International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 12 Issue: 02 | Feb 2025 www.irjet.net p-ISSN: 2395-0072

# Content-Based Generative Probabilistic Model Implementation for Finding Experts in Academic Domain

**Harshada V. Talnikar[1] , Snehalata B. Shirude[2]**

[1] *Asst. Prof. Institute of Industrial and Computer Management and Research, Nigdi, Pune. Maharashtra, India*
[2] *Asst. Prof. School of Computer Sciences, K. B. C. North Maharashtra University Jalgaon, Maharashtra, India*

---

**Abstract –** *As numerous and heterogeneous-structured data is available, it is the challenge to identify individuals with expert knowledge on a specific topic. This paper provides an analysis and review of different Information Retrieval (IR) models. One of the models is elaborated focusing on its experimental results and effectiveness. Literature study depicts that, when user searches documents to find an expert for any particular domain, in earlier IR models it often produces a ranked list of documents according to relevance. Modern IR algorithms are designed for achieving high mean average precision and better recall rates. These models have challenges in balancing precision, recall, and computational efficiency. We conducted experiments on the publication details of different experts with a single research domain. We analyzed expert finding models- BM25, TF-IDF, and Latent Dirichlet Analysis and it is observed that in domain specific expert retrieval, BM25 ranking function outperforms other ranking methods, if corresponding precision and recall metrics are considered.*

***Key Words***: **Expert finding, information retrieval; relevance, precision, recall, expert finding models, BM25, ranking functions.**

## 1. INTRODUCTION

In current scenario, there is growing interest for the expert finding task. It involves finding a list of person with recent knowledge or expertise in required specific domain. Research has focused on developing and implementing methods for identifying experts within specific academic disciplines. If anyone searches experts in a specific academic field then it has many applications, as research consultation, forming reviewers' panels [1], organizing experts' speech and reviewing papers as a part of a Peer Review Process etc. [2, 3].

The paper aims to explore the experts finding task with the help of model implementation in recent research field related to academic domain. Our approach of study to find the experts in academics relies heavily on acquiring a robust dataset of publications that can serve as a basis for assessing expertise. Google Scholar serves as a valuable initiative for retrieving the necessary data for expert determination purpose. Google Scholar considers a researcher's publications as representatives of their expertise [3]. Each paper record in Google Scholar typically includes only the title, which can be used for calculating the relevance of papers to specific queries. Therefore, we use it as a supplementary data source [4].

We propose three models for expert finding, considering Google Scholar as a data source:

### 1.1. Weighted Language Model:

This model determines an expert by finding the relativity with associated documents. It pre determines some prior probability for the related documents, which enhances the performance compared to the other basic expert finding models termed as language models.

The language model evaluates the relation in a query and relevant documents to number an expert's expertise. For instance, take a query q with equal relevance probability of 0.1 for different documents, d1, d2 of authors' a1 as well a2 respectively, with association probabilities of 0.2. It is assumed that, d1 has 100 citations, while d2 has only 10. In the basic language model, a1 and a2 are considered equally likely experts for topic q. However, considering citation counts reveals that d1 is more impactful than d2, making it more intuitive to consider a1 as the stronger candidate.

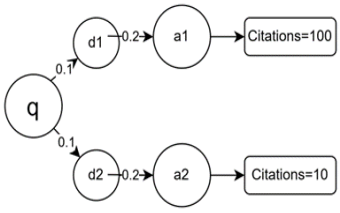


**Fig -1**: Basic language model

To overcome the limitations, a novel weighted model is used. It integrates the prior probability p (d), representing the likelihood that a document is published by an expert. Prior, probability depends on the document's importance and considers factors like citation counts.

Consider D represents set of related documents and Ca denotes expert candidate [5, 6]. If combined both the

document's relevance to a query and its significance, the author's expertise may be determined through scrutiny.

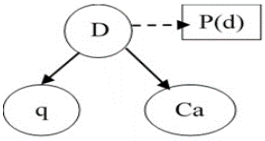


**Fig -2**: Weighted language model

### 1.2. Topic-Based Model:

In topic based models, expert candidate is denoted as weighted mixture of all the relevant topics. Such structure evaluates the significance of related documents. It enhances results compared to the language model. This approach associates each expert candidate with multiple topics and enables the ranking based on the weighted combination of the topics.

### 1.3 Hybrid Model:

Hybrid model integrates the advantages of weighted language model as well the topic-based model. Here the system is evaluated using the input dataset and judgments to assess how the experts supports those query topics. It is observed that, the hybrid model is effective as compared with other models.

Literature survey focuses on query-dependent and query-independent strategies for expert finding. In these methods, expert finding model searches expert related documents. Then using that information, it calculates how much the candidate is expert [7]. The approach which is query independent [8] involves constructing a "virtual document" or profile for each candidate [9], using all documents associated with them, and determining their ranking score relative to a user query based on this profile. Opposite to this, the query-dependent approach [10] begins by ranking documents in the dataset centred on the query and ascertains possible experts using the subgroup of retrieved input documents [11].

We studied strengths and weaknesses of each approach [7]. Query-independent profiles are often much smaller in size compared to the original retrieved data, making them more useful for data storage and management. However, this approach does not allow for evaluating the individual contribution of each document to the profile, which can limit its effectiveness. In contrast, the approach which depends on query facilitates use of enhanced techniques for text sculpting to rank specific documents, offering a more accurate assessment.

Different studied expert finding models comprise of following primary components:

- **Candidate:** A researcher or person eligible as an expert.
- **Document:** The diverse sources of expertise-related data.
- **Topic:** The specific domain or area of expertise [4].

The expert finding task typically involves five key aspects:

a. **Selection of Expertise Evidence (Data Sources)**: This step involves identifying and extracting documents relevant to expertise, which serves as the base for discovering experts. Relevant data about individuals must be gathered to determine their expertise for the specific domain.
b. **Expert Depiction**: The major goal of any expert finding system is searching inferred data that locates expertise and helps users in evaluating and selecting the most relevant experts. This requires identifying which information will be helpful for users who are unfamiliar with the experts in making decisions. Expertise is generally assessed through documented evidence, relationships, and contextual factors, such as human interactions.
c. **Model Development**: Model building considers pre-processing, ranking, and retrieval.

   i. **Pre-processing and ranking**: Similar to document retrieval systems, this involves creating indexes. However, expertise retrieval introduces additional challenges, such as identifying expert identifiers (e.g., names, emails) within documents and integrating data from diverse sources. Pre-processing techniques, such as removing stop words, are adapted from traditional information retrieval (IR) systems [1]. There are many challenges as handling ambiguities like different names referring to the same individual or identical names belonging to different individuals. Techniques like disambiguation and Named Entity Recognition are critical [12]. For example, some systems use matching methods which are rule-based to define experts through exact matches, initials with last names, or email addresses [13].

   ii. **Modelling and Retrieval**: In this phase it constructs models to tie candidate experts and user queries to rank them centred on the strength of relativity [1]. It includes common approaches as probabilistic models, network graph-based models, voting systems, and other hybrid techniques [4].

d. **Evaluation of Models**: An expert finding system model is evaluated on the basis of its accuracy to retrieve relevant experts to respond user queries. The test datasets are used to evaluate efficiency [14].
e. **Interaction Design**: To present expert search results to users is the challenging task. The list containing only expert names is not sufficient to evaluate a candidate's expertise [15]. It needs snippets which are able to review quickly. So it is essential that, result pages should contain additional details such as numbered lists of related documents, publications or projects, conferences, journals, affiliations, contact information, photos, and links etc. To enrich results, links for personal homepage are given using supervised learning methods [16-18].

This document is template. We ask that authors follow some simple guidelines. In essence, we ask you to make your paper look exactly like this document. The easiest way to do this is simply to download the template, and replace(copy-paste) the content with your own material. Number the reference items consecutively in square brackets (e.g. [1]). However the authors name can be used along with the reference number in the running text. The order of reference in the running text should match with the list of references at the end of the paper.

## 2. METHODOLOGY

### 2.1. Generative Probabilistic Model:

The generative probabilistic model evaluate existing associations between query and expert by manipulating the probability of specific topic generated by candidate e, given that he/she is expert on query q. It is represented as (P (e | q)).
Following two methods are used to estimate this probability:
i. **Candidate generation model:** This method directly generates candidates based on query domain (q) and the used probability model for identifying the expert e.
ii. **Topic generation model:** This method relies on Bayes' Theorem, and it is expressed as:

$$P(e/q) = (rank)\, P(q/e).\, P(e) \qquad \text{------- eq. (1)}$$

Here
P (e) is the probability of any candidate e being an expert.
P (q) is the possibility of the query q. As P (q) remains constant for an input query, it is ignored.
Thus, the probability of a candidate expert e for inputted query q is proportionate to that query. It is weighted by the prior certainty in the candidate's expertise [19].

Query-dependent and document-based methods search relevant documents to a given query. Then rank candidates on the consideration of a grouping of document significance and individual's correlation to those documents.

As the result of our studies, content-based generative probabilistic models [20] have been explored to rank researchers' profiles:

1. **Okapi BM25: (OKAPI)** Here a researchers' profiles are represented as a collection of terms. Two profiles are compared, where one is treated as a query and the other is treated as an individual document.
2. **Kullback-Leibler Divergence: (KLD)** In the model, a researcher's profile is considered as a probability distribution. Generally it is a multinomial distribution on term frequencies. KLD derives the similarity between these distributions and rank those profiles.
3. **Latent Dirichlet Allocation: (LDA)** Here researcher profile is represented as mixtures of topics, rather than a single distribution. This method is useful if researchers are with diverse interests.
4. **Trace-Based Similarity: (REL)** This model uses density based matrices. These matrices denote vector subspaces which capture complex relationships between terms.

Various research papers have applied these techniques to tasks like finding similar experts within a university [21], recommending beneficial collaborators [22], and forming research groups from heterogeneous documents.

Further for the implementation purpose we studied the document based model Okapi BM25 in depth. Okapi BM25 is a popular ranking function in Information Retrieval (IR) known for its accuracy, especially when dealing with short documents [23]. It's based on a probabilistic retrieval model.

The BM25 formula considers several factors:

- **Term Frequency:** Denoted as TF. It represents how many times a term appears in a document or query.
- **Inverse Document Frequency:** Denoted as IDF. It represents how rarely a term is in the entire document set.
- **Document Length:** It is length of the input document.
- **Query Term Weight:** It signifies the prominence of a term in the query.

$$BM25 = \sum_{t \in q} \log\left(\frac{N - n_t}{n_t}\right) \cdot \frac{(k_1 + 1) fd, t}{k + fd, t} \cdot \frac{(k_3 + 1) fd, q}{k_3 + fd, q}$$

-------eq. (2)

Here,
q is user query.
N is total number of documents retrieved.
$n_t$ is total no. of documents which contain term t.

$fd, t$ represents number of occurrence of the term t in the document d referred as term frequency.
$fd, q$ denotes number of occurrence of term (t) in query q.

$$k = k_1.\left((1-b) + \frac{b.dld}{avl}\right)$$ ------- eq. (3)

Dld is number of terms in d document
Avl is average document length
b, $k_1$, $k_3$ are parameter used for tuning which has values as b=0.75, *k1*=1.2 , *and* $k_3$=1000, that can be adjusted to fine-tune the ranking.

By considering these factors, BM25 effectively grades documents based on their importance to an inputted query q, making it valuable tool in search engines and other information retrieval applications.

**Algorithm**

The pseudo code for BM25 implementation:

**Begin**
**Step 1:** Tokenize query and documents into individual terms. //Tokenization

**Step 2:**
For each document in corpus: // Calculate Term //Frequency (TF)
For each query term:
Compute Term Frequency = Count of term t occurrence in the document d.

**Step 3:**
For each query term: // Find Inverse Document //Frequency (IDF)
Calculate IDF = *log* ((Total number of documents + 1) / (number of documents comprising the term + 1)).

**Step 4:**
// Adjust TF //based on document length
Compute average document length = total length of all retrieved documents / Total retrieved documents.
For every document in corpus:
for each query term:
Adjusted TF = TF / (1 - b + b * (document length / average document length))
// 'b' refers the tuning parameter.

**Step 5:**
For every document in corpus: // Compute BM25 // Score
Initialize BM25 Score = 0.
For each query term:
Compute weighted TF-IDF score = Adjusted TF * IDF.
Add weighted score to BM25 Score.
Return BM25 Scores for all documents.
**End**

BM25 is effective model as it has the ability to balance both term frequency and document length. It does parameter tuning. The parameters k1 and b are adjusted to fine-tune the algorithm for particular datasets. The model treats documents as a collection of words irrespective of word order.

## 3. RESULT AND DISCUSSIONS

The experiments are conducted using a dataset retrieved from Google Scholar. Google Scholar is chosen as the primary resource for obtaining publication data of experts. Large number of records are collected from Google Scholar, as outlined in Table 1. Given the diverse and heterogeneous origins of the data, it was exposed to missing values, noise, and inconsistencies. The superiority of the data highly impacts the outcome of data mining. Hence, raw data pre-processing was carried to improve its quality. This critical step involves cleaning and transforming the data in proper format for subsequent mining processes. Pre-processing helps to lessen the raw data size, establish correlations between features, normalize the data, eliminate outliers, and retain only relevant features by eliminating less important ones. Key techniques applied during this phase include data cleaning, integration, transformation, and reduction [24].

**Table -1:** Summary of retrieved data set

| Metadata | Value in numbers |
|---|---|
| Attributes | 26 |
| Instances | 2000 |
| Categorical Attributes | 12 |
| Continuous Attributes | 14 |
| Missing Values Rows | 81 |
| Missing Values Columns | 06 |
| **Resultant instances after Data pre-processing** | **1806** |
| **Resultant attributes after feature selection** | **03** |

Following the data cleaning process, approximately 10% of the records were removed. The clean data was then scaled and normalized using methods Standard Scaling, Maximum Absolute Scaling, and Median-Quantile Scaling. Subsequently, feature scaling and abstraction were carried out, resulting in three highly correlated features for the next stages of the expert finding task. These steps produced cleaned, scaled data which effectively contributed to accurate and precise results.

For finding experts using the pre-processed data, the OKAPI BM25 model has been implemented and analysed. To assess its performance, the number of records processed in each execution are varied. The assessment of the BM25 implementation results typical metrics recall and precision, which are commonly used to gauge the accuracy of ranking and document retrieval. Precision, a measure of correctness [25]. It evaluates how effectively the model excludes irrelevant documents, while recall, a unit of completeness, assesses the model capability to retrieve all relevant documents which meet the user needs. Pertinent documents in the corpus are ordered based on the relevance scores [26]. Performance of the model is evaluated using the methods at different precision values as P@5, P@10, P@100, and P@1000 as shown in Table 2.

**Table -2:** Precision@5, Precision@10, Precision@100, Precision@1000 results using BM25 algorithms on GoogleScholar retrieved Dataset.

| Number of records processed | Accuracy | Precision | Recall | F1 score |
|---|---|---|---|---|
| 5 | 0.8000 | 0.8000 | 1.0000 | 0.8888 |
| 10 | 0.7000 | 0.7142 | 0.8333 | 0.7692 |
| 100 | 0.7500 | 0.8095 | 0.8831 | 0.8447 |
| 1000 | 0.8125 | 0.8670 | 0.9247 | 0.8949 |

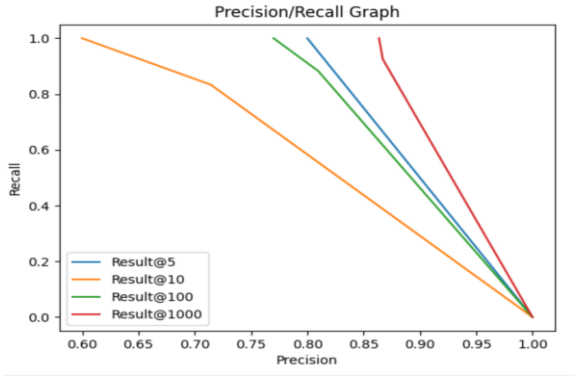


**Chart -1**: Precision / Recall Graph

## 4. CONCLUSION

In proposed paper, it is analysed and experimentally swotted the BM25 algorithms. The ranking and comparison measures are assessed in the context of altered number of input records. In our experiment using Google scholar dataset, we found that increasing the retrieval threshold impacts the model's performance: precision tends to drop as more results are retrieved, but recall improves. The choice of threshold depends on whether precision or recall is more critical for the specific task. High recall is preferred when missing relevant results is unacceptable. This paper highlights various approaches to find expert in any particular domain and implementation of OKAPI BM25 model. It offers in depth analysis of how the present day expert finding models function and highlight their strengths as well point out weaknesses. It helps to identify the gaps.

## 5. REFERENCES

[1] Talnikar HV, Shirude SB. An Architecture to Develop an Automated Expert Finding System for Academic Events. In Proceedings of the International Conference on Paradigms of Computing, Communication and Data Sciences: PCCDS 2022 2023 Feb 24 (pp. 297-306). Singapore: Springer Nature Singapore.

[2] D. M. Mimno and A. McCallum. Expertise modeling for matching papers with reviewers. In KDD, pages 500–509, 2007.

[3] Bharti PK, Ghosal T, Agarwal M, Ekbal A. PEERRec: An AI-based approach to automatically generate recommendations and predict decisions in peer review. International Journal on Digital Libraries. 2024 Mar;25(1):55-72.

[4] Google Scholar. URL:http://scholar.google.com/.

[5] K. Balog, L. Azzopardi, andM. de Rijke. Formal models for expert finding in enterprise corpora. In SIGIR, pages 43–50, 2006.

[6] I. Soboroff, A. de Vries, and N. Craswell. Overview of the trec-2006 enterprise track. In Proceedings of TREC 2006.

[7] D. Petkova and W. B. Croft. Hierarchical language models for expert finding in enterprise corpora. In ICTAI, pages 599–608, 2006.

[8] K. Balog andM. de Rijke. Determining expert profiles (with an application to expert finding). In IJCAI, pages 2657– 2662, 2007.

[9] C. Macdonald and I. Ounis. Voting for candidates: adapting data fusion techniques for an expert search task. In CIKM, pages 387–396, 2006.

[10] H. Fang and C. Zhai. Probabilistic models for expert finding. In ECIR, pages 418–430, 2007.

[11] Balog, K.; Fang, Y.; De Rijke, M.; Serdyukov, P.; Si, L. Expertise Retrieval. Found. Trends® Inf. Retr. 2012, 6, 127–256.

[12] Petkova, D.; Croft, W.B. Proximity-Based Document Representation for Named Entity Retrieval. In Proceedings of the 16th ACM Conference on Conference

on Information and Knowledge Management, Lisbon, Portugal, 6–10 November 2007; pp. 731–740.

[13] Balog, K.; Azzopardi, L.; De Rijke, M. Formal Models for Expert Finding in Enterprise Corpora. In Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle,WA, USA, 6–11 August 2006; pp. 43–50.

[14] Shirude, S.B., Kolhe, S.R., Classification of Library Resources in Recommender System Using Machine Learning Techniques. In: Mandal, J., Sinha, D. (eds) Social Transformation – Digital Way. CSI 2018. Communications in Computer and Information Science, vol 836. Springer, Singapore.

[15] Lin, S.; Hong, W.; Wang, D.; Li, T. A Survey on Expert Finding Techniques. J. Intell. Inf. Syst. 2017, 49, 1–25.

[16] Fang, Y.; Si, L.; Mathur, A.P. Discriminative Graphical Models for Faculty Homepage Discovery. Inf. Retr. 2010, 13, 618–635.

[17] Das, S.; Mitra, P.; Giles, C. Learning To Rank Homepages for Researcher name Queries. In the International Workshop on Entity-Oriented Search; ACM, EOS, SIGIR 2011 Workshop: Beijing, China, 2011; pp. 53–58.

[18] Tang, J.; Zhang, J.; Yao, L.; Li, J.; Zhang, L.; Su, Z. Arnetminer: Extraction and Mining of Academic Social Networks. In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, NV, USA, 24–27 August 2008; pp. 990–998.

[19] Fang, H.; Zhai, C. Probabilistic Models for Expert Finding. In Proceedings of European Conference on Information Retrieval; Springer: Berlin/Heidelberg, Germany, 2007; pp. 418–430.

[20] Gollapalli, S.D.; Mitra, P.; Giles, C.L. Similar Researcher Search in Academic Environments. In Proceedings of the 12th ACM/IEEE-CS Joint Conference on Digital Libraries, Washington, DC, USA, 10–14 June 2012; pp. 167–170.

[21] Hofmann, K.; Balog, K.; Bogers, T.; De Rijke, M. Contextual Factors for Finding Similar Experts. J. Assoc. Inf. Sci. Technol. 2010, 61, 994–1014.

[22] Kong, X.; Jiang, H.; Wang, W.; Bekele, T.M.; Xu, Z.; Wang, M. Exploring Dynamic Research interest and Academic Influence for Scientific Collaborator Recommendation. Scientometrics 2017, 113, 369–385.

[23] Li, C.-T.; Shan, M.-K.; Lin, S.-D. Context-Based People Search in Labeled Social Networks. In Proceedings of the 20th ACM International Conference on Information and Knowledge Management, Glasgow, Scotland, UK, 24–28 October 2011; pp. 1607–1612.

[24] Tamilselvi, R. B. Sivasakthi and R. Kavitha., An efficient preprocessing and postprocessing techniques in data minig. Intl. J. Res. Comput. Appl. Rob. . 3: 80-85, 2015.

[25] de Campos LM, Fernández-Luna JM, Huete JF, Ribadas-Pena FJ, Bolaños N. Information retrieval and machine learning methods for academic expert finding. Algorithms. 2024 Jan 23;17(2):51.

[26] Shirude SB, Kolhe SR. Measuring similarity between user profile and library book. In2014 international conference on information systems and computer networks (ISCON) 2014 Mar 1 (pp. 50-54). IEEE.