Semi-Supervised Learning: A Comparative Study for Web Spam and Telephone User Churn*

András A. Benczúr Károly Csalogány László Lukács Dávid Siklósi

Informatics Laboratory
Computer and Automation Research Institute
Hungarian Academy of Sciences
11 Lagymanyosi u, H-1111 Budapest
and
Eötvös University, Budapest
{benczur, cskaresz, lacko, sdavid}@ilab.sztaki.hu
http://datamining.sztaki.hu/

Abstract. We compare a wide range of semi-supervised learning techniques both for Web spam filtering and for telephone user churn classification. Semi-supervised learning has the assumption that the label of a node in a graph is similar to those of its neighbors. In this paper we measure this phenomenon both for Web spam and telco churn. We conclude that spam is often linked to spam while honest pages are linked to honest ones; similarly churn occurs in bursts in groups of a social network.

1 Introduction

Semi-supervised learning, a new field of machine learning surveyed e.g. in [27] also exploits information from unlabeled data for learning. We focus on the applicability of classifying Web spam and telephone churn, i.e. users who cancel their telephone line. Our assumption is that the label (spam and churned, respectively) of a node in a graph is similar to those of its neighbors.

We compare various means of stacked graphical learning, a meta-learning scheme in which a base learner is augmented by expanding the features of one node with predictions on other related nodes in a graph is introduced recently by Kou and Cohen [15]. The methodology is used with success for Web spam detection in [5]: they use the average label of the neighbors as a new feature for the classifier.

We run our tests on the Web Spam Challenge datasets. The baseline decision tree utilized all graph based features related to a node (i.e. features related to the "home page" or the "maximum PageRank node within site" are not computed) [5] and a Naive Bayes classifier of the machine learning toolkit Weka [24] over the content based features of the Web Spam Challenge Phases I and II data. Depending on the data set the best forms of graph stacking improve the F-measure by 1-10% as shown in Section 3.2.

The other data set we use for evaluating and comparing graph labeling methods is a telephone call graph, a data type that appears less in the publications of the data

* Support from a Yahoo! Faculty Research Grant, projects NKFP-2/0024/2005 and NKFP-2004 Language Miner http://nyelvbanyasz.sztaki.hu.

mining community. Closely related to our work are the churn prediction results by machine learning methods on real data [23, 1, etc.]; these results however do not exploit neighborhood information embedded in the call graph.

The telephone call graph is formed from the call detail record, a log of all calls within a time period including caller and callee id, duration, cost and time stamp. The vertex set consists of all nodes that appear at least once as caller or calle; over this set calls form directed edges from caller to callee.

Churn classification uses customer information (price package, time since in service etc.) as well as traffic aggregates in various call zones and directions. We use one year call detail record and all customer information up to a given time; the classification target consists of users who leave service in the fourth month "in future" (in a time period with no information available for the classifier). Due to the sparsity of positive instances (below 1% churn in a month) and a large amount of churn explained by external reasons such as the customer moves churn classification is a hard task; baseline reaches $F = 0.08$ and this is improved to 0.1 by stacked graphical learning. In the industrial practice the goodness of the churn classification is measured by the recall of the top list of 10% of the customers, i.e. they are willing to involve a maximum of 10% of their customers in direct marketing campaigns and want to maximize the potential churn reached. In this sense our baseline classification has a recall of 40.8%, improved to 47% by stacked graphical learning.

In this paper we concentrate on spreading trust (or no churn) and distrust (churn) information from known nodes with the help of hyperlink based similarity measures. Our main goal is to identify features based on similarities to known honest and spam pages that can be used to classify unknown pages. We propose a set of spam and churn classification methods that combine graph based similarity to labeled nodes [2] with trust and distrust propagation both backward and forward. For example given a link farm alliance [10] with one known target labeled as spam, similarity based features will automatically label other targets as spam as well.

Our stacked graphical learning algorithms generate features by averaging known and predicted labels for similar nodes of the graph by the measures in Section 2.1. We compare various similarity measures, including simple and multi-step neighborhood, co-citation, cosine and Jaccard similarity of the neighborhood as well as their multi-step variants [8] described in detail in Section 2. For the purposes of evaluation we consider these algorithms separately, by performing one classification experiment for each feature.

1.1 Related results

Identifying and preventing spam is cited as one of the top challenges in web search engines in [13]. major search engines incorporate anchor text and link analysis algorithms into their ranking schemes, Web spam appears in sophisticated forms that manipulate content as well as linkage [11]. Spam hunters use a variety of both content [7, 19] and link [12, 6, 25, 3, 2] based features to detect Web spam; a recent measurement of their combination appears in [5].

Recently several results has appeared that apply rank propagation to extend initial trust or distrust judgments over a small set of seed pages or sites to the entire web, such

as trust [12, 26], distrust [20, 6] propagation in the neighborhood or their combination [25] as well as graph based similarity measures [2]. These methods are either based on propagating trust forward or distrust backwards along the hyperlinks based on the idea that honest pages predominantly point to honest ones, or, stated the other way, spam pages are backlinked only by spam pages. Trust and distrust propagation in trust networks originates in Guha et al. [9] for trust networks; Wu et al. [25] is the first to show its applicability for Web spam classification.

Trust and distrust propagation are in fact forms of semi-supervised learning surveyed by Zhu [27], a methodology to exploit unlabeled instances in supervised classification. Stacked graphical learning introduced by Kou and Cohen [15] is a simple implementation that outperforms the computationally expensive variants [15, 5].

Identifying spam pages is somewhat analogous to classifying web documents into multiple topics. Several results [21, and the references therein] demonstrate that classification accuracy can be significantly increased by taking into account the class labels assigned to neighboring nodes. In accordance with [2], Qi and Davison [21] found that most of the improvement comes from the neighborhood defined by co-citation.

Several link-based algorithms were designed to evaluate node-to-node similarities in networks that can be used to give alternate, similarity based weights to node pairs. We refer to [16] for an exhaustive list of the available methods ranging from co-citation to more complex measures such as max-flow/min-cut-based similarities of [17] in the vicinity graph of the query. Co-citation is in fact used in [9] as an elementary step of trust propagation. Another method [18] penalizes the biconnected component of a spam page in a subgraph obtained by backward distrust propagation.

2 The stacked graphical learning framework

2.1 Feature generation

For a given unknown node u and edge weight function w (that may be in or out-degree, cocitation, PageRank etc.), our algorithm selects the k largest weight neighbors of u to generate a new feature based on the known spam and honest hosts in this set. As in [2] we extract four different features from this set of size k or possibly less if u has less than k neighbors. Each element v is either classified as spam with weight $p(v)$ or else labeled spam or nonspam; in these cases we let $p(v)$ be 0 and 1, respectively. Let s and h be the sum of $p(v)$ and $1-p(v)$ in the set; remember $s+h < k$ is possible. We define a weighted version s^* and h^* as the sum of $w(uv) \cdot p(v)$ and $w(uv) \cdot (1-p(v))$.

We define our features as follows.

- Spam Ratio (SR): fraction of the number of spam within labeled spam and honest pages, $s/(s+h)$.
- Spam over Non-spam (SON): number of spam divided by number of honest pages in the top list, s/h.
- Spam Value Ratio (SVR): sum of the similarity values of spam pages divided by the total similarity value of labeled spam and honest pages under the appropriate similarity function, $s^*/(s^*+h^*)$.

– Spam Value over Non-spam Value (SVONV): similarity value sum for spam divided by same for honest, s^*/h^*.

In most of the experiments we use SVR that also performed best in [2]; a small comparison is made in Section 3.2.

We add the new feature defined by either of the above to the existing ones and repeat the classification process with the extended feature set. Since the features are unstable if the neighborhood $N(u)$ is small, we also define versions SR', SON', SVR', SVONV' by regressing towards the undecided 1/2 or 1 value:

$$\text{SR}' = 1/2 + (\text{SR} - 1/2) \cdot (1 - 1/\sqrt{|N(u)|}); \quad \text{SON}' = 1 + (\text{SON} - 1) \cdot (1 - 1/\sqrt{|N(u)|}).$$

2.2 Direction of propagation

We may use both the input directed graph, its transpose by changing the direction of each edge, or the undirected version arising as the union of the previous two graphs. We will refer to the three variants as *directed*, *reversed* and *undirected* versions. For an edge weight function $d : V \times V \to \mathbf{R}$ we use $d^-(u, v) = d(v, u)$ for the reversed and $d^\pm = d + d^-$ for the undirected version. We extend this notion for an arbitrary similarity measure $\text{sim}(u, v)$ computed over edge weights d and compute $\text{sim}^-(u, v)$ over d^- and $\text{sim}^\pm(u, v)$ over $d^\pm$.

Performance of directed, reversed or undirected varies problem by problem: the templatic nature of a Web spam farm is best characterized by similarity of out-links (directed), honest pages have incoming links from honest ones (reversed) and finally similarity in a telephone call graph is best characterized by the undirected graph since communication is typically bidirectional regardless of the actual caller–callee direction.

2.3 Multi-step propagation

There are several variants of weighting neighbors at distance k. We may consider reachability and exact reachability as $d^k(u, v)_{\text{reach}} = 1$ if v is reachable from u by a walk over k edges, 0 otherwise, respectively $d^k_{\text{exact}}(u, v) = 1$ if v is reachable from u in exactly k steps and over no shorter paths, 0 otherwise. We may take the number and the weighted number of such walks: $d^k_{\text{num}}(u, v)$ is the number of walks over k edges that reach from u to v and $d^k_{\text{wnum}}(u, v)$ is the probability of reaching v when starting at u and at each step choosing a random neighbor with probability proportional to the outgoing edge weights. The main multi-step feature we use is $\text{PPR}(u)$, PageRank personalized to $p(v)$, the estimated spamicity of node v as in Section 2.1:

$$\text{PPR}(u) = \sum_k c(1 - c)^k \sum_v p(v) \cdot d^k_{\text{wnum}}(u, v).$$

2.4 Cocitation, Jaccard and cosine

The cocitation $\text{coc}(u, v)$ is defined as the number of common in-neighbors of u and v. This measure turned out most effective for Web spam classification [2]. By the notation of Section 2.2 $\text{coc}^-(u, v)$ denotes bibliographic coupling (nodes pointed to by

F-measure ×1000 iterations	none	graph stacking: d 1	coc 1	coc 2	coc$^-$ 1	coc$^-$ 2	coc$^\pm$ 1	coc$^\pm$ 2	Jac 1	Jac 2	Jac$^-$ 1	Jac$^-$ 2	Jac$^\pm$ 1	Jac$^\pm$ 2	cosine 1	cosine 2	PPR 1	PPR 2
Web Spam I	689	695	707	709	669	677	722	**724**	715	703	689	690	679	680	698	699	715	719
Web Spam II small, text	592	589	601	**605**	598	599	599	601	590	590	592	594	593	595	600	601	599	600
Web Spam II small, link	762	752	788	**793**	774	765	748	738	756	762	782	777	766	756	760	760	731	737
Web Spam II large, link	939	962	983	984	987	988	983	984	984	985	975	976	961	953	982	982	958	960
Churn	086	**102**	063	052	079	088	**102**	083	067	065	059	066	097	089	084	065	092	087
Churn, nonchurn sampled	161	155	141	142	197	200	114	121	254	265	153	147	175	158	267	**280**	277	257

Table 1. 1000 times the F-measure shown for different data sets and edge weights.

both u and v) and $\text{coc}^\pm(u, v)$ is the undirected cocitation. We may also use cocitation downweighted by degree, $\text{coc}(u, v)/d(u) \cdot d(v)$.

The Jaccard and cosine similarity coefficients are useful for finding important connections and ignoring "accidental" unimportant connections. The Jaccard coefficient $\text{Jac}(u, v)$ is the ratio of common neighbors within all neighbors. The coefficient has variants that use the reversed or undirected graphs. For a weighted graph we may divide the total weight to common neighbors by the total weight of edges from u and v. This measure performs poor if for example edges ux and vy have low weight while uy and vx have very high since the Jaccard coefficient is high while the actual similarity is very low.

Cosine similarity fixes the above problem of the Jaccard coefficient. We consider the row of the adjacency matrix corresponding to node u as vector $\overline{u}$. The cosine similarity of nodes u and v is simply $\cos(u, v) = \overline{u}^T \overline{v}$. We may similarly define $\cos^-(u, v)$ over the transpose matrix and $\cos^\pm(u, v)$ over the sum.

Since filling a quadratic size matrix is infeasible, we calculate Jaccard and cosine only for existing edges. The resulting scheme downweights unimportant edges but is unable to add "uncaught contacts" to the network. It is possible to find all pairs with weight above a given threshold by fingerprinting techniques; we leave performance tests for future work.

3 Experiments

3.1 Data sets

For Web spam classification we follow the same methodology as Castillo et al. [5]. We use the Web Spam Challenge Phase I dataset WEBSPAM-UK-2006 [4] that consists of 71% of the hosts classified as normal, 25% as spam and the remainder 4% as undecided as well as the Phase II data set WEBSPAM-LIP6-2006. In this preliminary experiment we consider three tasks. First we use Phase I data (the *Domain Or Two Humans* classification that introduces additional nonspam domains and gives 10% spam among the 5622 labeled sites) with the publicly available features of [5] and then classify by the

	d	coc	coc$^{\pm}$	Jac	Jac$^{\pm}$	cosine	PPR
SON'	602	615	602	600	599	599	599
SON	600	614	602	599	599	599	598
SR'	**603**	618	609	**611**	**606**	**610**	**608**
SR	601	**619**	**611**	610	605	**610**	603
SVONV'	602	607	602	598	596	597	599
SVONV	600	606	602	598	596	598	599
SVR'	**603**	618	609	603	**606**	604	600
SVR	601	**619**	**611**	600	604	601	600

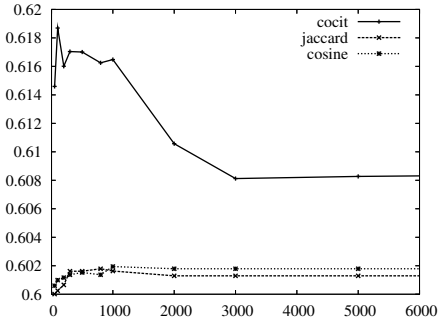

Table 2. Left: 1000 times the F-measure shown for different data weights and feature generation methods. **Right:** the effect of the top list size for SVR.

cost sensitive C4.5 implementation of the machine learning toolkit Weka [24] with bagging. Then we use the Phase II data set features and use the Naive Bayes classifier of Weka. Finally we compute all graph based features of [5] for the Phase II data graph and classify by C4.5 again. We combined the text and graph classifiers by SVM.

For churn classification we use data from a small Hungarian landline telephone service provider. We form features based on aggregated call cost duration in different cost segments, including daytime and off-peak, weekday and weekend as well as local and different long-distance call volumes. Part of the users perform calls via an alternate provider by dialing a prefix; these calls are aggregated similarly for each user. We also use the pricing package information that also includes a distinction of company and family lines as well as the start date of the service usage. For a time range of 12 months, after aggregating calls between the same pairs of callers we obtained a graph with $n = 66,000$ nodes and $m = 1,360,000$ directed edges.

We use the cost sensitive C4.5 implementation of the machine learning toolkit Weka [24] with bagging. Since the running times on the full data set were over 10 hours we also compiled a smaller data set where a random sample of non-churned users were dropped, resulting in $7,151$ users but we kept the entire graph.

3.2 Classification results

Table 1 shows the first three digits of the F-measure for the best selected settings, with the best result in bold. For the Web spam data we measure over the testing labels while for churn we use 10-fold crossvalidation. Since the text and link SVM-combined Web Spam II experiment is computationally very expensive, we only computed the base and the simple neighbor methods that give 0.738 and improve to 0.748 for the small and 0.338 vs. 0.449 for the large graph.

In Table 2 we can see that the difference between the feature generation methods of Section 2.1 are minor and the length of the top list has little effect in the range of k between 100 and 1000, although for cocitation the very long and for others the very short lists deteriorate the performance. Results are shown for the text features of the small Phase II graph and single-iteration stacked graphical classification.

4 Conclusion and Future Work

We presented Web spam and landline telephone churn classification measurements over the Web Spam Challenge Phase II and a small Hungarian landline telephone provider year 2005 datasets. Our experiments demonstrated that stacked graphical learning in combination with graph node similarity methods improve classification accuracy in both cases. Due to the large number of possible feature generation methods the results are by no means complete but show a very good performance of co-citation and little actual use of the neighborhood beyond two steps in the graph.

For future work we plan testing more complex multi-step variants of cocitation and the Jaccard coefficient. Jeh and Widom [14] define SimRank as a multi-step generalization of downweighted cocitation. In an alternate formulation [22] the k-step SimRank $\mathrm{Sim}^{(k)}_{v_1,v_2}$ equals the total weight of pairs of walks with length $k' \leq k$ that both end at u and one of them comes from v_1 while the other one from v_2. The weight of the pair of walks is the *expected* $(1-c)$ *meeting distance* as defined in [14]; notice we get downweighted cocitation if $k = 1$. Computing the full SimRank matrix requires quadratic space; we may use the algorithm of [22] instead. Finally Fogaras and Rácz [8] describe XJaccard as the weighted sum of Jaccard coefficients of the distance k neighborhoods and give an efficient randomized approximation algorithm to compute it.

References

1. W.-H. Au, K. C. C. Chan, and X. Yao. A novel evolutionary data mining algorithm with applications to churn prediction. *IEEE Trans. Evolutionary Computation*, 7(6):532–545, 2003.
2. A. A. Benczúr, K. Csalogány, and T. Sarlós. Link-based similarity search to fight web spam. In *Proceedings of the 2nd International Workshop on Adversarial Information Retrieval on the Web (AIRWeb), held in conjunction with SIGIR2006*, 2006.
3. A. A. Benczúr, K. Csalogány, T. Sarlós, and M. Uher. SpamRank – Fully automatic link spam detection. In *Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web (AIRWeb), held in conjunction with WWW2005*, 2005. To appear in *Information Retrieval*.
4. C. Castillo, D. Donato, L. Becchetti, P. Boldi, S. Leonardi, M. Santini, and S. Vigna. A reference collection for web spam. *SIGIR Forum*, 40(2):11–24, December 2006.
5. C. Castillo, D. Donato, A. Gionis, V. Murdock, and F. Silvestri. Know your neighbors: Web spam detection using the web topology. Technical report, DELIS – Dynamically Evolving, Large-Scale Information Systems, 2006.
6. I. Drost and T. Scheffer. Thwarting the nigritude ultramarine: Learning to identify link spam. In *Proceedings of the 16th European Conference on Machine Learning (ECML)*, volume 3720 of *Lecture Notes in Artificial Intelligence*, pages 233–243, Porto, Portugal, 2005.
7. D. Fetterly, M. Manasse, and M. Najork. Spam, damn spam, and statistics – Using statistical analysis to locate spam web pages. In *Proceedings of the 7th International Workshop on the Web and Databases (WebDB)*, pages 1–6, Paris, France, 2004.
8. D. Fogaras and B. Rácz. Scaling link-based similarity search. In *Proceedings of the 14th World Wide Web Conference (WWW)*, pages 641–650, Chiba, Japan, 2005.
9. R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In *Proceedings of the 13th International World Wide Web Conference (WWW)*, pages 403–412, 2004.

10. Z. Gyöngyi and H. Garcia-Molina. Link spam alliances. In *Proceedings of the 31st International Conference on Very Large Data Bases (VLDB)*, Trondheim, Norway, 2005.
11. Z. Gyöngyi and H. Garcia-Molina. Web spam taxonomy. In *Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web (AIRWeb)*, Chiba, Japan, 2005.
12. Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. Combating web spam with TrustRank. In *Proceedings of the 30th International Conference on Very Large Data Bases (VLDB)*, pages 576–587, Toronto, Canada, 2004.
13. M. R. Henzinger, R. Motwani, and C. Silverstein. Challenges in web search engines. *SIGIR Forum*, 36(2):11–22, 2002.
14. G. Jeh and J. Widom. SimRank: A measure of structural-context similarity. In *Proceedings of the 8th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 538–543, 2002.
15. Z. Kou and W. W. Cohen. Stacked graphical models for efficient inference in markov random fields. In *SDM 07*, 2007.
16. D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *Proceedings of the 12th Conference on Information and Knowledge Management (CIKM)*, pages 556–559, 2003.
17. W. Lu, J. Janssen, E. Milios, and N. Japkowicz. Node similarity in networked information spaces. In *Proceedings of the Conference of the Centre for Advanced Studies on Collaborative research*, page 11, 2001.
18. P. T. Metaxas and J. Destefano. Web spam, propaganda and trust. In *Proceedings of the First International Workshop on Adversarial Information Retrieval on the Web*, 2005.
19. A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly. Detecting spam web pages through content analysis. In *Proceedings of the 15th International World Wide Web Conference (WWW)*, pages 83–92, Edinburgh, Scotland, 2006.
20. PR10.info. BadRank as the opposite of PageRank, 2004. `http://en.pr10.info/pagerank0-badrank/` (visited June 27th, 2005).
21. X. Qi and B. D. Davison. Knowing a web page by the company it keeps. In *Proceedings of the 15th Conference on Information and Knowledge Management (CIKM)*, 2006.
22. T. Sarlós, A. A. Benczúr, K. Csalogány, D. Fogaras, and B. Rácz. To randomize or not to randomize: Space optimal summaries for hyperlink analysis. In *Proceedings of the 15th International World Wide Web Conference (WWW)*, pages 297–306, 2006. Full version available at http://www.ilab.sztaki.hu/websearch/Publications/.
23. C.-P. Wei and I.-T. Chiu. Turning telecommunications call details to churn prediction: a data mining approach. *Expert Syst. Appl.*, 23(2):103–112, 2002.
24. I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, second edition, June 2005.
25. B. Wu, V. Goel, and B. D. Davison. Propagating trust and distrust to demote web spam. In *Workshop on Models of Trust for the Web*, Edinburgh, Scotland, 2006.
26. B. Wu, V. Goel, and B. D. Davison. Topical TrustRank: Using topicality to combat web spam. In *Proceedings of the 15th International World Wide Web Conference (WWW)*, Edinburgh, Scotland, 2006.
27. X. Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005.