A Rough Set based Approach to Detect Plagiarism

M. Bhavani
Dept. of Computer science and Engineering, VIIT
Visakhapatnam, India
bhavani.mm@gmail.com

K.Thammi Reddy
Dept of Computer Science and Engineering, VIIT, Visakhapatnam, India
thammireddy@yahoo.com

M.Shashi
Dept. of Computer Science and systems Engineering, Andhra University, Visakhapatnam, India
smogalla2000@yahoo.com

Abstract

Plagiarism is the practice of claiming, or implying, original authorship or incorporating material from someone else's written or creative work, in whole or in part, into one's own without adequate acknowledgement. Unlike cases of forgery, in which the authenticity of the writing, document, or some other kind of object, itself is in question, plagiarism is concerned with the issue of false attribution [21]. Plagiarism has become a significant problem in the student community due to the fact that the wide accessability of digitalized information in the WWW. It has become difficult task for the teachers as well as adjudicators to catch the cheaters. There are many tools which are either internet based or pc based to detect plagiarism and both are having advantages and disadvantages.

To detect plagiarism there is a need to find the extent of similarity between a pair of text documents for providing access to topically relevant documents on one hand and for identifying document replication on the other hand. In this paper the details of a Rough Set based Document Ranking system (RSDRS) developed by the authors are presented. The terms associated with related concepts are grouped together to form equivalence classes by clustering the terms of the vocabulary. The query passage and the documents are represented as rough sets using these equivalence classes of terms and further partitioned into families of rough sets in higher level approximation spaces which impose partial ordering on the families of documents with reference to the query passage. Documents falling in the same family are ordered in accordance with their similarity to the query to form the relevance ranking of the documents

Keywords: Plagiarism, Rough set, clustering, Term frequency

1. Introduction

Plagiarism is the practice of claiming, or implying, original authorship or incorporating material from someone else's written or creative work, in whole or in part, into one's own without adequate acknowledgement. Unlike cases of forgery, in which the authenticity of the writing, document, or some other kind of object, itself is in question, plagiarism is concerned with the issue of false attribution [20], [21]. It also comes under the intellectual property crime.

A commercial software Turnitin is used to detect palgiarism, which uses student paper database for detection. The service is expencive and takes time to give the report. Essay Verification Engine ((EVE 2) is also a commercial software to detect plagiarism, which uses numerical value to specify the amount of plagiarism and also gives the url's of sites which contain the copied text [14].

To detect Plagiarism various techniques can be used and in this paper we presented a Rough Set based approach to detect Plagiarism along with Information retrieval technique. There are different IR systems that aim at retrieving relevant documents for a given query and attempt to establish a simple ordering of the documents using different ranking algorithms. Documents appearing at the top of the ordering are considered to be more likely to be relevant. Thus, ranking algorithms are at the core of information retrieval systems.

The most important feature of classical information retrieval models is the representation of documents in terms of keywords. When all words present in a document are used as index terms, the retrieval method is called full-text information retrieval. However, often only nouns are used as index terms, because they describe the content of a document better than adjectives, connectives and adverbs, which have less semantics. Even if only nouns are used as index terms, it is clearly shown that not all nouns are equally useful as index terms. Therefore, there has to be some measure of utility associated with each index term. This measure is called a weight. Assume there are t- distinct index terms in a collection of documents, and the set of all index terms K is equal to $\{k_1, k_2 \ldots k_t\}$. Associated with each combination of an index term k_i and a document d_j there is a weight w_{ij}. The weights are gathered in a document vector $d = (w_{1j}, w_{i,j}\ w_{2j} \ldots\ w_{tj})$ for each document d_j, which is used to find the relevant documents based on the similarity among the document vector. In traditional IR systems due to the high dimensionality of document vectors, relevance analysis which is based on similarity estimates may not provide good results. In order to overcome this drawback we have used the text mining techniques may be used to reduce the dimensionality of document vectors.

Text mining is concerned with the analysis of very large document collections, retrieval of relevant documents based on the full / partial matching of a document and the extraction of hidden knowledge from text based data. Text mining is gaining a lot of importance with the increasing amount of information available in electronic form. It is very close to the web content mining. Web content mining is mainly related to the data available on the web where as the Text mining deals with text documents in general, such as letters, articles, reports and emails which exist either in the internet or intranet. Document data stored in most of the text databases is semi structured in the sense that it is neither completely structured nor unstructured.

Digital libraries organize huge repositories of text and provide access to the documents associated with various topics. They need to find the extent of similarity of two text passages

978-1-4244-4547-9/09/$26.00 ©2009 IEEE

in order to avoid replication of documents on one hand and to provide access to topically relevant documents on the other hand, while the second task is a standard functionality of an IR system, the first task supports detection of Plagiarism or co-derivation [3], [18].

We have developed a Rough Set based Document Ranking System for document to document comparison that estimates the extent to which the documents available in the selected corpus are similar to a query passage and accordingly rank them which in turn used to detect plagiarism. A collection of documents are pre-processed and transformed into document-vectors [7]. The high dimensionality of the document vector is handled by clustering the terms associated with related concepts into equivalence classes [8], [15], [16]. Based on these equivalence classes of terms, documents and the query passage are represented as Rough Sets in terms of upper and lower approximations. The similarity of a document to a query passage is estimated in a two step process.

2. Rough set theory for detecting Plagiarism

Rough set theory [5] is an extension of conventional set theory that supports approximations in decision making. It possesses many features in common to a certain extent with the Dempster-Shafer theory of evidence and fuzzy set theory [6], [18]. The rough set is the approximation of a vague concept (set) by a pair of precise concepts called lower and upper approximation. The lower approximation is a description of the domain objects which are known with certainty to belong to the subset of interest, the objects that exemplify the vague concept. The upper approximation is a description of the domain objects which may possibly, belong to the subset.

The Rough Set theory is applied for document to document comparison and for ranking the documents in the relevance order for a given query passage. The terms of the vocabulary form the universe of objects, 'O' of the information system A=(O,R). The equivalence relation 'R' partitions the set of terms into clusters of terms associated with related concepts so that the terms belonging to a cluster represent the elementary set of objects in the Rough Set terminology. The objects of an elementary set are indiscernible in terms of discriminating attributes (term weights in various documents) of 'A'. Composed sets are formed as unions of a finite number of selected elementary sets and they are always definable.

Let 'o' be a subset of universal set 'O'. It can either be a definable set or a Rough Set. Whenever it is not possible to obtain 'o' as a union of finite number of elementary sets, 'o' can be represented as a Rough set in terms of its lower and upper approximations in the space A. The upper approximation of 'o' in A, $\bar{A}$ (o) is the smallest composed set in A containing 'o'. The lower approximation of 'o' in A, $\underline{A}$ (o) is the largest composed set in A that is contained in 'o'.

The following illustration shows the application of these concepts for document representation in Information systems. Let the vocabulary 'O' be the set of terms t_1 to t_9 which is expressed as 'O' = $\{t_1, t_2 \ldots t_9\}$.Suppose the term clustering implements the equivalence relation, R and partitions the vocabulary into elementary sets $\{t_1, t_5, t_9\}$, $\{t_2, t_4\}$, $\{t_3, t_6, t_7\}$ and $\{t_8\}$. A given document 'o1' containing terms {t1,t5,t8,t9} is definable because it is a composed set obtained as a union of the first and the last elementary sets. Another document 'o_2' containing terms $\{t_1, t_2, t_4, t_5, t_8\}$ is a Rough set because it is not a composed set. 'o2' can be represented in terms of its lower approximation $\underline{A}$ $(o_2) = \{t_2, t_4, t_8\}$ and the upper approximation $\bar{A}$ $(o_2) = \{t_1, t_5, t_9, t_2, t_4, t_8\}$.

The Rough set formalism for categorizing a subset 'o' of universal set in accordance with the values of lower and upper approximations is given below. Let 1 represent the empty set then, with respect to A, o is categorized into one of the following groups:

1.	'o' is Definable	iff $\bar{A}(o) = \underline{A}(o)$
2.	'o' is Roughly Definable	iff $\underline{A}(o) \neq 1$ and $\bar{A}(o) \neq O$
3.	'o' is Externally Undefinable	iff $\underline{A}(o) \neq 1$ and $\bar{A}(o) = O$
4.	'o' is Internally Undefinable	iff $\underline{A}(o) = 1$ and $\bar{A}(o) \neq O$
5.	'o' is Totally Undefinable	iff $\underline{A}(o) = 1$ and $\bar{A}(o) = O$

2.1. Rough relationships

The formalism for comparing two rough sets x and y in A is given below:

2.1.1 Equality based relationships

x and y are '**roughly bottom equal($\underset{\sim}{=}$)**', iff their lower approximations are equal and is mathematically expressed as i.e., $x \underset{\sim}{=} y$ iff $\underline{A}(x) = \underline{A}(y)$.

x and y are **'roughly top equal($\tilde{=}$)'**, iff their upper approximations are equal and is mathematically expressed as i.e., $x \tilde{=} y$ iff $\bar{A}(x) = \bar{A}(y)$.

x and y are **'roughly equal(≈)'**, iff their lower and upper approximations are equal and is mathematically expressed as i.e., $x \approx y$ iff $\bar{A}(x) = \bar{A}(y)$ AND $\underline{A}(x) = \underline{A}(y)$

2.1.2. Subset based relationships

x is **'roughly bottom included($\underset{\sim}{\subset}$)'** in y, iff lower approximation of x is a subset of the lower approximation of y which is mathematically expressed as i.e., $x \underset{\sim}{\subset} y$, iff $\underline{A}(x) \subset \underline{A}(y)$.

x is **'roughly top included($\underset{\sim}{\subset}$)'** in y, iff upper approximation of x is a subset of the upper approximation of y which is mathematically expressed as i.e., $x \underset{\sim}{\subset} y$ iff $\bar{A}(x) \subset \bar{A}(y)$.

x is **'roughly included ($\underset{\sim}{\subset}$)'**, in y, iff the lower and upper approximations of x are the subsets of the lower and upper approximations of y respectively. This is mathematically expressed as i.e., $x \underset{\sim}{\subset} y$ iff $\bar{A}(x) \subset \bar{A}(y)$ and $\underline{A}(x) \subset \underline{A}(y)$.

2.1.3. Superset based relationships

x 'roughly **bottom contains ($\underset{\sim}{\supset}$)'** y, iff lower approximation of x is a superset of the lower approximation of y which is mathematically expressed as i.e., $x \underset{\sim}{\supset} y$, iff $\underline{A}(x) \supset \underline{A}(y)$.

x **'roughly top contains ($\supset$)'** y, iff upper approximation of x is a superset of upper approximation of y which is mathematically expressed as i.e., $x \supset y$ iff $\bar{A}(x) \supset \bar{A}(y)$.

x **'roughly contains ($\underset{\sim}{\supset}$)'** y, iff the lower and upper approximations of x are supersets of lower and upper approximations of y respectively. This is mathematically expressed as i.e., $x \underset{\sim}{\supset} y$ iff $\bar{A}(x) \supset \bar{A}(y)$ and $\underline{A}(x) \supset \underline{A}(y)$.

The equality based relationships defined above obeys the symmetric, reflexive, transitive properties and there by identified as equivalence relationships defined on the Rough Sets in the approximation space A= (O, R). Higher level approximation spaces can be defined over A taking the power set of O as a set of objects and any one of the equivalence relationships defined in the Rough Set space for partitioning it into families of rough sets. The higher level approximation spaces defined over A are enumerated below.

1. $A^* = (P(O), \approx)$; 2. $\underline{A}^* = (P(O), \underset{\sim}{-})$; 3. $\bar{A}^* = (P(O), \bar{-})$

2.2. Families of rough sets

A higher level approximation space defined over A partitions its objects P(O) which are in turn rough sets into families of rough sets. Associated with each of the rough set, say x, there are three families of rough sets one in each of the higher level approximation spaces and it may be noted that the members of a family are in indiscernible in this space.

$E_{\underline{A}}(x)$ is the family of rough sets which are roughly bottom equal to x and it is mathematically expressed as: $E_{\underline{A}}(x) = \{y | \underline{A}(y) = \underline{A}(x)\}$.

$E_{\bar{A}}(x)$ is the family of rough sets which are roughly top equal to x and it is mathematically expressed as: $E\bar{A}(x) = \{y | \bar{A}(y) = \bar{A}(x)\}$.

$E_A(x)$ is the family of rough sets which are roughly equal to x and it is mathematically expressed as: $E_A(x) = \{y | (\underline{A}(y) = \underline{A}(x)) \text{ AND } (\bar{A}(y) = \bar{A}(x))\}$.

The rough sets which are related to x through subset and super set based relationships can be distributed among different families of rough sets as detailed below:

$I_{\underline{A}}(x)$ is the family of rough sets which are roughly bottom included in x and it is mathematically expressed as: $I_{\underline{A}}(x) = \{y | \underline{A}(y) \subset \underline{A}(x)\}$.

$I_{\bar{A}}(x)$ is the family of rough sets which are roughly top included in x and it is mathematically expressed as: $I\bar{A}(x) = \{y | \bar{A}(y) \subset \bar{A}(x)\}$.

$I_A(x)$ is the family of rough sets which are roughly included in x and it is mathematically expressed as: $I_A(x) = \{y | (\underline{A}(y) \subset \underline{A}(x)) \text{ AND } (\bar{A}(y) \subset \bar{A}(x))\}$.

$C_{\underline{A}}(x)$ is the family of rough sets which roughly bottom contain x and it is mathematically expressed as: $C_{\underline{A}}(x) = \{y | \underline{A}(y) \supset \underline{A}(x)\}$.

$C_{\bar{A}}(x)$ is the family of rough sets which roughly top contain x and it is mathematically expressed as: $C\bar{A}(x) = \{y | \bar{A}(y) \supset \bar{A}(x)\}$.

$C_A(x)$ is the family of rough sets which roughly contain in x and it is mathematically expressed as: $C_A(x) = \{y | (\underline{A}(y) \subset \underline{A}(x)) \text{ AND } (\bar{A}(y) \supset \bar{A}(x))\}$.

3. Rough set model for document ranking

In rough set modeling of the information system, documents are represented as rough sets in an approximation space A=(O,R) where O is the vocabulary of terms related to a selected corpus. The term clustering implements the equivalence relationship, R, and partition the vocabulary of words into groups of terms associated with related concepts. Using the equivalence classes of terms the dimensionality of a document can be reduced as illustrated below. In the previous example, since the 9 terms of 'O' are clustered into 4 groups, the dimensionality of a document is reduced from 9 to 4. Document o_2 containing terms $\{t_1, t_2, t_4, t_5, t_8\}$ can be represented in terms of its lower and upper approximations $\underline{A}(o_2)$ is represented as {C2, C4} in low dimensional space instead of $\{t_2, t_4, t_8\}$ and $\bar{A}(o_2)$ is represented as {C1C2, C4} instead of its high dimensional representation $\{t_1, t_2, t_4, t_5, t_8, t_9\}$.

Table-1 shows the representation of various documents as rough sets in high and low dimensional space and their relationships to document o_2. The documents numbered o_2 to o_7 described in table 1 are roughly definable where as document o1 is definable.

High level approximation spaces are defined over the approximation space A, for partitioning the rough sets into families. In the context of Information Retrieval, P (O) represents the set of all documents containing the terms of vocabulary. The approximation space $\underline{A}^*$ = (P (O), $\underset{\sim}{-}$) partitions the document of the corpus such that all documents which are roughly bottom equal ($\underset{\sim}{-}$) fall into the same partition. Table-1 indicates that the documents o2 and o6 fall into the same partition in the approximation space $\underline{A}^*$. Similarly the approximation space $\bar{A}^*$ = (P (O), $\tilde{-}$) partitions the documents such that all documents which are roughly top equal ($\tilde{-}$) fall into the same partition. Table-1 indicates that the document o2 and o7 fall into the same partition in $\bar{A}^*$.

"Table 1. Representation of documents as rough sets & their relationships"

object	Document	Lower approximation		Upper approximation		Relation ship
		High dimensional space	Low dimensional space	High dimensional space	Low dimensional space	
o1	{t1, t5,t6,t8}	{t1, t5, t6, t8}	{C1,C4}	{t1, t5,t6,t8}	{C1,C4}	Definable
o2	{t1, t2, t4, t5, t8}	{t2, t4, t8}	{C2,C4}	{t1,t2,t4,t5, t8, t9}	{C1,C2,C4}	Roughly Definable
o3	{t1,t2,t3, t4}	{t2,t4}	{C2}	{{t1, t5, t9, t2, t4, t3,t6,t7}	{C1,C2,C3}	$o3 \subset o2$
o4	{t1,t2,t4,t5 }	{t2, t4}	{C2}	{t1,t2,t4,t5, t9}	{C1,C2}	$o4 \underset{\sim}{\subset} o2$
o5	{t1,t2,t4, t8}	{t2,t4, t8}	{C2,C4}	{{t1,t2,t4,t5 t8,t9}	{C1,C2,C4}	$o5 \underset{\sim}{=} o2$
o6	{t2, t4, t8}	{t2, t4, t8}	{C2,C4}	{t2, t4, t8}	{C2,C4}	$o6 \underset{\sim}{-} o2$
o7	{t1, t4, t5,t8, t9}	{{t1, t5,t9, t8}	{C1,C4}	{t1, t2,t4,t5, t8, t9}	{C1,C2,C4}	$o7 \tilde{-} o2$

The third approximation space A^* = (P (O), ≈) partition the documents such that documents which are roughly equal (≈) belongs to the same partition. Table-1 indicates that the documents o2 and o5 fall into the same partition in the approximation space A^*.The documents of a partition are indiscernible in these higher level approximation spaces and thereby require to be further ordered based on their (higher dimensional) detailed representation.

In the information Retrieval context the query passage is also represented as a rough set. All those documents that are related to the query passage are classified into one of the families of rough sets denoted by $E_A(x)$, $E_{\underline{A}}(x)$, $E_{\bar{A}}(x)$, $C_A(x)$, $C_{\underline{A}}(x)$, $C_{\bar{A}}(x)$,$I_A(x)$, $I_{\underline{A}}(x)$, and $I_{\bar{A}}(x)$ in that order based on their relationship to query passage. This defines a partial ordering among the documents relevant to the query. This defines a partial ordering among the documents relevant to the query. However if the query is definable and there exists an equal and definable document then such a document is the most relevant to the query and is given the best ranking followed by the documents identified in various families of rough sets in the order mentioned above. If multiple documents are identified in the same family they are further ordered based on similarity estimates.

The documents which are roughly equal ($E_A(Qi)$), bottom equal ($E_{\underline{A}}(Qi)$), and top equal ($E_{\bar{A}}(Qi)$) to the given query passage, Qi, are further ordered based on the cosine similarity of the query to various documents of the same family. Multiple documents that fall into the families containing subsets or super sets of terms of the query passage are further ordered based on one of the similarity estimates as mentioned below.

Accordingly the similarity of i th query passage to the j th document denoted by $\underline{Sim}(Qi,dj)$ is the ratio of the number of common clusters found in both Qi and dj to the number of clusters found in either Qi or dj with reference to their lower approximations. Similarly $\overline{Sim}(Qi,dj)$ denotes the similarity of Qi and dj with reference to their upper approximations.

"Table 2. Similarity estimation of documents that include /contain the query terms."

Sl.no	Family	Similarity estimation
1.	$I_{\underline{A}}(Qi), C_{\underline{A}}(Qi)$	$\underline{Sim}(Qi,dj) = \frac{(\lvert \underline{A}(Qi) \wedge \underline{A}(dj) \rvert)}{(\lvert \underline{A}(Qi) \vee \underline{A}(dj) \rvert)}$
2.	$I_{\bar{A}}(Qi), C_{\bar{A}}(Qi)$	$\overline{Sim}(Qi,dj) = \frac{(\lvert \bar{A}(Qi) \wedge \bar{A}(dj) \rvert)}{(\lvert \bar{A}(Qi) \vee \bar{A}(dj) \rvert)}$
3.	$I_A(Qi), C_A(Qi)$	$Sim(Qi,dj) = \underline{Sim}(Qi,dj) + \overline{Sim}(Qi,dj)$

4. The rough set based document ranking system architecture (RSDRS)

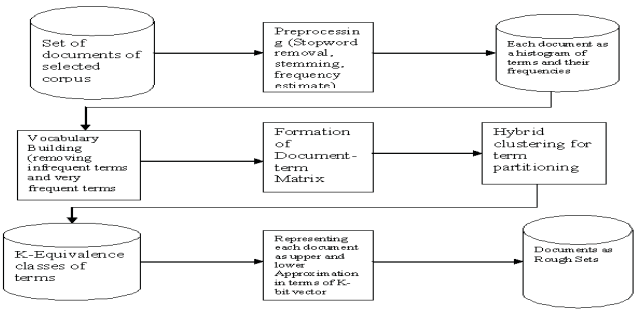
We have implemented Rough Set based Document Ranking System which aims at ordering the documents of a corpus according to their relevance to a given query passage. The system operates in two phases. In the first phase documents belonging to each corpus are represented as a separate collection of Rough Sets in a low dimensional space. This is a preparatory phase that transforms the documents into a suitable form for ranking based on the query which is shown in Figure 1. In the second phase the query passage is transformed into a Rough set in an approximate space corresponding to the corpus. The documents are partitioned into Rough Set families based on their relevance to the query passage and further ordered in accordance with similarity estimates as shown in Figure 2. The organization of the system is explained in detail below:

4.1. Preparatory Phase

4.1.1. Document pre-processing. The text contained in the document is tokenized and the stop words are removed. Words appearing in different morphological forms are mapped on to their common stems using a trie –like structure proposed by the authors [7], [11]. Thus each document is transformed into a list of stems along with their frequency of occurrence in the document.

4.1.2. Representation of terms in a vector space. Once all the documents belonging to a corpus are processed, the set of all stems found in various documents of the corpus is identified and their frequency of occurrence in various documents is counted. The vocabulary of index terms for the corpus is formed by removing the infrequent and the most frequent stems from this set. The documents term matrix for the corpus is formed, which contains the term weights wij of the jth term in ith document. Each row of the matrix represents a document vector and each column of the matrix represents an index term and its prominence in various documents.

4.1.3. Clustering related terms. The distance between all pairs of index terms are calculated using Euclidian distance estimate. A hybrid clustering algorithm [8] is used to partition the terms associated with the related concepts into K-equivalence classes referred to as the elementary sets in Rough Set terminology.



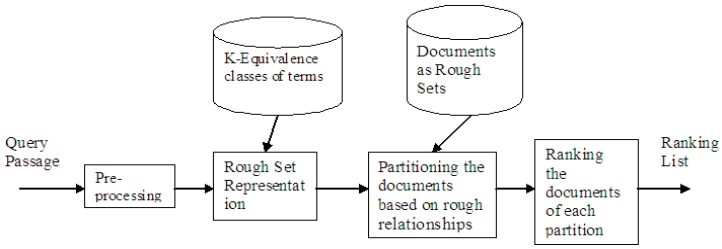
"Figure 1. System model for representation of documents as rough sets"

4.1.4. Document representation as rough sets. The document vector represented as a row in the document- term matrix is high dimensional because of the large number of terms in the vocabulary. The dimensionality of the document vector is reduced by representing a document in terms of equivalence classes of related terms rather than the actual index terms. The Rough Set concepts are found suitable to transform a document from a high dimensional space to a low dimensional space and each document is represented as a Rough set in terms of its lower and upper approximations [9] [19].

4.2. Ranking Phase

4.2.1. Query pre-processing. The query passage is tokenized and the stop words are removed. Words appearing in different morphological forms are mapped on to their common stems. Thus query is transformed into a list of stems along with their frequency of occurrence in the query passage.

4.2.2 Query representation as a rough set. With reference to the K-equivalence classes of related terms, a query is represented as a Rough Set in terms of lower and upper approximation as K-bit vector.



"Figure 2. Query processing. dynamic ranking"

4.2.3. Documents ranking in relevance to query passage. Compare the lower and upper approximations of the query passage (Qi) to check if it is definable and find a document which is equal to the definable query and rank it the best. Compare the lower approximation of the query with the lower approximation of the all the documents in the collection to identify those documents that are roughly bottom equal to the query. Compare the upper approximation of the query with the upper approximation of the all the documents in the collection to identify those documents that are roughly top equal to the query. Identify the documents which are both roughly bottom equal and top equal to the given query. If multiple documents exists in this group rank them based on their cosine similarity to the query using the formula given below.

$$\mathrm{Cos}(Q_i, d_j) = \frac{Q_i}{\|Q_i\|} \bullet \frac{d_j}{\|d_j\|}$$

The documents belonging to this group are followed by the group of documents that are only roughly bottom equal to the query and which are in turn followed by the documents which are roughly top equal. Documents of the same group are further ordered based on their cosine similarity to query passage. The remaining documents are classified into various rough set families based on whether they are related to the query passage through Rough inclusion, Roughly bottom inclusion, Roughly top inclusion, Rough containment, Roughly bottom containment, and Roughly top containment relationships. The document falling in various rough set families are partially ordered as $C_A(Qi)$, $C_{\underline{A}}(Qi)$, $C_{\bar{A}}(Qi)$, $I_A(Qi)$, $I_{\underline{A}}(Qi)$, and $I_{\bar{A}}(Qi)$. The similarity measures suitable for ranking the documents belonging to each family are given in table-2.

4.3. Incremental update to document collection

It is possible, while the system is in use to add a new collection of documents to an existing set of documents of a corpus as detailed below. Each document of the new collection is preprocessed and added as a row in the document term matrix with term frequencies. Using the K-equivalence classes of the corpus, the document is represented as a K-bit lower

and upper approximation and added to the existing collection of documents.

4.4 Interacting with the System

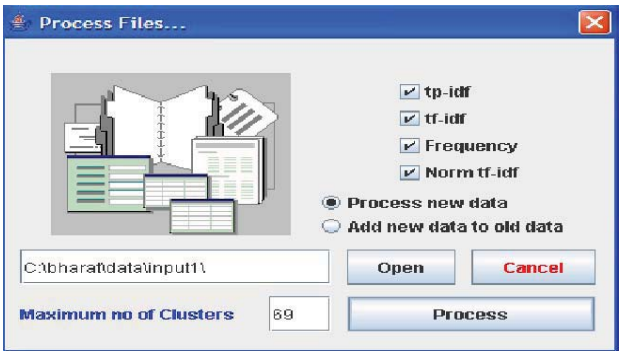
The Graphical User Interface for Interacting with the Rough Set based Document Ranking System (RSDRS) provides a user friendly environment for using the system by just selecting the required option to be performed. The GUI for RSDRS mainly consists of three important options as shown in the Screen 1. The first one being the 'process Files' button and by selecting it, the important task namely: document pre-processing starts.

Screen 1. The Graphical User Interface for the RSDRS System.

This preparatory stage is carried out interactively and is facilitated through subsequent frames of the GUI as follows. After selecting the corpus from the file menu through 'Open' option the user specifies the maximum number of clusters required, a default value is provided as shown in Screen 2. The user may select one among the term weight estimates provided in the GUI for transforming the documents into term-document matrix; by default all the term weight estimates namely tf-idf, frequency based, Norm tf-idf and tp-idf are used for term-document matrix representation. Once the options are selected and the 'Process' button is pressed the terms of the vocabulary are partitioned into equivalence classes by applying Hybrid Clustering algorithm and each document is represented in the form of Rough Set.

The GUI also contains an option to incrementally add the new documents to the document collection. All the newly added documents are represented as Rough Set by making use of the term partitioning done before. This facilitates the scalability of the system by allowing any number of the documents to be added to the corpus without performing much preprocessing.

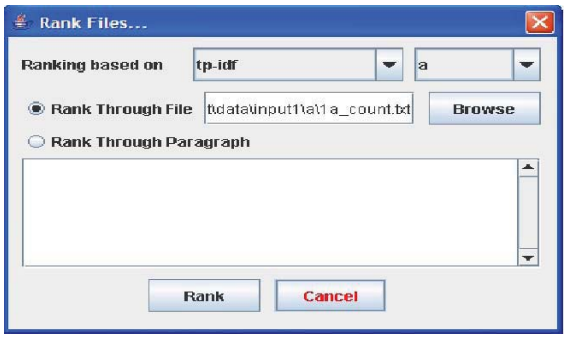
The second option provided in the GUI of the RSDRS being 'Rank Files' facilitates document ranking based on the users query as per the user's requirement. Here the user selects the corpus based on the subject he would like to explore, the term weight estimate to be used for query-document comparison and the query itself. The user can select one among the two options available as shown in Screen 3.



Screen 2. The GUI Showing the Preparatory stage.

One being the whole document as a query file provides the user with browsing facility and the other option being a query passage accepts a passage entered by the user at run time in a text box.

Once all the parameters are set then the 'Rank button' from the Rank Files GUI is selected to Rank the relevant documents from the selected corpus. The out put is displayed in the form of a table which contains Rank and the filenames in the relevance order The Rank Files GUI also provides an opportunity for the user to specify the query as a passage to check whether there are any documents having relevance to the given passage and ranks accordingly.



Screen 3. The GUI Showing the selection of the file to be searched.

5. Experimental analysis

The Ancillary Data set [13] which contain around 1500 abstracts of the thesis submitted in various universities on various subjects like Machine Learning, Image processing and so on is used to test the Rough set based Document Ranking system for ranking the documents. We have analyzed the performance of the system in detail on a set of 150 abstracts on Machine Learning which is referred to as corpus.

In order to represent the documents as rough sets the terms of the vocabulary of Machine Learning corpus are partitioned

into equivalence classes using Hybrid clustering algorithm. After accepting initial seed points as an outcome of agglomerative clustering algorithm, Incremental K-means algorithm [10] is applied to cluster the terms of the vocabulary. As the clustering quality of K-means algorithm is measured in terms of sum of squared error (SSE), we have plotted the SSE versus number of clusters(K) graph and selected the optimal range of values for K. The performance of the RSDRS is analyzed at various values K in this sample. Once the K is selected, documents are represented as K-bit rough sets. Based on the query the Rough Set based Document Ranking System generates relevance ordering for the documents of the corpus containing classified thesis abstracts.

The document ranking system is developed in Java under the eclipse environment. The performance of the system is analyzed by representing the term weights using four measures namely term frequency, tf-idf, Ho tf-idf [18] and the tp-idf (normalized tf-idf) suggested by the authors [8]. The retrieval efficiency is calculated using Average precision formula given by D.Harman [2]. This Measure of quality is based on the underlying idea that documents appearing at top position in the ranking list are expected to be more relevant to query.

At different cut-off points as suggested by D.Harman [2] in terms of the Average Precision formula given below. In general the precision and recall are the measures used for quality estimation of a Information Retrieval system. While the precision indicates the possibility of a retrieved document to be relevant, the recall indicates the possibility of retrieving every relevant document. However these measures are not directly suitable to evaluate a document ranking system as they do not discriminate a hit/fail happening at the earlier position to a hit/fail happening at the end of the ranking list. Relevance Ranking can be evaluated by computing precision

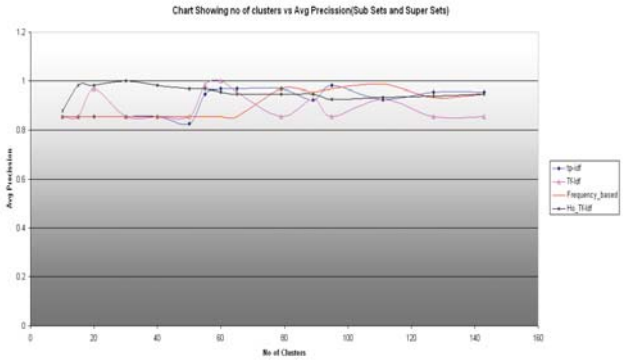
$$\frac{1}{R}\sum_{d\in D}\frac{\#\{relevant\ documents\ retrieved\ upto\ d\}}{\#\{number\ \text{of documents upto d}\}}$$

where R denotes the total number of relevant documents and d denotes successive positions in the ranking list, D.

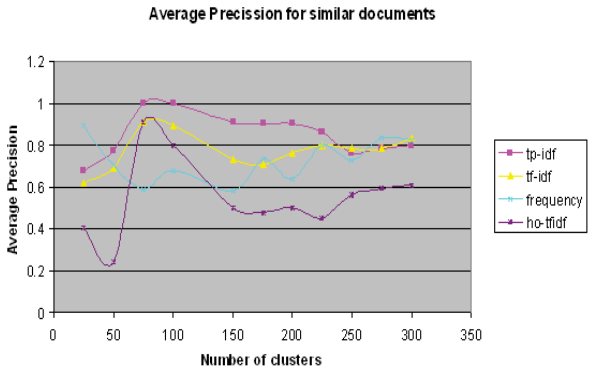
The performance of the system is analyzed by finding and plotting the average precision of the system for a varied number of clusters with different term weight estimates as parameter. It is found that the tp-idf suggested by the authors has shown consistently good results.

The system was thoroughly tested on different types of queries especially in two scenarios; the first scenario being the query passage that has an exactly matching document, some documents which are its super sets and some others that are subsets of the query passage. In this case the system could successfully recognize the exactly matching document and ranked it with the top most position, followed by the documents that are supersets and finally followed by the documents that are its subsets in accordance with their difference. The performance of the system in this scenario is depicted in Figure 3 for a corpus of size 40 documents with 320 index terms. The performance of the system is found to be ideal (with average precision close to 1) for larger corpus. The successful performance of the system in this scenario suggests that we can use it to detect plagiarism. Even if a document is slightly altered by adding some noise words or using some alternative terms in place of original terms, this system could successfully identify the original document and place it in the top position when the altered document is given as query.

In the second scenario a new query passage is given and the system could successfully recognize the documents relevant to the query. The performance of the system in this scenario is plotted in Figure 4 and 5 for a corpus of size 150 and 40 respectively. It can be observed that the performance of the document ranking system while identifying relevant documents to a query passage is found to be optimal at around K= 80 irrespective of the size of the corpus and term weight estimates. However, the selection of K value did not influence the performance of the system in the first scenario. Based on these observations in order to perform well in all situations the system should be operated with a minimum number of clusters which is 80 for Machine Learning corpus.



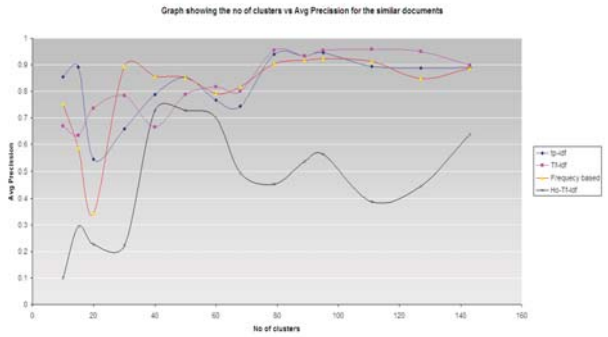
"Figure 3. Graph showing average Precision for 1st Scenario for a corpus size of 40 documents"



"Figure 4. Graph showing average precision for relevant document set for a corpus size of 150 documents"

As there are more than thousand index terms in the larger corpus, transformation of a document into rough set form

reduces its dimensionality up to 8% (1\12th of its original size). The above graphs indicate the performance of the system with different term weight estimates in the given range of K values. The term weight estimate tp-idf suggested by the authors gave consistently good results over the selected range of K values in both the scenarios irrespective of their size of the corpus.



"Figure 5. Graph showing average precision for relevant document set for a corpus size of 40 Documents".

6. Conclusions

The Rough Set based Document Ranking system to detect the extent of Plagiarism developed by the authors aims at organizing the documents on a subject as a separate corpus and ranking those documents in relevance order to a given query passage. The system operates in two phases; the preparatory phase partitions the terms into the equivalence classes and represents each document as a rough set. In the ranking phase the system compares the query passage, which is again represented as a rough set, with the documents of the relevant corpus in a high level approximation space built over the rough set space and generates the ranking list. The performance of the system is estimated in terms of average precision for different scenarios. In addition to finding relevant documents to a query passage, the system is expected to be successful in identifying the cases of plagiarism. Incremental update to the existing collection of documents is facilitated in the system to intermittently expand its scope.

7. References

[1] C.J. Van Rijsbergen, *Information Retrieval*, Second edition, London: Butterworths, 1979.

[2] D.Harmson "The Trec conferences in proceedings of Hypertext, *Information Retrieval*, Multimedia, "1995, pp 9-28..

[3] D. Metzler, Y. Bernstein, W.Bruce Croft, A. Moffat and J.Zobel, "Similarity Measures for Tracking Information Flow" in the proceedings of CIKM'05, October 31-November 5, 2005, Bremen, Germany.

[4] Jiawei Han, Micheline Kamber, *Data Mining Concepts and Techniques*, Morgan kaufmann publishers, 2006

[5] Lech Polkowski, Shusaku T Sumoto, T sau Y.Lin, "Rough Set Methods and Applications –New Developments in Knowledge discovery in Information Systems", *A Springer-Verlag* Company, 2000.

[6] L.J.Mazlack, Aijing He, Y Zhu, Sarah Coppock, "A Rough Set Approach in Choosing Partitioning Attributes". Proceedings of the ISCAv13th International Conference (CAINE-2000) November, 2000 1-6.

[7] K.Thammi Reddy, M.Shashi and L.Pratap Reddy, "Efficient implementation of stemming Algorithm using Trie-Like Structure", Proceedings of the *International Conference on Statistical Science,* OR and IT held at Tirupathi during 7-9 January 2007.

[8] K.Thammi Reddy, M.Shashi and L.Pratap Reddy, "Hybrid Clustering Approach for Term Partitioning in Document Data Sets", Proceedings of the *International Conference on Systemics*, Cybernetics and Informatics held at Hyderabad during 3-7 January 2007.

[9] Padmini Das Gupta, "Rough sets and information retrieval", *Journal of the ACM*, 1988.

[10] Pang-Ning Tan, Michael Steinbach, Vipin Kumar, *Introduction to Data Mining*, Addison Wesley publisher, 2006.

[11] Porter.M.F. "An algorithm for Suffix Stripping, Program 14(3), 1980, pp.130- 137.

[12] Ricardo Baeza-Yates, Berthier Ribeiro-Neto, *Modern Information Retrieval*, Pearson Education, 2004.

[13] Richard K Belew, *Finding Out About-A Cognitive Perspective on Search Engine Technology and the WWW*, Cambridge Press, 2000.

[14] Rosa G. Paredes, J. Alfredo Sanchez and Antonia Razo, "Drawing the line between fair use and plagiarism for digital documents", Proceedings of the Eighth Mexican International Conference on Current Trends in Computer Science, 2007, 113-122.

[15] S.K.M.Wong, W.Ziarko, "A machine learning approach to information retrieval", ACM Conference on research and development in information retrieval 1986, 228-233.

[16] S.M.Ruger, S.E.Gauch, "Feature Reduction for document Clustering and Classification", Doc TR 2000/8, Imperial college, London, 2002.

[17] Tu Bao Ho,Saori Kawasaki,Ngoc Binh Nguyen, "Non hierarchical Document Clustering Based on Tolerance Rough Set Midel", *International Journal of Intelligent Systems*, Vol. 17 (2002), No.2, 199-212.

[18] Th.W.Ch. Huibers, M.Lalmas and C.J. Van Rijsbergen "Information retrieval and situation theory". Journal of ACM SIGIR, Volume 30, pages 11-25 1996.

[19] Tu Bao Ho, Saori Kawasaki, Ngoc Binh Nguyen, "Text Mining with Tolerance Rough Set Models", *International Journal of Intelligent Systems*, 2002.

[20] Wikipedia, http://en.wikipedia.org/wiki/Plagiarism.

[21] www.bleacherreport.com