JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Classification of Documents using Clustering Algorithm with term frequency-inverse documents frequency

Dr.Vinod Sharma,RNTU, Bhopal

Dr. Shiv Shakti Shrivastava,RNTU, Bhopal

Abstract:-In order to overcome the limitations, this paper proposes a research paper classification system that can cluster research papers into the meaningful class in which papers are very likely to have similar subjects. Numerous research papers have been published online as well as offline with the increasing advance of computer and information technologies, which makes it difficult for users to search and categorize their interesting research papers for a specific subject Clustering analysis method is one of the main analytical methods in data mining,the method of clustering algorithm will influence the clustering results directly. The K-means clustering algorithm is computationally faster than the other clustering algorithms. However, it produces different clustering results for different number of clusters. So, it is required to determine the number of clusters (i.e., K value) in advance before clustering.

In order to overcome the limitations, this paper proposes a research paper classification system that can cluster research papers into the meaningful class in which papers are very likely to have similar subjects. Numerous research papers have been published online as well as offline with the increasing advance of computer and information technologies, which makes it difficult for users to search and categorize their interesting research papers for a specific subject Clustering analysis method is one of the main analytical methods in data mining, the method of clustering algorithm will influence the clustering results directly. [1].Here we review some of the different ways taxonomists have so far made use of the Web. We suspect that the large majority of taxonomists now use the Web in one form or another, and we make no pretense at

Taxonomic Data on the Web

To do good taxonomy, a researcher needs specimens to study, a catalogue of previous taxon hypotheses, and access to the relevant literature. Aids to all of these have been developed on the Web (Scoble, 2004). These resources are the raw material of taxonomy typically made available for the use of other taxonomists. Most major museums and herbaria have projects underway to provide catalogues of the specimens they hold on the Web (see the online appendix, available at http://SystematicBiology.org, section 1, for examples of relevant Web sites). The magnitude of the task varies.

The DiGIR project takes Web-based specimen catalogs one step further by allowing institutions to expose their data in a standard way allowing use of generic software tools that query multiple collections at once.. Several taxon-specific Web

sites have been established that act as portals to this global data set.

In addition to the raw specimen data, collection-level metadata on, for example, the number of specimens of a given species housed within a particular museum can provide an invaluable source of information for the taxonomist. Collecting and presenting on the Web this type of data can be done far more quickly than specimenlevel digitization (for example, see the www.biocase.org project, which included both elements). Knowing where a type or other specimen is housed is just the first step in using it to do new taxonomy.

Before any serious work can be done on a group of organisms, a catalogue of the names associated with the taxon is essential (Scoble, 1999). Producing a catalogue of names is one of the most important things a systematist can do to facilitate further research on a group. Catalogues and lists are very simply transferred to the Web and numerous projects exist to provide such resources on the Web (online appendix section 4). Catalogues vary greatly in the amount of information they contain, some being little more than a list of names, whereas others carry considerable information about synonymy, literature citation, location of type material, and even distribution and ecology. They also differ in completeness, some reflecting a systematic attempt to locate all names applied to a taxon, whereas others are more an accumulation of information as it becomes available. They may also differ in geographical scope, ranging from regional

In addition to specimens and a catalogue of names, the taxonomist requires access to the literature, which traditionally has meant a library

Taxonomy is different for two reasons. First, as discussed above, papers published decades if not centuries in the past are still of value to the subject.

Raw Data Syntheses

As will have become apparent from the brief review above, there are already a large number of sources of basic taxonomic data available on the Web, and this has naturally led to the need for a means to index and locate them. We discuss here a few of the projects that seek to do this on the Web. However, some of the most powerful tools for taxonomy on the Web are the major search engines, Google in particular, which are already a prime means of locating new and relevant sites (see also the discussion of mash-ups, below). However, specialized synthesis sites and portals may be able to provide more targeted search tools and summaries for the taxonomist.

Taxonomic Tools

All aspects of taxonomy are becoming increasingly computer dependent, and in addition to providing a means of sharing raw data, the Web also represents a source of the software needed by practicing taxonomists.

Nearly all revisionary taxonomy involves the manipulation of large amounts of data, a task that has become much easier with the widespread availability of standard spreadsheet and database programs. However, taxonomic data have their own particular structures and peculiarities, and handling these can be assisted with databases and data-handling routines explicitly designed for this market. Similarly, a very common task for taxonomists is the construction of identification keys. The traditional dichotomous key was a clever means of organizing information before the age of computers, but now identification can often be made easier by allowing multiple entry points into the key and for the user to choose which characters to score (Pankhurst, 1991; Farr, 2006).

Wikis—Collaborative Authoring

A very different model for how taxonomy might develop on the Web is through the idea of communitybased projects such as wikis. The changes made to a wiki are unmoderated, can be made by anyone, often without the need for registration, and are not checked for accuracy and quality. Without doubt the most famous site is Wikipedia (online appendix section 10), an

encyclopedia of everything, which currently (early 2007) contains more than 1.5 million articles in English and perhaps 3 million in all languages. Within broadly defined limits, users are free to add entries and to edit or overwrite entries written by others. Naturally, the quality of different articles varies, though a large community of people now uses it very extensively (see, e.g., Benkler, 2006). In 2005, the journal Nature controversially compared Wikipedia entries favorably to those in the Web-version of the venerable Encyclopaedia Britannica (Giles, 2005b), though the latter argued vigorously that the comparison was unfair (Encyclopedia Britannica, Inc., 2006). Wikipedia itself contains many entries to species and other taxa but the organization behind it, the Wikimedia Foundation, has gone further and in 2005 set up Wikispecies, a "free directory of life" consisting of articles organized by the Linnaean hierarchy. Currently it contains over 82,000 entries, though many of these are simply names ("stubs"). However, if the popularity of its parent organization is replicated here, the number of entries and their richness of information are likely to increase rapidly.

Mash-Ups—Federating Data

The idea of having a Web page for every species (Wilson, 2003) is an enticing vision underlying Wikispecies,Tree-of-Life (http://www.tolWeb.org/tree/) and, before its apparent demise, the All Species Foundation (http:// www.all-species.org/). The challenges of creating this resource are enormous, but could it be done automatically, by searching the Web for all the information available on a particular taxon? Sites produced in this way are called mash-ups (Butler, 2006).iSpeciesproject(http://darwin.zoology.gla.ac. uk/rpage/ispecies/)(http://darwin.zoology.gla.ac.uk/rpage/ispecies/) is a simple test of how this might be done; code is written to interrogate biodiversity, image, sequence, and literature databases, making use of standard application programming interfaces (APIs) that they expose to other software. The user enters a taxon name and the program returns a list of links to the resources indexed by that name. It is already quite impressive the number of links that are found for even quite obscure taxa, and it is certain that this will increase in the future. What iSpecies demonstrates is the enormous potential that could be achieved if all databases from natural history museums and herbariums were linked in a searchable distributed network.

However, the value of this approach depends on the reliability of the underlying taxonomic nomenclature, as well as the taxonomic accuracy of the original studies that the mash-up collates. Taxonomic names are not always unique identifiers for taxonomic hypotheses, and consequently software that uses a Linnean binomial to search other resources may aggregate data about different taxonomic concepts (Berendsohn, 1995). It will work best for well-known, easily distinguishable species, and poorly for those groups where either the taxonomy is in flux or specimens difficult to identify. Some of these problems can be addressed by the use of Globally Unique Identifiers such as the LifeSciences Identifiers (LSID) technology discussed above, which makes it possible to identify data in an explicit and machinereadable.

III. TECHNIQUES OF DOCUMENT CLASSIFICATION TECHNIQUES

1.Voting

In [6] calculation depends on strategy for classifier boards of trustees and depends on thought that given assignment that requires master opinion for learning. Here k number of specialists feeling might be superior to anything one if their individual decisions are properly consolidated. Distinctive mix rules are available as the most straightforward conceivable guideline is lion's share casting a ballot (MV)If a few classifiers are concede to a class for a test text document, the aftereffect of casting a ballot classifier is that class. Second weighted dominant part casting a ballot, in this technique, the loads is explicit for each class in this weighting strategy, mistake of every classifier is determined.

1.Centroid based classifier

The centroid-based characterization calculation is exceptionally basic. [7] For each arrangement of text documents having a place with a similar class, this paper figures their centroid vectors. In the event that there are k classes in the preparation set, this prompts k centroid vectors (C1, C2, C3...) where each Cn is the centroid for the stream class. The class of another text document x is resolved as, First the archive frequencies of the different terms registered from the preparation set Then, figure the likeness between x to all k centroid utilizing the cosine measure. At long last, in view of these likenesses, and relegate x to the class relating to the most comparable centroid.

2. K-Nearest Neighbors

K-NN classifier is a case-based learning [8] calculation that depends on a separation or closeness work for sets of perceptions, for example, the Euclidean separation

or Cosine comparability measure's. This technique was used for some application in [9] because of its viability, non-parametric and simple to usage properties. But this technique have some set of issues like the grouping time is long and hard to discover ideal estimation of number of cluster that means value of k .The best decision of k relies on the information for the most part, bigger estimations of k diminish the impact of noise on the arrangement, yet make limits between classes less particular.

3 . Naive Bayes

Naïve technique is somewhat module classifier [10] under known priori likelihood and class restrictive likelihood .it is essential thought is to figure the likelihood that text document D is has a place with class C. There are two occasion display are available for credulous Bias as multivariate Bernoulli and multinomial model. Out of these model multinomial model is progressively appropriate when database is substantial, yet there are distinguishes two significant issue with multinomial model first it is unpleasant parameter evaluated and issue it lies in taking care of uncommon classes that contain just couple of preparing archives.

SVM

The use of Support vector machine (SVM) technique to Text Classification has been proposed by [11]. The SVM need both positive and negative preparing set which are extraordinary for other characterization techniques. These positive and negative preparing set are required for the SVM to look for the choice surface that best isolates the positive from the negative information in the n dimensional space, this was shown in the hyper plane. The text document agents which are nearest to the choice surface are known as the support vector. There are issues with this technique like it don't work well for multiclass dataset.

1. TF-IDF:

TF-IDF[5,6](Term Frequency-inverse Document Frequency), puts weighting to a term based on its inverse document frequency. It means that if the more documents a term appears, the less important that term will be, and the weighting will be less. $TFIDF\ t = TFt * log\ N\ nt$ TF-IDF-CF: As per the Shortcomings of TF-IDF has, [5] introduce a new parameter to represent the in-class characteristics, and we call this class frequency, which calculates the term frequency in documents within one class. $TFIDFCF\ t = \log TFt + 1 * log\ N + 1\ nt * nc, Nc$

the number of documents where term t appears within the same class c document. Nc represents the number of documents within the same class c document.

The TF-IDF has been widely used in the fields of information retrieval and text mining to evaluate the relationship for each word in the collection of documents. In particular, they are used for extracting core words (i.e., keywords) from documents, calculating similar degrees among documents, deciding search ranking, and so on.

The TF in TF-IDF means the occurrence of specific words in documents. Words with a high TF value have an importance in documents. On the other hand, the DF implies how many times a specific word appears in the collection of documents. It calculates the occurrence of the word in multiple documents, not in only a document. Words with a high DF value do not have an importance because they commonly appear in all documents. Accordingly, the IDF that is an inverse of the DF is used to measure an importance of words in all documents. The high IDF values mean rare words in all documents, resulting to the increase of an importance.

Word frequency

The TF calculation step in Fig. 1 counts how many times the keywords defined in a keyword dictionary and the topics extracted by LDA appear in abstract data. The TF used in this paper is defined as

TFi,j=ni,j∑knk,jTFi,j=ni,j∑knk,j

The abstract data in this figure have the paper length of 64. As we can see in this figure, the keywords 'cloud computing', 'Internet of Things', and 'Big Data' have the TF value of 0.015 because of one occurrence in the abstract data. The keyword 'cloud computing' has the TF value of 0.03 because of two occurrences. Figure 5 shows map-reduce algorithm to calculate word frequency (i.e., TF). In this figure, *n* represents the number of occurrences of a keyword in a document with a paper title of *DocName*.

Document frequency

While the TF means the number of occurrences of each keyword in a document, the DF means how many times each keyword appears in the collection of documents. In the DF calculation step in Fig. 1, the DF is calculated by dividing the total number of documents by the number of documents that contain a specific keyword. It is defined as

DFi,j=|dj∈D:tj∈dj||D|DFi,j=|dj∈D:tj∈dj||D|

where, |D||D| represents total number of documents and |dj∈D:tj∈dj||dj∈D:tj∈dj| represents the number of documents that keyword tjtj occurs. Figure 6 shows an illustrative example when four documents are used to calculate the DF value.

TF-IDF

Keywords with a high DF value cannot have an importance because they commonly appear in the most documents. Accordingly, the IDF that is an inverse of the DF is used to measure an importance of keywords in the collection of documents. The IDF is defined as

IDFi,j=log|D||dj∈D:tj∈dj|IDFi,j=log⁡|D||dj∈D:tj∈dj|

Using Eqs. (2) and (4), the TF-IDF is defined as

TFIDF=TF×IDFTFIDF=TF×IDF

The TF-IDF value increases when a specific keyword has high frequency in a document and the frequency of documents that contain the keyword among the whole documents is low. This principle can be used to find the keywords frequently occurring in documents. Consequently, using the TF-IDF calculated by Eq. (5), we can find out what keywords are important in each paper.

shows the map-reduce algorithm for the TF-IDF calculation of each paper.

K-means clustering

Kmeans algorithm is an iterative algorithm that tries to partition the dataset into *K*pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to **only one group**. It tries to make the intra-cluster data points as similar as possible while also keeping the clusters as different (far) as possible. It assigns data points to a cluster such that the sum of the squared distance between the data points and the cluster's centroid (arithmetic mean of all the data points that belong to that cluster) is at the minimum. The less variation we have within clusters, the more homogeneous (similar) the data points are within the same cluster.

The way kmeans algorithm works is as follows:

1. Specify number of clusters *K*.

2. Initialize centroids by first shuffling the dataset and then randomly selecting *K* data points for the centroids without replacement.

3. Keep iterating until there is no change to the centroids. i.e assignment of data points to clusters isn't changing.

4. Compute the sum of the squared distance between data points and all centroids.

5. Assign each data point to the closest cluster (centroid)

6. Compute the centroids for the clusters by taking the average of the all data points that belong to each cluster.

Typically, clustering technique is used to classify a set of data into classes of similar data. Until now, it has been applied to various applications in many fields such as marketing, biology, pattern recognition, web mining, analysis of social networks, etc. [33]. Among various clustering techniques, we choose the k-means clustering algorithm, which is one of unsupervised learning algorithm, because of its effectiveness and simplicity. More specifically, the algorithm is to classify the data set of *N* items based on features into *k* disjoint subsets. This is done by minimizing distances between data item and the corresponding cluster centroid.

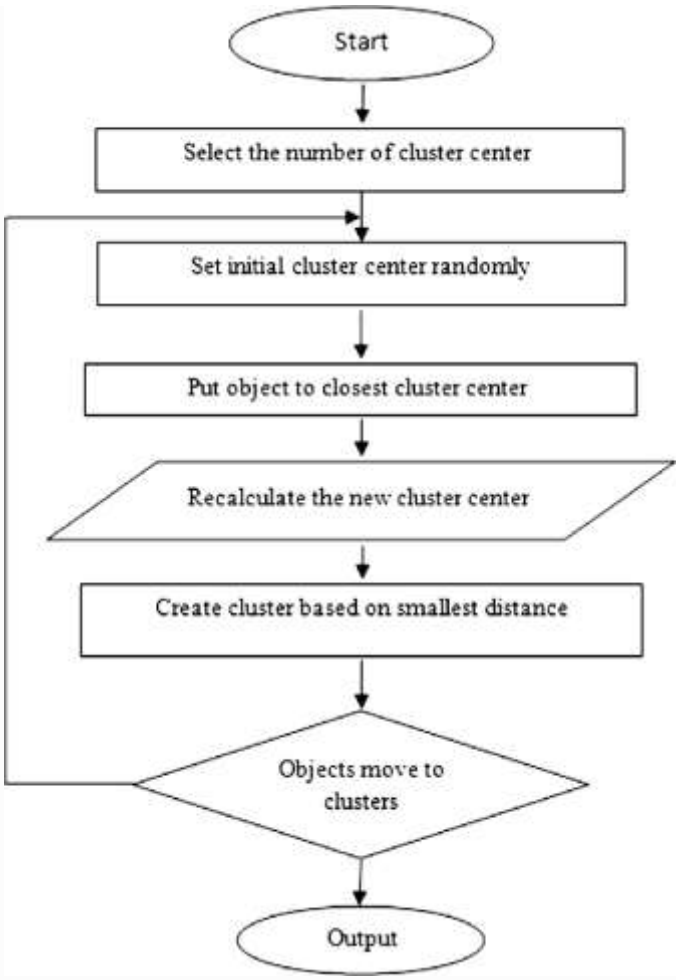

Fig:1 Flowchart of k-means clustering algorithm

Mathematically, the k-means clustering algorithm can be described as follows:

$$E=\sum_{i=1}^{k}\sum_{j\in Ci}\|xj-ci\|2E=\sum_{i=1}^{k}\sum_{j\in Ci}\|xj-ci\|2$$

where, *k* is the number of clusters, xjxj is the *j*th data point in the *i*th cluster CiCi, and cici is the centroid of CiCi. The notation ‖xj−ci‖2‖xj−ci‖2 stands for the distance between xjxj and cici, and Euclidean distance is commonly used as a distance measure. To achieve a representative clustering, a sum of squared error function, *E*, should be as small as possible.

The advantage of the K-means clustering algorithm is that (1) dealing with different types of attributes; (2) discovering clusters with arbitrary shape; (3) minimal requirements for domain knowledge to determine input parameters; (4) dealing with noise and outliers; and (5) minimizing the dissimilarity between data [34].

The TF-IDF value represents an importance of the keywords that determines characteristics of each paper. Thus, the classification of papers by TF-IDF value leads to finding a group of papers with similar subjects according to the importance of keywords. Because of this, this paper uses the K-means clustering algorithm, which is one of most used clustering algorithm, to group papers with similar subjects. The K-means clustering algorithm used in this paper calculates a center of the cluster that represents a group of papers with a specific subject and allocates a paper to a cluster with high similarity, based on a Euclidian distance between the TF-IDF value of the paper and a center value of each cluster.

The K-means clustering algorithm is computationally faster than the other clustering algorithms. However, it produces different clustering results for different number of clusters. So, it is required to determine the number of clusters (i.e., K value) in advance before clustering. To overcome the limitations, we will use the Elbow scheme [35] that can find a proper number of clusters. Also, we will use the Silhouette scheme [36, 37] to validate the performance of clustering results by K-means clustering scheme. The detailed descriptions of the two schemes will be provided in next section with performance evaluation.

where, ni,jni,j represents the number of occurrences ofword titi indocument djdj and ∑knk,j∑knk,j repre sents a total number of occurrences of words in document djdj. *K* and *D* are the number of keywords and documents (i.e., papers), respectively.

Evaluation on the accuracy of the proposed classification system

The accuracy the proposed classification systems has been evaluated by using the well-known F-Score [41] which measure how good paper classification is when compared with reference classification. The F-Score is a combination of the precision and recall values used in information extraction. The precision, recall, and F-Score are defined as follows.

Precision=TPTP+FPPrecision=TPTP+FP
Recall=TPTP+FNRecall=TPTP+FN

FScore=2×Precision × RecallPrecision + RecallF-Score=2×Precision × RecallPrecision + Recall

In the above equations, TP, TN, FP, and FN represents true positive, true negative, false positive, and false negative, respectively. We carried out our experiments on 500 research papers randomly selected among the total 3264 ones used for our experiments. This experiment is run 5 times and the average of F-Score values is recorded.

II. METHODOLOGY

A. Development of k-mean clustering algorithm

Given a dataset of n data points x1, x2, …, xn such that each data point is in R d , the problem of finding the minimum variance clustering of the dataset into k clusters is that of finding k points {mj} (j=1, 2, …, k) in R d such that is minimized, where d(xi , mj) denotes the Euclidean distance between xi and mj . The points {mj} (j=1, 2, …,k) are known as cluster centroids. The problem in Eq.(1) is to find k cluster centroids, such that the average squared Euclidean distance (mean squared error, MSE) between a data point and its nearest cluster centroid is minimized. The k-means algorithm provides an easy method to implement approximate solution to Eq.(1). The reasons for the popularity of k-means are ease and simplicity of implementation, scalability, speed of convergence and adaptability to sparse data.

The k-means algorithm can be thought of as a gradient descent procedure, which begins at starting cluster centroids, and iteratively updates these centroids to decrease the objective function in Eq.(1). The k-means always converge to a local minimum. The particular local minimum found depends on the starting cluster centroids. The problem of finding the global minimum is NP-complete. The k-means algorithm updates cluster centroids till local minimum is found. Fig.1 shows the generalized pseudocodes of k-means algorithm; and traditional k-means algorithm is presented in fig. 2 respectively. Before the k-means algorithm converges, distance and centroid calculations are done while loops are executed a number of times, say l, where the positive integer l is known as the number of k-means iterations. The precise value of l varies depending on the initial starting cluster

centroids even on the same dataset. So the computational time complexity of the algorithm is O(nkl), where n is the total number of objects in the dataset, k is the required number of clusters we

identified and l is the number of iterations, k≤n, l≤n

Step 1: Accept the number of clusters to group data into and the dataset to cluster as input values

Step 2: Initialize the first K clusters

- Take first k instances or

- Take Random sampling of k elements

Step 3: Calculate the arithmetic means of each cluster formed in the dataset.

Step 4: K-means assigns each record in the dataset to only one of the initial clusters - Each record is assigned to the nearest cluster using a measure of distance (e.g Euclidean distance).

Step 5: K-means re-assigns each record in the dataset to the most similar cluster and re-calculates the arithmetic mean of all the clusters in the dataset.

Fig 2: Generalised Pseudocode of Traditional k-means

```
1.  MSE = large number;
2 . Select initial cluster centroids {mj}j K = 1;
3   Do
4         OldMSE = MSE;
5            MSE1 = 0;
6             For j = 1 to k
7               mj = 0; nj = 0;
8                                 endfor
9       For i = 1 to n
10      For j = 1 to k
11 Compute squared Euclidean distance d 2 (xi , mj);
12       endfor
13    Find the closest centroid mj to xi ;
14 mj = mj + xi ; nj = nj+1;
15 MSE1=MSE1+ d2 (xi , mj);
16 endfor
17 For j = 1 to k
18 nj = max(nj , 1); mj = mj /nj ;
19 endfor
20 MSE=MSE1; while (MSE<OldMSE)
```

Fig.3: Traditional k-means algorithm

Analysis of classification results

An illustrative example for classification results. In this table, the papers in cluster 1 indicate that they are grouped by two keywords ‘cloud’ and ‘bigdata’ as a primary keyword. For cluster 2, two keywords ‘IoT’ and ‘privacy’ have an important role in grouping the papers in this cluster. For cluster 3, three keywords ‘IoT’, ‘security’ and ‘privacy’ have an important role. In particular, according to whether or not the keyword ‘security’ is used, the papers in cluster 2 and cluster 3 are grouped into different clusters.

Conclusion

The k-means always converge to a local minimum. The particular local minimum found depends on the starting cluster centroids.The classified document are retrived and stored to the destination by TF-IDF which is highly mention in retrival documents.In this research thought gained the empower of clustering approach. Here assigns each record through k-means to initial and nearest cluster and this iteration perfrom re-assignment of record with

to store recod documents to destination with TF-IDF . the papers in cluster 1 and cluster 2 are grouped into different clusters and process further calculative upto optimization of output.

References

Bafna P, Pramod D, Vaidya A (2016) Document clustering: TF-IDF approach. In: IEEE int. conf. on electrical, electronics, and optimization techniques (ICEEOT). pp 61–66.

S. Sujit Sansgiry, M. Bhosle, and K. Sail, "Factors that affect academic performance among pharmacy students," American Journal of Pharmaceutical Education, 2006.

Susmita Datta and Somnath Datta, "Comparisons and validation of statistical clustering techniques for microarray gene expression data," Bioinformatics, vol. 19, pp.459–466, 2003.

Rousseeuw P. J, "A graphical aid to the interpretation and validation of cluster analysis," Journal of Computational Appl Math, vol 20, pp. 53– 65, 1987.

Oyelade, O. J, Oladipupo, O. O and Obagbuwa, I. CRamos J (2003) Using TF-IDF to determine word relevance in document queries. In: Proc. of the first int. conf. on machine learning.

Havrlant L, Kreinovich V (2017) A simple probabilistic explanation of term frequency-inverse document frequency (TF-IDF) heuristic (and variations motivated by this explanation). Int J Gen Syst 46(1):27–36

Agosti, D. 2006. Biodiversity data are out of local taxonomists' reach. Nature 439:392.

Agosti, D., and N. F. Johnson. 2002. Taxonomists need better access to published data. Nature 417:222.

Beccaloni, G. W., M. J. Scoble, G. S. Robinson, and B. Pitkin, eds. 2003. The Global Lepidoptera Names Index (LepIndex). World Wide Web electronic publication. http://www.nhm.ac.uk/ entomology/lepindex [accessed 6 January 2007].

Benkler, Y. 2006. The wealth of networks. Yale University Press, New Haven, Connecticut.

Box, D., D., Ehnebuske, G. Kakivaya, A. Layman, N. Mendelsohn, H. Frystyk Nielsen, S. Thatte, and D. Winer. 2000. Simple Object Access Protocol (SOAP) 1.1. World Wide Web electronic publication. http://www.w3.org/TR/2000/NOTE-SOAP-20000508/ [accessed 2 April 2007].

Brummitt, R. K., F. Pando, S. Hollis, and N. A. Brummitt. 2001. World geographical scheme for recording plant distributions, edition 2, in Plant Taxonomic Database Standards, number 2, edition 2. Hurst Institute for Botanical Documentation, Carnegie Mellon University, Pittsburg, Pennsylvania.

Butler, D. 2006. Mashups mix data into global service. Nature 439:6–7.

Wilson, E. O. 2003. The encyclopedia of life. Trends Ecol. Evol. 18:77–80.

Giles, J. 2005a. Remote-control microscope to ease work in taxonomy. Nature 433:673.

Giles, J. 2005b. Internet encyclopaedias go head to head. Nature 438:900–901.

Alan Díaz-Manríquez , Ana Bertha Ríos-Alvarado, José Hugo Barrón-Zambrano, Tania Yukary Guerrero-Melendez, And Juan Carlos Elizondo-Leal. "An Automatic Document Classifier System Based on Genetic Algorithm and Taxonomy". accepted March 9, 2018, date of publication March 15, 2018, date of current version May 9, 2018.

Yuefeng Li, Abdulmohsen Algarni, Mubarak Albathan, Yan Shen, and Moch Arif Bijaksana. "Relevance Feature Discovery for Text Mining". IEEE transaction knowledge and Data ENGINEERING, VOL. 27, NO. 6, JUNE.

Souneil Park, Jungil Kim, Kyung Soon Lee, and Junehwa Song. "Disputant Relation-Based Classification for Contrasting Opposing Views of Contentious News Issues".IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 25, NO. 12, DECEMBER 2013.

Wang, J. Wang, et al., "Labelled LDA-Kernel SVM: A Short Chinese Text Supervised Classification Based on Sina Weibo." In 2017 4th International Conference on Information Science and Control Engineering (ICISCE), pp. 428-432. IEEE, 2017.

Mingyong Liu1 and Jiangang Yang. "An improvement of TFIDF weighting in text categorization". 2012 International Conference on Computer Technology and Science (ICCTS 2012).

Yiming Yang Christopher G. Chute "A Linear Least Squares Fit Mapping Method For Information Retrieval From Natural Language Texts" Acres De Coling-92 Nantes, 23-28 AOUT 1992.

B S Harish, D S Guru, S Manjunath " Representation and Classification of Text Documents: A Brief Review" IJCA Special Issue on "Recent Trends in Image Processing and Pattern Recognition"RTIPPR, 2010.

Gongde Guo, Hui Wang, David Bell, Yaxin Bi and Kieran Greer, "KNN Model-Based Approach in Classification", Proc. ODBASE pp- 986 – 996, 2003