Predicting Positive Links in an Online Social Network

Omer Yizhaq

Abstract

Online social networks are social structures made up of a set of actors and the interactions they create with one another through the web. Those interactions, or links, may have a positive or negative nature.

Previous work [1] has shown that the signs of links in online social networks can be predicted with high accuracy, using models and common features of social networks which describe the relationship between the networks` participants.

Focusing on the network of Wikipedia's elections for adminship, I find that when predicting positive links in the network, negative links have an equal contribution to that of positive links. That is, knowing who are the characters one undervalues is just as useful as knowing who he thinks highly of to predict the presence of additional characters he appreciates.

1. Introduction

People form links with one another over the web. While some of the links formed have a positive nature, indicating trust, approval or friendship, other links have a negative nature, indicating disapproval or disagreement. In Wikipedia, users vote for or against the promotion of others to adminship, creating positive or negative links respectively.

In the work of J. Leskovec et al. [1], it was shown that the link from one Wikipedia user to another and its sign can be predicted with high accuracy using machine learning models. The models were based on features describing the aggregated relations between the two individuals to their environment, and the relationships they have with their common acquaintances. The intuition behind this is that the relationship with common figures implies the nature of the link formed between the users.

A fundamental question encountered in the mentioned above work was the contribution of negative edges to the positive edges prediction problem - how useful are negative links when one wishes to predict the presence of positive links?
In order to answer this question, the accuracy of the predicting model designed was estimated under 2 different circumstances:

In the first case, only information derived from positive edges was used to predict the presence or absence of other positive edges, while in the other case information derived from both positive and negative edges was used to predict positive edges.

While it was found that the accuracy of the model predicting positive links between Wikipedia users was increased from 0.6983 to 0.7114 (improvement of +1.88%) when adding information derived from negative edges, the question regarding the contribution of negative edges remains intact as it is hard to estimate their contribution without any point of reference.

This paper aims to evaluate the contribution of negative edges to the positive edges prediction problem. This, by comparing it to the contribution of positive edges. For this purpose, the variable of each edge type`s contribution was isolated in a manner specified under section 3.

2. Dataset Description

Wikipedia is a multilingual open-collaborative online encyclopedia. It was created in 2001 and is maintained by an active community of volunteer editors ever since.

Some of the community members are administrators, having access to additional technical features that aid in maintenance. The admins are chosen by the Wikipedia community in a process called Request for Adminship (RfA):

Users can submit their RfA or may be nominated by other users. After the nomination is posted an active discussion takes place, during which users may give their opinions, ask the candidate questions, and make comments.
The discussion will usually end in 7 days, then a bureaucrat (mostly an admin who has extended permissions) will review the discussion to see whether there is a consensus for promotion.
Although there isn`t a numerical threshold for consensus at RfA, most RfAs Above 75% support pass, almost all RfAs below 65% fails, and RfAs between 65%-75% are subjected to the bureaucrat`s decision based on the strength of the rationales presented in the discussion.

The online social network studied is based on RfA votes which took place between April 2004 - January 2008:
Links are directed from voters to the community members they voted for or against. A signed link indicates a positive or negative vote by one user regarding the promotion of another:
+1 for supporting vote, -1 for opposing vote. Neutral votes were not represented as links in the network.
In some of the votes the RfA of a nominee was already considered before. As a result, other users

could have voted several times in different elections involving the same nominee. In that case, the sign of the link from a voting user to a voted candidate was determined by the first non-neutral vote the voter has made.

Based on 2,794 elections, the resulting network contains 7,118 nodes (users participating in the elections as nominees or voters) and 103,747 directed and weighted edges (indicating the nature of non-neutral votes). Out of all edges 78.44% are positive, 21.56% are negative.

3. Work Methodology

The Wikipedia elections network may be represented as a directed and weighted graph $G = (V, E)$, with the users as vertices (or nodes) in the graph and the links they form with one another as directed and weighted edges. The two possible weights are -1,+1 – for a given directed edge from u to v $< u, v >$ the weight +1 indicates that the user u supported v as a candidate for adminship, the weight -1 indicates his opposition of the promotion in question.

In order to estimate the contribution of negative edges to the prediction of positive edges, the negative edges` contribution to the accuracy is compared to the contribution of positive edges.

3 variants of the graph were made for the purpose of this comparison:
Given the directed and weighted graph representing the Wikipedia`s elections $G = (V, E)$, let N be the group of all negative edges in E, N = $\{e \in E \mid weight(e) = -1\}$.
Let P be another subgroup of E, made of randomly selected positive edges such that |P| = |N|.
Variant 1 is the Wikipedia`s elections graph, Given by $G_1 = (V, E) = G$.
Variant 2: The Wikipedia`s elections graph, without the edges from N: $G_2 = (V, E/N)$.
Variant 3: The Wikipedia`s elections graph, without the edges from N nor the edges from P: $G_3 = (V, E/(N \cup P))$
Variant 4: : The Wikipedia`s elections graph, without the edges from P: $G_4 = (V, E/P)$

Let S(X) denote the accuracy of a prediction model trained on the information derived from the set of edges X. Then the improvement in accuracy of models trained on variant 4 compared to variant 3 = $S(E/P) - S(E/(N \cup P))$ measures the contribution of negative edges.
The improvement in accuracy of models trained on variant 2 compared to variant 3 = $S(E/N) - S(E/(N \cup P))$ is a measure of the contribution of positive edges.
Having those two measures, examination of the contribution made by negative edges would be more accurate in the light of the contribution made by positive edges and would help evaluate it as

well.

4. Predicting Positive Edges

Given a full network or one of its variants represented as a directed and weighted graph, we wish to predict the absence or presence of a positive edge from a node to another.

4.1 A Machine Learning framework

Balancing the train data and evaluating the models` performance using 10-fold cross validation variant: For each pair of nodes in the graph (when the order of nodes sets the direction of the edge between them, if exists), a set of features describing the relationship between them was calculated. The models are trained and evaluated based on the calculated features, which are described bellow.
Examination of the features calculated would reveal that node pairs which are not connected are extremely overrepresented in the features dataset: this principle can be illustrated by considering the first variant, the full graph:
The graph representing the Wikipedia's election for adminship contains 7,118 nodes and 103,747 edges. The features calculated based on this graph contains 7,118x7118=50,665,924 node pairs, out of which 103,747 are connected by any edge and 50,562,177 are not connected whatsoever, creating a relation of 487:1 for node pairs not connected.
In the other graph variants from which edges were removed this relation is even more radical.
This sort of imbalance might interfere with the models` evaluation:
When training the models and testing them, the features dataset is split to test data and train data. In any nonspecific split, both the portions would be imbalanced. The model will "train" to predict there is no edge almost always, then it will demonstrate impressing accuracy on the test data caused by the fact it is highly imbalanced as way.
Preprocessing the test data to make sure it contains equal amounts of pairs connected by a positive edge and pairs not connected by any edge will also produce non informative results: the model trained to predict absence of edges will have an accuracy of about 0.5, the result of successful predictions for all the pairs not connected by an edge which compose half the test data.
In a same way, preprocess of the train data will also result in an accuracy of about 0.5 which is not a true representation the model`s performance.

Thus, it is crucial that both the test and train data will be balanced. This balance is achieved by under sampling pairs not connected by an edge using a 10-fold cross validation variant:
In the classic 10-fold cross validation method, the data is split to 10 equal portions. In each iteration of 10 iterations, 9 portions are used as train data while the remained portion is used as test

data. The overall performance of the model is its average performance in all iterations.
In the variation designed, 10 disjoint portions were selected out of the features data set such that each portion contains almost equal amounts of node pairs connected by a positive edge and node pairs not connected in any edge.

The balanced portions contain all the node pairs connected by a positive edge, a minority of the node pairs not connected by any edge, and none of the node pairs connected by a negative edge[1]. Then the models are trained and tested based on the 10 portions in the same way as in 10-fold cross validation, which means that node pairs not contained in any of the portions will not be used for training the models nor testing them.

The portions were also balanced according to the number of common neighbors for each node pair, which will be referred as their embeddedness value:
For each node pair connected by a positive edge, another node pair not connected by any edge that has the same embeddedness value was added to the portion, if exists. As the work of J. Leskovec et al. [1] has shown, the embeddedness value of the node pairs affects the accuracy of the models and so balancing the different embeddedness values will create a more informative test data.

Features. For a given node pair <u, v>, a triad involving those nodes is consisted of them both and a common neighbor of u, v, w: w is a node which has an edge either to or from u and an edge wither or to v as well.
There are 16 different triads types: The edges connect (u, w), (v, w) has 2 possible directions and 2 possible weights (+1 or -1), which leads to 2x2x2x2 = 16 distinct triads.
The features for each node pair are the counts of each triad type they both participate in. As the triads counts describes the aggregated relationship they have with their common neighbors, it might imply the relationship they share or not share, reflected in a directed edge.

Every common neighbor of 2 nodes raises their at least one of their triad counts by one. Thus, the greater the embeddedness value of a node pair, there is more information describing it, resulting in a better model`s performance is. Following the work of J. Leskovec et al. [1], the features dataset was restricted to node pairs with a minimum embeddedness value of 25.

Embeddedness value calculation. Given a directed and weighted graph $G = (V, E)$, it is possible to calculate an embeddedness matrix $E_{i,j}$ of the size $|V|x|V|$, in which the number in the cell $[i][j]$ equals to the number of common neighbors for the vertices i and j.

[1] It is important to remind that although node pairs connected by a negative edge where absent from both the train and test data, the features were calculated based on negative edges (together with positive edges), if existed in the graph variant.

With time complexity of $O(|V|^2)$ and space complexity of $O(|V|^2)$, it is a preferred solution for the problem:
Starting with an adjacency matrix $A_{i,j}$ representing the graph, it is needed to create a normalized matrix A_N so instead of 1 or -1 representing positive or negative edges in the graph, 1 will represent edge of any sign. As common with adjacency matrixes, 0 represents no edge.
Then A_N^T the transposed form of A_N is added to A_N to create the matrix B, since while the graph is directed the number of common neighbors doesn`t consider edges direction.
Finally, we multiply B and B^T: The result of that operation is a matrix $M_{i,j}$ where every cell $[i][j]$ is the sum of the multiplication of row i in B and column j in B^T – which will yield the numbers of corresponding cells that both equal 1 and sum them. Because every cell represents whether there is an edge between two nodes, and every row represents the edges of a single node, this vector multiplication will yield the number of common neighbors and thereby this matrix multiplication will yield the matrix $E_{i,j}$.

Learning methodology. A logistic regression classifier was used to predict positive edges.

Logistic regression is a 'Statistical Learning' technique dedicated to classification tasks. In that case, the classification of node pairs to node pairs which are connected by a positive edge or not.

The decision function is a logistic function, or sigmoid function of the form
$P(+|X) = \frac{1}{1+e^{-f(X)}}$, where $f(X)$ is a function consisting our features and their corresponding coefficients (which are estimated based on the training data) in a linear form:
$f(X) = b_0 + \sum_i^n b_i x_i$

4.2 Results

4 models where trained and evaluated on 4 different features datasets:
The first features dataset was calculated based on variant 1 of the graph, which is the full graph representing the social networks contains all positive and all negative edges: $G_1 = (V, E) = G$.
The second features dataset was calculated based on variant 2 of the graph, which is the graph without negative edges: $G_2 = (V, E/N)$.

The third features dataset was calculated based on variant 3 of the graph, which is the graph without negative edges and without the positive edges from P: $G_3 = \big(V, E/(N \cup P)\big)$.
The fourth features dataset was calculated based on variant 4 of the graph, which is the graph without the positive edges from P: $G_4 = (V, E/P)$.

Table 1 shows the predictive accuracy for positive edges of the models trained based on the features calculated for the different variants:

Table 1: Predicting the presence of positive edge

	Features	S = Accuracy
Variant 1	$G_1 = (V, E) = G$	0.73
Variant 2	$G_2 = (V, E/N)$	0.71
Variant 3	$G_3 = (V, E/(N \cup P))$	0.67
Variant 4	$G_4 = (V, E/P)$	0.71

As expected, the accuracy of the models trained on the features derive from the entire graph given by variant 1 is higher than the accuracy of the models trained on the features derived from positive edges only (variant 2) and the other variants from which edges were omitted.

The accuracy of the models trained on variant 1 and 2 are 0.73 and 0.71, which is slightly higher than the accuracy of the models designed by J. Leskovec et al. [1] (0.7114 and 0.6983, respectively). In addition, the difference in the level of accuracy of the models trained on variant 1 and the models trained on variant 2 is 0.0131 in the mentioned above work, compared to 0.02 for the models I used.
These differences could be explained by differences in the train and test methods used: 10-fold cross validation was used here, as opposed to leave-one-out cross validation performed for each positive edge in the datasets. Another possible explanation is that while I used the default logistic regression classifier provided by the library I chose, it is possible that parameters tuning done by J. Leskovec, D. Huttenlocher and J. Kleinberg has caused differences in the models` performance.

Table 2 sums the contribution of positive edges and negative edges. As mentioned in section 3, the positive edges contribution is expressed by the difference of the accuracy of the models trained on G_2 and G_3, which is $S(G_2) - S(G_3)$. Negative edges contribution is expressed by the difference of the accuracy of the models trained on G_4 and G_3, which is $S(G_4) - S(G_3)$.

Table 2: The contribution to accuracy of each edge type

	Contribution to Accuracy
Positive edges	$S(G_2) - S(G_3) = 0.04$
Negative edges	$S(G_4) - S(G_3) = 0.04$

It is vividly clear that the contribution of negative edges is equal to the donation of positive edges. This phenomenon could be explained by the nature of the RfA process in Wikipedia:
The public votes for adminship are encouraged to be based on solid rationales rather than emotions towards individuals. Among the criterions voters are taking into consideration when voting for or against a candidate are his seniority and level of activity in the community, and sometimes their agreement or disagreement with his editing choices.
When predicting whether a user voted for a candidate, a fundamental question is whether they share the same standards when it comes to the factors justifying support for adminship – if they do, it is likely that the user will vote for the candidate based on the last being up for the user`s standards.

This sort of agreement can be expressed by relationships of both approval and disapproval involving common figures, making relationships of disapproval equally valuable to relationships of approval for positive relationships prediction.

5. Conclusion

The experiment performed has showed that although the positive edges prediction problem involves only the positive relationships in the network, negative relationships have the same donation as positive relationships in the network of Wikipedia`s elections for adminship.

An interesting direction suggested by this work is to extend this comparison to a variety of social networks, as the weight of negative connections might vary due to differences between networks.

I am also interested in trying different classifiers with the same features to test their performance and the weight of the negative edges compared to positive edges with different classifications.

6. References

[1] J. Leskovec, D. Huttenlocher and J. Kleinberg. Predicting positive and negative links in online social networks. 2010