Expert Finding in Legal Community Question Answering

Arian Askari[1], Suzan Verberne[1], and Gabriella Pasi[2]

[1] Leiden Institute of Advanced Computer Science, Leiden University
first_initial.lastname@liacs.leidenuniv.nl
[2] Department of Informatics, Systems and Communication, University of Milano-Bicocca gabriella.pasi@unimib.it

Abstract. Expert finding has been well-studied in community question answering (QA) systems in various domains. However, none of these studies addresses expert finding in the legal domain, where the goal is for citizens to find lawyers based on their expertise. In the legal domain, there is a large knowledge gap between the experts and the searchers, and the content on the legal QA websites consist of a combination formal and informal communication. In this paper, we propose methods for generating query-dependent textual profiles for lawyers covering several aspects including sentiment, comments, and recency. We combine query-dependent profiles with existing expert finding methods. Our experiments are conducted on a novel dataset gathered from an online legal QA service. We discovered that taking into account different lawyer profile aspects improves the best baseline model. We make our dataset publicly available for future work.

Keywords: Legal Expert finding · Legal IR · Data collection

1 Introduction

Expert finding is an established problem in information retrieval [4] that has been studied in a variety of fields, including programming [8,30], social networks [13,17], bibliographic networks [16,25], and organizations [26]. Community question answering (CQA) platforms are common sources for expert finding; a key example is Stackoverflow for expert finding in the programming domain [21].

Until now, no studies have addressed expert finding in the legal domain. On legal CQA platforms, citizens search for lawyers with specific expertise to assist them legally. A lawyer's impact is the greatest when they work in their expert field [22]. In terms of expertise and authority, there is a large gap between the asker and the answerer in the legal domain, compared to other areas. For instance, an asker in programming CQA is someone who is a programmer at least on the junior level, and the answerer could be any unknown user. In legal CQA, the asker knows almost nothing about law, and the answerer is a lawyer who is a professional user. The content in legal CQA is a combination of formal and informal language and it may contain emotional language (e.g., in a topic about

child custody). As a result, a lawyer must have sufficient emotional intelligence to explain the law clearly while also being supportive [19].

A lawyer's expertise(s) is crucial for a citizen to be able to trust the lawyer to defend them in court [23]. Although there are some platforms in place for legal expert finding (i.e, Avvo, Nolo, and E-Justice), there is currently no scientific work addressing the problem.

In this paper, we define and evaluate legal expert finding methods on legal CQA data. We deliver a data set that consists of legal questions written by anonymous users, and answers written by professional lawyers. Questions are categorized in different categories (i.e, bankruptcy, child custody, etc.), and each question is tagged by one or more expertises that are relevant to the question content. Following prior work on expert finding in other domains [9,7,8], we select question tags as queries. We represent the lawyers by their answers' content. For a given query (required expertise), the retrieval task is to return a ranked list of lawyers that are likely to be experts on the query topic. As ground truth, we use lawyers' answers that are marked as best answers as a sign of expertise.

Our contributions are three-fold: (1) We define the task of *lawyer finding* and release a test collection for the task;[3] (2) We evaluate the applicability of existing expert finding methods for lawyer finding, both probabilistic and BERT-based; (3) We create query-dependent profiles of lawyers representing different aspects and show that taking into account query-dependent expert profiles have a great impact on BERT-based retrieval on this task.

2 Related work

The objective of expert finding is to find users who are skilled on a specific topic. The two most common ways to expert finding in CQA systems are topic-based and network-based. Because there is not a network structure between lawyers in legal CQA platforms, we focus on topic-based methods. The main idea behind topic-based models [3,10,11,20,14,27,29] is to rank candidate experts according to the probability $p(ca|q)$, which denotes the likelihood of a candidate ca being an expert on a given topic q. According to Balog et al. [3], expert finding can be approached by generative probabilistic modelling based on candidate models and document models. Recently, Nikzad et al. [18] introduces a multimodal method on academic expert finding that takes into account text similarity using transformers, the author network, and h-index of the author. We approach lawyer finding differently since a lawyer does not have an h-index, there is not a sufficiently dense network of lawyers in the comment sections of legal CQA platforms, and the content style in academia is different than in legal.

3 Data collection and preparation

Data source and sample. Our dataset has been scraped from the Avvo QA forum, which contains $5,628,689$ questions in total. In order to preserve the privacy of

[3] The data and code is available on https://github.com/EF_in_Legal_CQA

users, we stored pages anonymously without personal information and replaced lawyer names by a number. Avvo is a legal online platform where anyone could post their legal problem for free and receive responses from lawyers. It is also possible to read the answers to prior questions. Lawyers' profiles on Avvo have been identified with their real name, as opposed to regular users. The questions are organised in categories and each category (i.e. 'bankruptcy') includes questions with different category tags (i.e. 'bankruptcy homestead exemption'). For creating our test collection, we have selected questions and their associated answers categorised as 'bankruptcy' for California, which is the most populated state of the USA. We cover the period July 2016 until July 2021 which covers $9,897$ total posts and $3,741$ lawyers. The average input length of a candidate answer is 102 words.

Relevance labels and query selection. We mark attorneys as experts on a *category tag* when two conditions are met. The first is engagement filtering: Similar to the definition proposed in [9], a lawyer should have ten or more of their answers marked as accepted by the asker on a *category*, and a more than average number of best answers among lawyers on that *category tag*. A best answer is either labelled as the most useful by the question poster or if more than three lawyers agree that the answer is useful. Second, following the idea proposed in [28], the acceptance ratio (count of best answers/count of answers) of their answers should be higher than the average acceptance ratio (i.e. 4.68%) in the test collection on a category. Based on the two conditions, we select 61 lawyers as experts, who combined have given $5,614$ answers and $1,917$ best answers. From the top 20 percent tags which co-occur with 'bankruptcy', we select tags (84) as queries that at least have two experts. There are on average 5 experts (lawyers who met expert conditions on a *category tag*) per query in the test collection. Our data size is comparable with four TREC Expert Finding test collections between 2005-2008, that have 49–77 queries and $1,092$–$3,000$ candidates [6,24,2,5].

Evaluation setup. We split our data into train, validation, and test sets based on the relevant expert lawyers – instead of queries – to avoid our models being overfitted on previously seen experts. By splitting on experts, the retrieval models are expected to be more generalized and be able to detect new experts in the system. The distribution of relevant experts and queries in each set is shown in table 1. For each train/valid/test set, in retrieval, we have all non-relevant lawyers (3680 in total) plus relevant lawyers (experts) (20/20/21) to be ranked.

Table 1. Statistics on the counts of queries, answers, and relevant experts in our data.

	train	validation	test	train ∩ validation	train ∩ test
number of relevant experts	20	20	21	0	0
number of queries	76	69	71	61	65
number of answers	39,588	34,128	35,057	7,290	7,918

4 Methods

Lawyer finding is defined as finding the right legal professional lawyer(s) with the appropriate skills and knowledge within a state/city. Cities are provided by Avvo as metadata; we only keep the city of the asker in our ranking and filter out lawyers' answers from other cities. For relevance ranking, lawyers are represented by their answers, like in prior work on expert finding in other domains [3].

4.1 Baseline 1: Probabilistic language modelling

Following [9,7,8], we replicate two types of probabilistic language models to rank lawyers: document-level (model 1), and candidate-level (model 2) that were originally proposed by Balog et al. [3] In these models, the set of answers written by a lawyer is considered the proof of expertise.

In the **Candidate-based model,** we create a textual representation of a lawyer's knowledge based on the answers written by them. Following Balog et al. [3], we estimate $p(ca|q)$ by computing $p(q|ca)$ based on Bayes' Theorem. We call this model hereinafter *model 1*. In *model 1*, $P(q|ca)$ is estimated by:

$$p(q|ca) = \prod_{t \in q} \left\{ (1 - \lambda_{ca}) \times \left(\sum_{d \in D_{ca}} p(t|d) \times p(d|ca) \right) + \lambda_{ca} \times p(t) \right\} \quad (1)$$

Here, D_{ca} consists of documents (answers) that have been written by lawyer ca; $p(t|d)$ is the probability of the term t in document d; $p(t)$ is the probability of a term in the collection of documents; and $p(d|ca)$ is the probability of document d is written by candidate ca. In the legal CQA platform answers are written by one lawyer. Therefore, $p(d|ca)$ is constant.

In the **Document-based model**, the document-centric model builds a bridge between a query and lawyers by considering documents in the collection as link. Given a query q, and collection of answers ranked according to q, lawyers are ranked by aggregating the sum over the relevance scores of their retrieved answers:

$$p(q|ca) = \sum_{d \in D_{ca}} \left(\prod_{t \in q} \left\{ (1 - \lambda_d) \times p(t|d) + \lambda_d \times p(t) \right\} \times p(d|ca) \right) \quad (2)$$

λ_d and λ_{ca} are smoothing parameters that are dynamically computed per query and candidate lawyer document (lawyer's answer)/representation following [3]. Besides of the original *model 1*, and *model 2* based on probabilistic language modelling, we experiment with BM25 to rank expert candidates' profiles and documents and refer to those by *model 1 BM25*, and *model 2 BM25*.

4.2 Baseline 2: Vanilla BERT

By Vanilla BERT, we mean a pre-trained BERT model (BERT-Base, Uncased) with a linear combination layer stacked atop the classifier [CLS] token that is

Table 2. Baselines and proposed model results on the test set. Significant improvements over the probabilistic baselines (Model 1 LM/Bm25, Model 2 LM/BM25), and over the Vanilla BERT Document-based models are marked with $*$, and $\bullet$respectively.

Model	MAP	MRR	P@1	P@2	P@5
Model 1 (Candidate-based) LM	22.8%	40.9%	23.9%	19.7%	13.2%
Model 1 (Candidate-based) BM25	3.7%	7.0%	2.9%	1.4%	2.6%
Model 2 (Document-based) LM	19.4%	21.9%	13.5%	12.7%	7.8%
Model 2 (Document-based) BM25	21.0%	36.6%	22.5%	18.3%	11.5%
Vanilla BERT Document-based (VBD)	37.3%*	70.7%$*$	60.5%$*$	55.6%$*$	25.9%$*$
VBD + Profiles (weighted)	**39.3**%$*\bullet$	**73.2**%	**64.9**%$*\bullet$	**57.1**%	**27.7**%$*\bullet$

fine-tuned on our dataset in a pairwise cross-entropy loss setting using the Adam optimizer. We used the implementation of MacAvaney et al. [15] (CEDR).

After initial ranking with *model 2*, we fine-tune Vanilla BERT to estimate the relevance between query and answer terms. We select retrieved answers of the top-k(50) lawyers to re-calculate their relevance score by Vanilla BERT according to the query q. Finally, we re-rank the top-k by these relevance scores. Given a query and an answer, we train Vanilla BERT to estimate the relevance that the answer was written by an expert: "`[CLS]query[SEP]candidate answer[SEP]`".

4.3 Proposed Method

Given a query q and a collection of answers D that are written by different lawyers, we retrieve a ranked list of answers (D_q) using *model 1*. We create four query-dependent profiles for the lawyers L_q who have at least one answer in D_q. Each profile consists of text, and that text is sampled to represent different aspects of a lawyer's answers. The aspects are comments, sentiment-positive, sentiment-negative, and recency.

On the CQA platform it is possible to post comments in response to lawyer's answer. Therefore, there is a collection of comments C_{D_q} with regard to the query. We consider the comments as possible signals for the asker's satisfaction (i.e., a "thank you" comment would indicate that the asker received a good answer). Thus, for **comment-based profiles (CP)**, we shuffle the comments to l_i's answers and concatenate the first sentence of each comment. For **sentiment-positive (PP) and negative (NP) profiles**, we shuffle positive (negative) sentences from l_i's answers and concatenate them. Since our data in legal CQA is similar in genre to social media text, we identify answer sentiment using Vader [12], a rule-based sentiment model for social media text. For the **recency-based profile (RP)**, we concatenate the most recent answers of l_i. For each profile we sample the text until it exceeds 512 tokens.

We fine-tune Vanilla BERT on each profile. We represent the query as sentence A and the lawyer profile as sentence B in the BERT input: "`[CLS] query [SEP] lawyer profile [SEP]`" Finally, we aggregate the scores of the four profile-trained BERT models and *BERT Document-based* using a linear combination of the five models' scores inspired by [1]: $aggr_S(d,q) = w_1 S_{BD} + w_2 S_{CP} + w_3 S_{PP} + w_4 S_{NP} + w_5 RP$, where $aggr_S$ is the final aggregated score; the weights

w_i are optimized using grid search in the range [1..100] on the validation set. BD refers to the *BERT Document-based* score, and CP, PP, NP, RP to the four profile-trained BERT models.

5 Experiments and Results

Experimental setup We replicate [3] using Elasticsearch for term statistics, indexing, and BM25 ranking. Following the prior work on expert finding, we report MAP, MRR, and Precision@k ($k = 1, 2, 5$) as evaluation metrics.

Retrieval results The ranking results for models are shown in table 2. The best candidate-based and document-based lexical models are the original *model 1 LM* [3], and *model 2 BM25* respectively. We used *model 2 BM25* as our initial ranker for Vanilla BERT. *Vanilla BERT Document-based* outperforms all lexical models by a large margin. The best ranker in terms of all evaluation metrics is the weighted combination of BERT and the lawyer profiles. This indicates that considering different aspects of a lawyer's profile (comments, sentiment, recency) is useful for legal expert ranking. We employed a one-tailed t-test ($\alpha = 0.05$) to measure statistical significance.

Analysis of models' weights. We found $20, 13, 2, 4, 1$ as optimal weights for BERT, Comment, Recency, Sentiment positive and negative based models respectively. As expected, the BERT score plays the largest role in the aggregation as it considers all retrieved answers of a lawyer. The second weight is for the Comment profile which confirms our assumption that the content of askers' comments are possible signals for the relevance of the lawyer's answer. The Sentiment profile's weight shows positive sentiment is more informative than negative on this task.

Analysis of differences on seen and unseen queries. In Section 3, we argued that in our task, being robust to new lawyers is more important than being robust to new expertises (queries). We therefore split our data on the expert level and as a result there are overlapping queries between train and test set. We analyzed the differences in model effectiveness between seen and unseen queries. We found small differences: $p@5$ is 27% on seen queries, and 25% on unseen queries. This indicates the model generalizes quite well to unseen queries.

6 Conclusions

In this paper, we defined the task of legal expert finding. We experimented with baseline probabilistic, BERT-based, and proposed expert profiling methods on our novel data. BERT-based method outperformed probabilistic methods, and the proposed methods outperformed all models.

For future work, there is a need to study more in-depth the robustness of proposed methods on different legal categories. Moreover, by providing this dataset we facilitate other tasks such as legal question answering, duplicate question detection, and finding lawyers who will reply to a question.

ACKNOWLEDGMENTS

This work was supported by the EU Horizon 2020 ITN/ETN on Domain Specific Systems for Information Extraction and Retrieval (H2020-EU.1.3.1., ID: 860721).

References

1. Althammer, S., Askari, A., Verberne, S., Hanbury, A.: DoSSIER@ COLIEE 2021: Leveraging dense retrieval and summarization-based re-ranking for case law retrieval. arXiv preprint arXiv:2108.03937 (2021)
2. Bailey, P., De Vries, A.P., Craswell, N., Soboroff, I.: Overview of the trec 2007 enterprise track. In: TREC. Citeseer (2007)
3. Balog, K., Azzopardi, L., de Rijke, M.: A language modeling framework for expert finding. Information Processing & Management **45**(1), 1–19 (2009)
4. Balog, K., Fang, Y., De Rijke, M., Serdyukov, P., Si, L.: Expertise retrieval. Foundations and Trends in Information Retrieval **6**(2–3), 127–256 (2012)
5. Balog, K., Thomas, P., Craswell, N., Soboroff, I., Bailey, P., De Vries, A.P.: Overview of the trec 2008 enterprise track. Tech. rep., Amsterdam Univ (Netherlands) (2008)
6. Craswell, N., De Vries, A.P., Soboroff, I.: Overview of the trec 2005 enterprise track. In: Trec. vol. 5, pp. 1–7 (2005)
7. Dargahi Nobari, A., Sotudeh Gharebagh, S., Neshati, M.: Skill translation models in expert finding. In: Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval. pp. 1057–1060 (2017)
8. Dehghan, M., Abin, A.A., Neshati, M.: An improvement in the quality of expert finding in community question answering networks. Decision Support Systems **139**, 113425 (2020)
9. van Dijk, D., Tsagkias, M., de Rijke, M.: Early detection of topical expertise in community question answering. In: Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval. pp. 995–998 (2015)
10. Fu, J., Li, Y., Zhang, Q., Wu, Q., Ma, R., Huang, X., Jiang, Y.G.: Recurrent memory reasoning network for expert finding in community question answering. In: Proceedings of the 13th International Conference on Web Search and Data Mining. pp. 187–195 (2020)
11. Guo, J., Xu, S., Bao, S., Yu, Y.: Tapping on the potential of q&a community by recommending answer providers. In: Proceedings of the 17th ACM conference on Information and knowledge management. pp. 921–930 (2008)
12. Hutto, C., Gilbert, E.: Vader: A parsimonious rule-based model for sentiment analysis of social media text. In: Proceedings of the International AAAI Conference on Web and Social Media. vol. 8 (2014)
13. Li, G., Dong, M., Yang, F., Zeng, J., Yuan, J., Jin, C., Hung, N.Q.V., Cong, P.T., Zheng, B.: Misinformation-oriented expert finding in social networks. World Wide Web **23**(2), 693–714 (2020)
14. Liu, X., Ye, S., Li, X., Luo, Y., Rao, Y.: Zhihurank: A topic-sensitive expert finding algorithm in community question answering websites. In: International Conference on Web-Based Learning. pp. 165–173. Springer (2015)

15. MacAvaney, S., Yates, A., Cohan, A., Goharian, N.: Cedr: Contextualized embeddings for document ranking. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 1101–1104 (2019)
16. Neshati, M., Hashemi, S.H., Beigy, H.: Expertise finding in bibliographic network: Topic dominance learning approach. IEEE transactions on cybernetics **44**(12), 2646–2657 (2014)
17. Neshati, M., Hiemstra, D., Asgari, E., Beigy, H.: Integration of scientific and social networks. World wide web **17**(5), 1051–1079 (2014)
18. Nikzad-Khasmakhi, N., Balafar, M., Feizi-Derakhshi, M.R., Motamed, C.: Berters: Multimodal representation learning for expert recommendation system with transformers and graph embeddings. Chaos, Solitons & Fractals **151**, 111260 (2021)
19. Pekaar, K.A., van der Linden, D., Bakker, A.B., Born, M.P.: Emotional intelligence and job performance: The role of enactment and focus on others' emotions. Human Performance **30**(2-3), 135–153 (2017)
20. Riahi, F., Zolaktaf, Z., Shafiei, M., Milios, E.: Finding expert users in community question answering. In: Proceedings of the 21st international conference on world wide web. pp. 791–798 (2012)
21. Rostami, P., Neshati, M.: Intern retrieval from community question answering websites: A new variation of expert finding problem. Expert Systems with Applications **181**, 115044 (2021)
22. Sandefur, R.L.: Elements of professional expertise: Understanding relational and substantive expertise through lawyers' impact. American Sociological Review **80**(5), 909–933 (2015)
23. Shanahan, C.F., Carpenter, A.E., Mark, A.: Lawyers, power, and strategic expertise. Denv. L. Rev. **93**, 469 (2015)
24. Soboroff, I., de Vries, A.P., Craswell, N., et al.: Overview of the trec 2006 enterprise track. In: Trec. vol. 6, pp. 1–20 (2006)
25. Torkzadeh Mahani, N., Dehghani, M., Mirian, M.S., Shakery, A., Taheri, K.: Expert finding by the dempster-shafer theory for evidence combination. Expert Systems **35**(1), e12231 (2018)
26. Wang, Q., Ma, J., Liao, X., Du, W.: A context-aware researcher recommendation system for university-industry collaboration on r&d projects. Decision Support Systems **103**, 46–57 (2017)
27. Xu, F., Ji, Z., Wang, B.: Dual role model for question recommendation in community question answering. In: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval. pp. 771–780 (2012)
28. Yang, J., Tao, K., Bozzon, A., Houben, G.J.: Sparrows and owls: Characterisation of expert behaviour in stackoverflow. In: International conference on user modeling, adaptation, and personalization. pp. 266–277. Springer (2014)
29. Yang, L., Qiu, M., Gottipati, S., Zhu, F., Jiang, J., Sun, H., Chen, Z.: Cqarank: jointly model topics and expertise in community question answering. In: Proceedings of the 22nd ACM international conference on Information & Knowledge Management. pp. 99–108 (2013)
30. Zhou, G., Zhao, J., He, T., Wu, W.: An empirical study of topic-sensitive probabilistic model for expert finding in question answer communities. Knowledge-Based Systems **66**, 136–145 (2014)