

# Social Search: retrieving information in Online Social Platforms - A Survey

Maddalena Amendola<sup>a,b</sup>, Andrea Passarella<sup>b</sup>, Raffaele Perego<sup>c</sup>

<sup>a</sup>*Department of Computer Science, University of Pisa, Pisa, 56127, Italy*

<sup>b</sup>*Institute for Informatics and Telematics (IIT), National Research Council (CNR), Pisa, 56124, Italy*

<sup>c</sup>*Institute of Information Science and Technologies (ISTI), National Research Council (CNR), Pisa, 56124, Italy*

---

## Abstract

*Social Search* research studies methodologies exploiting social information to better satisfy user information needs in Online Social Media while simplifying the search effort and consequently reducing the time spent and the computational resources utilized. Starting from previous studies, in this work, we analyze the current state of the art of the Social Search area, proposing a new taxonomy and highlighting current limitations and open research directions. We divide the Social Search area into three subcategories, where the social aspect plays a pivotal role: *Social Question&Answering*, *Social Content Search*, and *Social Collaborative Search*. For each subcategory, we present the key concepts and selected representative approaches in the literature in greater detail. We found that, up to now, a large body of studies model users' preferences and their relations by simply combining social features made available by social platforms. It paves the way for significant research to exploit more structured information about users' social profiles and behaviours (as they can be inferred from data available on social platforms) to optimize their information needs further.

*Keywords:* Social Search, Search in Online Social Networks, Social Information Retrieval

---

## 1. Introduction

Online Social Network (OSN) platforms have been one of the major evolutions of the Web for at least a decade. Aside from their social implications, OSNs are impacting many established areas of computer science, which are also progressing based on the pervasive diffusion of OSN services among users. A chief example is the area of *Social Search*. Social Search lies at the intersection between Information Retrieval (IR) and Social Networks. It is generally defined as a cooperative process that relies on implicit or explicit information about social relationships between users to satisfy an information need. The main focus of this paper is to provide a comprehensive survey of the main results in this very active field. IR processes are today at the basis of almost all the most common uses of the Web, whether the user searches for a restaurant, the solution to a problem, wants to learn something, or asks for opinions. With the progressive development of tools for creating and sharing personal content and the increase of explicit or hidden interactions between users, the process that leverages users' social information in the standard IR tasks is becoming of fundamental importance. Furthermore, the interplay between IR and OSNs is becoming increasingly crucial as OSNs are one of the leading platforms where users search for information. Therefore, social content in OSNs can complement traditional Web sources in providing fresh and almost real-time information about events and ongoing discussions [1]. The reason why people prefer, in many cases, to retrieve information from a social network rather than a conventional search engine lies in the origin of the information need. Through social networks, users can obtain tailored information and different opinions by interacting with the virtual counterpart of real individuals [2].

In this survey, we follow the definition and taxonomy proposed in [3] for Social Search (we refer the reader to Section 2 for related surveys and definitions). Thus, we consider Social Search as the IR process applied to social network platforms and content. In presenting the recent advances in Social Search, as in [3], we divide the broad topic into three categories based on the type of Social Search system. Specifically, the following three categories can be seen as a particular instance of a Social Search system focusing on a specific approach. Figure 1 shows the taxonomy we propose. The leaves in the taxonomy have different colours based on their role in the specific task: green if they are components needed to perform the task, and yellow if they are alternative approaches to accomplish the task. The three

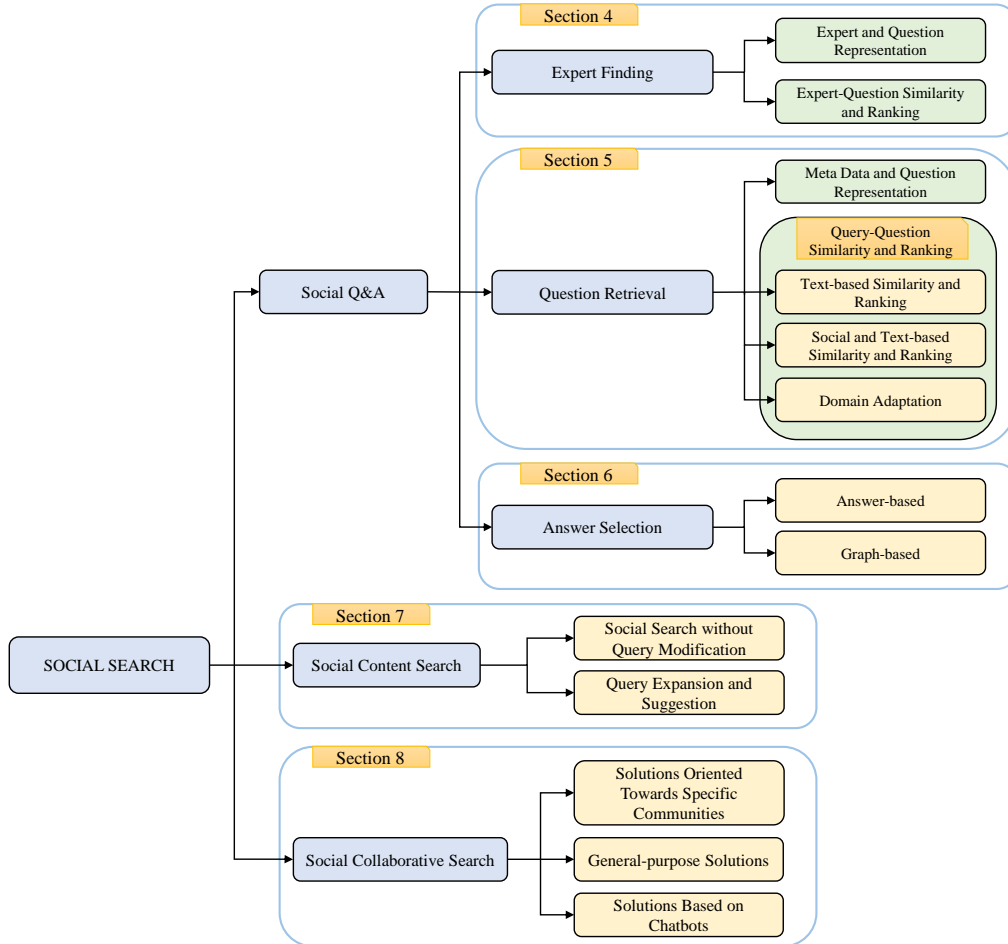


Figure 1: The proposed Social Search taxonomy.

subcategories are the following:

- **Social Q&A:** through Question&Answering (Q&A) communities, people can ask questions using more “natural” and flexible forms of natural language (be it via speech interfaces or typing text concerning more rigid keyword-based queries), expressing their information need more completely. Alternatively, the user can refer to existing questions and avoid waiting for someone to answer. These platforms differ from social networks in the type of relationships between users, which is

usually implicit here. By analyzing them, we have divided this category into three subcategories: *Expert Finding*, *Question Retrieval*, and *Answer Selection*. The Expert Finding problem aims to identify experts in the community to whom forward a newly posted question to satisfy the user information need in the shortest possible time. The Question Retrieval problem analyses the question archives to find old questions similar to the new ones published. Finally, Answer Selection deals with the problem of automatically selecting the best answer in a long thread of questions' answers and comments, which makes users unable to choose the correct one. Expert Finding, Question Retrieval, and Answer Selection are three very widely studied problems, which are therefore analyzed separately in Sections 4, 5, and 6, respectively. These three topics should be seen as complementary components of a fully-fledged Social Q&A system rather than as alternative ways to implement such a system;

- **Social Content Search:** refers to techniques that facilitate the users' searches in social networks, where the relationships between users are usually explicit and pre-established, unlike community Question&Answering (cQA) sites where they are usually implicit. In a Q&A platform, a user may also have no relationship with other users and still exploit the system by asking questions and taking advantage of the knowledge provided by "strangers." Instead, users usually use social networks to post/search for statuses, get information on events, stay up to date, and maintain connections. We divide the Social Content Search literature into two areas representing two alternative approaches to accomplish the task: the works that do not modify the query and those that propose terms for query expansion or suggest queries. The literature on Social Content Search is presented in Section 7;
- **Social Collaborative Search:** in Social Q&A and Social Content Search, the collaboration among users in the IR process is mostly implicit. However, in other cases, the search process involves multiple people who *explicitly* collaborate to achieve a common purpose. Collaborative Search allows for unifying knowledge, skills, and experience. We divide this area into three parts: (i) specific community-oriented solutions (i.e., academic purposes), (ii) general-purpose solutions, and (iii) chatbot-based solutions. The first two approaches are alterna-

tives that define the target of the Collaborative system. The last is a complementary organization of the Collaborative system that favours building dynamic communities focused on a common goal. The Social Collaborative Search literature is presented in Section 8.

Note that, to keep the paper self-contained despite the vastity of the literature, we use the following approach unless otherwise stated. We identify critical issues addressed in the literature for each leaf node in the taxonomy. For each such issue, we describe more extensively only one paper to exemplify the approach, and we briefly mention other papers highlighting only their key conceptual differences concerning the former. Finally, in section 9, we highlight critical open challenges, emphasizing that, in most cases, they could be addressed appropriately through a more structured approach to model social aspects of the users' behaviour as they emerge by analyzing data available on the considered Online Social platform. These challenges arise mainly from the nature of social data, its volume, dynamics, and the need for the user to be more aware of the personal data used.

This work aims to enrich the Social Search research area by offering a new schema of the types of social systems existing today and the different methodologies used to make these systems more tailored to users. After identifying the macro-areas of Social Search, we selected and analyzed in detail the methods proposed over the last few years, highlighting the most efficient ones. Moreover, this analysis underlines open challenges that could be addressed by exploiting user models and social information in a different way. The goal is to introduce the reader to the Social Search research area by providing an easy-to-follow overview of its up-to-date state-of-the-art.

## **2. Definitions and Related Surveys**

The initial concepts regarding Social Search relate to the modern idea of Social Web Search, which aims to improve the IR processes of search engines leveraging social information. In the absence of popular social networks, the first approach was to consider the history of searches as a source of additional information for future searches via automatic query terms expansion [4]. According to [5], Social Search should contrast the simple surfing, where users follow hyperlinks until they find the information. Considering the Web as modeled by a graph that connects information in a highly complex way, its huge size presents users with two difficulties: deciding which site to choose

and judging a priori the informative value of the returned results. Following a Social Search approach, users should first use information from other users' experiences to start with a site and further exploit it to refine the information obtained through regular surfing. Subsequently, the definition of Social Search was transformed by the introduction and popularity of the first social networks. In [6], Social Search has been defined as "*an approach to Web search that tempts to offer communities of like-minded individuals more targeted search services, based on the search behavior of their peers, bringing together ideas from Web search, social networking and personalization.*" Social Search becomes a combination between social networking and Web search, exploiting the information about the communities that social networks allow to create. In this way, search engines have become more sensitive to the needs and preferences of specific communities of users. Personalizing search results based on like-minded individuals has also been known as *Social Navigation*. According to [7], Social Navigation aims at taking advantage of the human tendency to follow other people with similar interests and repeat their steps. The authors in [8] combine Social Navigation and Social Search techniques to take full advantage of the collaboration both can offer. With the former, it is possible to leverage the past browsing behavior of users to guide other users to relevant information; the latter aims to use the search user and users' community patterns to adapt the result list to the needs and preferences of a specific community.

Another evolution in the concept of Social Search emerged with the advent of Web 2.0. Because in Web 2.0 everyone can create content, users have created annotations for Web pages at a surprising speed. An example of a popular bookmarking site was Delicious, with millions of registered users. Annotations represent new information to be exploited to improve IR processes. In [9], the authors integrate social annotations into the concept of Social Search by proposing two new algorithms based on PageRank: Social-SimRank and SocialPageRank. Accordingly, the concept of Social Search has been expanded. In [10], the authors define Social Search as "*an umbrella term used to describe search acts that make use of social interactions with others. These interactions may be explicit or implicit, co-located or remote, synchronous or asynchronous.*" Following such a broad definition, in [11], the authors study the effectiveness of different social network types for personalization: (i) familiarity-based network, (ii) similarity-based network, and (iii) overall network. They use the notion of Social Search to describe the search process over Web 2.0 *social data* representing different entities and

their interrelations.

A few studies in the literature have proposed categorization of Social Search research. Authors of [12] rely on the simple notion that “*Social Search refers to the use of social media to aid finding information on the Internet,*” identifying five crucial dimensions for understanding Social Search:

1. **Collaboration - asynchronous vs. synchronous:** in an asynchronous scenario, the data coming from social media are used to improve the Web search without any interaction between users. Vice versa, the synchronous scenario involves user interactions during the search process;
2. **Collaboration - implicit vs. explicit:** the tacit collaboration, as for the asynchronous scenario, is where the search system implicitly integrates social information obtained from users. When people interact during the collaboration, we talk about explicit collaboration;
3. **Search target - finding people vs. finding resources:** Social Search is often seen as searching for people, while Social Search systems offer a means to also search for resources by considering social information;
4. **Finding - search vs. discovery:** Social Search includes traditional information retrieval mechanisms for searching by re-ranking search results based on social information. However, it also offers the possibility to discover information and resources. For example, on a social bookmarking site, the user can discover new websites that generally are not returned by the system during a search process;
5. **Search results - sense-making vs. content selection:** most of the time, people do not fully understand why they get specific results. Social Search can improve sense-making and content selection, providing contextual information.

Overall, the above five dimensions of Social Search are strictly interconnected and work together to enhance the search process by providing users with relevant information and context to understand search results. The Social Search process starts considering the search target, which can be finding people or resources. The finding process can be distinguished into search or

discovery. During the finding process, people establish some kind of collaboration that can be asynchronous or synchronous and implicit or explicit. Finally, the Social Search process can improve the sense-making of the results or help users in selecting the right content. By understanding these dimensions, researchers and developers can design and implement effective Social Search systems that meet users' needs.

Finally, in [3], the authors propose a taxonomy of social IR defined as *“the process of leveraging social information (both social relationships and the social content), to perform an IR task with the objective of better satisfying the users' information needs.”* The taxonomy includes Social Web Search, Social Search, and Social Recommendation. According to the authors, Social Web Search deals with techniques that can improve IR processes by using social information. At the same time, Social Search is *“the process of finding information only with the assistance of social entities, by considering the interactions or contributions of users.”* In this way, Social Search is associated with platforms specifically designated for managing social data, such as Facebook or Twitter. The authors further divide Social Search into three categories: Social Question&Answering (Q&A), Social Content Search, and Social Collaborative Search.

### 3. Paper Selection Methodology

Before presenting the literature according to the taxonomy of Figure 1, we briefly describe the methodology used to identify relevant works on Social Search. Table 1 reports summary data. We started collecting documents following the Social Search taxonomy proposed in [3]. For each topic (first column in Table 1), we first retrieved papers from Google Scholar using the keywords specified in Table 1. We filtered results based on the year of publication: we chose to consider the works published from 2018 onward to focus on the most recent results in this area and to offer readers up-to-date knowledge. We selected the first 20 results, then analyzed these papers and kept only the most relevant ones according to the topics. Using these papers as initial seeds, we explored the literature using ConnectedPapers<sup>1</sup>. ConnectedPaper is a visual tool to find papers relevant to a specific field. Given a queried paper, Connected Papers explores the Semantic Scholar Paper Corpus to identify papers with the strongest connection: this is done

---

<sup>1</sup><https://www.connectedpapers.com>



by building a directed graph where nodes are related papers and edges are weighted according to a similarity measure based on Co-citation and Bibliographic Coupling [13]. ConnectedPaper returned 41 related papers for each one of the queried works. We removed duplicated papers, and, as done for Google Scholar, we only considered the papers from 2018 onwards. The total number of papers collected through this process is reported in the “CP” (Collected Papers) column in Table 1. These papers also include articles cited by the most relevant works (according to our opinion) and those citing each work. The set of collected papers is the starting point for our detailed analysis. Specifically, after reading those papers, we kept only a subset of them, whose number is shown in the “SP” (Selected Papers) column of Table 1. Although collecting papers through keywords or tools (such as Connected Papers) produced many articles, only a small percentage of them used community information to improve information retrieval within social platforms: this explains the substantial decrease in the number of selected papers.

Further scrutiny of those papers leads us to report a subset of them in the survey, as indicated in the “RP” (Reported Papers) column of Table 1. The main criteria we used for filtering those papers are (i) the impact of the work in the community (by analyzing citation scores), (ii) the originality of the proposed approach concerning the literature, (iii) the quality of the publication venue; (iv) the level of innovation on the topic. The most crucial factor that influenced the exclusion or consideration of a paper was the use of social data: we focused more on works that integrated data relating to social relationships between users. The number of citations and the quality of the venue were secondary factors. Given that the number of citations reflects the article’s impact on the scientific community, we preferred studies that attracted citations to those not cited. Finally, we evaluated the originality of the approach and the novelty brought to this research field.

#### 4. Expert Finding in Social Q&A

Community Question&Answering sites host domain-specific communities where people share their experiences to help other users through point-to-point interactions. However, the engagement of expert users in the community is complex, and not receiving any answer can create a sense of frustration in a user posting a question. A question can fall through unanswered due to several factors, but one of the main reasons behind this behaviour is that the system needs to propose the question to the right expert users. This con-

Topic	Keywords	CP	SP	RP
<b>Expert Finding</b>	Expert Finding in Community Question&Answering, Expert Representation, Expert Recommendation, Question Routing, User Expertise, Expert Ranking	240	23	15
<b>Question Retrieval</b>	Question Retrieval in Community Question&Answering, Duplicate Question Detection	190	27	11
<b>Answer Selection</b>	Answer Selection in Community Question&Answering, Answer Ranking	105	29	5
<b>Content Search</b>	Social Content Search, Social Platforms, Social Information Retrieval, Content Search in Microblogs, Query Suggestion, Query Expansion, Query Auto-Completion	240	29	10
<b>Social Collaborative Search</b>	Collaborative Search System, Collaborative Web Searching, Collaborative Information Seeking	208	22	10

Table 1: Keywords and the number of collected (CP), selected (SP), and reported (RP) papers for each topic.

sequence is reasonable, considering the large number of questions submitted to these sites daily. The task of adequately matching user questions and expert users is commonly called the *Expert Finding Task*. Previous surveys divide the existing literature according to the specific methodology to address the above problems. In [14], the authors classify the studies into eight categories: simple models, language models, topic models, network-based methods, classification methods, expertise probabilistic methods, collaborative filtering methods, and hybrid methods. Instead, in [15], the authors classify the studies into matrix factorization-based models, gradient boosting

tree-based models, deep learning-based models, and ranking-based models. We formalize the problem as follows:

**Definition 1.** *Given a question  $\mathbf{q}$  posted by a user  $\mathbf{u}$ , a set of  $\mathbf{n}$  experts  $\mathbf{E} = \{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_n\}$ , and a positive integer  $\mathbf{k} \leq \mathbf{n}$  find the best ranked list of  $\mathbf{k}$  experts who are more likely to answer correctly question  $\mathbf{q}$ .*

This task involves mainly two phases:

1. **Expert and Question representation:** the representation of experts should capture their *knowledge* and *interests*. Usually, past questions and answers are the information source considered to extract topics addressed by the experts and their interests. The *reputation score* is also essential as it captures the community’s view of the expert quality. Similar to the reputation score, some studies compute an *authority score* that involves the construction of an interaction network on which researchers apply link analysis techniques. The final representation model should cover expertise, reputation, user activity, and the experts’ likelihood of answering the question. Analogously, question representation models the topic involved in the question and, in some cases, the interest and profile of the asker.
2. **Expert-Question similarity and ranking:** a score for each expert-question pair is computed and used to rank the experts based on some measure of interest, e.g., the expert’s likelihood to answer the given question.

Unlike these prior works, we classify existing literature according to the two phases described above, representing a more comprehensive way of characterizing the topic. Therefore, the following subsections present two required modules in any workable Q&A system.

#### 4.1. Expert and Question Representation

The expert and question representation phase aggregates all the user aspects (i.e., activities and reputation) to capture their expertise and represent queries, permitting an effective matching between them. The first group of work focuses on encoding information about the user’s profile, generally described as features fed into conventional topic modelling or word embedding tools. In [16], authors propose a method to detect a group of possible answers based on their history of questions and answers. The authors use features

from the user’s profile and the community to model the user. Precisely, they formalize a measure called *TagTrueness* ( $TT$ ) that captures the tendency of a user to answer questions with the tags in her profile:

$$TT = \frac{\text{Total Number of Accepted Answers of Profile Tag}}{\text{Total Number of Accepted Answers}}$$

Given a question, users belong in the set of possible experts if they have given at least one accepted answer to a question with the same tags. Each user is associated with a score computed considering  $TT$ , the Number of Tags in the Profile ( $NPT$ ), the Number of Accepted Answers ( $NAA$ ), the Average Number of Accepted Answers with Tags in Common between the question and profile ( $AACT$ ), and a Recent Activity Measure ( $RAM$ ). The idea is that users who have *recently* answered questions with tags in common with the question should have a higher score than users who have responded to similar questions in the past. The score is computed as follows:

$$Score = TT \cdot \log(NPT + 10) \cdot \log(NAA + 10) \cdot AACT \cdot RAM \quad (1)$$

In the same group (user’s features modelling), [17] proposes a framework combining textual information, community feedback, and user interactions. Authors use word embeddings to represent users’ answers and degrees of expertise to capture the semantic relationship between the question and user expertise. The proposed approach overcomes the well-known limitations of Latent Dirichlet Allocation (LDA) [18] for topic modelling. On the other hand, the original approach of [19] is to define embeddings that favour experts who provide few but high-quality answers to complex questions rather than experts who offer solutions that do not satisfy the user or answer only accessible questions. Finally, [20] accounts that sometimes it is possible to learn from past answers/questions which topic a user could face with some expertise. For example, suppose it is possible to deduce from the user information that she is a researcher: the system could forward the question “How can I write a scientific paper?” even if she had never answered this topic. To this aim, the authors define a new approach based on the Recurrent Memory Reasoning Network (RMRN), composed of different reasoning memory cells that implement attention mechanisms to focus on various aspects of the question and retrieve relevant information from the user’s history. Similarly to [17, 20], word embeddings are also used in [21] to enrich the new question with tags semantically similar to those already present to include also experts that, considering only the original tags, would not be considered.

Even if the studies above consider some community-based features, they do not explicitly consider the social network between answerers and askers. Expert Finding approaches based primarily on text, such as those mentioned above, aim to quantify the user’s knowledge about a topic based on the questions/answers asked/given in the past. On the other hand, network-based approaches aim to determine the user’s authority within the network by considering past activities independently of the new question. In [22], authors introduce three new concepts: question hardness, question answerer association, and answerer’s performance-enhanced authority. These three concepts are based on three assumptions, respectively: (i) complicated questions require a good knowledge of the topic and help identify the experts; (ii) an answerer is very familiar with a given topic if she has answered multiple questions on that topic; (iii) good answers usually get many votes, so merging information from the network can help. The framework includes the Knowledge Analyzer component aimed to estimate an answerer’s knowledge of a topic by combining the output of the following three components:

- **Question similarity finder:** it computes the similarity between the new question and archived questions by treating them as documents and using Query Likelihood Language (QLL) [23];
- **Question answerer association provider:** it establishes the relationship between a question and each answerer using conditional probability;
- **Question hardness estimator:** it computes the complexity of the question using features such as Time Response and Number of Answerers. The assumption is that the delay in receiving the first answer to the question and a low number of replies are indices of the complexity of the question.

The Authority Analyzer is network-based and consists of two parts: Network Builder and Authority Estimator. The first deals with building a Competition Based Expertise Network (CBEN) [24] where each node is an answerer, and there is a direct link from  $a_i$  to  $a_j$  if both users answer the same question and  $a_j$  is the best answerer of the question among the ones available. The weight of the link  $v_{i,j}$  measures how much  $a_j$  is better than  $a_i$  by considering the number of votes received, and it is computed as follows:

$$v_{i,j} = \sum_{q \in Q} \delta_i^j(q) \cdot \sigma(a_i, q) \quad (2)$$

where  $\delta_i^j(q)$  is equal to 1 if  $a_i$  and  $a_j$  are co-answers of  $q$  and  $a_j$  is the best answer, 0 otherwise. Instead, the factor  $\sigma(a_i, q)$  incorporates into the computation the information about the quality of the answer  $a_i$  concerning the question  $q$ , and it considers the number of votes received by  $a_i$  for  $q$  and the maximum votes received by any answerer for question  $q$ . Instead, the Authority Estimation calculates the authorship of the answers using link analysis techniques such as AuthorRank [25] and Weighted HITS [26].

Other variations of this approach have been proposed: [27] presents a Preference-Enhanced Hybrid Expertise Retrieval (PEHER) system with a preferability estimator. The preferability estimator incorporates the preferences of an answerer  $a \in A$  by considering two preferences: *intra-profile* and *inter-profile*. The intra-profile ones capture a given user’s preferences for specific terms. In contrast, inter-profile preferences capture the features of the particular answerer concerning all other relevant answerers in the system.

#### 4.2. Expert-Question Similarity and Ranking

Depending on the models adopted for representing experts and user queries, we have different solutions for matching and ranking experts likely to answer a given question effectively. Most works cited in Section 4.1 also address this problem. Therefore, we first briefly describe the approach proposed by them.

In [16], after ranking the experts using the score computed by Eq. 1, the candidates are re-ranked based on their answering behaviour by considering if a user is generally more active during the day or night, during the week, or on the weekend. Given a question and the corresponding posting time, the experts on the question topic are re-ranked by putting in the top positions the experts that usually are active at that specific time of day. Similarly, Kundu et al. [28] rank the experts by combining their activeness over time with other features measuring expertise and answering intensity. Mumtaz et al. [17] rank the experts using a weighted combination of the cosine similarity between users’ expertise and question representations. In [19], the authors apply the Learning to Rank methods to the expert ranking task. Specifically, they use the LambdaMART algorithm to learn a ranking model trained on the pairs  $(Questions, Users)$  modelled with the four categories of features described in the previous section and labelled based on the historical interaction information. For a new question, the candidate experts are ranked by the score predicted by this model. In [22], the knowledge and authority scores described in the previous section are combined using a Reciprocal

Rank Fusion (RRF) technique [29]. RRF is also used in [27] to generate the expert list.

Other works in the literature focus more specifically on the issue of Expert-Question Similarity and Ranking, while they do not deal mainly with the problem considered in Section 4.1. Conceptually similar to [22, 27], in [30], the authors develop a network-based framework combining four features: (i) the similarity between the question’s content and the user profile; (ii) the similarity between question topics and user topics; (iii) the similarity between asker and answerer in the network; (iv) the activity level of the user. Authors compute the similarity of the content through cosine similarity, incorporating the Term Frequency - Inverse Document Frequency (TF-IDF) to weigh the words differently. Instead, they first use LDA to extract topics and then the TF-IDF again to calculate the similarity between topics. For the similarity between the answerer and asker within the social network, they build a graph with an edge between two user nodes  $u_1$  and  $u_2$  if the user  $u_2$  has answered a question of user  $u_1$ . Authors compute the proximity of two nodes by performing Random Walks with Restart (RWR): precisely, given the graph’s size, they apply the fast RWR algorithm proposed in [31], whose intuition is to divide the graph into small communities connected by bridge edges and combine the RWR scores computed on these communities. Finally, authors use metrics available on the platforms for the answerer’s activity level, such as the reputation score or the number of awards received. Lastly, the authors compute the final score by combining these four factors weighted by parameters learned using the history of the response activity. The study of [32] also considers the answers’ quality and consistency. The authors propose a model based on an adaptation of the bibliometric g-index to measure the consistency of the user in providing high-quality answers. They also compute another score that measures the user’s reputation (REPF) using voter reputation, up-vote to down-vote ratio, participant-based reputation, and popular tags. Weighted Exp-PC combines the two scores to compute the final user’s expertise.

Another aspect to consider when routing new questions to experts is that, in a cQA site, each user can cover both the asker and answer role simultaneously, which may vary over time. In [33], the authors represent the user considering the evolution of her role over time, and they study how much this aspect can improve the question routing process in a cQA site. To track the evolution of user roles, they propose a Time-aware Role Model (TRM) based on LDA to model the latent topical relationship between question

content, user, time, and role.

Moreover, many approaches do not consider the intimacy between the asker and the answerer. In [34], the authors consider this factor based on the assumption that an expert has more incentive to answer a question posted by an asker if she is interested in the topic and if there is intimacy between the two. They propose a User Intimacy Model (UIM), i.e., an LDA-style model that incorporates social interactions in modelling and learning the intimacy between the entities on topics.

Finally, one further aspect to consider in the Expert Finding task is the probability that the expert to whom a given question is forwarded will respond. To approach the problem, according to [35], it is necessary to find a ranking function that quantifies a user’s expertise and the likelihood that they will answer the question simultaneously. To this end, the authors exploit the *Social Identity Theory* [36, 37] to define a graph joining different possible experts who could answer the question.

## 5. Question Retrieval in Social Q&A

Users often post questions that already exist in the system and have been answered. Posting an already-asked question on a cQA site can negatively affect user engagement by not getting any answers from community experts. It also causes an unnecessary increase in the amount of data the system stores. For this reason, cQA sites provide a service whereby, given a user question, they return all past questions, which may be the same or very similar to the user’s question. We can formally define the task as the following retrieval task:

**Definition 2.** *Given a query  $q$ , a historical set of already posted and successfully answered questions  $Q = \{q_1, \dots, q_m\}$ , and a positive integer  $k$ , return a ranked list of the  $k \leq m$  historical questions that are the most similar to  $q$ .*

At this point, the user submitting  $q$  is presented with the answers of the  $k$  similar past questions and can quickly decide whether the answers satisfy the information need or whether to post the new question. However, the previous task is complex: different users with the same information need can write semantically equivalent questions using completely different sentences from a syntactical point of view. A significant body of works tries to overcome this *semantic gap* problem using NLP techniques and user metadata information.



As highlighted in Figure 1, the problem consists of two logical phases: *Metadata and Question Representation* and *Query-Question Similarity and Ranking*. The two phases are conceptually similar to those in the Expert Finding problem but now refer to a different task. In the Metadata and Question Representation phase, the proposed approaches address the problem of representing the questions and which user metadata could improve the performance of the Question Retrieval service offered by the site. Instead, in the Query-Question Similarity and Ranking phase, the proposed approaches develop methodologies to exploit the content of the questions and leverage the user information available to provide accurate matches between a new query and existing questions. We dedicate the first subsection to the Metadata and Question Representation. Instead, we describe the Query-Question Similarity and Ranking methodologies dividing them into three subsections corresponding to three approaches that, as shown in Figure 1, are alternatives to address this problem: (i) Text-based Similarity and Ranking, which only use only textual data; (ii) Social and Text-based Similarity and Ranking, which also introduces social aspects by modelling users and interactions; (iii) Domain Adaptation, where data is adapted or integrated to improve performances.

### 5.1. Metadata and Question Representation

A very popular technique to represent questions and answers relies on Word Embeddings, which transforms a sentence into a fixed-size dense vector, somehow capturing the semantics of the sentence. In [38, 39] authors use GloVe, while in [40, 41, 42] authors use Word2Vec. For example, in [41], after the first filtering phase in which authors apply *text cleaning*, *tokenization*, *stopwords removal*, and *stemming*, each question is represented as a list of terms. Each word is mapped into a fixed-length vector using a pre-trained Word2Vec model. Considering that short texts amplify the mismatch problem, the authors propose expanding the query with other terms close in the semantic space created by the Word2Vec model.

An interesting study from the question representation point of view is presented in [43]. Authors try to address five problems when representing questions: *synonymy*, *polysemy*, *word order*, *question length*, and *data sparsity*. The framework consists of two steps:

1. **Word and concept learning:** they use a Skip-gram model to learn the embeddings of the words. In this way, the meanings of words

are distributed across the dimension of a semantic space. Furthermore, previous works show that conceptual information can handle the polysemous problem: in [44], the authors compute the contextual representation of a word by combining its vector and the most context-appropriate concept vector that delivers an unambiguous meaning. For example, considering two queries [python zoo] and [python string], the embedding of the first “python” is obtained by combining the intrinsic vector of “python” and the concept vector of “animal category.” In this way, the model is context-dependent. The problem with the above approach is that the objective function locates the right concept and then searches for a word underneath the chosen concept. In [43], the authors add a regularization function to overcome the problem, which aims to leverage the concept information regarding the knowledge base without considering the context. Considering the context-independent relation could avoid the error introduced by choosing the wrong concept vector;

2. **Question and answer embedding learning:** to encode concept, syntactic, and word order information, authors propose the High-level feature embedded Convolutional Semantic Model (HCSM). The model assumes that if two questions have similar answers, they could have semantic relations. HCSM is a convolutional architecture composed of wide convolution (without zero-padding) and pooling. The authors design three variants: the first variant emphasizes the relation (i) between concepts, (ii) between words, and (iii) between concepts and words. The second variant emphasizes only the first two relations, (i) and (ii). The last variant combines the word-embedding question matrix and concept-embedding question matrix before inputting them into the network.

### 5.2. Text-based Similarity and Ranking

Joty et al. [45] provide a remarkably complete approach to addressing this problem. Specifically, the authors observe that some questions may have a lot of answers, some significant and some useless. A long thread of answers for a question can bore the user who decides not to read it and post a duplicate question. The authors thus try to accomplish the two tasks by introducing a third task: checking whether an answer in a long answer thread is good. The three sub-tasks are defined as follows: (i) determine whether the  $m$ -th

answer  $c_m^i$  in a given thread is good or not for its related question  $q_i$  (task  $A$ ); (ii) calculate whether an archived question  $q_i$  is relevant to the new query  $q$  (task  $B$ ); (iii) rank each answer for  $q_i$  as relative or non-relative for  $q$  (task  $C$ ). These tasks have dependencies: if an answer  $c_m^i$  is good for the archived question  $q_i$  and  $q_i$  is relative to the query  $q$ , then  $c_m^i$  is a good answer for  $q$ ; if an answer  $c_m^i$  is relevant to  $q$ , then the question  $q_i$  to which it refers is also inherent to  $q$ . Therefore, the idea is to jointly exploit the interactions and dependencies between these tasks to solve the Question Retrieval and Answer Selection tasks. The framework consists of two phases:

1. A Deep Neural Network (DNN) is used to solve the three tasks separately. The subtask-specific hidden layer activations are used as the embedded feature representations for the second step, where a Conditional Random Field (CRF) model performs joint learning with global inference exploiting the dependencies between the three subtasks. As shown in Figure 2(a), the input for the first step is a tuple  $(q, q_i, c_m^i)$  composed of the new query  $q$ , a historical question  $q_i$ , and the  $m$ -th answer  $c_m^i$  of the question  $q_i$ . These elements are converted into fixed-length vectors through syntactic and semantic embeddings and passed to a feed-forward Neural Network intending to learn task-specific embeddings for the three tasks separately. Considering task  $A$ , shown in the lower part of Figure 2(a), the vectors of answer  $c_m^i$  and question  $q_i$  are concatenated and passed to the first hidden layer of the DNN. The output is passed to a task-specific hidden layer that combines these signals with a pairwise similarity between  $c_m^i$  and  $q_i$ . The merging between the final hidden-layer output and the pairwise similarity gives the final task-specific embedding. The last layer outputs the prediction variable  $y_{i,m}^a$ . Similarly, the authors compute the embeddings of tasks  $B$  and  $C$ ;
2. The authors build a large undirected graph consisting of six sub-graphs:  $G_a, G_b, G_c$  for the three tasks and  $G_{ab}, G_{ac}, G_{bc}$  for the three possible interactions between the different tasks. The *intra-subtask* edges are created for the variable of the same sub-task, i.e.,  $y_i^b$  and  $y_j^b$ . Instead, regarding *across-subtask* edges, authors exploit three different types of connection: (i) null or no connection, (ii) 1:1 connection between tasks  $A$  and  $C$ , i.e.,  $y_{i,m}^a$  and  $y_{i,m}^c$ , and (iii) M:1 connection to  $B$ , where all the nodes of subtasks  $A$  and  $C$  are connected to the thread-level  $B$  node. Each node  $u$  is associated with its embedding vector  $x_u$  and its output variable  $y_u$ . Each edge that links the two nodes  $(u, v)$  is associated

with an input-feature vector  $\mu(x_u, x_v)$  composed of node-level features, which are (i) comment features and (ii) meta-features. Dependencies between the output variables are modelled by learning node and edge factor functions that jointly optimize a global performance criterion using the CRF model [46].

The results of the experiments show that the DNN alone achieves good results. Still, the CRF model allows a global inference on a graph structure, considering the dependencies between the different tasks.

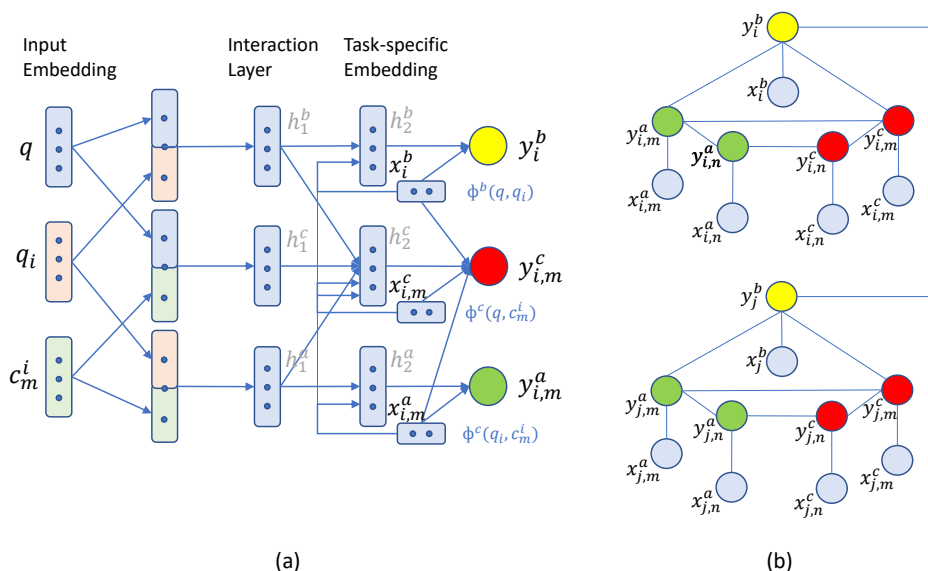


Figure 2: Graphical representation of the cQA framework in [45]. On the left (a), we have three feed-forward neural networks to learn task-specific embeddings for the three cQA subtasks. On the right (b), a global conditional random field (CRF) models intra- and inter-subtask dependencies.

Other approaches provide less rich but noticeable solutions for the Similarity and Raking task. The authors of [47] propose an unsupervised framework to compute the similarity between two questions, called Reduced Attentive Matching Network (RAMN). It is based on an attention autoencoder to semantically represent the question, pre-trained using a large amount of unlabeled data, and an attention mechanism to focus on the different parts of the sequence of terms. A different approach to capture the similarity between questions is designed in [48], where authors use a Convolutional Neural

Network (CNN) whose input is a set of six matrices deriving from different measurements of similarity between question pairs. Four matrices are vector distances, and two are interaction measurements. These matrices are the input of a Deep CNN with a last fully connected layer which outputs the probability of the question pair being duplicated or related.

Finally, [39] uses a Siamese NN to address the problem. Two questions,  $q_1$  and  $q_2$ , are passed as input to two CNNs to capture their most meaningful features,  $\hat{q}_1$  and  $\hat{q}_2$ . Two additional operations are then performed on  $\hat{q}_1$  and  $\hat{q}_2$ : (i) matrix multiplication to compute the similarity and (ii) matrix reduction to compute the closeness between the two questions. The result of these two operations is combined with  $\hat{q}_1$  and  $\hat{q}_2$  and given in input to a Multi Perceptron Layer (MLP) that outputs the duplicate and non-duplicate label.

### 5.3. Social and Text-based Similarity and Ranking

In [40], the authors propose a framework that simultaneously learns the content of questions, their category, and the information in the users' social network to represent the question's intent richly. The framework is illustrated in Figure 3. The assumptions under this approach are two: (i) a user tends to ask questions similar to friends and colleagues because it is likely that they have common interests, and (ii) questions in the same category are more similar than those belonging to different categories. Considering the above observations, the authors build a heterogeneous graph composed of three nodes: users, questions, and categories. A question node is connected to its category node and the user who posted it. This graph is used to sample paths that include different types of nodes applying deep Random Walks inspired by [49]. Questions are represented using pre-trained word embeddings and LSTM. User embeddings are instead learned from matrices of social relations  $M \in R^{(m \times m)}$  where entry  $s(i, j) = 1$  if users  $i$ -th and  $j$ -th are friends. Given a new query  $q$  along with asker information at inference time, the proposed approach learns, through this heterogeneous network, the latent representation of the queried question. Specifically, it learns the heterogeneous network of graph  $G$  and then embeds the asker  $u$  and the queried question  $q$  to return the top- $n$  similar questions submitted in the past.

Finally, in [50], authors address the problem of detecting *misflagged* duplicate questions, i.e., questions marked erroneously as duplicates. It is a complementary approach to the one described in [40], and authors use social

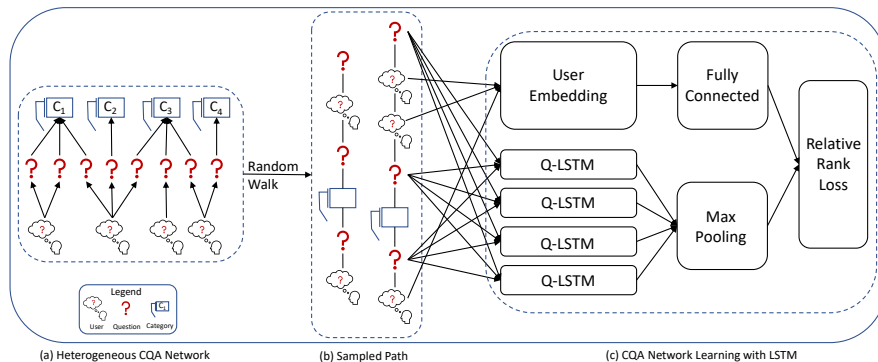


Figure 3: The overview of heterogeneous network ranking learning. (a) The heterogeneous CQA network is constructed by integrating questions content, corresponding categories, and social network relations. (b) A deep random walker walks on the heterogeneous CQA networks to sample the data paths. (c) The questions, askers and categories are encoded into fixed feature vectors and used to train the model for scoring the questions [40].

information related to users (such as their reputation and the scores of their answers) by defining features that help identify errors in marking duplicates.

#### 5.4. Domain Adaptation

Finding duplicate questions is a costly process, which is alleviated when experts manually mark duplicate questions. However, some cQA platforms do not offer this functionality, making the Duplicate Question Retrieval process more intricate. One solution is to use Domain Adaptation from another forum. The authors in [38] propose an approach that uses Adversarial Domain Adaptation to help cQA sites that do not have data on duplicate questions. Authors study when the transfer domain can work and what domains' properties are important for successfully transferring knowledge from the source domain. The model consists of a question encoder, a similarity function, and a domain adaptation classifier. The encoder maps the sequence of tokens representing a question into a dense vector. The cosine similarity function predicts if two question vectors  $v_1$  and  $v_2$  are duplicates. Finally, the domain adaptation classifier predicts whether a given question is part of the source or target domain. The encoder is trained not only to perform well with source data but also to be incapable of distinguishing between question pairs from the source vs target domain. The adversarial component reduces the difference between the target and source distributions. For the domain adaptation component, the authors choose the Wasserstein method

[51], which diminishes the Wasserstein distance between the source and target distributions. The experiments use three datasets: StackExchange, SprintFAQ, and Quora. The first two datasets are focused on technical domains, while the Quora dataset covers many different topics. Results show that Domain Adaptation works well when the source and target domain are similar (i.e., with StackExchange and SprintFAQ datasets).

Finally, Domain Adaptation is also used to exploit questions written in different languages concerning the target query. These methods include translating the original question written in L1 to the target language L2 and then using a monolingual Question Retrieval model. A relevant example of this approach is presented in [52].

## 6. Answer Selection in Social Q&A

Answer Selection is an essential process in cQA as it automatically identifies the most valuable answers for an incoming question and proposes them to the user to increase her engagement and satisfy her information need faster. The two main problems for this task are that: (i) answers are usually noisy, and (ii) it is likely that they address different facets of the associated question, not all relevant to the current information need. The problem can be formalized as follow:

**Definition 3.** *Given a new query  $\mathbf{q}$  and a similar historical question  $\hat{\mathbf{q}}$  along with its thread of  $n$  answers  $\mathbf{A} = \{\mathbf{a}_1, \dots, \mathbf{a}_n\}$ , the Answer Selection problem asks for selecting the best answer in  $\mathbf{A}$  relevant for  $\mathbf{q}$ .*

Several state-of-the-art works cast the Answer Selection problem to a classification or a ranking task and face it by considering mainly the text of the candidate’s question and answer pairs. Notable examples of this approach, identified as *answer-based methods* in the taxonomy presented in Figure 1, are [53, 54], where semantic self-attention and co-attention mechanisms are exploited to focus on the interaction between questions, answers, and their contexts. In the following, instead, we focus on studies that build a graph to model questions, answers, and user expertise, trying to exploit hidden relations between all the entities involved in the selection process.

### 6.1. Graph-based

The work in [55] proposes a framework that jointly addresses the *redundancy*, *heterogeneity*, and *multi-modality* of questions and answers of the

CQA platforms. Text and images are first converted into feature vectors using word embeddings and the ResNet deep CNN model [56], respectively, to consider both textual and visual content. An attention mechanism is then applied to reduce redundancy by constructing a question-answer attentive interaction matrix that focuses on helpful word-pair interactions. Moreover, the authors build a heterogeneous network in which the vertices are questions, answers, users, and tags. Instead, the edges represent the relations among these entities. A meta-path heterogeneous network embedding algorithm is applied to leverage the social information, which allows the discovery of special neighbours of nodes through specific meta-path schemes. For example, considering the schema “*answer-user-answer-question-tag*,” the last vertex *tag* could be viewed as a special neighbour of the first answer node in the schema. The aim is to learn representations  $\vec{u}_q$  and  $\vec{u}_a$  of questions and answers that consider indirect relations among question, answer, user and tags. Finally, based on the idea that the similarity between word-pair can contribute differently to the final score in a different context, the authors incorporate context information in the similarity computation.

Liu et al. [57] observe that modelling user expertise only through historical answers, as in [53, 54] works well only if sufficient data is available. They thus investigate the impact of *static* and *dynamic* social influence on the answer selection problem. They claim that a form of static influence is applied indirectly by other experts or users a person follows in the community: users with a passion for a field will follow experts in that field and read their posts/answers to improve their knowledge of the field. Instead, the dynamic influence refers to dynamic personal interests: users can have different interests, but only a few are activated when they face a new question. Considering all the user interests as equally important can thus negatively affect expertise modelling. Relations in cQA sites are modelled as a directed graph where the nodes are users, and there is an edge from  $u_1$  to  $u_2$  if  $u_1$  follows  $u_2$ . The framework is illustrated in Figure 4. The authors compute social influence in two phases. During the first phase, the user embedding is the concatenation of the user interest feature vector  $x_u$  (obtained using the word embedding of explicit user topics) and a randomly initialized latent vector  $p_u$ . The second phase updates the embedding with  $k$  aggregation steps by incorporating, at each stage, the information of the user’s neighbours nodes. Moreover, the authors model user interest dynamics through a context-based dynamic representation. The embeddings of topics representing user interests are stacked, thus composing a topic embedding matrix. They learn a



hidden representation of each topic through a dense layer. Assuming that the answer given by the user is a personal expression containing information about the topics activated at that moment, the authors compute the weights for each topic using the specific answer by a multi-layer perceptron attention mechanism. Instead, personal interests consider the weighted sum of the interesting topics and the global user interest obtained by applying the mean-pooling overall topic representations. The representation  $u$  of user context is finally obtained by concatenating social influence and personal interest information. Answer selection is computed by a matching layer that takes as input a triple  $(q, a, u)$  and outputs the matching score for ranking the candidate answers.

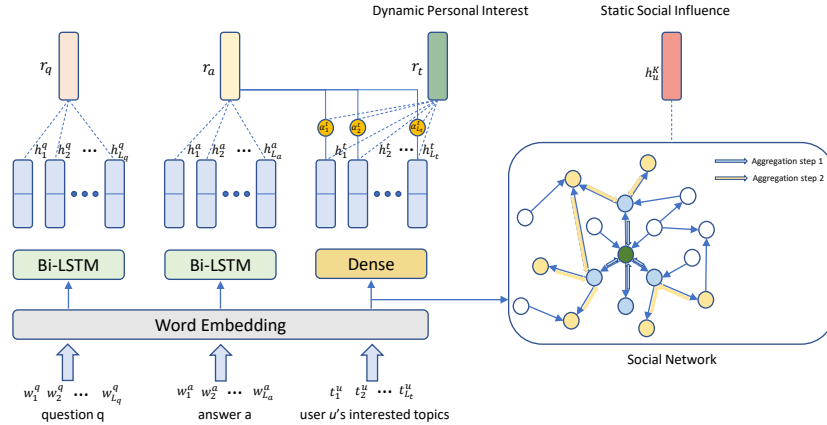


Figure 4: Overview of the proposed SCAD model. It encodes the user expertise by jointly considering the dynamic personal interest and the static social influence. The aggregation process of the social influence takes  $K = 2$  as an example [57].

Similar to [57], the authors of [58] also remark that users have many interests, but only a few are relevant to a specific question. They address the answer selection process by learning a Graph Neural Network (GNN) from a graph modelling the relations among questions, answers, and terms, weighted with TF-IDF values to capture their different importance for answers and questions. They introduce a novel GTAN (Graph-based Tri-Attention Network) model to encode answer correlation into their representations. Based on the GNN representation, target-aware answerer representation, answer-specific question representation, and context-aware answer representation are computed through attention mechanisms. These three representations are used to measure the final score for the answers. The approach outperforms

the one used in [54], demonstrating the possibility of explicitly encoding answer correlations by GNN modelling and learning answer-specific question representations and target-aware respondent representations.

## 7. Social Content Search

Social Content Search refers to the process of retrieving relevant content on social platforms by exploiting information such as clicks, social annotations, social interactions, and personal interests. We broadly divide this body of works into two classes presented in Sections 7.1 and 7.2, respectively. As shown in Figure 1, the two classes are alternatives for facing the Social Content Search task. Approaches in the first class do not modify the issued query and use social information to determine the relevance of the content that possibly answers the query. Instead, approaches in the second class manipulate the given query, exploiting social information sometimes.

### 7.1. Social Content Search without Query Modification

Most of the time, Content Search uses social information to define specific features in a more complex representation of users, which also involves representations of the content they have generated. In this sense, social information should be defined as any information about the interaction between users or between users and the content they provide (e.g., likes, retweets) available on the social platform.

The work in [59] is a relevant example in this class. Content modelling always remains essential for retrieval, and, as discussed above, LDA is widely used in this area. The LDA approach has limitations with short texts and sparse data. Sparsity is observed mainly on social platforms like Twitter, given the limits imposed on the length of posts. To overcome the problem, in [59], the authors propose a new pooling scheme for Twitter topic modelling that groups tweets based on authors who belong to the same community. The method is called Community Pooling and makes use of a social graph. The nodes in this graph are users, and an edge connects two users based on their retweet behaviour. Authors extract communities using the Louvain method for community detection [60]. The authors associate each community with a document containing all author’s tweets who belong to the community. This way, the number of documents is small, but they have enough words for the LDA algorithm to work well. The method is compared with baselines that group tweets by author, hashtag, or thread of replies under a tweet.

The methodology works well for the document retrieval task, where, given a query, the model returns the most similar tweets based on their LDA topic similarity.

Xu et al. [61] use a conceptually similar approach tailored to platforms allowing for the annotation of documents. Specifically, each document is associated with a “document” vector, which is the outcome of an LDA to extract its topic distribution, and a “social” vector, which represents the distribution of topics for users who annotated the document. Authors of [62] focus on Twitter and design a system whereby a range of information related to tweets is used for a complex ranking process to define the relevance of tweets. Specifically, features considered include (i) static features (e.g., the creation time, influence of author); (ii) real-time features (e.g., number of social actions received); (iii) query features (e.g., query terms and number of clicks for query results); (iv) personalization features (e.g., the follower-follower graph). It also includes specific information related to the platform’s social features, used to identify relevant tweets given a particular query.

Similarly, [63] focuses on Facebook instead of Twitter. The goal is to define a proper embedding of the user issuing the query and the possible documents to be provided. In addition to textual features, the embedding includes social features extracted from the interaction between users and between users and content considered for the query.

### 7.2. Query Expansion and Suggestion

To improve the performance of the retrieval process and increase the number of relevant documents returned as output, one of the most common techniques is Query Expansion (QE), which adds new terms to the original query. Formally:

**Definition 4.** *Given a query  $\mathbf{q} = \mathbf{t}_1, \dots, \mathbf{t}_n$  composed of  $n$  terms and a positive integer  $k$ , the Query Expansion (QE) approach adds  $k$  terms to  $\mathbf{q}$  that are likely to occur in relevant documents thus formulating an enriched query  $\mathbf{q}^* = \mathbf{t}_1, \dots, \mathbf{t}_n, \mathbf{t}_{n+1}, \dots, \mathbf{t}_{n+k}$*

In literature, many works perform QE by relying on document content. Instead, the expansion of terms also should include user interests and the context in a social context. An interesting study is the one of Bouadjenek et al. [64], where the authors propose an approach for selecting terms for query expansion that is based on (i) the semantic similarity between a query word

and a candidate term, (ii) the social closeness between the query and the user (how interesting a query term can be for the user), and (iii) a strategy for expanding user queries online. The approach consists of three phases: (i) select candidate terms that are similar and related to the query based on their co-occurrences with resources and users; (ii) profile the issuer user based on interests and activities; (iii) expand the query considering both the semantic similarity and the user’s profile. The method is based on *Folksonomies*, whose definition is the following:

**Definition 5.** *Let  $\mathbf{U}, \mathbf{T}, \mathbf{R}$  be the set of Users, Tags, and Resources, respectively. A folksonomy  $\mathcal{F}(\mathbf{U}, \mathbf{T}, \mathbf{R})$  is a subset of the cartesian product  $\mathbf{U} \times \mathbf{T} \times \mathbf{R}$  such that every triple  $(\mathbf{u}, \mathbf{t}, \mathbf{r}) \in \mathcal{F}$  is a bookmark, representing a user  $\mathbf{U}$  who used a tag  $\mathbf{T}$  to annotate a resource  $\mathbf{R}$ .*

The approach consists of two parts. The offline part deals with building the social graph and converting it into a tag graph of a folksonomy  $\mathcal{F}$ , where two tags are connected if they are semantically correlated. In addition, the offline part is also responsible for building and updating user profiles. The online part performs query expansion based on the tag graph and user profile. There are two approaches to constructing the tag graph, starting from the folksonomy:

- **Extracting semantics from resources:** semantically related tags are expected to occur on the same resources. The semantic relationship is computed through similarities that require a reduction in the dimensionality of the tripartite graph  $\mathcal{F}$  in a bipartite graph. Specifically, the authors use a function that performs a projective aggregation on the entire folksonomy  $\mathcal{F}$ , resulting in a bipartite *Tag-Resource* graph. Then, a  $T_R$  tag graph is extracted in which two tag nodes are connected by a weighted edge according to the Jaccard, Dice, or Overlap similarity metrics;
- **Extracting semantics from users:** the same users use related tags to annotate resources. Similar to the previous case, the authors perform a projective aggregation on the resource dimension by creating a *Tag-User* bipartite graph. Hence, they obtain a  $T_U$  tags graph using the metrics used to obtain  $T_R$ .

Choosing only one method can lead to a loss of information, so the authors combine the two graphs to get a single  $T_{UR}$  tag graph that incorporates all

the information. The edge between two tags is weighted using the Weighted Borda Fuse (WBF) technique. As explained below, the authors use the graph to extract terms that are semantically related to a given query term. As a further refinement, the user’s interests should be considered to return a result personalized for each user. In the context of folksonomies, a user annotates a resource using tags that summarize her understanding of that resource. These tags are a good summary of users’ interests. Therefore, each user is represented as a set of tags, and their weight is computed by adapting the TF-IDF measure based on the user’s profile. In the online part, the authors consider the similarity between a query term  $t_i$  and a potential candidate term  $t_j$  for query expansion based on tag graph and user interests. The first step is to extract the user profile, given a query and its issuer. Then, the authors extract all adjacent tags for each query term using the  $T_{UR}$  graph. For each candidate tag  $t_j$ , they compute a ranking metric that combines (i) the similarity of a query term  $t_i$  and the candidate term  $t_j$  given by the graph and (ii) the measure of user interest for tag  $t_j$  calculated considering the similarity between  $t_j$  and all the tags in the user profile. Finally, they order the terms of the candidates according to this ranking measure. Instead, in [65], the authors integrate social scores of users into the similarity between query and document, which are based on two criteria: positivity of a user and feedback.

An alternative to QE is Query Suggestion (QS), which is more powerful concerning synonymy and polysemy problems. The QS process offers the user alternative queries, allowing her to choose which one to explore. In [66], the authors propose an improved Query Suggestion system for LinkedIn by simultaneously modelling structured personalized user features and unstructured text data in a sequence-to-sequence model (Seq2Seq). Differently, Chen et al. [67] designed a Hierarchical Neural Query Suggestion system that combines a session-level and a user-level neural network to model a user’s short- and long-term search histories. Finally, in [68], the authors suggest a novel training approach that considers user feedback as additional machine translation.

Auto-completion is another search engine feature that provides possible query completions as the user is typing. An interesting study is the one of Jaech et al. [69], where the authors show how an adaptive language model based on a recurrent neural network can generate personalized completions.

## 8. Social Collaborative Search

Although the information search process is usually an individual action, it may involve the collaboration of several subjects. In this context, when two or more people collaborate to satisfy an information need, we speak about *Collaborative Search*. According to Golovchinsky et al. [70], the Collaborative Search can be studied along four dimensions: (i) the intent, which is explicit or implicit; (ii) the mediation, applied through user interfaces or algorithms; (iii) the concurrency, that can be synchronous or asynchronous; (iv) the location, which can be remote or co-located. Afterwards, Morris et al. [71] suggested two additional dimensions: the role (symmetric or asymmetric) and the medium (Desktop or emerging devices). Given the focus of this study, we present the literature according to the following categories: (i) solutions oriented towards specific communities, (ii) general-purpose solutions, and (iii) solutions based on chatbots. The three categories are alternative approaches, as Figure 1 illustrates.

### 8.1. Solutions Oriented Towards Specific Communities

With the specific aim of helping the academic world, in [72], the authors propose SciNoon, a system capable of facilitating the exploratory search process students and researchers perform in their daily work. Unlike traditional search, exploratory search requires a significant effort, mainly to estimate the results' relevance, where a direct collaboration can help make assessment faster and more precise. To this end, SciNoon provides collaborators with a shared workspace of collected articles and a chatbot that allows reporting every activity of team members. It integrates Google Scholar suggestions for related papers and queries. It also extracts from retrieved papers keywords related to the user search intent. The information is organized in the workspace according to a graph-based data model: the nodes are articles, authors, searches, and associated results; the edges express the relationship among them. The system provides a content-based recommendation tool that allows users to expand an article node based on citing and cited relationships.

SearchX [73] is a collaborative search system designed explicitly for large-scale Collaborative Search (CSE) research enabling users to implement and run their CSE experiments. According to [74, 75], we can categorize the features of a collaborative system along three lines: division of labour, sharing

of knowledge, and awareness. SearchX provides these features with a group chat for direct communication, a shared workspace, and color-coding.

### 8.2. *General-purpose Solutions*

People often perform collaborative search tasks during daily activities, e.g., choosing a restaurant or a product to buy. In such a context, group members often have to reach a consensus, which is challenging when the criteria of each member are conflicting. There is a need for communication between the members and awareness of each other's preferences. To this end, in [76], the authors propose a component that allows users to reach a consensus through the awareness of preferences, integrating it into a decision-making system. The component, called Collaborative Dynamic Queries (CDQ), acts as a moderator with three roles:

1. It provides each member with awareness of the preferences of others without the need for communication;
2. It shows candidates who match both individual and group preferences, allowing the identification of ideal candidates;
3. In case of no agreement, the moderator identifies the sources of disagreement and suggests relaxing the preferences.

The design encourages mutual awareness among members. In particular, each member has a colour associated. These colours will appear both under each filter and next to a candidate. The former allows viewing the preferences of each member, and the latter shows who may or may not agree on the choice of that candidate as the final solution. The authors conduct two user studies and show the moderator's effectiveness in maintaining awareness in the group and facilitating effective and efficient communication by reducing the effort it usually requires.

Furuie et al. [77], instead, designed a system for personal smartphones that allows comparing web pages shared among users of the same group. This function is automatically invoked differently based on the orientation of the terminal. In addition to the traditional search process involving keywords, entity-based search has been introduced over the years, where named entities replace keywords. In [78], the authors propose Querytogether, a multi-device search tool in which entities such as keywords, documents, and authors can be used to compose a query or can be shared with other members.

Considering the asynchronous scenario, looking at the search process and colleagues' results can fill the gap of a user without background; at the same time, her search process and results can help others discover new aspects. In [79], building on previous work [80, 81], the authors propose LogCanvas, a prototype of a search tool supporting asynchronous remote collaborative web search. LogCanvas is explicitly designed for search history visualization and, differently from other platforms, aims to reconstruct the semantic relationship among the users' search activities. The search process of a session is represented by a knowledge graph consisting of all queries and essential related concepts and their relationships using Yahoo's Fast Entity Linking toolkit (yahooFEL) [82]. Moreover, LogCanvas groups queries and entities according to the topic by applying a Wikipedia-based categorization method, i.e., TagTheWeb [83].

### *8.3. Solutions Based on Chatbots*

In [84], motivated by a study revealing that users use messaging channels external to the collaborative platform to accomplish their search tasks, two different chatbots are incorporated into SLACK: the first bot explicitly requests information, and the second one instead deduces this information through conversations. The study shows that chatbots improve collaborative search thanks to a greater awareness of team members' activities and a greater simplicity in communicating and sharing information and content.

SECC [85] uses a social engine that allows users to communicate across multiple channels. However, only some people may have the same information need, leading to a cold start problem as the number of people is unsuitable for a collaborative search. To overcome the problem, the same authors in [86] propose to insert an auxiliary intelligent robot in a collaborative search platform capable of supporting the communication between the system and a search engine. In this way, if there is only one user in the chat channel, she can continue to search by interacting with the search engine through the mediation of the chatbot. To this goal, the authors apply a Machine Reading Comprehension (MRC) technique to support the conversation between users and the chatbot. The system consists of a Search engine, a Cluster engine, and an Interactive engine. The Search engine provides all the functionalities of search engines, such as query expansion/suggestion and web page ranking. The Cluster engine groups users dynamically based on the keywords of the query they submit. Finally, the Interactive engine is composed of the Social engine and InfoBOT. The former provides communication channels to



different user groups to allow collaborative research. The latter collects the conversations, detects the new query, consults the search engine to collect the related documents, and modifies the query by incorporating past discussions. Finally, it passes the documents and the modified query to the MRC model that returns a response.

## 9. Open Challenges

This section identifies some open challenges that characterize Social Search tasks. We focus primarily on those where adequate social data processing can improve the effectiveness or efficiency of the Social Search system, as this is the area where we have identified the main gaps and subjects for future research. The current literature does not yet fully exploit the potential of social relationships. Some open challenges listed below can be addressed by integrating community information, while others can lead to a better representation of users and relationships.

- **Modelling Users and their Relations**

The social aspect of Social Search platforms plays a pivotal role. The studies analyzed in this survey show that social features are poorly exploited in some fields (e.g., Question Retrieval). In other areas, such as the Expert Finding task, the social dimension is widely adopted, for instance, by constructing networks to calculate users' authority. However, embedding social information is often not sophisticated and is implemented through a simple combination of features made available by social platforms. One challenge is to model users and explicit/implicit interactions within platforms in a more structured way, leveraging advanced methods proposed, for example, in the social network analysis literature. Integrating advanced interaction models could make the service more user-centric, capturing latent information about relationships. For example, in the Expert Finding task, such a more extended representation of users and their relationships could lead to identifying more efficiently and effectively experts who can provide good answers, thanks to the additional contextual information embedded in the expert finding process.

- **Cross-Linking**

The poor use of data about users and their interactions may derive from

the need for more availability of such information on a specific social platform. It is known that different social platforms offer different and complementary services, thus collecting content and information that reflect diverse aspects of the user and her social relationships. A way to improve Social Search processes is to exploit several social sources through cross-linking mechanisms of user profiles. Therefore, integrating cross-linking mechanisms would allow for a deeper behavioural and social knowledge of users that the social system can widely exploit to offer a performing service. The impact of cross-linking across different social platforms may be extremely relevant. On the one hand, it could allow Social Search tasks to have rich information about users, irrespective of how active they are on a specific social platform, by complementing sparse information available on many platforms. On the other hand, even for users with rich profiles on a given platform, cross-linking could allow us to obtain a complete profile (either individual or social) of that user, as the behaviour of a given user on different social platforms may be significantly different.

- **Explainability**

One way to make the user more aware of the data used and processed could be to explain why the system provides a specific result or suggestion. A user-friendly explanation of why the user received a particular outcome could make her more aware of the underlying mechanisms. Moreover, this would increase trust, making the users more inclined to share their data and favour personalization. In the analyzed works, the extensive use of black-box models, such as deep neural networks, further intensifies the need for explainability methods [87, 88, 89, 90, 91, 92, 93]. In this respect, social information may be used to expose different levels of information to different social groups based on the strength of social ties between users. Explainability techniques would make the platform more transparent regarding processing the user's data. At the same time, the user could feel more stimulated in using the social platform.

- **Scalability**

The pervasive use of social platforms has led to an exponential growth of content created and shared by users. Often, the methodologies proposed in state-of-the-art are computationally expensive when applied to large-scale data. Therefore, there is a need to design methodologies

that can cope with large data, guaranteeing a good trade-off between effectiveness and efficiency. From this standpoint, social information may be used to build effective filters limiting the amount of data searched to retrieve the few relevant results presented to the users based on their social relationships. The scalability property will allow the system to take advantage of the large amounts of data fully. By limiting algorithms' application to a properly identified subset of data, the system can improve responsiveness by avoiding unnecessary or redundant computations. In this case, the major impact we expect would be to reduce the noise of information used to identify good experts and good answers, with an overall improvement in the effectiveness of the search process.

- **Dynamicity**

Users are dynamic entities that continually evolve, changing their interests, behaviours, information, and social interactions. A Social Search system should address the dynamicity of data to offer an effective personalized service focused on the user. Researchers can achieve it by implementing algorithms that can continuously update the indexes with the most recent data while seeking a trade-off between computational costs and the accuracy of the results. Social information may also be used to predict the near-term evolution of social links (e.g., via advanced link prediction algorithms) and configure search processes accordingly. By monitoring the dynamicity of users and their relations, the social platform can guarantee an updated personalized service with the consequence of better user engagement.

- **Bias Mitigation**

Social Search systems must learn as much as possible about users' interests and behaviours to provide tailored results that can fulfil the user's expectations. However, the mechanism by which a system provides results based on the user information could introduce a bias creating a situation in which users are presented with homogeneous results from a limited set of similar sources [94]. In practice, however, proposing to users diversified results slightly differing from their profile could bring additional value to the service provided by expanding user vision and knowledge of a concept. The introduction and evaluation of diversification methodologies to mitigate social systems' bias have been only

partially investigated and deserve further attention. Contrasting the bias effect could foster the diffusion of new sources and sets of expertise, making the platform’s results complete and more interesting.

- **System Evaluation**

While offline evaluation is a fast and easy way to test systems, it may reflect something other than reality and is often characterized by biases. On the other hand, online evaluation allows for directly evaluating users’ degree of engagement and satisfaction. Unfortunately, it remains impractical or reserved for the owners of social platforms. Most of the studies considered in this survey are based on offline evaluation methods. Still, the research community should put additional effort into improving the assessment methodologies by providing standard benchmark datasets and using the same evaluation metrics chosen based on the specific tasks considered. Introducing several up-to-date benchmark datasets can avoid bias and offer researchers a way to compare their studies with social media platform owners as well.

- **Cold Start Problem**

The cold start problem [95, 96, 97, 98] represents one of the limitations that has always existed regarding social platforms. It occurs when more data about a specific user is needed to provide good service (e.g., for new users). The availability of rich information about the communities in the system and the scarce information about the user could mitigate the problem. By solving the cold start problem, the system can guarantee high performance to new users who will feel immediately satisfied, allowing platforms to engage users in a critical stage, i.e., when they enter the system. Cross-linking (discussed above) is a related challenge, which could also help address the cold-start problems, ultimately contributing to an easier engagement of new users.

- **Multimodality**

Given the heterogeneity of the information present within the social platforms, it would be convenient to exploit Deep Multimodal Fusion methodologies to take full advantage of the data available [99, 100, 101, 102, 103, 104, 105, 106]. In particular, with the fusion of visual media (e.g., images shared by the user), text, and relational information, the user profile could be complete and represent reality as faithfully as possible. Integrating several types of information allows for capturing

complementary aspects of the user, allowing the system to increase the performance of the personalization mechanisms.

## References

- [1] J. Teevan, D. Ramage, M. R. Morris, # twittersearch: a comparison of microblog search and web search, in: Proceedings of the fourth ACM international conference on Web search and data mining, 2011, pp. 35–44.
- [2] G. Y. Jeon, S. Y. Rieh, The value of social search: Seeking collective personal experience in social q&a, Proceedings of the American Society for Information Science and Technology 50 (1) (2013) 1–10.
- [3] M. R. Bouadjenek, H. Hacid, M. Bouzeghoub, Social networks and information retrieval, how are they converging? a survey, a taxonomy and an analysis of social information retrieval approaches and platforms, Information Systems 56 (2016) 1–18.
- [4] L. Fitzpatrick, M. Dent, Automatic feedback using past queries: social searching?, in: Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval, 1997, pp. 306–313.
- [5] B. A. Huberman, L. A. Adamic, Novelty and social search in the world wide web, CoRR cs.MA/9809025 (1998).  
URL <https://arxiv.org/abs/cs/9809025>
- [6] J. Freyne, B. Smyth, An experiment in social search, in: International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, Springer, 2004, pp. 95–103.
- [7] P. Brusilovsky, R. Farzan, J.-w. Ahn, Comprehensive personalized information access in an educational digital library, in: Proceedings of the 5th ACM/IEEE-CS joint conference on Digital libraries, 2005, pp. 9–18.
- [8] J. Freyne, R. Farzan, P. Brusilovsky, B. Smyth, M. Coyle, Collecting community wisdom: integrating social search & social navigation, in: Proceedings of the 12th international conference on Intelligent user interfaces, 2007, pp. 52–61.

- [9] S. Bao, G. Xue, X. Wu, Y. Yu, B. Fei, Z. Su, Optimizing web search using social annotations, in: Proceedings of the 16th international conference on World Wide Web, 2007, pp. 501–510.
- [10] B. M. Evans, E. H. Chi, Towards a model of understanding social search, in: Proceedings of the 2008 ACM conference on Computer supported cooperative work, 2008, pp. 485–494.
- [11] D. Carmel, N. Zwerdling, I. Guy, S. Ofek-Koifman, N. Har’El, I. Ronen, E. Uziel, S. Yogev, S. Chernov, Personalized social search based on the user’s social network, in: Proceedings of the 18th ACM conference on Information and knowledge management, 2009, pp. 1227–1236.
- [12] M. McDonnell, A. Shiri, Social search: A taxonomy of, and a user-centred approach to, social web search, Program (2011).
- [13] M. M. Kessler, Bibliographic coupling between scientific papers, American documentation 14 (1) (1963) 10–25.
- [14] X. Wang, C. Huang, L. Yao, B. Benatallah, M. Dong, A survey on expert recommendation in community question answering, Journal of Computer Science and Technology 33 (4) (2018) 625–653.
- [15] S. Yuan, Y. Zhang, J. Tang, W. Hall, J. B. Cabotà, Expert finding in community question answering: a review, Artificial Intelligence Review 53 (2) (2020) 843–874.
- [16] P. K. Roy, J. P. Singh, A. Nag, Finding active expert users for question routing in community question answering sites, in: International Conference on Machine Learning and Data Mining in Pattern Recognition, Springer, 2018, pp. 440–451.
- [17] S. Mumtaz, C. Rodriguez, B. Benatallah, Expert2vec: Experts representation in community question answering for question routing, in: International Conference on Advanced Information Systems Engineering, Springer, 2019, pp. 213–229.
- [18] D. M. Blei, A. Y. Ng, M. I. Jordan, Latent dirichlet allocation, Journal of machine Learning research 3 (Jan) (2003) 993–1022.

- [19] R. Tondulkar, M. Dubey, M. S. Desarkar, Get me the best: predicting best answerers in community question answering sites, in: Proceedings of the 12th ACM Conference on Recommender Systems, 2018, pp. 251–259.
- [20] J. Fu, Y. Li, Q. Zhang, Q. Wu, R. Ma, X. Huang, Y.-G. Jiang, Recurrent memory reasoning network for expert finding in community question answering, in: Proceedings of the 13th International Conference on Web Search and Data Mining, 2020, pp. 187–195.
- [21] K. Fukui, T. Miyazaki, M. Ohira, Suggesting questions that match each user’s expertise in community question and answering services, in: 2019 20th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), IEEE, 2019, pp. 501–506.
- [22] D. Kundu, D. P. Mandal, Formulation of a hybrid expertise retrieval system in community question answering services, Applied Intelligence 49 (2) (2019) 463–477.
- [23] J. M. Ponte, W. B. Croft, A language modeling approach to information retrieval, in: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, 1998, pp. 275–281.
- [24] Ç. Aslay, N. O’Hare, L. M. Aiello, A. Jaimes, Competition-based networks for expert finding, in: Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval, 2013, pp. 1033–1036.
- [25] X. Liu, J. Bollen, M. L. Nelson, H. Van de Sompel, Co-authorship networks in the digital library research community, Information processing & management 41 (6) (2005) 1462–1480.
- [26] L. Li, Y. Shang, W. Zhang, Improvement of hits-based algorithms on web documents, in: Proceedings of the 11th international conference on World Wide Web, 2002, pp. 527–535.
- [27] D. Kundu, R. K. Pal, D. P. Mandal, Preference enhanced hybrid expertise retrieval system in community question answering services, Decision Support Systems 129 (2020) 113164.

- [28] D. Kundu, R. K. Pal, D. P. Mandal, Finding active experts for question routing in community question answering services, in: International Conference on Pattern Recognition and Machine Intelligence, Springer, 2019, pp. 320–327.
- [29] M. Zhang, R. Song, C. Lin, S. Ma, Z. Jiang, Y. Jin, Y. Liu, L. Zhao, S. Ma, Expansion-based technologies in finding relevant and new information: Thu trec 2002: Novelty track experiments, Nist special publication sp 251 (2003) 586–590.
- [30] L. T. Le, C. Shah, Retrieving people: Identifying potential answerers in community question-answering, *Journal of the association for information science and technology* 69 (10) (2018) 1246–1258.
- [31] H. Tong, C. Faloutsos, J.-Y. Pan, Fast random walk with restart and its applications, in: Sixth international conference on data mining (ICDM'06), IEEE, 2006, pp. 613–622.
- [32] M. S. Faisal, A. Daud, A. U. Akram, R. A. Abbasi, N. R. Aljohani, I. Mehmood, Expert ranking techniques for online rated forums, *Computers in Human Behavior* 100 (2019) 168–176.
- [33] C. Fu, Tracking user-role evolution via topic modeling in community question answering, *Information Processing & Management* 56 (6) (2019) 102075.
- [34] C. Fu, User intimacy model for question recommendation in community question answering, *Knowledge-Based Systems* 188 (2020) 104844.
- [35] Y. Qian, J. Tang, K. Wu, Weakly learning to match experts in online community, in: J. Lang (Ed.), *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018*, July 13-19, 2018, Stockholm, Sweden, ijcai.org, 2018, pp. 3841–3847. doi:10.24963/ijcai.2018/534. URL <https://doi.org/10.24963/ijcai.2018/534>
- [36] J. C. Turner, P. J. Oakes, The significance of the social identity concept for social psychology with reference to individualism, interactionism and social influence, *British Journal of Social Psychology* 25 (3) (1986) 237–252.



- [37] H. Tajfel, J. C. Turner, The social identity theory of intergroup behavior. (2004).
- [38] D. Shah, T. Lei, A. Moschitti, S. Romeo, P. Nakov, Adversarial domain adaptation for duplicate question detection, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 1056–1063. doi:10.18653/v1/D18-1131.  
URL <https://www.aclweb.org/anthology/D18-1131>
- [39] D. A. Prabowo, G. B. Herwanto, Duplicate question detection in question answer website using convolutional neural network, in: 2019 5th International Conference on Science and Technology (ICST), Vol. 1, IEEE, 2019, pp. 1–6.
- [40] Z. Chen, C. Zhang, Z. Zhao, C. Yao, D. Cai, Question retrieval for community-based question answering via heterogeneous social influential network, *Neurocomputing* 285 (2018) 117–124.
- [41] N. Othman, R. Faiz, K. Smaïli, Manhattan siamese lstm for question retrieval in community question answering, in: OTM Confederated International Conferences“ On the Move to Meaningful Internet Systems”, Springer, 2019, pp. 661–677.
- [42] N. Othman, R. Faiz, K. Smaïli, Enhancing question retrieval in community question answering using word embeddings, *Procedia Computer Science* 159 (2019) 485–494.
- [43] P. Wang, L. Ji, J. Yan, D. Dou, N. D. Silva, Y. Zhang, L. Jin, Concept and attention-based cnn for question retrieval in multi-view learning, *ACM Transactions on Intelligent Systems and Technology (TIST)* 9 (4) (2018) 1–24.
- [44] J. Cheng, Z. Wang, J.-R. Wen, J. Yan, Z. Chen, Contextual text understanding in distributional semantic space, in: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, 2015, pp. 133–142.
- [45] S. Joty, L. Márquez, P. Nakov, Joint multitask learning for community question answering using task-specific embeddings, in: Proceedings of

the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 4196–4207. doi:10.18653/v1/D18-1452.  
URL <https://aclanthology.org/D18-1452>

- [46] K. P. Murphy, Machine learning: a probabilistic perspective, MIT press, 2012.
- [47] M. Zhang, Y. Wu, An unsupervised model with attention autoencoders for question retrieval, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 32, 2018.
- [48] W. E. Zhang, Q. Z. Sheng, Z. Tang, W. Ruan, Related or duplicate: distinguishing similar cqa questions via convolutional neural networks, in: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, 2018, pp. 1153–1156.
- [49] B. Perozzi, R. Al-Rfou, S. Skiena, Deepwalk: Online learning of social representations, in: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, 2014, pp. 701–710.
- [50] D. Hoogeveen, A. Bennett, Y. Li, K. Verspoor, T. Baldwin, Detecting misflagged duplicate questions in community question-answering archives, in: Proceedings of the International AAAI Conference on Web and Social Media, Vol. 12, 2018.
- [51] M. Arjovsky, S. Chintala, L. Bottou, Wasserstein generative adversarial networks, in: International conference on machine learning, PMLR, 2017, pp. 214–223.
- [52] A. Rücklé, K. Swarnkar, I. Gurevych, Improved cross-lingual question retrieval for community question answering, in: The World Wide Web Conference, 2019, pp. 3179–3186.
- [53] J. Wen, J. Ma, Y. Feng, M. Zhong, Hybrid attentive answer selection in cqa with deep users modelling, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 32, 2018.

- [54] Y. Xie, Y. Shen, Y. Li, M. Yang, K. Lei, Attentive user-engaged adversarial neural network for community question answering, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 9322–9329.
- [55] J. Hu, S. Qian, Q. Fang, C. Xu, Attentive interactive convolutional matching for community question answering in social multimedia, in: Proceedings of the 26th ACM international conference on Multimedia, 2018, pp. 456–464.
- [56] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
- [57] Y. Liu, M. Liu, J. Yin, Toward community answer selection by jointly static and dynamic user expertise modeling, *APSIPA Transactions on Signal and Information Processing* 10 (2021).
- [58] W. Zhang, Z. Chen, C. Dong, W. Wang, H. Zha, J. Wang, Graph-based tri-attention network for answer ranking in cqa, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 14463–14471.
- [59] F. Albanese, E. Feuerstein, Improved topic modeling in twitter through community pooling, in: International Symposium on String Processing and Information Retrieval, Springer, 2021, pp. 209–216.
- [60] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, *Journal of statistical mechanics: theory and experiment* 2008 (10) (2008) P10008.
- [61] B. Xu, H. Lin, Y. Lin, Y. Guan, Integrating social annotations into topic models for personalized document retrieval, *Soft Computing* 24 (3) (2020) 1707–1716.
- [62] Y. Xia, Y. Sun, T. Wang, J. Caicedo Carvajal, J. Fan, B. Mangipudi, L. Huang, Y. Saraf, Relevance ranking for real-time tweet search, in: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, 2020, pp. 2829–2836.

- [63] J.-T. Huang, A. Sharma, S. Sun, L. Xia, D. Zhang, P. Pronin, J. Padmanabhan, G. Ottaviano, L. Yang, Embedding-based retrieval in facebook search, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020, pp. 2553–2561.
- [64] M. R. Bouadjenek, H. Hacid, M. Bouzeghoub, Personalized social query expansion using social annotations, in: Transactions on Large-Scale Data-and Knowledge-Centered Systems XL, Springer, 2019, pp. 1–25.
- [65] H. Khalifi, S. Dahir, A. El Qadi, Y. Ghanou, Enhancing information retrieval performance by using social analysis, Social Network Analysis and Mining 10 (1) (2020) 1–7.
- [66] J. Zhong, W. Guo, H. Gao, B. Long, Personalized query suggestions, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2020, pp. 1645–1648.
- [67] W. Chen, F. Cai, H. Chen, M. de Rijke, Attention-based hierarchical neural query suggestion, in: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, 2018, pp. 1093–1096.
- [68] M. Kazi, W. Guo, H. Gao, B. Long, Incorporating user feedback into sequence to sequence model training, in: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, 2020, pp. 2557–2564.
- [69] A. Jaech, M. Ostendorf, Personalized language model for query auto-completion, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2018, pp. 700–705.
- [70] G. Golovchinsky, J. Pickens, M. Back, A taxonomy of collaboration in online information seeking, in: Proceedings of the 1st International Workshop on Collaborative Information Retrieval, 2008.
- [71] M. R. Morris, Collaborative search revisited, in: Proceedings of the 2013 conference on Computer supported cooperative work, 2013, pp. 1181–1192.

- [72] Y. Nedumov, A. Babichev, I. Mashonsky, N. Semina, Scinoon: Exploratory search system for scientific groups., in: IUI Workshops, 2019.
- [73] S. R. Putra, K. Grashoff, F. Moraes, C. Hauff, On the development of a collaborative search system., in: DESIRES, 2018, pp. 76–82.
- [74] S. E. Brennan, X. Chen, C. A. Dickinson, M. B. Neider, G. J. Zelinsky, Coordinating cognition: The costs and benefits of shared gaze during collaborative search, *Cognition* 106 (3) (2008) 1465–1477.
- [75] C. I. Maican, A.-M. Cazan, R. C. Lixandriou, L. Dovleac, A study on academic staff personality and technology acceptance: The case of communication and collaboration applications, *Computers & Education* 128 (2019) 113–131.
- [76] S. Hong, M. Suh, N. Henry Riche, J. Lee, J. Kim, M. Zachry, Collaborative dynamic queries: Supporting distributed small group decision-making, in: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 2018, pp. 1–12.
- [77] N. Furuie, T. Yabuuchi, T. Yamamoto, H. Takada, Showing-displays-together: A co-located web search support exploiting the terminal orientation, in: *International Conference on Collaboration Technologies and Social Computing*, Springer, 2020, pp. 167–174.
- [78] S. Andolina, K. Klouche, T. Ruotsalo, P. Floréen, G. Jacucci, Query-together: Enabling entity-centric exploration in multi-device collaborative search, *Information Processing & Management* 54 (6) (2018) 1182–1202.
- [79] L. Xu, T. Tolmochava, X. Zhou, Search history visualization for collaborative web searching, *Big Data Research* 23 (2021) 100180.
- [80] L. Xu, Z. T. Fernando, X. Zhou, W. Nejdl, Logcanvas: visualizing search history using knowledge graphs, in: *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, 2018, pp. 1289–1292.
- [81] T. Tolmachova, L. Xu, I. Marenzi, U. Gadiraju, Visualizing search history in web learning, in: *International Conference on Web-Based Learning*, Springer, 2019, pp. 229–240.

- [82] R. Blanco, G. Ottaviano, E. Meij, Fast and space-efficient entity linking for queries, in: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, 2015, pp. 179–188.
- [83] J. F. Medeiros, B. P. Nunes, S. W. M. Siqueira, L. A. P. P. Leme, Tagtheweb: Using wikipedia categories to automatically categorize resources on the web, in: European Semantic Web Conference, Springer, 2018, pp. 153–157.
- [84] S. Avula, Searchbots: Using chatbots in collaborative information-seeking tasks, in: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2017, pp. 1375–1375.
- [85] C. Zhang, P. Zhang, J. Li, D. Song, Secc: A novel search engine interface with live chat channel, in: Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, 2016, pp. 1137–1140.
- [86] J. Wang, P. Zhang, C. Zhang, D. Song, Scss-lie: A novel synchronous collaborative search system with a live interactive engine, in: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019, pp. 1309–1312.
- [87] J. Singh, A. Anand, Exs: Explainable search using local model agnostic interpretability, in: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, 2019, pp. 770–773.
- [88] Z. T. Fernando, J. Singh, A. Anand, A study on the interpretability of neural retrieval models using deepshap, in: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019, pp. 1005–1008.
- [89] Y. Zhang, X. Chen, et al., Explainable recommendation: A survey and new perspectives, Foundations and Trends® in Information Retrieval 14 (1) (2020) 1–101.
- [90] J. Singh, A. Anand, Model agnostic interpretability of rankers via intent modelling, in: Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, 2020, pp. 618–628.

- [91] Z. Zhang, K. Rudra, A. Anand, Explain and predict, and then predict again, in: Proceedings of the 14th ACM International Conference on Web Search and Data Mining, 2021, pp. 418–426.
- [92] H. Zhuang, X. Wang, M. Bendersky, A. Grushetsky, Y. Wu, P. Mitrichev, E. Sterling, N. Bell, W. Ravina, H. Qian, Interpretable ranking with generalized additive models, in: Proceedings of the 14th ACM International Conference on Web Search and Data Mining, 2021, pp. 499–507.
- [93] M. Völske, A. Bondarenko, M. Fröbe, B. Stein, J. Singh, M. Hagen, A. Anand, Towards axiomatic explanations for neural ranking models, in: Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval, 2021, pp. 13–22.
- [94] J. Chen, H. Dong, X. Wang, F. Feng, M. Wang, X. He, Bias and debias in recommender system: A survey and future directions, ACM Transactions on Information Systems 41 (3) (2023) 1–39.
- [95] Y. Lu, Y. Fang, C. Shi, Meta-learning on heterogeneous information networks for cold-start recommendation, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020, pp. 1563–1573.
- [96] Y. Wei, X. Wang, Q. Li, L. Nie, Y. Li, X. Li, T.-S. Chua, Contrastive learning for cold-start recommendation, in: Proceedings of the 29th ACM International Conference on Multimedia, 2021, pp. 5382–5390.
- [97] M. Dong, F. Yuan, L. Yao, X. Xu, L. Zhu, Mamo: Memory-augmented meta-optimization for cold-start recommendation, in: Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining, 2020, pp. 688–697.
- [98] D. Cai, S. Qian, Q. Fang, J. Hu, C. Xu, User cold-start recommendation via inductive heterogeneous graph neural network, ACM Transactions on Information Systems 41 (3) (2023) 1–27.
- [99] D. Lu, L. Neves, V. Carvalho, N. Zhang, H. Ji, Visual attention model for name tagging in multimodal social media, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2018, pp. 1990–1999.

- [100] X. Chen, N. Zhang, L. Li, S. Deng, C. Tan, C. Xu, F. Huang, L. Si, H. Chen, Hybrid transformer with multi-level fusion for multimodal knowledge graph completion, in: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2022, pp. 904–915.
- [101] P. Hu, L. Zhen, D. Peng, P. Liu, Scalable deep multimodal learning for cross-modal retrieval, in: Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval, 2019, pp. 635–644.
- [102] J. Yu, J. Jiang, L. Yang, R. Xia, Improving multimodal named entity recognition via entity span detection with unified multimodal transformer, Association for Computational Linguistics, 2020.
- [103] Y. Cui, Z. Yu, C. Wang, Z. Zhao, J. Zhang, M. Wang, J. Yu, Rosita: Enhancing vision-and-language semantic alignments via cross-and intra-modal knowledge integration, in: Proceedings of the 29th ACM International Conference on Multimedia, 2021, pp. 797–806.
- [104] X. Zhu, Z. Li, X. Wang, X. Jiang, P. Sun, X. Wang, Y. Xiao, N. J. Yuan, Multi-modal knowledge graph construction and application: A survey, IEEE Transactions on Knowledge and Data Engineering (2022).
- [105] C. Zheng, J. Feng, Z. Fu, Y. Cai, Q. Li, T. Wang, Multimodal relation extraction with efficient graph alignment, in: Proceedings of the 29th ACM International Conference on Multimedia, 2021, pp. 5298–5306.
- [106] L. Sun, J. Wang, Y. Su, F. Weng, Y. Sun, Z. Zheng, Y. Chen, Riva: a pre-trained tweet multimodal model based on text-image relation for multimodal ner, in: Proceedings of the 28th International Conference on Computational Linguistics, 2020, pp. 1852–1862.