5-1-2017

Editing Behavior Analysis and Prediction of Active/ Inactive Users in Wikipedia

Harish Arelli
Boise State University

EDITING BEHAVIOR ANALYSIS AND PREDICTION OF ACTIVE/INACTIVE USERS IN WIKIPEDIA

by

Harish Arelli

A Project

submitted in partial fulfillment

of the requirements for the degree of

Master of Science in Computer Science

Boise State University

May 2017

BOISE STATE UNIVERSITY GRADUATE COLLEGE

DEFENSE COMMITTEE AND FINAL READING APPROVALS

of the project submitted by

Harish Arelli

Project Title: Editing Behavior Analysis and Predication of Active/Inactive Users in Wikipedia

Date of Final Oral Examination: 24 April 2017

The following individuals read and discussed the project submitted by student Harish Arelli, and they evaluated his presentation and response to questions during the final oral examination. They found that the student passed the final oral examination.

Francesca Spezzano, Ph.D.	Chair, Supervisory Committee
Edoardo Serra, Ph.D.	Member, Supervisory Committee
Dianxiang Xu, Ph.D.	Member, Supervisory Committee

The final reading approval of the project was granted by Francesca Spezzano, Ph.D., Chair of the Supervisory Committee.

For my family

ACKNOWLEDGEMENTS

I would like to thank my advisor, Dr. Francesca Spezzano, for the dedication and encouragement she gave during my graduate studies at Boise State University. I would also like to thank my committee members, Dr. Edoardo Serra and Dr. Dianxiang Xu for the guidance they provided. And finally, I would like to thank my family and friends for their encouragement, love and support; I could not have done this without them.

ABSTRACT

In this project, we focus on English Wikipedia, one of the main user-contributed content systems, and study the problem of predicting what users will become inactive and stop contributing to the encyclopedia. We propose a predictive model leveraging frequent patterns appearing in user's editing behavior as features to predict active vs. inactive Wikipedia users. Our experiments show that our method can effectively predict inactive users with an AUROC of 0.97 and significantly beats competitors in the task of early prediction of inactive users. Moreover, we study differences in editing behavior of inactive vs. active users to explain why some users are leaving and provide some rules explaining our predictive model.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CAPTER ONE: INTRODUCTION

Overview

Nowadays, a huge part of the information present on the Web is delivered through user-contributed content (UCC) systems, such as Yahoo! Answers, Wikipedia, YouTube, Flickr, Slashdot.org, Stack Overflow, Amazon product reviews, and many more. Here, many users create, manipulate, and consume content every day. For example, English Wikipedia contains over 5 million articles that have been written collaboratively by volunteers around the world, and almost all of its articles can be edited by anyone who can access the Wikipedia website. About 300K editors, from expert scholars to casual readers, edit Wikipedia every month. However, just a small part of them keep actively contributing. For instance, in 2016, over 3,000 new editors had made more than 100 edits every month [6].

Many studies have examined this user-contributed content phenomenon and, in particular, the reasons that motivate users to become contributors, to continue contributing, and to increase contribution [1], [2], [3]. However, many users stop contributing after a certain period of time [4], [5]. The exit of active contributors from a particular UCC community may affect quantity and quality of content provision not only on the specific community, but also on the Web in general.

Other works have studied Wikipedia users' editing behavior to check how long they will keep active [7], [8], what their roles are [9], [10], why they contribute [11], [12], [13], and to identify malicious users [14], [15].

In 2011, Wikimedia Foundation, Kaggle, and IEEE ICDM organized a research competition [16] called Wikipedia Par- ticipation Challenge (WikiChallenge) where participants were asked to build a predictive model that accurately predicts the number of edits a Wikipedia editor will make in the next months based on his or her edit history so far.

However, very few studies attempted to uncover the reasons why many contributors become inactive. Jian and MacKie-Mason [4] formulated some hypotheses about the problem but did not validate them with any experiment, while Asadi et al. [5] performed a study on Persian Wikipedia to understand motivations and discouraging factors towards contribution on a very small case study (15 users). Thus, the problem of understanding the reasons why Wikipedia editors become inactive is still an open problem. Most importantly, be able to early predict which user will become inactive is very valuable for the community in order to perform engaging actions on time to keep these users contributing longer.

In this project, we focus on English Wikipedia, one of the main UCC systems, and study the editing behavior of active and inactive editors on a large scale to (1) predict what users will become inactive and stop contributing to Wikipedia and (2) explain the reasons behind the quitting of so many users. Our contributions in the project are the following.

- We propose a machine learning based model that leverages frequent patterns appearing in user's editing behavior as features to predict active vs. inactive Wikipedia users.

- We experimentally show that our model reaches an excellent Area Under the ROC curve of 0.97 and a precision of 0.99 in predicting editors who will become inactive. Moreover, we show that the proposed model is able to early predict inactive editors much more effectively than competitors. For instance, by looking at the first 3 edits we can predict inactive users with an AUROC of 0.72 vs. 0.55 for the competitor [7]. We think that the early prediction of inactive users is useful for Wikipedia administrators or other users to perform recovering actions on time to avoid the loss of contributors.
- We further investigate differences in the editing behavior of active vs. inactive users according to users' involvement in edit wars, reverted edits, meta-pages editing, and categories edited. We also show some rules we extracted that explain our proposed predictive model.

Structure

This project documentation is organized as follows. Chapter 2 discusses related work. Chapter 3 introduces the dataset we used in the project. Chapter 4 presents methods we use to solve this project. Chapter 5 describes our behavior-based approach for predicting inactive users. Chapter 6 reports on our experiments and compares our approach with the state of the art. Chapter 7 studies differences in editing behavior of inactive vs. active users to explain why some users are leaving. Finally, conclusions are drawn in Chapter 8.

CHAPTER TWO: RELATED WORK

Jian and MacKie-Mason [4] discuss their hypothesis about why some editors stop contributing to Wikipedia. They considered editor roles such as creator, preserver, and destroyer, and variables like proportion of creations, proportion of re- versions, and proportion of damages as possible features that correlate with the leaving behavior. Based on two variables, namely Ontime (number of minutes that an edit persists and Deled (number of times an edit gets deleted), they hypothesize that the probability of leaving decreases according to the variation of Ontime (between the last week of edits and all other weeks), and, symmetrically, increases according to the variation of Deled. They also hypothesize that the higher the editor work intensity, the more likely they will leave. Based on the article stability, they hypothesize that the more stable the articles that an editor cares about, the more likely this editor will stop contributing.

Asadi et al. [5] addressed a research about discovering motivations for writing and editing in Persian Wikipedia, discouraging factors towards contribution, and reasons for contributing or giving up contributing in Wikipedia. They concluded that to understand whether an editor is active or not, it is necessary to know how often they edit and how many edits they make as well as how recent their last contribution is. After they interviewed 15 Persian Wikipedia active editors, they found the following answers. They said that personal motivations such as knowledge and experience sharing, receiving help from other users, and becoming more familiar with the structure of Wikipedia are also important motivations for continuing to contribute. In addition, they mentioned that

cognitive motivations and personal satisfaction are important to maintain ongoing participation in Persian Wikipedia. Other encouraging factors they found in their study are enriching Persian web content, starting new topics and content production, as well as competition with Wikipedia in other languages. They also found that personal beliefs and concerns may be a motivation to start writing and editing, but it is also more likely to lead to edit wars and, as a result, frustration and discontinuation. The reasons they individuated for not continuing in Wikipedia are: (1) lack of time to contribute to Wikipedia, (2) finding other web-based entertainment, (3) being impatient and lacking tolerance for criticism. Note that this is only a case study on a small group of 15 members of the community.

Lai and Yang [11] investigated the underlying reasons that drive individuals to edit Wikipedia content. They considered Wikipedia as a platform that allows individuals to show their expertise. Based on expectation-confirmation theory and expectancy-value theory for achievement motivations, they proposed an integrated model that incorporates psychological and contextual perspectives. They picked English-language Wikipedia for their survey. Analytical results, they indicated, confirmed that subjective task value, commitment, and procedural justice were significant to satisfaction of Wikipedia users, and satisfaction significantly influenced continuance intention to edit Wikipedia content. This work discusses individuals' interest in continuing edits of Wikipedia content, which is quite opposite to our problem.

Takashi et al. [9] analyzed the editing patterns of Wikipedia contributors using dynamic social network analysis. They have developed a tool that converts the edit flow among contributors into a temporal social network. They used this approach to identify

the most creative Wikipedia editors among the few thousand contributors who make most of the edits among the millions of active Wikipedia editors. In particular, they identify the key category of “coolfarmers”, the prolific authors starting and building new articles of high quality. As a second category of editors they look at the “egoboosters”, i.e. people who use Wikipedia mostly to showcase themselves. They said that understanding these different patterns of behavior gives important insights about the cultural norms of online creators.

Suin et al. [12], studied multilingualism by collecting and analyzing a large dataset of the content written by multilingual editors of the English, German, and Spanish editions of Wikipedia. This dataset contains over two million paragraphs edited by over 15,000 multilingual users from July 8 to August 9, 2013. The authors analyzed these multilingual editors in terms of their engagement, interests, and language proficiency in their primary and non-primary (secondary) languages and found that the English edition of Wikipedia displays different dynamics from the Spanish and German editions. Users primarily editing the Spanish and German editions make more complex edits than users who edit these editions as a second language. In contrast, users editing the English edition as a second language make edits that are just as complex as the edits by users who primarily edit the English edition. In this way, English serves a special role in bringing together content written by multilinguals from many language editions. In addition, they found that multilinguals are less engaged and show lower levels of language proficiency in their second languages. They also examine the topical interests of multilingual editors and found that there is no significant difference between primary and non-primary editors in each language. The dataset they used for their study is also very small.

In 2011, Wikimedia Foundation, Kaggle, and IEEE ICDM organized a competition about developing a model to predict the number of edits an editor will make in the five months after the end date of the training dataset they provided (see the contest at [16]). The dataset, which was open for all contestants, was randomly sampled from English Wikipedia. The time period of this dataset was from January 2001 to August 2010. The team prognoZit, who won the first prize in the WikiChallenge contest, developed their own algorithm to solve the problem. They used 13 features to predict the future editing activity: number of edits in 9 different periods, number of reverted edits in 2 different time periods, and number of deltas in another 2 different time slots. Their Wikipedia page is available at [7]. Another team, zeditor, won third place in the contest [8]. They solved this problem by using features such as number of edits, number of edited articles, and the length of time between first edit and last edit in 10 different exponentially long time intervals with Gradient Boosted Trees as classifier. Their Wikipedia page is available at [17].

CHAPTER THREE: DATASET

To conduct our study, we used the UMDWikipedia dataset (available at [18]) that consists of edits made by both benign and vandal users. We considered benign users only. This dataset contains a list of 16K randomly selected benign users who registered between January 01, 2013 and July 31, 2014. For each user, their edit history is available for the given time period (up to 500 edits per users), for a total of 609K edits made by benign users. For each edit the available information includes author's username, edit ID, edit timestamp, page title, page type (Wikipedia article or meta-page), page category, and if the edit was reverted and when. A meta-page is a page which is not a regular article, but it can be, for instance, a User page (where editors describe themselves) or an article Talk page (where editors discuss about the content of the associated Wikipedia article). The information about edit reversion is extracted by the edit reversion dataset provided by [19] which marks an edit as "reverted" if it has been reverted within the next 15 edits on the page.

The UMDWikipedia dataset also provides a User Log Dataset that consists, for each user u, of the chronological sequence of each consecutive pair (p_1, p_2) of pages edited by u. For each pair (p_1, p_2), a description of the pair is provided by using the following features.

- **r/n**: Whether p_2 is a page that has already been edited by the user before (p_2 is a re-edit – **r**), or p_2 is a page edited for the first time by user u (p_2 is a new edit – **n**).

- **m/n**: Whether p_2 is a meta-page (**m**) or a normal page (**n**).
- If p_2 is a re-edit:
    - **c/n:** Whether p_1 is equal to p_2, i.e. these are two consecutive edits (**c**) on the same page or not (**n**).
    - **r/n:** Whether a previous edit of p_2 by the user u has been reverted (**r**) by any other Wikipedia user or not (**n**).
- Otherwise (p_2 is a new edit):
    - **t/m/u:** Hop distance between pages p_1 and p_2 in the Wikipedia hyperlink graph: at most 3 hops (**t**); more than 3 hops (**m**); or unknown distance (**u**).
    - **z/o/u:** Common categories between pages p_1 and p_2: zero categories in common (**z**), at least one category in common (**o**), or info unavailable (**u**).
- **v/f/s:** Time difference between the two edits: less than 3 minutes (very fast edit - **v**), less than 15 minutes (fast edit - **f**), more than 15 minutes (slow edit - **s**).

Given the benign users in the UMDWikipedia dataset, we divided them into active and inactive users by using the following rule: if a user does not make any edit for Γ months, then we considered this user as an inactive user, i.e. a user who performed some edits and then, at some point, stopped editing and left the community. If an inactive user started editing again after more than Γ months, then we considered this user as a new user. All other users who do not have a gap of more than Γ months in their edit history are considered active users.

In our experiments, we set Γ = 2 months, which corresponds to a total of 16,191 inactive and 305 active users. We also performed our experiments by setting Γ = 3 months, which gave us a total of 16, 170 inactive and 326 active users. The experimental

results that we got with $\Gamma = 2$ months are relatively comparable to the results we got with $\Gamma = 3$ months.

CHAPTER FOUR: METHODS

In this chapter we describe in detail the methods we use to solve the problem of predicting inactive users and giving reasons behind their leaving from the wiki community.

Frequent Pattern Mining

According to [19], frequent patterns are a set of items, subsequences, subgraphs, etc., that appear in a data set with frequency no less than a user-specified threshold. For instance, frequent itemset is, a set of items, such as bread and jam, that appear frequently together in a transaction data set. A subsequence, such as buying first a bike, then a bike locker, and then two lights, if it occurs frequently in a shopping history database, is a (frequent) sequential pattern. A substructure can refer to different structural forms, such as subtrees, sublattices, or subgraphs, which may be combined with itemsets or subsequences. If a substructure occurs frequently in a graph database, it is called a (frequent) structural pattern. Finding frequent patterns plays an important role in mining associations, correlations, causation, dependence and many other interesting relationships among data. Furthermore, it is helpful in data indexing, classification, clustering, and other data mining tasks as well. Thus, frequent pattern mining has become major data mining task and a focused topic in data mining research.

A variety of pattern mining methods existed for frequent pattern mining. For example, sequential pattern mining, periodic pattern mining, high-utility pattern mining, etc.,

Sequential Frequent Pattern Mining

As per Wikipedia definition [20], sequential pattern mining is a topic of data mining concerned with finding statistically relevant patterns between data examples where the values are delivered in a sequence. Sequential pattern mining, which discovers frequent subsequences as patterns in a sequence database, is an important data mining problem with broad applications.

The sequential pattern mining problem was first propsed by Agarwal and Srikant in [21] based on their study of customer purchase sequences, which was "Given a set of sequences, where each sequence consists of a list of elements and each element contains a set of items, and given a user-specified min_support threshold, sequential pattern mining is to find all frequent subsequences, i.e., the subsequences whose occurrence rate in the set of sequences is no less than min_support".

Let our running sequence database be D gave in Table 4.1 and min_support = 2. The set of items in the database is {p, q, r, s, t, u, v}.

Table 4.1. A Sequence Database

Sequence _id	Sequence
1	(p (pqr) (pr) s (ru))
2	((ps) r (qr) (pt))
3	((tu) (pq) (su) (rq))
4	(tv (pu) rqr)

A sequence (p (pqr) (pr) s (ru)) has five elements: (p), (pqr), (pr), (s) and (ru), where items p and r appear more than once respectively in different elements. It is a nine-

sequence since there are nine instances appearing in that sequence. Item p happens three times in this sequence, so it contributes 3 to the length of the sequence.

However, the whole sequence (p (pqr) (pr) s (ru)) contributes only one to the support of (p). Also, sequence (p (qr) su) is a subsequence of (p (pqr) (pr) s (ru)). Since both sequences 1 and 3 contain subsequence a = ((pq) r), a is a sequential pattern of length 3 (i.e., 3-pattern). From this example, Agarwal and Srikant who worked in this [21] say, one can see that sequential pattern mining problem can be stated as "given a sequence database and the min support threshold, sequential pattern mining is to find the complete set of sequential patterns in the database."

More formally, a sequence database is a set of sequences where each sequence is a list of itemsets [22]. An itemset is an unordered set of distinct items. A sequential pattern is a sequence. Suppose consider a sequence $S_M = P_1, P_2, \ldots P_k$, where $P_1, P_2, \ldots P_k$ are itemsets. Then this sequence S_M is said to occur in another sequence $S_N = Q_1, Q_2, \ldots Q_m$, where $Q_1, Q_2, \ldots Q_m$ are itemsets, if and only if there exist integers 1 <= i1 < i2... < ik <= m such that $P_1 \subseteq Q_{i1}, P_2 \subseteq Q_{i2}, \ldots P_k \subseteq Q_{ik}$ [23]. The support of a sequential pattern is defined as follows. The number of sequences where the pattern occurs divided by the total number of sequences in the database. If the support of a sequential pattern is no less than the min_sup parameter, which will be provided by the user, then that sequential pattern is a frequent sequential pattern.

Commonly used algorithms for this sequential frequent pattern mining include FreeSpan, PrefixSpan, GSP, SPADE, etc.,We picked PrefixSpan algorithm [24] to find out frequent patterns to help in our problem. We used the PrefixSpan implementation,

provided by the SPMF open-source frequent pattern mining Java library [25]. We used min_sup = 0.1 or 10% and maximum pattern length equal to 5.

Clustering Time Series Data

Clustering

As per the definition given by Liao and T. Warren in their work [26], clustering is to identify structure in an unlabeled data set by objectively organizing data into homogeneous groups where the within-group-object similarity is minimized and the between-group-object dissimilarity is maximized.

K-Means Clustering

The algorithm starts with k initial centroids. In practice, these centroids are randomly chosen instances from the dataset. These initial instances form the initial set of k clusters. Then, we assign each instance to one of these clusters based on its distance to the centroid of each cluster. The calculation of distances from instances to centroids depends on the choice of distance measure. Euclidean distance is the most widely used distance measure.

After distributing all instances to a cluster, the centroids, are computed again by taking the average (mean) of all instances within the clusters (therefore, the name k-means). This method of computation is repeated using the newly calculated centroids. Note that this method of procedure is repeated till convergence. The most basic criterion to decide convergence is to check whether centroids are no longer changing.

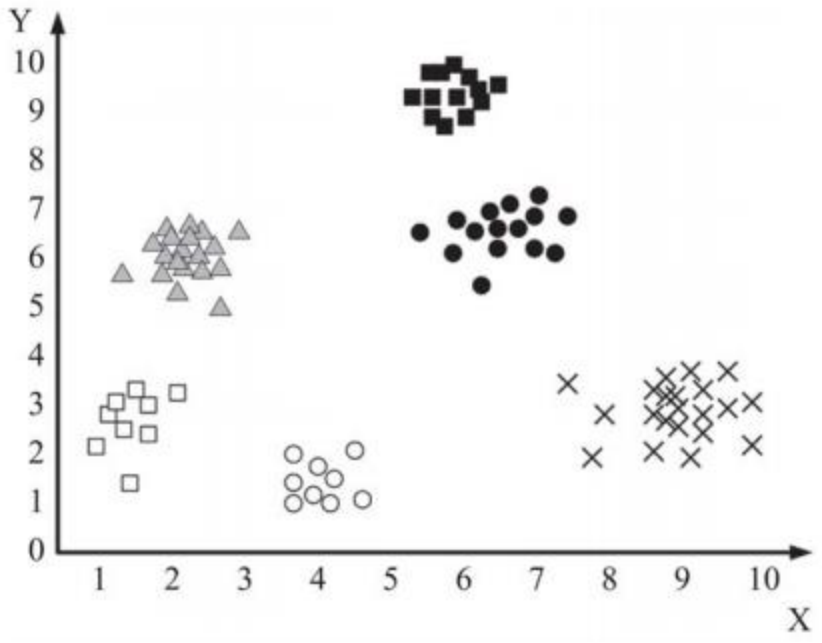


Figure 4.3 K-means Output on a sample dataset. K-means is run with k = 6, and the clusters found are visualized using different symbols.

The steps discussed above to apply K-means algorithm and the above image are form the textbook [27].

Time Series Clustering

To compute the clustering of time series, each value in a time series is normalized by computing its corresponding z-score, i.e. the number of standard deviations the number is away from the mean of all points series. Then, classical k-means algorithm is applied on the normalized time series. We used a number of clusters k = 5.

Features

We consider the following features extracted from our dataset for time series clustering: reverts percentage, meta-page percentage, unique meta-page percentage, edit-wars, and common categories. The description of these features is provided in Chapter 7.

Time Series

For each user and for each of the five features we extract from the dataset, we compute the value of feature over the time, where time is divided into equal time intervals (two weeks). Then we cluster series with common shape features together. This constitutes identifying common trends occurring at different times or similar sub patterns in the data. We perform time series clustering for active and inactive users separately and studied differences in their editing behavior.

Analysis

We did not notice significant differences between active and inactive users according to the temporal clustering of these features. However, we show the time series clustering obtained in Appendix C for completeness.

CHAPTER FIVE: PREDICTING INACTIVE USERS

We propose an editing behavior-based approach to predict which user will become inactive and leave the community. In order to find a set of features that differentiate the editing behavior of active vs. inactive users, we mined a set of features as follows [14].

First, we mined frequent patterns on the User Log Dataset for both active user logs and inactive logs by using the Prefix Span [28] algorithm. Each pattern represents a sub-sequence of a user's edit log and contains a sequence of pairs of pages consecutively edited by the user where each pair is described by using the features in Appendix.

Second, for each frequent pattern f we mined, we computed the frequency of f for both the classes of active and inactive users. We found patterns that appear in both classes of users, while other patterns are exclusive for active users. We did not find any pattern that appears for the class of inactive users only.

Third, we ordered frequent patterns by descending frequency absolute difference between the two classes. Then, we selected as set of features for classification the set of top k patterns of length l that appear for both active and inactive users and for active users only. We used $k = 13$ and $l \in \{1,2,3\}$. The result was a total of 78 features. The value of each feature is a Boolean value indicating whether or not that pattern appears in the edit history of the user.

It is worth noting that, in predicting inactive users, we did not consider the duration of a user's edit history, from the first edit to their most recent edit, as this feature

is biased towards inactive users who are short-lived because they stop editing Wikipedia. Moreover, our editing behavior-based features do not look at edited content and, therefore, our resulting system has the advantage of being general and applicable not only for English, but also for different other language versions of Wikipedia.

The complete list of 78 features is shown in Appendix A (Table A.1). In the following, we discuss the top 10 features that turned out to be the most important for the classification task.

Most Important Features

To compute the most important features, we used forests of 250 randomized trees. The relative importance (for the classification task) of a feature f in a set of features is

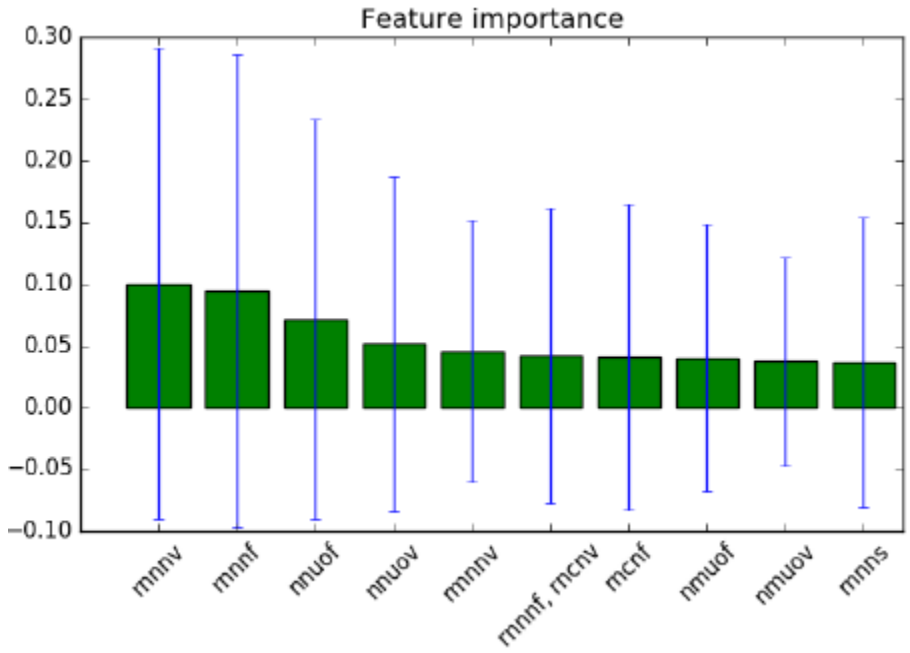


Figure 5.1 Top 10 most important features

given by the depth of f when it is used as a decision node in a tree. Features used at the top of the tree contribute to the final prediction decision of a larger fraction of the input samples. The expected fraction of the samples they contribute to can thus be used as an

estimate of the relative importance of the features. Figure 5.1 shows the importance of the top 10 features for the classification task. The green bars in the plot show the feature importance using the whole forest, while the blue bars represent the variability across the trees.

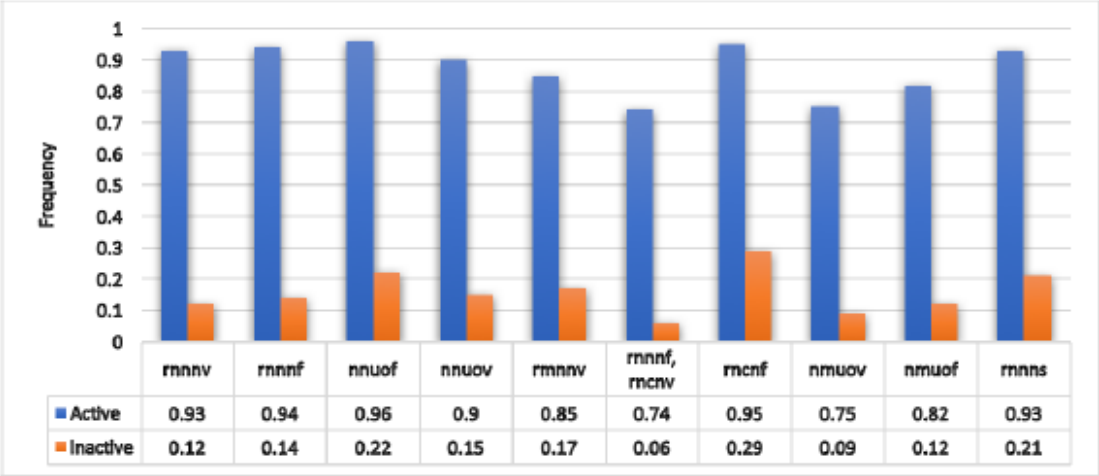


Figure 5.2 Frequency of top 10 most important features for active (blue) vs. inactive (red) users.

Figure 5.2 shows the frequency of the top 10 most important features for the class of active (blue) and inactive (red) users. There is a significant gap in the frequency of these patterns for the two different classes of users. All the patterns are highly frequent for active users and less frequent for inactive users. As we will see in Chapter 6, our pattern-based features extracted from users' editing behavior will allow us to differentiate between active and inactive users with an area under the ROC curve of 0.975.

The top 10 most important features are explained in detail here below.

rnnnv: there exists a pair of edits (p_1, p_2) s.t. p_2 is a is a re-edit of a non-meta page, non-consecutively ($p_1 \neq p_2$), not due to reversion, and very fast (p_2 is edited within

less than 3 mins from the edit on p_1). This pattern is frequent for 93% of active users vs. 12% of inactive users.

rnnnf: same as the feature above but the pages are edited within less than 15 mins (fast) one from the other. This pattern is frequent for 94% of active users vs. 14% of inactive users.

nnuof: there exits of a pair of edits (p_1, p_2) s.t. p_2 has never been edited before by the user u, p_2 is an article page (non-meta), a path doesn't exist between p_1 and p_2 in the hyperlink graph, the pages have at least one category in common, and p_2 is edited within less than 15 mins from the edit on p_1. This pattern is frequent for 96% of active users vs. 22% of inactive users.

nnuov: same as the feature above but the pages are edited within less than 3 mins (very fast), one from the other. This pattern is frequent for 90% of active users vs. 15% of inactive users.

rmnnv: this pattern means that the user is re-editing a meta-page, non-consecutively ($p_1 \neq p_2$), not due to reversion, and very fast. This pattern is frequent for 85% of active users vs. 17% of inactive users.

rnnnf, rncnv: there exists a pair of edits as in pattern 2 (rnnnf) followed by another pair of edits (p_1, p_2) s.t. p_2 is a is a re-edit of a non-meta page, consecutively ($p_1 = p_2$), not due to reversion, and the re-edit happens very fast, i.e. within 3 mins (rncnv). This pattern is frequent for 74% of active users vs. 6% of inactive users.

rncnf: this pattern means that the user is re-editing an article page, consecutively, not due to reversion, and the re-edit happens fast (within 15 mins). This pattern is frequent for 95% of active users vs. 29% of inactive users.

nmuov: there exists a pair of edits (p_1, p_2) s.t. p_2 has been never edited before by the user u, p_2 is a meta- page, a path does not exist between p_1 and p_2 in the hyperlink graph, the two pages have at least one category in common, and p_2 is edited within less than 3 mins from the edit on p_1. This pattern is frequent for 75% of active users vs. 9% of inactive users.

nmuof: same as the feature above but the pages are edited within less than 15 mins (fast), one from the other. This pattern is frequent for 82% of active users vs. 12% of inactive users.

rnnns: there exists a pair of edits (p_1, p_2) s.t. p_2 is a re-edit of a non-meta page, non-consecutively ($p_1 = p_2$), not due to reversion, and the edit on p_2 happens slowly with respect to the edit on p_1 (more than 15 mins after). This pattern is frequent for 93% of active users vs. 21% of inactive users.

CHAPTER SIX: EXPERIMENTAL RESULTS

To test the features that we constructed, we are proposing for the classification task, we considered different classifiers, namely Support Vector Machine (SVM), Logistic Regression, and Random Forest. To deal with class unbalance, we used class weighting. To evaluate the performances, we performed 10-fold cross validation and measured the results according to Area Under the ROC curve (AUROC), precision, and recall.

Table 6.1 Performances of our features and comparison with prognoZit according to precision, recall, and Area Under the ROC curve (AUROC) metrics

Our Features	Precision (Active Users)	Precision (Inactive Users)	Recall (Active Users)	Recall (Inactive Users)	AUROC
SVM	0.286	**0.998**	**0.902**	0.957	**0.975**
Logistic Regression	0.257	**0.998**	0.901	0.950	0.973
Random Forest	**0.599**	0.987	0.318	**0.996**	0.968

prognoZit	Precision (Active Users**)**	Precision (Inactive Users)	Recall (Active Users)	Recall (Inactive Users)	AUROC
SVM	**0.730**	0.987	0.358	0.981	0.941
Logistic Regression	0.487	**0.998**	**0.943**	0.980	0.959
Random Forest	0.631	0.993	0.647	**0.989**	**0.963**

The first three rows in Table 6.1 show classification performances for our features when we consider the whole user's edit history. The best performing classifier is SVM with an AUROC of 0.975. Precision and recall for the class of inactive users are also very high: 0.998 precision and 0.957 recall (a better recall result of 0.996 is obtained with Random Forest). The best recall for active users is also obtained with SVM (0.902) while the corresponding precision is 0.286 and a better one can obtained with Random Forest.

Comparison with Related Work

We compare our results with the first prize winner, the prognoZit team [7] of the WikiChallenge competition [16]. prognoZit used features based on number of edits and number of reverted edits on different time periods to predict the future user's number of edits. This task is very close to our problem, since predicting that a user will do zero or very few edits in the future is like saying that they will become an inactive user. As prognoZit extracted features according to the dates of the dataset provided in the WikiChallenge (which are from January 2001 to September 2010), we scaled the time periods to be in our dataset period, i.e. from January 2013 to July 2014.

As prognoZit extracted features according to the dates of the dataset provided in the WikiChallenge (which are from January 2001 to September 2010), we scaled the time periods to be in our dataset period, i.e. from January 2013 to July 2014. The features we used are as follows.

- No of Reverted Edits (re1) from 2013-01-01 to 2014-05-31 (73 weeks).
- No of Reverted Edits (re2) from 2014-05-31 to 2014-07-31 (8 weeks).
- No of Edits (e1) from 2013-01-01 to 2013-10-01 (39 weeks).
- No of Edits (e2) from 2013-10-01 to 2014-03-15 (23 weeks).

- No of Edits (e3) from 2014-03-15 to 2014-06-01 (11 weeks).
- No of Edits (e4) from 2014-06-01 to 2014-06-15 (2 weeks).
- No of Edits (e5) from 2014-06-15 to 2014-07-01 (2 weeks).
- No of Edits (e6) from 2014-07-01 to 2014-07-10 (1 week).
- No of Edits (e7) from 2014-07-10 to 2014-07-20 (1 week).
- No of Edits (e8) from 2014-07-20 to 2014-07-25 (4 days).
- No of Edits (e9) from 2014-07-25 to 2014-07-31 (5 days).

The second three rows in Table 6.1 show classification performances of prognoZit according to three different classification algorithms and when we consider the whole users' edit history. In this case, the best performing classifier is Random Forest with an AUROC of 0.963, which is very close to our AUROC (0.975). Results obtained for precision and recall are also comparable to ours. In general, we can say that our approach is comparable to the one proposed by the prognoZit team when we consider the whole edit history of the users. However, the next experiment shows that our approach is much better in the early prediction of inactive users.

Early Prediction of Inactive Users

In this experiment, we compared our performances with prognoZit in the task of early predicting inactive users. More specifically, we computed the average AUROC on 10-fold cross validation of our method and the competitor by using the first k user's edits only. We varied k from 3 to 500.

Results are shown in Figure 6.1 and Table 6.2. As we can see, by considering the first 3 edits only, we are able to differentiate between inactive and active users with an AUROC of 0.72, while the corresponding AUROC for prognoZit is 0.55. Moreover, we

need to look at the first 9 edits to have an AUROC of 0.81, while prognoZit needs 18 edits to reach the same result. In general, our curve is much higher than the competitor's.

Moreover, we note that the features built by prognoZit have a bias towards the length of a user's edit history. In fact, before beginning to edit and after leaving the community, many features will be zero because the time periods used for the features are based in the global dataset dates. Considering edit history length is not helpful if we want to early predict inactive users in order to perform actions to keep them contributing longer in the Wikipedia community.

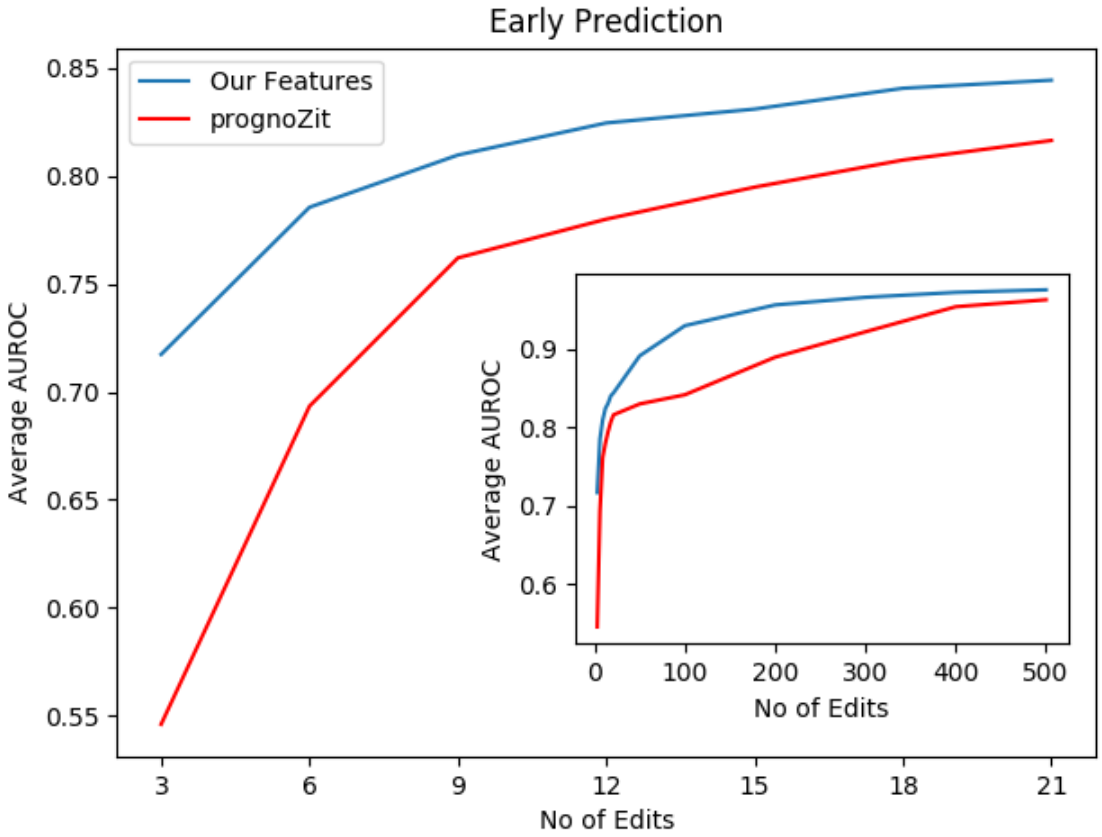


Figure 6.1 Average AUROC for early prediction of inactive users. The blue line represents our features + SVM and the red one represents prognoZit features + Random Forest

Table 6.2 Average AUROC for Our vs prognoZit features for first 21 edits

No of Edits	3	6	9	12	15	18	21
Our Features	0.72	0.79	0.81	0.82	0.83	0.84	0.84
prognoZit	0.55	0.69	0.76	0.78	0.79	0.81	0.81

Other Experiments

Along with the experiments show in this chapter, we did other experiments (reported in Appendix D and E) where we tried other methods to construct features for prediction and compared the results with related work.

Appendix D shows the results of an alternative experiment where we learned the features following the method proposed in this chapter, but where only the first k edits are considered for a user. By comparing these results with the ones listed in Table 6.2, the average AUROC is less for first 6 edits though it is comparable with first 3 and 9 edits.

Appendix E reports on an experiment where we mined frequent patterns from the User Log Dataset according to the same method proposed in this chapter, but by considering only if two consecutive edits are executed very fast, fast, or slow. Results achieved with these time-based features are in favor of prognoZit in the early prediction, while they are comparable when we consider the whole edit history. When we consider the whole edit history, and we add the time-based features to the 78 ones considered in this chapter, results are comparable to the ones reported in Tables 6.1 and 6.2.

CHAPTER SEVEN: WHY LEAVE WIKIPEDIA?

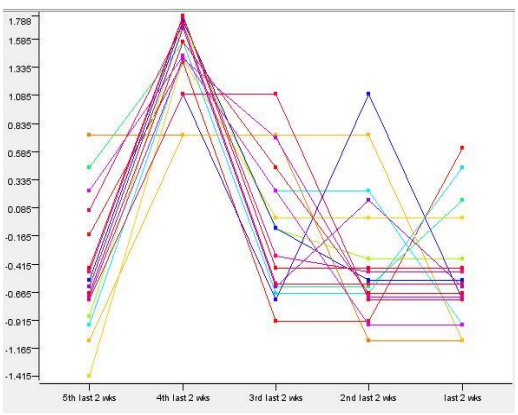
In this section, we study the differences in the editing behavior of active vs. inactive users according to users' involvement in edit wars, reverted edits, meta-pages editing, and categories edited. Finally, we show some rules we extracted that explain our proposed predictive model.

Involvement in Edit Wars

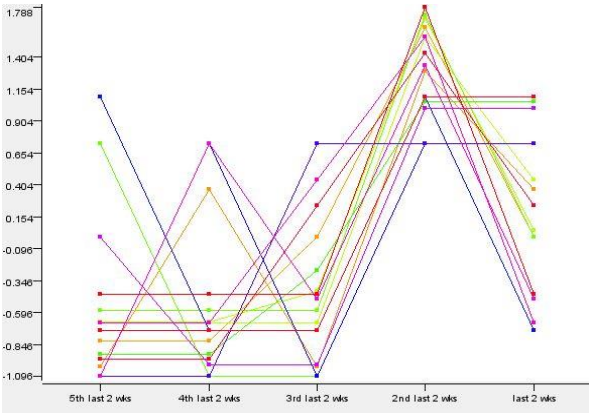
Edit warring occurs when other users do not agree on the content of a page or revision made by another user [29]. We define an edit war as one user making a revision to a page, followed by other users reverting that revision, and this pattern happens at least 2 consecutive times. We say that a user is involved in an edit war if their edit is reverted within an edit war.

By comparing how active vs. inactive users are involved in edit wars, we see that active users are highly involved in edit wars (85.9% of them are involved in at least one edit war), while the percentage is much smaller for inactive users (20.6%). The average number of edit wars a user is involved in is 4.28 for active users vs. 0.33 for inactive ones.

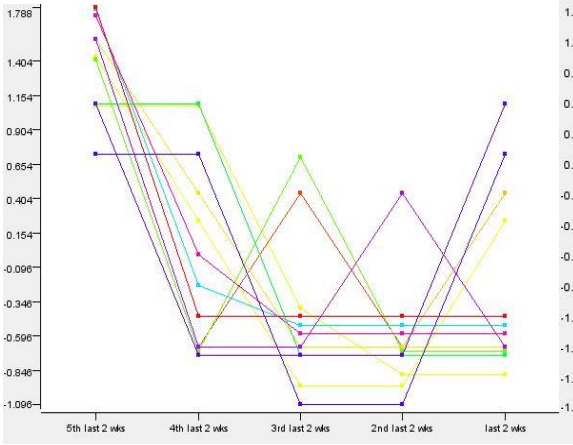
Moreover, we studied the number of edit wars a user is involved in over time. Figure 7.1 shows the clustering of all inactive users' time series so that users having a common shape are shown together in the same plot computed according to the method described in Chapter 4. We observe that, within 10 weeks before contribution stops, there is a significant peak (a rapid increment followed by a rapid decrement) in the number of



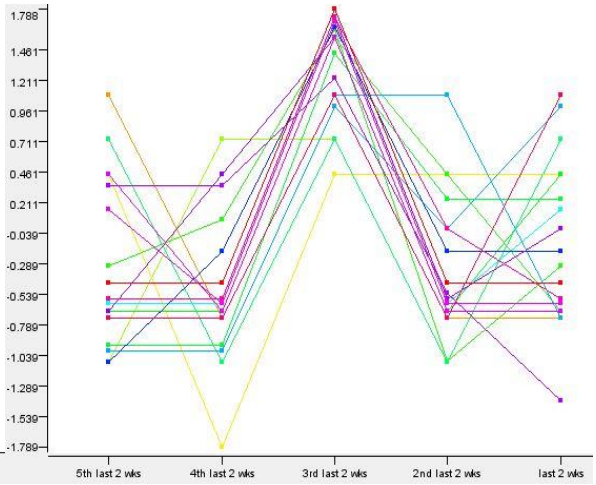
(a) Cluster 0 (124 users)



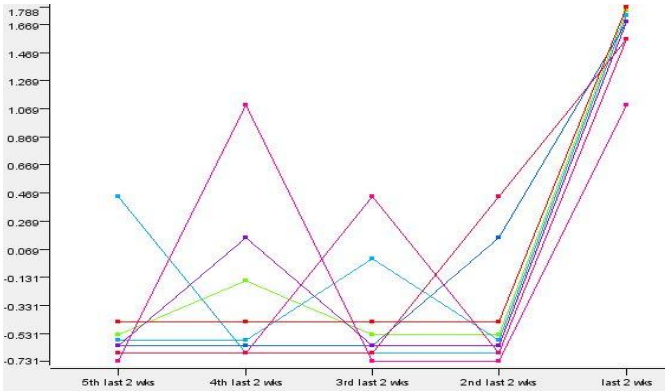
(b) Cluster 1 (242 users)



(c) Cluster 2 (112 users)



(d) Cluster 3 (162 users)



(e) Cluster 4 (1630)

Figure 7.1 Time Series clustering of inactive users' involvement in edit wars

edit wars an inactive user is involved in for 68% inactive users involved in at least one edit war. We also observe that active users have these kinds of peaks in their edit history (check Appendix B), but this seems not to affect their willingness to contribute. In particular, for 9.8% of active users we observe the unique pattern of an increasing involvement in edit wars, meaning that they positively accept critiques from other people in the community.

Reverted Edits

Regarding reverted edits, we observe that the edits made by inactive users are reverted more, compared to active users. On average, the percentage of reverted edits is 9.12% for inactive users vs. 5.25% for active ones.

Editing Meta-pages

There are two different types of pages on Wikipedia: regular article pages and meta-pages. Examples of meta-pages are User pages (where editors describe them-selves), article Talk pages (where editors discuss the content of the associated Wikipedia article), User Talk pages (talk pages associated with user pages), and Wikipedia Project pages.

In studying how users are editing meta-pages, we observe that, on average, inactive users write more on meta-pages than article pages (63.3% of all their edits), while active users write less on meta-pages (30.3%). Also, inactive users write on a more diverse set of meta-pages: the percentage of unique meta- pages among all meta-pages edited by a user is, on average, 29% for inactive users vs. 10.3% for active ones.

Figure 7.2 shows the average number of meta-page edits by meta-page type. As we can see, both classes of users have the same trend: the most edited type of meta-pages

is, on average, User page, followed by Talk pages, User Talk pages, and Project pages. Inactive users edit, on average, many more User pages than active ones (77% vs. 36%).

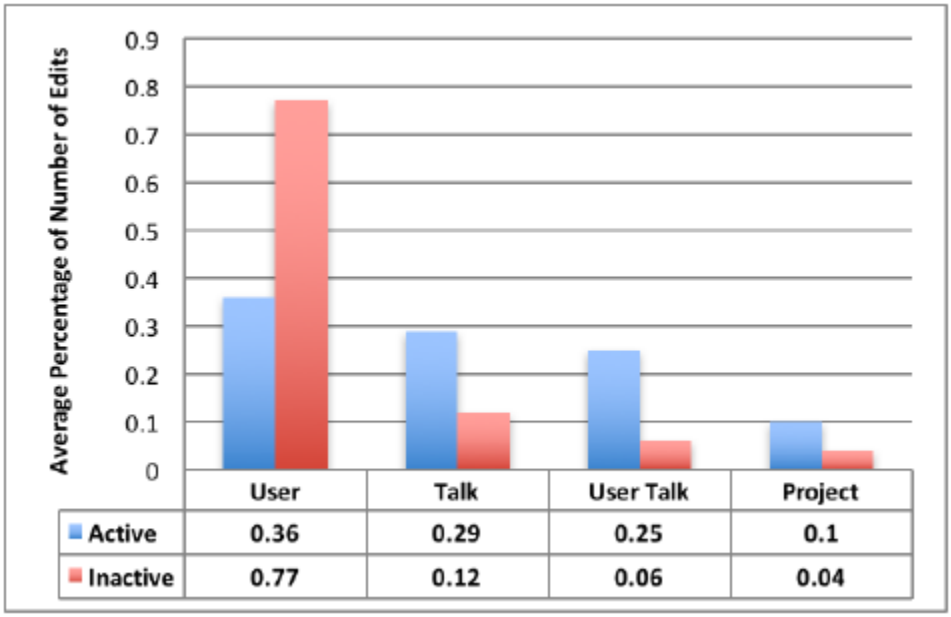


Figure 7.2 Average percentage of number of edits on different types of Wikipedia meta-pages for active (blue) vs. inactive (red) users.

Categories of Edited Pages

Active users edit many more pages from different categories than inactive users: the average number of different categories edited by active users during all their edit history is 868.5 vs. 48.9 for inactive users.

When we look at pairs $(p_1; p_2)$ of consecutive edits, we have that, when $p_1 \neq p_2$, active users consecutively edit pages that have much more categories in common (48 on average), while the corresponding number of inactive users is 4, on average. Thus, active users consecutively edit pages that are much more similar (in terms of common categories) between them than inactive users.

Explaining Our Model

The model we propose in this project to identify inactive users is based on a SVM, working with behavior-based features. SVM is a very complex model represented by an hyperplane which is not very intuitive and easy to understand by humans. Thus, this model does not give any easy explanation on why some users are predicted to leave the community. In order to understand why some users stop contributing on Wikipedia, we used the following technique to explain complex separators such as SVM, which consists of computing decision rules to explain the model produced by the SVM [30]. More specifically, the SVM is seen as a black box and it is used to generate an artificial dataset that is used by traditional rule learning methods (e.g. decision trees such as CART or C4.5) to extract rules from the artificial dataset. The artificial dataset consists of replacing the class of the points in the training set with the class predicted by the learned SVM. The extracted rules are much easier to understand and usually give a better picture in explaining the prediction done by the SVM.

Table 7.1 reports the top 6 rules we extracted to explain our SVM-based model ordered by descending number of users classified by the rule. The body of each rule (left side) expresses a conjunction of patterns that must be present or not present (for patterns preceded by the negation symbol ¬) in order to conclude the prediction in the head of the rule (right side). For instance, the last rule in the table says that if a user does not have the pattern "rnnnf, rncnv" and has all the patterns "nnuos, rnnnf", rnnnv, "nnuof, rncnv", and "rnnnf, rncnf" in their edit history, then the user is active.

"*rnnnf, rncnv*" means re-edit of a non meta-page, non consecutively, not due to reversion and fast (rnnnf) followed by a re-edit of a non meta-page, consecutively, not

due to reversion, and very fast (rncnv).

"*nnuos, rnnnf*" means new edit of a non meta-page not having any path from the previous edit in the hyperlink graph, which has at least one category in common with the previous edit, and the edit happens within more than 15 mins from the previous one (nnuos), followed by a re-edit of a non meta-page, non consecutively, not due to reversion, and fast.

rnnnv is re-edit of a non meta-page, non consecutively, not due to reversion, and very fast.

"*nnuof, rncnv*" means new edit of a non meta-page not having any path from the previous edit in the hyperlink graph, which has at least one category in common with the previous edit, and fast (nnuof), followed by a re-edit of a non meta-page, consecutively, not due to reversion, and very fast (rncnv).

"*rnnnf, rncnf*" means re-edit of a non meta-page, non consecutively, not due to reversion, and fast (rnnnf) followed by re-edit of a non meta-page, consecutively, not due to reversion, and fast (rncnf).

The set of rules extracted gives an approximation of the SVM predictive model with a fidelity of 99.6% and explains more clearly why a user is predicted to be an active or inactive one. The fidelity is the number of users where the classification of rules agree with the classification of the SVM upon total number of users.

Table 7.1 Top-6 rules explaining our model. A negation (¬) before a pattern f means that f is not present in the edit history of the user. The last column reports the number of users covered by each rule.

Rule	No. of Users
¬ "rnnnf,rncnv" ∧ ¬ rmnrf ∧ ¬ "nnuof,rncnv,rncnv" ∧ ¬ nmuov ∧ ¬ "rnnnf,rnnnf" ∧ ¬ "nnuof,rncnv" ∧ ¬ rmnrv ∧ ¬ "rnnnf,rncnf"'→ Inactive	13160
¬ "rncnv,rnnnf" ∧ ¬ rnnrf ∧ ¬ rmnrf ∧ ¬ "rncnf,nnuof" ∧ ¬ nnuuv ∧ ¬ nmuov ∧ ¬ "rnnnf,rnnnf" ∧ ¬ "nnuof,rncnv" ∧ ¬ rmnrv ∧ ¬ "rnnnf,rncnf" → Inactive	607
¬ "nnuof,rnnnv" ∧ ¬ nmuof ∧ ¬ "nnuof,rncnv,rnnns" ∧ ¬ nnuuv ∧ ¬ "rnnnf,rnnnf" ∧ ¬ nmuov ∧ ¬ "nnuof,rncnv" ∧ ¬ rmnrv ∧ ¬ "rnnnf,rncnf" → Inactive	368
¬ "nnuos,rncnv,rnnns" ∧ ¬ rnnrf ∧ ¬ "rnnnf,rncnv" ∧ ¬ rmnrf ∧ ¬ "nnuof,rncnv,rncnv" ∧ ¬ nmuov ∧ ¬ "rnnnf,rnnnf" ∧ ¬ "nnuof,rncnv" ∧¬ rmnrv ∧ ¬ "rnnnf,rncnf" → Inactive	220
¬ rmcrf ∧ ¬ "nnuof,nnuos,rncnv" ∧ ¬ "rncnv,rnnnf " ∧ ¬ "rnnnf,rnnns" ∧ ¬ rmnrv ∧ ¬ nnuuv ∧ ¬ "rnnnf,rnnnf" ∧ ¬ "nnuof,rncnv" ∧ ¬ "rnnnf,rncnf" → Inactive	203
¬ "rnnnf,rncnv" ∧ ¬ "nnuos,rnnnf" ∧ ¬ rnnnv ∧ ¬ "nnuof,rncnv" ∧ ¬ "rnnnf,rncnf" → Active	195

CHAPTER EIGHT: CONCLUSIONS

In this project, we proposed a predictive model based on users' editing behavior that is able to predict which editor will become inactive in the Wikipedia community with an AUROC of 0.97 and a precision of 0.99. Moreover, we showed that our model significantly beats competitors in the task of early prediction of inactive users. By comparing editing behavior of active vs. inactive users, we discovered that active users are more involved in edit wars and positively accept critiques, and edit much more different categories of pages. On the other hand, inactive users have more edits reverted and edit more meta-pages (and in particular User pages).

REFERENCES

[1] S. L. Bryant, A. Forte, and A. Bruckman, “Becoming wikipedian: Transformation of participation in a collaborative online encyclopedia,” in Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group Work, 2005, pp. 1–10.

[2] O. Nov, “What motivates wikipedians?” Commun. ACM, vol. 50, no. 11, pp. 60–64, 2007.

[3] Y. Chen, F. M. Harper, J. Konstan, and S. Xin Li, “Social comparisons and contributions to online communities: A field experiment on movie- lens,” The American economic review, vol. 100, no. 4, pp. 1358–1398, 2010.

[4] L. Jian and J. K. MacKie-Mason, “Why leave wikipedia?” in iConference, 2008.

[5] S. Asadi, S. Ghafghazi, and H. R. Jamali, “Motivating and discouraging factors for wikipedians: the case study of persian wikipedia,” Library Review, vol. 62, no. 4/5, pp. 237–252, 2013.

[6] Wikipedia English Statistics, https://stats.wikimedia.org/EN/TablesWikipediaEN.htm#activitylevels

[7] WikiChallenge First Prize Winner’s Wikipedia Page, https://meta.wikimedia.org/wiki/Research:WikiParticipation Challenge_prognoZit

[8] D. Zhang, K. Prior, M. Levene, R. Mao, and D. van Liere, “Leave or stay: The departure dynamics of wikipedia editors,” in International Conference on Advanced Data Mining and Applications, 2012, pp. 1–14.

[9] T. Iba, K. Nemoto, B. Peters, and P. A. Gloor, “Analyzing the creative editing behavior of wikipedia editors: Through dynamic social network analysis,” Procedia-Social and Behavioral Sciences, vol. 2, no. 4, pp. 6441–6456, 2010.

[10] H. T. Welser, D. Cosley, G. Kossinets, A. Lin, F. Dokshin, G. Gay, and M. Smith, "Finding social roles in wikipedia," in Proceedings of the 2011 iConference, 2011, pp. 122–129.

[11] C.-Y. Lai and H.-L. Yang, "The reasons why people continue editing wikipedia content–task value confirmation perspective," Behaviour & Information Technology, vol. 33, no. 12, pp. 1371–1382, 2014.

[12] S. Kim, S. Park, S. A. Hale, S. Kim, J. Byun, and A. H. Oh, "Understanding editing behaviors in multilingual wikipedia," PloS one, vol. 11, no. 5, p. e0155305, 2016.

[13] L. Jian, J. MacKie-Mason, B. Chiao, A. Levchenko, A. Zellner, J. Kmenta, J. Dreze, and W. Oberhofer, "Incentive-centered design for user-contributed content," The Oxford Handbook of the Digital Economy, p. 399, 2012.

[14] S. Kumar, F. Spezzano, and V. Subrahmanian, "Vews: A wikipedia vandal early warning system," in 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015, pp. 607– 616.

[15] T. Green and F. Spezzano, "Spam users identification in wikipedia via editing behavior," in The 11th International AAAI Conference on Web and Social Media, 2017, pp. –.

[16] Wiki Challenge Competition, https://www.kaggle.com/c/wikichallenge.

[17] WikiChallenge Third Prize Winner's Wiki Page, https://meta.wikimedia.org/wiki/Research:Wiki_Participation_Challenge_zeditor

[18] VEWS, http://www.cs.umd.edu/~vs/vews.

[19] Han, J., Cheng, H., Xin, D. et al. Data Min Knowl Disc (2007) 15: 55. doi:10.1007/s10618-006-0059-1

[20] Sequential pattern mining, https://en.wikipedia.org/wiki/Sequential_pattern_mining

[21] R. Agrawal and R. Srikant, "Mining Sequential Patterns," Proc. 1995 Int'l Conf. Data Eng. (ICDE '95), pp. 3-14, Mar. 1995

[22] http://www.philippe-fournier viger.com/spmf/index.php?link=documentation.php#examplePrefixSpan

[23] Pratik Saraf, R R Sedamkar and Sheetal Rathi. Article: PrefixSpan Algorithm for Finding Sequential Pattern with Various Constraints. International Journal of Applied Information Systems 9(3):37-41, June 2015.

[24] Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, Jianyong Wang, Helen Pinto, Qiming Chen, Umeshwar Dayal, Mei-Chun Hsu. Mining Sequential Patterns by Pattern-Growth: The PrefixSpan Approach. IEEE Transactions on Knowledge and Data Engineering, 2004, 16(10), pp. 1-17.

[25] http://www.philippe-fournier-viger.com/spmf/index.php?link=download.php

[26] Liao, T. Warren. “Clustering of time series data - a survey.” Pattern Recognition 38 (2005): 1857-1874.

[27] Zafarani, Reza, Mohammad Ali Abbasi, and Huan Liu. Social media mining: an introduction. Cambridge University Press, 2014.

[28] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu, “Prefixspan: Mining sequential patterns by prefix-projected growth,” in Proceedings of the 17th International Conference on Data Engineering, 2001, pp. 215–224.

[29] Edit Warring, http://en.wikipedia.org/wiki/Wikipedia:Edit warring.

[30] N. Barakat and A. P. Bradley, “Rule extraction from support vector machines: A review,” Neurocomput., vol. 74, no. 1-3, pp. 178–190, 2010.

APPENDIX A

Our Features

The following table reports the list of the 78 editing patterns used as features in our model to predict inactive users and extracted as explained in Chapter 5.

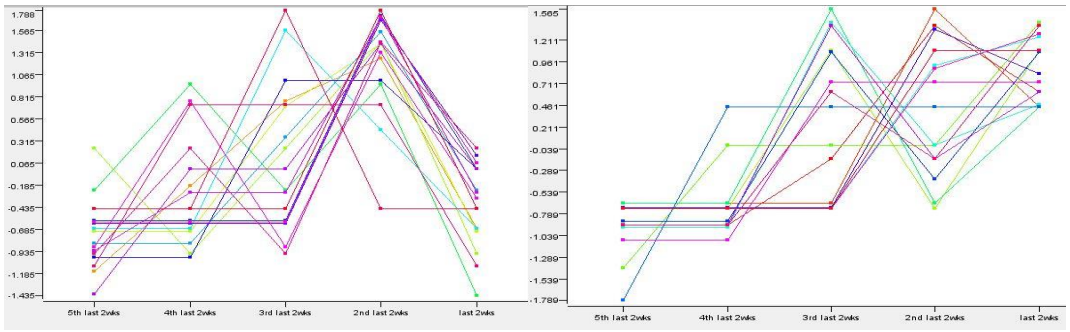
Table A.1 Complete list of features used in our model

Feature0 – rnnnv	Feature39 - "rncnf,nnuof"
Feture1 – rnnnf	Feature40 - "nnuos,rncnf,rnnns"
Feature2 – nnuov	Feature41 - "nnuos,rnnns,nnuos"
Feature3 – nnuof	Feature42 - "rncnv,nnuos,rnnns"
Feature4 – rnnns	Feature43 - "nnuos,nnuos,nnuof"
Feature5 - "rncnv,rnnnf"	Feature44 - "nnuos,nnuof,rncnv"
Feature6 - "rncnv,rnnnv"	Feature45 - rmcrf
Feature7 - "nnuof,rncnf"	Feature46 - rmnrv
Feature8 - "rnnnf,rncnv"	Feature47 - rncrs
Feature9 - "nnuof,rnnns"	Feature48 - rnnrf
Feature10 - "nnuof,rncnv,nnuos"	Feature49 - rnnrv
Feature11 - "rncnv,nnuof,rncnv"	Feature50 - "nnuos,rnnnv"
Feature12 - "nnuof,nnuos,rncnv"	Feature51 - "rnnnv,rncnv"
Feature13 - "nnuof,rncnv,rncnv"	Feature52 - "nnuof,rnnnv"
Feature14 - "nnuof,nnuos,nnuos"	Feature53 - "rnnnv,nnuos"
Feature15 – nmuov	Feature54 - "rnnnf,rnnns"
Feature16 – nnuuv	Feature55 - "nnuof,nnuof,nnuos"

Feature17 – rnnrs	Feature56 - "nnuof,nnuof,rncnv"
Feature18 – rncrf	Feature57 - "rncnv,rncnv,rnnnv"
Feature19 – rmcrv	Feature58 - "rncnv,rncnv,rnnnf"
Feature20 - "nnuof,rnnnf"	Feature59 - "nnuof,rncnv,rncnf"
Feature21 - "rnnnf,nnuos"	Feature60 - rncrv
Feature22 - "rnnnf,rnnnf"	Feature61 - nnuos
Feature23 - "rnnnf,rncnf"	Feature62 - rmnnf
Feature24 - "rncnf,rnnnv"	Feature63 - "nnuof,nnuos"
Feature25 - "rncnv,rnnnf,rncnv"	Feature64 - "nnuof,rncnv”
Feature26 - "rncnv,rnnns,nnuos"	Feature65 - "nnuof,nnuof"
Feature27 - "rncnv,nnuof,nnuos"	Feature66 - "rncnv,rnnns,rncnv"
Feature28 - "rnnns,nnuos,rncnv"	Feature67 - "rncnf,nnuos,rncnf"
Feature29 - "nnuof,rncnv,rnnns"	Feature68 - "nnuos,rncnv,rnnns"
Feature30 – nmuof	Feature69 - rmcrs
Feature31 – rmnnv	Feature70 - rmnrf
Feature32 – nmuos	Feature71 - rmnrs
Feature33 – rncnf	Feature72-"rnnns,rnnnv"
Feature34 – rncns	Feature73-"rnnns,nnuof"
Feature35 - "rnnns,nnuos"	Feature74-"rnnnf,nnuof"
Feature36 - "rncnf,rnnnf"	Feature75-"nnuos,rncnv,nnuof"
Feature37 - "nnuos,rnnnf"	Feature76-"nnuof,rncnf,nnuos"
Feature38 - "rncnv,nnuof"	Feature77-"nnuof,rncnf,rncnv"

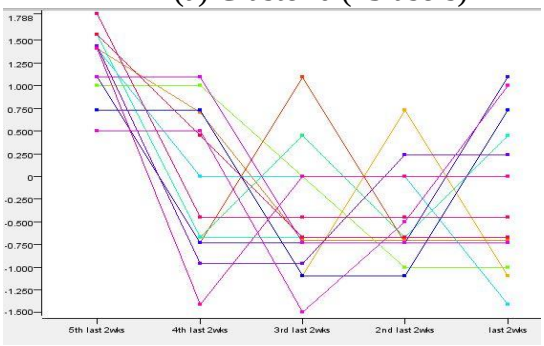
APPENDIX B

Active User's Involvement in Edit Wars

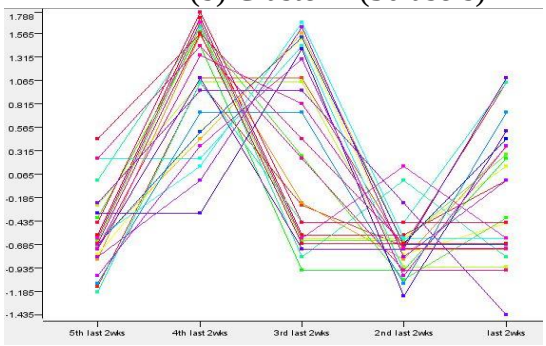


(a) Cluster 0 (43 users)

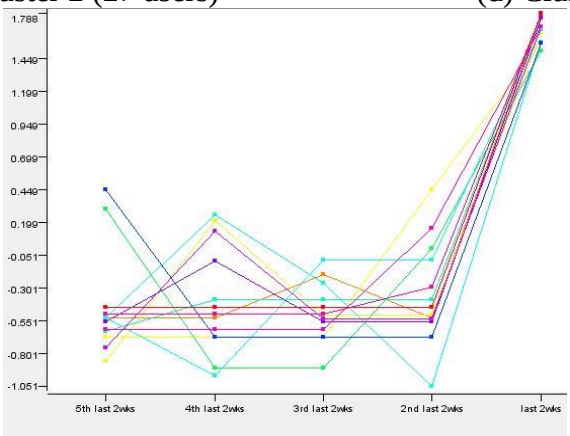
(b) Cluster 1 (30 users)



(c) Cluster 2 (27 users)



(d) Cluster 3 (42 users)



(a) Cluster 5 (73 users)

Figure B.1 Time-series clustering of active user's involvement in edit wars

APPENDIX C

In this Appendix we report time-series clustering of both active and inactive users for features like reverted edits percentage, number of common categories, meta-page and unique-meta page percentage.

Reverted Edits Percentage Feature

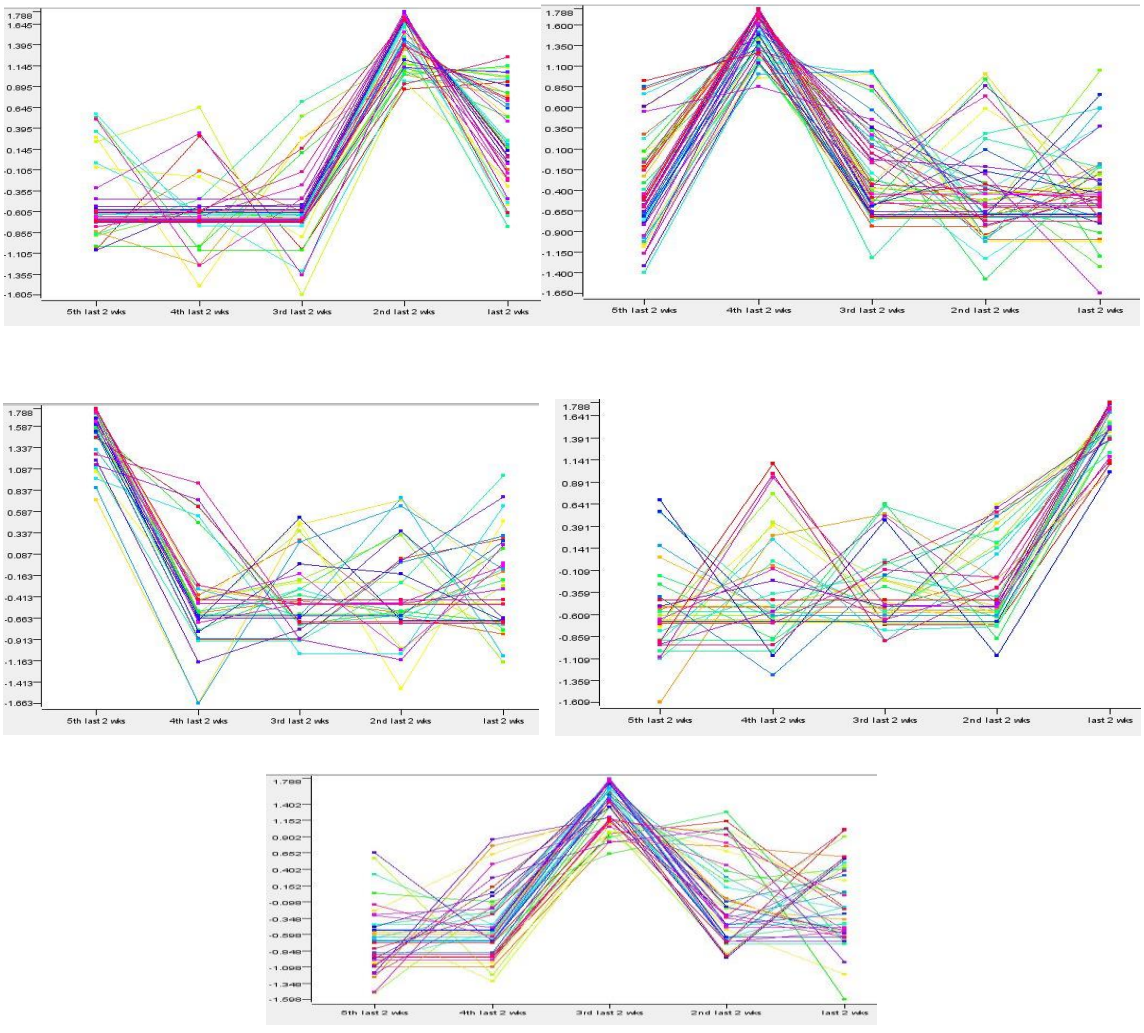


Figure C.1 Time series clustering of active user's involvement in reverted edits percentage

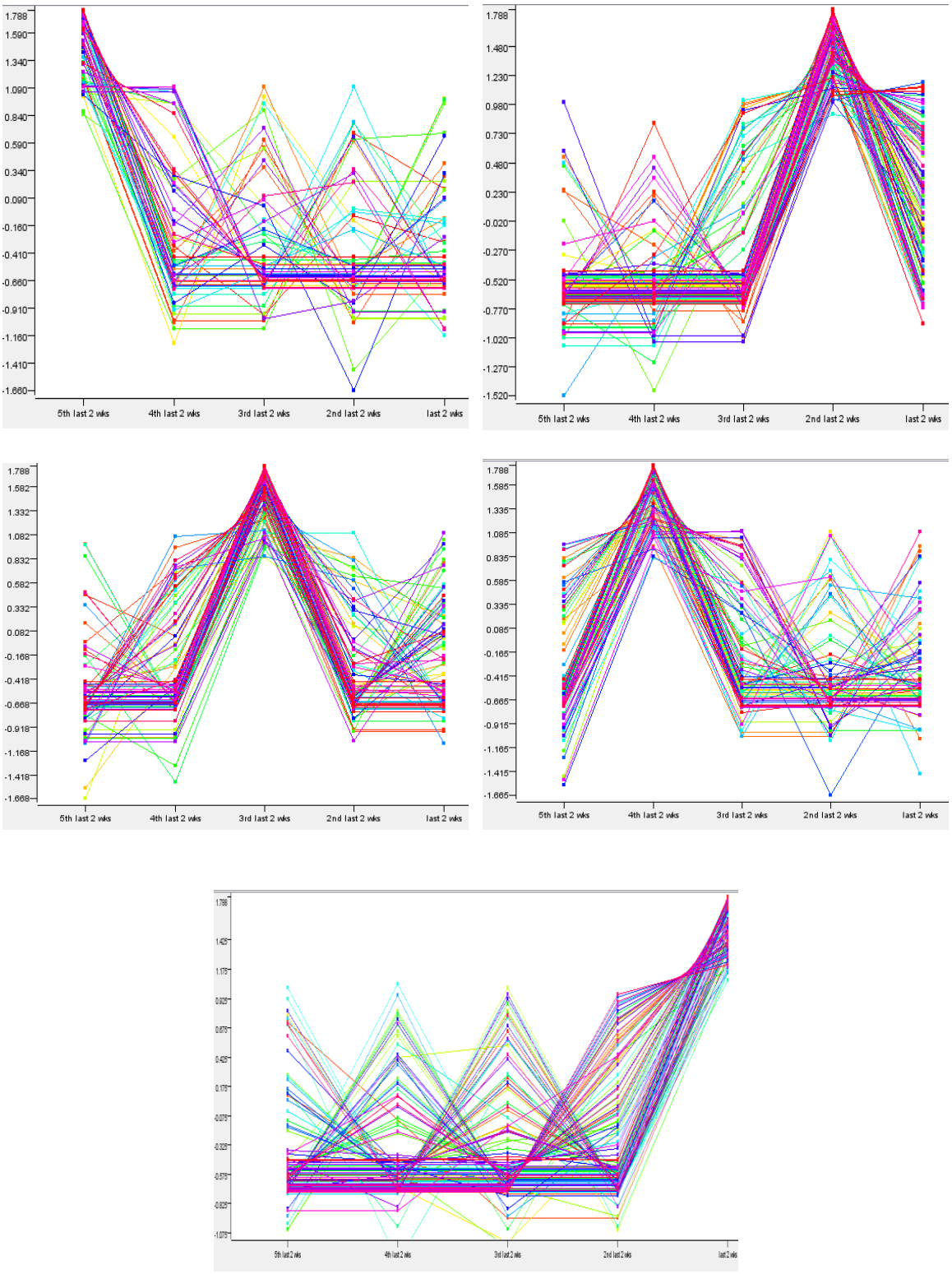


Figure C.2 Time series clustering of inactive user's involvement in reverted edits percentage

Meta Page Percentage Feature

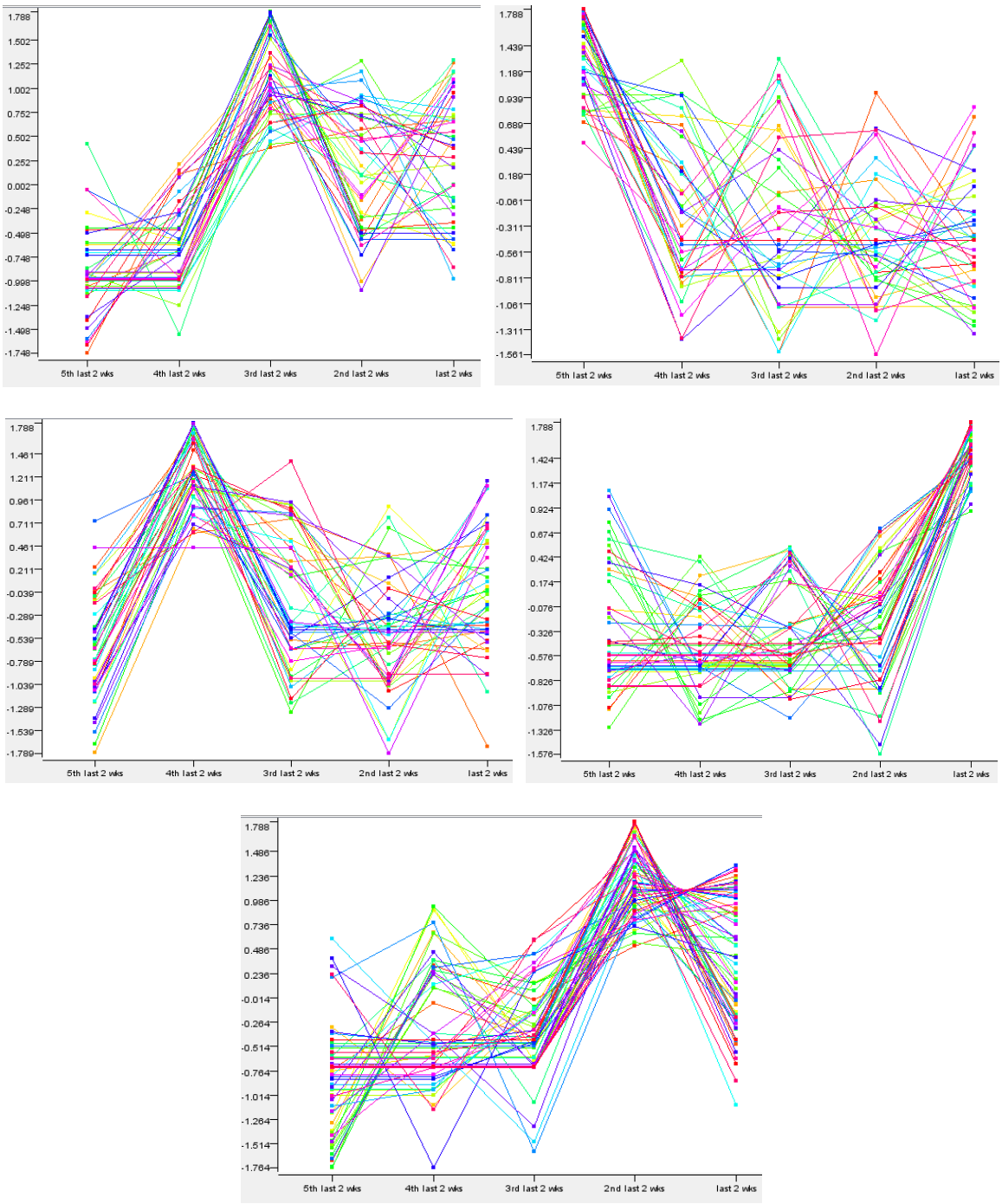


Figure C.3 Time series clustering of active user's involvement in meta page percentage

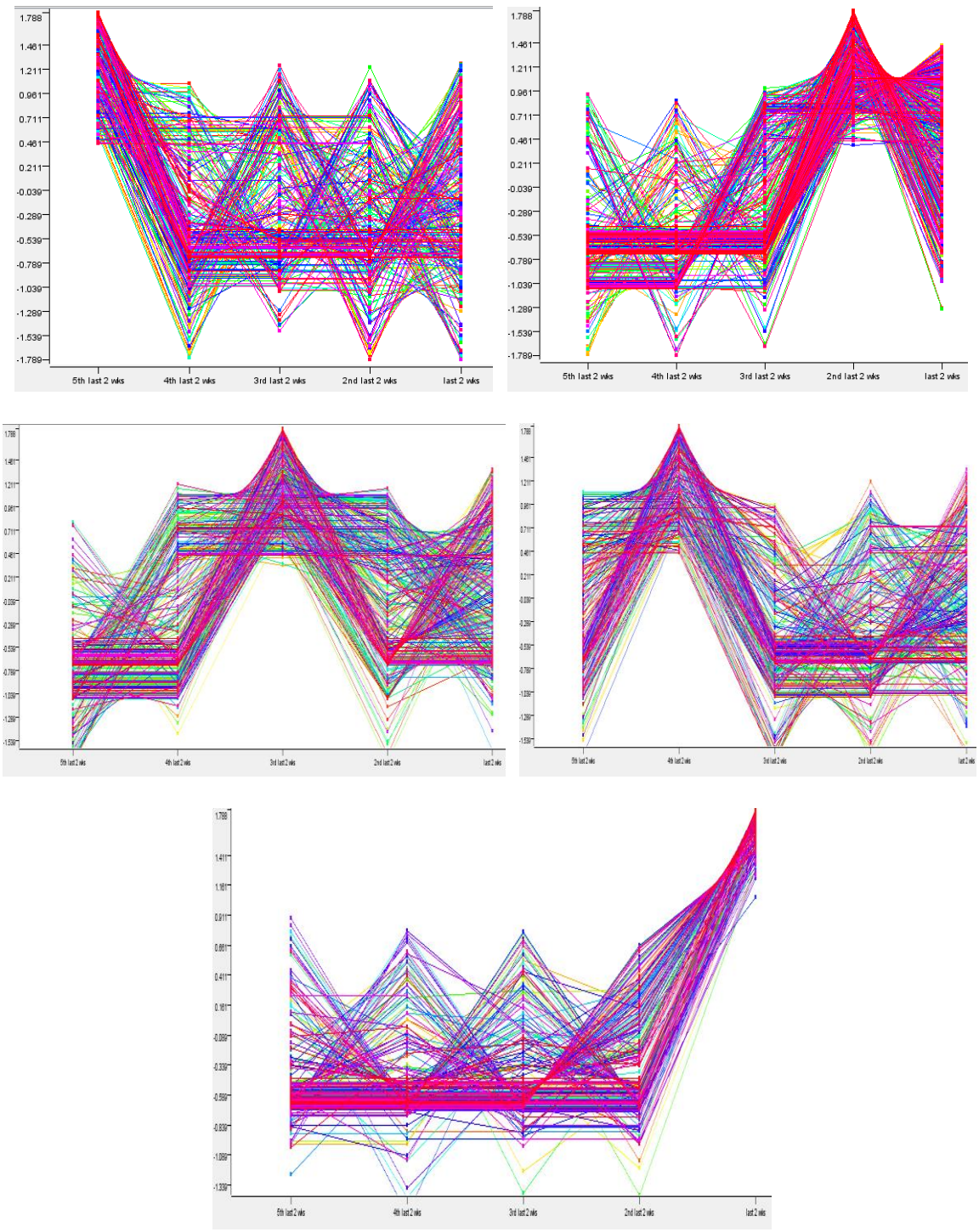


Figure C.4 Time series clustering of inactive user's involvement in meta page percentage

Unique-meta Page Percentage

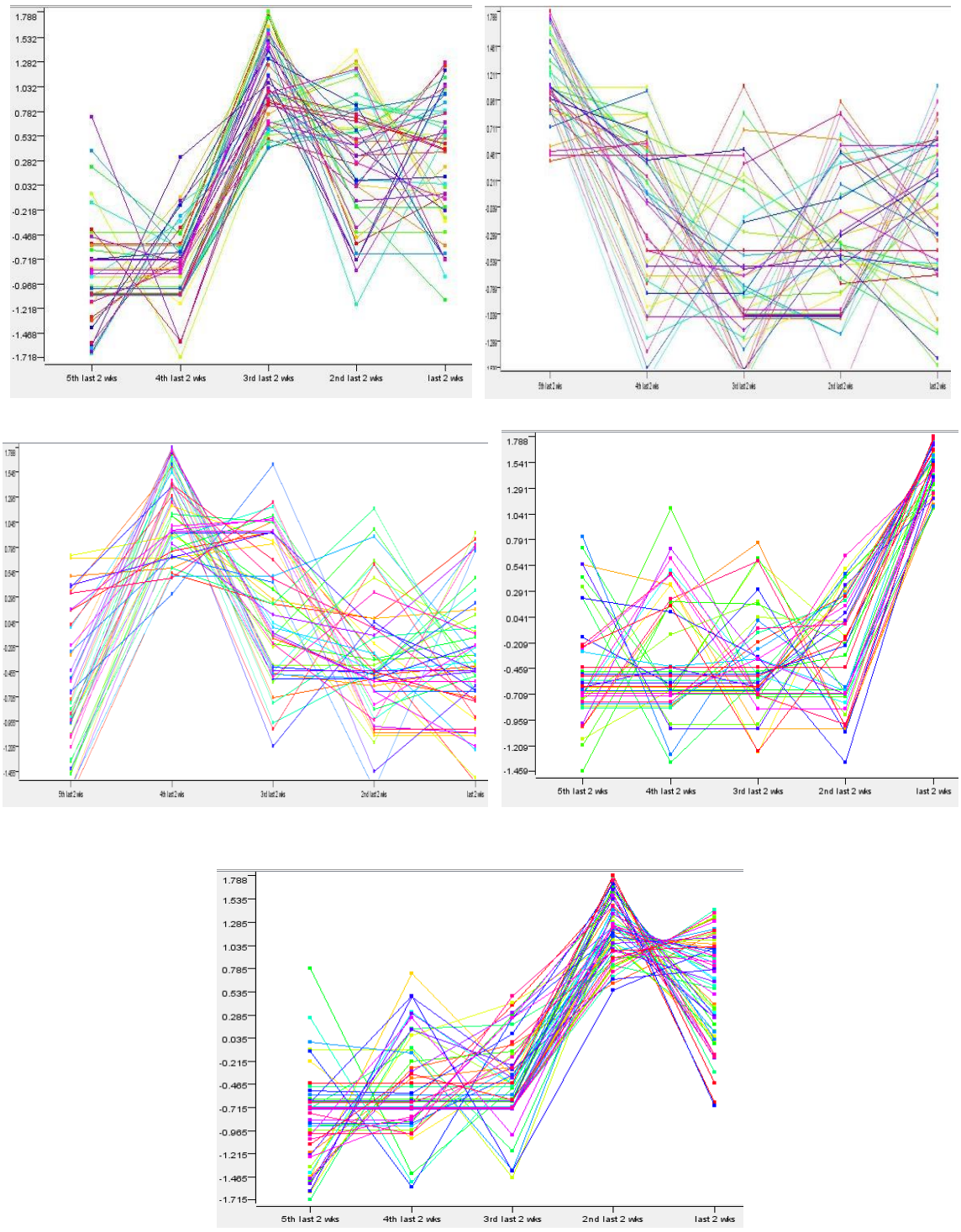


Figure C.5 Time series clustering of active user's involvement in meta page percentage

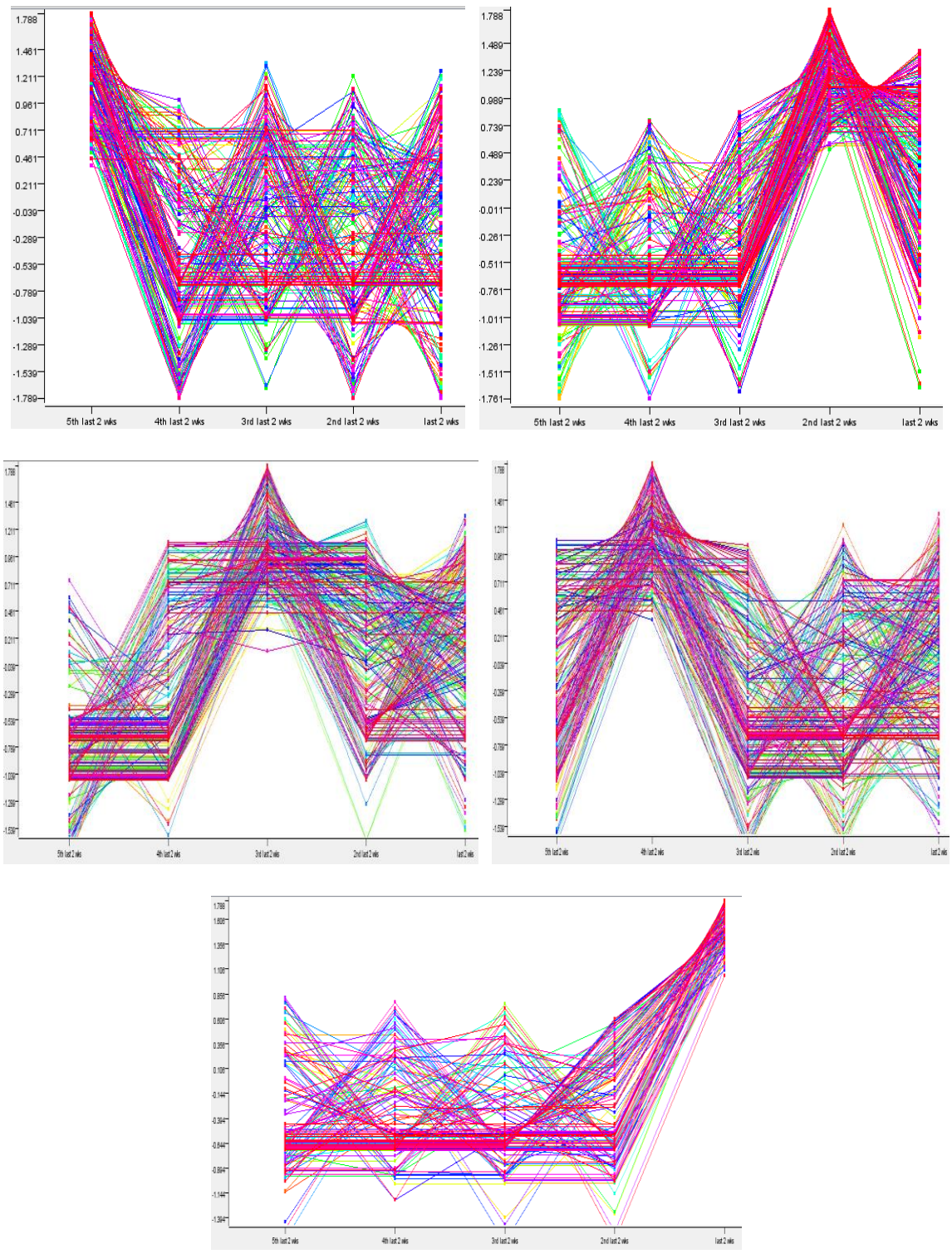


Figure C.6 Time series clustering of inactive user's involvement in meta page percentage

Number of Common Categories

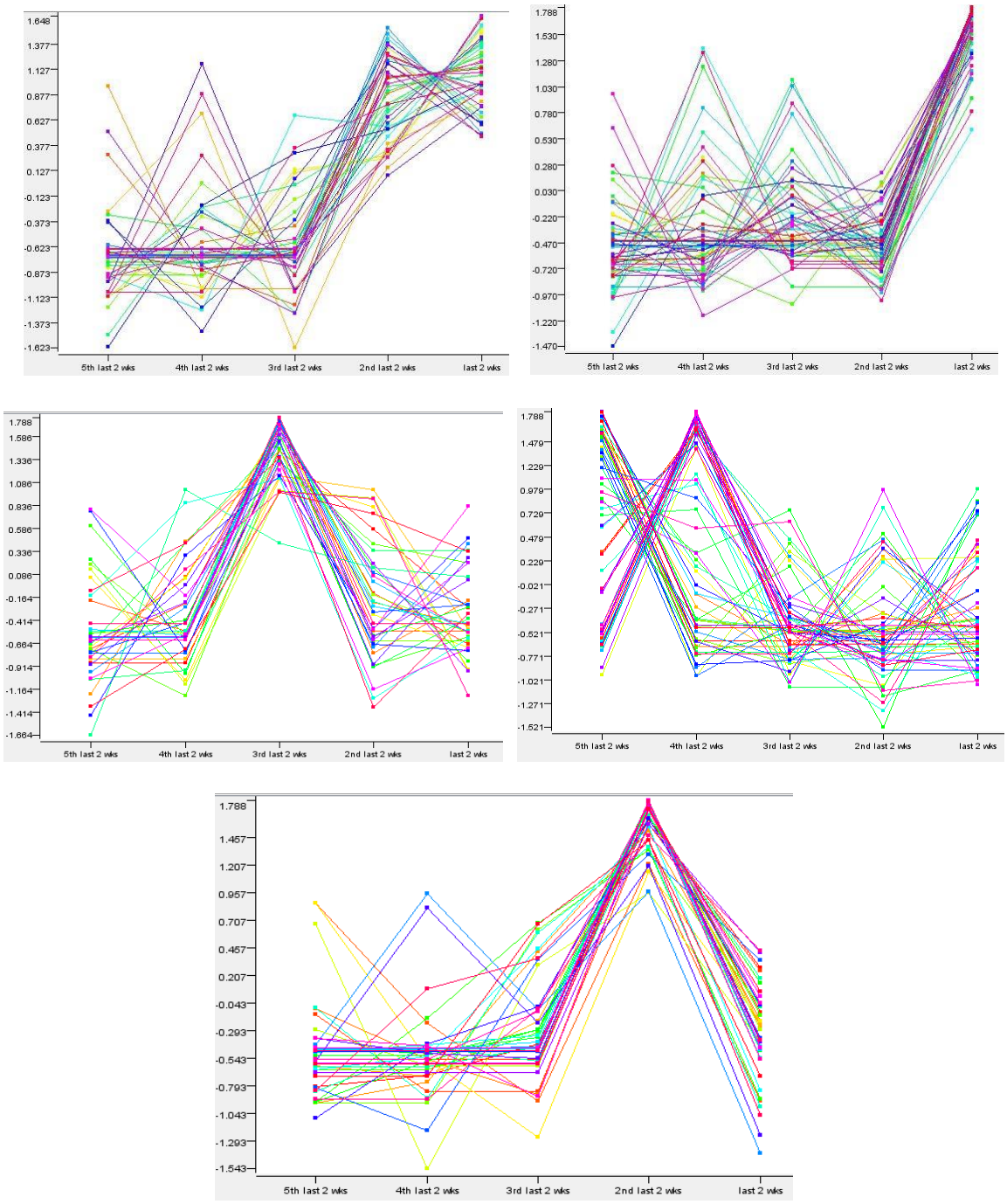


Figure C.7 Time series clustering of active user's involvement in number of common categories

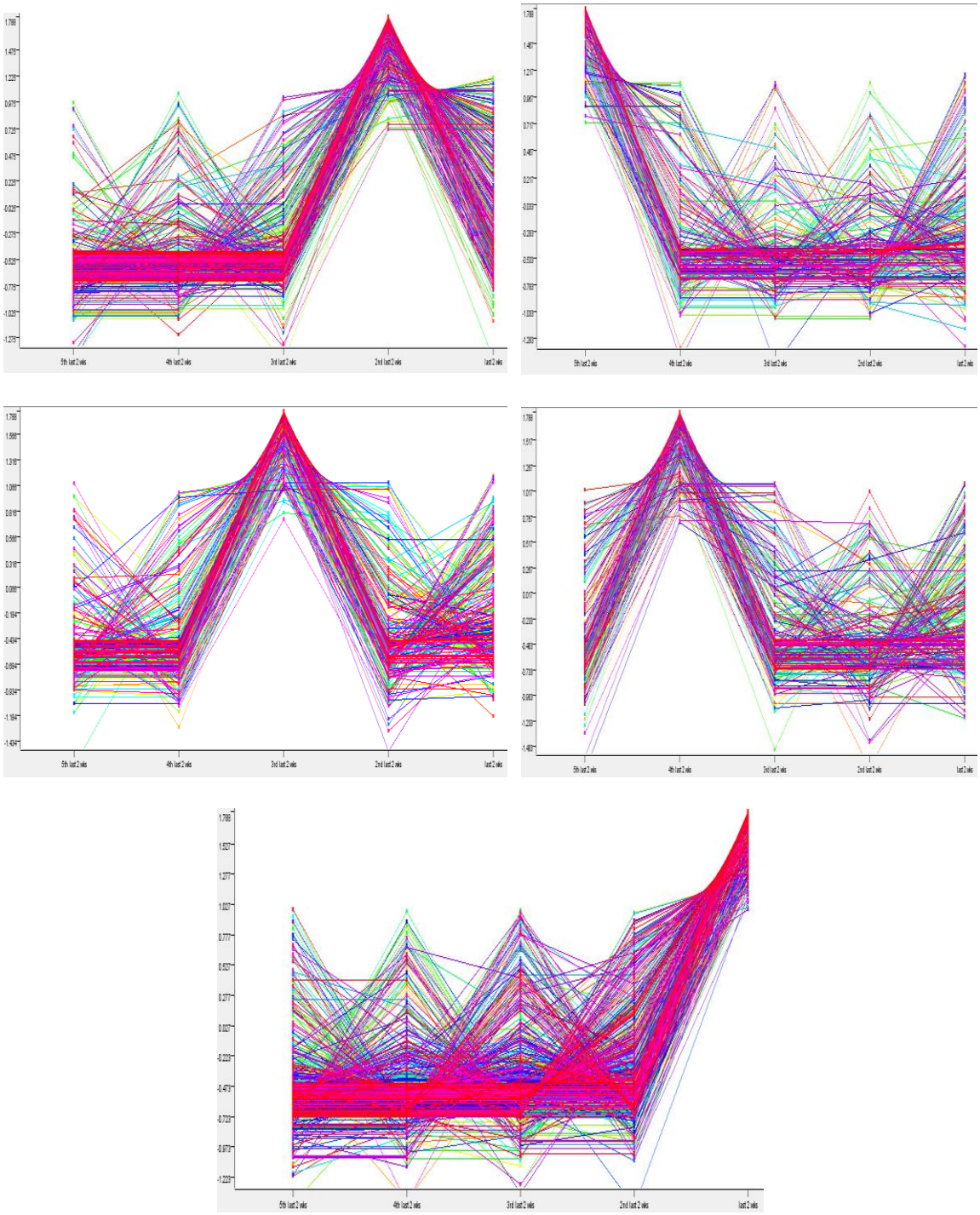


Figure C.8 Time series clustering of inactive user's involvement in number of common categories

APPENDIX D

Other Early Prediction Experiment

We performed an alternative experiment where we learned the features following the method proposed in Chapter 5, but where only the first k edits are considered for a user. Results and comparisons with prognoZit are reported in the following.

First 3 Edits

Feature Extraction

By considering only the first 3 edits of users among their all edits, we mined the frequent patterns and extracted the features in the same way as described in Chapter 5. We considered the 16 most frequent patterns as our features. These features are listed in the Table D.1.

Table D.1 List of features constructed considering first 3 edits

Feature0 - nnuos	Feature8 - rncnv
Feature1 - rncnf	Feature9 - rncns
Feature2 - rmcnv	Feature10 - nmuos
Feature3 - nnuov	Feature11 - nnuos,nnuos
Feature4 - nmuuf	Feature12 - rmcnf
Feature5 - rmcns	Feature13 - nmuus
Feature6 - nnuus	Feature14 - nmuuv
Feature7 - nnuos	Feature15 - rncnv

Comparison with prognoZit

With these new features and different types of classifiers, we computed AUROC average percentage and compared these scores with prize winner prognoZit scores. This comparison is reported in Table D.2.

Table D.2 Average AUROC Our vs prognoZit for first 3 edits

Classifier	Our Features	progoZit Features
Random Forest	0.729	0.546
SVM	0.733	0.552
Logistic Regression	**0.745**	0.532

Looking at the above table it is cleared that Logistic Regression is the best performing classifier with an AUROC of 0.745.

First 6 Edits

This experiment is the same as the previous one but is conducted by considering the first 6 user edits among all their edits. After mining frequent patterns, we ended up taking 32 patterns as our features in this experiment. These features are listed in Table D.3.

Table D.3 List of features constructed considering first 6 edits

Feature0 - rncnv	Feature16 - nnuos
Feature1 - rncnf	Feature17 - nnuof
Feature2 - rncns	Feature18 - nnuos,nnuos
Feature3 - rmcnv	Feature19 - rncnv,rncnv
Feature4 - nnuov	Feature20 - nnuos,rncnv
Feature5 - nnuus	Feature21 - rncnf,rncnv

Feature6 - nmuus	Feature22 - nnuof,nnuos
Feature7 - nmuos	Feature23 - rncnv,rncnf
Feature8 – rncns	Feature24 - rmcnv
Feature9 - rncnv	Feature25 - nmuus
Feature10 - nnuof	Feature26 - rmcns
Feature11 - nnuos	Feature27 - rmcnv,rmcnv
Feature12 - rncnf	Feature28 - nmuuv
Feature13 - nnuus	Feature29 - rmcnf,rmcnv
Feature14 - nmuuf	Feature30 - rmcnv,rmcnf
Feature15 - rmcnf	Feature31 - rmcnf,rmcnf

Comparison with prognoZit

The comparison between the features learned from first 6 edits and the ones of prognoZit is shown in Table D.4.

Table D.4 Average AUROC Our vs prognoZit for first 6 edits

Classifier	Our Features	progoZit Features
Random Forest	0.593	0.693
SVM	0.778	0.683
Logistic Regression	**0.788**	0.668

Looking at the above table, Logistic Regression is the best performing classifier with an AUROC of 0.78.

First 9 Edits

In this case we considered the first 9 user edits among all their edits and ended up taking 48 patterns as our features in this experiment. These features are listed in Table D.5

Table D.5 List of features constructed considering first 9 edits

Feature0 - nnuos	Feature24 - nnuos,nnuos
Feature1 - rncnv	Feature25 - rncnv,rncnv
Feature2 - rncnf	Feature26 - rncns
Feature3 - nnuof	Feature27 - nnuos,rncnv
Feature4 - nnuov	Feature28 - rncnf,rncnv
Feature5 - rncnf,rncnf	Feature29 - nnuos,rncnf
Feature6 - rmcnv	Feature30 - rncnv,nnuos
Feature7 - nnuus	Feature31 - rncnv,rncnf
Feature8 – nmuos	Feature32 - nnuof,nnuos
Feature9 - nmuuf	Feature33 - nnuof,nnuos
Feature10 - rnnns	Feature34 - nnuof,nnuof
Feature11 - nmuus	Feature35 - nnuos,nnuof
Feature12 - nmuuv	Feature36 - nmuus
Feature13 - rncns	Feature37 - nnuos
Feature14 - rncnv	Feature38 - rmcnv,rmcnv
Feature15 - rncnf	Feature39 - rmcnf,rmcnv
Feature16 - rmnns	Feature40 - rmcnv,rmcnf
Feature17 - rncnv,rncnv	Feature41 - rmcnf,rmcnf

Feature18 - nmuos	Feature42 - nnuof
Feature19 - nnuus	Feature43 - rmcnv,rmcnv,rmcnv
Feature20 - rmcns	Feature44 - nmuus,rmcnv
Feature21 - nmuuf	Feature45 - rmcnv,rmcns
Feature22 - rmcnf	Feature46 - rmcns,rmcns
Feature23 - rmcnv	Feature47 - rmcns,rmcnv

Comparison with prognoZit

The comparison between the features learned from first 9 edits and the ones of prognoZit is shown in Table D.6.

Table D.6 Average AUROC Our vs prognoZit for first 9 edits

Classifier	Our Features	progoZit Features
Random Forest	0.645	0.761
SVM	**0.812**	0.761
Logistic Regression	0.806	0.734

Looking at the above table it is cleared that SVM classifier is best performing classifier with an AUROC of 0.81.

Overall, the results obtained with this alternative methodology are comparable to the ones presented in Chapter 6.

APPENDIX E

Time Factor Experiments

In this experiment, we considered frequent patterns extracted by considering only the time elapsed between two consecutive edits. Results and comparisons with prognoZit are reported in the following.

Feature Extraction

We mined frequent patterns from the User Log Dataset according to the same method proposed in Chapter 5, but by considering only if two consecutive edits are executed very fast, fast, or slow. We extracted 18 patterns as features and did experiments with first 3, 6, 9 and all edits. Features we used are reported in Table E.1. Experimental results are shown in Table E.2.

Table E.1 List of features constructed considering all edits

Feature0 - f	Feature9 – f,v
Feature1 - v	Feature10 – s,v
Feature2 - s	Feature11 – f,s
Feature3 – v,f	Feature12 – s,f,v
Feature4 – s,f	Feature13 – f,s,v
Feature5 – s,v	Feature14 – s,v,v
Feature6 – v,s,f	Feature15 – f,s,v
Feature7 – s,v,f	Feature16 – s,v,v
Feature8 – s,f,f	

Table E.2 Time-based features comparision with prognoZit according to AUROC

No of Edits	SVM		Random Forest		Logistic Regression	
	Our Features	progonoZit	Our Features	pognoZit	Our Features	prognoZit
3 Edits	0.500	**0.552**	0.510	0.546	0.491	0.532
6 Edits	0.598	0.683	0.589	**0.693**	0.592	0.668
9 Edits	0.662	**0.761**	0.649	0.761	0.674	0.734
All Edits	0.951	0.941	0.952	**0.963**	0.954	0.959

As we can see from Table E.2, we are not able to beat prognoZit in the early prediction by considering time-based features only, and we are comparable when we consider the whole edit history.

Table E.3 shows what happens if we add the time-based features mined in this appendix to the ones computed in Chapter 5. Also in this case, results are comparable to the ones reported in Tables 6.1 and 6.2.

Table E.3 Features used in our model plus time-based features comparision with prognoZit according to AUROC

No of Edits	SVM		Random Forest		Logistic Regression	
	Our Features	progonoZit	Our Features	pognoZit	Our Features	prognoZit
3 Edits	**0.730**	0.552	0.675	0.546	0.722	0.532
6 Edits	**0.788**	0.683	0.526	0.693	0.775	0.668
9 Edits	**0.822**	0.761	0.630	0.761	0.814	0.734
All Edits	**0.980**	0.941	0.978	0.963	0.980	0.959