ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

Implementation of an Online News Aggregator by using Count Matrix

M. Sundara Chinna, K.Srinivasa Rao

Assistant Network Admin, Department of CGARD, NIRD&PR, Hyderabad, India

Research scholar, Department of CS&SE, AU College of Engineering, Visakhapatnam, India

ABSTRACT*:* The aggregator computes the clusters of news using the descriptions provided in the RSS feeds. The clusters created by the algorithm will be displayed in the Web Browser as groups. Online News Aggregator is Python-PHP Server. ONA has four modules "RSS URL handler", "RSS content Parser" ,"Content Preprocessor" , "Content Clusterer" and "User Interface layer". RSS URL handler is used to read the URLs from the URL repository (which could also be a text file where the administrator can add or delete the RSS feed URL's). RSS content Parser reads the content in the RSS feed and store the content in the database. The Content Preprocessor removes the unwanted content for clustering and stores the content back into the database. The Content Clusterer groups similar news together and stores it back to database. Finally, the content that is grouped is displayed in the Web Browser using the User Interface module. The Content Clusterer first creates the Count Matrix using the pre-processed content. The Count matrix is used to generate the TFidf (Term Frequency - Inverse Document Frequency) matrix which is intern used to create the Cosine similarity matrix. Finally, the cosine similarity matrix is used for clustering the content. The ONA uses Hierarchical clustering algorithm for grouping similar news.

KEYWORDS*:* Really Simple Syndication; Aggregator; Web Browser; Count matrix;Term Frequency - Inverse Document Frequency; Hierarchal Clustering; User Interface

I. INTRODUCTION

Many newspapers publish their news in Internet. A recent research from the Italian Institute of statistics shows that there is an increasing trend of mastheads publishing their contents on the Net often joining to the paper edition an Internet edition with special and more complete information [1]. Internet newspapers may update their contents frequently: thus there is not a daily issue but the news are continuously updated and published. As a consequence, hundreds of thousands of partially overlapping news are daily published. The amount of information daily published is so wide that is unimaginable for a user. On the other hand, the availability of news generates new updated information needs for people. The RSS technology supports Internet users in staying updated news is published in the form of RSS feeds that are periodically downloaded by specific applications called feed readers. In order to improve the users selection of the interesting feeds from different newspapers, publishers group feeds in categories.

The RSS technology and the news classification in categories do not solve all the "news overload" issues. First, the categories are not fixed, and then the same topic may be called in different sites in different ways. Consequently, a user that wants to be updated about a specific topic has to manually browse the categories of potentially all the newspapers looking for interesting news. Then, the amount of news feeds daily published is so wide that automatic tools are required. If we consider the feeds published only by the five main Italian newspapers in one day, more than one thousands of news is available in their websites. Such news is partially overlapping, since different newspapers publish the same information in different news. RSS feeds[10] [11] from different newspapers may carry the same information in different places, and therefore can confuse the reader

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

II. RELATED WORK

In [2] authors describe their approach to collaborative filtering for generating personalized recommendations for users of Google News. Presented novel approaches to clustering over dynamic datasets using Min Hash and PLSI, both of which were adapted to scale arbitrarily using the Map reduce framework developed at Google. In [3], Velthune, a news search engine is proposed. The tool is based on a naive classifier that classifies the news in few categories. In [5] the authors propose an aggregator, called RCS (RSS Clusgator System), implementing a technique for temporal updating the contents of the clusters. NewsInEssence [6] is an advanced aggregator that computes similar news on the basis of a TF*IDF clustering algorithm, and provides to the reader a synthesis of them.

III. PROPOSED ALGORITHM

The online news aggregator solves this issue by creating an aggregator which pulls the published content across different news websites and apply the text clustering algorithm on the content to group similar news. The Grouped news would be then published on website. Figure 1 shows the flow diagram of online news aggregator.

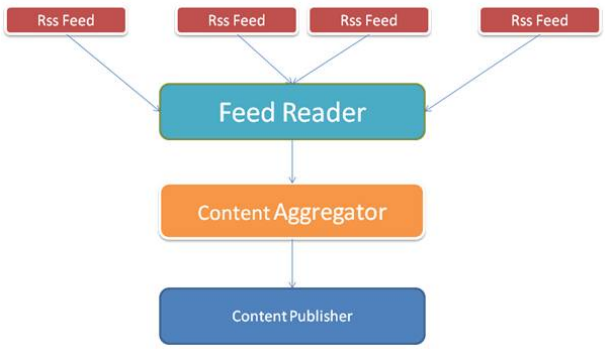


Figure 1: Online Aggregator Flowchart

The user can now read all the similar news at one place collected from different websites Online News Aggregator (ONA) literally analyses documents to find the underlying meaning or concepts of those documents. If each word only meant one concept, and each concept was only described by one word, then ONA would be easy since there is a simple mapping from words to concepts. Figure 2 explains this concept.

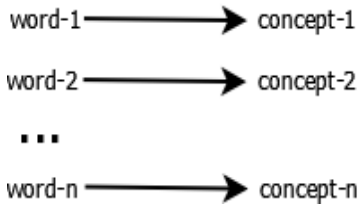


Figure 2: word – concept mapping

Unfortunately, this problem is difficult because English has different words that mean the same thing (synonyms), words with multiple meanings, and all sorts of ambiguities that obscure the concepts to the point where even people can have a hard time understanding. This concept was shown in figure 3.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

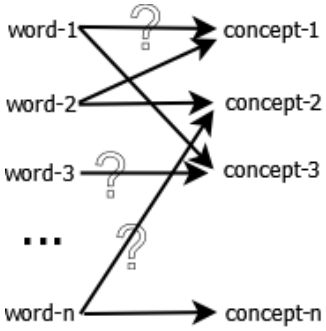


Figure 3: word – concept mapping

For example, the word bank when used together with mortgage, loans, and rates probably means a financial institution. However, the word bank when used together with lures, casting, and fish probably means a stream or river bank.

How Online News Aggregator Works

The fundamental difficulty arises when we compare words to find relevant documents, because what we really want to do is compare the meanings or concepts behind the words. ONA attempts to solve this problem by mapping both words and documents into a "concept" space and doing the comparison in this space.

Since authors have a wide choice of words available when they write, the concepts can be obscured due to different word choices from different authors. This essentially random choice of words introduces noise into the word-concept relationship. ONA filters out some of this noise and also attempts to find the smallest set of concepts that spans all the documents.

In order to make this difficult problem solvable, ONA introduces some dramatic simplifications.

1. Documents are represented as "bags of words", where the order of the words in a document is not important, only how many times each word appears in a document.
2. Concepts are represented as patterns of words that usually appear together in documents. For example "leash", "treat", and "obey" might usually appear in documents about dog training.
3. Words are assumed to have only one meaning. This is clearly not the case (banks could be river banks or financial banks) but it makes the problem tractable.

IV. PSEUDO CODE

The following assumptions are considered while applying Online News aggregator algorithm

- Each News entry in RSS feed is considered as a document
- All the news entries in a RSS feed are Unique

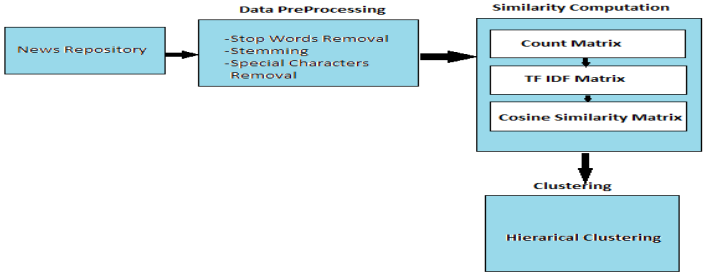


Figure 4: Online News Aggregator

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

Figure 4 shows the functional flow diagram, including the following tasks:

Step 1: Content Preprocessing

Each document is processed for applying ONA algorithm. In this stage all the words that are most frequently used (like it,or,and of) are removed from the document. Generally the words that are not considered in ONA algorithm are prepositions, conjunction. These words are called "Stop words". The following are the stop words that are not considered in ONA algorithm as they play no role in determining the similarity between documents.

Stop Words

A:a','about','above','across','after','afterwards','again','against','all','almost','alone','along','already','also','although','always','am','among','amongst','amoungst','amount','an','and','another','any','anyhow','anyone','anything','anyway','anywhere','are','around','as','at'

B:'back','be','became','because','become','becomes','becoming','been','before','beforehand','behind','being','below','beside','besides','between','beyond','bill','both','bottom','but','by',

C:'call','can','cannot','cant','co','computer','can','could','couldnt','cry'

D:'de','describe','detail','do','done','down','due','during'

E:'each','eg','eight','either','eleven','else','elsewhere','empty','enough','etc','even','ever','every','everyone','everything','everywhere','except'

F:'few','fifteen','fify','fill','find','fire','first','five','for','former','formerly','forty','found','four','from','front','full','further'

G:'get','give','go'

H:'had','has','hasnt','have','he','hence','her','here','hereafter','hereby','herein','hereupon','hers','herse','him','himse','his','how','however','hundred'

I: 'i','ie','if','in','inc','indeed','interest','into','is','it','its','itse'

K: 'keep'

L: 'last','latter','latterly','least','less','ltd'

M:'made','many','may','me','meanwhile','might','mill','mine','more','moreover','most','mostly','move','much','must','my','myse'

N:'name','namely','neither','never','nevertheless','next','nine','no','nobody','none','noone','nor','not','nothing','now','nowhere'

O: 'of','off','often','on','once','one','only','onto','or','other','others','otherwise','our','ours','ourselves','out','over','own'

P: 'part','per','perhaps','please','put'

R: 'rather','re',

S:'same','see','seem','seemed','seeming','seems','serious','several','she','should','show','side','since','sincere','six'sixty','so','some','somehow','someone','something','sometime','sometimes','somewhere','still','such','system',

T:'take','ten','than','that','the','their','them','themselves','then','thence','there','thereafter','thereby','therefore','therein','thereupon','these','they','thick','thin','third','this','those','though','three','through','throughout','thru','thus','to','together','too','top','toward','towards','twelve','twenty','two'

U: 'un','under','until','up','upon','us'

V: 'very','via'

W:'was','we','well','were','what','whatever','when','whence','whenever','where','whereafter','whereas','whereby','wherein','whereupon','wherever','whether','which','while','whither','who','whoever','whole','whom', 'whose', 'why','will','with','within','without','would'

Y : 'yet','you','your','yours','yourself','yourselves'

Z: 'zoo.'

The following special characters are also removed from the document as they have no role in determining the similarity between the documents. Special Characters that are removed: ! @ $ % ^ & * ( ) _ + { } | : " < ? [ ] \ ; ' , . /

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

Step 2: Creating the Count Matrix

The first step is to create the word by title (or document) matrix. In this matrix, each index word is a row and each title is a column. Each cell contains the number of times that word occurs in that title. For example, the word "book" appears one time in title T3 and one time in title T4, whereas "investing" appears one time in every title.

In general, the matrices built tend to be very large, but also very sparse (most cells contain 0). That is because each title or document usually contains only a small number of all the possible words. This sparseness can be taken advantage of in both memory and time by more sophisticated implementations. Figure 5 shows the words-titles plot and figure 6 shows count matrix.

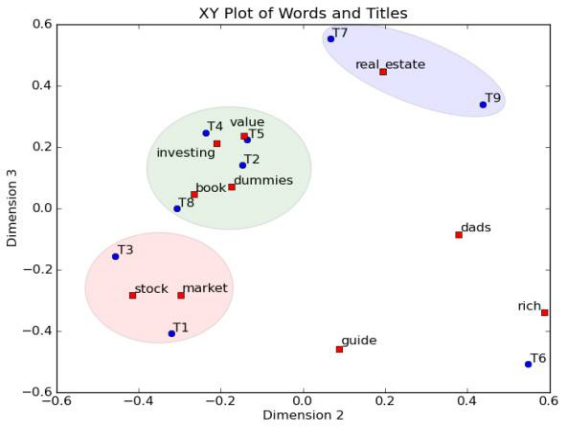


Figure 5: Words–Titles Plot

In the following matrix, we have left out the 0's to reduce clutter.

Index Words	Titles								
	T1	T2	T3	T4	T5	T6	T7	T8	T9
Book			1	1					
Dads						1			1
Dummies		1						1	
Estate							1		1
Guide	1					1			
Investing	1	1	1	1	1	1	1	1	1
Market	1		1						
Real							1		1
Rich						2			1
Stock	1		1					1	
Valve				1	1				

Figure 6: Count Matrix

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

Step 3: Modify the Counts with TFIDF

The raw matrix counts are usually modified so that rare words are weighted more heavily than common words. For example, a word that occurs in only 5% of the documents should probably be weighted more heavily than a word that occurs in 90% of the documents. The most popular weighting is TFIDF (Term Frequency - Inverse Document Frequency). Under this method, the count in each cell is replaced by the following formula.

$TFIDF_{i,j} = (N_{i,j} / N_{*,j}) * \log(D / D_i)$ where

$N_{i,j}$ = the number of times word i appears in document j (the original cell count).

$N_{*,j}$ = the number of total words in document j (just add the counts in column j).

D = the number of documents (the number of columns).

D_i = the number of documents in which word i appears (the number of non-zero columns in row i).

In this formula, words that concentrate in certain documents are emphasized (by the $N_{i,j} / N_{*,j}$ ratio) and words that only appear in a few documents are also emphasized (by the $\log(D / D_i)$ term).

Step 4: Modify the TFIDF matrix with Cosine Similarity

We now have the ability to find related documents. We can test if two documents are in the concept space by looking at the cosine of the angle between the document vectors. We use the cosine of the angle as a metric for comparison. If the cosine is 1 then the angle is 0° and hence the vectors are parallel (and the document terms are related). If the cosine is 0 then the angle is 90° and the vectors are perpendicular (and the document terms are not related).Using the TFidf matrix the Cosine similarity matrix is generated which would give us the similarity factor between any two documents.

Step 5: Clustering Using Hierarchical Clustering algorithm

Cluster Analysis [4],[9][12] also called data segmentation, has a variety of goals. All relate to grouping or segmenting a collection of objects (also called observations, individuals, cases, or data rows) into subsets or "clusters", such that those within each cluster are more closely related to one another than objects assigned to different clusters. Central to all of the goals of cluster analysis is the notion of degree of similarity (or dissimilarity) between the individual objects being clustered. There are two major methods of clustering -- hierarchical clustering and k-means clustering.

In hierarchical clustering [7],[8] the data are not partitioned into a particular cluster in a single step. Instead, a series of partitions takes place, which may run from a single cluster containing all objects to n clusters each containing a single object. Hierarchical Clustering is subdivided into *agglomerative* methods, which proceed by series of fusions of the n objects into groups, and *divisive* methods, which separate n objects successively into finer groupings. Agglomerative techniques are more commonly used, and this is the method implemented in XLMiner. Hierarchical clustering may be represented by a two dimensional diagram known as dendrogram which illustrates the fusions or divisions made at each successive stage of analysis. An example of such a dendrogram is given figure 7.

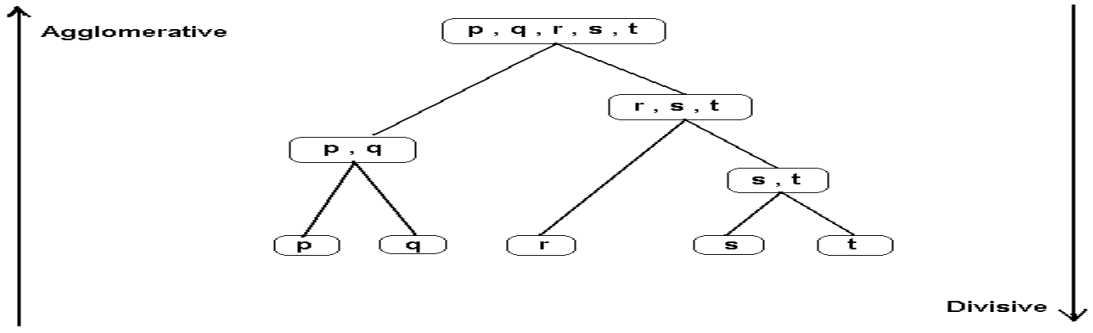


Figure 7: Agglomerative Clustering

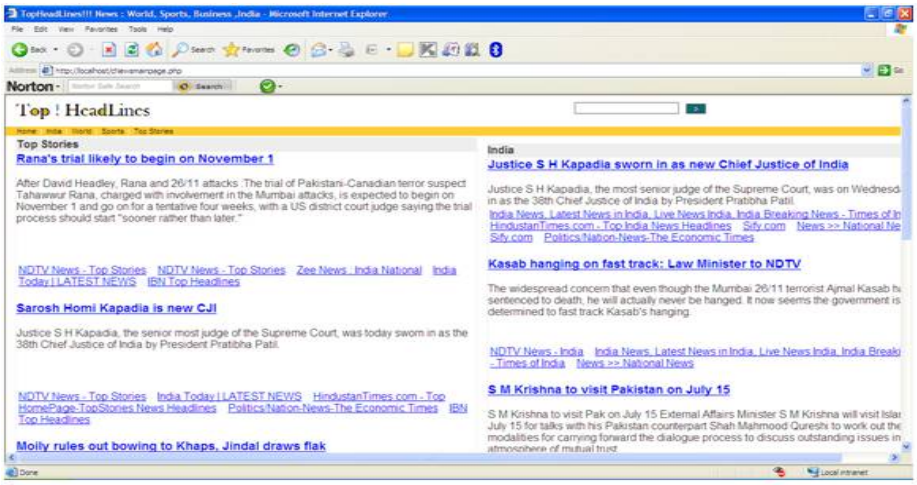
ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

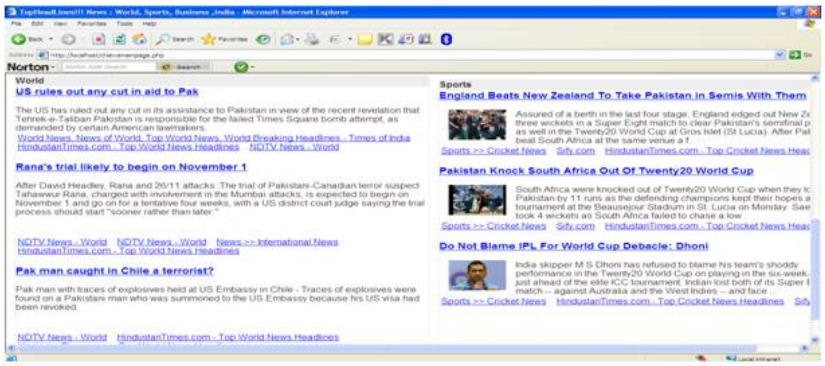
(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

V. RESULTS



The Result of ONA algorithm news website where the user can see the similar news from different news websites grouped together. Display the homepage with top stories and india news in all the categories



The Result of ONA algorithm news website where the user can see the similar news from different news websites grouped together. Display the homepage with world and sports news in all the categories

VI. CONCLUSION AND FUTURE WORK

We proposed online news aggregator by using count matrix a news aggregator to group similar news using through artificial intelligence techniques and clustering. As usual in data analysis, start-up phase requires the setting of several critical parameters. The first phase requires the feed without any human intervention, the parameter setting and techniques calculates the relevant news. The parameters and selection determine the quality of relevant values of news, in future work designer has to carefully evaluate the result and change some parameters to improve the result.

REFERENCES

1. S. Bergamaschi, F. Guerra, M. Orsini, and C. Sartori. Extracting relevant attribute values for improved search. *IEEE Internet Computing*, pages 26–35, Sep-Oct 2007.
2. A. Das, M. Datar, A. Garg, and S. Rajaram. Google news personalization: scalable online collaborative filtering. In Williamson et al. [11], pp.271–280.
3. A. Gulli. The anatomy of a news search engine. In Allan Ellis and Tatsuya Hagino, editors, www(Special interest tracks and posters), pages 880–881. ACM, 2005.
4. M. Halkidi, Y. Batistakis, and M. Vazirgiannis. On clustering validation techniques. J. Intell.Inf. Syst., 17(2-3):107–145, 2001.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

5. X. Li, J. Yan, Z. Deng, L. Ji, W. Fan, B. Zhang, and Z. Chen. A novel clustering-based rss aggregator. In Williamson et al. [11], pages 1309–1310.
6. D.R.Radev, J.Otterbacher, A.Winkel,andS.Blair-Goldensohn.News in essence: summarizing online news topics. Communication. ACM, 48(10):95–98, 2005.
7. P.J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math., 20:53–65, 1987
8. J.B. Kruskal and J.M. Landwehr, Icicle plots: better displays for hierarchical clustering, Amer. Statist. 37 (1983) 162-168.
9. H. Spaeth, Cluster Analysis Algorithms (Ellis Horwood, Chichester, 1980).
10. Data Mining with Big Data , Xindong Wu, Fellow, IEEE, Xingquan Zhu, Senior Member, IEEE, Gong-Qing Wu, and Wei Ding, Senior Member, IEEE –January 2014
11. Big Data Analysis Using HACE Theorem, Deepak S. Tamhane, Sultana N. Sayyad – January 2015.
12. Guha, S., Rastogi, R., and Shim K. (1998). CURE: An Efficient Clustering Algorithm for Large Databases. In Proceedings of the ACM SIGMOD Conference.

BIOGRAPHY

M.Sundara Chinna Network Administrator in the Computer Information Technology Department, National Institute of Rural Development and Panchayati Raj Hyderabad india he received Master of Technology (MTech CST with Specialization in AI&R) degree in 2010 from Andhra University, Visakhapatnam, AP India. His research interests are Computer Networks using AI techniques, Algorithms etc.

K.SrinivasaRao is a Research Scholar in Department of CS&SE, AU College of Engineering, Andhra University. He received M.Tech(CST with Specialization in AI&R) degree in 2010 from Andhra University, AP, India. His research interests are Web mining and Machine Learning

.