Location based Web Object Search using Probabilistic Classification Model

Anjan Kumar K N, Chandrashekar B S

Abstract :The classical Web search engines focus on satisfying the information need of the users by retrieving relevant Web documents corresponding to the user query. The Web document contains the information on different Web objects such as authors, automobiles, political parties e.t.c. The user might be accessing the Web document to procure information about a specific Web object, the remaining information in the Web object [2-6] becomes redundant specific to the user. If the size of Web documents is significantly large and the user information requirement is small fraction of the document, the user has to invest effort in locating the required information inside the document. It would be much more convenient if the user is provided with only the required Web object information located inside the Web documents. Web object search engines provide Web search facility through vertical search on Web objects. In this paper the main goal we considered is the objective information present in different documents is extracted and integrated into an object repository over which the Web object search facility is built.

***Keywords*: Web Object, Web Search Engine, Object Query, Feature Selection.**

I. INTRODUCTION

The Web search engines have been instrumental in providing information from all over the globe to the user The central theme of all Web search engines is to provide the relevant information expected by the user.

The architecture of Web search engine is as shown in Fig 1.1. The user query is give in to to the search engine interface, which is then transformed by the query parser, to remove ambiguity and perform stemming. This transformed query is submitted to the backend engine, which utilizes Web crawlers and indexes to obtain relevant documents. The ranking component performs ranking of these relevant documents by using scoring functions so that, the most user relevant documents are provided.

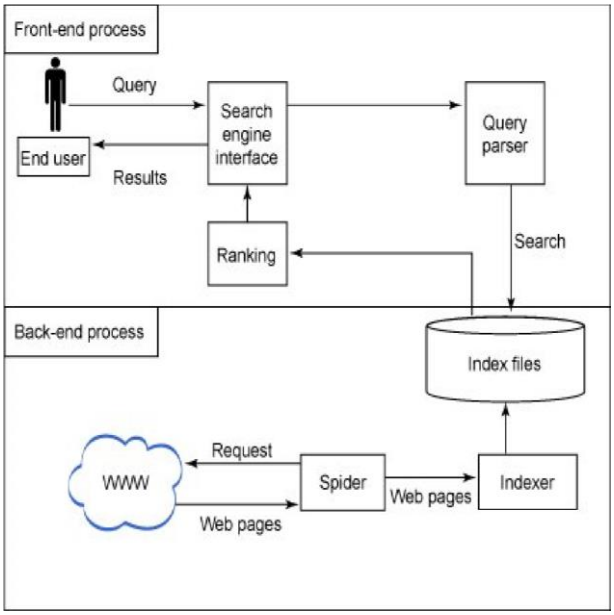


Fig 1.1 Web Search Engine Architecture

object query [3]. But, the documents of the result set might contain additional information along with the required object information. The user has to navigate these documents to find his required information. Clearly, the information which is not specific to the object query in any given document is redundant to the user and also more suitable for the user, if only essential information is delivered.

The first task before classifying documents is to perform feature selection. The feature selection plays a pivotal role in achieving high degree of classification accuracy. If poor features are selected then, the classification accuracy might be extremely low. Also, high dimensionality of text features, and noise inside the documents, can reduce classification accuracy [7]. There are broadly two methods to represent a text document: in the first method, a document is represented as a collection of words, in which the words are assumed to be independent of each other, in the second method, the document is considered as a collection of strings or sequence of words. It is important to evaluate the classifying potential of the selected features so that, the most influential features can be selected.

The Gini index is one of the popular techniques to evaluate feature potential. Let, $p_i(w)$ represent the conditional probability of a document, which has word w, and it belongs

Revised Manuscript Received on December 12, 2019.

*Corresspondace Author

***Anjan Kumar K.N**,Assitant Professor,Department of Computer Science and Engineering,Rns Institute of Technoloy and Engineering,Bangalore,India

Email:anjankn05@gmail.com

Chandrashekhar B.S, Assitant Professor,Department of Computer Science and Engineering,Rns Institute of Technoloy and Engineering,Bangalore,India

Email:samparkisu@gmail.com

to class i. The Equation 1.1, shows the condition that has to be satisfied.

$$\sum_{i=1}^{k} pi(w) = 1 \qquad (1.1)$$

The Gini index $G(w)$ for the word w is shown in Equation 1.2. The Gini index score is always between $\frac{1}{k}$ and 1. A word is said to have higher discriminative power, if it has higher Gini index score.

$$G(w) = \sum_{i=1}^{k} p_i(w)^2 \qquad (1.2)$$

Another important metric to measure the feature potential is Information Gain. The information gain [1] measure for word w given by $I(w)$, is shown in Equation 1.3. Here, P_i indicates the global probability for class i, and $F(w)$ indicates the fraction of documents, that contain word w. Higher values of $I(w)$ indicates, greater discriminatory [15] potential of word w.

$$I(w) = -\sum_{i=1}^{k} P_i \log(P_i) + F(w) \sum_{i=1}^{k} p_i(w) \log(p_i(w))$$

$$+(1 - F(w)) \sum_{i=1}^{k} (1 - p_i(w)) \log(1 - p_i(w)) \qquad (1.3)$$

The overall influence of word w is calculated through average or maximum values of $M_i(w)$, which are shown in Equations 1.4 and 1.5.

$$M_{avg}(w) = \sum_{i=1}^{k} P_i M_i(w) \qquad (1.4)$$

$$M_{max}(w) = \max_i [M_i(w)] \qquad (1.5)$$

This classifier performs condition check on a set of attribute values for the data vector. This condition check is performed in a hierarchical manner such that, after reaching the end of the decision tree, the data vector would be assigned to a certain class. For performing document classification using decision tree classifier [13], the attributes are assumed to be certain words in the document.

The two important metric to design effective rules are: support confidence and support. The confidence metric identifies the number of documents inside the training set, which satisfy both the right hand side and left hand side of any given rules. The support metric identifies the number of documents inside the training set, which are relevant to the rule.

The probabilistic classifiers [2] assign probability for each class that a document might belong. The highest probability class is assigned as the designated class for the document. The naive Bayes classifier is one of the earliest probabilistic classifier. It assumes independence among the terms of the document. Two models are used to build this classifier: multivariate Bernoulli model and multinomial model. These models ignore the position of the words. In the first model, the frequencies of the terms are ignored, and the presence and absence of certain terms are used as features to represent the documents. In the second model, the frequencies of terms are utilized as features to represent the documents.

The user query is also modeled as a vector. The Bernoulli model for the naive Bayes classifier is shown in Equation 1.6. Here, i is the i^{th} class, Q is the bag of words for a given document. The Equation 1.6 can be rewritten as Equation 1.7. The value of $P(i)$ is assigned as the fraction of documents inside the repository, which belongs to class i, $P(t_j \in T|i)$ is the fraction of documents in the repository, which contain term t_j, and belong to the class i. Here, T is the term index built for the whole document repository.

$$\hat{\imath} = \underset{i}{argmax}\, P(i|Q) \qquad (1.6)$$

$$\hat{\imath} = \underset{i}{argmax}\, P(i) \times \prod_{t_j \in Q} P(t_j \in T|i) \times \prod_{t_j \in Q} \left(1 - P(t_j \in |i)\right) \qquad (1.7)$$

The support vector machine (SVM) is 2 class classifiers, which attempt to provide maximum separation between the 2 classes. The Fig 3.1 illustrates the SVM technique. There are 3 different hyper planes which separate the 2 classes of document vectors. The SVM technique, selects the hyper plane A, because it provides the maximum separation between the 2 classes.

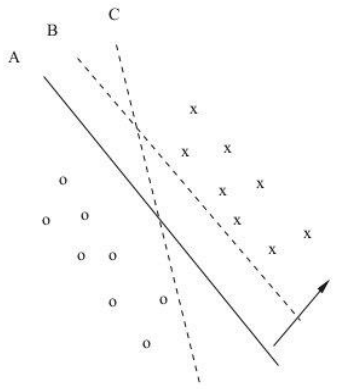


Fig 3.1 SVM Technique

II. RELATEDWORK

A Web search engine been built, which collected documents uniformly from different geographical locations [2-7]. So that, the query results that can cover different geographical locations can be provided to the user. But, this system did not address the problem of automatically tagging geographical locations to Web documents.

The Web document blocks were used to develop temporal and spatial topic distribution Gao *et al.* (2006). Each Web document block had a spatial label to refer the geographical location affinity. The geographical locations of different countries and state were utilized. But this work, only applies to Web search engines.

The preliminary work on Web objects of geographical labeling was performed. The Gaussian mixture model (GMM) was utilized. Multiple reasons for utilizing GMM are:

i. The relevance levels are a function of geographic coordinates.
ii. The relevance level has multiple peaks.
iii. The relevance level continuously decreases, when distance from peak increases.
iv. The relevance level decrease is not necessarily isotropic.

III. PROBLEM STATEMENT

Let, $(o_1, o_2, \ldots, o_N)$ represents a Web object set, here each object assigned to appropriate location label. The name of the location is appearing in the sequence of the subsequent Web object which is utilized to generate the feature vector. This feature vector is shown in Equation 3.1. In this, $\boldsymbol{d_j}$ is analogous feature vector for the object o_j, l is the most number of location names considered for the labeling process, $p_i(1 \leq i \leq l)$ is a particular location name and $tf(p_i)$ is the term frequency of p_i in o_j. The quandary addressed here is to label a appropriate location name to $\boldsymbol{d_j}$, is corresponding to the real location identifier for $\boldsymbol{d_j}$.

$$\boldsymbol{d_j} = \begin{bmatrix} tf(p_1) \\ tf(p_2) \\ . \\ . \\ . \\ tf(p_l) \end{bmatrix} \quad (3.1)$$

IV. PROPOSED MODEL APPROACH

A. *4.1 Variation Implication Categorization Model*

This categorization model uses the solidity function of the data point which is shown in Equation 4.1. The working of the solidity function is shown in Equations 4.2 and 4.3.

$$log\mathrm{P}(d_j) = L(q) + KL(q||P) \quad (4.1)$$

$$L(q) = \int_{x=1}^{l} q(x)\, log\left(\frac{P(x, d_j)}{q(x)}\right) dx \quad (4.2)$$

$$KL(q||P) = -\int_{x=1}^{l} q(x)\, log\left(\frac{P(x|\, d_j)}{q(x)}\right) dx \quad (4.3)$$

The functional $L(q)$ to be maximized w.r.t $q(x)$ and the functional value $\hat{q}(x)$ that maximizes $L(q)$ is considered as the estimate for $P(x|\boldsymbol{d_i})$. It is denoted in Equation 4.4. The estimation procedure assumes normal distribution for the joint density of $(x, \boldsymbol{d_j})$. This case is shown in Equation 4.5.

$$\hat{q}(x) \approx P(x)|\ d_j \quad (4.4)$$

$$P(x, d_j) = N(x, d_j|\mu, \Sigma) \quad (4.5)$$

Normal distribution function parameters shown in Equation 4.5 is predictable through maximizing the likelihood function w.r.t μ and Σ. Equation 4.6 shows maximization training set utilization procedure. The values $\hat{\mu}$ and $\hat{\Sigma}$ are considered as the estimated values of μ and Σ.

$$\underset{\mu,\Sigma}{argmax}\ log\ likeihood\ function = log \prod_{i=1}^{n} N\,(x, d_i|\mu, \Sigma) \quad (4.6)$$

B.

The labeled data point which consists of highest probability class label

C. *4.2 Model Selection*

Model scores are utilized to select the suitable model.

Equation 4.1 is used for the most suitable model for quantify and data point class labeling per formation. Here divided training set into two exclusive subsets. In first subset two models were used for training purpose. The second subset pairs is utilized for r ordered, to calculate the model score. The model m score, which indicates labeling effectiveness of m to perform class labeling of data points is indicated by model score(m). Lower model score of model m gives improved accuracy to predict class labels of data points. Here, x_j indicates the class label for data point predictor $\mathbf{d_j}$ through the utilization of m, and x_j indicates the real class label. The accuracy function present in Equation 4.1 is shown in Equation 4.2.

$$model_{score(m)} = \frac{\sum_{j=1}^{r} accuracy(\bar{x}_j, x_j)}{r} \quad (4.1)$$

$$accuracy(\bar{x}_j, x_j) = \begin{cases} 0 & if\ \bar{x}_j = x_j \\ 1 & if\ \bar{x}_j \neq x_j \end{cases} \quad (4.2)$$

The metric Labeling Accuracy shown in Equation 4.3 is used in the result investigation, some part of Web objects in the test set that were exactly classified. Here, accurate labeling foe metric is shown in Equation 4.4. Total labeling performance time metric is shown in Equation 4.5. It shows the total time taken by the labeling model to label all the test set documents. In this labeling time($\boldsymbol{d_j}$) specifies the time taken by the labeling model to label $\boldsymbol{d_j}$.

$$Labeling\ Accuracy = \frac{\Sigma_{d_j \epsilon test\_set} accurate\_labeling(d_j)}{|test\_set|} \quad (4.3)$$

$$accurate\_labeling(d_j) = \begin{cases} 0 & if\ d_j\ is\ accurately\ classified \\ 1 & otherwise \end{cases} \quad (4.4)$$

$$Total\ Labeling\ Time = \sum_{d_j \epsilon test\_set} labeling\_time(d_j) \quad (4.5)$$

D. Here we create Web object repository by extracting the relevant object information from different Web documents.

The below proposed algorithm outlines the geographical labeling model, in which the training set holds n ordered pairs. The function Split (Training Set) divides the training set into two equally limited sub-sets indicated by T_1 and T_2. Here, T_1 and T_2 holds n − r and r ordered pairs correspondingly. Where $\hat{w}$ Bayesian classification and $\hat{q}$ estimates the Variation Inference classification model

Suppose, $srer e_B > score_V$ the Bayesian classification model is considered for labeling of web object performance which is labeled as class x. The classification distance$(x, \boldsymbol{d_j})$ is used for calculating the distance of each label through by considering the lowest classification distance which will be the chosen location label of $\boldsymbol{d_j}$.

Similarly, if $score_B < score_V$ the Variation Inference classification model is considered for labeling of web object performance which is labeled as class x. The value $\hat{q}(x)$ is calculated for each class, and the highest value for $\hat{q}(x)$ is chosen as the location label of $\boldsymbol{d_j}$.

Algorithm: - Geographical labeling of Web objects Technique.

split(Training set)
$\hat{w} = Bayesian\_model\ (T_1)$
$\hat{q} = Variation\_model\ (T_1)$
$score_B = model\_score(B, T_2)$
$score_v = model\_score(V, T_2)$
$if\ score_B > score_V$ then
$for\ x = 1$ to l do
$cd(x) = classificationdistance(x, d_j)$
end for
$class\_label(d_j) = \min_x cd(x)$
end if
$if\ score_V > score_B$ then
$for\ \bar{x} = 1$ to l do
$Pr(x) = \hat{q}(x)$
end for
$class\_label(d_j) = max_x Pr(x)$
end if

$\hat{w} = score_B$
$\hat{q} = score_V$

V. OBSERVATIONS AND RESULTS

The proposed probabilistic Web object labeling model (it will be referred as label-new) is matched with current proposed labeling model (it will be referred as label-old). Web objects of size 1000 were considered for training and this is further divided into two subsets. The web objects of size 500 called T_1 were used for training, ie label new and old. The web objects of size 500 were called T2 used to select the appropriate model for label-new. Objects of size 2000 were used as test set in examining the old and new label performance by considering 10 to 50. This metric produces values between 0 to 1, where 0 indicates inaccurate and 1 indicates accurate Web objects classification. The result analysis w.r.t. DBLP and IMDB object repository is presented.

The first trial examines the performance of label-old and used by considering the maximum location names. The accuracy of label is shown in Fig 5.1, and the values are shown in Table 5.1.

The label-new provides best accuracy because of its probabilistic design there is an variation in performance of label-new. The result is obtained w.r.t were analyzed w.r.t which is based on all the test set feature

vector model shown in Fig 5.2 and the values are shown in Table 5.2. GMM is used to perform classification.

The second trial examines the training period for both label-new and label-old by using the maximum number of different locations training set. The experimental results are shown in Fig 5.3, and values are shown in Table 5.3

Table 5.1 Accuracy vs No of Places (DBLP)

No of Tuples	Labeling Accuracy (label-old)	Labeling Accuracy (Label-new)
10	0.4	0.71
20	0.42	0.73
30	0.41	0.82
40	0.51	0.75
50	0.53	0.65

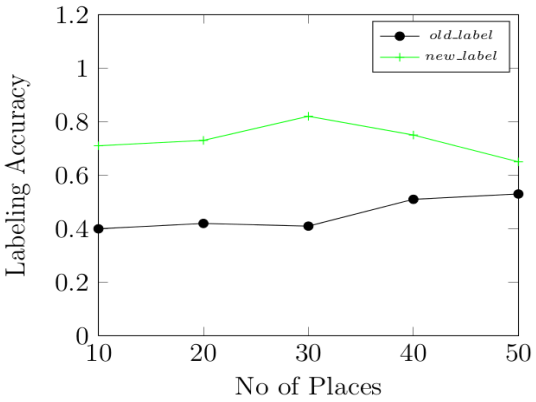


Fig 5.1: Accuracy vs No of Places (DBLP)

Table 5.2 Exe Time vs No of Places (DBLP)

No of Places	Total Labeling Time(label-old)(s)	Total Labeling Time(label-new)(s)
10	33	15
20	34	23
30	41	27
40	48	28
50	51	31

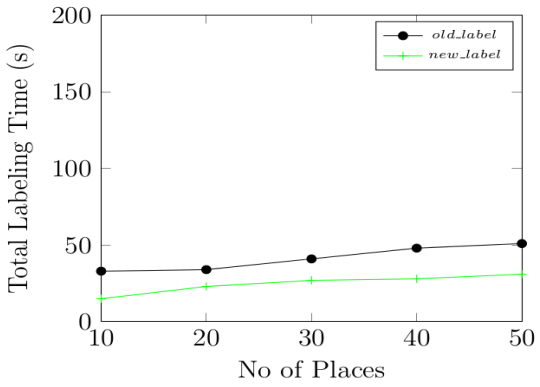


Fig 5.2 Ee Time vs No of Places (DBLP)

Table 5.3 Training Time vs No of Places (DBLP)

No of Places	Total Training Time (label-old) (s)	Total Training Time (label-new) (s)
10	43	14
20	44	17
30	46	18
40	51	20
40	51	20

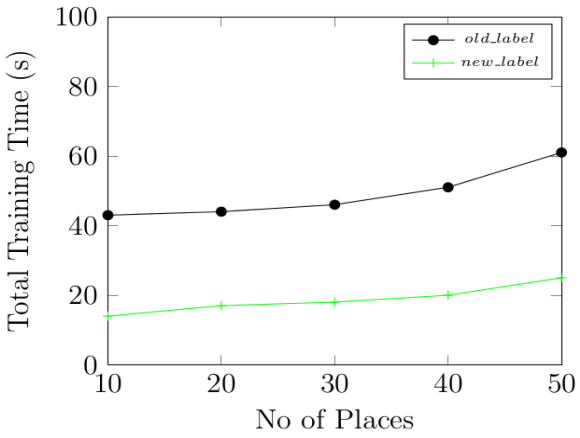
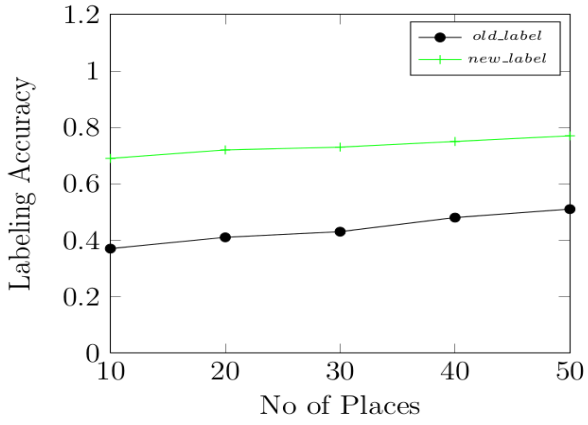


Fig 5.3 Training Time vs No of Places (DBLP) Noise Rate (%)

The result investigation w.r.t. IMDB is shown in Fig 5.4, 5.5 and 5.6; similarly, the corresponding analysis tabulated values are shown in Tables 5.4, 5.5 and 5.6.

Table 5.4 Accuracy vs No of Places (IMDB)

No of Places	Labeling Accuracy (label-old)	Labeling Accuracy (label-new)
10	0.37	0.69
20	0.41	0.72
30	0.43	0.73
40	0.48	0.75
50	0.51	0.77



.Fig 5.4 Accuracy vs No of Places (IMDB)

Table 5.5 Exe Time vs No of Places (IMDB)

No of Places	Total Labeling Time (label-old) (s)	Total Labeling Time (label-new) (s)
10	37	16
20	44	21
30	45	28
40	48	32
50	52	36

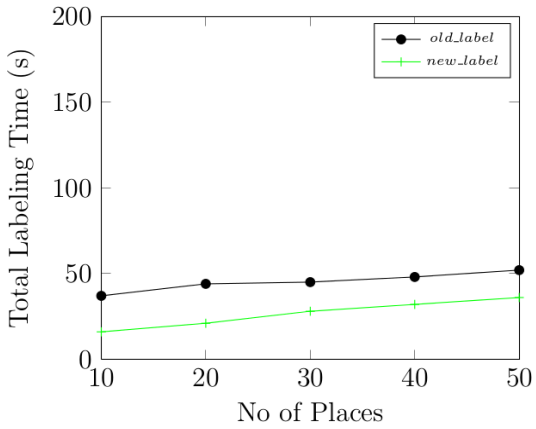


Fig 5.5 Exe Time vs No of Places (IMDB)

Table 5.6 Training Time vs No of Places (IMDB)

No of Places	Total Training Time (label-old) (s)	Total Training Time (label-new) (s)
10	45	14
20	49	16
30	56	19
40	59	23
50	64	25

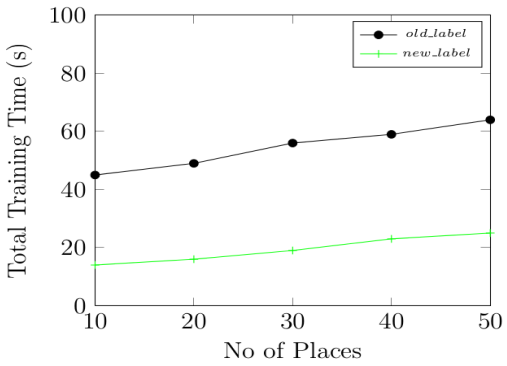


Fig 5.6 Training Time vs No of Places (IMDB)

VI. CONCLUSION

In this Work the importance of Web object search engine is presented, and also importance of location based Web object search is described, major techniques for achieving location based search and probabilistic classification model were presented, The empirical results of all the proposed technique were demonstrated over real world data sets, the proposed techniques exhibit appreciable performance w.r.t. user relevance and execution efficiency when compared with contemporary model.

REFERENCES

1. Buyukkokten, O, Cho, J, Garcia-Molina, H, Gravano, L & Shivakumar, N 1999, 'Exploiting geographical location information of Web pages', In proceedings of the ACM SIGMOD Workshop on the Web and databases, Philadelphia, Pennsylvania.

2. Anjan Kumar, K N, Chitra, S & Satish Kumar, T 2018, 'Probabilistic Cluster Classification Techniques to Perform Geographical Labeling of Web Objects' published in Cluster Computing, print ISSN 1386-7857.
3. Davis, CA & Fonseca, FT 2007, 'Assessing the certainty of locations produced by an address geo coding system', Geoinformatica, vol. 11(1), pp.103-129.
4. Sengar, V, Joshi, T, Joy Prakash, S & Toyama, K 2007, 'Robust location search from text queries In proceedings of the 15th', Annual ACM International Symposium on Advances in Geographic Information Systems, Seattle, Washington, Article No 24.
5. Bernardini, A, Carpineto, C & D'amico, M 2009, 'Full-subtopic retrieval with keyphrase-based search results clustering', In Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence. IEEE Computer Society, pp. 206–213.
6. Amitay, E, Har, EN & Sivan, R, 'Soffer A geotagging web content , In proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, United Kingdom, pp.273-280.
7. Amati, G, Carpineto, C & Romano, G 2001, 'Fubat TREC-10 Web Track: A probabilistic framework for topic relevance term weighting', In Proceedings of the 10th Text REtrieval Conference (TREC'10). NIST Special Publication 500–250. National Institute of Standards and Technology (NIST), Gaithersburg, MD, pp. 182–191.
8. Besnik Fetahu, Ujwal Gadiraju & Stefen Dietze 2015, 'Improving Entity Retrieval on Strucutred Data in proceedings of the14th', International Conference on The Semantic Web-ISWC.
9. Carpineto, C, Osinski, S, Romano, G & Weiss, D 2009, A survey of Web clustering engines. ACM Comput. Surv. vol. 41(3).
10. Dempster, A, Laird, N & Rubin, D 1977, 'Maximum likelihood from incomplete data via the EM algorithm', J. Royal Statist. Soc. Series B (Methodological) vol. 39(1), pp. 1–38
11. Aliaksandr Talaika, Jonna Biega, Antonie Amarillii & Fabin M Suchanek 2015, 'IBEX: Harvesting Entities from the Web using identifiers In proceedings of the 18th', International Workshop on Web and Databases, May 31-june 04,2015,
12. Melbourne, VIC, Australia.Amati, G 2003, 'Probabilistic models for information retrieval based on divergence from randomness', Ph.D. thesis, Department of Computing Science, University of Glasgow, UK.
13. Manning, CD, Raghavan, P & Schutze, H 2008, 'Introduction to Information Retrieval', Cambridge University Press. MARON, M. E. AND KUHNS, J. L. 1960. On relevance, probabilistic indexing and information retrieval. J. ACM vol.7(3), pp. 216–244.
14. Mei Q Liu, C, Su, H & Zhai, C 2006, 'A probabilistic approach to spatiotemporal theme pattern mining on weblogs', In proceedings of the 15th International World Wide Web Conference, Edinburgh, Scotland, pp.533-542.
15. Taro Tezuka, Hiroyuki Kondo & Kastumi Tanaka 2008, 'Estimation of Geographic Relevance for Web objects Using Probabilistic Models Springer-Verlag Berlin Heidelberg'.
16. Taro Tezuka, Hiroyuki Kondo & Kastumi Tanaka 2008, 'Estimation of Geographic Relevance for Web objects Using Probabilistic Models Springer-Verlag Berlin Heidelberg'.

AUTHORS PROFILE

Dr. Anjan Kumar K N working as Assistant Professor in the Department of Computer Science and Engineering R N S Institute of Technology, Bangalore. He has a B.E. degree in Computer Science and Engineering from Bangalore University in 2000 and an M.Tech from the VTU, Ph.D.in Faculty of Information and Communication Engineering from the Anna University in 2019. His research primarily focuses on Web Mining, Network Security and Computer networks.

Chandrashekar B S working as Assistant Professor in the Department of Computer Science and Engineering R N S Institute of Technology, Bangalore. He has a B.E. and M.Tech degree in Computer Science and Engineering from VTU His research primarily focuses on Web Mining, IoT and Cloud Computing.