Data Extract: Mining Context from the Web for Dataset Extraction

Ayush Singhal and Jaideep Srivastava

Abstract—**In this paper we address the problem of dataset extraction from research articles. With the growing digital data repositories and the demand of data centric research in data mining community, finding appropriate dataset for a research problem has become an essential step in scientific research. But given the wide variety of data usage in scientific research it is very difficult to figure out which datasets are most useful for a particular research topic. To alleviate this problem, an automated dataset search engine is a powerful tool. In this work we propose a novel approach to extract dataset names from research articles. We propose a novel way of using "web intelligence" from academic search engines and online dictionaries to mine dataset names from research articles. We also show a comparison between different sources of "web knowledge" by comparing different academic search engines such as Google scholar, Microsoft academic search. The performance of this approach is evaluated using standard information retrieval metric such as precision, recall and F-measure. We get an F-measure of 80%. This accuracy is significant for an unsupervised approach.**

Index Terms—**Dataset, information retrieval, web mining, search engines.**

I. Introduction

The abundance of data availability through many sources such as sensors, social media (Facebook, Amazon and Flickr to name a few), simulations, has led to a massive data-driven research deluge in several sciences and in particular computational sciences. With the present scenario, data-driven scientists, working to establish or verify some theories or algorithms use these real world dataset to verify evaluate their findings. However, in the present "information age" when digital libraries and databases are ever expanding with data being collected from all walks of life, finding the most appropriate datasets for a research problem is a hard problem.

Under such a situation, several dataset repositories have been developed and made public for researchers. However, getting information about dataset usage involves a keyword search or manually going through the details of the works that have used the datasets. An alternative solution to the problem of finding datasets would be to use scientific research articles. The research articles published in conferences and journal in most cases refer to the dataset being used for experiments. These articles are best source to find a dataset usage in relation to the topic of research. However, given the massive amount of research articles in digital libraries, scanning entire paper for dataset names using some supervised classification algorithm would be a cumbersome task. Previous efforts in finding important words such as keywords have used the conventional TF-IDF based weighing approaches and used supervised classification techniques for finding desired keywords [1], [2], [3]. However, such systems have inherent disadvantages because of the supervised training approach used for learning- not generalisable, performance dependent on training set, ineffective in real time applications. The other alternative is to develop unsupervised techniques to automatically mine dataset names without scanning the entire document.

In this work we propose a novel unsupervised approach to find datasets from research articles. We have used "web *intelligence*" from academic search engines such as Google scholar and Microsoft academic research search engines. Such academic search engines provide the information about various research articles in an organized form which can be le-averaged for mining knowledge. In this work we have used the academic search engines to provide exogenous context for mining desired items. The context provided by the web using the search engines has been shown to be more informative than the local context generated from a single document [4]. We exploit this advantage of the web through the search engines to extract dataset names from a research articles.

The proposed approach was tested using different search engines to see what difference exists among the different search engines. We also evaluated the performance of the approach for different search engines using standard information retrieval metrics such as precision, recall and F-measure. The results thus obtained using the proposed approach show that the proposed approach is a promising. We get an average F-measure of 80% using the proposed approach.

The proposed approach is applicable in the real world situation when there are organized libraries of research articles categorized by their research topics. Porting our algorithm to such an environment will form a dataset search engine to give dataset names for a queried research topic. However, in this work we do not propose a dataset search engine but our work contributes towards an important intermediate step for an automated dataset search engine.

The rest of the paper is organized in the following way. In Section II, we discuss the related work. In section III we mathematically describe the problem proposed in this work. Then in Section IV and V, the proposed approach is explained in detail. The experimental results and discussion follow in Section VI. Finally, Section VII concludes the work and describes directions of future research.

Manuscript received February 5, 2013; revised April 17, 2013.

The authors are with CS department at University of Minnesota, Minnesota, MN 55414, USA (e-mail: ayush@ cs.umn.edu, srivas@ cs.umn.edu).

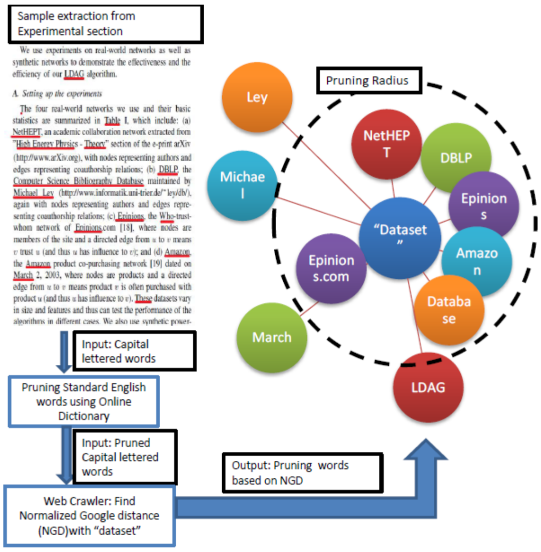
DOI: 10.7763/IJMLC.2013.V3.306

II. Related Work

The background literature for this work can be divided into two sub headings. One sub sections corresponds to state of art in dataset extraction or keyword extraction. The second sub section reviews the various techniques developed to use “web intelligence” in information retrieval.

One class of keyword extraction techniques are based on keyword matching or Vector Space models with basic TF-IDF weighting [5]. The TF-IDF weighting is obtained by using only the content of the document itself. Then several similarity measures used to compare the similarity of the two documents based on their feature vectors [6]. The other class is based on using context information to improve keyword extraction. Recently, there has been lot of work on developing different machine learning methods to make use of the context in the document [7]. Zhang *et al.* [3] discusses the use of support vector machines for keyword extraction from documents using both the local and global context. There are number of techniques developed to use local and global context in keyword extraction [3], [7], [8]. The other class of techniques used to enhance information retrieval uses concepts of semantic analysis such as ontology based similarity measures [9], [10]. In these approaches the ontology information is used to find similarity between words and find words even if the exact match is not available. Other ways in which semantic information is extracted is using Wordnet libraries. Wordnet based approaches have used concepts such as relatedness of words for information retrieval [11]-[14].

The second category of literature which is exists is the various uses of “web intelligence” in information retrieval. Croft et al. in his book [15], describes the various uses of search engines in information retrieval. Recent works [16] have shown the use of encyclopedic knowledge for information retrieval. Lian et al [17], describe the use of Google distance to find concept similarity. Google distance based approaches have been used in various applications such as relevant information extraction [4], [18], keyword prediction [19], and tag filtering [20]. However, the concept of using “web intelligence” in dataset extraction has not been discussed in the literature.


Sample extraction from Experimental section
Input: Capital lettered words
Pruning Standard English words using Online Dictionary
Input: Pruned Capital lettered words
Web Crawler: Find Normalized Google distance (NGD)with “dataset”
Output: Pruning words based on NGD
Pruning Radius
Ley
NetHEPT
DBLP
Michael
Epinions
“Dataset”
Amazon
Epinion s.com
Database
March
LDAG

We use experiments on real-world networks as well as synthetic networks to demonstrate the effectiveness and the efficiency of our LDAG algorithm.

A. Setting up the experiments

The four real-world networks we use and their basic statistics are summarized in Table I, which include: (a) NetHEPT, an academic collaboration network extracted from "High Energy Physics - Theory" section of the e-print arXiv (http://www.arXiv.org), with nodes representing authors and edges representing coauthorship relations; (b) DBLP, the Computer Science Bibliography Database maintained by Michael Ley (http://www.informatik.uni-trier.de/~ley/db/), again with nodes representing authors and edges representing coauthorship relations; (c) Epinions, the Who-trust-whom network of Epinions.com [18], where nodes are members of the site and a directed edge from u to v means v trust u (and thus u has influence to v); and (d) Amazon, the Amazon product co-purchasing network [19] dated on March 2, 2003, where nodes are products and a directed edge from u to v means product v is often purchased with product u (and thus u has influence to v). These datasets vary in size and features and thus can test the performance of the algorithms in different cases. We also use synthetic power-

Fig. 1. Flow Chart explanation of DataExtract algorithm

III. Problem Statement

Given a corpus of research papers *C*, the objective of the problem is to find dataset list $D=<d_1, d_2, d_3, \ldots d_k>$ from all the papers in the corpus *C*. The total number of datasets in *D* can be greater than *N*, the number of research papers in the corpus *C* when some papers in the corpus use more than one dataset for experimentation.

If each paper uses only one dataset then $k=N$ otherwise, $k>N$. The proposed approach will extract a dataset list *D*’ for the corpus and the results compare the extracted dataset list *D*’ with the original list *D*.

IV. Normalized Google Distance

In this section we show a use case where search engines can be used to generate exogenous context using Normalized Google Distance (NGD) [21]. The NGD is used to measure distance between two words which appear in the web pages. The formula for Google distance is given as:

$$NGD\,(x, y) = \frac{\max(\log\ f(x), \log\ f(y)) - \log\ f(x, y)}{\log\ M\ - \min(\log\ f(x), \log\ f(y))}$$

where, *M* is the total number of web pages searched by Google; $f(x)$ and $f(y)$ are the number of hits for search terms x and y, respectively; and $f(x,y)$ is the number of web pages on which both *x* and *y* occur.

If the two search terms *x* and *y* never occur together on the same web page, but do occur separately, the normalized Google distance between them is infinite. If both terms always occur together, their NGD is zero, or equivalent to the coefficient between *x* squared and *y* squared [21].

V. Proposed Approach

In this section we discuss a novel algorithm, DataExtract, for dataset extraction form the papers in the corpus. The novelty of this approach is its unsupervised nature and usage of world knowledge from sources such as online dictionary and academic search engines. The approach is pictorially represented in the flow diagram (Fig. 1). There are 5 steps involved in this approach and they are described as follows:

- *File conversion*: In the first step, the pdf files of the papers are converted into text file. Working directly with pdf files is not a feasible solution because it is difficult to parse text from pdfs. So it is a general approach to convert pdf to text files for any text processing that needs to be done,
- *Content reduction*: The second step is basically the appropriate section selection step. A research paper is an organized document which contains several sections but all the sections might not be relevant for the proposed problem of dataset name extraction. Thus we select only some sections of the papers to be parsed further.

It is a general practice among researchers to give a description of datasets in the experimental section of their work. We use this general observation as our hypothesis for section selection. Thus we select sections which started with heading such as 'Experiments', 'Results', 'Evaluation' and other similar common terms used for experiment section in research articles. An example of such an extract is given in the Fig. 3.

- *Pre-processing:* After the appropriate sections are extracted from the paper, the next step is to do basic natural language pre-processing such as pruning of stop words, removal of full stop and commas etc. The pre-processed text version is denoted by P_o.
- *Candidate term selection*: In this step, the capital lettered terms/words were extracted from Po. These words forms a refined candidate set for "dataset name". We call this candidate set of words as CD_1. For the example explained in the Fig. 1, these words are highlighted by underlining these words.
- *Dictionary based pruning:* The candidate words obtained from step 4 are further pruned using online English dictionary. For this step we have assumed that most of the "dataset names" used in referred in research works are non-standard English terms. Thus such names should be an outlier for a dictionary. The words from CD_1 which are found in the dictionary can thus be pruned from CD_1. The output of the dictionary based outlier pruning is a new candidate set CD_2.
- *NGD computation:* Once we get the pruned candidate set. Then we find the NGD of each word in this candidate set with the term "dataset". As described earlier, NGD gives a quantitative distance of each word with the term "dataset". In real world this is an estimate of the number of documents in which the candidate word and the term dataset are used together. For the NGD computation we have used the academic search engines because they contain the most relevant documents in the database. We have not used the general search engines because a general search engine will also return results taking into account non-relevant documents.
- *NGD radius based pruning:* After obtaining the NGD of each word in CD_2, the final step is to prune some of the irrelevant words from CD_2. In order to determine this parameter (λ, the pruning radius) we use portion of the dataset to find the optimal value of the pruning radius. Then the pruning is done using this radius (as sown in Fig 1). The words from CD_2 which fall inside this radius are the predicted dataset names and rests are dropped. Thus, for the example in the figure, our final output for the prediction is "NetHEP", "DBLP", "Epinions", "Amazon", "Database". The remaining terms are dropped. The steps from 1 through 7 are repeated for each paper in the corpus C and the final output is the predicted dataset list D'.

VI. Experimental Results

In this section we evaluate the performance of the proposed approach on two academic search engines namely, Google Scholar (GS) and Microsoft academic search (MAS. We want to compare and discuss the performance using different search engines and propose what kind of search engines are most suited for dataset extraction in real application. In order to evaluate the proposed approach we have used real dataset. The dataset was constructed as described below.

A. Dataset Description

In order to evaluate the performance of the DataExtract algorithm proposed in this work, the following dataset was constructed. We have selected 50 research papers from various computer science conferences such as KDD, ICDM, WWW published between 2000 till 2012. The collection of these 50 papers is the corpus on which the Data Extract algorithm is tested. In order to construct this dataset we considered only full length papers and excluded any workshop or poster papers from the corpus. Also, in order to reduce the search space we have removed papers which did not experimented on real datasets. The test dataset consists of dataset names associated with each paper.

B. Evaluation Metrics Used

As described in the last section, we have used 50 papers from the corpus C to extract dataset names from these papers. For each paper we get a list of dataset names that were extracted from the paper. We have used the standard evaluation metrics such as precision, recall and F-measure. In the standard information retrieval terminology, these metrics are defined as follows:

Precision (P): The ratio between the number of relevant items in retrieved items and the total number of retrieved items. Items here mean the dataset names. This is computed for each of the test paper pi and then averaged for all the papers to get an average precision.

Recall (R): The ratio between the number of relevant items in the retrieved items and the total number of relevant items. Recall is computed for each of the test paper pi and then averaged for all the papers to get an average recall.

F-measure (F): A measure that combines precision and recall is the harmonic mean of precision (P) and recall(R). The F-measure is computed using the average precision and average recall values.

$$F = 2.\frac{(precision \times recall)}{(precision + recall)}$$

C. Results and Discussion

Table I summarizes the performance of the dataset extraction algorithm using two different search engines namely, GS and MAS in terms of precision, recall and F-measure. The precision, recall and F-measure values in this table are computed for datasets in 25% of the total papers considered for this experiment. As described earlier, the pruning radius (λ) goes in as the parameter for this algorithm.

We determine the optimal value of this radius λ by constraint maximization. In order to do this, the original test data (consisting of 50 papers) is divided into two parts. The first part, the training set, consists of 75% of the total papers and the test set consists of the remaining 25% of the papers. Once the parameter λ is computed from the training set, we evaluate the algorithm's performance on the test set. In the experiment we also show the difference in the λ obtained from GS and MAS.

As shown in Fig. 2, the precision, recall and the F- measure values are computed for different values of λ (ranging from 0.1 to 1.0) on the training set. From this figure, we can observe how the precision, recall and F-measure value change as λ is increased from 0.1 to 1.0. The precision for small λ tend to be as high as 100% because very less datasets were extracted at this radius. Since the recall is low at this radius.

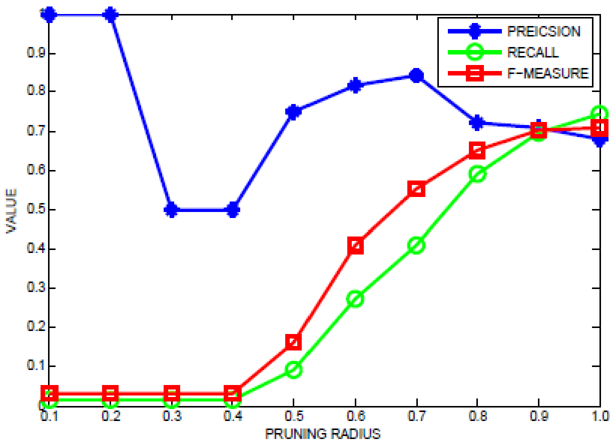


Fig. 2. Precision and recall curve for different values of pruning radius for GS search engine.

We can say that the number of correct datasets extracted were not significant compared to the total number of original datasets in the training set.

On increasing the pruning radius the precision drops significantly. This shows that the small pruning radius is not suitable for this algorithm.

As the pruning radius is further increased the precision values get stabilized. Finally, the precisions starts decreasing at λ = 0.7. The optimal pruning radius is determined by the intersections point of the precision, recall and F-measure curves. The optimal value estimated for λ is 0.9. We then use this value of λ for evaluating the performance on the test set and the results are shown in the Table I. The values of precision, recall and F-measure are greater than 80 %.

Similarly, the Fig. 3 shows the precision, recall and F-measure curve for MAS search engine. However, the pruning radius variation for MAS starts from 0.4 to 1.0 because at a smaller radius then this no information was retrieved. But if we compare the two curves, we observe some similarities and differences. One similarity in the two plots is that in both the plots, the precision curve after λ=0.4 first increases and then decreases. Although the precision at λ=0.4 for MAS is approx. 70% whereas its value for GS is only 50% at this λ. We can also see the differences in the recall values at λ=0.4. For MAS search engine we get a recall close to 10% while in case of GS, the recall is nearly 0%. Another important difference between the two plots is the point when we get the maximum F-measure value. In case of the MAS (Fig. 2) the highest F-measure value of 68% is obtained at λ =0.8 and it saturated thereafter. Whereas, in the case of GS, the highest F-measure value is 70% which occurs at λ=0.9 and then saturates.

Although the highest F-measure value does not have a big difference, but we can observe the following interesting difference between the two search engines. As we observe the recall curve in both the plot (green curve), we see that the recall curve in the Fig. 3 (MAS) increase very rapidly as the pruning radius is increased from 0.4 to 0.7, whereas the increase in recall is not so sharp in Fig. 1 (GS). This observation can be attributed to the fact that GS include research articles from a wide variety of domains whereas the MAS include research articles from fewer domains. So the impact of λ variation is more significant in MAS search engines than in GS search engine. However, both the search engines show equivalent performance on the dataset used in this work.

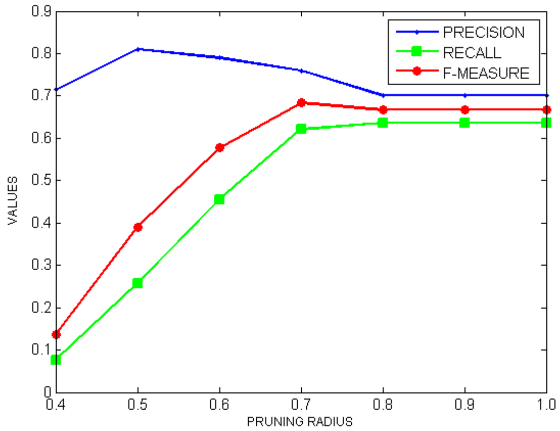


Fig. 3. Precision and recall curve for different values of pruning radius for MAS search engine.

Table I summarizes the performance of both the search engines on the test data. Based on this result, we see that GS is performing better than MAS in terms of the F-measure. Although the recall of both the search engines is same but the precision value for MAS is lower than that for GS. This implies that MAS retrieves greater number of extra terms than GS.

TABLE I: PRECISION, RECALL AND F-MEASURE VALUES COMPARISON FOR DIFFERENT SEARCH ENGINES

Measure	GS	MAS
Recall	88%	87%
Precision	83%	65%
F-measure	85%	75%

VII. CONCLUSIONS AND FUTURE WORK

In this work we have proposed a novel approach to automatically extract dataset names from scientific research articles. We have used context information from the “web” instead of using the local context information. The web knowledge used in the approach is basically derived from the academic search engines which have information about various research articles in organized manner. We have also compared the performance of our approach using two widely used research engines Google scholar and Microsoft academic search. The main contribution of this work is : (1) to automatically extract dataset names from research articles (2) to demonstrate the use of “web intelligence” to speed up information retrieval.

In the results, we show the performance evaluation using real world data. The results show that the proposed approach, though simple, gives F-measure as high as 85 %. Thus the approach is promising for real world application in dataset search engines. These search engines will enable data scientist to find the datasets useful for their research. As a part of future research we will build upon this system and develop a dataset search engine for academic researchers. We will expand this work to use in several other domains where data sets are required for research. This will require a sophisticated version of the proposed work.

Acknowledgment

This research work was supported by the NSF award IIS-1242869 "EAGER: Collaborative Research: Some Assembly Required: Understanding the Emergence of Teams and Ecosystems of Teams." We also thanks the lab members of DMR (database and Multimedia mining research group) group at University of Minnesota to provide valuable feedback and research articles as a dataset for this work

References

[1] D. P. Turney and P. Pantel, "From frequency to meaning: vector space models of semantics," *Journal of Artificial Intelligence Research*, pp. 141-188, 2010.

[2] M. Yutaka and M. Ishizuka, "Keyword extraction from a single document using word co-occurrence statistical information," *International Journal on Artificial Intelligence Tools* 13.01, pp. 157-169, 2004.

[3] K. Zhang *et al,*. "Keyword extraction using support vector machine," *Advances in Web-Age Information Management,* pp. 85-96, 2006.

[4] P. I. Chen and S. J. Lin, "Word AdHoc network: using Google core distance to extract the most relevant information," *Knowledge-Based Systems,* vol. 24, no. 3, pp. 393-405, 2011.

[5] Joachims and Thorsten, *A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization*, No. CMU-CS-96-118. Carnegie-Mellon Univ Pittsburgh Pa Dept of Computer Science, 1996.

[6] W. T. Yih and C. Meek, "Improving similarity measures for short segments of text," in *Proceedings of the National Conference on Artificial Intelligence*. vol. 22. no. 2. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2007.

[7] K. Jasmeen and V. Gupta, "Effective approaches for extraction of keywords," *IJCSI International Journal of Computer Science Issues* 7.6, 2010.

[8] H. Bao and D. Zhen, "An Extended Keyword Extraction Method," *Physics Procedia,* vol. 24, pp. 1120-1127, 2012.

[9] Fernández and Miriam *et al.*, "Semantically enhanced Information Retrieval: an ontology-based approach," *Web semantics: Science, services and agents on the worldwide web* 9.4, pp. 434-452, 2011.

[10] K. Soner *et al.,* "An ontology-based retrieval system using semantic indexing," *Information Systems,* vol. 37, no. 4, pp. 294-305, 2012.

[11] J. J. Feng *et al.*, "Keyword extraction based on sequential pattern mining," in *Proceedings of the Third International Conference on Internet Multimedia Computing and Service*. ACM, 2011.

[12] F. Liu, F. F. Liu, and Y. Liu, "A supervised framework for keyword extraction from meeting transcripts," *Audio, Speech, and Language Processing, IEEE Transactions on* vol.19, no. 3, pp. 538-548, 2011.

[13] S. H. Xu, S. H. Yang, and F. CM Lau, "Keyword extraction and headline generation using novel word features," in *Proc. of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010.

[14] I. Keisuke and N. Mc. Cracken, "Automated Keyword Extraction of Learning Materials Using Semantic Relations," 2010.

[15] C. W. Bruce, D. Metzler, and T. Strohman. *Search engines: Information retrieval in practice*. Addison-Wesley, 2010.

[16] V. P. Schaik and J. Ling, "An integrated model of interaction experience for information retrieval in a Web-based encyclopaedia," *Interacting with Computers,* vol. 23, no.1, PP.18-32, 2011.

[17] L. Yu, et al. "Concepts Similarity Algorithm Based on Google and KL Distance," *Jisuanji Gongcheng/ Computer Engineering,* vol. 37, no. 19, 2011.

[18] P. I. Chen, S. J. Lin, and Y. C. Chu, "Using Google latent semantic distance to extract the most relevant information," *Expert Systems with Applications,* vol. 38, no. 6, pp. 7349-7358, 2011.

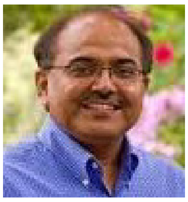
[19] P. I. Chen and S. J. Lin, "Automatic keyword prediction using Google similarity distance," *Expert Systems with Applications,* vol. 37, no.3, pp. 1928-1938, 2010.

[20] X. M. Qian , X. H. Hua, and X. S. Hou, "Tag filtering based on similar compatible principle," *Image Processing (ICIP), 2012 19th IEEE International Conference on*. IEEE, 2012.

[21] C. L. Rudi and P. MB Vitanyi, "The google similarity distance," *Knowledge and Data Engineering, IEEE Transactions on* vol. 19, no. 3, pp. 370-383, 2007.

Ayush Singhal was born in India in the year 1990. He is currently a second year PhD student in the computer science department at the University of Minnesota, USA. He completed his under graduation from the Indian Institute of Technology Roorkee, India in 2011. His major is computer science. His current research interests are data mining, information retrieval, web mining and social network analysis.

He has been working as a research assistant in the University of Minnesota for 2 years now. He has also worked in IBM Research labs, New Delhi India as a summer intern in year 2010.



Jaideep Srivastava received the Btech degree in computer science from The Indian Institute of Technology, Kanpur, India, in 1983, the MS and PhD degrees in computer science from the University of California, Berkley, in 1985 and 1988 respectively. He has been on the faculty of the Department of Computer Science and Engineering of the University of Minnesota, Minneapolis, since 1988 and is currently a professor.

He served as a research engineer with Uptron Digital Systems in Lucknow, India, in 1983. He as published more than 250 papers in refereed journals and conference proceedings in the areas of databases, parallel processing, artificial intellig3ence, multimedia and social network analysis; and he has delivered a number of invited presentations and participate in panel discussions on these topics. His professional activities have included service on various program committees and he has refereed papers for varied journals and proceedings, for events sponsored by the US National Science Foundation. He is a Fellow of the IEEE, and a Distinguished Fellow of Allina Hospitals' center for Healthcare Innovation. He has given over 150 invited talks in over 30 countries, including more than a dozen keynote addresses at major conferences.