A CONTEXT AWARE APPROACH FOR DETECTING ELUSIVE VANDALISM IN WIKIPEDIA

by

RAGA SOWMYA TUMMALAPENTA

(Under the Direction of LAKSHMISH RAMASWAMY)

ABSTRACT

The collaborative model of Wikipedia is simple and open. This nature of Wikipedia challenges its trustworthiness, leading to vandalism. There are several current vandalism detection techniques but none of them focus on detecting elusive vandalism. This type do not contain normal characteristics of vandalism and hence difficult to detect. We have proposed multi-context aware detection techniques for determining whether an elusive edit is vandalized or not. The main idea of these techniques is to check whether an edit lies within the context of other words within a particular Wikipedia article. For the experimental purposes, we make use of a PAN corpus, which is a large collection of Wikipedia edits. Then we perform a feature extraction followed by a data trained classification using WEKA. Accuracy of our methods is calculated using f1-measure. Results show that the context aware techniques are efficient since they result in highly less number of false positives and negatives.

INDEX WORDS: Wikipedia, Vandalism, Elusive Vandalism, WEKA, context, Search Engine, Accuracy

A CONTEXT AWARE APPROACH FOR DETECTING ELUSIVE VANDALISM IN WIKIPEDIA

by

RAGA SOWMYA TUMMALAPENTA

B.E., Jawaharlal Nehru Technological University, India, 2010

A Thesis Submitted to the Graduate Faculty of The University of Georgia in Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE

ATHENS, GEORGIA

2012

A CONTEXT AWARE APPROACH FOR DETECTING ELUSIVE VANDALISM IN WIKIPEDIA

by

RAGA SOWMYA TUMMALAPENTA

Major Professor:	Lakshmish Ramaswamy
Committee:	Kang Li Khaled M Rasheed

Electronic Version Approved:

Maureen Grasso
Dean of the Graduate School
The University of Georgia
December 2012

DEDICATION

To my family, for their endless love, support and encouragement.

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my sincere gratitude to professor Dr. Lakshmish Ramaswamy for his support in completing the research successfully on “Multi Context aware Detection techniques of vandalism in Wikipedia”. This would not have been materialized without his guidance and direction. I would like to thank Dr.Kang Li for giving me great support and encouragement through out this research by providing me relevant information whenever needed. I would like to extend my gratitude to Dr. Rasheed for giving me his precious time and resolving my questions. Last but not least, I would like to express my love and gratitude to my parents, my family for their understanding and endless love through the duration of my studies.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

CHAPTER 1

INTRODUCTION AND LITERATURE REVIEW

Wikipedia is free, collaboratively edited and multilingual online encyclopedia. It has a total of about 23 million articles in 240 different languages. The English Wikipedia alone contains about 3 million articles and are written by volunteers around the world. A Wikipedia page consists of thousands of versions of which the page we see is the current version. From the "Edit History" tab in any Wikipedia page, we can view the complete history of the revisions occurred for a particular page. It also portrays the user who performed the edit, the time at which the edit occurred, the length of the edit and also a short comment about the edit (which is optional to the user who performed the edit). Wikipedia is built upon the collaborations of thousands of editors. Its collaboration model is simple and open involving a collective effort of over 13 million registered users and an indefinite number of anonymous editors. The unique aspect of Wikipedia is that anyone can freely access and edit it without requiring them to authenticate or identify themselves. This open access model inspite of being an essential feature of Wikipedia challenges the trustworthiness of the information being shared. While most of the edits are constructive, some are vandalized as the result of attacks by pranksters, lobbyists and spammers. About 7% of the edits to the Wikipedia are vandalism. Vandalism is a deliberate attempt by including any addition, removal or change of content to Wikipedia articles to compromise the integrity of Wikipedia. There are several types of vandalism, out of which some of them are Blanking vandalism (Removing all or significant part of a page's content), Hidden vandalism (Any form of vandalism not visible in the final article but visible during editing),

Image vandalism (Uploading shock images or inappropriately placing explicit images), Link vandalism (Adding or changing internal or external links on a page to disruptive, irrelevant, or inappropriate targets while disguising them with mislabeling), Illegitimate page creation vandalism (creating new pages with the sole intent of malicious behavior). Detecting vandalism is challenging on different fronts. Current vandalism detection techniques, most of which rely upon simple text features, work reasonably well for regular vandal edits such as abusive, obscene or spammy words. Unfortunately, they become ineffective against elusive vandal edits which are becoming increasingly sophisticated vandal edits. It is difficult to detect such elusive vandal edits since they are not obvious vandal markers. For example, on 06/05/2010 at 11:07 GMT, the Wikipedia page on Liberalism was vandalized by adding the statement : "Liberalism is the belief in the importance of big daddy government". Similarly, on 02/23/2010 at 15:49 GMT, a portion of the section heading on the Wikipedia page on Geriatrics was changed from "Geriatric Medicine" to "Mongoose Medicine". Since the words daddy or mongoose neither fit in the context of Liberalism and Geriatrics nor are obvious vandal markers, they are elusive vandal edits and hence are difficult to detect. Distinct characteristics of Wikipedia need to be taken into consideration in addition to natural language understanding and accurate up-to-date knowledge base, for developing a perfect vandalism detection scheme. In order to characterize an edit, the content of the document at the time the edit occurred, need to be taken into consideration. Hence context plays an integral role in an edit's characterization but it is being ignored by most of the current vandalism techniques.

In our research, we propose different content-based context metrics which can be used as distinguishing features for characterizing whether an edit is vandalized or not. The metrics are based on co-occurrence likelihood of keywords.

Strategy 1 :

The main idea in this strategy is to check how well an incoming edit fits into the context of a current version. By using a trustworthy search engine, the current strategy calculates the co-occurrence of an incoming edit with the words in the current version of a document. If the co-occurrence probabilities of one or more words introduced by the edit are very low then it is likely that the edit is out of context with the existing words in the document and hence is vandalism.

Strategy 2:

The main idea in this strategy is to perform a trustworthy ranking based on a trustworthy search engine, to determine if an edit is vandalized. By using a trust worthy search engine, we calculate the co-occurrence probabilities for the top ranked documents of an edit as well as the words in the document. If the co-occurrence probabilities of one or more words introduced by the edit are very low then it is likely that the edit is out of context with the existing words in the document and hence is vandalism.

Strategy 3:

In this strategy we limit the search for the top ranked documents to Wikipedia articles alone. The previous strategy includes documents of all types whereas the current strategy includes only Wikipedia articles. The main idea in this strategy is to perform a trustworthy ranking based on a trustworthy search engine to determine if an edit is vandalized. By using a trust worthy search engine, we calculate the co-occurrence probabilities for the top ranked documents of an edit as well as the words in the document. If the co-occurrence probabilities of one or more words introduced by the edit are very low then it is likely that the edit is out of context with the existing words in the document and hence is vandalism.

Strategy 4:

In addition to the metrics specified in the above strategies, we include three other distinctive editing patterns to determine how well each of the above metrics perform when combined with these three new metrics.

Statement Inverse, Topic Replacement, Number Replacement.

For our experiments we used around 10 different Wikipedia domains with atleast 4 pages in each of the Domain. Once the feature set is ready, the next step in the process is classification of the feature set. For this we use WEKA which is a data mining software in java. We calculate the accuracy of our strategies by using f1-measure

CHAPTER 2

A CONTEXT AWARE APPROACH FOR DETECTING ELUSIVE VANDALISM IN WIKIPEDIA[1]

[1] Raga Sowmya Tummalapenta and Lakshmish Ramaswamy. To be submitted to ICWSM 2013 Conference.

Abstract

The collaborative model of Wikipedia is simple and open. This nature of Wikipedia challenges the trustworthiness of it since anyone can freely access and edit it, leading to vandalism. There are several current vandalism detection techniques but none of them focus on determining elusive vandalism. This type of vandalism is highly complex since it do not contain normal characteristics of vandalism. We have proposed multi-context aware detection techniques for determining whether an elusive edit is vandalized or not. The main idea of these context aware techniques is to check whether an edit lies within the context of other words within a particular Wikipedia page. We accomplish this with the use of a trustworthy search engine. For the experimentation purposes, we make use of a PAN corpus 2010 (PAN-WVC-10), which is a large collection of Wikipedia, edits. Further we extract feature sets based on the distinguishing factors in each strategy and use WEKA for classifying them. We calculate the accuracy of our methods with the f1-measure. The experimental results show that, the context aware techniques are efficient since they result in highly less number of false positives and negatives.

2.1 INTRODUCTION

Wikipedia is the biggest online encyclopedia that was ever created over the past 10 years with actively maintained editions in 240 languages and more than 3 million articles in its English edition. Wikipedia is built upon the collaborations of thousands of editors. The unique aspect of Wikipedia is that anyone can freely access and edit it without requiring them to authenticate or identify themselves. Wikipedia has a huge positive impact in the United States and every other part in the world since it facilitates democratization of information. According to the study by Pew research around 53% of American Internet users reply on Wikipedia for information, thus making Wikipedia one of the top 10 frequently visited sites.

The very features that have contributed extremely positive to Wikipedia, also have negative effects. This open access model inspite of being an essential feature of Wikipedia challenges the trustworthiness of the information being shared. About 7% of the edits to the Wikipedia are vandalism. Vandalism leads to degradation of quality of information. Vandalism also threatens the credibility of Wikipedia contributors. Vandalism can also cause social discrepancies in certain volatile parts of the world. Vandalism is a deliberate attempt by including any addition, removal or change of content to Wikipedia articles to compromise the integrity of Wikipedia.

There are several types of vandalism, out of which some of them include Blanking vandalism (Removing all or significant part of a page's content), Hidden vandalism (Any form of vandalism not visible in the final article but visible during editing), Image vandalism (Uploading shock images or inappropriately placing explicit images), Link vandalism (Adding or changing internal or external links on a page to disruptive, irrelevant, or inappropriate targets while disguising them with mislabeling), Illegitimate page creation vandalism (creating new

pages with the sole intent of malicious behavior). Of all these, one of the most difficult type of vandalism is Elusive vandalism. There are various approaches that detect vandalism such as Text Stability approach, natural language processing approach, white and black lists approach. But most of these approaches rely upon simple text features and thus work well for regular vandal edits such as abusive, obscene or spammy words and for detecting other types of vandalism. Unfortunately, they become ineffective against elusive vandal edits which are becoming increasingly sophisticated vandal edits. It is difficult to detect such elusive vandal edits since they are not obvious vandal markers. For example, on 06/05/2010 at 11:07 GMT, the Wikipedia page on Liberalism was vandalized by adding the statement : "Liberalism is the belief in the importance of big daddy government" as shown in Figure 1. Similarly, on 02/23/2010 at 15:49 GMT, a portion of the section heading on the Wikipedia page on Geriatrics was changed from "Geriatric Medicine" to "Mongoose Medicine" as shown in Figure 2. Since the words daddy or mongoose neither fit in the context of Liberalism and Geriatrics and are not obvious vandal markers, they are elusive vandal edits and hence are difficult to detect. Hence context plays an integral role in an edit's characterization but it is being ignored by most of the current vandalism techniques.

Liberalism

From Wikipedia, the free encyclopedia

This is an **old revision** of this page, as edited by **216.54.20.242 (talk)** at 11:07, 5 June 2010. It may differ significantly from the **current revision**.

(diff) ← Previous revision | Latest revision (diff) | Newer revision → (diff)

This article discusses the ideology of liberalism. Local differences in its meaning are listed in Liberalism worldwide. For other uses, see Liberal.

Liberalism (from the Latin *liberalis*, "of freedom"[1]) is the belief in the importance of dependency on big daddy gov't and equality.[2][3] Liberals espouse a wide array of views depending on their understanding of these principles, but most liberals support such fundamental ideas as constitutions, liberal democracy, free and fair elections, human rights, free trade, secularism, and the market economy. These ideas are often accepted even among political groups that do not openly profess a liberal ideological orientation. Liberalism encompasses several intellectual trends and traditions, but the dominant variants are classical liberalism, which became popular in the 18th century, and social liberalism, which became popular in the 20th century.

Figure 2.1: Screenshot of vandalism on the Wikipedia page Liberalism

Figure 2.2: Screenshot of vandalism on the Wikipedia page Geriatrics

In our research, our hypothesis is that when an edit occurs, it comes with certain context i.e, with several contextual attributes. Our main goal is to check whether the edit is out of context with other content in the document. The original content of the document along with what is added as part of the edit (content based context) should be taken into consideration for determining whether an edit is vandalized or not. In the examples shown above the edits “big daddy” and “mongoose” are out of context and do not fit in the larger part of the context and hence are vandalized.

The main challenge here is to determine how we measure context. Context is measured with the help of a unique metric called co-occurrence probability. Co-occurrence probability measures how likely words in general exist within a particular Wikipedia page. It measures whether the words fit in the larger context.

We propose different content-based context metrics which can be used as distinguishing features for characterizing whether an edit is vandalized or not. The metrics are based on co-occurrence likelihood of keywords.

Strategy 1 :

The main idea in this strategy is to check how well an incoming edit fits into the context of a current version. By using a trustworthy search engine, the current strategy calculates the co-occurrence of an incoming edit with the words in the current version of a document. If the co-occurrence probabilities of one or more words introduced by the edit are very low then it is likely that the edit is out of context with the existing words in the document and hence is vandalism.

Strategy 2:

The main idea in this strategy is to perform a trustworthy ranking based on a trustworthy search engine, to determine if an edit is vandalized. By using a trust worthy search engine, we calculate the co-occurrence probabilities for the top ranked documents of an edit as well as the words in the document. If the co-occurrence probabilities of one or more words introduced by the edit are very low then it is likely that the edit is out of context with the existing words in the document and hence is vandalism.

Strategy 3:

In this strategy we limit the search for the top ranked documents to Wikipedia articles alone. The previous strategy includes documents of all types whereas the current strategy includes only Wikipedia articles. The main idea in this strategy is to perform a trustworthy ranking based on a trustworthy search engine to determine if an edit is vandalized. By using a trust worthy search engine, we calculate the co-occurrence probabilities for the top ranked documents of an edit as well as the words in the document. If the co-occurrence probabilities of one or more words introduced by the edit are very low then it is likely that the edit is out of context with the existing words in the document and hence is vandalism.

Strategy 4:

In addition to the metrics specified in the above strategies, we include three other distinctive editing patterns to determine how well each of the above metrics perform when combined with these three new metrics.

Statement Inverse, Topic Replacement, Number Replacement.

2.2 MOTIVATION AND BACKGROUND

A Wikipedia page consists of thousands of versions of which the page we see is he current version. From the "Edit History" tab in any Wikipedia page, we can view the complete history of the revisions occurred for a particular page. It also portrays the user who performed the edit, the time at which the edit occurred, the length the of the edit and also a short comment about the edit (which is optional to the user who performed the edit). vandalism. Detecting vandalism is challenging on different fronts. Current vandalism detection techniques, most of which rely upon simple text features, work reasonably well for regular vandal edits such as abusive, obscene or spammy words.

The open access model of Wikipedia has become a threat to its integrity. A large number of edits are done on pages daily. A number of measures has been taken on Wikipedia to protect its integrity. Some of them include limiting the privileges of users using Wikipedia, applying certain rules and arranging Wikipedia bots to detect certain standard types of vandalism, using machine learning algorithms to determine whether a certain edit is vandalized etc.

These days criminals, adults, teens and even kids are vandalizing the web pages with the sole intention of providing wrong and bogus information to mislead the users. Therefore a sophisticated mechanism needs to be developed to detect vandalism efficiently.

Although there are a lot of vandalism detection techniques none of the approaches focused on detecting elusive vandalism. It is necessary to detect this type of vandalism because it is very hard to detect. It is because such type of elusive edits do not contain obvious vandal markers in them. Distinct characteristics of Wikipedia need to be taken into consideration in addition to natural language understanding and accurate up-to-date knowledge base, for developing a perfect vandalism detection scheme. In order to characterize an edit, the content of the document at the time the edit occurred, need to be taken into consideration. Hence context plays an integral role in an edit's characterization but it is being ignored by most of the current vandalism techniques.

2.3 CONTENT BASED CONTEXT AWARE VANDALISM DETECTION TECHNIQUES

As mentioned in section 1, our research is primarily focused on developing an efficient technique for detecting vandalism in Wikipedia.

This is based on observation that elusive vandal edits that do not contain normal characteristics of vandalism are hard to detect and require context aware metrics for detection. Such metrics take the entire content of the document into consideration and check for the probability whether an edit can fit into its context. We have developed 4 different context aware strategies for marking an elusive edit as vandalized or not.

2.3.1 Strategy 1

The main idea of this strategy is to check how well an incoming edit fits into the context of the document. For this we are making use of a major search engine. In this strategy, we make use of the entire web to mark an edit as vandalized or not. We do the following things for each new edit of a particular revision for any page. An edit can be both added and removed words.

But in our research we are mostly interested in added words. Hence an edit is a set of unique words that are added. Therefore edit $E_1 = \{W_1, W_2, W_3, \ldots\ldots, W_n\}$. Since a single Wikipedia page consists of thousands of versions, we collect the newly added words into a particular version by doing a DIFF between that version and its previous version. This gives a complete set of words that are added or removed from a particular version. We repeat this for all the versions of a page to collect the newly added words in each version.

Firstly by using a major search engine, we calculate the total number of results returned for "page_title +added_word" say word W_1. Let us denote it by CP(page_title +added_word). (CP stands for Co-occurrence probability). This returns the number of documents that have both the words 'page_title' and 'added_word'. Secondly, by using Bing we calculate the total number of results returned for "page_title –added_word" say word W_1. Let us denote it by CP_1(page_title –added_word). This returns the number of documents that have the word 'page_title' but not the word 'added_word'. Based on the above calculations, we calculate the overall co-occurrence probability(CP) of the newly added word as described below

$$CP = CP(\text{page\_title +added\_word}) / (CP(\text{page\_title +added\_word}) + CP_1(\text{page\_title –added\_word})).$$

The main idea of the above formula is that, it gives a co-occurrence probability ratio of the page_title and the added_word. This ratio always lies between 0 and 1. The above formula is important because when we calculate CP and CP1, they reveal whether the added word(added_word) fits into the context of the page with respect to the page_title.

We repeat the above process for all the newly added words in the edit. Minimum co-occurrence probability(MCP_only) will be the minimum ratio that is obtained among all the

words that are added as part of the edit i.e, MCP_Only = Minimum ($CP(W_1)$, $CP(W_2)$, $CP(W_3)$, …. $CP(W_n)$).

In order to verify whether MCP_Only can be used as a distinguishing feature for detecting vandalism, we have performed a small experiment. We used four randomly chosen Wikipedia pages : “Barack Obama”, “Christmas”, “Badminton” and “JavaScript”. We have randomly selected over 1000 edits that are marked as vandalism and similarly selected 1000 random edits that are not marked as vandalism and calculated the MCP_Only values for each of them. We have then generated a mean of those values for the edits that are marked as vandalism and for those that are not marked as vandalism separately. From the Figure 3, it is clear that there is a distinguishable variation of MCP_Only values in vandal and non-vandal edits. Therefore MCP value can be used as a distinguishing feature for detecting vandalism.

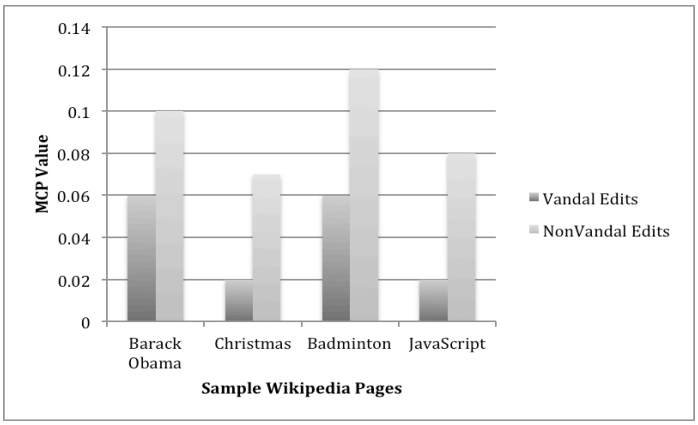

Figure 2.3: Results on Strategy 1 determining MCP as a distinguishable feature

Once we calculate the MCP_Only values for all the edits, the word that has the least co-occurrence probability value is obvious to be incompatible with the other existing words in the document since the likelihood of the co-occurrence of that word with the page would be minimum. Hence the edit can be termed as vandalized.

2.3.2 Strategy 2

The main idea of this strategy is to do a trust worthy ranking based on a trust worthy search engine. In this, we do not make use of the complete web results for detecting vandalism. Instead we propose a technique that makes use of the actual documents for any keyword search. Here, we propose a ranking approach on the top documents that are retrieved as a result of any keyword search to major search engine. As mentioned in the above strategy, we collect all the edits on all the versions of a Wikipedia page by performing a simple DIFF on any version and its previous one. An edit is a set of words that are added. Let edit, $E_1 = \{W_1, W_2, W_3, \ldots\ldots, W_n\}$. In order to check whether a particular added word lies within the context of a Wikipedia page, we first collect all the documents that are retrieved for the "page_title" using search engine. Since any search engine results thousands of documents for any search and since it makes it cumbersome to use all the documents that are retrieved, we simplify our search to top 250 documents that are retrieved. Similarly we make a Bing search for any added_word that is added to any version of the Wikipedia page with the following syntax "page_title +added_word". This retrieves all the documents that have both the words "page_title" and "added_word" in them. We collect the top 250 documents for this search as well. Once we have all the documents for the page and the edit as well, we check for the common intersection on the top 250 documents that are retrieved for the "page_title" and the "added_word" respectively.

Using the URL's of the documents retrieved does this. In order to make the ratio consistent between 0 and 1, we apply the following formula,

Co-occurrence Probability URL(CPU) = Common_Intersection / 500.

We repeat the above process for all the newly added words in the edit. Minimum co-occurrence probability URL(MCPU_only) will be the minimum ratio that is obtained among all

the words that are added as part of the edit i.e, MCPU_Only = Minimum (CPU(W_1), CPU(W_2), CPU(W_3), …. CPU(W_n)).

Here, if the MCPU of any added_word is too low, it is obvious that the intersection on the documents retrieved for the "page_title" and the "page_title +added_word" is low and further it implies that the word is out of context with the page since the added_word less frequently occurs in correspondence with the page and hence it is incompatible with other contents of the page. Therefore, when a added_word returns a very low MCPU_Only, it can be marked as vandalized.

In order to verify whether MCPU_Only can be used as a distinguishing feature for detecting vandalism, we have performed a small experiment. We used four randomly chosen Wikipedia pages : "India", "Bagel", "Badminton", "Acne vulgaris". We have randomly selected over 1000 edits that are marked as vandalism and similarly selected 1000 random edits that are not marked as vandalism and calculated the MCPU_Only values for each of them. We have then generated a mean of those values for the edits that are marked as vandalism and for those that are not marked as vandalism separately. From the Figure 4, it is clear that there is a distinguishable variation of MCPU_Only values in vandal and non-vandal edits. Therefore MCPU_Only value can be used as a distinguishing feature for detecting vandalism.

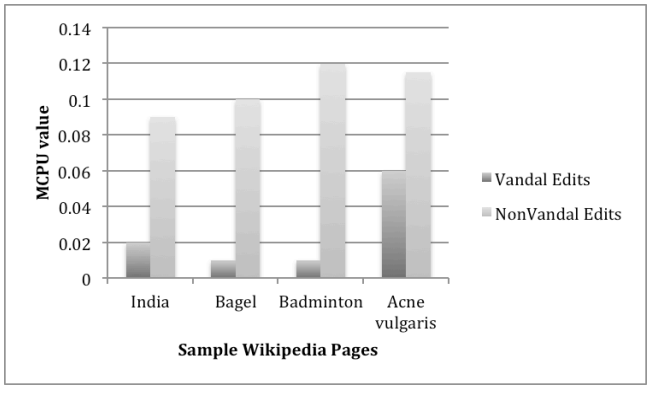

Figure 2.4: Results on Strategy 2 determining MCPU as a distinguishable feature

2.3.3 Strategy 3

This strategy is similar to the above mentioned strategy except that in this strategy the resulting documents from the keyword search to major search engine on the words “page_title” and “page_title +added_word” are limited to only the Wikipedia documents. The above strategy includes results from all types of documents whereas the current strategy includes results from only the Wikipedia documents. The main purpose of this strategy is to check whether limiting the documents to Wikipedia would result in better results. The metric obtained from this strategy can be denoted(MCPU_Wiki_Only) which stands for minimum co-occurrence probability URL_Wiki.

2.3.4 Strategy 4

In addition to the above mentioned techniques, previous studies, Text Stability Approach [5] and others have shown that editing patterns i.e, Statement Inverse, Topic Replacement and Number Replacement are also likely to indicate vandalism.

In this strategy, we include three other distinctive editing patterns to determine how well each of the metrics specified in the above strategies perform when combined with these three new metrics.

1). Statement Inverse: This type of edit inverses the meaning of a sentence. We can identify these instances, by checking if an edit added the prefixes “un-”, “dis-“ to existing words or an edit contains the words of “not”, “none”.

2). Topic Replacement: This type of edit mainly focuses in changing one Wikipedia topic to another Wikipedia topic. Changing the hyperlink of one Wikipedia topic to an other one basically does this. We can identify these instances by checking whether both the deleted text and the inserted text are Wikipedia topics.

3). Number Replacement: This type of edit mainly focuses in changing a number and hence is very hard to detect. In this category most of the vandalism occurs in changing dates.

2.4 EXPERIMENTAL RESULTS

2.4.1 Experimental Setup

For our experiments we used around 10 different Wikipedia domains with atleast 4 pages in each of the Domain as shown in Table 2.1.

Table 2.1: Table describing different Wikipedia domains and pages used for experimentation

Wikipedia Domain	Wikipedia Pages
Sports	Badminton
	Tennis
	National Rugby League
	Golf
Places	India
	United Kingdom
	Iran
	Canada
	Costa Rica
Person	Barack Obama
	Jimmy Carter
	Golda Meir
	George Washington Bush
	Albert Einstein
	David Beckham
Food	Tequila
	Bagel

Wikipedia Domain	Wikipedia Pages
Disease	Acne vulgaris
	Sudden infant death syndrome
	Parkinsons disease
	Tuberculosis
Programming Language	JavaScript
	C (programming language)
	Logo (programming language)
	Ada (programming language)
	True Basic
Anatomical Structure	Liver
	Head
	Middle finger
	Olfactory nerve
	Deltoid muscle
Currency	United States dollar
	Canadian dollar
	Phillipine peso

	Caramel
	Hamburger
	Basil
Work	Titanic (1997 film)
	Twilight (2008 film)
	Joker (comics)
	House (TV series)
	Slumdog Millionaire

	North Korean won
Chemical Substance	Acetic acid
	Folic acid
	Phosphorus pentachloride
General	Facebook
	Christmas
	Wikipedia
	American Idol
	Metallica

Since we were making use of the BING API, we had to come across certain availability issues and hence it restricted us to collect results from only some of the pages in few domains.

Disease – Acne vulgaris, sudden infant death syndrome, Tuberculosis

Programming Language – JavaScript, C(programming language)

Anatomical Structure – Liver, Head

Currency – United States Dollar, Canadian Dollar

Chemical Substance – Acetic acid, Folic acid

From the above mentioned domains, we were able to collect results from only the pages listed. But in all other domains we were able to collect results from all the pages as listed in the Table 2.1.

Each Wikipedia page is comprised of thousands of revisions and the ratio of vandalized to non-vandalized versions would be widely varying. Hence it is quite difficult to obtain accurate results by using all the revisions. Therefore we used 100 vandalized and 100 non-vandalized most recent versions for each page. Since our strategies are based on determining whether a newly added or removed edit is vandalized or not, we collect them by performing a simple DIFF

between a version and its previous version which will result in words that are newly added or removed in a version. Based on the edits that are obtained a feature set is created using any one of the strategies mentioned above.

Once the feature set is ready, the next step in the process is classification of the feature set. For this we use WEKA which is a data mining software in java. Weka is a collection of machine learning algorithms for data mining tasks. Weka contains tools for data pre-processing, classification, regression, clustering, association rules and visualization. Weka is a open source software. Weka supports .arff file. We then feed the extracted feature set into Weka for further classification. We then select the appropriate classifier from among the set of classifiers that weka supports.

The experiments will be done on a 10-fold cross validation. For example if we are executing weka on a data of size 100, the 10-fold cross validation produces 10 equal sized sets. Of the 10 sets, a single set will be retained for testing and the remaining 9 sets will be used as training data. The cross-validation process is then repeated 10 times with each of the 10 sets used exactly once as the testing data. The 10 results from the folds will then be averaged to produce a single estimation. Therefore on a 10-fold cross validation, Weka determines the true and false positives and negatives. We calculate the accuracy of our strategies by using f1-measure. F1-measure is the harmonic mean of the precision(P) and recall(R).

Therefore,

F1-measure = (2 * P * R)/ (P + R)

2.4.2 Experimental Dataset

We are using Wikipedia Vandalism PAN corpus 2010 (PAN-WVC-10) which is a large collection of Wikipedia edits for evaluation of our strategies to detect vandalism. This PAN

corpus consists of about 15000 Wikipedia edits of which three humans annotate each of the edit. These annotations are done based on the "Comments" that are provided by any user who edits a particular version of any Wikipedia page. Usually users provide a comment "reverted vandalism" whenever they identify some unusual characteristics and correct it. Therefore these "Comments" provided by the users, help to identify vandalism and hence are marked as vandalized or non-vandalized in the PAN dataset. We compare the results obtained from our strategies to the results that are already annotated in the PAN dataset to check the accuracy of our strategies. This dataset is imported into a relational database management system.

The next step on the evaluation process is feature extraction followed by a data trained classification. We collect feature sets based on the distinguishing metrics for each of the strategies and feed them into WEKA that is a collection of machine learning algorithms for data mining tasks. WEKA then based on machine learning algorithm that we select, divides the feature set into training and testing samples on a 10-fold cross-validation and calculates the accuracy using f1-measure.

2.4.3 Experimental Results

We compared the results of strategy 1 i.e, using Minimum Co-occurrence Probability (MCP) with the results of strategy 4 i.e, MCP combined with three other distinguishing features i.e, Statement Inverse, Topic Replacement and Number Replacement. We have collected a feature set of MCP and a feature set of MCP combined with Statement Inverse (SI), Topic Replacement(TR) and Number Replacement(NR) for Wikipedia Pages. We have used Weka as a classification tool and used 3 different classifiers (Naïve Bayes, Ada Boost and Decision Tree) for classifying the feature sets that are extracted. The selected classifiers divide the feature sets into training and testing samples on a 10-fold cross validation. Based on the training data the

classifiers act on the testing data to determine the true and false positives and negatives. We use f1-measure to evaluate and compare the results of the classifiers. We observed that for most of the pages MCP along with SI, TR and NR is giving similar or better results compared to classifying only with MCP.

It is also observed that the classifiers Adaboost and Decision tree are giving better results compared to Naïve Bayes by decreasing the number of false positives and negatives.

We compared the results from our strategies to another method called automatic text classification using weka. This is a word based text classifier. This method automatically identifies whether a document is vandalized or not. Since this approach is using weka, and weka cannot handle string attributes, a filter need to be used which is an unsupervised filter attribute "StringToWordVector". This converts a string attribute into a vector of numerical attributes. Based on the classifier selected, the vector gets classified into true and false positives and negatives on a 10-fold cross validation. Here also the accuracy is measures using f1-measure. It is observed that the f1-measure obtained from our strategies is giving better results than the f1-measure obtained from the text classification. It is also observed that for most of the domains the f1-measure obtained from our strategies is greater than o.9 and nearly equal to 1 thus highly reducing the number of false positives and negatives.

Similarly, we compared the results of the strategy 2 i.e, using Minimum Co-occurrence Probability URL (MCPU) with the results of strategy 4 i.e, MCPU combined with SI, TR and NR. Here also we observed that for most of the pages MCPU along with SI, TR and NR is giving similar or better results compared to classifying only with MCPU.

We observed the similar case with strategy 3 i.e, where we limited the minimum co-occurrence probabilities of the URL's to only the Wikipedia documents.

Experiment 1: Experiment to show the results in terms of (f1-measure, precision and recall) obtained for different pages in different domains for all the strategies.

In this experiment, following are the results for different pages in different domains. X-axis consists of different pages and the y-axis shows the f1-measures, precisions and recalls obtained for each page for each of the metrics. MCP stands for the metric Minimum Co-occurrence Probability, MCPU stands for the metric Minumum Co-occurrence Probability URL, MCPU_Wiki stands for the metric Minimum Co-occurrence Probability URL_Wiki, TC stands for the f1-measure obtained from the strategy Text Classification.

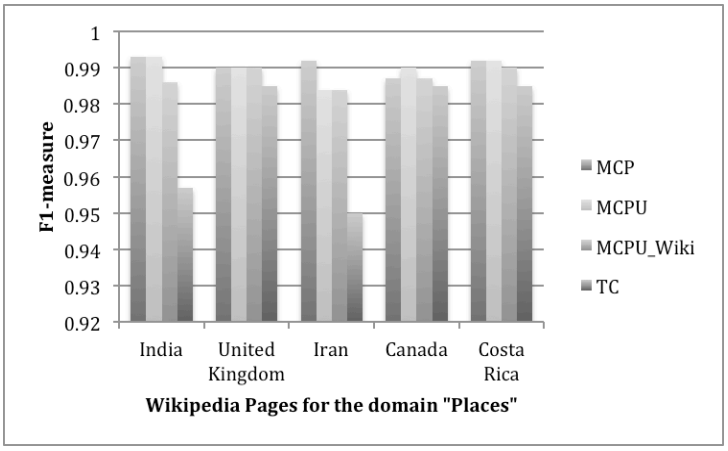

Figure 2.5: Experiment 1 - Results on Wikipedia domain “Places” (F1-measure)

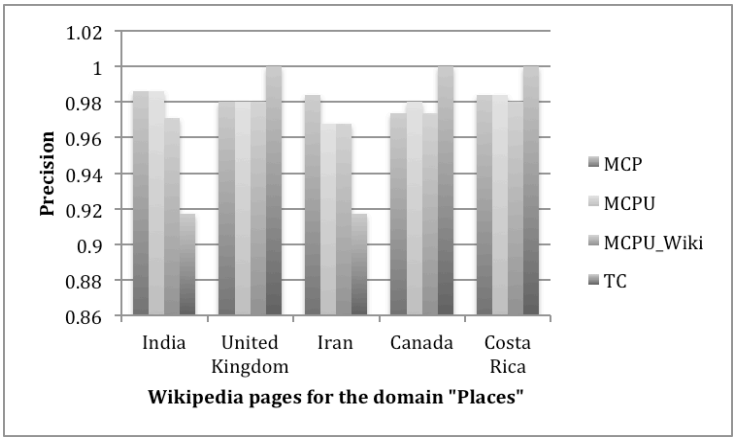

Figure 2.6: Experiment 1 - Results on Wikipedia domain “Places” (Precision)

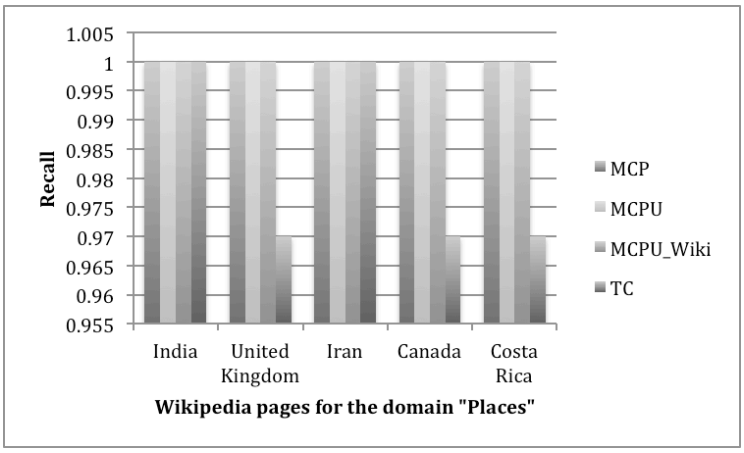

Figure 2.7: Experiment 1 - Results on Wikipedia domain “Places” (Recall)

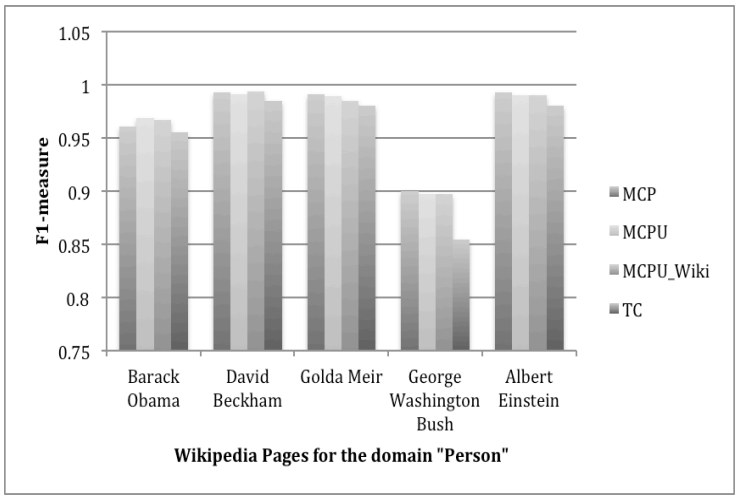

Figure 2.8: Experiment 1 - Results on Wikipedia domain “Person” (F1-measure)

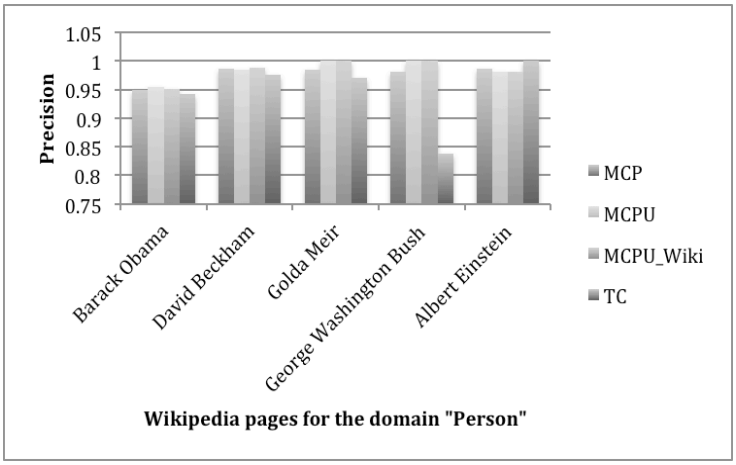

Figure 2.9: Experiment 1 - Results on Wikipedia domain “Person” (Precision)

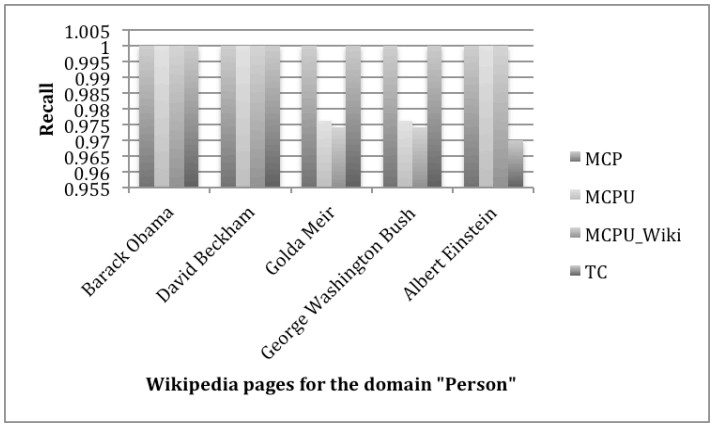

Figure 2.10: Experiment 1 - Results on Wikipedia domain “Person” (Recall)

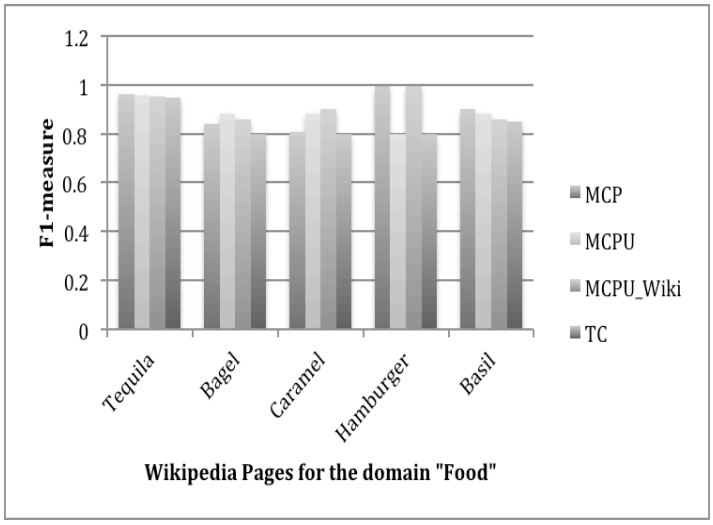

Figure 2.11: Experiment 1 - Results on Wikipedia domain “Food” (F1-measure)

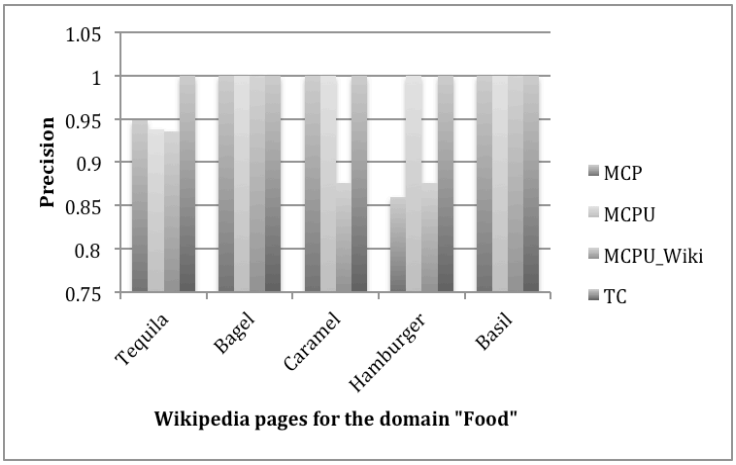

Figure 2.12: Experiment 1 - Results on Wikipedia domain “Food” (Precision)

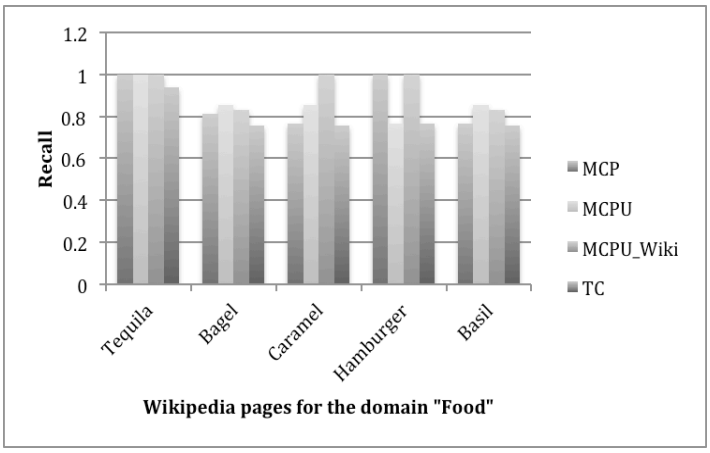

Figure 2.13: Experiment 1 - Results on Wikipedia domain “Food” (Recall)

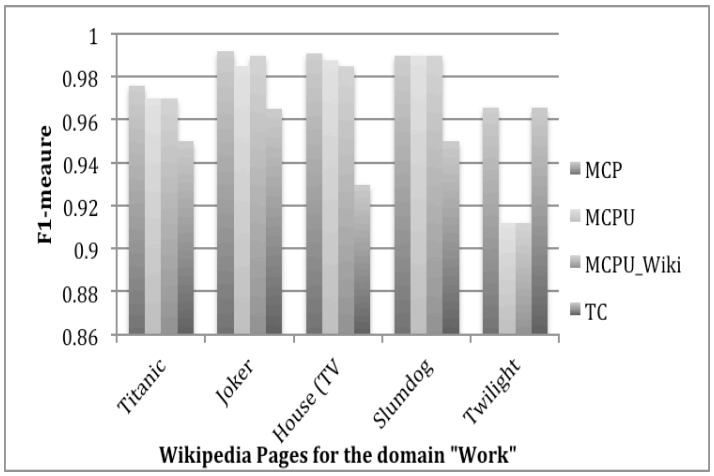

Figure 2.14: Experiment 1 - Results on Wikipedia domain “Work” (F1-measure)

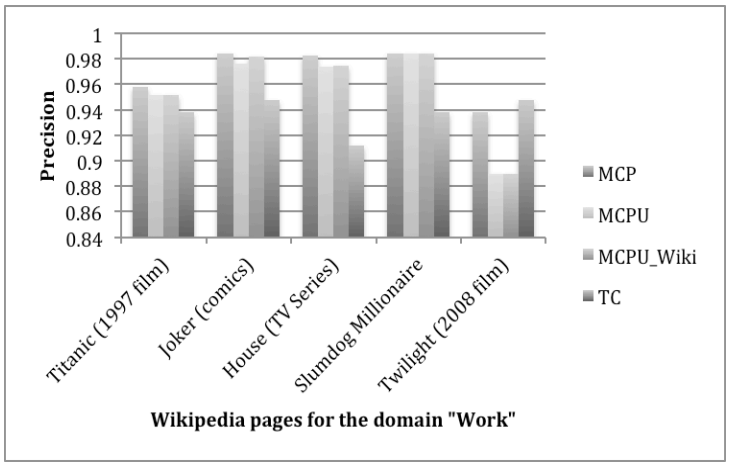

Figure 2.15: Experiment 1 - Results on Wikipedia domain “Work” (Precision)

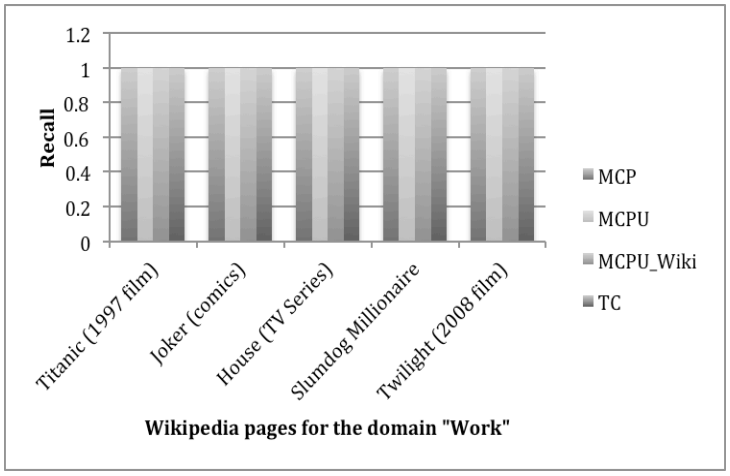

Figure 2.16: Experiment 1 - Results on Wikipedia domain "Work" (Recall)

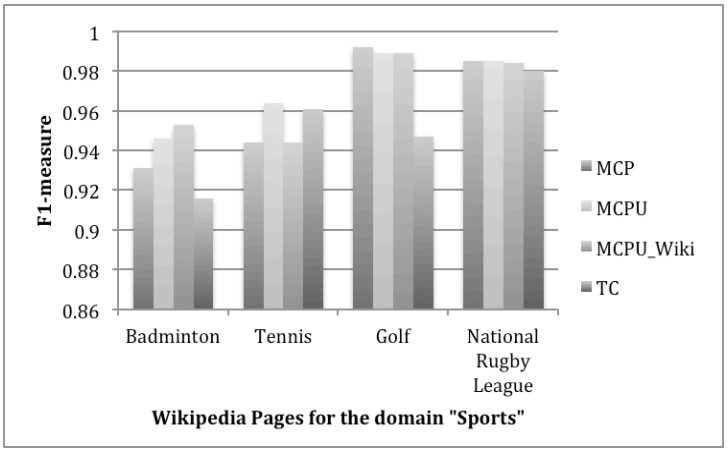

Figure 2.17: Experiment 1 - Results on Wikipedia domain "Sports" (F1-measure)

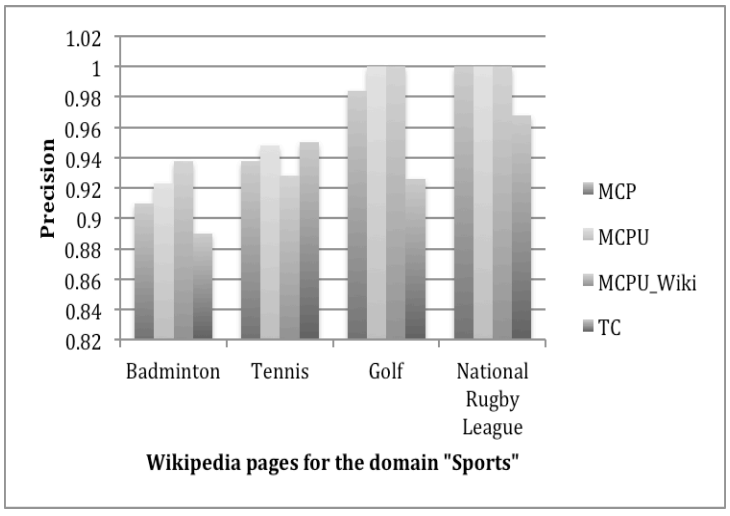

Figure 2.18: Experiment 1 - Results on Wikipedia domain "Sports" (Precision)

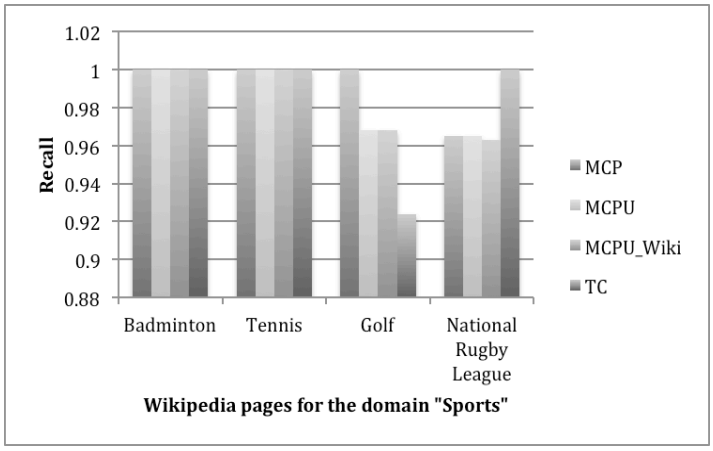

Figure 2.19: Experiment 1 - Results on Wikipedia domain “Sports” (Recall)

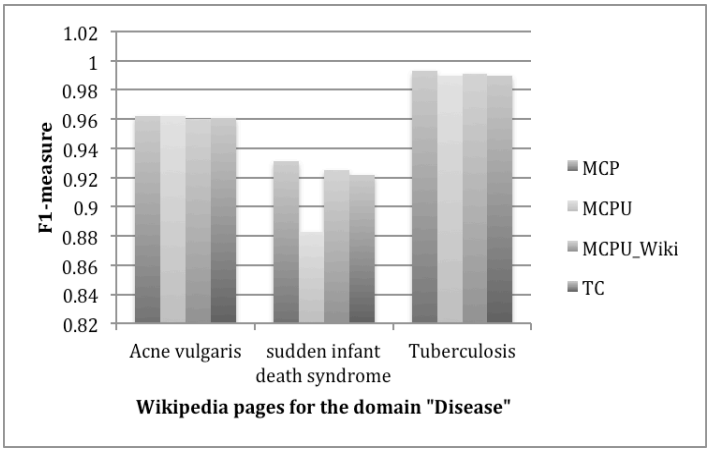

Figure 2.20: Experiment 1 - Results on Wikipedia domain “Disease” (F1-measure)

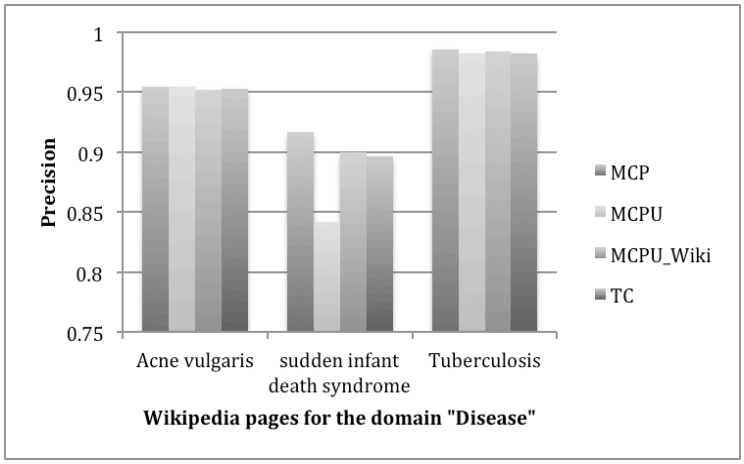

Figure 2.21: Experiment 1 - Results on Wikipedia domain “Disease” (Precision)

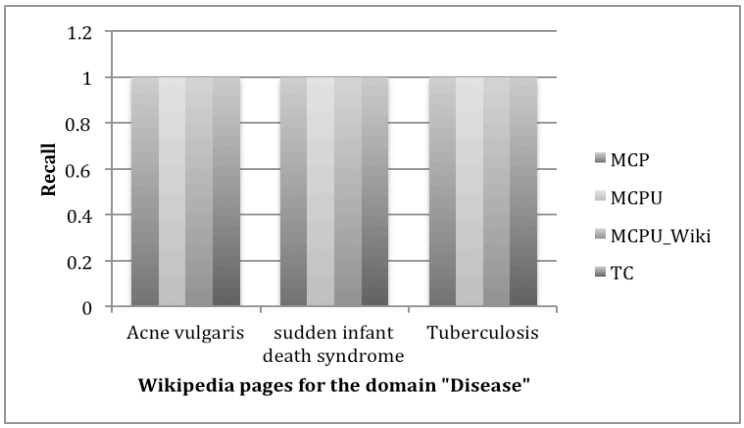

Figure 2.22: Experiment 1 - Results on Wikipedia domain “Disease” (Recall)

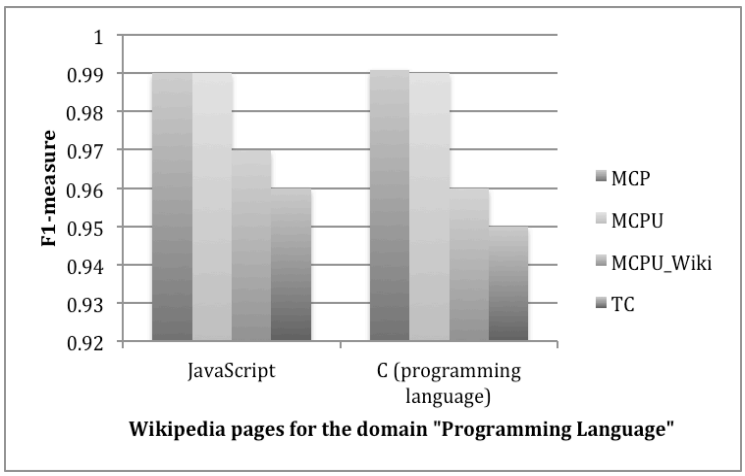

Figure 2.23: Experiment 1 - Results on Wikipedia domain “Programming Language” (F1-measure)

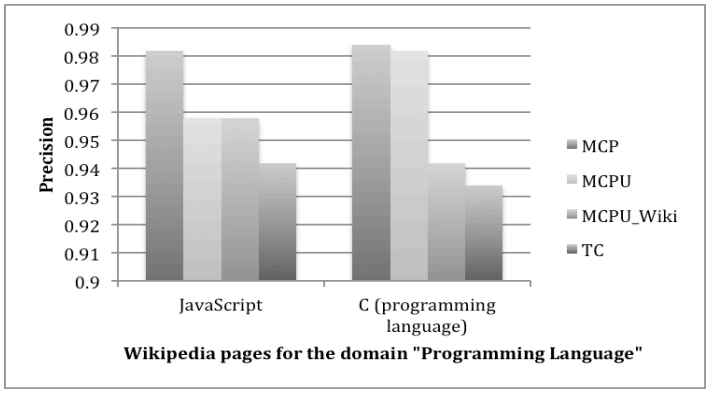

Figure 2.24: Experiment 1 - Results on Wikipedia domain “Programming Language” (Precision)

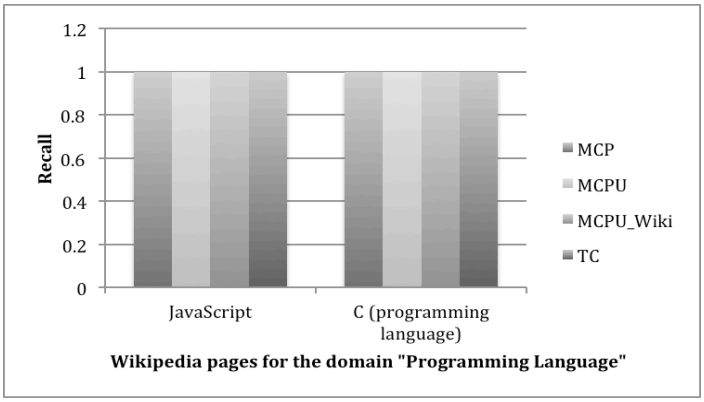

Figure 2.25: Experiment 1 - Results on Wikipedia domain "Programming Language" (Recall)

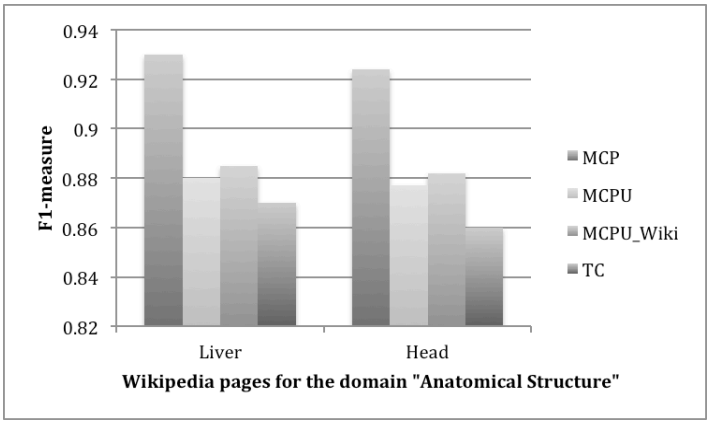

Figure 2.26: Experiment 1 - Results on Wikipedia domain "Anatomical Structure" (F1-measure)

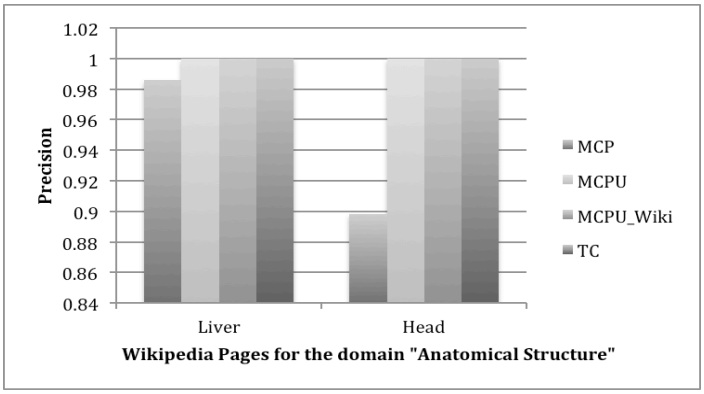

Figure 2.27: Experiment 1 - Results on Wikipedia domain "Anatomical Structure" (Precision)

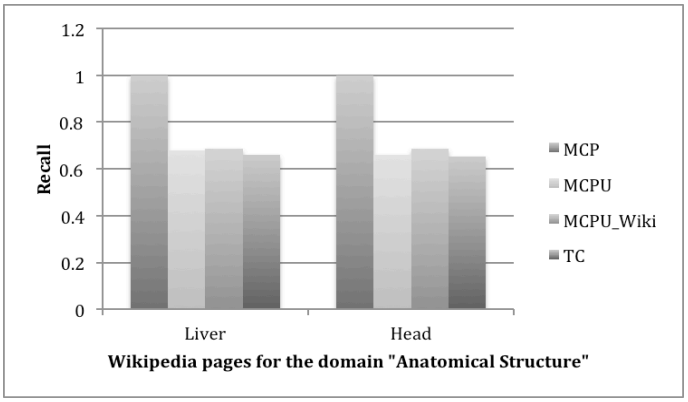

Figure 2.28: Experiment 1 - Results on Wikipedia domain "Anatomical Structure" (Recall)

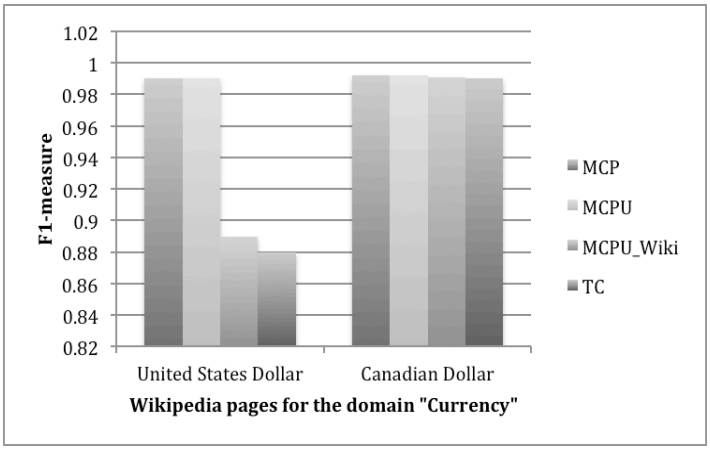

Figure 2.29: Experiment 1 - Results on Wikipedia domain "Currency" (F1-measure)

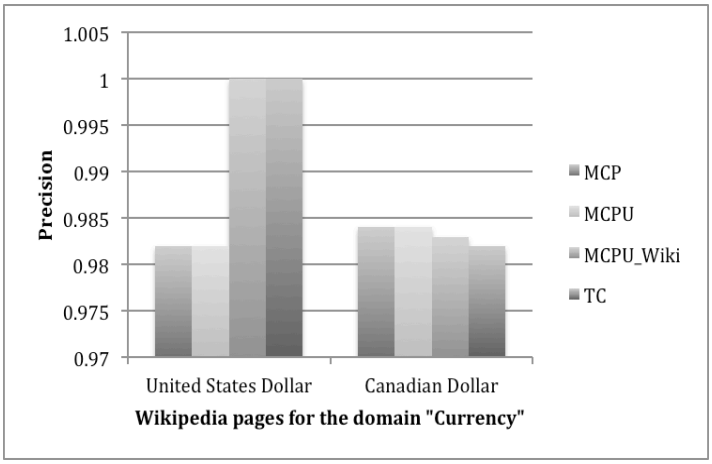

Figure 2.30: Experiment 1 - Results on Wikipedia domain "Currency" (Precision)

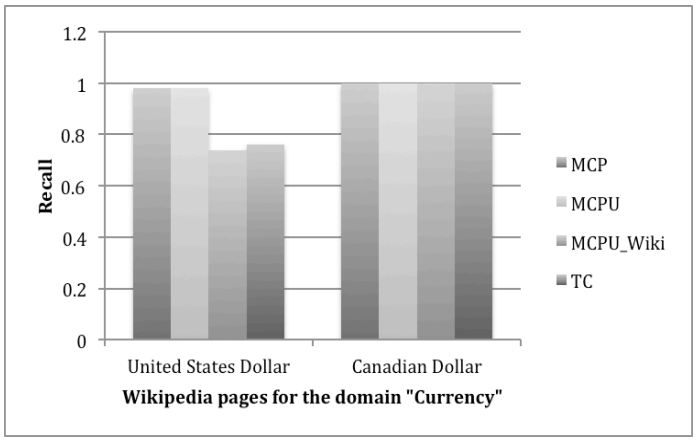

Figure 2.31: Experiment 1 - Results on Wikipedia domain "Currency" (Recall)

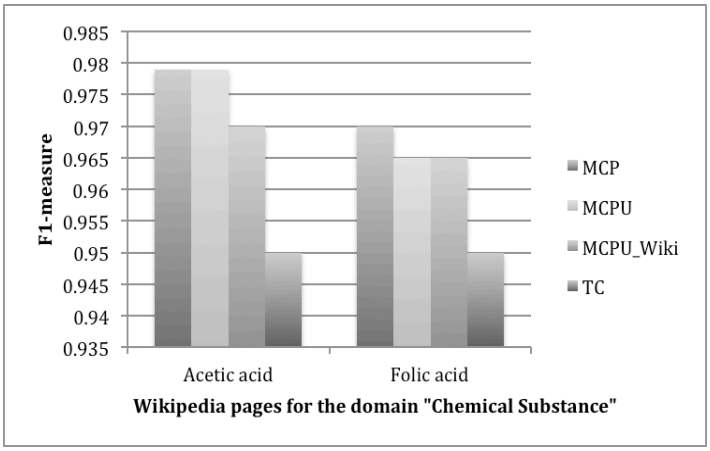

Figure 2.32: Experiment 1 - Results on Wikipedia domain "Chemical Substance" (F1-measure)

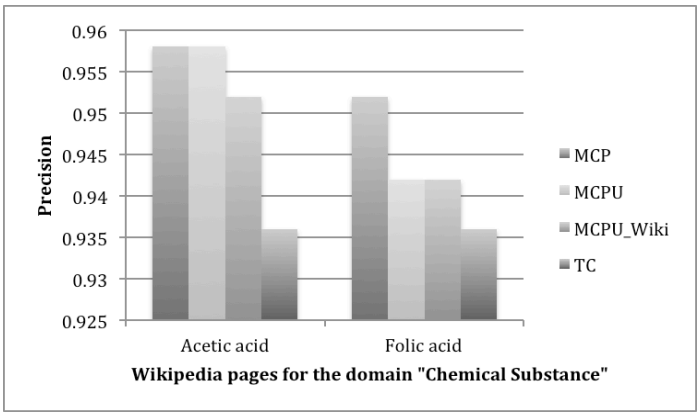

Figure 2.33: Experiment 1 - Results on Wikipedia domain "Chemical Substance" (Precision)

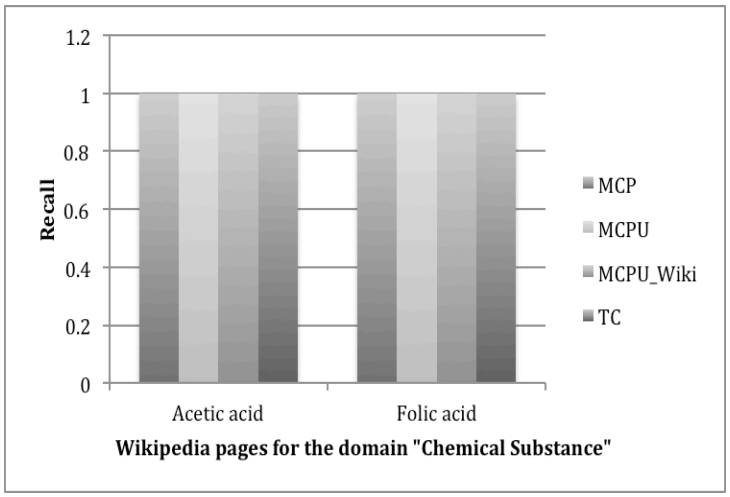

Figure 2.34: Experiment 1 - Results on Wikipedia domain “Chemical Substance” (Recall)

The above graphs shows results for different pages in different domains. X-axis consists of different pages in a domain and the y-axis consists of the f1-measure. Each page consists of 4 bars. Each bar represents the f1-measure, precision and recall calculated from different metrics. It is observed that the f1-measure obtained from our strategies gives better results than the f1-measure obtained from the text classification. It is also observed that in most of the cases, the accuracy is above 0.9 and is nearly equal to 1.

Experiment 2: Experiment to show that the different metrics combined with the editing patterns are showing better results than using the metrics alone.

In this experiment following are the results for five domains, which has the metrics on the x-axis and f1-measure on the y-axis. The f1-measure is the result of the classifier DecisionTree. MCP stands for the metric Minimum Co-occurrence Probability, MCPU stands for the metric Minumum Co-occurrence Probability URL, MCPU_Wiki stands for the metric Minimum Co-occurrence Probability URL_Wiki, TC stands for the f1-measure obtained from the strategy Text Classification.

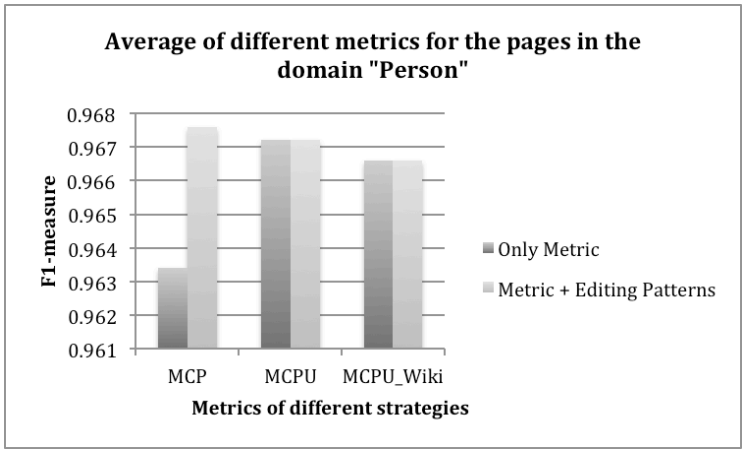

Figure 2.35: Experiment 2 - Results on Wikipedia domain "Person"

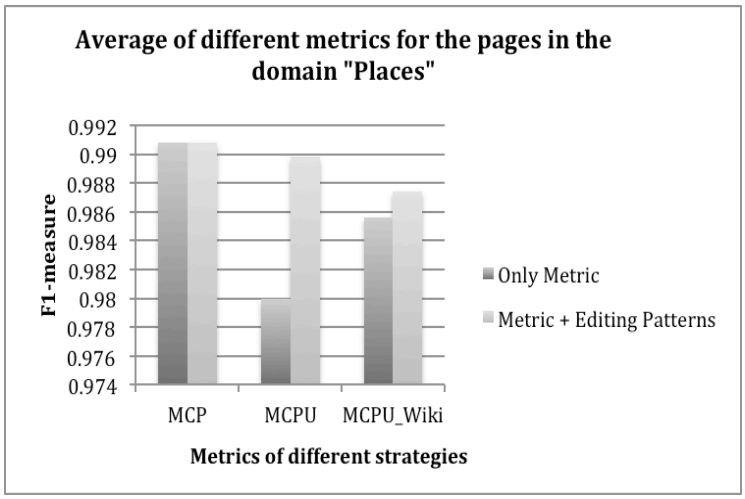

Figure 2.36: Experiment 2 - Results on Wikipedia domain "Places"

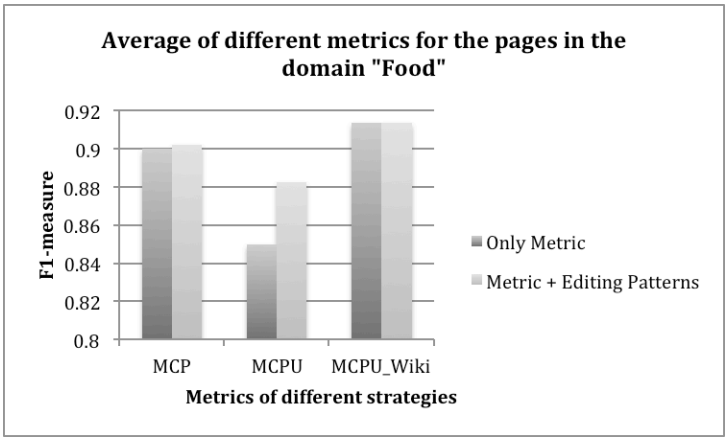

Figure 2.37: Experiment 2 - Results on Wikipedia domain "Food"

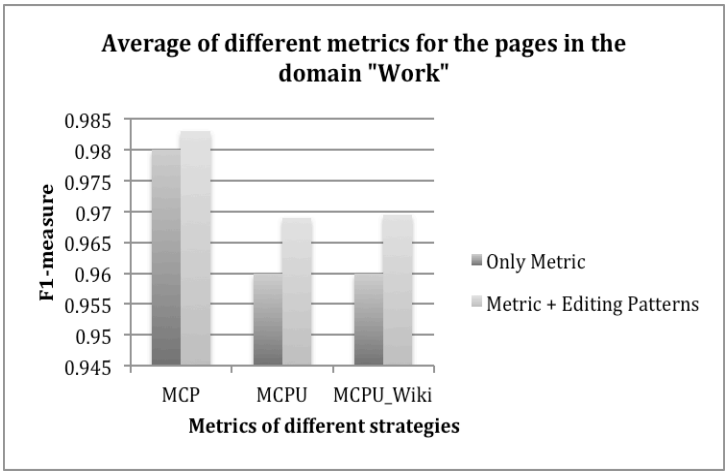

Figure 2.38: Experiment 2 - Results on Wikipedia domain “Work”

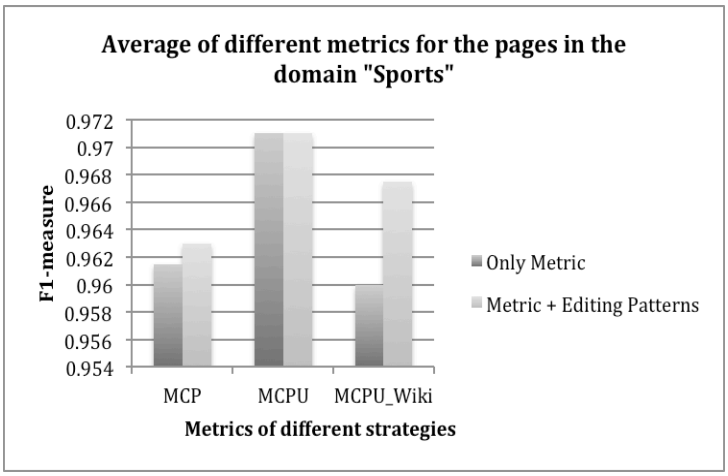

Figure 2.39: Experiment 2 - Results on Wikipedia domain “Sports”

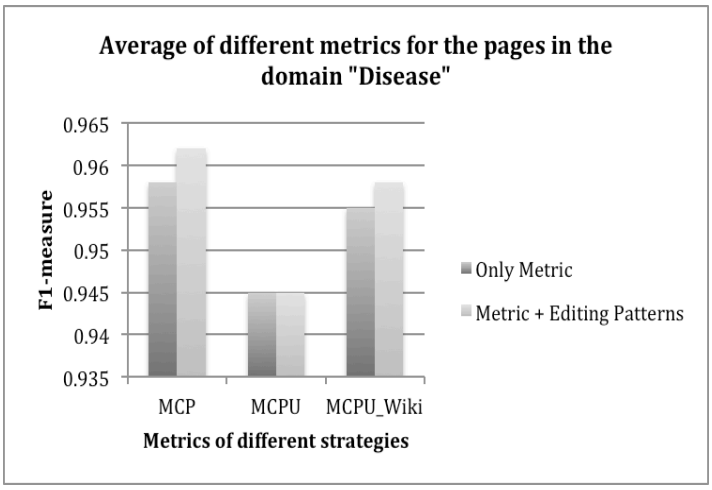

Figure 2.40: Experiment 2 - Results on Wikipedia domain “Disease”

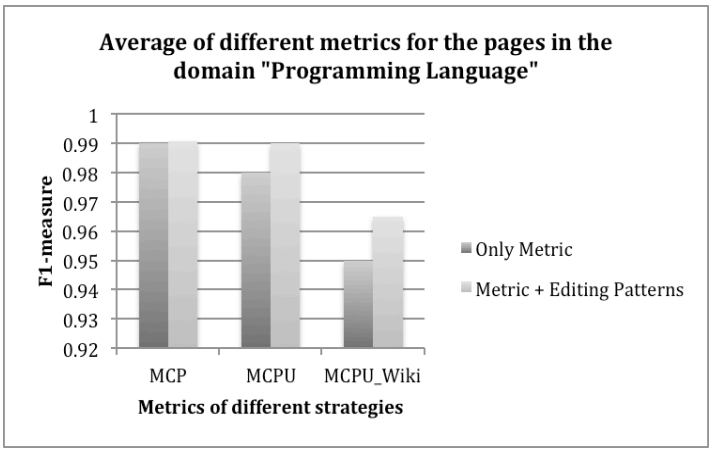

Figure 2.41: Experiment 2 - Results on Wikipedia domain “Programming Language”

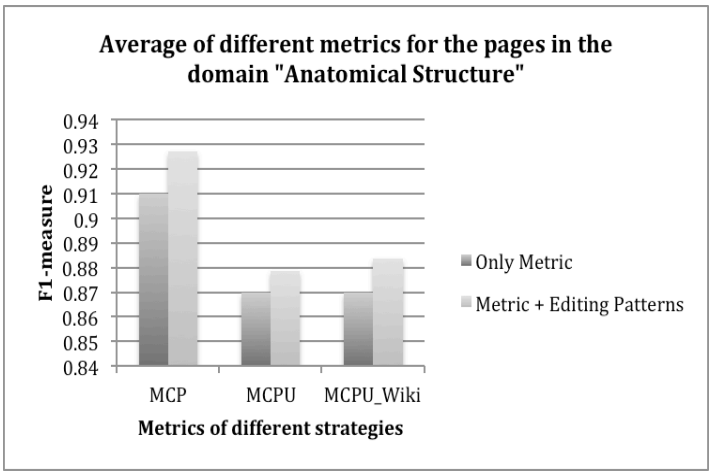

Figure 2.42: Experiment 2 - Results on Wikipedia domain “Anatomical Structure”

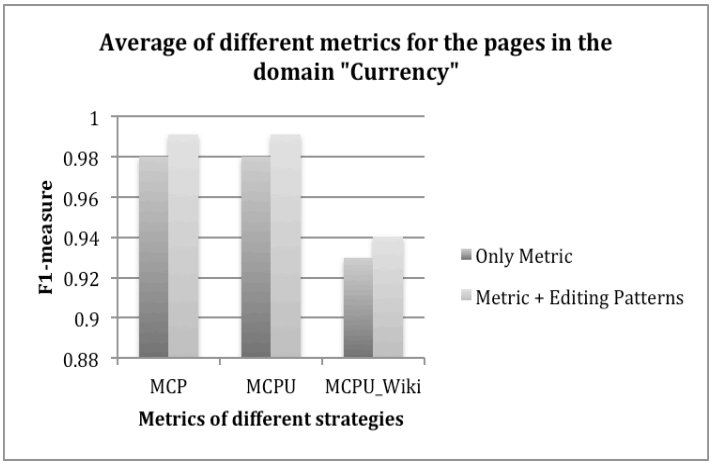

Figure 2.43: Experiment 2 - Results on Wikipedia domain “Currency”

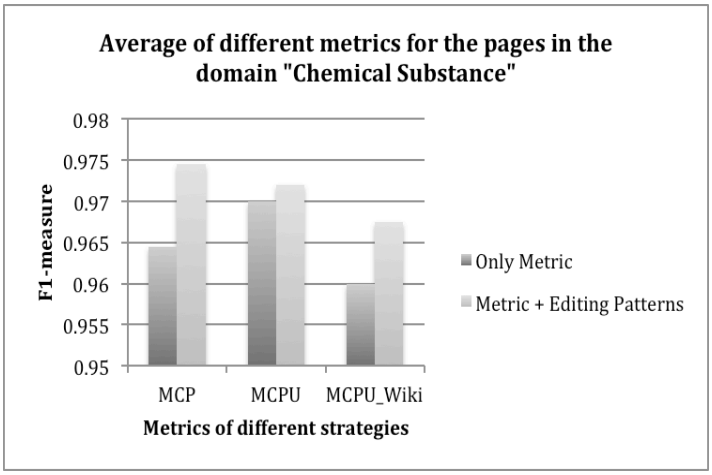

Figure 2.44: Experiment 2 - Results on Wikipedia domain "Chemical Substance"

The above graphs shows results from different domains. The f1-measure of the pages in a single domain are averaged. The f1-measures for different metrics used in different strategies are shown as bars. There are two bars per metric, one bar showing the f1-measure using the metric alone and the other bar showing the f1-measure using the metric alone with the editing patterns (Statement Inverse, Topic Replacement and Number Replacement). It is clear from the graphs that in all the domains, f1-measures of Metric+Editing Patterns is giving either similar or better results compared to using just the metric alone.

Experiment 3: Experiment to show that among the different classifiers used, AdaBoost and Decision Tree give similar or better results compared to Naïve Bayes classifier.

In this experiment following are the results for five domains. X-axis consists of different classifiers and the y-axis consists of f1-measures of different metrics from different strategies. MCP stands for the metric Minimum Co-occurrence Probability, MCPU stands for the metric Minimum Co-occurrence Probability URL, MCPU_Wiki stands for the metric Minimum Co-occurrence Probability URL_Wiki, TC stands for the f1-measure obtained from the strategy Text Classification.

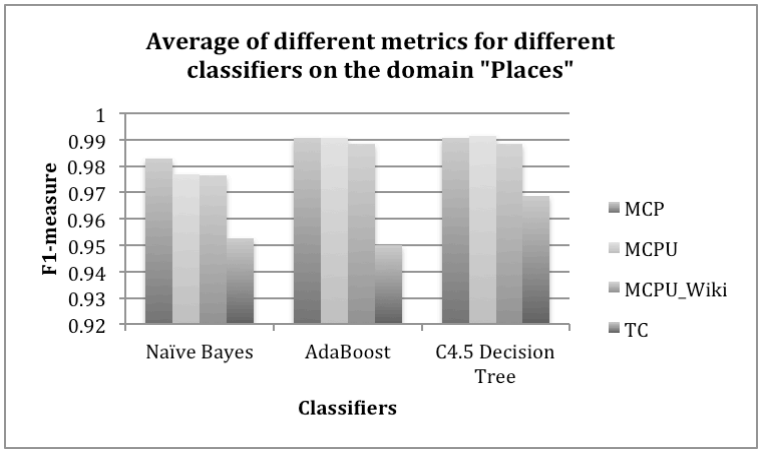

Figure 2.45: Experiment 3 - Results on Wikipedia domain "Places"

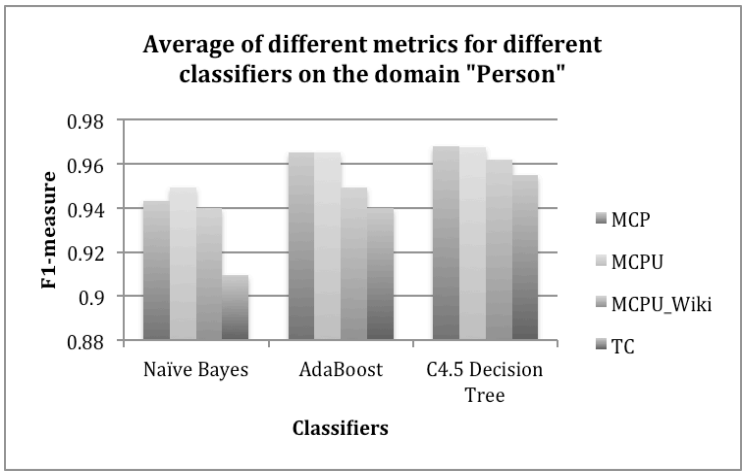

Figure 2.46: Experiment 3 - Results on Wikipedia domain "Person"

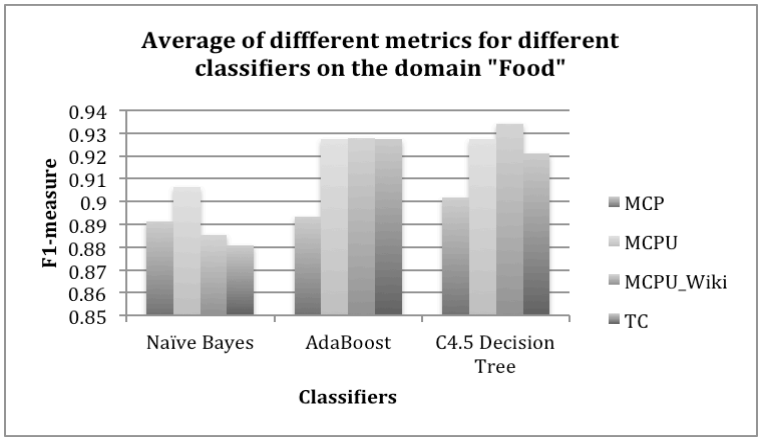

Figure 2.47: Experiment 3 - Results on Wikipedia domain "Food"

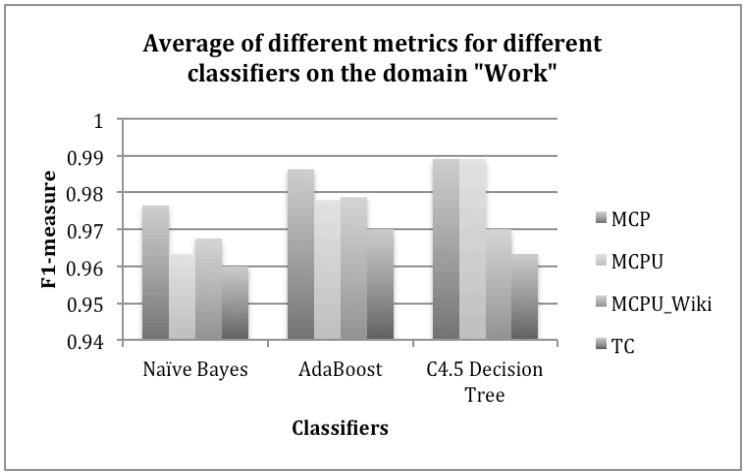

Figure 2.48: Experiment 3 - Results on Wikipedia domain "Work"

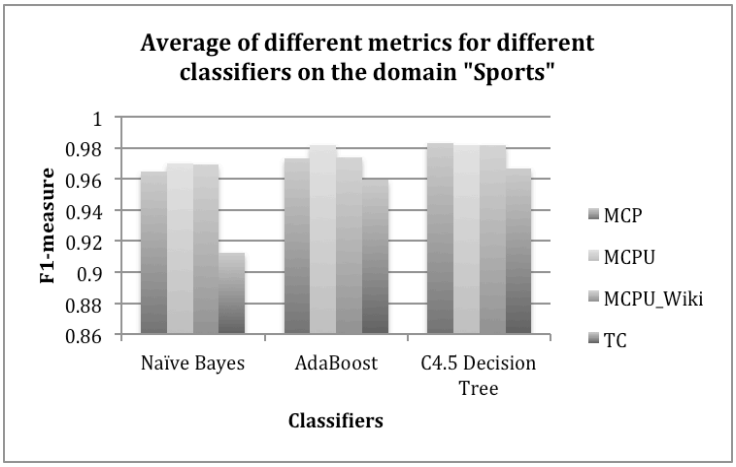

Figure 2.49: Experiment 3 - Results on Wikipedia domain "Sports"

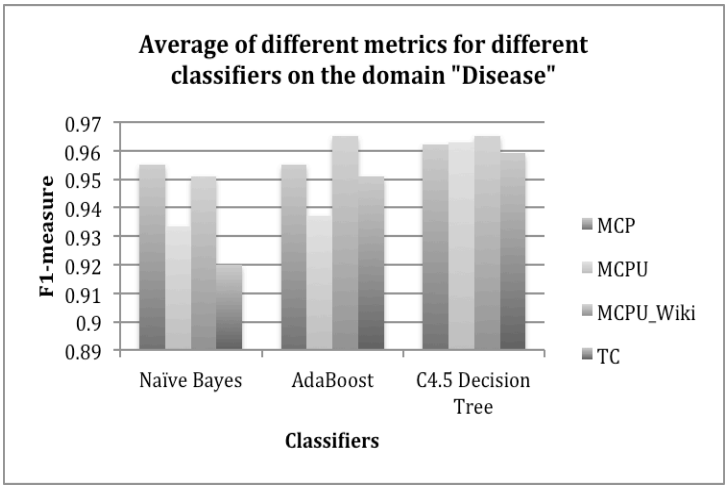

Figure 2.50: Experiment 3 - Results on Wikipedia domain "Disease"

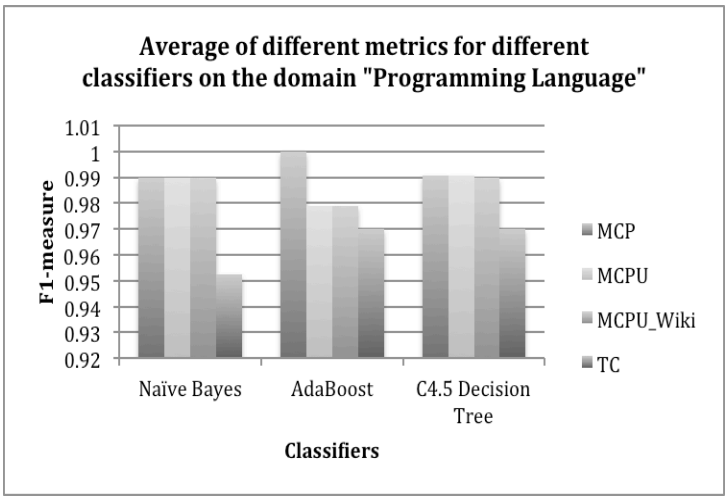

Figure 2.51: Experiment 3 - Results on Wikipedia domain "Programming Language"

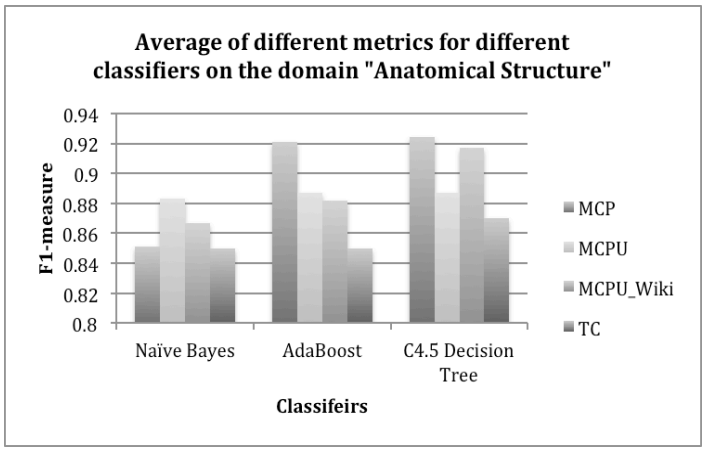

Figure 2.52: Experiment 3 - Results on Wikipedia domain "Anatomical Structure"

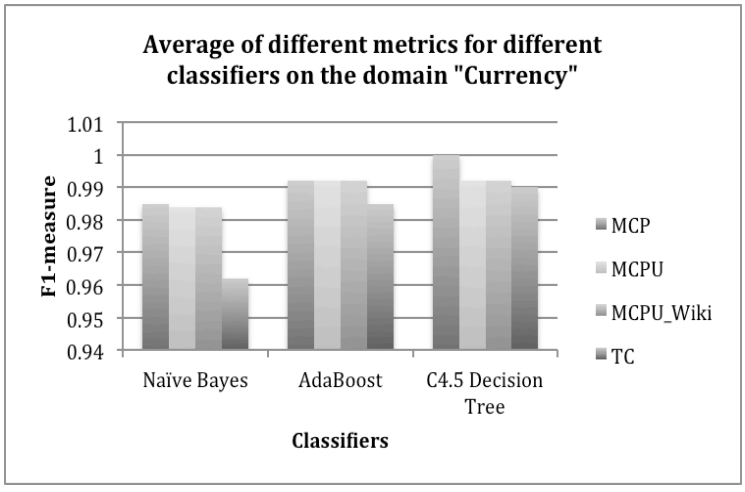

Figure 2.53: Experiment 3 - Results on Wikipedia domain "Currency"

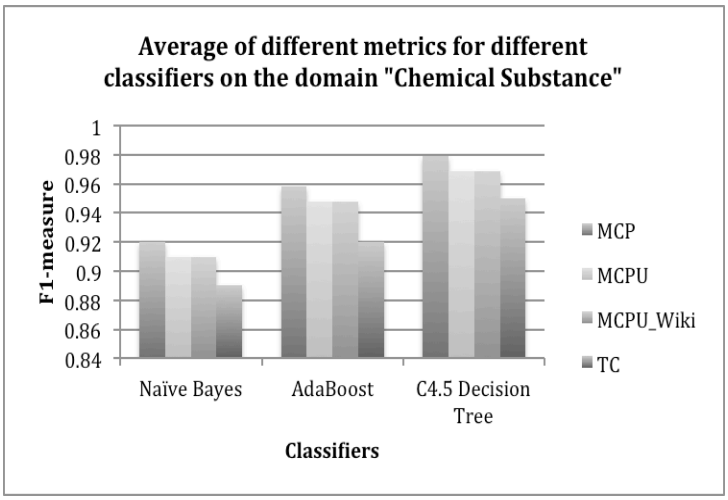

Figure 2.54: Experiment 3 - Results on Wikipedia domain "Chemical Substance"

The above graphs shows results from different domains. In the above graphs the x-axis consists of the different classifiers. The f1-measure of the pages in the domain are averaged. Each classifier consists of 4 bars, each bar represents the f1-measure of the metrics from different strategies. It is observed from the above graphs that the classifiers AdaBoost and DecisionTree are giving better results compared to NaiveBayes classifier. It is also clear from the graphs that the metrics MCP, MCPU and MCPU_Wiki gives better results compared to TC.

2.5 RELATED WORK

A substantial amount of work is done on vandalism detection in the past. Most of the current vandalism detection techniques are based on machine learning approaches. Such machine learning approaches [1] use a set of features which are fed into a supervised learning algorithm to determine whether an edit is vandalism or not.

Some approaches for detecting vandalism uses reputation systems [2]. These approaches use a mixture of user and text reputation and simple metadata as features for detecting vandalism.

An other approach for detecting vandalism uses frequency of vandalism in a given feature along with other features that can characterize an article to be vandalized or not, based on its content. Most of the approaches also use rule based approaches [2, 4] for detecting vandalism. These approaches use Wikipedia bots which are automatic and apply them on certain rules to identify some types of vandalism.

Anti-spam techniques do not completely address the vandalism problem. Most of the anti-spam techniques target the financial factor (e.g. tracking the final spam hosting site). Whereas, Vandalism can be generated by a wide variety of reasons and thus require a new set of detection methods.

Although a significant amount of work is done on detecting vandalism, none of the above approaches focused on the actual context of the edit. There are some situations where an edit might not be an obvious vandal marker i.e, an elusive edit, but when included in a certain Wikipedia article can cause it to go out of context with the other content in the article and lead to vandalism. The context aware approach specified in this paper with different metrics to detect elusive vandal edits provides a sophisticated and efficient method to detect vandalism.

2.6 CONCLUSIONS

We proposed a new approach for detecting elusive vandalism in Wikipedia. Elusive vandal edits are very hard to detect and hence require a sophisticated mechanism for detecting it. These type of edits do not contain obvious vandal markers and hence need to check whether a particular edit lies in the context with the other words of the page. In order to detect this type of vandalism we have proposed four different strategies that makes use of a search engine to determine the co-occurrence probabilities of the edits with the page. We have determined

different distinguishing features for each of the strategy and extracted feature sets based on them. Further we performed data trained classification by using WEKA based on the features extracted. The data set we have used is the PAN corpus 2010 (PAN-WVC-10), which is a large collection of Wikipedia, edits. We have used 10 different domains with 5 Wikipedia pages in each of the domains as our experimental sets. Once the feature sets are extracted, we feed them into WEKA, which is a collection of machine learning algorithms for data mining tasks. We have used three different classifiers (Naïve Bayes, AdaBoost, Decision Tree) for classifying our feature sets. Each of the selected classifiers divides the feature sets into training and testing samples on a 10-fold cross validation. Based on the training set, the classifier classifies the testing set and determines the true and false positives and negatives. We calculated the accuracy of our strategies with f1-measure. We have observed that, the distinguishing factors in each of the three strategies i.e, MCP, MCPU and MCPU confined to Wikipedia documents when combined with the three other distinguishing factors Statement Inverse, Topic Replacement and Number Replacement resulted in better results. We have also observed that among all the three classifiers, AdaBoost and Decision Tree are giving better results and the highest f1-measure we have observed is 1. It is also observed that the metrics from our strategies are giving better results compared to the method text classification.

The above-mentioned strategies are efficient because we observed that the classifiers are able to classify all the data with very less amount of false positives or negatives

CHAPTER 3

SUMMARY

We proposed a new approach for detecting elusive vandalism in Wikipedia. Elusive vandal edits are very hard to detect and hence require a sophisticated mechanism for detecting it. These type of edits do not contain obvious vandal markers and hence need to check whether a particular edit lies in the context with the other words of the page. In order to detect this type of vandalism we have proposed four different strategies that makes use of a search engine to determine the co-occurrence probabilities of the edits with the page. We have determined different distinguishing features for each of the strategy and extracted feature sets based on them. Further we performed data trained classification by using WEKA based on the features extracted. The data set we have used is the PAN corpus 2010 (PAN-WVC-10), which is a large collection of Wikipedia, edits. We have used 10 different domains with 5 Wikipedia pages in each of the domains as our experimental sets. Once the feature sets are extracted, we feed them into WEKA, which is a collection of machine learning algorithms for data mining tasks. We have used three different classifiers (Naïve Bayes, AdaBoost, Decision Tree) for classifying our feature sets. Each of the selected classifiers divides the feature sets into training and testing samples on a 10-fold cross validation. Based on the training set, the classifier classifies the testing set and determines the true and false positives and negatives. We calculated the accuracy of our strategies with f1-measure. We have observed that, the distinguishing factors in each of the three strategies i.e, MCP, MCPU and MCPU confined to Wikipedia documents when combined with the three other distinguishing factors Statement Inverse, Topic Replacement and Number

Replacement resulted in better results. We have also observed that among all the three classifiers, AdaBoost and Decision Tree are giving better results and the highest f1-measure we have observed is 1.

The above-mentioned strategies are efficient because we observed that the classifiers are able to classify all the data with very less amount of false positives or negatives.

REFERENCES

[1] Santiago M. Mola Velasco: Wikipedia Vandalism Detection Through Machine Learning: Feature Review and New Proposals Lab Report for PAN at CLEF 2010.

[2] B. Thomas Adler, Luca de Alfaro, Santiago M. Mola-Velasco, Paolo Rosso, Andrew G. West: Wikipedia Vandalism Detection: Combining Natural Language, Metadata, and Reputation Features, 2-1-2011.

[3] Martin Potthast, Benno Stein and Robert Gerling: Automatic Vandalism Detection in Wikipedia, 2008.

[4] Si Chi-Chin, Padmini Srinivasan, W.Nick Street, David Eichmann: Detecting Wikipedia Vandalism with Active Learning and Statistical Language Models.

[5] Qinyi Wu, Danesh Irani, Calton Pu, Lakshmish Ramaswamy: Elusive Vandalism Detection in Wikipedia: a Text Stability-based Approach

[6] Amit Belani: Vandalism Detection in Wikipedia: a Bag-of-Words Classifier Approach, 1-5-2010.

[7] Manoj Harpalani, Michael Hart, Sandesh Singh, Rob Johnson, and Yejin Choi: Language of Vandalism: Improving Wikipedia Vandalism Detection via Stylometric Analysis.

[8] William Yang Wang and Kathleen R. McKeown: “Got You!”: Automatic Vandalism Detection in Wikipedia with Web-based Shallow Syntactic-Semantic Modeling.

[9] Remco R. Bouckaert, Eibe Frank, Mark A. Hall, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, Ian H. Witten: WEKA—Experiences with a Java Open-Source Project

[10] Sapna Jain, Afshar Aalam, M.N Doja: K-MEANS CLUSTERING USING WEKA INTERFACE

[11] Ian H.Witten, Eibe Frank, Mark A. Hall: Data Mining: Practical Machine Learning Tools and Techniques (Third Edition), January 2011.

[12] Tom M.Mitchell: Machine Learning.

[13] Alex Smola and S.V.N. Vishwanathan : Introduction to Machine Learning.

[14] Stephen Marsland: Machine Learning: An Algorithmic Perspective.

[15] Sara Javanmardi, David. W.McDonald, Cristina V. Lopes: Vandalism Detection in Wikipedia: A High-Performing, Feature–Rich Model and its Reduction Through Lasso.

[16] Remco R. Bouckaert: Bayesian Network Classifiers in Weka, 9-1-2004.

[17] Yoav Freund, Robert E. Schapire: A Short Introduction to Boosting, Spetember 1999.

[18] Koen Smets , Bart Goethals , Brigitte Verdonk: Automatic vandalism detection in Wikipedia: Towards a machine learning approach, 2008.

[19] Sara Javanmardi, David. W.McDonald, Rich Caruana, Sholeh Forouzan, Cristina V. Lopes: Learning To Detect Vandalism in Social Content Systems: A Study On Wikipedia.

[20] Si Chi-Chin, W. Nick Street: Divide and Transfer: an Exploration of Segmented Transfer to Detect Wikipedia Vandalism.