The Illiterate Editor: Metadata-driven Revert Detection in Wikipedia

Jeffrey Segall
Drexel University Department of Computer Science
3141 Chestnut Street
Philadelphia, PA 19104
js572@drexel.edu

Rachel Greenstadt
Drexel University Department of Computer Science
3141 Chestnut Street
Philadelphia, PA 19104
greenie@cs.drexel.edu

ABSTRACT

As the community depends more heavily on Wikipedia as a source of reliable information, the ability to quickly detect and remove detrimental information becomes increasingly important. The longer incorrect or malicious information lingers in a source perceived as reputable, the more likely that information will be accepted as correct and the greater the loss to source reputation. We present The Illiterate Editor (IllEdit), a content-agnostic, metadata-driven classification approach to Wikipedia revert detection. Our primary contribution is in building a metadata-based feature set for detecting edit quality, which is then fed into a Support Vector Machine for edit classification. By analyzing edit histories, the IllEdit system builds a profile of user behavior, estimates expertise and spheres of knowledge, and determines whether or not a given edit is likely to be eventually reverted. The success of the system in revert detection (0.844 F-measure) as well as its disjoint feature set as compared to existing, content-analyzing vandalism detection systems, shows promise in the synergistic usage of IllEdit for increasing the reliability of community information.

1. INTRODUCTION

Since its inception in 2001, Wikipedia has transformed from a burgeoning social idea, to a respected source of information, to the largest general reference work on the Internet. Evolving social policy and community structure have placed greater importance on information verification and consistency. So much so, that in 2005, Wikipedia was shown, at least in part, to have an error rate consistent with that of traditional encyclopedias and has been progressively improving [6].

As the reliabilty of Wikipedia grows, and with it the community's trust in the information it holds, the problems of vandalism and misinformation present more risk. Not only is this risk inherent in the use of Wikipedia as a source by end users, but also in the use of automated crawlers that gather information to either present or cache for later use. To ensure that all users receive the quality of information that has become expected of Wikipedia, it is important to identify and remove bad information as quickly as possible.

We present The Illiterate Editor (IllEdit), a content-agnostic metadata-driven classification approach to Wikipedia revert detection. Through an extensive feature extraction process, we have transformed a full edit history dump of Single English Wikipedia into a dataset for use in machine learning. Unlike other reversion and vandalism detection approaches, IllEdit performs no computationally-expensive content analysis of the discerning edit, relying solely on the editing user's community history and edit meta information. We also compare the IllEdit system to ClueBot NG, a leader in automatic Wikipedia vandalism detection, and discuss the utility of both algorithms working in parallel. It is important to note that the IllEdit algorithm is meant to detect reverted edits, not gross vandalism. Vandalism is a subclass of reverted edits that exhibits a willful misrepresentation of information or defacement. We present an algorithm that can be used to detect edits that are reverted based on accidental misinformation as well.

With a 96.8% correct classification rate, a 0.844 F-measure, and a 0.823 Matthews Correlation Coefficient, the IllEdit algorithm provides an alternative approach to existing revert and vandalism detection algorithms. In initial comparison, the overlap in reverted edit detection between IllEdit and ClueBot was minimal. Of the reverts detected by the ClueBot NG algorithm, 63% were missed by IllEdit, showing the potential for a combined effort with the IllEdit and ClueBot systems running in tandem.

The paper is organized as follows: Section 2 describes related work. Section 3 describes our experimental design and approach. Section 4 presents and analyzes our experimental results, as well as compares our results to those of other approaches. Section 5 addresses possible threats to validity. Section 6 discusses potential future work. Section 7 concludes.

2. RELATED WORK

Due to its prominence, longevity, size, and openness, Wikipedia has become a staple research platform for topics such

as visualization, natural language processing, and social interation.

Javanmardi et. al. define and explore a Wikipedia Trust Metric (WTM) [8] that represents the level of trust one can put into a Wikipedia user. The value is calculated using Hidden Markov Models and and amount of user contribution that remains in articles over time. They then determine the levels of trust inherent in both registered and unregistered users [9], showing that while on average, registered users were more trustworthy, the WTM algorithm leads to a situation where approximately half of Wikipedia users had trust values either below 10% or above 90%. Similarly, Adler and de Alfaro propose WikiTrust, a reputation system for both Wikipedia users and content where trust is placed in users whose content is preserved through subsequent edits [4]. They employ their algorithm as a part of an experiment in the space of vandalism detection and, like IllEdit, include the consideration of metadata in their decision making, but to a much lesser degree [1]. Instead, their approach looks more closely at content-based features such as ratios of uppercase text (used to draw attention), pronoun frequency (biased writing style), and lists of biased or restricted words.

The ClueBot NG algorithm [3], one of Wikipedia's most respected vandalism detection bots, employs deep content analysis using Bayesian filtering and Artificial Neural Networks to detect vandalism in real-time from Wikipedia feeds. ClueBot NG also relies on a community consensus for vandalism ground truth determination. The IllEdit algorithm hopes to provide a contrast in process and use ClueBot as both a performance benchmark and as a potential partner in revert detection.

Suh et. al. [16] describe a visual analysis tool that presents patterns of user behavior in Wikipedia edit histories. Like IllEdit, the tool uses hashing to detect reverted edits, but also uses regular expression searches of edit comments to detect partial reverts.

Raph Levien's Advogato score shows an attack resistant trust metric for collaborative communities, but relies on the definition of roots of trust [12]. Though there exists a notion of structure and authority, these starting points may or may not exist in the Wikipedia community.

Kamvar, Schloser, and Garcia-Molina [10] analyze reputation in peer-to-peer file sharing networks. However, the "documents" analyzed in a P2P setting are not the collaborative works seen in Wikipedia, but are instead each their own entity and anomalies are almost entirely malicious in nature.

Geiger and Ribes discuss the collaboration between vandalism detection systems and human actors in the process of banning a vandal from Wikipedia, as well as the prevalence of bots in the edit process [5]. The IllEdit approach can be viewed as playing the role of a vandalism detection bot in such a system, though it does not focus simply on vandalism.

Page et. al. present PageRank [14], a link ranking algorithm that assigns weights to hyperlinked documents based on relevance. The IllEdit knowledge influence matrix algorithm is influenced by the PageRank algorithm, propagating influence through a graph of related nodes, but institutes harsh teleportation and dampening factors to keep knowledge localized as opposed to letting the system reach equilibrium.

3. APPROACH

Our approach begins with the collection of a full edit history snapshot of Simple English Wikipedia. We parse the raw XML into a set of relevant classification features and store the data in a MySQL relational database. As compared to the XML, MySQL allows for the quick creation of training and test sets and also allows us to more easily perform ground truth reversion detection by providing a manner of obtaining page-at-a-time edit sequence data in chronological order. From the full edit sequence, we build training and test sets using randomly selected sample data. Using the training set and a Support Vector Machine we build a 2-class model of reverted and non-reverted edits. This model is then used to classify the instances in our test data set.

3.1 Dataset

Simple English Wikipedia, like most alternate language WikiMedia offerings, exists in a similar structure to English Wikipedia. Information is compiled into articles and pages. Any user, registered or anonymous, can create, delete, or edit content. Users can be voted into positions of authority by a community review process. Recognition can be awarded or revoked from pages through a similar process. The wiki exists as a subset of English Wikipedia articles written with simple vocabulary and grammar, but with an emphasis on keeping a level of information quality consistent with Wikipedia as a whole. Simple English Wikipedia's substantially smaller size provides a data set that allows us to focus on the machine learning aspect of reversion detection instead of that of big data.

The Feburary 2012 snapshot of Simple English Wikipedia used for this experiment is comprised of approximately 3.1 million edits spanning approximately 240,000 pages made by approximately 175,000 users and is the full edit sequence history of Simple English Wikipedia up to that point in time.

WikiMedia provides full, XML-formatted archives of the full edit sequence history of Simple English Wikipedia at given snapshots. This includes the histories of all articles and most non-article pages. Article revisions contain a timestamp, unique identifier, editing user, and snapshot text. Even given its reduced size when compared to English Wikipedia, the data is overly bulky for use in metadata experiments when article content is ignored. The pertinent features, as described below, were extracted from the archive into a relational database system from which experimental data sets could be more easily compiled.

To determine a ground truth for article reversions, we first hashed the content of each article at each snapshot using the Murmur2 64-bit hashing algorithm ([2]) and stored the hashed value and content length for each page edit. Two edits with the same hash value and content length can be considered equivalent and, given a time-ordered list of edits to a page, any edit state between two equivalent edits can be considered reverted and removed from the current product. The ground truth reversion state allows the IllEdit system to determine its correctness and utility in the detection of reverted edits.

3.2 Features and Extraction

Table 1 provides an overview of the feature set for the IllEdit classifier, which is comprised entirely of edit metadata, rather than the content of edits themselves. They can

Table 1: Illiterate Editor Feature Set

User Features	**type**
is_administrator	boolean
is_bureaucrat	boolean
is_bot	boolean
edit_count	numeric
reverted_count	numeric
revert_percentage	numeric
knowledge_score	numeric
Page Features	
is_good	boolean
is_verygood	boolean
Edit Sequence Features	
delta_time	numeric
content_length	numeric
delta_length	numeric

be viewed as being part of three groups of features:

3.2.1 User Features

Information regarding individual users and their community presence.

User titles: The Wikipedia community structure defines two levels of authority for users. Administrators have the ability to block and unblock users, and protect, delete, undelete, and rename pages. Bureaucrats have the same abilities, plus the ability to promote and demote administrators. Both are elected positions and require not only nomination from the community, but also an extensive questioning and discussion process during which fellow editors consider the nomination. Given enough support, users can be promoted indefinitely into one of these positions. With a title may come inherent trust as the community has already made a conscious decision to recognize a user.

Bots: The community also allows for approved bots to make automatic changes to Wikipedia articles and pages. Many times, these bots serve to correct grammar and spelling mistakes, provide links to sources, and add additional language pages. All bots must be approved for use on each wiki and, as such, may be afforded some inherent trust.

Edit counts: A user's edit counts speak to his/her prior contributions to the Wikipedia community.

Knowledge score: A user's estimated knowledge on the topics covered in the current edit, either positive or negative. Our algorithm calculates this knowledge score for each edit. Relevant topics are determined using the the Simple English Wikipedia category hierarchy graph and the knowledge score algorithm, described in more detail in Section 3.3.

3.2.2 Page Features

Information regarding articles, pages, and page awards.

Page awards: Page awards are granted and revoked, much like user titles, with a formal review process. Articles may be nominated for status, normally by those who have made significant contributions to the article and are familiar with the subject matter. The review process assumes that pages that have been nominated have been held to a higher standard, so the process is more focused on addressing reviews critical of the nomination as opposed to supportive. The final decision is made when the Feature Article director or one of the Feature Article delegates decides that enough of a consensus has been reached during the review process to either confirm or deny a candidate. Pages that are marked "good" or "featured" may have a higher level of trusted content and edits made to those pages may be held to a higher standard. Such pages may also be more highly targeted by vandalism.

3.2.3 Edit Features

Information regarding individual edits and their relationship to other edits.

Delta time: Pages edited in quick succession may signify an edit war, but may also occur after an automatic change by a bot.

Content length: Tracking content length and delta length provides an easy check for an entire page being deleted or mostly deleted.

3.3 Knowledge Score

To better understand the contributions that users are making to Wikipedia, we must consider the motivation of each user to make edits. Those that contribute positively to discussion do so with an underlying knowledge of the topic at hand and, similarly, those that contribute negatively do so either with a misunderstanding of information or a willful intent. The first iteration of the Illiterate Editor considered all edits equally when recommending reverts and included no notion of topic or field. However, as we consider Wikipedia editors as contributors with information to share, we also consider that this information comes from knowledge editors have (or mistakenly believe they have) on given topics. For example, a user editing pages on combustion engines may have relevant knowledge on combustion or automobiles, but not on computing or artificial intelligence. As such, the Illiterate Editor was modified to consider spheres of knowledge as part of its detection algorithm.

We use a combination of the Wikipedia category hierarchy and user edit history to encapsulate this information. Each Wikipedia article belongs to a number of categories, usually between one and four. Further, each category, itself, belongs to one or two parent or related categories. By scraping category page information and compiling the links between categories, we've assembled a graph hierarchy, with each graph node representing a category and each graph edge representing a parent/child relationship between categories. We use the word "hierarchy" to describe the graph not because it is a strict tree structure, but because the parent/child relationship between nodes tends to follow from extremely general concepts, such as "science", down through increasingly specific topics.

Before considering individual edits, we must first determine how each category influences its neighbors. If we assume that knowledge in each category is relevant to only that category, this step is unnecessary. However, since we posit that knowledge in a given field will relate to knowledge in similar fields, we must determine that relationship.

This relationship takes the form of an *influence matrix*, an NxN matrix where value (i,j) is equal to category i's influence on category j. Each matrix row, i, is calculated by initially placing one point of influence in category i, with all others at zero, and propagating that influence amongst i's graph neighbors using the algorithm in Figure 1. On each iteration of the propagation algorithm, each node distributes half of its influence equally amongst its neighbors. After two

Algorithm: Influence Matrix Row Calculation

```
cur_val = {}
cur_val[i] = 1
for j in 0..2:
    next_val = {}
    for node in cur_val:
      nb = neighbors(node)
      pass_value = cur_val[node] * 0.5
      next_val[node] = cur_val[node] - pass_value
      for neighbor in nb:
        next_val[neighbor] += pass_value / nb.size()
    cur_val = next_val
```

Figure 1: Influence Matrix Row Calculation. Each row in the influence matrix corresponds to one category and is calculated by the above algorithm.

iterations of propagation, the influence value of each node, j, is added to the influence matrix at position (i,j). This continues until all rows have been calculated. The algorithm is similar to the PageRank algorithm [14] using a teleportation probability of 0.5, but employs a significantly higher level of dampening. While PageRank continues to propagate until an equilibrium has been reached, the Illiterate Editor influence algorithm stops after two iterations. Due to its hierarchical nature and relatively small size, performing too many iterations of propagation can lead to tangetially-related categories sharing too much influence.

Once an influence matrix has been built, it can be used to calculate the *knowledge score* for each edit. The knowledge score is an unbounded, positive or negative floating point value that represents the editor's estimated knowledge on the topics of a given edit. The score is equal to the sum of the knowledge scores of all of the editing page's categories for the editing user. Each new user starts with a knowledge score of zero in every category. When a user makes an edit to a page, the knowledge score value of each of its categories is either incremented or decremented by $\frac{1}{numcategories}$ for a non-reverted or reverted edit, respectively. That knowledge is then distributed to neighboring categories as per the relevant rows in the influence matrix.

Consider the following examples of computing the influence matrix and knowledge scores on a simplified category graph:

Figure 2 shows a greatly simplified piece of the Wikipedia category graph surrounding the "human behavior" category. Initially, one point of influence is placed in the "human behavior" node. In Figure 3, the "human behavior" node takes $\frac{1}{2}$ of its 1 point of influence and distributes it evenly amongst its neighbors, $\frac{1}{6}$ point to "habits", $\frac{1}{6}$ point to "human skills", and $\frac{1}{6}$ point to "humans", the result of which is shown in Figure 4. Figure 5 shows the final iteration of the process. Each node distributes half of its influence equally amongst its neighbors. At the end of execution, as seen in Figure 6, the sparse row for "human behavior" would be "human behavior": { "human behavior": $\frac{11}{24}$, "habits": $\frac{1}{6}$, "human skills": $\frac{1}{6}$, "humans": $\frac{1}{6}$, "centenarians": $\frac{1}{24}$}. Thus, for each successful point of knowledge placed in "human behavior", $\frac{11}{24}$ of that point stays in the category, $\frac{1}{6}$ of that point is knowledge in "habits", $\frac{1}{6}$ is in "human skills", and so on. The knowledge score algorithm does not discriminate between parent and child categories, as the flow of knowledge could just as easily pass in each direction.

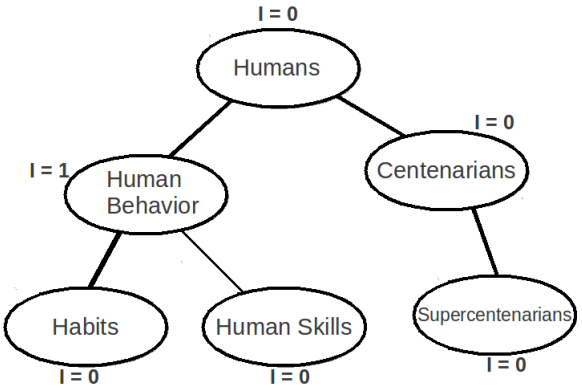

Figure 2: Initial influence is added to the human behavior node. When calculating the influence matrix for each node row, one point of influence is initially placed in the node to distribute to neighbors. Each graph node represents a Wikipedia article category and each edge represents a parent/child link between categories.

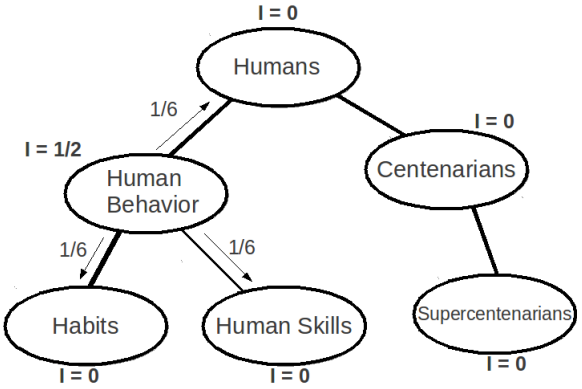

Figure 3: Half of the influence of the human behavior node is distributed equally amongst its neighbors. This represents the proposed similarity that the topic of human behavior has with its nearest neighbor topics in the category graph.

Next we compute the knowledge scores for a single user. The "Computer Aggression" page on Simple English Wikipedia belongs to 2 categories, "human behavior" and "violence". Consider Tom, a new user with no prior edits and a knowledge score of zero for every category. If Tom makes a successful, non-reverted, edit to the "Computer Aggression" article, he will receive 1 point of knowledge, divided equally between "human behavior" and "violence". The $\frac{1}{2}$ point of knowledge given to "human behavior" will, given our influence matrix row for that category, be distributed as $\frac{11}{24} * \frac{1}{2}$ point to "human behavior", $\frac{1}{6} * \frac{1}{2}$ point to "habits", $\frac{1}{6} * \frac{1}{2}$ point to "human skills", $\frac{1}{6} * \frac{1}{2}$ point to "humans", and $\frac{1}{24} * \frac{1}{2}$ point to "centenarians". The other $\frac{1}{2}$ point will be distributed to "violence" and its neighbors according to its own influence matrix row. Tom, then, makes an edit to the "Procrastination" page, which belongs to the "human behavior" and

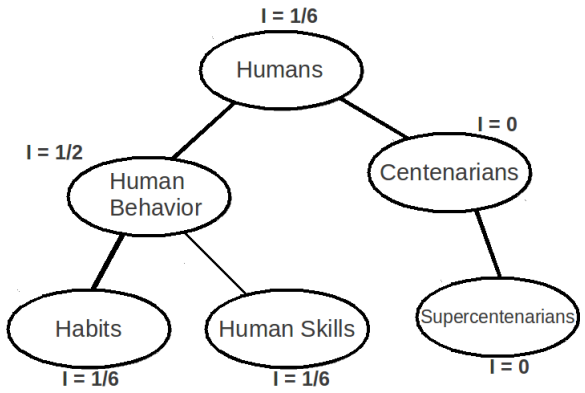

Figure 4: Human behavior influence graph after one iteration. All of the topics immediately neighboring the human behavior topic have obtained a portion of its influence.

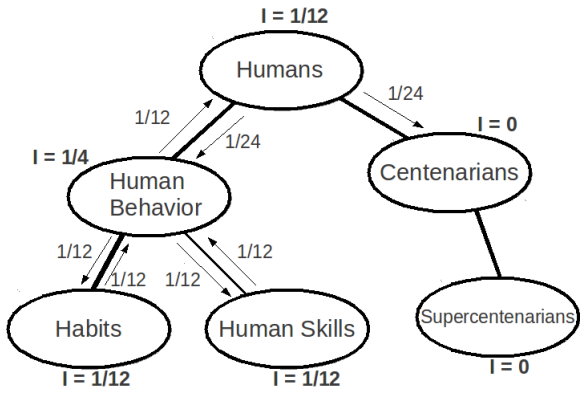

Figure 5: Each node distributes half of its influence equally amongst its neighbors, representing similarity between each node topic and their nearest neighbor topics.

"habits" categories. Due to his prior, non-reverted, edit to the "Computer Aggression" article, he has a knowledge score for this article equal to the sum of the knowledge scores for each article category, or $\frac{11}{48} + \frac{1}{12} = \frac{15}{48}$. As Tom makes additional edits to other pages, his knowledge scores on various topics will rise on successful edits and fall on reverted ones.

3.4 Classification Approach

From the full edit history of Simple English Wikipedia, we compiled a random sampling of approximately 750,000 edits for classifier training and 75,000 edits for testing.

Using the Weka machine learning suite, we built a support vector machine classifier using sequential minimization optimization [11] [15]. The classifier is trained using the aforementioned feature set along with a ground truth reversion state, either reverted or non-reverted, for each edit. Algorithm success is a metric of the classifier placing test set edits into one of the two classes successfully. Cross-validation of the training set showed best results using a polynomial kernel of degree 1.

4. EVALUATION

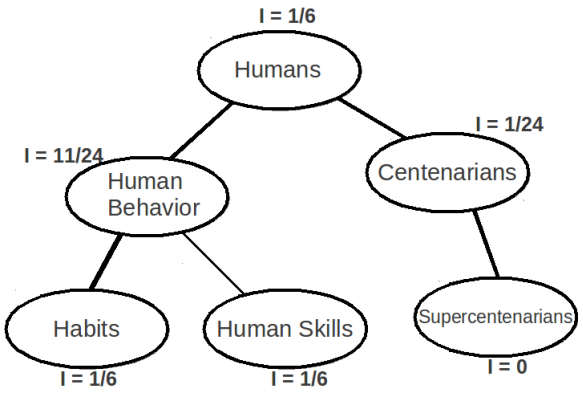

Figure 6: Final influence graph for human behavior. Each node's influence represents the similarity of that node's category to the initial node, human behavior. When a user makes an edit to a page in the human behavior category, all of the nodes above with influence will be affected.

4.1 IllEdit Revert Classification

Table 2 shows an overview of the classification results. Our support vector machine with SMO classification resulted in an overall 96.8% rate of correct classification. Approximately 8100 of the total 75,000 edits were reversions, either malicious or non-malicious, for a 9.2% revert rate. The confusion matrix in Table 4 breaks the results down further. For reverted edits, the classifier correctly identified 6492 for a 79.8% true positive rate and missed 738 for a 20.2% false negative rate. For non-reverted edits, the classifier correctly identified 66139 for a 98.9% true negative rate and missed 1644 for a 1.1% false positive rate.

Due to the strong skew in the percentages of reverted and non-reverted edits, the accuracy measure is not particularly valuable in determining system success. Given a data set with the average 8.8% reverted edit percentage, even classifying all edits as non-reverted would result in a 91.2% accuracy. Even this data set, with its 9.2% revert rate would yield an 90.8% accuracy in such a naive classifier. Table 3 provides a detailed look into the classification results of the system. The precision and recall values calculated show that the system, while successful overall, can much more easily classify non-reverted edits than reverted edits. It is important to keep the false positive rate as low as possible so as to not discourage users from contributing. The F-measure of 0.844 shows the algorithm is generally able to classify correctly, but could be improved.

More interesting is the Matthews correlation coefficient of the classifier, which is more useful in binary classification experiments as, unlike the F-measure, it doesn't ignore the false negative component of the confusion matrix. The Matthews coefficient, calculated as

$$\frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

, is defined as a value between -1 and 1, where -1 indicates complete disagreement, 0 indicates an equivalence with random prediction, and 1 indicates a perfect classification. The IllEdit algorithm's 0.829 Matthews correlation coefficient shows its substantial benefit over random prediction.

Table 3: Detailed Classification Results

Class	Precision	Recall	F-Measure	ROC Area
Reverted	0.897	0.798	0.844	0.893
Non-Reverted	0.975	0.989	0.982	0.893

4.2 Categories of Common Misclassifications

After classification, we evaluated the list of misclassified edits manually and observed the emergence of possible patterns of misclassification. The misclassified non-reverted edits (non-reverted edits classified by IllEdit as reverted) fall mainly into a few categories:

The Newbie: Those with a zero, or near zero, knowledge score and high percentage of reverted edits, but a low number of edits overall. These are the users that are new to the community (or are at least recently registered) and may have made poor edits their first few times. This is even more reasonable in a community designed for those learning a language. These users have learned from their mistakes and are making positive contributions to the community. They will eventually gain more trust. The system may gain more utility from explicitly flagging this type of instance and handling it differently. Overall, the community is harmed when new users are punished or discouraged [7].

The IP Collision: Similar features to new users, those unregistered users who happen to be contributing from an IP address that has made poor prior edits.

Bad Ground Truth: The IllEdit ground truth reversion detection algorithm uses hashes of page content to detect when reversions have occurred. This will not catch partial reversions and will not mark them as such. Though rare, some of the misclassified non-reverted edits are from users with extremely low knowledge score values and extremely high revert percentages and edit counts. Though it is possible these are benign edits and are representing a rare positive contribution by an otherwise negative user, it is more likely that these edits are actually malicious and their classification by the IllEdit system may be accurate.

The Adventurer: Users with close to zero knowledge scores, fairly high edit counts, and are in the medium range of revert percentages. These are users that have made positive contributions to the community at large, but not in the sphere of influence of the article. They have a high enough percentage of reverted edits that demand caution, but can overcome the misclassification as they make good edits within the new topics and their knowledge scores rise.

The misclassified reverted edits (reverted edits classified by IllEdit as non-reverted) also fall into some patterns:

The Rogue Bot: Given the requirements to become an approved Wikipedia bot, the IllEdit system values their input very highly. Coding isn't a perfect art and sometimes these bots make mistakes that are reverted. Most of the time these errors are fixed and the bots continue with their approved function. Some malicious bots are also able to become approved and cause damage before they are spotted and removed.

The Mistake: Users with high knowledge scores and low revert rates or near-zero knowledge scores and extremely low revert rates. Possibly users with administrator or bureaucrat status. This is the rare miss. A generally positive user makes a bad edit. Possibly a typo or duplicate information.

The Edit War: Sometimes good edits are reverted by bad users as part of an argument.

The Edge Case: Though the exact algorithm lies within the support vector machine, there are some misclassified reverts that, to a human observer, seem as though they are possibly just outside the threshold for the IllEdit system to flag. These are edits from users with medium range (20.30%) revert rates and zero to slightly negative knowledge scores. With additional training data, these edits may be correctly classified in the future.

Table 2: Classification Results Summary

Correctly Classified Instances	96.8%
Incorrectly Classified Instances	3.2%
Mean Absolute Error	0.038
Root Mean Squared Error	0.178

Table 4: Confusion Matrix

	Reverted	Non-Reverted
Reverted	0.798	0.202
Non-Reverted	0.011	0.989

4.3 Feature Analysis

To better understand the contributions of our feature set to the final classification decision, we calculated the information gains of our features across multiple training sets of data. Results showed that the contribution of the knowledge score feature increased as the percentage of edits in the training set with non-zero knowledge scores increased. When 100% of the training set edits have non-zero knowledge score values, the feature has an information gain of 0.446, but this number drops to 0.097 when only 60% of edit knowledge scores are non-zero. Other large contributions come, expectedly, from the user's revert percentage, with a higher percentage of reverted edits recommending a future reverted edit, and from the user's status as an approved bot. Bots on Wikipedia tend to make a large number of benign edits of the same type and, having gone through the approval process, are rarely malicious. The contributions of these two features also vary depending on the percentage of non-zero knowledge scores in the training set. As the knowledge score feature's contributions to the decision process are decreased, the revert percentage and bot status features pick up some of the slack.

4.4 Performance

Though the classifier model training time can be substantial, on the order of approximately 10 minutes (5000 instances) to 3 days (750,000 instances) on an Intel Core i72670QM with 8 GB of RAM, the lack of content analysis makes classification of testing data extremely fast. Individual instances are classified on the order of microseconds, with most decisions being made in under 10 microseconds.

4.5 IllEdit and ClueBot

The IllEdit and ClueBot systems are designed as vastly different approaches to a similar goal. Both hope to enhance the Wikipedia experience by limiting the amount of detrimental contributions made to articles and minimizing the effect of bad information on the community. IllEdit's Support Vector Machine classification provides a quick decision on a current edit based on a user's edit history and community status. ClueBot's content analysis approach employs artificial neural networks and Bayesian filtering to detect edit vandalism. In an ideal world, these two vectors of attack could be used in parallel to detect an even greater amount of potential misinformation.

Using a separate dataset of 150,000 edits, we compared the abilities of IllEdit and ClueBot to determine how the results of the detection algorithms overlap. The datasets were split into 90% training and 10% test sets.

Over a 15,000 edit test set with 1401 reverted edits, the IllEdit system correctly identified 594 reverts and ClueBot correctly identified 27. These low true positive rates can be attributed to the relatively small training sets used for experimenation. Further, to ensure the equivalence of the data sets used for each system, edits were marked as reverted as according to the IllEdit system algorithm, rather than a community vandalism consensus. As such, many edits marked as reverted/vandalism for the purposes of this test may not have been vandalism, but simply misinformation. It is important to note, then, that the purpose of this experiment is not to show superiority of the IllEdit system, but to explore the results of the classification algorithms on the same data set.

The training times of each system were comparable, with the IllEdit SVM model building in 38 minutes and the ClueBot model building in 19 minutes. The IllEdit algorithm decision making, however, is orders of magnitude faster than the ClueBot algorithm, making decisions on the scale of microseconds per instance versus ClueBot's milliseconds per instance.

Of the correctly classified reverted edits, the ClueBot algorithm successfully identified 17 that were misclassified as non-reverted by IllEdit. Specifically, the ClueBot algorithm was adept at detecting large-scale content deletion by new users. This shows a potential for collaboration between algorithms to increase the rate of detection.

5. LIMITATION OF GENERALIZABILITY AND SPECIFICITY

Our choice of Simple English Wikipedia as a dataset, though it provides a rate of article reversion consistent with English Wikipedia, may not provide the same community interaction model. For example, even a consistent number of reversions may include more unintentional misninformation rather than edit wars, especially in a system designed for those just learning a language. However, as these users learn, their edits may trend from negative to positive as opposed to English Wikipedia where many users are simply apt to create trouble. The IllEdit system is designed to interpret a user's previous contributions to the community and, given a community with a higher disparity between good and bad users, may even perform better than on its current dataset.

Unlike the work of [9] [8] [4] [1], the IllEdit system does not perform any content analysis and therefore only keeps track of full edit reversions instead of assigning some merit for partial contribution. This could serve to inflate user knowledge scores for users who make edits containing both positive and negative contributions. At the same time, this presents a system where only users that submit entirely negative edits are punished and those that provide at least some useful content are rewarded.

6. DISCUSSION AND FUTURE WORK

Our feature set, while growing, is currently a mix of both Wikipedia-specific and Wikipedia-agnostic features. Rethinking the concepts behind this data set may allow the IllEdit system to be adapted to other collaborative document editing systems.

The knowledge score algorithm that we have proposed in this work has shown itself to be the most relevant feature in our feature set, however, there is more possible research into its potential effectiveness. The algorithm has multiple parameters that can be investigated and it may be possible to learn the ideal settings for these parameters to improve results.

The jump from Simple English Wikipedia to English Wikipedia is a high priority goal in confirming the utility of of the IllEdit algorithm. While our current data set provides insight into how such a system may perform, the addition of any new experimental dataset will only help to assess the algorithm further. Though a subset of English Wikipedia could be used initially, the innerworkings of the knowledge score algorithm demand a full edit history for a group of both pages and users. Any Wikipedia data subset would be required to include this full, overlapping history.

Though our feature set includes the notion of the administrator and bureaucrat titles, it does not take into account user awards as it does page awards. User awards, or "barnstars", may be presented from any user to any user for any reason, at any time, and are usually given as commendation for some action. Many barnstars varieties exist, from meticulous editing, to translation, to providing art resources and citations. More can even be created for any reason. These features aren't currently included because of their open nature. While some level of trust can possibly be gained from holding a title, the effect of holding barnstars on trust is less clear. McDonald et. al. have researched the ability to recognize observable behavior that may lead to barnstars, but the barnstars themselves don't equate to trust [13]. To ensure barnstars are attributed a fair value, a graph of user trust must be created and awarded barnstars must be valued based on the trust placed in the awarding user. Though this is a possiblity, it may require roots of trust, a la [12], and is thus left to future work.

An interesting potential feature involves the monitoring of talk pages for user activity. Talk pages have become used as a forum for the discussion of potential edits before they are made to articles themselves. It may be likely that users who have recently made edits to an article's talk page may be more likely to make positive edits to that article.

7. CONCLUSIONS

The experiments and results described above show the potential of a fully metadata-centric approach to Wikipedia revert detection. The IllEdit approach is unlike most current approaches, which may use some metadata features,

but rely heavily on expensive content analysis and community intervention. Given this extensive focus on metadata alone, the IllEdit system may identify different types of edit reversions than other detection algorithms, for example, innocuous misinformation as opposed to malicious vandalism. For this reason, it is important to view the IllEdit approach not at a single solution, but as one of many that can be used in conjunction to preserve the integrity of Wikipedia information.

8. REFERENCES

[1] B. Adler, L. de Alfaro, S. Mola-Velasco, P. Rosso, and A. West. Wikipedia vandalism detection: Combining natural language, metadata, and reputation features. In *Proceedings of the 12th International Conference on Intelligent Text Processing and Computational Linguistics*, pages 277–288, 2011.

[2] A. Appleby. Murmurhash. https://sites.google.com/site/murmurhash.

[3] C. Carter. Cluebot ng. en.wikipedia.org/wiki/User:ClueBot_NG.

[4] L. de Alfaro, A. Kulshreshtha, I. Pye, and B. Adler. Reputation systems for open collaboration. In *Communications of the ACM*, volume 54, 2011.

[5] R. Geiger and D. Ribes. The work of sustaining order in wikipedia: The banning of a vandal. In *Proceedings of the 2010 ACM conference on Computer Supported Cooperative Work*, pages 117–126, 2010.

[6] J. Giles. Internet encyclopedias go head to head. *Nature*, pages 900–901, 2005.

[7] A. Halfaker, A. Kittur, and J. Riedl. Don't bite the newbies: how reverts affect the quantity and quality of wikipeida work. In *International Symposium on Wikis and Open Collaboration*, 2011.

[8] S. Javanmardi. Modeling trust in collaborative information systems. In *Proceedings of the 3rd International Conference on Collaborative computing, Networking, Applications and Worksharing*, 2007.

[9] S. Javanmardi, Y. Ganjisaffer, C. Lopes, and P. Baldi. User contribution and trust in wikipedia. In *Proceedings of the 5th International Conference on Collaborative computing: Networking, Applications and Worksharing*, 2009.

[10] S. Kamvar, M. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *Proceedings of the Twelfth International World Wide Web Conference*, 2003.

[11] S. Keerthi, S. Shevade, C. Bhattacharyya, and K. Murthy. Improvements to platt's smo algorithm for smv classifier design. *Neural Computation*, 13(3):637–649, 2001.

[12] R. Levien. *Attack Resistant Trust Metrics*. PhD thesis, UC Berkeley, 2004. Draft Only.

[13] D. McDonald, S. Javanmardi, and M. Zachry. Finding patterns in behavioral observations by automatically labeling forms of wikiwork in barnstars. In *International Symposium on Wikis and Open Collaboration*, 2011.

[14] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report SIDL-WP-1999-0120, Stanford InfoLab, 1999.

[15] J. C. Platt. Sequential minimal optimization: A fast algorithm for training support vector machines. Technical Report MSR-TR-98-14, Microsoft Research, April 1998.

[16] B. Suh, E. H. Chi, B. A. Pendleton, and A. Kittur. Us vs. them: Understanding social dynamics in wikipedia with revert graph visualizations. In *IEEE Symposium on Visual Anlytics Science and Technology*, 2007.