Predicting Reverts from WikiGraph Properties

Saravana Chellappan
Stanford University
saravanc@stanford.edu

Benjamin Pastel
Stanford University
bpastel@stanford.edu

Zack Swafford
Stanford University
zswaff@stanford.edu

Abstract

We investigated three networks related to edits on Wikipedia, and found graph metrics that strongly predict whether an edit is reverted. Specifically, we investigated the Article-Article graph, with edges between articles that link to each other; the User Revert graph, with an edge between users that have reverted each others' edits, and the temporal User-Article Graph, with an edge between each user and the articles they have edited. Metrics such as article PageRank, the ratio of in-degree to out-degree of an article, and the history of a user with a given article or similar articles proved to be highly correlated with edits being reverted. We demonstrate the added value of these network metrics against a baseline regression using non-network features.

1 Introduction

Wikipedia, launched in 2001, has been open to contributions, edits, revisions, and updates on its articles from the public since its inception. Any visitor to the site can start contributing to the articles immediately. Changes are applied automatically, with no need for systemic approval. This collaborative system is the basis of Wikipedia's success, but of course not all edits made are up to the site's quality standard. In some way or another, a portion of edits are classified technically as vandalism and are reverted by other uses when noticed.

In an attempt to monitor, curtail, and remedy vandalism to the encyclopedia, Wikipedia introduced several targeted datasets (e.g. [6]) that can be used to study vandalism. The techniques and algorithms used solve this problem have developed extensively over the years and many NLP techniques have been applied [9, 4].

Much of the difficulty in solving this vandalism problem enters in simply defining vandalism, which is a somewhat nebulous concept. We have avoided this problem by studying a metric which is closely related and, we think, potentially more relevant: we examine edits that are reverted on Wikipedia, i.e. edits on an article that are completely undone by a later edit to an article.

Prior study of both vandalism and reverts uses NLP on the edit content, and machine learning on the edit metadata. But to our knowledge no current approaches use the structure of the Wikipedia graph to aid prediction. In this paper we identify several key graph metrics and demonstrate their significant predictive power over a baseline model.

2 Related Work

There is a relatively large body of work on classifying Wikipedia edits using NLP and other features inherent to the edits themselves [9, 7, 2, 10]. This approach is interesting, intelligent, and effective– but it could be made more so by a more rigorous exploration of network properties.

The closest work to the research we conducted is West, Kannan, and Lee [12], which discusses using spatial and temporal techniques to detect Wikipedia vandalism for the first time. The article

specifically mentions that it will use both geographical and graph-based measures of space and spatial positioning. Less relevant to our work here, the authors also incorporate time-based information into their eventual analysis. The graph structure features chosen were particularly relevant to our research. Specifically, 'User Reputation', 'Article Reputation', and 'Category Reputation' were all calculated with graph theoretical models. However, this work falls short of its goal of thoroughly exploring network properties for a classifier.

Other particularly notable work was conducted by Shachaf and Hara [8], which examines the case of Internet trolls at large, but particularly in the microcosm of Wikipedia and with the lens of graph theory. It is a much more humanistic, almost anthropologic look at Wikipedia vandals. It delves deeply into their psyches and motivations and examines what might be really driving a career troll in a deep way. These authors were, unsurprisingly, missing a rigorous, data-driven approach. Although the article really clues us in to many issues and may even help to inform the theory and modeling of these relationships, without a more analytically approach it is difficult to read deeply into any of the anecdotal case studies provided.

Despite this theory and speculation, no automatic system for classifying vandalism or bad edits uses features of the network such as the links between articles, or the relationships between users who revert each other or edit the same articles.

3 Data Collection

Wikipedia itself has released several datasets particularly tailored to vandalism [6]. However, these datasets are generally hand-labeled and very small.

We have therefore extracted what we need from a more complete dataset. The Snap Wikipedia dataset contains metadata for the full edit history of Wikipedia up to 2008, including edit comments and inter-article links for ≈ 116 million edits [5].

The unknown users in this dataset are represented by an IP address. We treated these IP addresses the same as user addresses throughout our analysis.

First, we inspected the first 2,000 edit comments manually and found that approximately 1 in 10 contained the word "revert" or the abbreviation "rv", and that these comments generally referred to the previous edit or consecutive sequence of edits by one user. Although it was not always possible to definitively identify which edit a revert referred to, we decided to mark all such comments as reverts and assume they referred to the previous consecutive sequence of edits by the same user. Following this process, we marked ≈ 7.0 million edits as reverts, and ≈ 9.7 million edits as reverted.

Second, we parsed the inter-article links from the same dataset. To simplify analysis we replaced article names with integer ids and dropped all except the latest versions of each article.

For our prediction setup, we sorted the edits by increasing timestamp. The final 1 million rows were used as the test set; the previous 10 million rows were used as the training set; and the first $\sim$ 105 million rows were included in the feature calculations but not used directly as training rows, to improve runtime.

4 Baseline Without Network Features

We first performed a logistic regression using all of our data, but without containing any network-specific features. The improvement over the baseline in our final result demonstrates the value of features derived from network characteristics.

We included the following features:

- Is the user anonymous?
- Count of previous bad edits for this user.
- Count of previous total edits for this user.
- Ratio of bad to total previous edits for this user.
- Count of previous bad edits for this article.
- Count of previous total edits for this article.

```
 - Ratio of bad to total edits previous for this article.
```

And had the following performance:

```
F1 score: 0.628

Out of 1 million test rows:
 - False positives: 53,326
 - True positives: 55,473
 - False negatives: 12,368
 - True negatives: 878,833
```

5 Network Investigations

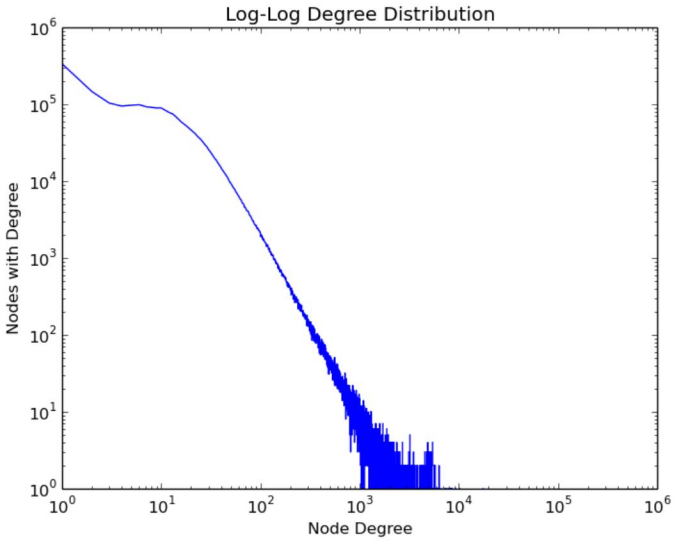
We examined the structure of three relevant networks to find features for use in predicting reverts:

- The Article-Article Network, with directed edges for each link between articles.
- The Revert Network, with directed edges between users who reverted each others' edits.
- The User-Article Network, with edges between users and articles they have edited.

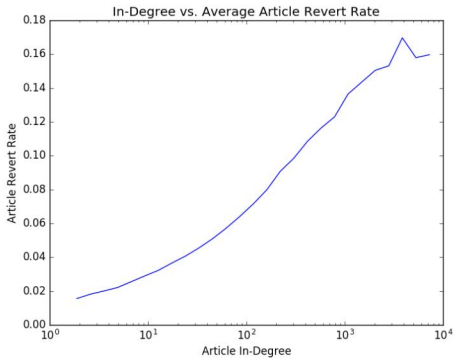
We also investigated the User-User co-edit network, where an edge exists between two users if they both edited the same article, but found that it to be intractable with 8 billion edges.

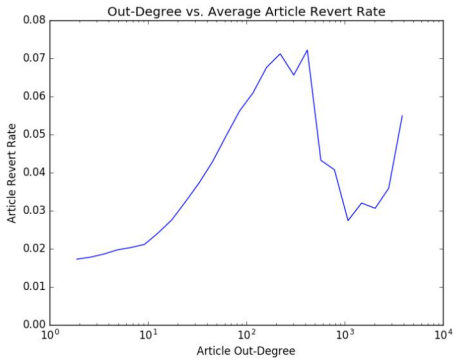
5.1 Article-Article Network

The first network we examined is the graph where each article is a node, and there is a edge directed edge (A, B) if article A has a link to article B. We found some surprising relationships in this graph that led to effective features for predicting reverts.



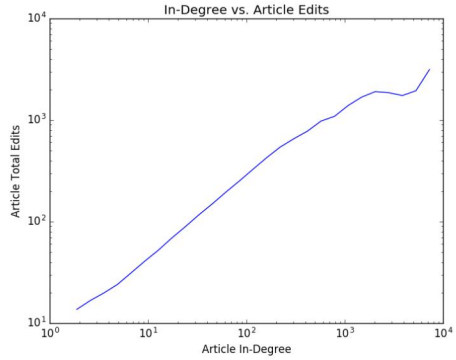
The degree distribution follows something close to a power law, but with a hump on articles with 3 or 4 links. We found the degree of an article to be a strong predictor of the article having a high revert rate:

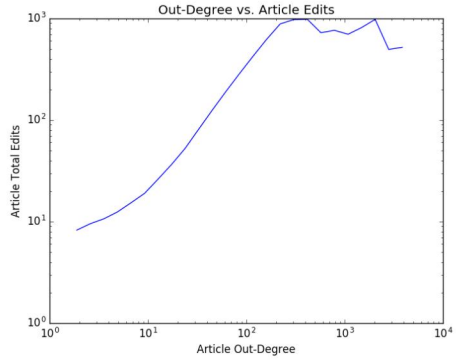




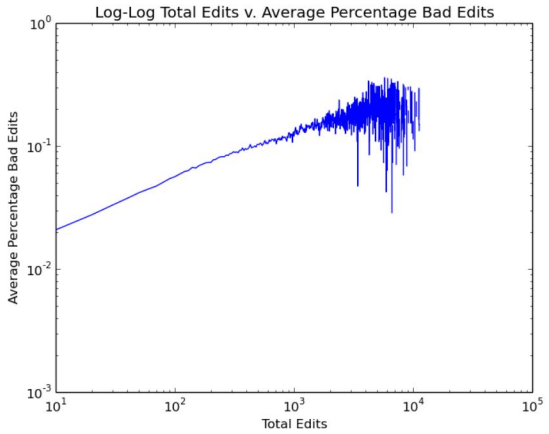
The reason for the dip in revert rate for graphs with high out-degree appears to be that there are some "aggregation" articles with many out-links and little controversy. In particular, each year has its own article, and the article content is just a list of links to notable events that occurred during the year.

Degree distribution has a similar relationship with edit count; presumably, the articles that receive more edits are the larger, more popular articles that also tend to have many in-links and out-links.

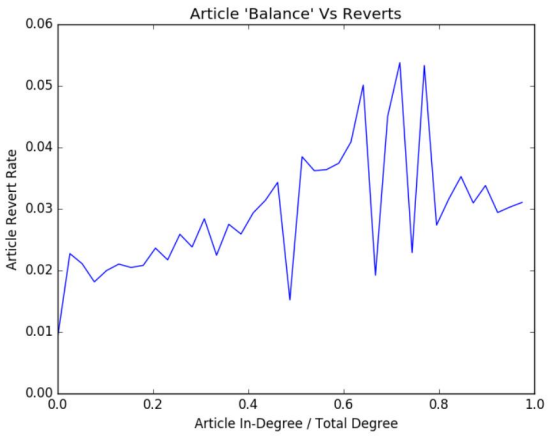




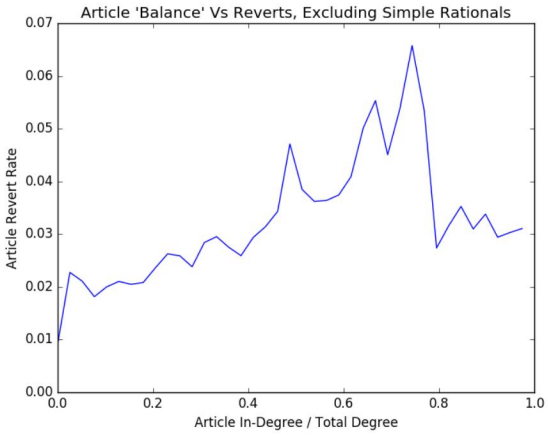
Accordingly, the total edits of an article were also tightly related to the article revert rate:



This relationship between edit count and revert rate caused a lot of fascinating patterns in the graph metrics. For example, one metric of how "balanced" an article's links are is the ratio of the sum of the edits of an article's in-neighbors with the sum of the edits of both the in-neighbors and out-neighbors. If this is close to even, then the articles that link to the given article are roughly balanced in popularity with the articles that this article links to.

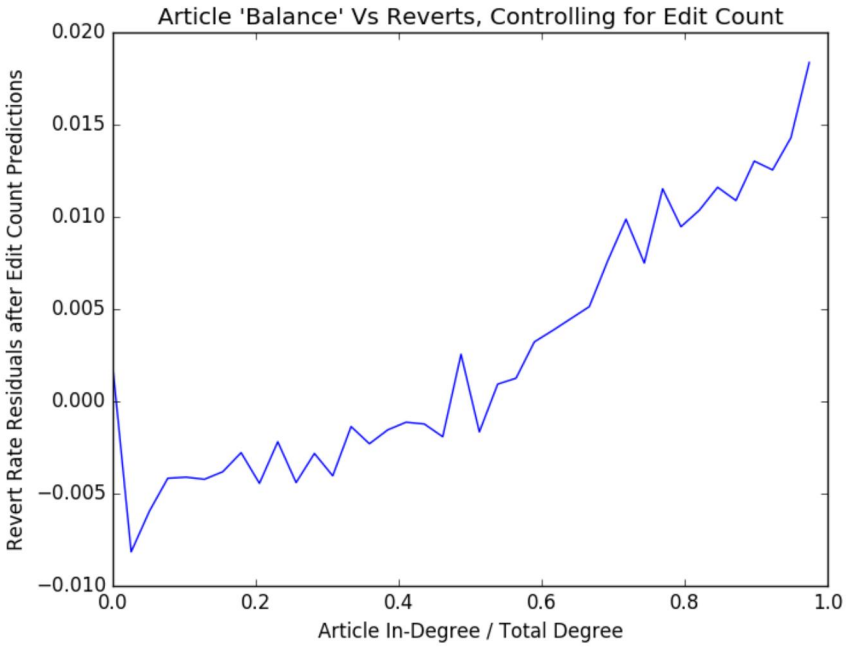


It looks like noise, but actually the above graph has a dip at simple rational numbers on the x-axis - $\frac{1}{3}, \frac{1}{2}, \frac{2}{3}$, etc. Why? Articles with very few edits are more likely to be simple rational fractions, and articles with very few edits tend to have a low revert rate. Here's the same graph, excluding the **exact** x-values $x = \frac{a}{b} | a, b \in (1, 2, ...5)$:



Note also the dip at both ends. Why would extremely unbalanced articles - with either many more in-links than out-links OR many more out-links than in-links - have fewer reverts than balanced articles? It turns out that articles with more edits tend to be more well-balanced. These bizarre patterns appear to be merely secondary factors caused by the underlying edit counts.

To control for edit count effects, we performed a simple logistic regression predicting the article's revert rate with two features: article edit count and $\log$(article edit count). Then, we found the residuals by subtracting those predictions from the observed revert rates. We found that a high in-degree and a large edit count among the in-neighbors predicted high residuals but out-degree and a high edit count amount the out-neighbors did not predict high residuals. And balanced articles are no longer the worst; instead, the higher the proportion of in-links, the worse the residuals.



From the above analysis, we tried adding the following Article-Article network metrics to the logistic regression:

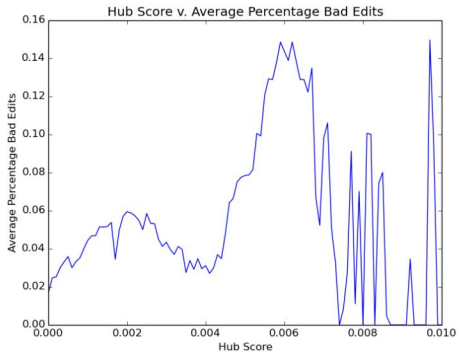
- Average edits of out-neighbors
- Average edits of in-neighbors
- Average bad rate of out-neighbors
- Average bad rate of in-neighbors
- Ratio of in-degree to total degree

The addition of just these features to the baseline increased the F1 score from **0.628** to **0.718**. As expected, the ratio of in-degree to total degree had a large positive coefficient, indicating that it is a significant risk factor.

One interpretation of these results is that both edit count and in-degree measure different aspects of the popularity or visibility of an article, and that popular or highly visible articles are more likely to have a high proportion of reverts. We next investigated several popularity-related metrics: PageRank and Hub and Authority scores.

PageRank, for example, would be correlated to percentage of bad edits because nodes with a higher rank get more attention as they are central to browsing. This means that vandals are more likely to target the page, but also that counter-vandals are more likely to detect and revert vandalism more quickly. Hub and Authority scores were included for similar reasons. Indeed, all of these metrics exhibited statistically significant correlations with article revert rate.

Below is a plot of hub score against percentage of edits that are bad. The relationship is not monotone, although we believe that the positive trend of bad edit percentage as hub score increases is only interrupted by lack of data. Nonetheless, the hub score was still a useful feature in the logistic regression.



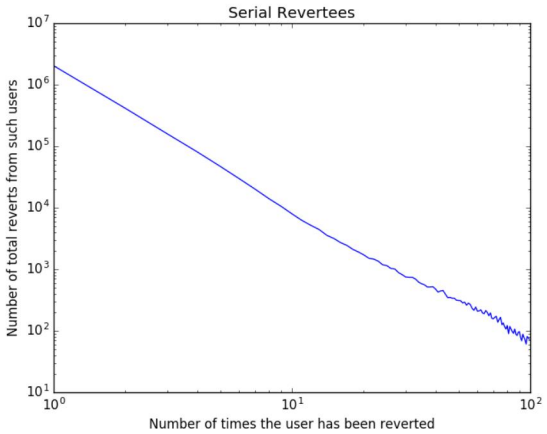
Adding just PageRank, hub and authority scores, in-degree, out-degree, total-degree, and clustering coefficients, the F1 score of the logistic regression on the test set increases from **.628** to **.689**. There were far fewer false positives but actually more false negatives with these additional features.

5.2 Revert Network

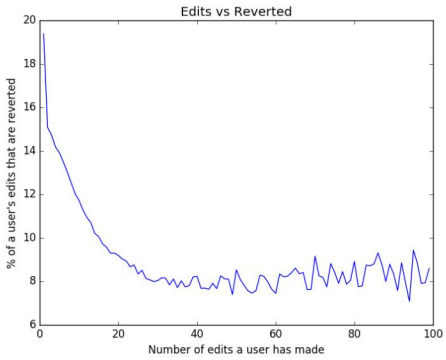
Our second network is the directed graph with a node for each user, and an edge (A, B) if user A reverted user B's edit. In this section, we will refer to user A as the "reverter" and user B as the "revertee" with respect to the revert edge (A, B).

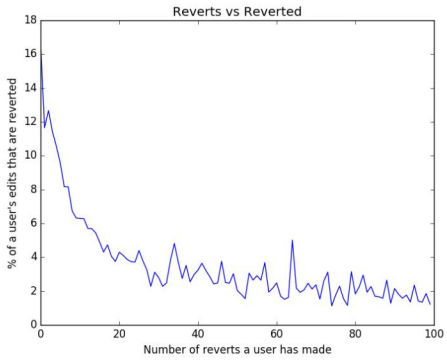
There are about 11 million users in our dataset, and 6.8 million reverts. A very small fraction of users are reverters; only around 150 thousand users have ever reverted an edit. By contrast, 2.9 million users have had an edit reverted.

First we examine the properties of revertees. The number of times a given user has been reverted follows a power law:



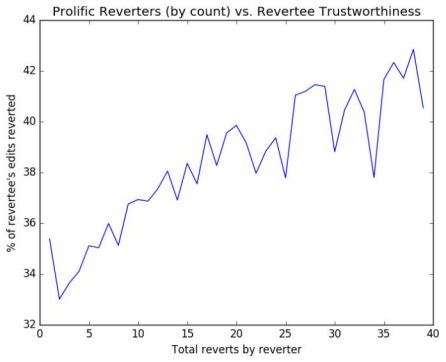
By maximum likelihood estimate, we found $\hat{\alpha} \approx 1.16$. This is a very low $\hat{\alpha}$, so a lot of the mass is concentrated in "Serial Revertees"; i.e. users who have been reverted many more than the average times. We looked at various properties of the revertee, to try to find characteristics that predict which nodes are reverted many times.

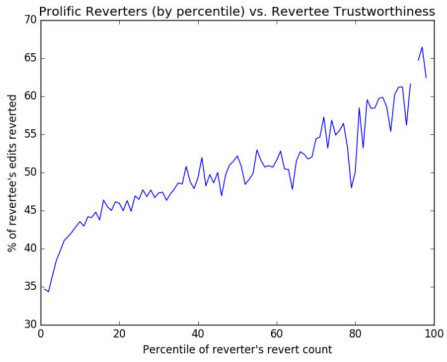




We can see that both the number of overall edits a user has made, and the number of reverts have a strong relationship with the proportion of the revertee's edits that are reverted.

Next, we looked at how the characteristics of the reverter relate to the characteristics of the revertee. We found that the out-degree of the reverter was strongly correlated with the percentage of the revertee's edits reverted. In other words, anyone reverted by nodes with high out-degree are more likely to be 'bad':





Unfortunately, these insights on the Revert Graph did not translate effectively to our prediction problem. Our final model does not include any features derived from this graph.

5.3 User-Article Edit Network

The User-Article Edit Network is an undirected defined as follow: the set of vertices which is the users and articles. There is an edge between user U and article T if the user U has ever edited T. There are about 2.9 million article nodes and about 11 million user nodes. And there are about 116 million total interactions between these two entities that evolved over time, with 59 million unique edges.

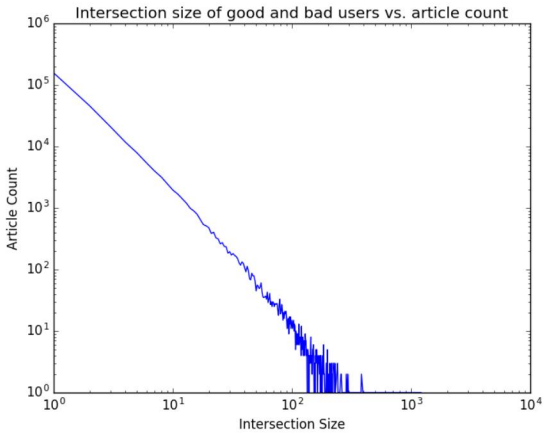
As [14] indicates that communities can play an important role in link prediction in collaboration network such as Wikipedia Edit Network. In such networks, one user plays different roles in different communities. Since this is a large graph with more than 14 million nodes, we attempted the CNM algorithm for community detection [11]. Unfortunately we found the algorithms to have prohibitively high memory requirements.

We next looked at user-article affinity. About 8.1 million users perform only good edits, 1.6 million users perform only bad edits, and about 1.2 million users perform both good and bad edits. On the articles side, about 721,000 articles have both good and bad edits, 2.2 million articles have only good edits, and 264 articles have only bad edits.

We studied the neighbors of the 721,000 articles with both good and bad edits. For each of article, we found interesting patterns in its neighbors. About 45% of these articles neighbors exhibit bipolarity,

i.e. the set of users that edit these articles is a bipartite graph with users doing good edits and reverts on one side and users doing bad edits which get reverted on another side.

The distribution of intersection of users that do good and bad edits and number of articles is shown in the following diagram. About 95% of the articles have intersection size <5, and 98% of articles have intersection size <10 and 99% of these articles have intersection size <20. These observations confirm that different users behave differently towards different articles.



Based on the above observations, our hypothesis is that the tendency of a user towards an article is influenced by past interactions with that article specifically. Since de-duping the edges removed the temporal data in the network, we augmented the graph with a temporal look up function as follows:

For a given edge and a time t, function returns the probability of a good edit and bad edit by a user on an article based on previous interactions of this user with that article for any time $< t$ in the evolution of network. The underlying data structure for the look up function is a key value store with ArticleId, UserId as the key and the good and bad probabilities as value. This can be computed at scale by using a Map Reduce system by partitioning the temporal edge list with edge sign (+ve for good edit and -ve for bad edit) between user-article in the edit network and computing GoodEditProbability and BadEditProbability for each combination.

By adding just these two new features to the baseline, we observed a significant improvement in F1 score over the baseline from **0.628** to **0.741**.

6 Results

Combining the features from the previous sections, our final result demonstrates a significant improvement over the baseline, only by adding network features to the logistic regression:

```
F1 score: 0.751
Out of 1 million test rows:
false positives: 23,948
true positives: 55,211
false negatives: 12,628
true negatives: 908,213
```

Compared to the initial F1 score of 0.628, this is a significant improvement.

Although we do not believe this result reaches state-of-the-art by itself, it fulfills our goal of clearly demonstrating the value of network metrics for predicting reverts on Wikipedia. Existing automatic classification and detection techniques for vandalism should be able to make significant improvement if they include our network features.

7 Contributions

Saravana: Analyzed the User-Article network; built the temporal data model, ran the final models; investigated Community Detection which proved too slow to use; investigated Wikipedia article category as a proxy for community structure and using [13] as signal.
Benjamin: Parsed the raw data; wrote the baseline model; analyzed the Revert Graph; analyzed neighbor properties in the Article-Article Graph.
Zack: Wrote literature review and explored the current state of the art in general; analyzed the properties of the Article-Article Graph; investigated User Coedit Graph and other related networks which proved less feasible or interesting.

References

[1] Luciana S Buriol et al. "Temporal analysis of the wikigraph". In: *2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2006 Main Conference Proceedings)(WI'06)*. IEEE. 2006, pp. 45–51.

[2] Damian Zaremba (DamianZaremba). *ClueBot NG*. 2016. URL: https://github.com/DamianZaremba/cluebotng (visited on 2016-11-12).

[3] Manoj Harpalani. "Wiki Vandalysis. Wikipedia Vandalism Analysis". PhD thesis. The Graduate School, Stony Brook University: Stony Brook, NY., 2010.

[4] Kelly Y Itakura and Charles LA Clarke. "Using dynamic markov compression to detect vandalism in the wikipedia". In: *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. ACM. 2009, pp. 822–823.

[5] Jure Leskovec. *Complete Wikipedia edit history (up to January 2008)*. 2016. URL: https://snap.stanford.edu/data/wiki-meta.html (visited on 2016-11-12).

[6] PAN. *Trust - Credibility Analysis - Wikipedia Vandalism Detection*. 2010. URL: http://pan.webis.de/data.html (visited on 2016-11-12).

[7] Martin Potthast, Benno Stein, and Robert Gerling. "Automatic vandalism detection in Wikipedia". In: *European Conference on Information Retrieval*. Springer. 2008, pp. 663–668.

[8] Pnina Shachaf and Noriko Hara. "Beyond vandalism: Wikipedia trolls". In: *Journal of Information Science* 36.3 (2010), pp. 357–370.

[9] Koen Smets, Bart Goethals, and Brigitte Verdonk. "Automatic vandalism detection in Wikipedia: Towards a machine learning approach". In: *AAAI workshop on Wikipedia and artificial intelligence: An Evolving Synergy*. 2008, pp. 43–48.

[10] Steve Goring (stg7). *Wikipedia Vandalism Detection Bot*. 2016. URL: https://github.com/webis-de/wikipedia-vandalism-bot (visited on 2016-11-12).

[11] Ken Wakita and Toshiyuki Tsurumi. "Finding Community Structure in Mega-scale Social Networks: [Extended Abstract]". In: *Proceedings of the 16th International Conference on World Wide Web*. WWW '07. Banff, Alberta, Canada: ACM, 2007, pp. 1275–1276. ISBN: 978-1-59593-654-7. DOI: 10.1145/1242572.1242805. URL: http://doi.acm.org/10.1145/1242572.1242805.

[12] Andrew G West, Sampath Kannan, and Insup Lee. "Detecting wikipedia vandalism via spatio-temporal analysis of revision metadata?" In: *Proceedings of the Third European Workshop on System Security*. ACM. 2010, pp. 22–28.

[13] Wikipedia. *Wikipedia:List of controversial issues*. 2016. URL: https://en.wikipedia.org/wiki/Wikipedia:List_of_controversial_issues (visited on 2016-11-12).

[14] Elena Zheleva et al. "Using Friendship Ties and Family Circles for Link Prediction". In: *Proceedings of the Second International Conference on Advances in Social Network Mining and Analysis*. SNAKDD'08. Las Vegas, NV, USA: Springer-Verlag, 2010, pp. 97–113. ISBN: 3-642-14928-6, 978-3-642-14928-3. URL: http://dl.acm.org/citation.cfm?id=1883692.1883698.