École polytechnique fédérale de Lausanne

Project in
A Network Tour of Data Science

Detection of Spammers in a Social Network

by
Ali El Abridi
Yann Bouquet
Tariq Kalim
Jonas Müller

January 10, 2020
v1.0

Contents

1 Story

On the Internet, messages can be sent by ill-intentioned people known as "spammers". These messages or "spams" have various functions: phishing, virus attacks, marketing, etc... Today social networks offer an ideal breeding ground for spammers and their proliferation tends to tarnish the image and quality of services offered by such systems. Moreover, spammers who escape the automatic control of the networks have managed to manipulate their accounts to blend in with the masses and thus only be detected by manual administration. We are therefore interested in detecting these spammers by classifying the users of a service called Tagged.com corresponding to what is called a multi-relational social network. This classification will be done according to the user data provided by Tagged.com but also the meaningful social interaction between the users by graphing these relationships and interactions between users which will provide new information on the behavior of each user within the social network according to the type of relationship.

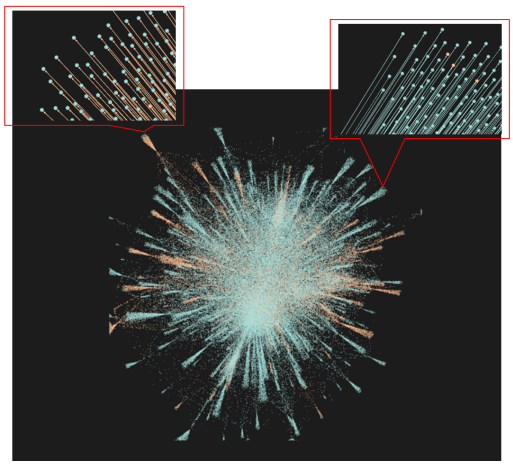
2 Acquisition

The given dataset contains the different interactions between users of a social network. We will refer to the users as the nodes of our graph (network). There are in total 7 different interactions or relations (edges) possible between users. We decided to consider an unweighted directed graph. Two nodes can have at most 14 edges between them. Two edges per relation type and one edge per direction. We decided to subdivide the network by relationship. Thus we get 7 networks, one graph per network. In a graph, two n_i and n_j nodes are connected by the edge e_{ij} if a relational link exists from the source n_i to the destination n_j.

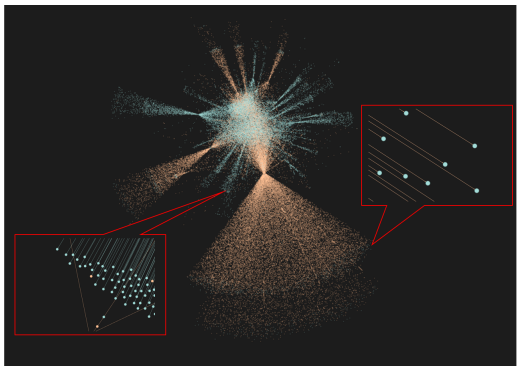
3 Exploration

3.1 Visualization

These acquired networks being too large to be visualized, we had to sample the networks. We therefore used a traversal method by breadth first search [1] for sampling in order to lose as little as possible the properties of the original graph while keeping a correspondence between the graph node and a user and the original properties of the different graphs. By retrieving 20 000 users with this method for the relations 1 and 3 taken as examples, we obtain the visualizations in 3.1.



(a) Graph visualization for sampled relation 1



(b) Graph visualization for sampled relation 3

Figure 3.1: Graph visualization for some relations (1 and 3)
The orange edges and nodes are related to spammers users, the blues ones are related to the non-spammers users.

From this figure, we can observe the hubs. Even though they may or may not come from spammers, we

can still see that spammers connect to many users who appear to be non-spammers while non-spammers hubs connect to both spammers and non-spammers.
This sampling method was necessary to create graphs useful for extracting graph-based features for relations 3, 4, 5, 6. We analyzed the properties of the graphs for relations 1,2,7 and compared those of the graphs before and after sampling for relations 3,4,5.

3.2 Degree Distributions

[Figure 7.1] The degree distributions of relations 1,2 and 7 follow power law distributions meaning that those graphs are scale-free networks. The degree distributions of the relations 3,4,5,6 considered before or after the sampling process [Figure 7.2] are heavy tail, meaning that most of the nodes have low degrees and hubs exists, such as the other relations. They seem to be in sublinear regime following a stretched exponential distribution:

$$p_k \;=\; k^{-\alpha} \exp\left(-\frac{2\mu(\alpha)}{\langle k\rangle(1-\alpha)} k^{(1-\alpha)}\right)$$

limiting the size and the number of the hubs. However, the sampling modify the α and μ parameters with μ depending weakly on α. Furthermore [Figure 7.3] low degree nodes are less numerous in spammers than in non-spammers, whereas high degree nodes are more numerous in spammers than in non-spammers. Indeed, this shows that spammers are more likely to connect to a large number of users compared to non-spammers. By observing the in and out degree distributions [Figure 7.4] separately we can observe that differences exist. According to the out degree distributions, spammers tend to connect to more user than non spammers. However the in_degree distributions are relatively similar between spammers and non spammers. As a consequence, the behavior of a spammer can be close to a non spammer. if the behavior of spammer node is similar to a non spammer node, we can look at the neighbors of a node to see if the neighbors of a spammer node and the ones of a non spammer node have different behaviors.

3.3 Average Clustering Coefficient

Globally, the average clustering coefficient is higher for non spammers than for spammers [Figure 7.5]. It means that the neighbors of a spammer are less connected to each other than the ones of a non spammer user. It can be explained by the fact that a spammer would tend to connect to users who will not necessarily inform other users of his presence in the network. As a consequence the spammer will target users that connected to each others.
Through this exploration we can see that graphs-based features can help us to differentiate a spammer from a non spammer by comparing their behavior as nodes in the networks graphs.

4 Features Extraction

4.1 Features from the full network

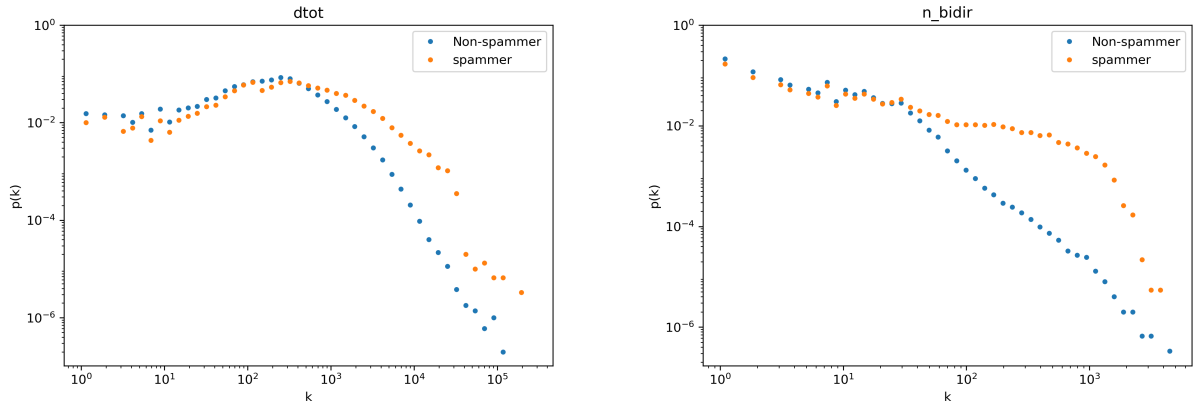
The following features are extracted from the preprocessed full dataset which is essentially a sorted to allow local computations. Note that the preprocessing takes about 5h while the feature extraction is in the range of 1h (on a regular notebook). In what follows the number sign # denotes that the feature is calculated for each relation individually.

- dtot_#: The total degree.
- dout_#: The outgoing degree.
- duni_#: The number of unique neighbors.
- dnbi_#: The number of bidirectional edges.
- n_bidir: The number of bidirectional edges over all relations.

Note that multiple relations between two nodes are possible and therefore the number of unique neighbors is in general not equal to the degrees. The difference between dtot and duni represents the number of edges that are repeated. In other words dtot counts the number of "communications" from a node while duni counts the number of "recipients". The number of bidirectional edges dnbi is a measure of how many interactions are actually answered at least once. This too, is in general different from the total degree and the unique neighbors. For example, in a social network a "like" from one node might not elicit a "like" from the recipient. Therefore dnbi represents a measure of how many interactions cause a reaction.
Furthermore, the n_bidir is generally different from the sum of the dnbi since a given interaction might cause an interaction of a different kind (relation). For example, the "like" of one node might induce a "message" from the recipient as a response. Note also that n_bidir can in fact even be smaller than the sum of dnbi in the case when a certain edge is actually bidirectional in multiple relations which is counted as one bidirectional relation in n_bidir. Furthermore it might be worthwhile to point out that an edge is counted as bidirectional if it has at least one interaction in both directions. However, there might in fact be much more interactions on a certain edge (multiple messages send between two users) which is not measured by this feature. Formally speaking the n_bidir is simply the cardinality of the intersection between the two sets of sources and destinations attached to a node.

4.1.1 Feature Distribution

Examining the obtained features we find that the tail of the total degree distribution in the loglog plot is well approximated linearly and therefore following a power law (cf. fig 4.1a). This finding exhibits the scale-free nature of the graph under consideration and highlights the existence of hubs (high-degree nodes or popular users).



(a) Total degree distribution of the nodes

(b) Distribution of bidirectional relations

Figure 4.1: Feature distributions showing the probability of having a k-node for dout and n_bidir on a loglog scale

Furthermore it is interesting to note that in fig 4.1b the distribution of the number of bidirectional edges seems to diverge in the tail for the non-spammer and the spammer group. This fact could be used to create more features related to the fact that edges are in general repeated. For example, counting the number of edges per node that are at least n-times repeated for n=3,4,5... This might thus be a way to further improve the subsequent classification.

4.2 Features by relations with sampling

Based on one network per relation and sampling process for relations 3 to 6, we managed to retrieve graph-based features inspired from the original paper [2]. Degree, in_degree and out_degree for every node. PageRank algorithm is ranking nodes by their importance according to the number of edges connected to them and the quality of those edges. Graph Coloring algorithm assign a color to each

node so that neighbors doesn't share the same color. Triangle Count or the number of triangles the given node participates in.

5 Exploitation

5.1 Pre-processing

Our dataset is very large, which would render computations challenging if not unfeasible with the limited ressources at hand. Adding to that the classes are highly imbalanced, which hurts classification significantly in terms of detecting spammers signals. So we decided to sample our dataset into a much smaller one (100k data points) that we split in a 80/20 manner for training and testing, which should be enough for training. We also made sure that the sampled dataset would be balanced in regards to labels.

5.2 Methods and metrics

Given the binary classification task ahead and the (large) number of features our dataset contains (60 in total), we focused mainly on decision tree-based algorithms. We tested the following methods for this task: **Random Forest, XGBoost, Gradient Boosting, PCA + KNN, PCA + Random Forest, Ridge Regression.**
For each of these algorithms we ran a grid-search coupled with 4-folds cross-validation on our sampled training set to obtain their best parameters and then tested them on our sampled testing set. As for the full dataset, the spammer's class constitutes a minority, we decided not to focus on the classifier's accuracy, but on other metrics that would translate well to class-imbalanced data-sets, such as the areas under the ROC and precison-recall curve as well as the F1-score. The F1-score was used as the scoring method for our cross-validation.

5.3 Results

We first started by only using the graph-based features to get a "baseline" to validate the usefulness of the user data features. This gave us the results in 5.2a. Then we added the user data features and obtained the results in 5.2b.

Table 5.1: Results of the models on the sampled test set

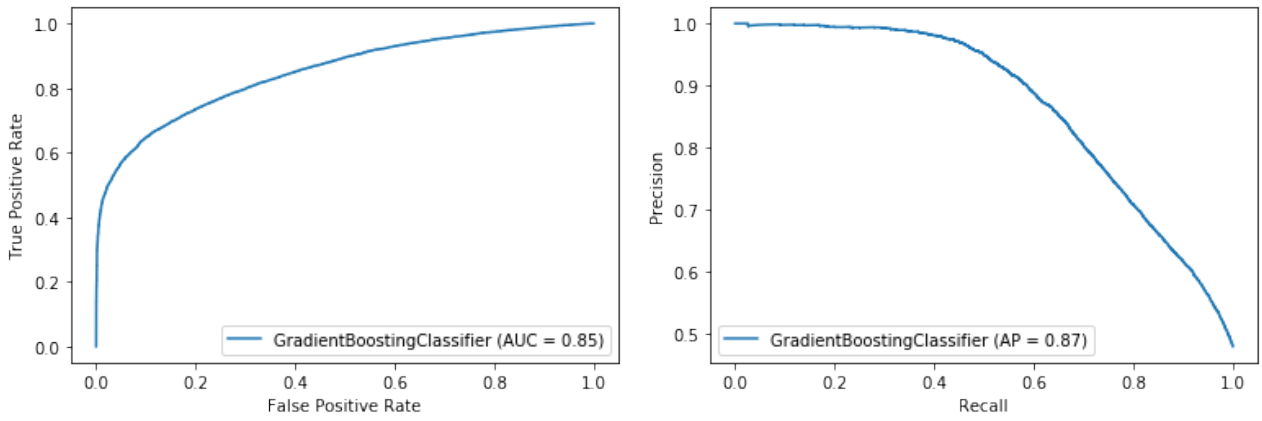
(a) graph-based features

Model	AUROC	AUPR	F1-score
PCA + KNN	0.78	0.80	0.71
PCA + Random Forest	0.80	0.82	0.73
Random Forest	0.81	0.83	0.73
XGBoost	0.81	0.83	0.74
Gradient Boosting	0.81	0.83	0.74
Ridge Regression	0.70	0.72	0.62

(b) graph-based features + user data features

Model	AUROC	AUPR	F1-score
PCA + KNN	0.82	0.84	0.76
PCA + Random Forest	0.83	0.85	0.78
Random Forest	0.85	0.87	0.77
XGBoost	0.85	0.87	0.77
Gradient Boosting	**0.85**	**0.87**	**0.78**
Ridge Regression	0.80	0.82	0.73

We can clearly see that the demographic features do indeed improve the classification for every measured metric. We can see when using all our available features that the best classifiers (Random Forest, XGBoost and Gradient) are producing the same results, which suggests that "better" models won't help us that much. We're suspecting that the path to better results (especially a greater f1) would come through the features side of the problem. Our current features are not "sufficient", for example they don't capture time and sequences (we can expect spammers to engage with a frequency distinctively different from non-spammers). Sequential di-gram features and Markov Models could be great features to explore in order to close the remaining gap.



(a) ROC curve

(b) Precision-Recall curve

Figure 5.1: ROC and Precision-recall curves for Gradient Boosted Decision Trees on the sampled test set using graph-based features and user data

Table 5.3: Classification report for Gradient Boosted Decision Trees on sampled test set

	precision	recall	f1-score	support
0	0.74	0.89	0.81	10422
1	0.84	0.66	0.74	9578
accuracy			0.78	20000
macro avg	0.79	0.77	0.77	20000
weighted avg	0.79	0.78	0.78	20000

6 Conclusion

As a conclusion, we can see that the graph based features have been the most important signal in the spammer/non-spammer classification. These signals represent the different meaningful social interaction between the users that unveil the purpose of the social network user and whether the users' behavior is abusive or not. Moreover, some basic user features such as age, time, and gender have also been extremely useful in the classification process as abusive social network users tend to use a certain gender and age range when creating their fake profiles. However, the recall metrics of spammers in our model is low given that the spammer class is much less prevalent than genuine non spammer class users. If the model were to used in production, we would have increase the threshold in order to increase the precision and lower the recall in order to prevent false positives and to prevent genuine users from being labelled as spammers.

References

[1] F. Zhang, S. Zhang, P. C. Wong, H. R. Medal, L. Bian, J. E. Swan, and T. J. Jankun-Kelly, "A visual evaluation study of graph sampling techniques," in Visualization and Data Analysis.

[2] S. Fakhraei, J. Foulds, M. Shashanka, and L. Getoor, "Collective spammer detection in evolving multi-relational social networks," in Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '15. ACM Press, pp. 1769–1778. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2783258.2788606

7 Appendix

7.1 Exploration Figures

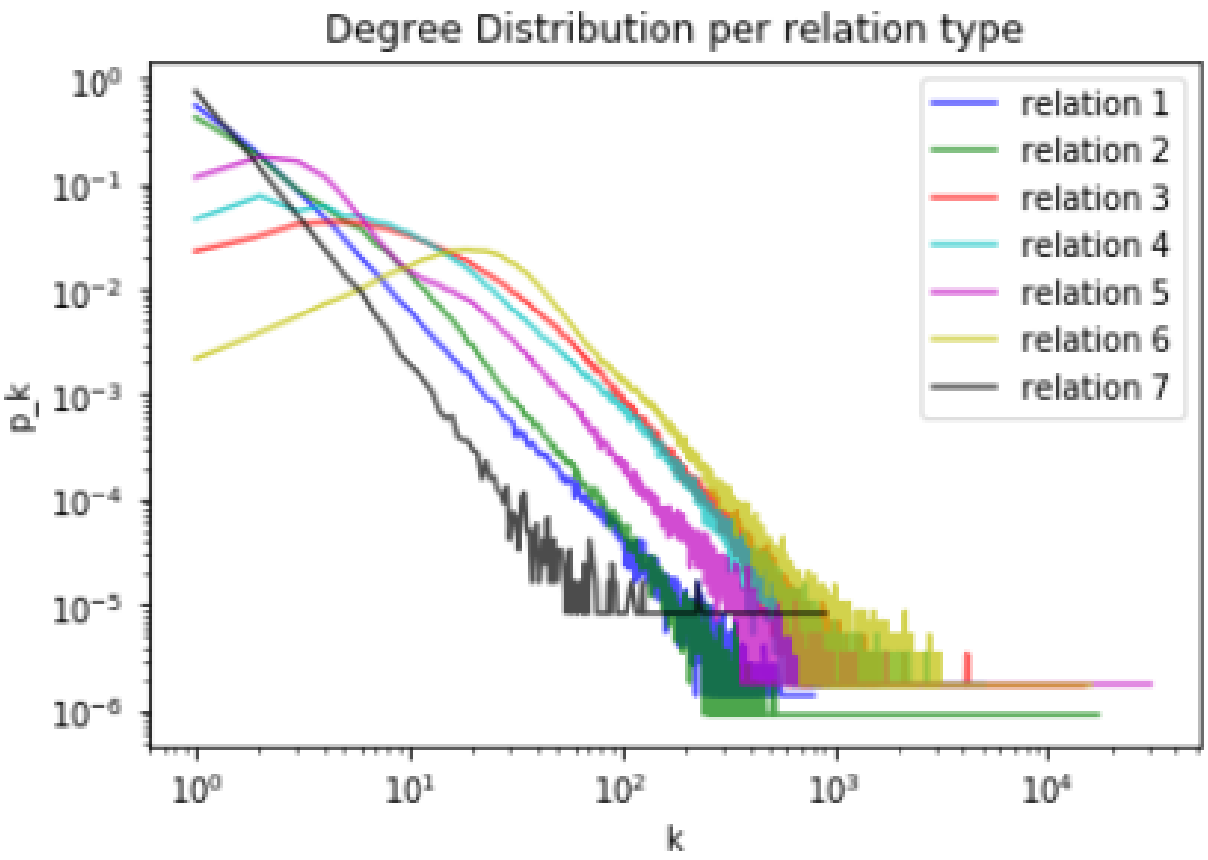


Figure 7.1: Degree distribution of all graphs used for graph based features extraction with sampling methods

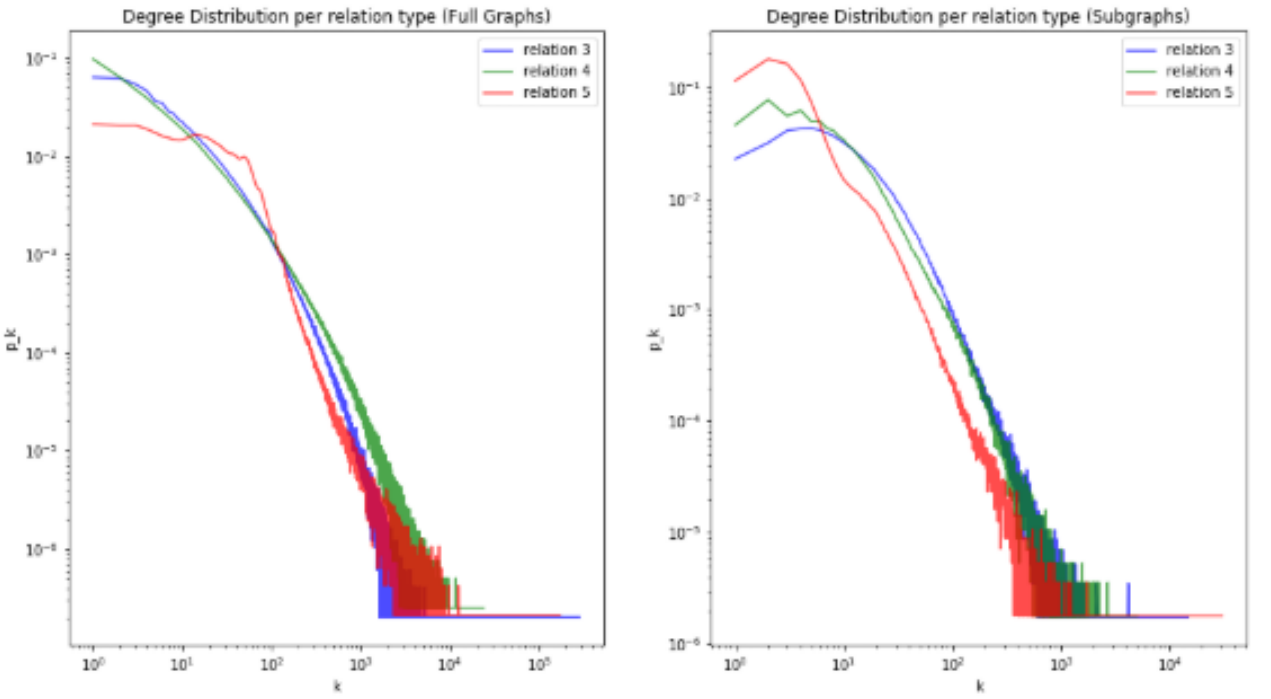


Figure 7.2: Comparing the degree distributions between the full graphs and the subgraphs (after sampling) for relations 3,4 and 5

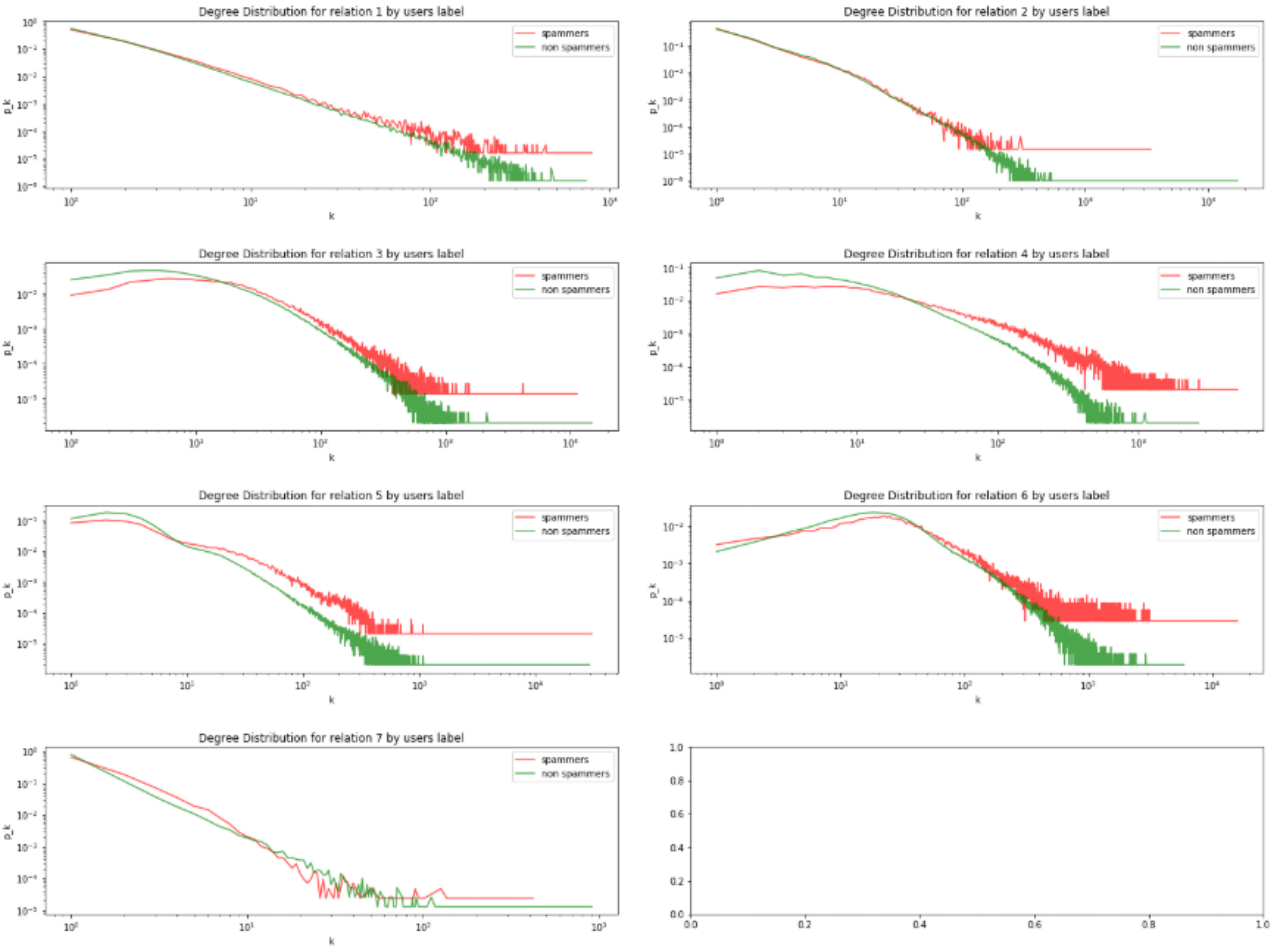


Figure 7.3: Comparing the degree distribution between spammers and non spammers for every graph

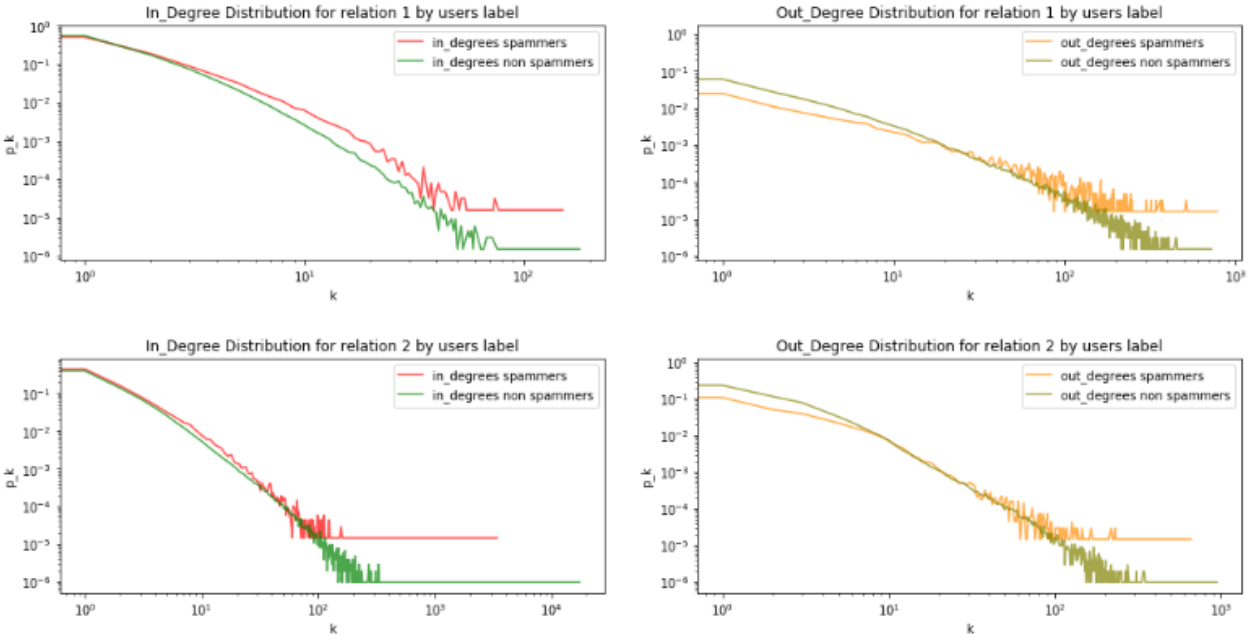


Figure 7.4: Comparing in_ and out_ degree distributions between spammers and non spammers for relations 1 and 2

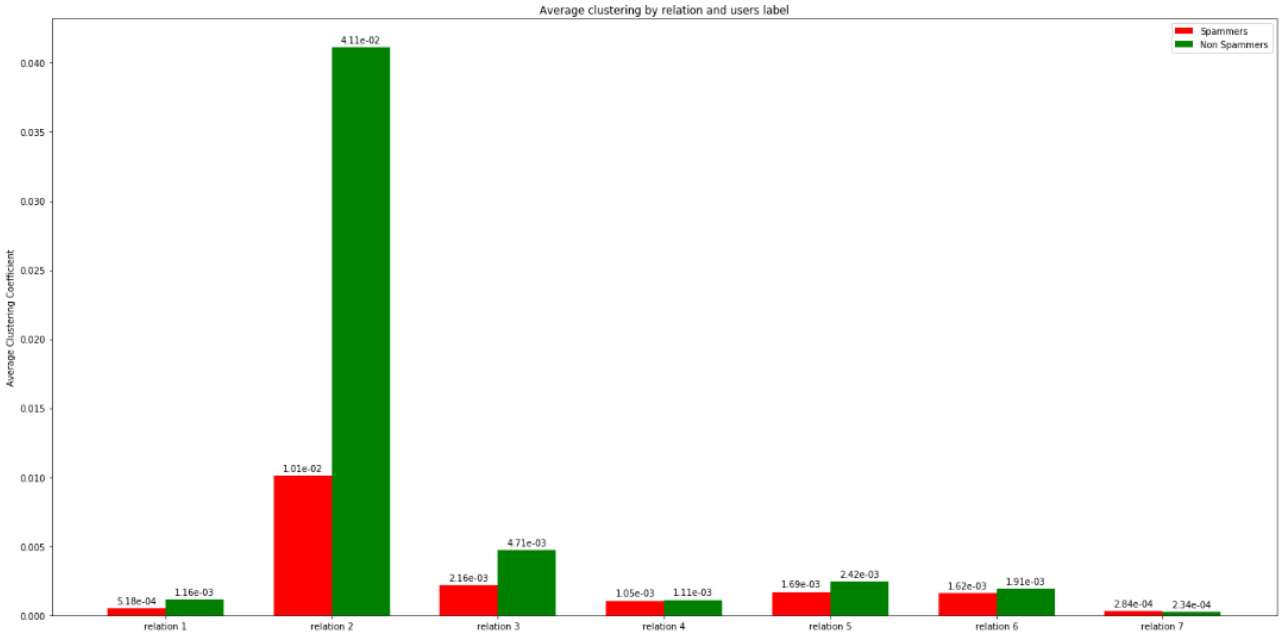


Figure 7.5: Comparing average clustering coefficient between spammers and non spammers for every relation