Wikipedia Requests for Adminship

Ashish Bhat
Department of Computer Science
Virginia Tech
Blacksburg, Virginia, USA
ashishbhat@vt.edu

Humaid Desai
Department of Computer Science
Virginia Tech
Blacksburg, Virginia, USA
humaiddesai@vt.edu

Priyanka Savla
Department of Computer Science
Virginia Tech
Blacksburg, Virginia, USA
priyankas@vt.edu

ABSTRACT

There are many applications for developing a model that makes predictions about the relationships between people. Making use of the available features on social platforms, this work can help analyze social networks. It has been shown by former research that sentiment analysis and signed-network analysis can be used independently to predict the relationship between people. However, combining them both can produce a robust relationship prediction model. We have implemented the processes described in [2] in order to achieve the said results.

INTRODUCTION

Person-to-person assessments are ubiquitous in all types of conversations and are essential for creating reputations, strengthening social bonds, and influencing public opinion. Individual evaluations can be assessed using signed social networks and textual sentiment analysis, but this ignores the deep relationships between language and social context. Our goal is to develop a model that combines information from the signed social network in which P and Q are embedded, with sentiment analysis of the evaluative messages related to P and Q, in order to forecast individual P's views of individual Q. We intend to create a model that combines textual and social network data to predict the polarity of person-to-person evaluations. Combining the network and sentiment model would make it better to predict new edge labels in the network.

RELATED WORK

1. SENTIMENT ANALYSIS

Sentiment analysis is a term used in natural language processing to describe how information about emotions, attitudes, views, and social identities is processed and communicated through language[1]. Most of the work adopts a dimensional model in which valence/polarity and arousal/intensity are the primary determinants of emotion [8][9][10]. Predicting the valence of a substance is the most common application. Substance here includes product, company, and service reviews. For our basic sentiment model, we use the conceptual assumptions from this paper, but our focus is on person-to-person evaluations and their social ramifications. This includes work on modeling political affiliation [11][12][13], bias [14][15], and stance on debate topics [7][16][17][18], but these aspects of trust and social identity are not our primary focus. Rather, we expect them to predict the sentiment classifications we want to make—for example, if two people have similar political ideas, they will prefer to rate each other favorably. Recent sentiment analysis research has resulted in the use of current, contextual, and social data to make more complex language predictions [19][20][21]. Our model, which is based on these findings, aims to change the way sentiment predictions are made based on data from the network (and vice versa).

2. SIGNED NETWORK ANALYSIS

Many social networks use signed edges between users to encode person-to-person sentiment information. These edges summarize users' perceptions about one another. To discover and analyze patterns in these relationships, sociological theories of pairwise relationships and group-level structure might be used in this situation [22][23]. Simple intuitions like 'a friend of my friend is my friend,' 'an enemy of my enemy is my friend,' and 'an enemy of my friend is my enemy' underpin balance theory. These are claims concerning the edge signs of connected node triangles in graph theory: given the signs of two edges, balance theory predicts the third, as shown in Fig. 2, where the two given edges (gray) determine the third (black). [24] propose a *status theory* for directed interactions, which states that networks arrange according to social status: a node has positive edges to others of higher status and negative edges to those of lower status. The structure of various directed signed triangles is shown in Fig. 2, where the sign of the third edge (black) can be inferred from the signs and directions of the other two edges (gray). [24]

show that signed edges in networks emerge in a way that is broadly consistent with both of these theories, and that social network structure can allow accurate edgesign predictions on its own [25]. In a scenario where all observed edges are positive, [26] anticipate hidden positive and negative edges.[27][28] use a hinge-loss Markov random field to forecast the sign, which is a form of probabilistic graphical model first developed by [34]. To make even more reliable predictions, we combined these principles with a sentiment model.

3. UNION OF SENTIMENT AND SIGNED NETWORK ANALYSIS

Sentiment models and signed networks have proven to be effective in a variety of situations. However, a thorough grasp of how emotion expression and social networks interact is missing from the existing scientific picture. Contextual and demographic data can be added to a text-based sentiment model to capture some of these connections, but those features can only approximate the rich relational structure represented in a signed network.

Using an expansion of graph-cuts technique used in [24], [16] and [31] leverage on this understanding. [16] use party affiliation and mentions in speeches to predict voting tendencies, and [31] utilize Twitter following and mentions to predict opinions about political and social events. [33] and [32] follow similar concepts by including terms in their models that enforce homophily among friends in terms of preferences.

We assume some of the previous writers' assumptions, but our aim is fundamentally different in two ways. First, we model person-to-person evaluations, whereas they model person-to-item evaluations; this is also the focus of [34], who, however, employ an unsigned network, whereas our work is aimed around identifying positive and negative edge labels. Second, although the previous models assume broad homophily, we allow our model to explore the entire collection of triangle configurations provided by Fig. 2.

APPROACH

The dataset used in the work were extracted from:

Wikipedia Requests for Adminship (with text): https://snap.stanford.edu/data/wiki-RfA.html

The dataset consists of network information, text information and recommendation information. It consists of 10,835 nodes, 159,388 edges, and 956,428 triangles. Networkx was used to create the directed graph and edges with missing information (source, target, vote, comment) were removed. Two nodes were assigned at most one edge to make studying the graph simpler. Neutral votes were also removed to keep only upvotes (1) and downvotes (0). After that, we made ten sets to average the mistake. We randomly selected 10 nodes in the graph and used breadth first search to extract 10 subgraphs of 200 nodes (the original number in the paper was 350, but we reduced it because the computational time would be too long otherwise). The overlapping edges between the subgraphs were then deleted, leaving them with no shared nodes or edges. Because the graph includes so many edges, the number of overlapping edges were likely to be low. We started with the first subgraph and computed the overlapping edges with all nine other subgraphs pairwise, then removed those edges from the nine subgraphs. We then moved on to the second subgraph and repeated the process with the remaining eight graphs. This was repeated until the ninth graph was reached. The subgraphs that resulted showed different groups of people voting for others. As seen in Figure 1, the subgraph edges were made up of vote data and a comment about the vote.

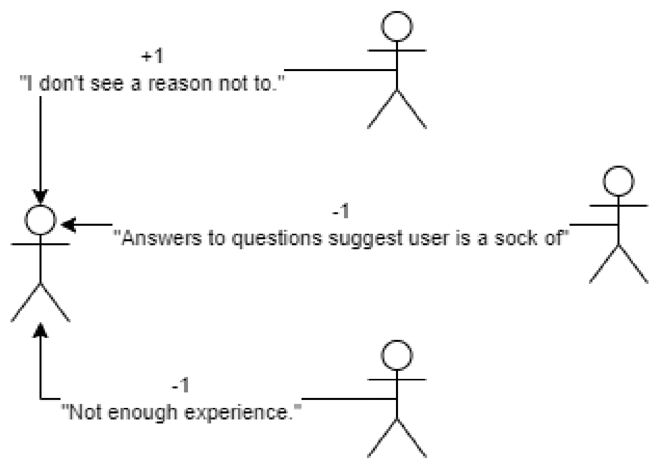


Figure 1: The RfA dataset consists of the results after voting for new Wikipedia admins alongwith the comments by users for candidates.

Linguistic content and graph-like structure was the reason why the dataset was chosen. The chosen dataset was adaptable and suited well to different types of graphs, with the goal of establishing that strong edge inference could be obtained using both sentiment analysis and network social theories. The sentiment distribution dataset from Wikipedia was an excellent predictor of edge sign (average AUROC of 0.85 across the dataset), however it had a poor network structure (103.4 triangles per node on average).

We compared four prediction models: a random model, a sentiment model, a network model, and a combined model in this study. Both the sentiment and network models were used in the combined model. For the edges in the testing dataset in the random model, we chose an edge sign at random. This random model acted as a benchmark against which the other models were measured. For the sentiment model, our sentiment classifier predicted the edge signs. In the instance of the Rfa Dataset, the comment was the model's input, while the sentiment probability was the model's output. We minimized a cost function based on the edge signs of the triangles in the testing graph for the network model. The Rfa dataset was one example of this. For triangles that had a cost function, it was minimized if they respected the social status theory, while other configurations were maximized.

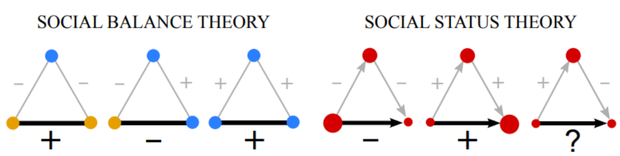


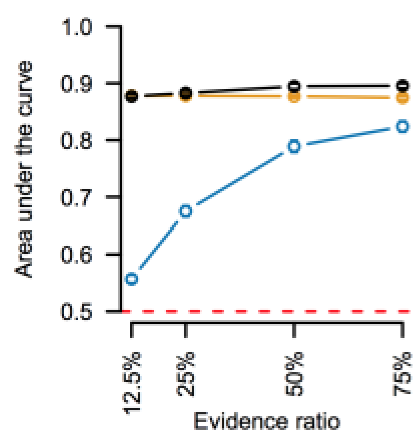
Figure 2: Theories of social balance and status

We minimized a cost function based on the sentiment probability distribution over the edges and triangles configurations (following the social status/social balance theory) for the combined model. For "best" fit, the weights for the individual components in the cost function could be modified based on the dataset. The goal then was to show that the combined model outperformed the random model, network model, and sentiment model.

EXPERIMENT

For the sentiment model, a custom constructed sentiment classifier/preprocessed sentiment probabilities were used, for the network model, probabilistic soft logic (PSL) was used, and for the combined model, both were used. We used logistic regression and the L2 penalty to build a sentiment classifier using the RfA dataset's text comments. The 10,000 most common terms were chosen as model features, and 1,000 comments and the vote were chosen at random from the graph for training. To feed the classifier, we preprocessed the comments by eliminating stop words, stemming the words, and finding duplicate words spelled differently. The weights used for the PSL rules define the cost function for the network and combined model. The PSL rules are expressed as a logical statement with a weight for each rule, with the weight indicating the rule's relevance. The edges of the testing graph can be subjected to a prior. For the combined model implementation the sentiment probability was used as our prior. For further details on the rules, see the appendix. We used four criterias to compare the efficiency of the various models. The receiver operating characteristic's areas under the curve (AUC/ROC), the negative precision-recall curve's areas under the curve (AUC/negPR), and the positive and negative precision-recall curve. The purpose of the metric was to demonstrate the combined model's robustness in comparison to other models. To demonstrate the combined model, we reduced the contribution of the sentiment or network models and tested if our combined model could dynamically get the best outcomes. To examine the network analysis contribution in the first scenario, we lowered the evidence ratio of the testing graphs. To put it another way, we lowered the number of observations from the edges.

The network model's performance would deteriorate as projected, but the sentiment model's performance would remain stable. On the other hand, we eliminated the top features of the sentiment classifier to evaluate how robust the combined model was when the sentiment model started to perform poorly in order to test the sentiment analysis contribution. The features on the full graph were ranked by their mutual information with the edge signs (the higher the MI, the better the feature predicted the edge sign).The preprocessing and analysis took roughly two weeks, while the coding took more than 50 hours. Several engineering issues arose, which took a long time to resolve. We had to decrease the testing sets of the RfA corpus to 200 nodes (rather than 350 nodes) because the PSL package required a significant amount of computing resources. It was done on a personal laptop.



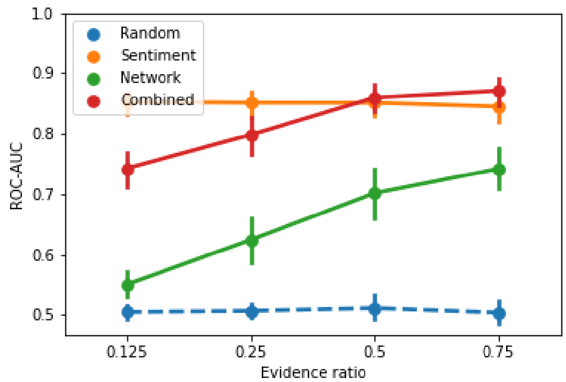
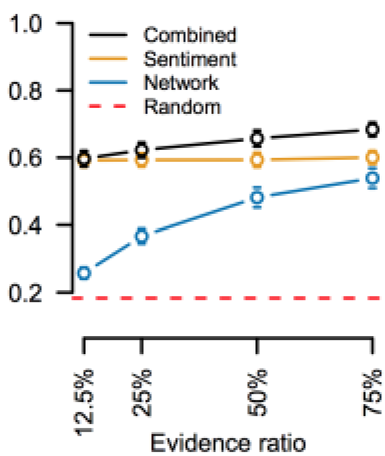


Figure 3: Analysis of Wikipedia data shown as AUC for ROC versus Evidence Ratio with a 95% confidence interval (original results on top and replicated results on bottom).



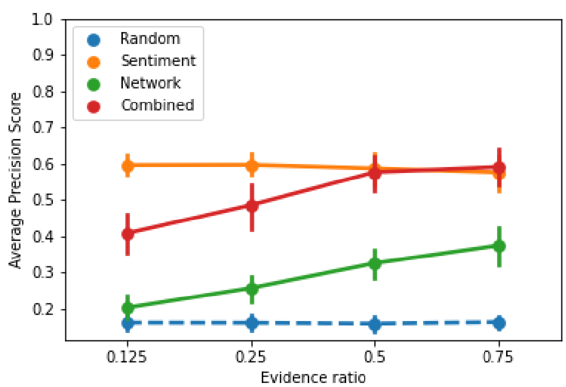
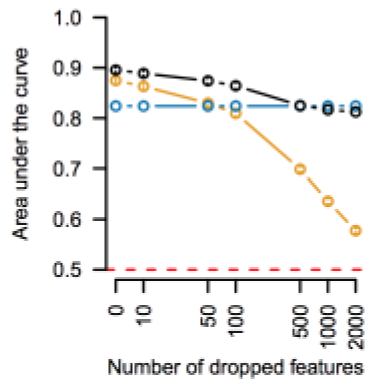


Figure 4: Analysis of Wikipedia data shown as AUC for negative PR versus Evidence Ratio with a 95% confidence interval (original results on top and replicated results on bottom).



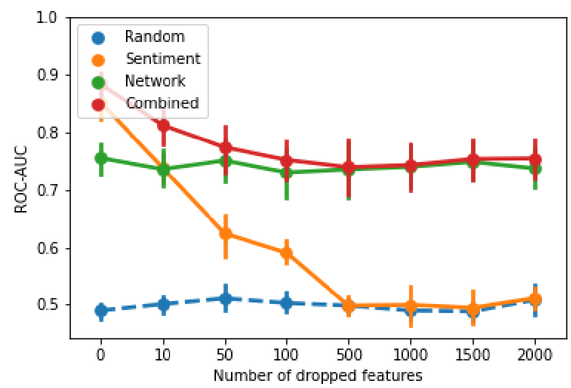


Figure 5: Analysis of Wikipedia data shown as ROC versus number of dropped features with a 95% confidence interval (original results on top and replicated results on bottom).

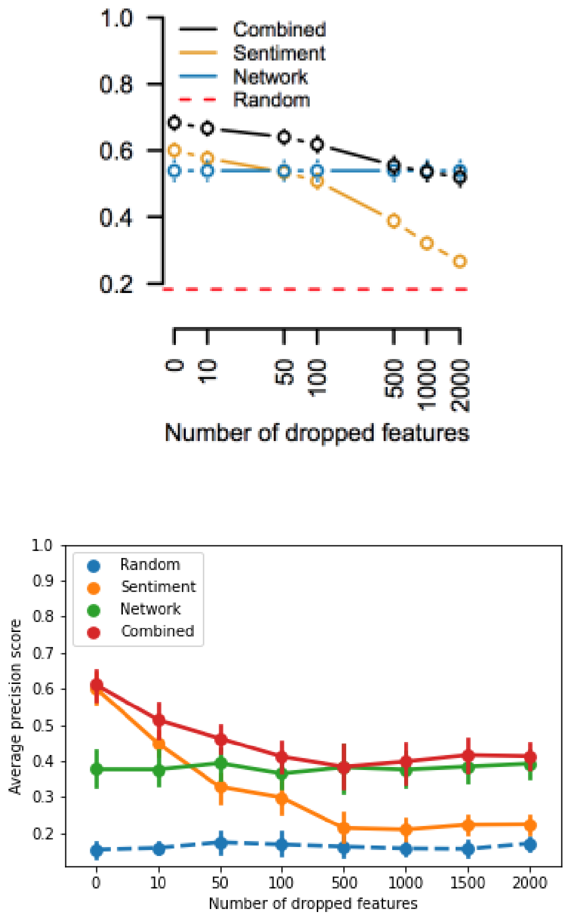


Figure 6: Analysis of Wikipedia data shown as negative PR versus number of dropped features with a 95% confidence interval (original results on top and replicated results on bottom).

CONCLUSION

Our findings were quite comparable to those given in the original research paper. The replication results are compared to the original findings with graphs of the model's performance using the same metrics, scale, and trace colors.

The graphs on the following page show that the combined model is more sensitive to the contribution of sentiment on the edges than to the contribution of network structure. Indeed, the combined model's initial performance is slightly better than the network model in the first two graphs (figures 3 and 4). However, in the next two graphs, figures 5 and 6, the decline in performance is more pronounced, indicating a stronger impact of the sentiment model's poor performance. We should note that the combined model is robust, performing well even when the sentiment model fails. When the sentiment model's contribution is greatly reduced, the combined model performs similarly to the network model. It is worth noting that our results appear to be superior to the results published in the original paper. We were mystified that our results improved despite the fact that we employed a simplified approach to model implementation (the weights were not found using an unsupervised learning strategy, and the sentiment probabilities were not subjected to a nonuniform weight distribution). The disparity in our results can be attributed to the fact that the original authors' weights were not optimal for the situation. While looking for weight values in their technique, the optimizer could have fallen into a local optimum. The model is sensitive to the weights in the cost function and rule weights, which will be investigated further in the extension.

LIMITATIONS AND DISCUSSION

The advantage of using both the network structure and textual data gives the combined model its strength. Modern social media data contains a lot of network and textual data. The network structure in social media can be defined by the "friends" network structure and the comments that build debates on the edges between friends. In some circumstances, a social media dataset with a weak network structure may have higher sentiments, and vice versa. The combined models are flexible to a wide range of datasets since they employ both sentiment and network structure to support the prediction.

The weights for both the PSL model rules and the cost function in this study were chosen based on a heuristic. It's interesting to note that this approach improved the performance of our combined model implementation in various respects. The performance difference could be due to a suboptimal weight selection observed in a local optimum.

ACKNOWLEDGMENTS

The authors of this paper would like to express their gratitude to Dr. Patrick Butler for his assistance.

REFERENCES

[1] Pang, Bo and Lee, Lillian. 2008. Opinion mining and sentiment analysis., Foundations and Trends in Information Retrieval.
[2] Robert West and Hristo S. Paskov and Jure Leskovec and Christopher Potts. 2014. Exploiting Social Network Structure for Person-to-Person Sentiment Analysis, CoRR.
[3] Bach, Stephen H. and Broecheler, Matthias and Huang, Bert and Getoor, Lise. 2017. Hinge-Loss Markov Random Fields and Probabilistic Soft Logic, Journal of Machine Learning Research (JMLR).
[4] Cartwright, Dorwin and Harary, Frank. 2008. Structure balance: A generalization of Heider's theory., Psychological Review.
[6] Platt, John. 1999. Probabilistic outputs for support vector machines and comparisons to regularize likelihood methods., Advances in Large Margin Classifiers.

[7] Wei-Hao Lin, Theresa Wilson, Janyce Wiebe, and Alexander Hauptmann. 2006. Which side are you on? Identifying perspectives at the document and sentence levels. In Proceedings of the 10th Conference on Computational Natural Language Learning.
[8] James A. Russell. 1980. A circumplex model of affect. Journal of Personality and Social Psychology, 39(6):1161–1178.
[9] Lisa Feldman Barrett and James A. Russell. 1998. Independence and bipolarity in the structure of affect. Journal of Personality and Social Psychology, 74(4):967–984.
[10] David C. Rubin and Jennifer M. Talerico. 2009. A comparison of dimensional models of emotion. Memory, 17(8):802–808.
[11] Rakesh Agrawal, Sridhar Rajagopalan, Ramakrishnan Srikant, and Yirong Xu. 2003. Mining newsgroups using networks arising from social behavior. In Proceedings of the 12th International Conference on World Wide Web.
[12] Robert Malouf and Tony Mullen. 2008. Taking sides: User classification for informal online political discourse. Internet Research, 18(2):177–190
[13] Bei Yu, Stefan Kaufmann, and Daniel Diermeier. 2008. Classifying party affiliation from political speech. Journal of Information Technology and Politics, 5(1):33–48.
[14] Tae Yano, Philip Resnik, and Noah A. Smith. 2010. Shedding (a thousand points of) light on biased language. In Proceedings of the NAACL-HLT Workshop on Creating Speech and Language Data With Amazon's Mechanical Turk.
[15] Marta Recasens, Cristian Danescu-Niculescu-Mizil, and Dan Jurafsky. 2013. Linguistic models for analyzing and detecting biased language. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics.
[16] Matt Thomas, Bo Pang, and Lillian Lee. 2006. Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing.
[17] Swapna Somasundaran and Janyce Wiebe. 2010. Recognizing stances in ideological on-line debates. In Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text
[18] Pranav Anand, Marilyn Walker, Rob Abbott, Jean E. Fox Tree, Robeson Bowmani, and Michael Minor. 2011. Cats rule and dogs drool! Classifying stance in online debate. In Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis.
[19] John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics.
[20] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing.
[21] Dan Jurafsky, Victor Chahuneau, Bryan R. Routledge, and Noah A. Smith. 2014. Narrative framing of consumer sentiment in online restaurant reviews. First Monday, 19(4–7).
[22] Fritz Heider. 1946. Attitudes and cognitive organization. The Journal of Psychology, 21(1):107–112.
[23] Dorwin Cartwright and Frank Harary. 1956. Structure balance: A generalization of Heider's theory. Psychological Review, 63(5):277–293.
[24] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010b. Signed networks in social media. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.
[25] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010a. Predicting positive and negative links in online social networks. In Proceedings of the 19th International Conference on World Wide Web.
[26] Jerome Kunegis, Julia Preusse, and Felix Schwagereit. ˆ 2013. What is the added value of negative links in online social networks? In Proceedings of the 22nd International Conference on World Wide Web.
[27] Stephen Bach, Bert Huang, Ben London, and Lise Getoor. 2013. Hinge-loss Markov random fields: Convex inference for structured prediction. In Proceedings of the 29th Conference on Uncertainty in Artificial Intelligence.
[28] Bert Huang, Angelika Kimmig, Lise Getoor, and Jennifer Golbeck. 2013. A flexible framework for probabilistic models of social trust. In Proceedings of the 2013 International Social Computing, Behavioral-Cultural Modeling and Prediction Conference.
[29] Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics.
[30] Chenhao Tan, Lillian Lee, Jie Tang, Long Jiang, Ming Zhou, and Ping Li. 2011. User-level sentiment analysis incorporating social networks. In Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.
[31] Xia Hu, Lei Tang, Jiliang Tang, and Huan Liu. 2013. Exploiting social relations for sentiment analysis in microblogging. In Proceedings of the 6th ACM International Conference on Web Search and Data Mining
[32] Hao Ma, Dengyong Zhou, Chao Liu, Michael R Lyu, and Irwin King. 2011. Recommender systems with social regularization. In Proceedings of the 4th ACM International Conference on Web Search and Data Mining.
[33] Jiliang Tang, Huiji Gao, Xia Hu, and Huan Liu. 2013. Exploiting homophily effect for trust prediction. In Proceedings of the 6th ACM International Conference on Web Search and Data Mining.
[34] Matthias Broecheler, Lilyana Mihalkova, and Lise Getoor. 2010. Probabilistic similarity logic. In Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence.

APPENDIX

WIKIPEDIA NETWORK MODEL, PSL RULES

```
1.0: (knows(A,B) & knows(B,C) & knows(A,C) & trusts(A,B) & trusts(B,C) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2
1.0: (knows(A,B) & knows(B,C) & knows(A,C) & trusts(A,B) & ~trusts(B,C) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(A,B) & knows(B,C) & knows(A,C) & ~trusts(A,B) & trusts(B,C) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(A,B) & knows(B,C) & knows(A,C) & ~trusts(A,B) & ~trusts(B,C) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2

1.0: (knows(A,B) & knows(C,B) & knows(A,C) & trusts(A,B) & trusts(C,B) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2
1.0: (knows(A,B) & knows(C,B) & knows(A,C) & trusts(A,B) & ~trusts(C,B) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(A,B) & knows(C,B) & knows(A,C) & ~trusts(A,B) & trusts(C,B) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(A,B) & knows(C,B) & knows(A,C) & ~trusts(A,B) & ~trusts(C,B) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2

1.0: (knows(B,A) & knows(B,C) & knows(A,C) & trusts(B,A) & trusts(B,C) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2
1.0: (knows(B,A) & knows(B,C) & knows(A,C) & trusts(B,A) & ~trusts(B,C) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(B,A) & knows(B,C) & knows(A,C) & ~trusts(B,A) & trusts(B,C) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(B,A) & knows(B,C) & knows(A,C) & ~trusts(B,A) & ~trusts(B,C) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2
```

Wikipedia Requests for Adminship

```
1.0: (knows(B,A) & knows(C,B) & knows(A,C) & trusts(B,A) & trusts(C,B) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2
1.0: (knows(B,A) & knows(C,B) & knows(A,C) & trusts(B,A) & ~trusts(C,B) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(B,A) & knows(C,B) & knows(A,C) & ~trusts(B,A) & trusts(C,B) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(B,A) & knows(C,B) & knows(A,C) & ~trusts(B,A) & ~trusts(C,B) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2
```

WIKIPEDIA NETWORK-SENTIMENT (COMBINED) MODEL, PSL RULES

```
16.0: (knows(A,B) & prior(A,B) & (A-B)) >> trusts(A,B) ^2
16.0: (knows(A,B) & trusts(A,B) & (A-B)) >> prior(A,B) ^2

1.0: (knows(A,B) & knows(B,C) & knows(A,C) & trusts(A,B) & trusts(B,C) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2
1.0: (knows(A,B) & knows(B,C) & knows(A,C) & trusts(A,B) & ~trusts(B,C) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(A,B) & knows(B,C) & knows(A,C) & ~trusts(A,B) & trusts(B,C) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(A,B) & knows(B,C) & knows(A,C) & ~trusts(A,B) & ~trusts(B,C) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2

1.0: (knows(A,B) & knows(C,B) & knows(A,C) & trusts(A,B) & trusts(C,B) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2
1.0: (knows(A,B) & knows(C,B) & knows(A,C) & trusts(A,B) & ~trusts(C,B) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(A,B) & knows(C,B) & knows(A,C) & ~trusts(A,B) & trusts(C,B) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(A,B) & knows(C,B) & knows(A,C) & ~trusts(A,B) & ~trusts(C,B) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2

1.0: (knows(B,A) & knows(B,C) & knows(A,C) & trusts(B,A) & trusts(B,C) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2
1.0: (knows(B,A) & knows(B,C) & knows(A,C) & trusts(B,A) & ~trusts(B,C) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(B,A) & knows(B,C) & knows(A,C) & ~trusts(B,A) & trusts(B,C) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(B,A) & knows(B,C) & knows(A,C) & ~trusts(B,A) & ~trusts(B,C) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2
```


```
1.0: (knows(B,A) & knows(C,B) & knows(A,C) & trusts(B,A) & trusts(C,B) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2
1.0: (knows(B,A) & knows(C,B) & knows(A,C) & trusts(B,A) & ~trusts(C,B) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(B,A) & knows(C,B) & knows(A,C) & ~trusts(B,A) & trusts(C,B) & (A - B) & (B - C) & (A - C)) >> ~trusts(A,C) ^2
1.0: (knows(B,A) & knows(C,B) & knows(A,C) & ~trusts(B,A) & ~trusts(C,B) & (A - B) & (B - C) & (A - C)) >> trusts(A,C) ^2
```