.052)	1.37 ± (0.062)
Wikipedia	585	3.59 ± (0.334)	3.56 ± (0.690)	3.76 ± (0.716)	3.56 ± (0.406)	3.69 ± (0.608)	3.42 ± (0.395)
	5,856	4.70 ± (0.334)	4.59 ± (0.305)	4.47 ± (0.118)	4.39 ± (0.236)	4.38 ± (0.180)	4.59 ± (0.277)
	58,565	4.05 ± (0.122)	3.92 ± (0.142)	3.99 ± (0.169)	3.97 ± (0.271)	3.84 ± (0.141)	4.04 ± (0.162)
	585,659	2.74 ± (0.030)	2.69 ± (0.044)	2.74 ± (0.075)	2.73 ± (0.057)	2.70 ± (0.061)	2.77 ± (0.056)
	5,856,596	1.30 ± (0.012)	1.30 ± (0.020)	1.30 ± (0.029)	1.30 ± (0.011)	1.29 ± (0.021)	1.32 ± (0.014)

Table 6
Accuracy (%) comparison with 95% confidence intervals in Exp-4.2.

Dataset	Network Size	NMF(Node)	NMF(Edge)	NMF(Biased)	NMF(Node+)	NMF(Edge+)	NMF(Biased+)
Digg	3×10^4	4.28 ± (0.099)	4.27 ± (0.070)	4.15 ± (0.126)	4.22 ± (0.099)	4.13 ± (0.112)	3.98 ± (0.091)
	6×10^4	3.78 ± (0.075)	3.73 ± (0.103)	3.69 ± (0.060)	3.68 ± (0.050)	3.57 ± (0.117)	3.53 ± (0.040)
	9×10^4	3.49 ± (0.027)	3.49 ± (0.067)	3.43 ± (0.089)	3.39 ± (0.025)	3.39 ± (0.043)	3.36 ± (0.063)
	12×10^4	3.42 ± (0.044)	3.36 ± (0.049)	3.34 ± (0.066)	3.33 ± (0.029)	3.30 ± (0.030)	3.25 ± (0.026)
	15×10^4	3.22 ± (0.026)	3.15 ± (0.050)	3.16 ± (0.047)	3.12 ± (0.018)	3.12 ± (0.019)	3.04 ± (0.048)
YouTube	3×10^5	1.72 ± (0.058)	1.76 ± (0.034)	1.67 ± (0.014)	1.71 ± (0.037)	1.73 ± (0.044)	1.65 ± (0.026)
	6×10^5	1.64 ± (0.084)	1.58 ± (0.069)	1.52 ± (0.097)	1.54 ± (0.062)	1.54 ± (0.070)	1.49 ± (0.045)
	9×10^5	1.45 ± (0.064)	1.49 ± (0.079)	1.40 ± (0.078)	1.43 ± (0.022)	1.46 ± (0.059)	1.42 ± (0.044)
	12×10^5	1.39 ± (0.022)	1.49 ± (0.103)	1.32 ± (0.035)	1.40 ± (0.055)	1.40 ± (0.078)	1.36 ± (0.053)
	15×10^5	1.34 ± (0.051)	1.37 ± (0.051)	1.34 ± (0.071)	1.35 ± (0.042)	1.38 ± (0.050)	1.31 ± (0.040)
Wikipedia	3×10^5	2.38 ± (0.042)	2.35 ± (0.052)	2.43 ± (0.023)	2.37 ± (0.026)	2.34 ± (0.032)	2.43 ± (0.039)
	6×10^5	1.94 ± (0.045)	1.90 ± (0.050)	1.93 ± (0.027)	1.90 ± (0.029)	1.90 ± (0.054)	1.93 ± (0.034)
	9×10^5	1.70 ± (0.032)	1.70 ± (0.045)	1.73 ± (0.020)	1.70 ± (0.026)	1.69 ± (0.024)	1.73 ± (0.025)
	12×10^5	1.53 ± (0.026)	1.55 ± (0.020)	1.56 ± (0.024)	1.52 ± (0.022)	1.52 ± (0.032)	1.56 ± (0.018)
	15×10^5	1.37 ± (0.018)	1.38 ± (0.035)	1.38 ± (0.021)	1.38 ± (0.032)	1.38 ± (0.010)	1.39 ± (0.031)

Table 7
Accuracy (%) comparison with 95% confidence intervals in Exp-5.1.

Dataset	k	AA	RA	BIGCLAM	NMF	NMF(Biased)	NMF(Biased+)
Digg	467	15.42	3.85	4.37 ± (1.393)	12.16 ± (1.178)	16.75 ± (0.927)	18.76 ± (1.040)
	4,678	9.66	4.10	6.40 ± (1.487)	9.27 ± (0.631)	11.55 ± (0.468)	11.66 ± (0.415)
	46,781	4.59	2.77	4.30 ± (0.403)	5.80 ± (0.312)	6.32 ± (0.185)	6.28 ± (0.112)
	467,816	2.35	1.71	2.61 ± (0.101)	2.82 ± (0.121)	2.94 ± (0.041)	2.89 ± (0.041)
YouTube	806	2.85	2.73	1.99 ± (0.391)	2.66 ± (1.074)	5.16 ± (0.929)	5.68 ± (0.300)
	8,062	2.28	2.27	1.95 ± (0.241)	2.46 ± (0.296)	3.50 ± (0.181)	3.66 ± (0.138)
	80,621	1.64	1.66	1.61 ± (0.201)	2.21 ± (0.099)	2.26 ± (0.122)	2.33 ± (0.072)
	806,213	1.00	1.00	1.09 ± (0.107)	1.32 ± (0.059)	1.34 ± (0.059)	1.37 ± (0.062)
Wikipedia	585	3.76	3.93	4.03 ± (0.593)	3.15 ± (0.411)	3.76 ± (0.716)	3.42 ± (0.395)
	5,856	3.50	3.40	3.60 ± (0.405)	3.57 ± (0.383)	4.47 ± (0.118)	4.59 ± (0.277)
	58,565	2.70	3.63	2.82 ± (0.126)	4.03 ± (0.275)	3.99 ± (0.169)	4.04 ± (0.162)
	585,659	2.29	3.10	2.02 ± (0.046)	2.40 ± (0.082)	2.74 ± (0.075)	2.77 ± (0.056)
	5,856,596	1.56	2.69	1.16 ± (0.018)	1.20 ± (0.031)	1.30 ± (0.029)	1.32 ± (0.014)

Table 8
Accuracy (%) comparison with 95% confidence intervals in Exp-5.2.

Dataset	Network Size	AA	RA	BIGCLAM	NMF	NMF(Biased)	NMF(Biased+)
Digg	3×10^4	3.45	2.89	3.94 ± (0.238)	4.39 ± (0.089)	4.15 ± (0.126)	3.98 ± (0.091)
	6×10^4	2.82	2.13	2.62 ± (0.112)	3.73 ± (0.103)	3.69 ± (0.060)	3.53 ± (0.040)
	9×10^4	2.67	1.97	2.43 ± (0.100)	3.41 ± (0.128)	3.43 ± (0.089)	3.36 ± (0.063)
	12×10^4	2.59	1.90	2.45 ± (0.178)	3.33 ± (0.064)	3.34 ± (0.066)	3.25 ± (0.026)
	15×10^4	2.49	1.76	2.15 ± (0.103)	3.10 ± (0.104)	3.16 ± (0.047)	3.04 ± (0.048)
YouTube	3×10^5	1.32	1.01	1.23 ± (0.073)	1.57 ± (0.098)	1.67 ± (0.014)	1.65 ± (0.026)
	6×10^5	1.15	1.00	1.28 ± (0.021)	1.37 ± (0.301)	1.52 ± (0.097)	1.49 ± (0.045)
	9×10^5	1.08	0.89	1.14 ± (0.052)	1.38 ± (0.151)	1.40 ± (0.078)	1.42 ± (0.044)
	12×10^5	1.02	0.87	1.04 ± (0.072)	1.39 ± (0.138)	1.32 ± (0.035)	1.36 ± (0.053)
	15×10^5	1.00	1.00	1.06 ± (0.025)	1.38 ± (0.086)	1.34 ± (0.071)	1.31 ± (0.040)
Wikipedia	3×10^5	1.78	2.54	1.05 ± (0.014)	2.25 ± (0.050)	2.43 ± (0.023)	2.43 ± (0.039)
	6×10^5	1.68	2.50	1.64 ± (0.034)	1.77 ± (0.074)	1.93 ± (0.027)	1.93 ± (0.034)
	9×10^5	1.69	2.54	1.50 ± (0.027)	1.49 ± (0.088)	1.73 ± (0.020)	1.73 ± (0.025)
	12×10^5	1.69	2.64	1.34 ± (0.012)	1.40 ± (0.053)	1.56 ± (0.024)	1.56 ± (0.018)
	15×10^5	1.61	2.73	1.21 ± (0.029)	1.25 ± (0.083)	1.38 ± (0.021)	1.39 ± (0.031)

where $\overline{x}$ is the average of the accuracy, α is the significance level (we set α to 0.05 for the $1-\alpha = 95\%$ confidence level), s is the sample standard deviation, n is the number of repeated times of our experiments and $t_{[n-1,\alpha/2]}$ is the $(\alpha/2)$-quantile of Student's t distribution with $n-1$ degrees of freedom [**?**]. The confidence interval gives an indication of how much uncertainty there is in the estimate of the accuracy. The narrower the interval, the more precise is the estimate.

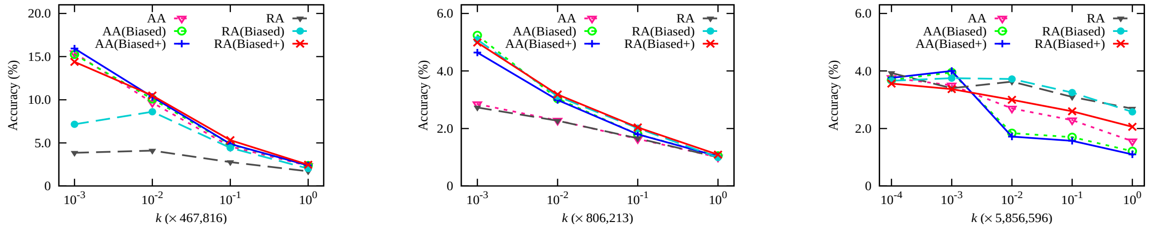
The accuracy results in Exp-1.2, Exp-1.3, Exp-2.1 and Exp-2.2 are shown in Table 1 – Table 4, respectively. Further, the accuracy results in Exp-4.1, Exp-4.2, Exp-5.1 and Exp-5.2 are shown in Table 5 – Table 8, respectively.

Remarks. From the results of accuracy comparison with confidence intervals, we find the following.

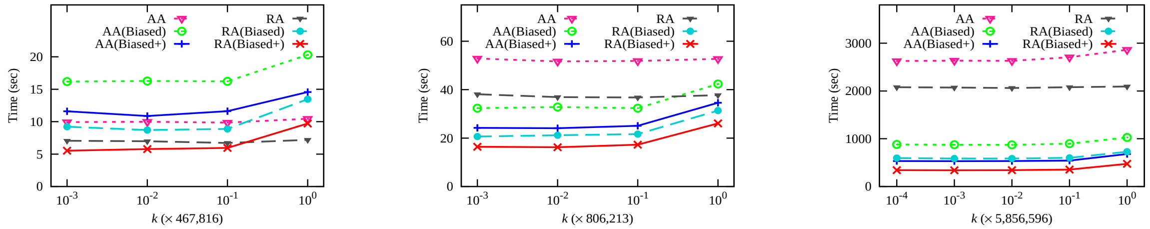
(1) NMF(Biased) and NMF(Biased+) are more accurate than the other methods on most datasets. Even when k is the number of the ground truth links (See in Table 7), NMF(Biased) improves the accuracy by $(2.8\%, 24.3\%, 70.8\%, 11.9\%)$ (*resp.* $(1.5\%, 34.0\%, 34.0\%, 22.9\%)$) over NMF, AA, RA and BIGCLAM on Digg and YouTube, respectively. Moreover, bagging+ and bagging methods perform better than the other methods on most datasets.

(2) The accuracy of all methods decreases with the increase of k, which is consistent with the previous experimental analysis.

(3) The confidence intervals of all methods are narrow, which means that the accuracy estimate is reasonable. Thus, we only

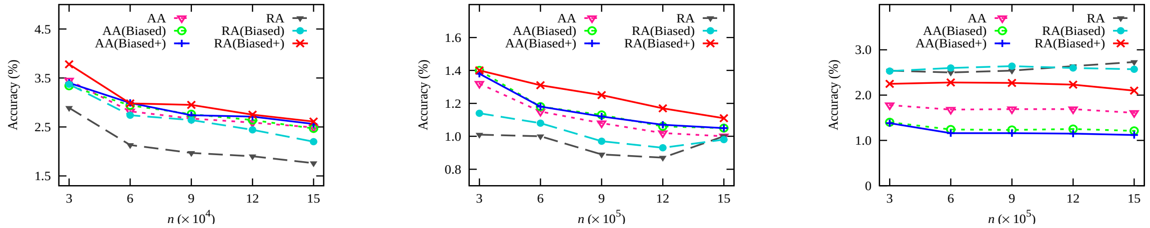


(a) Digg (b) YouTube (c) Wikipedia

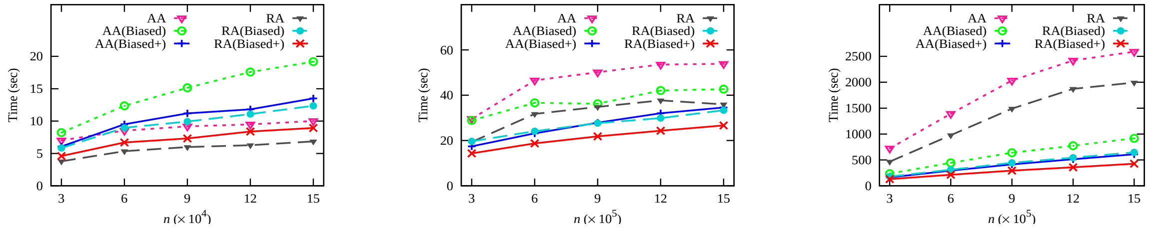


(d) Digg (e) YouTube (f) Wikipedia

Figure 6. Ensemble-Enabled AA & RA on accuracy and efficiency: with respect to the number k of predicted links.



(a) Digg (b) YouTube (c) Wikipedia



(d) Digg (e) YouTube (f) Wikipedia

Figure 7. Ensemble-Enabled AA & RA on accuracy and efficiency: with respect to the network sizes.

report the average accuracy for clarity.

3 APPENDIX C: ENSEMBLE-ENABLED AA & RA

One interesting thing should be noted that our ensemble-enabled method is in principle a general method for decomposing the large network link prediction problem into smaller subproblems. It can not only be applied for NMF, and may be applied to other prediction methods. Therefore, we implemented AA(Biased) and AA(Biased+) (*resp.*RA(Biased) and RA(Biased+)) by replacing NMF with AA (*resp.*RA) in NMF(Biased) and NMF(Biased+). Furthermore, we evaluated the effectiveness and efficiency of the ensemble-enabled approach with AA and RA.

Exp-7.1: Impacts of k. We varied k from hundreds to the number of links in the ground truth data and fixed other parameters to their default values. The results of accuracy and efficiency are reported in Figures 6(a)–6(c) and Figures 6(d)–6(f), respectively.

The accuracy results tell us that (a) AA(Biased) and AA(Biased+) (*resp.*RA(Biased) and RA(Biased+)) are more accurate than AA (*resp.*RA) on most datasets, (b) the accuracy of AA(Biased+) (*resp.*RA(Biased+)) is close to that of AA(Biased) (*resp.*RA(Biased)), and (c) the improvements of the ensemble-enabled methods are significant when k is small.

Note that AA(Biased) and AA(Biased+) (*resp.*RA(Biased+)) perform worse than their counterpart AA (*resp.*RA) on Wikipedia because their diversities on this dataset are poor. For instance, the pairwise overlapping of predicted links of AA(Biased) (*resp.*AA(Biased+) and RA(Biased+)) is 0.62 (*resp.* 0.63 and 0.52). One reason for the decreasing of diversity may be that the dataset contains some extreme high degree nodes, *i.e.,* nodes with degree between 5×10^4 to 1.8×10^5 while the network contains only 1.6×10^6 nodes.

The efficiency results tell us that AA(Biased) and AA(Biased+) (*resp.*RA(Biased) and RA(Biased+)) are faster than AA (*resp.*RA) on YouTube and Wikipedia but slower on Digg. This is consistent with the complexity analysis that the ensemble-enabled AA and RA require $O(nd_1^2 \log(k)\mu/f)$ time, where d_1 is the average degree of each ensemble component. It is means that the ensemble-enabled AA and RA would be faster when d_1 is small. Indeed, d_1 of AA(Biased) is 9 and 56 on YouTube and Wikipedia while the average degree of these datasets is 5 and 33 respectively. However, d_1 of AA(Biased) is 30 on Digg while the average degree of this datasets is 10.

Exp-7.2: Impacts of network sizes. To evaluate the impacts of network sizes, we used networks with different sizes that used in Exp-1.3, fixed k to the number of links in the ground truth data and fixed other parameters to their default values. The results of accuracy and efficiency are reported in 7(a)–7(c) and Figures 7(d)–7(f), respectively.

The accuracy results tell us that (a) the ensemble-enabled methods are more accuracy than their counterparts AA and RA on most of the datasets, and (b) the improvements of RA(Biased) and RA(Biased+) on Digg and YouTube are significant. This means that the ensemble-enabled methods are accurate and robust with the increase of network sizes.

The efficiency results tell us that (a) the ensemble-enabled methods are faster than their counterparts AA and RA on most of the datasets, (b) the bagging+ methods are faster than their counterparts bagging methods, and (c) the running time of all methods increase nearly linearly with the increase of network sizes. Note that we also tested the ensemble-enabled approach with AA and RA on the large networks Twitter and Friendster, and the average speedup of AA(Biased) and AA(Biased+) (*resp.*RA(Biased) and RA(Biased+)) are (5, 4) and (14, 5) (*resp.* (7, 5) and (20, 6)) on (Twitter and Friendster), which justifies the efficiency of our ensemble-enabled approach.

Remarks. From these results, we find the following.

(1) These results show that ensemble-enabled AA and RA are very promising. The ensemble-enabled AA and RA improve the accuracy compared with their counterparts AA and RA on some networks; The ensemble-enabled AA and RA have an efficiency advantage when the average degree of each ensemble component is small, *e.g.,* on YouTube and Wikipedia.

(2) Although our ensemble-enabled approach is a general framework that may be applied to any link prediction methods, the sampling techniques for generating ensembles should be redesigned for different link prediction methods, such as AA and RA to fully employ the advantage of the framework.

REFERENCES

[1] G. Casella and R. L. Berger. *Statistical Inference (2nd edition).* Duxbury Press, 2001.

[2] Y. Yang, R. N. Lichtenwalter, and N. V. Chawla. Evaluating link prediction methods. *Knowl Inf Syst*, 45:751–782, 2015.