A Fast Active Learning Algorithm for Link Classification[⋆]

Nicolò Cesa-Bianchi[1], Claudio Gentile[2], Fabio Vitale[3], and Giovanni Zappella[4]

[1] Dipartimento di Informatica, Università degli Studi di Milano, Italy
`nicolo.cesa-bianchi@unimi.it`
[2] Dipartimento di Scienze Teoriche ed Applicate, Università dell'Insubria, Italy
`claudio.gentile@uninsubria.it`
[3] Dipartimento di Informatica, Università degli Studi di Milano, Italy
`fabio.vitale@unimi.it`
[4] Dipartimento di Matematica, Università degli Studi di Milano, Italy
`giovanni.zappella@unimi.it`

Abstract. We present a very efficient active learning algorithm for link classification in signed networks. Our algorithm is motivated by a stochastic model in which edge labels are obtained through perturbations of an initial sign assignment consistent with a two-clustering of the nodes. We provide a theoretical analysis within this model, showing that we can achieve an optimal (to within a constant factor) number of mistakes on any graph $G = (V, E)$ such that $|E| = \Omega(|V|^{3/2})$ by querying $\mathcal{O}(|V|^{3/2})$ edge labels. More generally, we show an algorithm that achieves optimality to within a factor of $\mathcal{O}(k)$ by querying at most order of $|V| + (|V|/k)^{3/2}$ edge labels. The running time of this algorithm is at most of order $|E| + |V| \log |V|$.

1 Introduction

A rapidly emerging theme in the analysis of networked data is the study of signed networks. From a formal viewpoint, signed networks are graphs whose edges host a sign encoding the positive or negative nature of the relationship between the incident nodes. E.g., in a protein network two proteins may interact in an excitatory or inhibitory fashion. The domain of social networks and e-commerce offers several examples of signed relationships: Slashdot users can tag other users as friends or foes, Epinions users can rate other users positively or negatively, Ebay users develop trust and distrust towards sellers in the network. More generally, two individuals that are related because they rate similar products in a recommendation website may agree or disagree in their ratings.

The availability of signed networks has stimulated the design of link classification algorithms, especially in the domain of social networks. Early studies of signed social networks are from the Fifties. E.g., [6] and [1] model dislike and distrust relationships among individuals as (signed) weighted edges in a graph. The conceptual underpinning is provided by the theory of *social balance*, formulated as a way to understand the structure of conflicts in a network of individuals

[⋆] This work was supported in part by Google Inc. through a Google Research Award, by the PASCAL2 Network of Excellence under EC grant 216886 and by "Dote ricerca: FSE, Regione Lombardia. This publication only reflects the authors views.

whose mutual relationships can be classified as friendship or hostility [7]. The advent of online social networks has revamped the interest in these theories, and spurred a significant amount of recent work —see, e.g., [5, 8, 10, 3, 2], and references therein.

Many heuristics for link classification in social networks are based on a form of social balance summarized by the motto "the enemy of my enemy is my friend". This is equivalent to saying that the signs on the edges of a social graph tend to be consistent with some two-clustering of the nodes. By consistency we mean the following: The nodes of the graph can be partitioned into two sets (the two clusters) in such a way that edges connecting nodes from the same set are positive, and edges connecting nodes from different sets are negative. Although two-clustering heuristics do not require strict consistency to work, this is admittely a rather strong inductive bias. Despite that, social network theorists and practitioners found this to be a reasonable bias in many social contexts, and recent experiments with online social networks reported a good predictive power for algorithms based on the two-clustering assumption [8–10, 3]. Finally, this assumption is also fairly convenient from the viewpoint of algorithmic design.

In the case of undirected signed graphs $G = (V, E)$, the best performing heuristics exploiting the two-clustering bias are based on spectral decompositions of the signed adiacency matrix. Noticeably, these heuristics run in time $\Omega\bigl(|V|^2\bigr)$, and often require a similar amount of memory storage even on sparse networks, which makes them impractical on large graphs.

In order to obtain scalable algorithms with formal performance guarantees, we focus on the active learning protocol, where training labels are obtained by querying a desired subset of edges. Since the allocation of queries can match the graph topology, a wide range of graph-theoretic techniques can be applied to the analysis of active learning algorithms. In the recent work [2], a simple stochastic model for generating edge labels by perturbing some unknown two-clustering of the graph nodes was introduced. For this model, the authors proved that querying the edges of a low-stretch spanning tree [4] of the input graph $G = (V, E)$ is sufficient to predict the remaining edge labels making a number of mistakes within a factor of order $(\log |V|)^2 \log\log |V|$ from the theoretical optimum. The overall running time is $O(|E| \ln |V|)$. This result leaves two main problems open: First, low-stretch trees are a powerful structure, but the algorithm to construct them is not easy to implement. Second, the tree-based analysis of [2] does not generalize to query budgets larger than $|V| - 1$ (the edge set size of a spanning tree). In this paper we introduce a different active learning approach for link classification that can accomodate a large spectrum of query budgets. We show that on *any* graph with $\Omega(|V|^{3/2})$ edges, a query budget of $\mathcal{O}(|V|^{3/2})$ is sufficient to predict the remaining edge labels within a *constant* factor from the optimum. More in general, we show that a budget of at most order of $|V| + \bigl(\frac{|V|}{k}\bigr)^{3/2}$ queries is sufficient to make a number of mistakes within a factor of $\mathcal{O}(k)$ from the optimum with a running time of order $|E| + (|V|/k)\log(|V|/k)$. Hence, a query budget of $\Theta(|V|)$, of the same order as the algorithm based on low-strech trees, achieves an optimality factor $\mathcal{O}(|V|^{1/3})$ with a running time of just $\mathcal{O}(|E|)$.

2 Results

We consider undirected and connected graphs $G = (V, E)$ with unknown edge labeling $Y_{i,j} \in \{-1, +1\}$ for each $(i, j) \in E$. Edge labels can collectively be represented by the associated *signed* adjacency matrix Y, where $Y_{i,j} = 0$ whenever $(i, j) \notin E$. We define a simple stochastic model for assigning binary labels Y to the edges of G. We assume that edge labels are obtained by perturbing an underlying labeling which is initially consistent with an arbitrary (and unknown) two-clustering. More formally, given an undirected and connected graph $G = (V, E)$, the labels $Y_{i,j} \in \{-1, +1\}$, for $(i, j) \in E$, are assigned as follows. First, the nodes in V are arbitrarily partitioned into two sets, and labels $Y_{i,j}$ are initially assigned consistently with this partition (within-cluster edges are positive and between-cluster edges are negative). Then, given a nonnegative constant $p < \frac{1}{2}$, labels are randomly flipped in such a way that $\mathbb{P}\big(Y_{i,j} \text{ is flipped}\big) \le p$ for each $(i, j) \in E$. We call this a p-stochastic assignment. Note that this model allows for correlations between flipped labels.

A learning algorithm in the link classification setting receives a training set of signed edges and, out of this information, builds a prediction model for the labels of the remaining edges.

Fact 1. *For any training set $E_0 \subset E$ of edges, and any learning algorithm that is given the labels of the edges in E_0, the number M of mistakes made by the algorithm on the remaining $E \setminus E_0$ edges satisfies $\mathbb{E}\, M \ge p \left|E \setminus E_0\right|$, where the expectation is with respect to a p-stochastic assignment of the labels Y.*

An active learner for link classification is a special learning algorithm that first constructs a query set E_0 of edges, and then receives the labels of all edges in the query set. Based on this training information, the learner builds a prediction model for the labels of the remaining edges $E \setminus E_0$. We assume that the only labels ever revealed to the learner are those in the query set, no labels being revealed during the prediction phase. It is clear from Fact 1 that any active learning algorithm that queries the labels of at most a constant fraction of the total number of edges will make on average $\Omega(p|E|)$ mistakes.

Theorem 1. *An active learning algorithm parameterized by interger $k \ge 2$ exists such that for any graph $G = (V, E)$ with $|E| \ge 2|V| - 2 + 2\big(\frac{|V|-1}{k} + 1\big)^{\frac{3}{2}}$, and diameter D_G, the number M of mistakes made by the algorithm on G satisfies $\mathbb{E} M = \mathcal{O}(\min\{k, D_G\})\, p|E|$, while the query set size is bounded by $|V| - 1 + \big(\frac{|V|-1}{k} + 1\big)^{\frac{3}{2}} \le \frac{|E|}{2}$.*

Hence, even if D_G is large, setting $k = |V|^{1/3}$ yields a $\mathcal{O}(|V|^{1/3})$ optimality factor just by querying $\mathcal{O}(|V|)$ edges. On the other hand, a truly constant optimality factor is obtained by querying as few as $\mathcal{O}(|V|^{3/2})$ edges (provided the graph has sufficiently many edges). As a direct consequence (and surprisingly enough), on graphs which are only moderately dense we need not observe too many edges in order to achieve a constant optimality factor. It is instructive to compare the bounds obtained by our algorithm to the ones we can achieve by using the CCCC algorithm of [2], or the low-stretch spanning trees [4].

Because CCCC operates within a harder adversarial setting, it is easy to show that Theorem 9 in [2] extends to the p-stochastic assignment model by replacing $\Delta_2(Y)$ with $p|E|$ therein.[5] The resulting optimality factor is of order $\left(\frac{1-\alpha}{\alpha}\right)^{\frac{3}{2}}\sqrt{|V|}$, where $\alpha \in (0,1]$ is the fraction of queried edges out of the total number of edges. A quick comparison to Theorem 1 reveals that our algorithm achieves a sharper mistake bound for any value of α. For instance, in order to obtain an optimality factor which is lower than $\sqrt{|V|}$, CCCC has to query in the worst case a fraction of edges that goes to one as $|V| \to \infty$.

A low-stretch spanning tree achieves a polylogarithmic optimality factor by querying $|V|-1$ edge labels. The results in [4] show that we cannot hope to get a better optimality factor using a single low-stretch spanning tree combined with the analysis in [2]. For a comparable amount $\Theta(|V|)$ of queried labels, Theorem 1 offers the larger optimality factor $|V|^{1/3}$. However, we can get a *constant* optimality factor by increasing the query set size to $\mathcal{O}(|V|^{3/2})$. It is not clear how multiple low-stretch trees could be combined to get a similar scaling.

Finally, besides being easy to implement, our algorithm is also very fast.

Theorem 2. *For any input graph $G = (V, E)$ which is dense enough to ensure that the query set size is no larger than the test set size, the total time taken by our algorithm for predicting all test labels is $\mathcal{O}\left(|E| + \frac{|V|}{k}\log\frac{|V|}{k}\right)$. In particular, whenever $k|E| = \Omega(|V|\log|V|)$ we have that our algorithm works in constant amortized time. The space required is always linear in the input graph size $|E|$, independent of k.*

References

1. Cartwright, D. and Harary, F. Structure balance: A generalization of Heider's theory. *Psychological review*, 63(5):277–293, 1956.
2. Cesa-Bianchi, N., Gentile, C., Vitale, F., Zappella, G. A correlation clustering approach to link classification in signed networks. In COLT 2012.
3. Chiang, K., Natarajan, N., Tewari, A., and Dhillon, I. Exploiting longer cycles for link prediction in signed networks. In *20th CIKM*, 2011.
4. Elkin, M., Emek, Y., Spielman, D.A., and Teng, S.-H. Lower-stretch spanning trees. *SIAM Journal on Computing*, 38(2):608–628, 2010.
5. Guha, R., Kumar, R., Raghavan, P., and Tomkins, A. Propagation of trust and distrust. In *13th WWW*, pp. 403–412. ACM, 2004.
6. Harary, F. On the notion of balance of a signed graph. *Michigan Mathematical Journal*, 2(2):143–146, 1953.
7. Heider, F. Attitude and cognitive organization. *J. Psychol*, 21:107–122, 1946.
8. Kunegis, J., Lommatzsch, A., and Bauckhage, C. The Slashdot Zoo: Mining a social network with negative edges. In *18th WWW*, 2009.
9. Leskovec, J., Huttenlocher, D., and Kleinberg, J. Signed networks in social media. In *28th ICHFCS*, 2010.
10. Leskovec, J., Huttenlocher, D., and Kleinberg, J. Predicting positive and negative links in online social networks. In *19th WWW*, 2010.

[5] This theoretical comparison is admittedly unfair, as CCCC has been designed to work in a harder setting than p-stochastic. Unfortunately, we are not aware of any other general active learning scheme for link classification to compare with.